Guía del usuario

AWS Systems Manager



Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS Systems Manager: Guía del usuario

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Las marcas comerciales y la imagen comercial de Amazon no se pueden utilizar en relación con ningún producto o servicio que no sea de Amazon, de ninguna manera que pueda causar confusión entre los clientes y que menosprecie o desacredite a Amazon. Todas las demás marcas registradas que no son propiedad de Amazon son propiedad de sus respectivos propietarios, que pueden o no estar afiliados, conectados o patrocinados por Amazon.

Table of Contents

¿Qué es AWS Systems Manager? 1
¿Cómo puede Systems Manager beneficiar las operaciones de mi organización? 2
¿Cómo se debe usar Systems Manager? 2
¿Cuáles son las principales características de Systems Manager? 3
Regiones de AWS admitidas 4
Cómo acceder a Systems Manager 4
Historial de nombres de servicio de Systems Manager 6
Sistemas operativos y tipos de equipos compatibles 6
Sistemas operativos compatibles con Systems Manager 6
Tipos de equipos compatibles con entornos híbridos y multinube 14
¿Qué es la consola unificada? 14
Configuración de nodos administrados para AWS Systems Manager 17
Administrar instancias de EC2 con Systems Manager 17
Configuración de permisos de instancia requeridos para Systems Manager 18
Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager 30
Administrador de nodos en entornos híbridos y multinube con Systems Manager 35
Creación del rol de servicio de IAM requerido para Systems Manager en entornos híbridos y multinube 37
Creación de una activación híbrida para registrar nodos con Systems Manager 46
Instalar SSM Agent en nodos de Linux híbridos 53
Instalación de SSM Agent en nodos híbridos de Windows Server 60
Administración de dispositivos periféricos con Systems Manager 64
Creación de un rol de servicio de IAM para dispositivos periféricos 66
Configuración de dispositivos periféricos para AWS IoT Greengrass 72
Actualización del rol de intercambio de tokens de AWS IoT Greengrass e instalación de SSM Agent en dispositivos periféricos 73
Creación de un administrador delegado de AWS Organizations para Systems Manager 73
Uso de un administrador delegado con Change Manager 74
Uso de un administrador delegado con Explorer 75
Uso de un administrador delegado con OpsCenter 75
Uso de un administrador delegado con Quick Setup 75
Configuración general 76
Cómo crear una Cuenta de AWS 76

Creación de un usuario con acceso administrativo ... 76
Configuración de AWS Systems Manager ... 79
Configuración del acceso a la consola de Systems Manager ... 79
Política de incorporación de Systems Manager ... 79
Política de operadores de la consola de AWS Systems Manager ... 85
Política de solo lectura para operadores de la consola de AWS Systems Manager ... 89
Configuración de la consola unificada de Systems Manager para una organización ... 92
Requisitos previos ... 92
Recursos de la consola unificada ... 93
Configurar la consola unificada ... 95
Configuración de la consola unificada de Systems Manager para una sola cuenta y región ... 98
Recursos de la consola unificada ... 98
Configurar la consola unificada ... 100
Deshabilitación de la consola unificada de Systems Manager ... 101
Ejecución de tareas de administración de nodos ... 103
Revisión de la información sobre los nodos ... 103
Adición o eliminación de widgets ... 106
Reorganización de widgets ... 107
Exploración de nodos ... 108
Exploración de nodos mediante filtros de consola ... 109
Exploración de nodos mediante peticiones de texto en Amazon Q ... 114
Visualización de los detalles de los nodos individuales y realización de acciones en un nodo ... 120
Descarga o exportación de un informe de nodos administrados ... 122
Administración del contenido y de la apariencia de los informes de nodos ... 126
Acceso a los nodos justo a tiempo mediante Systems Manager ... 127
Cómo configurar acceso justo a tiempo con Systems Manager ... 129
Cómo iniciar una sesión de acceso a los nodos justo a tiempo ... 179
Cómo administrar las solicitudes de acceso justo a tiempo ... 181
Cambio al acceso a los nodos justo a tiempo desde Session Manager ... 183
Cómo deshabilitar el acceso justo a tiempo con Systems Manager ... 188
Preguntas frecuentes ... 188
Diagnóstico y corrección ... 190
Diagnóstico y corrección de implementaciones con error ... 191
Diagnóstico y corrección de configuraciones desviadas ... 192
Diagnóstico y corrección de instancias de Amazon EC2 no administradas ... 193

Tipos de impacto de la corrección de las acciones del manual de procedimientos 201
Visualización de los detalles del historial de ejecución para las correcciones 204
Ajuste de la configuración de Systems Manager 205
Ajustes de la configuración de cuenta 205
Ajustes de la configuración organizativa 205
Configuraciones de características 206
Configuración de diagnóstico y corrección 210
Uso de buckets de Amazon S3 y políticas de buckets para Systems Manager 212
Cómo utilizar herramientas de AWS Systems Manager 221
Herramientas de nodos 221
Herramientas de administración de cambios 223
Herramientas de aplicaciones 225
Herramientas de operaciones 226
Realización de una tarea de nodos con Systems Manager 227
Requisitos previos 227
Lanzar una instancia mediante el uso de una AMI con SSM Agent preinstalado 227
Conexión a una instancia administrada mediante Systems Manager 229
Elimine la instancia 229
Herramientas de nodos 230
Conformidad 230
Distributor 251
Fleet Manager 307
Activaciones híbridas 398
Inventario 399
Patch Manager 499
Run Command 839
Session Manager 909
State Manager 1066
Herramientas de administración de cambios 1168
Automation 1169
Change Calendar 1658
Change Manager 1678
Documentos 1769
Maintenance Windows 1918
Quick Setup 2055
Herramientas de aplicaciones 225

AWS AppConfig ... 2108
Application Manager ... 2108
Parameter Store ... 2157
Herramientas de operaciones ... 2318
Administrador de incidentes de ... 2319
Explorer ... 2319
OpsCenter ... 2361
Paneles de CloudWatch ... 2451
Uso de SSM Agent ... 2453
Información sobre detalles técnicos acerca de SSM Agent ... 2454
Comportamiento de las credenciales de la versión 3.2.x.x de SSM Agent ... 2454
Prioridad de credenciales de SSM Agent ... 2454
Configuración de SSM Agent para su uso con el estándar federal de procesamiento de información (FIPS) ... 2456
Acerca de la cuenta ssm-user local ... 2458
SSM Agent y Instance Metadata Service (IMDS) ... 2459
Mantener el SSM Agent actualizado ... 2459
Ratificación de que el directorio de instalación de SSM Agent no se modifique, mueva o elimine ... 2460
Actualizaciones continuas de SSM Agent por Regiones de AWS ... 2460
Comunicaciones de SSM Agent con buckets de S3 administrados de AWS ... 2461
Búsqueda de AMIs con SSM Agent preinstalado ... 2469
Verificación del estado de SSM Agent ... 2471
Uso de SSM Agent en instancias de EC2 para Linux ... 2475
Verificación de la firma de SSM Agent ... 2475
Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux ... 2491
Configuración de SSM Agent para utilizar un proxy en nodos de Linux ... 2551
Cómo utilizar SSM Agent en instancias de EC2 para macOS ... 2556
Instalación y desinstalación manual de SSM Agent en instancias de EC2 para macOS ... 2557
Cómo utilizar SSM Agent en instancias de EC2 para Windows Server ... 2559
Cómo instalar y desinstalar de forma manual SSM Agent en instancias de EC2 para Windows Server ... 2560
Configurar el SSM Agent para usar un proxy para las instancias de Windows Server ... 2563
Verificación del estado de SSM Agent e inicio del agente ... 2567
Verificación del número de versión de SSM Agent ... 2569
Visualización de registros de SSM Agent ... 2574

Permiso del registro de depuración de SSM Agent ... 2575
Restricción del acceso a los comandos de nivel raíz con SSM Agent ... 2577
Automatización de las actualizaciones de SSM Agent ... 2578
Actualización automática de SSM Agent ... 2580
Suscripción a las notificaciones de SSM Agent ... 2582
Solución de problemas de SSM Agent ... 2583
SSM Agent está desactualizado ... 2583
Solucionar problemas con los archivos de registro de SSM Agent ... 2584
Los archivos de registros del agente no rotan (Windows) ... 2584
No es posible conectarse a los puntos de enlace de SSM ... 2585
Verificación de la configuración VPC ... 2586
Compruebe los atributos relacionados con el DNS de la VPC ... 2587
Verificación de las reglas de entrada en los grupos de seguridad de punto de conexión ... 2588
Utilice `ssm-cli` para solucionar los problemas de disponibilidad de los nodos administrados ... 2589
Seguridad ... 2590
Protección de los datos ... 2591
Cifrado de datos ... 2592
Privacidad del tráfico entre redes ... 2595
Identity and Access Management ... 2595
Público ... 2596
Autenticación con identidades ... 2596
Administración de acceso mediante políticas ... 2600
Cómo funciona AWS Systems Manager con IAM ... 2604
Ejemplos de políticas basadas en identidad ... 2615
Políticas administradas de AWS ... 2627
Resolución de problemas ... 2691
Cómo utilizar roles vinculados a servicios ... 2693
Rol de datos de Inventory y Explorer ... 2694
Rol de detección de cuentas de OpsCenter y Explorer ... 2697
Rol de creación de OpsData y OpsItems ... 2701
Rol de creación de información operativa ... 2705
Rol de comprobación de estado de la implementación de Quick Setup ... 2708
Exportar el rol de servicio de OpsData ... 2711
Rol de servicio de acceso a los nodos justo a tiempo ... 2714
Rol de servicio para notificaciones de solicitudes de acceso a los nodos justo a tiempo ... 2718

Registro y monitoreo 2721
Validación de conformidad 2724
Resiliencia 2725
Seguridad de la infraestructura 2726
Configuración y análisis de vulnerabilidades 2726
Prácticas recomendadas de seguridad 2726
Prácticas recomendadas preventivas de seguridad de Systems Manager 2727
Prácticas recomendadas de monitorización y auditoría de Systems Manager 2731
Ejemplos de código 2734
Conceptos básicos 2742
Hola, Systems Manager 2745
Conceptos básicos 2749
Acciones 2801
Registro y monitoreo 3145
Herramientas de supervisión 3146
Envío de registros de nodos a los Registros de CloudWatch (agente de CloudWatch) unificado 3146
Migrar la recopilación de registros del nodo de Windows Server al agente de CloudWatch 3148
Almacenar la configuración del agente de CloudWatch en Parameter Store 3159
Revertir a la recopilación de registros con SSM Agent 3160
Envío de registros de SSM Agent a CloudWatch Logs 3164
Supervisión de los eventos de las solicitudes de cambio 3167
Monitoreo de las automatizaciones 3170
Métricas de Automation 3170
Monitoreo de métricas de Run Command con Amazon CloudWatch 3171
Métricas y dimensiones de Systems Manager Run Command 3172
Registro de llamadas a la API de AWS Systems Manager con AWS CloudTrail 3173
Eventos de datos de Systems Manager en CloudTrail 3175
Eventos de administración de Systems Manager en CloudTrail 3177
Ejemplos de eventos de Systems Manager 3177
Registro de salida de acción de Automation con CloudWatch Logs 3183
Configuración de Registros de Amazon CloudWatch para Run Command 3187
Especificación de Registros de CloudWatch al momento de enviar comandos 3188
Visualización de la salida de comandos en Registros de CloudWatch 3189
Monitoreo con Amazon EventBridge 3189
Configuración de EventBridge para eventos de Systems Manager 3191

Ejemplos de eventos de Amazon EventBridge para Systems Manager 3195
Escenarios de ejemplo: destinos de Systems Manager en reglas de Amazon EventBridge 3211
Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS 3213
Cómo configurar notificaciones de Amazon SNS para AWS Systems Manager 3214
Ejemplo de notificaciones de Amazon SNS para AWS Systems Manager 3226
Uso de Run Command para enviar un comando que devuelve notificaciones de estado 3227
Uso de un periodo de mantenimiento para enviar un comando que devuelve notificaciones de estado 3231
Integraciones de productos y servicios 3237
Cómo integrar con Servicios de AWS 3237
Computación 3237
Internet de las cosas (IoT) 3240
Almacenamiento 3241
Herramientas para desarrolladores 3242
Seguridad, identidad y conformidad 3243
Criptografía y PKI 3246
Administración y gobierno 3247
Redes y entrega de contenido 3253
Análisis 3253
Integración de aplicaciones 3255
AWS Management Console 3256
Ejecución de scripts desde Amazon S3 3257
Referencia a los secretos de AWS Secrets Manager desde los parámetros de Parameter Store 3261
Cifrado de AWS KMS para los parámetros SecureString del Parameter Store 3268
Uso de secretos de AWS Secrets Manager en Amazon Elastic Kubernetes Service 3282
Cómo utilizar parámetros Parameter Store en funciones AWS Lambda 3302
Integración a otros productos y servicios 3327
Ejecución de scripts desde GitHub 3331
Utilización de perfiles de Chef InSpec con la conformidad de Systems Manager 3339
Integración con ServiceNow 3345
Referencia de AWS Systems Manager 3347
Cómo utilizar los AWS SDK 3348
Buckets de Amazon S3 para operaciones de aplicación de revisiones 3349

Buckets que contienen documentos de comandos SSM para operaciones de aplicación de revisiones (Linux y Windows Server) ... 3350
Buckets que contienen documentos de comandos SSM para operaciones de aplicación de revisiones (macOS) ... 3353
Buckets que contienen instantáneas de la línea de base de revisiones gestionadas de AWS ... 3356
Patrones y tipos de eventos de Amazon EventBridge para Systems Manager ... 3359
Tipo de evento: Automation ... 3360
Tipo de evento: Change Calendar ... 3361
Tipo de evento: Change Manager ... 3362
Tipo de evento: conformidad de la configuración ... 3362
Tipo de evento: Inventory ... 3362
Tipo de evento: periodo de mantenimiento ... 3363
Tipo de evento: OpsCenter ... 3367
Tipo de evento: Parameter Store ... 3367
Tipo de evento: Run Command ... 3368
Tipo de evento: State Manager ... 3369
Expresiones cron y rate ... 3370
Información general sobre las expresiones cron y rate ... 3370
Expresiones cron y rate para asociaciones ... 3376
Expresiones cron y rate para los períodos de mantenimiento ... 3379
ec2messages, ssmmessages y otras operaciones de la API ... 3381
Operaciones de la API relacionadas con el agente (puntos de conexión de `ssmmessages` y `ec2messages`) ... 3382
Operaciones de la API relacionadas con las instancias del espacio de nombres `ssm:*` ... 3384
ssm:* espacio de nombres otras operaciones de la API ... 3385
Formatos de cadenas de fecha y hora para Systems Manager ... 3386
Dar formato a cadenas de fecha y hora para Systems Manager ... 3386
Crear cadenas de fecha y hora personalizadas para Systems Manager ... 3387
Casos de uso y prácticas recomendadas ... 3390
Eliminación de recursos y artefactos de Systems Manager ... 3393
Elección entre State Manager y Maintenance Windows ... 3398
State Manager y Maintenance Windows: casos de uso clave ... 3398
Información relacionada ... 3407
Historial de documentos ... 3409
Actualizaciones anteriores a junio de 2018 ... 3652

Convenciones del documento 3674

¿Qué es AWS Systems Manager?

AWS Systems Manager le permite visualizar, administrar y operar los nodos de forma centralizada a escala en entornos de AWS, en las instalaciones y multinube. Con el lanzamiento de una experiencia de consola unificada, Systems Manager consolida varias herramientas para ayudarlo a completar tareas de nodos comunes en todas las Cuentas de AWS y Regiones de AWS.

Para usar Systems Manager, los nodos deben ser administrados, lo que significa que SSM Agent está instalado en la máquina y el agente se puede comunicar con el servicio de Systems Manager. Para ayudarle a identificar por qué los nodos no se notifican como administrados, Systems Manager ofrece, mediante un solo clic, un manual de procedimientos de diagnóstico y solución de problemas del agente que puede configurar para que se ejecute automáticamente según la programación que defina. Esta característica ayuda a identificar los motivos por los que los nodos no se pueden conectar a Systems Manager, incluidos los errores de configuración de la red. Esta característica también proporciona manuales de procedimientos recomendados para solucionar los problemas de red y otros errores que impiden que los nodos se configuren como nodos administrados.

La experiencia de consola unificada también incluye un panel de control que brinda información general de alto nivel sobre los nodos. Puede profundizar para obtener información más específica sobre los nodos, por ejemplo, qué nodos utilizan software de sistema operativo (SO) obsoleto. También puede usar filtros para obtener vistas granulares basadas en los metadatos de instancias, como los sistemas operativos y sus versiones, las versiones Regiones de AWS, Cuentas de AWS y SSM Agent. Estos filtros le ayudan a recuperar información relevante y específica sobre el nivel de la cuenta o aplicación en toda la organización.

Temas

- ¿Cómo puede Systems Manager beneficiar las operaciones de mi organización?
- ¿Cómo se debe usar Systems Manager?
- ¿Cuáles son las principales características de Systems Manager?
- Regiones de AWS admitidas
- Cómo acceder a Systems Manager
- Historial de nombres de servicio de Systems Manager
- Sistemas operativos y tipos de equipos compatibles
- ¿Qué es la consola unificada?

¿Cómo puede Systems Manager beneficiar las operaciones de mi organización?

Entre los beneficios de Systems Manager se incluyen los siguientes:

- Mejora de la visibilidad en toda su infraestructura

  Systems Manager proporciona una vista centralizada de los nodos de las cuentas y regiones de su organización. Acceda rápidamente a la información de la instancia, como el ID, el nombre, los detalles del sistema operativo y los agentes instalados. Utilice Amazon Q Developer para consultar los metadatos de las instancias mediante un lenguaje natural, lo que le ayudará a identificar los problemas y a tomar medidas con mayor rapidez.
- Aumento de la eficiencia operativa con la automatización

  Automatice las tareas operativas comunes y reduzca el tiempo y el esfuerzo necesarios para mantener los sistemas. Systems Manager proporciona una administración remota segura y protegida de los nodos, a escala y sin necesidad de iniciar sesión en sus servidores. Ya no necesita usar hosts bastión, SSH o PowerShell remoto. Systems Manager también ofrece una manera sencilla de automatizar tareas administrativas comunes en grupos de nodos, como ediciones en registros, administración de usuarios e instalaciones de software y revisiones.
- Simplificación de la administración de nodos a escala en cualquier entorno

  Systems Manager le ayuda a administrar los nodos en entornos de AWS, en las instalaciones y multinube. Programe diagnósticos automatizados para identificar los problemas de SSM Agent y solucionarlos con manuales de procedimientos mediante un solo clic. Una vez que los nodos se configuren como nodos administrados, podrá ejecutar tareas operativas críticas, como aplicar revisiones de seguridad, iniciar sesiones registradas y ejecutar comandos de forma remota.

¿Cómo se debe usar Systems Manager?

Systems Manager lo utilizan los administradores y operadores de operaciones de TI, los ingenieros de DevOps, los administradores de seguridad y cumplimiento y los directores de TI y CIO. En términos generales, Systems Manager es adecuado para las siguientes organizaciones:

- Organizaciones que desean mejorar la administración y la seguridad de sus nodos a escala.
- Organizaciones que desean aumentar la visibilidad y la agilidad operativa a la hora de administrar la infraestructura.

- Organizaciones que desean aumentar la eficiencia operativa a escala.

¿Cuáles son las principales características de Systems Manager?

Las características principales de Systems Manager se comparten entre la consola unificada y las herramientas individuales que proporciona Systems Manager para ayudarlo a administrar los nodos a escala.

Consola unificada

La consola unificada proporciona una experiencia centralizada para ver y administrar los nodos. Esta consola utiliza varias herramientas de Systems Manager y más para ofrecerle lo siguiente:

- Vistas centralizadas de los nodos
- Información detallada sobre nodos
- Diagnóstico y corrección automatizados de problemas comunes en los nodos

Para obtener más información sobre la consola unificada, consulte ¿Qué es la consola unificada?.

Herramientas

Las herramientas constan de las capacidades individuales de Systems Manager y sus características, como Run Command, Session Manager, Automatización y Parameter Store. Puede hacer lo siguiente con las herramientas de Systems Manager:

- Aplicar revisiones a los nodos a escala
- Conectarse de forma segura a los nodos sin abrir los puertos de entrada
- Ejecutar comandos de forma remota en los nodos
- Almacenar de forma segura los datos referenciados por las aplicaciones
- Automatizar tareas comunes de administración de sistemas

Para obtener más información acerca de las herramientas de Systems Manager, consulte Cómo utilizar herramientas de AWS Systems Manager.

Regiones de AWS admitidas

Para obtener una lista de Regiones de AWS compatibles con las herramientas de Systems Manager, consulte Systems Manager service endpoints en la Referencia general de Amazon Web Services.

La consola unificada de Systems Manager, lanzada el 21 de noviembre de 2024, está disponible en las siguientes Regiones de AWS:

- Región Este de EE. UU. (Norte de Virginia)
- Región del este de EE. UU. (Ohio)
- Región del oeste de EE. UU. (Norte de California)
- Región del Oeste de EE. UU (Oregón)
- Región de Canadá (centro)
- Región de América del Sur (São Paulo)
- Región de Asia-Pacífico (Bombay)
- Asia Pacífico (Tokio)
- Región de Asia-Pacífico (Seúl)
- Región de Asia-Pacífico (Singapur)
- Región de Asia-Pacífico (Sídney)
- Región Europa (Fráncfort)
- Región Europa (Estocolmo)
- Región de Europa (Irlanda)
- Región de Europa (Londres)
- Región Europa (París)

Cómo acceder a Systems Manager

Puede trabajar con Systems Manager de cualquiera de las siguientes formas:

Consola de Systems Manager

> La consola de Systems Manager es una interfaz basada en navegador para acceder a Systems Manager y utilizarlo.

Consola de AWS IoT Greengrass V2

Puede ver y administrar dispositivos de borde configurados por AWS IoT Greengrass en la Consola Greengrass.

Herramientas de línea de comando de AWS

Con el uso de las herramientas de línea de comandos de AWS, puede emitir comandos en la línea de comandos de su sistema para llevar a cabo tareas de Systems Manager y de AWS. Las herramientas son compatibles con Linux, macOS y Windows. El uso de la AWS Command Line Interface (AWS CLI) puede ser más rápido y práctico que el uso de la consola. Las herramientas de línea de comandos también son útiles para crear scripts que realicen tareas de AWS.

AWS proporciona dos conjuntos de herramientas de línea de comandos: la AWS Command Line Interface y las AWS Tools for Windows PowerShell. Para obtener información acerca de la instalación y el uso de la AWS CLI, consulte la Guía del usuario de AWS Command Line Interface. Para obtener información acerca de la instalación y el uso de Tools for Windows PowerShell, consulte la Guía del usuario de Herramientas de AWS para PowerShell.

> **Note**
>
> En las instancias de Windows Server, Windows PowerShell 3.0 o una versión posterior, se necesita ejecutar determinados documentos de SSM (por ejemplo, el documento `AWS-ApplyPatchBaseline` heredado). Compruebe que las instancias de Windows Server estén ejecutando Windows Management Framework 3.0 o una versión posterior. El marco incluye Windows PowerShell.

AWS SDK

AWS proporciona kits de desarrollo de software (SDK) que se componen de bibliotecas y código de muestra para diversos lenguajes de programación y plataformas (por ejemplo, Java, Python, Ruby, .NET, iOS y Android, y otros). Los SDK brindan una manera conveniente de otorgar acceso a Systems Manager mediante programación. Para obtener información sobre los SDK de AWS (por ejemplo, cómo descargarlos e instalarlos), consulte Herramientas para Amazon Web Services.

Historial de nombres de servicio de Systems Manager

AWS Systems Manager (Systems Manager) antes era conocido como Amazon Simple Systems Manager (SSM) y Amazon EC2 Systems Manager (SSM). El nombre abreviado original del servicio, SSM, sigue apareciendo en diferentes recursos de AWS, incluidas algunas consolas de servicio. Presentamos algunos ejemplos:

- Systems Manager Agent: SSM Agent
- Parámetros de Systems Manager: parámetros de SSM
- Puntos de enlace de servicio de Systems Manager: `ssm.region.amazonaws.com`
- Tipos de recurso de AWS CloudFormation: `AWS::SSM::Document`
- Identificador de reglas de AWS Config: `EC2_INSTANCE_MANAGED_BY_SSM`
- Comando de AWS Command Line Interface (AWS CLI): `aws ssm describe-patch-baselines`
- Nombres de políticas administradas de AWS Identity and Access Management (IAM): `AmazonSSMReadOnlyAccess`
- ARN de recursos de Systems Manager: `arn:aws:ssm:region:account-id:patchbaseline/pb-07d8884178EXAMPLE`

Sistemas operativos y tipos de equipos compatibles

Antes de trabajar con Systems Manager, verifique que el sistema operativo (SO), la versión del sistema operativo y el tipo de equipo sean compatibles como nodos administrados.

Temas

- Sistemas operativos compatibles con Systems Manager
- Tipos de equipos compatibles con entornos híbridos y multinube

Sistemas operativos compatibles con Systems Manager

En las siguientes secciones se enumeran los sistemas operativos y las versiones de sistemas operativos compatibles con Systems Manager.

> **Note**
>
> Si planea administrar y configurar dispositivos de núcleo de AWS IoT Greengrass mediante Systems Manager, estos dispositivos deben cumplir los requisitos de AWS IoT Greengrass. Para obtener más información, consulte Configuración de dispositivos de núcleo de AWS IoT Greengrass en la Guía para desarrolladores de AWS IoT Greengrass Version 2.
> Si planea administrar y configurar dispositivos de borde de AWS IoT y que no son de AWS, estos dispositivos deben cumplir los requisitos enumerados aquí y se deben configurar como nodos administrados locales para Systems Manager. Para obtener más información, consulte Administración de dispositivos periféricos con Systems Manager.

> **Important**
>
> Patch Manager, una herramienta de Systems Manager, podría no ser compatible con todas las versiones del SO que se enumeran en este tema. Para ver una lista de las versiones del sistema operativo compatibles con Patch Manager, consulte Requisitos previos de Patch Manager.

Tipos de sistemas operativos

- Linux
- macOS (solo instancias de Amazon EC2)
- Raspberry Pi OS (anteriormente Raspbian)
- Windows Server

Seleccione una plataforma de sistema operativo para ver las versiones principales y secundarias compatibles.

Linux

AlmaLinux

Versiones	x86	x86_64	ARM64
8.3 a 8.10		✓	✓

Versiones	x86	x86_64	ARM64
9.x		✓	✓

Amazon Linux 1

Versiones	x86	x86_64	ARM64
2012.03–2018.03	✓	✓	

> **Note**
>
> A partir de la versión 2015.03, se lanzó Amazon Linux 1 en versiones x86_64. Amazon Linux 1 finalizó su soporte estándar el 31 de diciembre de 2020 y llegó al final de su vida útil el 31 de diciembre de 2023, según se anunció en la Actualización sobre el fin de la vida útil de la AMI de Amazon Linux en el blog de noticias de AWS. AWS ya no ofrece Amazon Machine Images (AMIs) para este sistema operativo. Sin embargo, AWS Systems Manager aún ofrece soporte para las instancias de Amazon Linux 1 existentes.

Amazon Linux 2

Versiones	x86	x86_64	ARM64
Versiones 2.0 y todas las posteriores		✓	✓

Amazon Linux 2023

Versiones	x86	x86_64	ARM64
2023.0.20230315.0 y todas las posteriores		✓	✓

Bottlerocket

Versiones	x86_64	ARM64
Versión 1.0.0 y todas las posteriores	✓	✓

CentOS

Versiones	x86	x86_64	ARM64
6.x[1]	✓	✓	
Versiones 7.1 y 7.x posteriores.		✓	✓
8.x		✓	✓

[1] Para usar estas versiones, debe usar una versión 3.0.x de SSM Agent. Le recomendamos que utilice la última versión 3.0.x disponible de SSM Agent. Las versiones posteriores de SSM Agent (3.1 o posterior) no son compatibles.

CentOS Stream

Versiones	x86	x86_64	ARM64
8		✓	✓
9		✓	✓

Servidor Debian

Versiones	x86	x86_64	ARM64
Jessie (8)		✓	
Stretch (9)		✓	✓
Buster (10)		✓	✓

Versiones	x86	x86_64	ARM64
Bullseye (11)		✓	✓
Bookworm (12)		✓	✓

Oracle Linux

Versiones	x86	x86_64	ARM64
7.5–7.8		✓	
8.x		✓	
9.x		✓	

Red Hat Enterprise Linux (RHEL)

Versiones	x86	x86_64	ARM64
6.x[1]	✓	✓	
7.0–7.5		✓	
7.6–8.x		✓	✓
9.x		✓	✓
10.x		✓	✓

[1] Para usar estas versiones, debe usar una versión 3.0.x de SSM Agent. Le recomendamos que utilice la última versión 3.0.x disponible de SSM Agent. Las versiones posteriores de SSM Agent (3.1 o posterior) no son compatibles.

Rocky Linux

Versiones	x86	x86_64	ARM64
8.x		✓	✓

Versiones	x86	x86_64	ARM64
9.x		✓	✓

SUSE Linux Enterprise Server (SLES)

Versiones	x86	x86_64	ARM64
Versiones 12 y 12.x posteriores.		✓	
15.x		✓	✓

Servidor Ubuntu

Versiones	x86	x86_64	ARM64
12.04 LTS y 14.04 LTS	✓	✓	
16.04 LTS y 18.04 LTS		✓	✓
20.04 LTS y 20.10 STR		✓	✓
22.04 LTS		✓	✓
23.04, 23.10		✓	✓
24.04 LTS		✓	✓
24.10		✓	✓
25.04		✓	✓

macOS (solo instancias de Amazon EC2)

Versión	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)		✓	✓
15.x (Sequoia)		✓	✓

> **Note**
>
> macOS no se admite en todas las Regiones de AWS. Para obtener más información sobre la compatibilidad de Amazon EC2 con macOS, consulte Instancias de Mac de Amazon EC2 en la Guía del usuario de Amazon EC2.

Raspberry Pi OS (anteriormente Raspbian)

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

Más información

- Administración de dispositivos de Raspberry Pi mediante AWS Systems Manager

Windows Server

SSM Agent requiere Windows PowerShell 3.0 o una versión posterior para ejecutar determinados documentos de AWS Systems Manager (documentos de SSM) en instancias de Windows Server (por ejemplo, el documento `AWS-ApplyPatchBaseline` heredado). Compruebe que las instancias de Windows Server estén ejecutando Windows Management Framework 3.0 o una versión posterior. Este marco contiene Windows PowerShell. Para obtener más información, consulte Windows Management Framework 3.0.

Versión	x86	x86_64	ARM64
2008[1]	✓	✓	
2008 R2[1]		✓	
2012 y 2012 R2[2]		✓	
2016		✓	
2019		✓	
2022		✓	
2025		✓	

[1] Compatibilidad de Windows Server 2008 y 2008 R2: A partir del 14 de enero de 2020, Windows Server 2008 ya no es compatible para obtener actualizaciones de características o seguridad de Microsoft. Amazon Machine Images heredadas (AMIs) para Windows Server 2008 y 2008 R2 aún incluyen la versión 2 de SSM Agent preinstalada, pero Systems Manager ya no admite oficialmente las versiones 2008 ni actualiza el agente para estas versiones de Windows Server. Además, es posible que la versión 3 de SSM Agent no sea compatible con todas las operaciones en Windows Server 2008 y 2008 R2. La versión final oficialmente admitida de SSM Agent para las versiones Windows Server 2008 es 2.3.1644.0.

[2] Compatibilidad de Windows Server 2012 y 2012 R2: La compatibilidad con Windows Server 2012 y 2012 R2 finalizó el 10 de octubre de 2023. Para utilizar SSM Agent con estas versiones, se recomienda utilizar las actualizaciones de seguridad ampliada (ESU) de Microsoft. Para obtener más información, consulte Finaliza el soporte para Windows Server 2012 y 2012 R2 en el sitio web de Microsoft.

Tipos de equipos compatibles con entornos híbridos y multinube

Systems Manager admite varios tipos de equipos como nodos administrados. Un nodo administrado es cualquier equipo configurado para funcionar con Systems Manager.

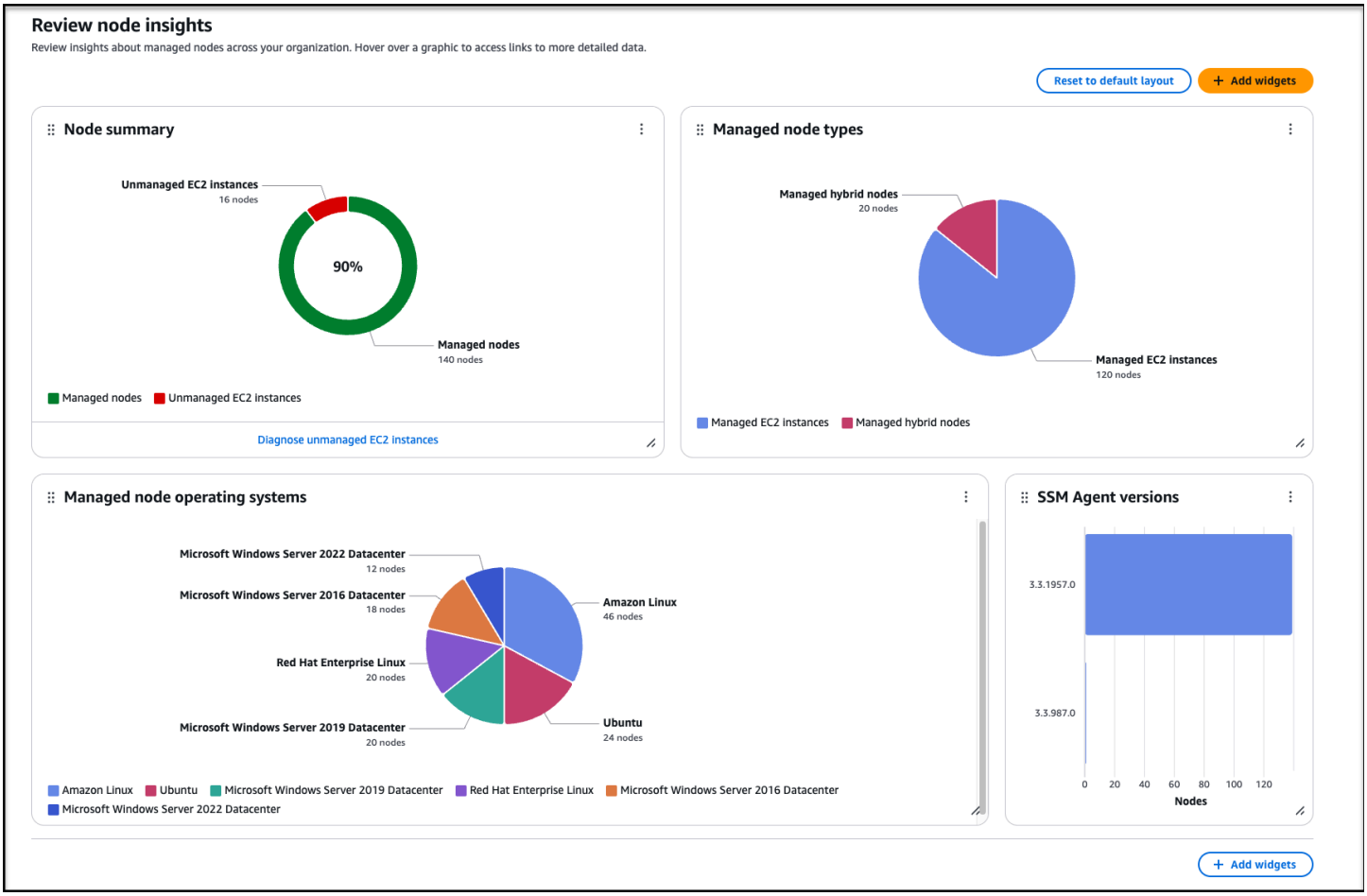
En esta guía del usuario se utilizan los términos híbrido y multinube para referirse a un entorno que contiene cualquier combinación de los siguientes tipos de equipos:

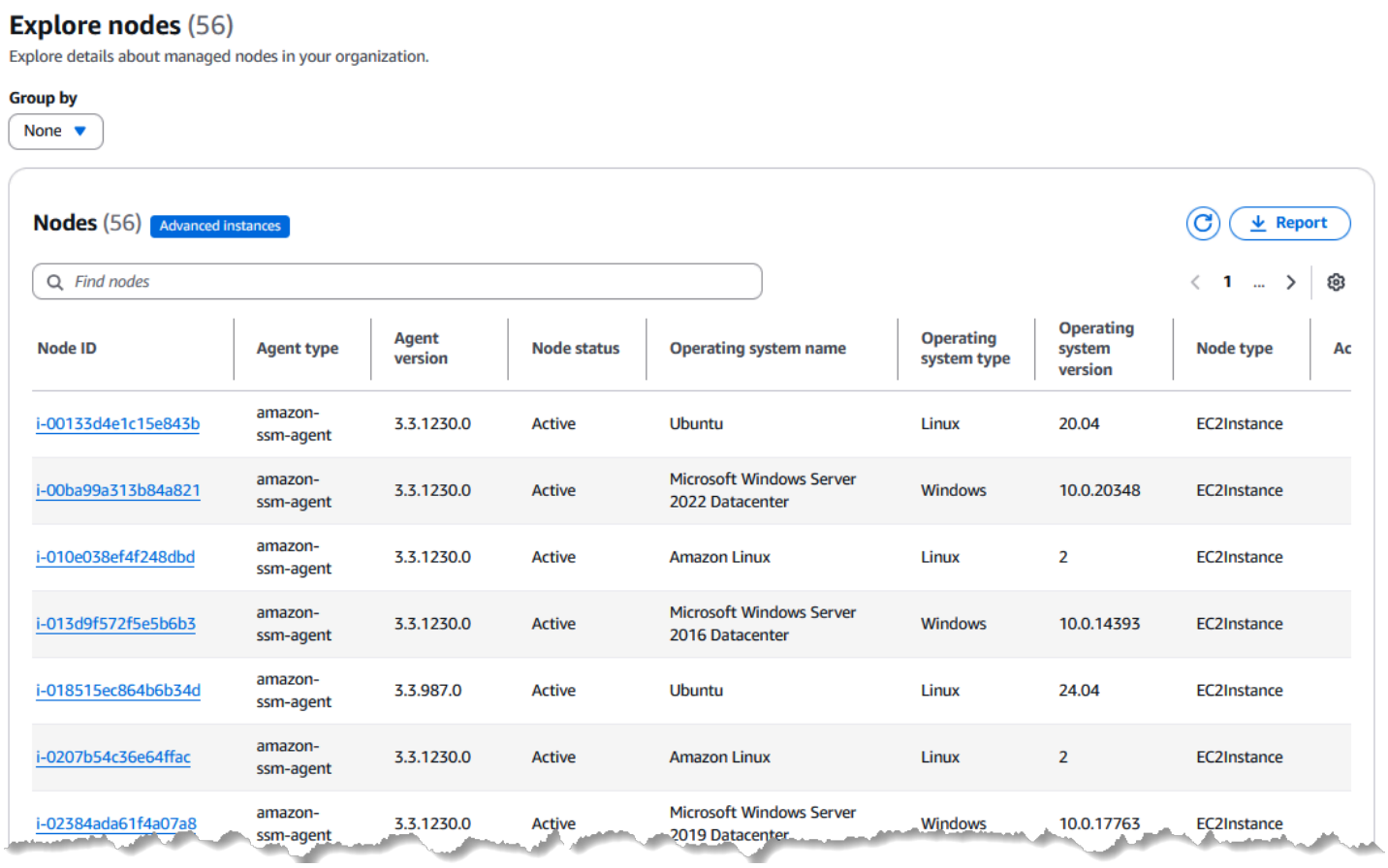
- Instancias de Amazon Elastic Compute Cloud (Amazon EC2)
- Servidores en sus propias instalaciones (servidores en las instalaciones)
- Dispositivos de núcleo de AWS IoT Greengrass
- Dispositivos de AWS IoT y periféricos que no sean de AWS
- Máquinas virtuales (VM), incluidas las VM de otros entornos de nube

Para obtener información acerca de la compatibilidad de AWS con entornos híbridos y multinube, consulte Soluciones de AWS para entornos híbridos y multinube.

¿Qué es la consola unificada?

La consola unificada de Systems Manager es una experiencia consolidada que combina varias herramientas para ayudarlo a completar tareas de nodos comunes en varias Cuentas de AWS y Regiones de AWS de una organización de AWS Organizations o en una sola cuenta y región. Los nodos pueden ser instancias de EC2, servidores híbridos o servidores que se ejecutan en un entorno multinube. En la consola unificada, se le proporciona información detallada sobre sus nodos. Puede generar informes para sus nodos, diagnosticar y corregir problemas comunes que impiden que los nodos informen como administrados por Systems Manager; por ejemplo, a causa de problemas de conectividad. Además de los resúmenes sobre sus nodos, puede ver detalles específicos sobre un nodo, como el inventario de software y el estado de las revisiones.


Review node insights
Review insights about managed nodes across your organization. Hover over a graphic to access links to more detailed data.
Reset to default layout
Add widgets
Node summary
Unmanaged EC2 instances
16 nodes
90%
Managed nodes
140 nodes
Managed nodes
Unmanaged EC2 instances
Diagnose unmanaged EC2 instances
Managed node types
Managed hybrid nodes
20 nodes
Managed EC2 instances
120 nodes
Managed EC2 instances
Managed hybrid nodes
Managed node operating systems
Microsoft Windows Server 2022 Datacenter
12 nodes
Microsoft Windows Server 2016 Datacenter
18 nodes
Red Hat Enterprise Linux
20 nodes
Microsoft Windows Server 2019 Datacenter
20 nodes
Amazon Linux
46 nodes
Ubuntu
24 nodes
Amazon Linux
Ubuntu
Microsoft Windows Server 2019 Datacenter
Red Hat Enterprise Linux
Microsoft Windows Server 2016 Datacenter
Microsoft Windows Server 2022 Datacenter
SSM Agent versions
3.3.1957.0
3.3.987.0
0
20
40
60
80
100
120
Nodes
Add widgets



Tanto si tiene nodos en varias cuentas y regiones de una organización, como nodos en una sola cuenta y región, le recomendamos que utilice la consola unificada. Systems Manager sigue agregando la capacidad de completar más tareas de nodos a la consola unificada. Para obtener más información sobre las tareas de nodos que puede llevar a cabo ahora con la consola unificada, consulte Ejecución de tareas de administración de nodos con AWS Systems Manager.

Continúe con los temas siguientes para configurar los nodos para que los administre Systems Manager. Para obtener más información sobre cómo configurar sus nodos para Systems Manager, consulte Configuración de nodos administrados para AWS Systems Manager. Una vez configurados los nodos, puede configurar Systems Manager y la consola unificada. Para obtener más información sobre cómo configurar Systems Manager, consulte Configuración de AWS Systems Manager.

Configuración de nodos administrados para AWS Systems Manager

Complete las tareas de esta sección para instalar y configurar roles, cuentas de usuario, permisos y recursos iniciales para usar las herramientas de AWS Systems Manager. Las tareas que se describen en esta sección las realizan normalmente los administradores de sistemas y una Cuenta de AWS. Una vez completados estos pasos, los usuarios de la organización pueden utilizar Systems Manager para configurar, administrar y acceder a los nodos administrados. Un nodo administrado es cualquier máquina configurada para su uso con Systems Manager en un entorno híbrido y multinube.

> **Note**
>
> Si tiene previsto utilizar las instancias de Amazon EC2 y sus propios recursos informáticos en un entorno híbrido y multinube, siga los pasos que se indican en Administrar instancias de EC2 con Systems Manager. En este tema se presentan los pasos en el orden más conveniente para completar la configuración de Systems Manager para instancias de EC2 y equipos que no sean de EC2.

Si ya utiliza otros Servicios de AWS, ya ha completado algunos de estos pasos. Sin embargo, otros pasos son específicos de Systems Manager. Por lo tanto, le recomendamos revisar toda esta sección para asegurarse de que lo tenga todo listo para utilizar todas las herramientas de Systems Manager.

Temas

- Administrar instancias de EC2 con Systems Manager
- Administrador de nodos en entornos híbridos y multinube con Systems Manager
- Administración de dispositivos periféricos con Systems Manager
- Creación de un administrador delegado de AWS Organizations para Systems Manager
- Configuración general para AWS Systems Manager

Administrar instancias de EC2 con Systems Manager

Complete las tareas de esta sección para instalar y configurar roles, permisos y recursos iniciales para AWS Systems Manager. Las tareas que se describen en esta sección las realizan normalmente

los administradores de sistemas y una Cuenta de AWS. Una vez completados estos pasos, los usuarios de la organización pueden utilizar Systems Manager para configurar, administrar y acceder a las instancias de Amazon Elastic Compute Cloud (Amazon EC2).

> **Note**
>
> Si tiene previsto utilizar Systems Manager para administrar y configurar máquinas locales, siga los pasos de configuración en Administrador de nodos en entornos híbridos y multinube con Systems Manager. Si tiene previsto utilizar tanto instancias de Amazon EC2 como máquinas que no sean EC2 en un entorno híbrido y multinube, siga primero los pasos que se indican a continuación. En esta sección, se presentan los pasos en el orden recomendado para configurar los roles, los usuarios, los permisos y los recursos iniciales que se van a utilizar en sus operaciones de Systems Manager.

Si ya utiliza otros Servicios de AWS, ya ha completado algunos de estos pasos. Sin embargo, otros pasos son específicos de Systems Manager. Por lo tanto, le recomendamos revisar toda esta sección para asegurarse de que lo tenga todo listo para utilizar todas las herramientas de Systems Manager.

Contenido

- Configuración de permisos de instancia requeridos para Systems Manager
- Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager

Configuración de permisos de instancia requeridos para Systems Manager

De forma predeterminada, AWS Systems Manager no tiene permiso para realizar acciones en sus instancias. Puede proporcionar permisos de instancia a nivel de cuenta mediante un rol de AWS Identity and Access Management (IAM) o a nivel de instancia mediante un perfil de instancia. Si su caso de uso lo permite, le recomendamos que conceda el acceso a nivel de cuenta mediante la configuración de administración de host predeterminada.

> **Note**
>
> Puede omitir este paso y permitir que Systems Manager aplique por usted los permisos necesarios a las instancias cuando se configure la consola unificada. Para obtener más información, consulte Configuración de AWS Systems Manager.

Configuración recomendada para permisos de instancia de EC2

La configuración de administración de host predeterminada permite a Systems Manager administrar sus instancias de Amazon EC2 de forma automática. Tras activar esta configuración, todas las instancias que utilicen la versión 2 del servicio de metadatos de instancias (IMDSv2) en la Región de AWS y en la Cuenta de AWS con la versión 3.2.582.0 de SSM Agent o posterior instalada se convertirán automáticamente en instancias administradas. La configuración de administración de host predeterminada no admite la versión 1 del servicio de metadatos de instancias. Para obtener información sobre la transición a IMDSv2, consulte Transición al uso de Servicio de metadatos de instancia versión 2 en la Guía del usuario de Amazon EC2. Para obtener información sobre cómo comprobar la versión de SSM Agent instalada en la instancia, consulte Verificación del número de versión de SSM Agent. Para obtener información sobre cómo actualizar SSM Agent, consulte Actualización automática de SSM Agent. Entre los beneficios de administrar instancias, se incluyen los siguientes:

- Conéctese a sus instancias de forma segura mediante Session Manager.
- Realice escaneos de revisiones automatizados mediante Patch Manager.
- Consulte información detallada sobre sus instancias mediante Systems Manager Inventory.
- Realice un seguimiento y administre las instancias mediante Fleet Manager.
- Mantenga SSM Agent actualizado automáticamente.

Fleet Manager, Inventario, Patch Manager y Session Manager son herramientas de AWS Systems Manager.

La configuración de administración de host predeterminada permite la administración de instancias sin el uso de perfiles de instancia y garantiza que Systems Manager tenga permisos para administrar todas las instancias de la región y la cuenta. Si los permisos proporcionados no son suficientes para su caso de uso, también puede agregar políticas al rol de IAM predeterminado creado en la configuración de administración de host predeterminada. Como alternativa, si no necesita permisos para todas las capacidades proporcionadas por el rol de IAM predeterminado, puede crear sus

propias políticas y roles personalizados. Cualquier cambio realizado en el rol de IAM que elija para la configuración de administración de host predeterminada se aplica a todas las instancias de Amazon EC2 administradas en la región y la cuenta. Para obtener más información acerca de la política que usa la configuración de administración de host predeterminada, consulte Política administrada por AWS: AmazonSSMManagedEC2InstanceDefaultPolicy. Para obtener más información sobre la configuración de administración de host predeterminada, consulte Administración automática de instancias EC2 con la configuración de administración de hosts predeterminada.

Important

Las instancias registradas mediante la Configuración de la administración de hosts predeterminada almacenan la información de registro localmente en los directorios `/lib/amazon/ssm` o `C:\ProgramData\Amazon`. Si se eliminan estos directorios o sus archivos, la instancia no podrá adquirir las credenciales necesarias para conectarse a Systems Manager mediante la Configuración de la administración de hosts predeterminada. En estos casos, debe utilizar un perfil de instancia para proporcionar los permisos necesarios a la instancia, o bien volver a crearla.

Note

Este procedimiento está pensado para que solo lo realicen los administradores. Implemente el acceso con privilegios mínimos cuando permita que las personas configuren o modifiquen la configuración de administración de host predeterminada. Debe activar la configuración de administración de host predeterminada en cada Región de AWS en la que desee administrar automáticamente sus instancias de Amazon EC2.

Para activar la configuración de administración de host predeterminada

Puede activar la configuración de administración de host predeterminada desde la consola de Fleet Manager. Para completar correctamente este procedimiento con la AWS Management Console o la herramienta de línea de comandos que prefiera, debe tener permisos para las operaciones de las API GetServiceSetting, ResetServiceSetting y UpdateServiceSetting. Además, debe tener permisos `iam:PassRole` para el rol de IAM `AWSSystemsManagerDefaultEC2InstanceManagementRole`. A continuación, se muestra una política de ejemplo. Reemplace cada *`example resource placeholder`* con su propia información.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting",
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "arn:aws:ssm:us-east-1:111122223333:servicesetting/ssm/
managed-instance/default-ec2-instance-management-role"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::111122223333:role/service-role/
AWSSystemsManagerDefaultEC2InstanceManagementRole",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
                        "ssm.amazonaws.com"
                    ]
                }
            }
        }
    ]
}
```

Antes de empezar, si tiene perfiles de instancia adjuntos a sus instancias de Amazon EC2, elimine todos los permisos que permitan la operación `ssm:UpdateInstanceInformation`. SSM Agent intenta usar los permisos del perfil de instancia antes de usar los permisos de configuración de administración de host predeterminados. Si permite la operación `ssm:UpdateInstanceInformation` en los perfiles de su instancia, la instancia no utilizará los permisos de la configuración de administración de host predeterminada.

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Seleccione Configurar la Configuración de la administración de hosts predeterminada en el menú desplegable Administración de la cuenta.
4. Active Habilitar configuración de administración de host predeterminada.
5. Elija el rol de IAM que se utiliza para habilitar las herramientas de Systems Manager en sus instancias. Recomendamos utilizar el rol predeterminado que proporciona la configuración de administración de host predeterminada. Contiene el conjunto mínimo de permisos necesarios para administrar sus instancias de Amazon EC2 mediante Systems Manager. Si prefiere utilizar un rol personalizado, la política de confianza del rol debe permitir que Systems Manager sea una entidad de confianza.
6. Elija Configurar para completar la configuración.

Tras activar la configuración de administración de host predeterminada, es posible que las instancias tarden 30 minutos en utilizar las credenciales del rol que ha elegido. Debe activar la configuración de administración de host predeterminada en cada región en la que desee administrar automáticamente sus instancias de Amazon EC2.

Configuración alternativa para permisos de instancia de EC2

Para otorgar acceso a nivel de instancias individuales, debe utilizar un perfil de instancia de AWS Identity and Access Management (IAM). Un perfil de instancias es un contenedor que pasa información del rol de IAM a una instancia de Amazon Elastic Compute Cloud (Amazon EC2) al momento del lanzamiento. Puede crear un perfil de instancias para Systems Manager, adjuntando una o más políticas de IAM que definan los permisos necesarios para un nuevo rol o para un rol que ya haya creado.

> **Note**
>
> Puede utilizar Quick Setup, una herramienta de AWS Systems Manager, para configurar rápidamente un perfil de instancias en todas las instancias de su Cuenta de AWS. Quick Setup también crea un rol de servicio de IAM (o rol de asunción), lo que permite a Systems Manager ejecutar comandos de forma segura en las instancias en su nombre. Si utiliza Quick Setup, puede omitir este paso (paso 3) y el paso 4. Para obtener más información, consulte AWS Systems Manager Quick Setup.

Tenga en cuenta los siguientes detalles sobre cómo crear un perfil de instancias de IAM:

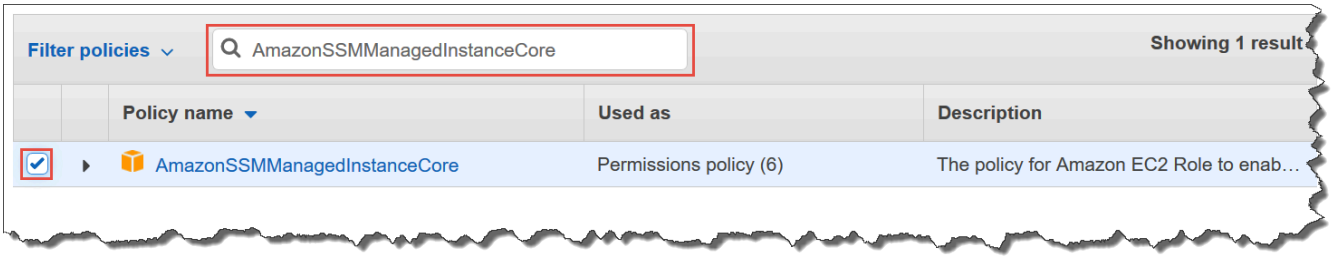
- Si está configurando equipos que no son de EC2 en un entorno híbrido y multinube para Systems Manager, no es necesario crear un perfil de instancia para ellos. En su lugar, configure los servidores y las máquinas virtuales para que utilicen el rol de servicio de IAM. Para obtener más información, consulte Creación del rol de servicio de IAM requerido para Systems Manager en entornos híbridos y multinube.
- Si cambia el perfil de instancias de IAM, puede pasar un tiempo antes de que las credenciales de la instancia se actualicen. El SSM Agent no procesará solicitudes hasta que esto ocurra. Para acelerar el proceso de actualización, puede reiniciar el SSM Agent o la instancia.

En función de si va a crear un nuevo rol para su perfil de instancias o si va a agregar los permisos necesarios para un rol existente, utilice uno de los siguientes procedimientos.

Para crear un perfil de instancias para instancias administradas de Systems Manager (consola)

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, seleccione Roles y luego seleccione Crear rol.
3. En Trusted entity type (Tipo de entidad de confianza), elija Servicio de AWS.
4. Inmediatamente debajo de Use case (Caso de uso), seleccione EC2 y, a continuación, Next (Siguiente).
5. En la página Agregar permisos, haga lo siguiente:

   - Utilice el campo Search (Buscar) para localizar la política AmazonSSMManagedInstanceCore. Seleccione la casilla de verificación situada junto a su nombre, como se muestra en la siguiente ilustración.



     La consola conserva la selección aunque busque otras políticas.

   - Si ha creado una política de bucket de S3 personalizada en el procedimiento anterior, (Opcional) Crear una política personalizada para el acceso al bucket de S3, búsquela y seleccione la casilla de verificación situada junto a su nombre.

- Si tiene previsto unir instancias a una instancia de Active Directory administrada por AWS Directory Service, busque AmazonSSMDirectoryServiceAccess y seleccione la casilla de verificación situada junto a su nombre.
- Si tiene previsto utilizar EventBridge o los Registros de CloudWatch para administrar o supervisar la instancia, busque CloudWatchAgentServerPolicy y seleccione la casilla de verificación situada junto a su nombre.

6. Elija Siguiente.
7. En Role name (Nombre del rol), ingrese un nombre para el nuevo perfil de instancias, por ejemplo, **`SSMInstanceProfile`**.

   > **Note**
   >
   > Anote el nombre del rol. Elegirá este rol cuando cree instancias nuevas que desee administrar mediante Systems Manager.

8. (Opcional) En Description (Descripción), actualice la descripción de este perfil de instancia.
9. (Opcional) En Tags (Etiquetas), agregue uno o varios pares de valores etiqueta-clave para organizar, seguir o controlar el acceso a este rol, y luego elija Create role (Crear rol). El sistema le devuelve a la página Roles.

Para agregar permisos de perfil de instancias para Systems Manager a un rol existente (consola)

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, elija Roles y, a continuación, elija el rol existente que desea asociar con un perfil de instancias para operaciones de Systems Manager.
3. En la pestaña Permissions (Permisos), elija Add permissions, Attach policies (Agregar permisos, Adjuntar políticas).
4. En la página Attach policy (Adjuntar política), lleve a cabo las siguientes operaciones:

   - Utilice el campo Search (Buscar) para localizar la política AmazonSSMManagedInstanceCore. Seleccione la casilla de verificación situada junto a su nombre.
   - Si ha creado una política de bucket de S3 personalizada, búsquela y seleccione la casilla de verificación situada junto a su nombre. Para obtener más información sobre las políticas personalizadas del bucket de S3 para un perfil de instancia, consulte (Opcional) Crear una política personalizada para el acceso al bucket de S3.

- Si tiene previsto unir instancias a una instancia de Active Directory administrada por AWS Directory Service, busque AmazonSSMDirectoryServiceAccess y seleccione la casilla de verificación situada junto a su nombre.
- Si tiene previsto utilizar EventBridge o los Registros de CloudWatch para administrar o supervisar la instancia, busque CloudWatchAgentServerPolicy y seleccione la casilla de verificación situada junto a su nombre.

5. Seleccione Asociar políticas.

Para obtener información acerca de cómo se actualiza un rol para que incluya una entidad de confianza o que restrinja aún más el acceso, consulte Modificación de un rol en la Guía del usuario de IAM.

(Opcional) Crear una política personalizada para el acceso al bucket de S3

La creación de una política personalizada para el acceso a Amazon S3 solo es necesaria si utiliza un punto de enlace de la VPC o un bucket de S3 de su propiedad en sus operaciones de Systems Manager. Puede adjuntar esta política al rol de IAM predeterminado creado por la configuración de administración de host predeterminada o a un perfil de instancia que haya creado en el procedimiento anterior.

Para obtener más información acerca de los buckets de S3 administrados de AWS a los que proporciona acceso en la siguiente política, consulte Comunicaciones de SSM Agent con buckets de S3 administrados de AWS.

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, seleccione Policies (Políticas) y, a continuación, seleccione Create policy (Crear política).
3. Seleccione la pestaña JSON y sustituya el texto predeterminado con lo siguiente.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "s3:GetObject",
            "Resource": [
```

```
                "arn:aws:s3:::aws-ssm-us-east-2/*",
                "arn:aws:s3:::aws-windows-downloads-us-east-2/*",
                "arn:aws:s3:::amazon-ssm-us-east-2/*",
                "arn:aws:s3:::amazon-ssm-packages-us-east-2/*",
                "arn:aws:s3:::us-east-2-birdwatcher-prod/*",
                "arn:aws:s3:::aws-ssm-distributor-file-us-east-2/*",
                "arn:aws:s3:::aws-ssm-document-attachments-us-east-2/*",
                "arn:aws:s3:::patch-baseline-snapshot-us-east-2/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:PutObject",
                "s3:PutObjectAcl",
                "s3:GetEncryptionConfiguration"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket/*",
                "arn:aws:s3:::amzn-s3-demo-bucket"
            ]
        }
    ]
}
```

[1] El primer elemento `Statement` solo es necesario si utiliza un punto de enlace de la VPC.

[2] El segundo elemento `Statement` solo es necesario si utiliza un bucket de S3 que ya haya creado para usarlo en sus operaciones de Systems Manager.

[3] El permiso de la lista de control de acceso de `PutObjectAcl` solo es necesario si planea permitir el acceso entre cuentas para los buckets de S3 en otras cuentas

[4] El elemento `GetEncryptionConfiguration` es necesario si el bucket de S3 está configurado para utilizar el cifrado.

[5] Si el bucket de S3 está configurado para utilizar el cifrado, entonces la raíz del bucket de S3 (por ejemplo, `arn:aws:s3:::amzn-s3-demo-bucket`) debe aparecer en la sección Resource (Recurso). Su usuario, grupo o rol debe estar configurado con acceso al bucket raíz.

4. Si está utilizando un punto de enlace de la VPC en sus operaciones, haga lo siguiente:

   En el primer elemento `Statement`, sustituya cada marcador de *`región`* con el identificador de la Región de AWS en la que se utilizará esta política. Por ejemplo, utilice `us-east-2` para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

   > **Important**
   >
   > Recomendamos que evite el uso de caracteres comodín (*) en lugar de regiones específicas en esta política. Por ejemplo, utilice `arn:aws:s3:::aws-ssm-us-east-2/*` y no `arn:aws:s3:::aws-ssm-*/*`. El uso de caracteres comodín podría conceder acceso a buckets de S3 a los que no quiere darlo. Si desea utilizar el perfil de instancia para más de una región, le recomendamos repetir el primer elemento de `Statement` de cada región.

   -o bien-

   Si no va a utilizar un punto de enlace de la VPC en sus operaciones, puede eliminar el primer elemento `Statement`.

5. Si está utilizando un bucket de S3 de su propiedad en sus operaciones de Systems Manager, haga lo siguiente:

   En el segundo elemento de `Statement`, sustituya *`amzn-s3-demo-bucket`* con el nombre de un bucket de S3 en su cuenta. Utilizará este bucket para sus operaciones de Systems Manager. Otorga permisos para objetos del bucket mediante `"arn:aws:s3:::my-bucket-name/*"` como recurso. Para obtener más información acerca de cómo se proporcionan permisos para buckets u objetos en buckets, consulte el tema Acciones de políticas para Amazon S3 en la Guía del usuario de Amazon Simple Storage Service y en la publicación del blog de AWS IAM Policies and Bucket Policies and ACLs! Oh, My! (Control del acceso a los recursos de S3).

   > **Note**
   >
   > Si utiliza más de un bucket, facilite el ARN de cada uno. Consulte el siguiente ejemplo sobre los permisos de los buckets.

```
"Resource": [
"arn:aws:s3:::amzn-s3-demo-bucket1/*",
"arn:aws:s3:::amzn-s3-demo-bucket2/*"
              ]
```

-o bien-

Si no va a utilizar un bucket de S3 de su propiedad en las operaciones de Systems Manager, puede eliminar el segundo elemento `Statement`.

6. Elija Siguiente: etiquetas.
7. (Opcional) Para agregar etiquetas, elija Add tag (Agregar etiqueta) e ingrese las etiquetas preferidas para la política.
8. Elija Siguiente: Revisar.
9. En Name (Nombre), ingrese un nombre para identificar esta política, por ejemplo, **`SSMInstanceProfileS3Policy`**.
10. Seleccione Crear política.

Consideraciones sobre políticas adicionales para las instancias administradas

En esta sección, se describen algunas de las políticas que puede agregar al rol de IAM predeterminado creado por la configuración de administración de host predeterminada o a los perfiles de instancia para AWS Systems Manager. Para proporcionar permisos para la comunicación entre las instancias y la API de Systems Manager, le recomendamos que cree políticas personalizadas que reflejen las necesidades del sistema y los requisitos de seguridad. En función de su plan de operaciones, es posible que necesite permisos representados en una o varias de las otras políticas.

Política: **`AmazonSSMDirectoryServiceAccess`**

Solo es necesaria si planea unir instancias de Amazon EC2 para Windows Server a un directorio de Microsoft AD.

Esta política administrada de AWS permite a SSM Agent acceder a AWS Directory Service en su nombre para las solicitudes de unión al dominio por parte de la instancia administrada. Para

obtener más información, consulte Cómo unir fácilmente una instancia de EC2 de Windows en la Guía de administración de AWS Directory Service.

Política: **`CloudWatchAgentServerPolicy`**

Solo es necesaria si planea instalar y ejecutar el agente de CloudWatch en sus instancias para leer métricas y datos de registro en una instancia y escribirlos en Amazon CloudWatch. Esto sirve para monitorear, analizar y responder rápidamente a problemas o cambios en los recursos de AWS.

Su rol de IAM predeterminado creado por la configuración de administración de host predeterminada o el perfil de instancia necesita esta política solo si utilizará características como Amazon EventBridge o Registros de Amazon CloudWatch. (También puede crear una política más restrictiva que, por ejemplo, limite la escritura de acceso a un flujo de registro específico de los Registros de CloudWatch).

> **Note**
>
> El uso de las características de EventBridge y los Registros de CloudWatch es opcional. No obstante, le recomendamos que las configure al principio de su proceso de configuración de Systems Manager si ha decidido usarlas. Para obtener más información, consulte la Guía del usuario de Amazon EventBridge y la Guía del usuario de Registros de Amazon CloudWatch.

Para crear políticas de IAM con permisos para herramientas adicionales de Systems Manager, consulte los siguientes recursos:

- Restringir el acceso a los parámetros de Parameter Store mediante políticas de IAM
- Configuración de Automation
- Paso 2: Verificar o agregar permisos de instancia para Session Manager

Adjuntar el perfil de instancia de Systems Manager a una instancia (consola)

El siguiente procedimiento describe cómo asociar un perfil de instancia de IAM a una instancia de Amazon EC2 mediante la consola de Amazon EC2.

1. Inicie sesión en la AWS Management Console y abra la consola de Amazon EC2 en https://console.aws.amazon.com/ec2/.

2. En el panel de navegación, bajo Instances, elija Instances.
3. Diríjase a la instancia de EC2 en la lista y elíjala.
4. En el menú Actions (Acciones), elija Security (Seguridad), Modify IAM role (Modificar rol de IAM).
5. En IAM role (Rol de IAM), seleccione el perfil de instancia creado utilizando el procedimiento que se describe en Configuración alternativa para permisos de instancia de EC2.
6. Elija Update IAM role (Actualizar rol de IAM).

Para obtener más información acerca de cómo adjuntar roles de IAM a instancias, elija una de las siguientes opciones, en función del tipo de sistema operativo seleccionado:

- Asociación de un rol de IAM a una instancia en la Guía del usuario de Amazon EC2
- Asociación de un rol de IAM a una instancia en la Guía del usuario de Amazon EC2

Siga en Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager.

Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager

Puede mejorar la posición de seguridad de los nodos administrados (incluidas las máquinas que no son de EC2 en su entorno híbrido y multinube) mediante la configuración de AWS Systems Manager para que use un punto de conexión de VPC de interfaz en Amazon Virtual Private Cloud (Amazon VPC). Mediante un punto de conexión de VPC de la interfaz (punto de conexión de la interfaz), puede conectarse a servicios con tecnología de AWS PrivateLink. AWS PrivateLink es una tecnología que permite obtener acceso de forma privada a las API de Amazon Elastic Compute Cloud (Amazon EC2) y Systems Manager mediante direcciones IP privadas.

AWS PrivateLink restringe todo el tráfico de red entre las instancias administradas, Systems Manager y Amazon EC2 y la red de Amazon. Esto significa que las instancias administradas no tienen acceso a Internet. Si utiliza AWS PrivateLink, no necesita una puerta de enlace de Internet, un dispositivo NAT ni una puerta de enlace privada virtual.

No es necesario configurar AWS PrivateLink, pero es recomendable. Para obtener más información sobre AWS PrivateLink y los puntos de conexión de VPC, consulte AWS PrivateLink y los puntos de conexión de VPC.

> **Note**
>
> La alternativa a usar un punto de enlace de la VPC es permitir el acceso a Internet saliente en las instancias administradas. En este caso, las instancias administradas también deben permitir el tráfico saliente HTTPS (puerto 443) a los siguientes puntos de enlace:
>
> - `ssm.region.amazonaws.com`
> - `ssmmessages.region.amazonaws.com`
> - `ec2messages.region.amazonaws.com`
>
> SSM Agent inicia todas las conexiones al servicio de Systems Manager en la nube. Por este motivo, no es necesario configurar el firewall para permitir el tráfico entrante a las instancias de Systems Manager.
> Para obtener más información acerca de los puntos de enlace, consulte Referencia: ec2messages, ssmmessages y otras operaciones de la API.

Acerca de Amazon VPC

Puede utilizar Amazon Virtual Private Cloud (Amazon VPC) para definir una red virtual en su propia área aislada lógicamente dentro de la Nube de AWS, conocida como una nube privada virtual (VPC). Puede lanzar recursos de AWS, como, por ejemplo, instancias, en su VPC. Una VPC es prácticamente idéntica a una red tradicional que usted puede operar en su propio centro de datos, con los beneficios que supone utilizar la infraestructura escalable de AWS. Puede configurar la VPC, seleccionar su rango de direcciones IP, crear subredes y configurar tablas de enrutamiento, puerta de enlace de red y ajustes de seguridad. Ahora puede conectar sus instancias de la VPC a Internet. Puede conectar la VPC a su propio centro de datos corporativo, lo que convierte la Nube de AWS en una ampliación del centro de datos. Para proteger los recursos de cada subred, puede utilizar varias capas de seguridad, incluidos grupos de seguridad y listas de control de acceso a la red. Para obtener más información, consulte la Guía del usuario de Amazon VPC.

Temas

- Restricciones y limitaciones de los puntos de enlace de la VPC
- Creación de puntos de enlace de la VPC para Systems Manager
- Crear una política de punto de enlace de la VPC de tipo interfaz

Restricciones y limitaciones de los puntos de enlace de la VPC

Antes de configurar los puntos de enlace de la VPC para Systems Manager debe conocer las siguientes restricciones y limitaciones.

Interconexiones de VPC

A los puntos de enlace de la interfaz de VPC se puede acceder a través de una interconexión con VPC dentro de las regiones y entre regiones. Para obtener más información acerca de las solicitudes de conexión de emparejamiento de VPC para los puntos de conexión de la interfaz de la VPC, consulte Cuotas de Amazon VPC en la Guía del usuario de Amazon Virtual Private Cloud.

Las conexiones de punto de conexión de puerta de enlace de VCP no se pueden ampliar más allá de una VPC. Los recursos del otro lado de una interconexión de VPC en la VPC no pueden utilizar el punto de enlace de gateway para comunicarse con los recursos del servicio de punto de enlace de gateway. Para obtener más información acerca de las solicitudes de conexión de emparejamiento de VPC para puntos de conexión de puertas de enlace de la VPC, consulte Cuotas de Amazon VPC en la Guía del usuario de Amazon Virtual Private Cloud

Conexiones entrantes

El grupo de seguridad asociado al punto de enlace de la VPC debe permitir las conexiones entrantes en el puerto 443 desde la subred privada de la instancia administrada. Si no se permiten las conexiones entrantes, la instancia administrada no podrá conectarse a los puntos de enlace de SSM y EC2.

Resolución de los DNS

Si utiliza un servidor DNS personalizado, debe agregar un reenviador condicional para cualquier consulta sobre el dominio `amazonaws.com` al servidor DNS de Amazon de su VPC.

Buckets de S3

Su política de punto de conexión de VPC debe permitir al menos el acceso a los siguientes buckets Amazon S3 enumerados en Comunicaciones de SSM Agent con buckets de S3 administrados de AWS:

> **Note**
>
> Si utiliza un firewall local y planea usar Patch Manager, ese firewall también debe permitir el acceso al punto de enlace de línea de base de revisiones correspondiente.

Registros de Amazon CloudWatch

Si no permite que las instancias accedan a Internet, cree un punto de enlace de la VPC para que los Registros de CloudWatch utilicen características que envíen Registros a CloudWatch. Para obtener más información acerca de cómo crear un punto de enlace para los Registros de CloudWatch, consulte Creación de un punto de enlace de la VPC para los Registros de CloudWatch en la Guía del usuario de los Registros de Amazon CloudWatch.

DNS en un entorno híbrido y multinube

Para obtener más información sobre cómo se configura DNS para trabajar con los puntos de conexión de AWS PrivateLink en entornos híbridos y multinube, consulte DNS privado para puntos de conexión de interfaz en la Guía del usuario de Amazon VPC. Si desea utilizar su propio DNS, puede utilizar Route 53 Resolver. Para obtener más información, consulte Resolving DNS queries between VPCs and your network en la Guía para desarrolladores de Amazon Route 53.

Creación de puntos de enlace de la VPC para Systems Manager

Utilice la siguiente información para crear puntos de conexión de interfaces de VPC para AWS Systems Manager. Este tema se vincula con los procedimientos en la Guía del usuario de Amazon VPC.

> **Note**
>
> *`region`* representa el identificador de Región de AWS compatible con AWS Systems Manager, como `us-east-2` para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

Siga los pasos descritos en Create an interface endpoint (Creación de un punto de enlace de interfaz) para crear los siguientes puntos de enlace de la interfaz:

- **`com.amazonaws.`*`region`*`.ssm`**: el punto de conexión para el servicio de Systems Manager.
- **`com.amazonaws.`*`region`*`.ec2messages`**: Systems Manager utiliza este punto de conexión para realizar llamadas desde SSM Agent al servicio de Systems Manager. A partir de la versión 3.3.40.0 de SSM Agent, Systems Manager comenzó a utilizar el punto de conexión `ssmmessages:*` (Amazon Message Gateway Service) siempre que estaba disponible en lugar del punto de conexión de `ec2messages:*` (Amazon Message Delivery Service).

- **com.amazonaws.*region*.ec2**: si utiliza Systems Manager para crear instantáneas compatibles con VSS, debe asegurarse de que tiene un punto de conexión al servicio de EC2. Si el punto de conexión de EC2 no está definido, se produce un error en una llamada para enumerar los volúmenes de Amazon EBS adjuntos, lo que hace que el comando de Systems Manager no se ejecute correctamente.
- **com.amazonaws.*region*.s3**: Systems Manager utiliza este punto de conexión para realizar la actualizaciónSSM Agent. Systems Manager también utiliza este punto de conexión, si, de manera opcional, opta por recuperar scripts u otros archivos almacenados en buckets o cargar registros de salida a un bucket. Si el grupo de seguridad asociado a su instancia restringe el tráfico saliente, debe agregar una regla para permitir el tráfico hacia la lista de prefijos para Amazon S3. Para obtener más información, consulte Modificación del grupo de seguridad en la Guía de AWS PrivateLink.
- **com.amazonaws.*region*.ssmmessages**: este punto de conexión es necesario para que SSM Agent se comunique con el servicio de Systems Manager, para Run Command, y si se conecta a sus instancias a través de un canal de datos seguro mediante Session Manager. Para obtener más información, consulte AWS Systems Manager Session Manager y Referencia: ec2messages, ssmmessages y otras operaciones de la API.
- (Opcional) **com.amazonaws.*region*.kms**: crear este punto de conexión si desea utilizar cifrado de AWS Key Management Service (AWS KMS) para parámetros de Session Manager o Parameter Store.
- (Opcional) **com.amazonaws.*region*.logs**: crear este punto de conexión si desea utilizar Registros de Amazon CloudWatch (CloudWatch Logs) para registros de Session Manager, Run Command o SSM Agent.

Para obtener información acerca de los buckets de S3 de AWS administrados a los que SSM Agent debe tener acceso, consulte Comunicaciones de SSM Agent con buckets de S3 administrados de AWS. Si utiliza un punto de conexión de nube privada virtual (VPC) en las operaciones de Systems Manager, necesita conceder permiso explícito en un perfil de instancia de EC2 para Systems Manager o en un rol de servicio para nodos que no son de EC2 en un entorno híbrido y multinube.

Crear una política de punto de enlace de la VPC de tipo interfaz

Puede crear políticas para los puntos de enlace de la interfaz de VPC de AWS Systems Manager en la que puede especificar:

- la entidad principal que puede realizar acciones

- Las acciones que se pueden realizar
- los recursos en los que se pueden realizar acciones

Para obtener más información, consulte Control access to services with VPC endpoints en la Guía del usuario de Amazon VPC.

Administrador de nodos en entornos híbridos y multinube con Systems Manager

Se puede utilizar AWS Systems Manager para gestionar tanto las instancias de Amazon Elastic Compute Cloud (EC2) como varios tipos de máquinas que no son de EC2. En esta sección se describen las tareas de configuración que la cuenta y los administradores de sistemas realizan para administrar máquinas que no son de EC2 mediante Systems Manager en un entorno híbrido y multinube. Una vez que se hayan completado estos pasos, los usuarios que hayan recibido permisos del administrador de la Cuenta de AWS pueden utilizar Systems Manager para configurar y administrar las máquinas que no son de EC2 de la organización.

Cualquier máquina que se haya configurado para su uso con Systems Manager se denomina nodo administrado.

> **Note**
>
> - Puede registrar dispositivos de periferia como nodos administrados mediante los mismos pasos de activación híbrida que se utilizan para otras máquinas que no son de EC2. Estos tipos de dispositivos periféricos incluyen tanto dispositivos de AWS IoT como dispositivos distintos de los dispositivos de AWS IoT. Use el proceso descrito en esta sección para configurar estos tipos de dispositivos periféricos.
>
>   Systems Manager también admite dispositivos de borde que utilizan el software AWS IoT Greengrass Core. El proceso de configuración y los requisitos para los dispositivos de núcleo de AWS IoT Greengrass son diferentes de aquellos para los dispositivos AWS IoT y periféricos que no son dispositivos de periferia de AWS. Para obtener información sobre el registro de dispositivos AWS IoT Greengrass para usarlos con Systems Manager, consulte Administración de dispositivos periféricos con Systems Manager.
> - Las máquinas macOS que no son de EC2 no son compatibles con entornos de híbridos y multinube de Systems Manager.

Si tiene previsto utilizar Systems Manager para administrar instancias de Amazon Elastic Compute Cloud (Amazon EC2) o desea utilizar tanto instancias de Amazon EC2 como máquinas que no son de EC2 en un entorno híbrido y multinube, en primer lugar, siga los pasos indicados en Administrar instancias de EC2 con Systems Manager.

Luego de configurar el entorno híbrido y multinube para Systems Manager, puede realizar lo siguiente:

- Cree una manera uniforme y segura de administrar de forma remota las cargas de trabajo híbridas y multinube de una ubicación con las mismas herramientas o scripts.
- Centralizar el control de acceso para las acciones que se pueden llevar a cabo en las máquinas mediante AWS Identity and Access Management (IAM).
- Para centralizar la auditoría de las operaciones realizadas en sus máquinas, consulte la actividad de la API registrada en AWS CloudTrail.

  Para obtener información acerca de cómo utilizar CloudTrail para monitorear las acciones de Systems Manager, consulte Registro de llamadas a la API de AWS Systems Manager con AWS CloudTrail.
- centralizar el monitoreo mediante la configuración de Amazon EventBridge y Amazon Simple Notification Service (Amazon SNS) para que envíen notificaciones sobre la correcta ejecución de los servicios

  Para obtener información acerca de cómo utilizar EventBridge para monitorear los eventos de Systems Manager, consulte Monitoreo de eventos de Systems Manager con Amazon EventBridge.

Acerca de los nodos administrados

Una vez que haya terminado de configurar las máquinas que no son de EC2 para Systems Manager, tal y como se ha indicado en esta sección, las máquinas activadas de manera híbrida se enumerarán en la AWS Management Console y se describirán como nodos administrados. En la consola, los ID de los nodos administrados activados de manera híbrida se diferencian de las instancias de Amazon EC2 por el prefijo “mi-”. Los ID de las instancias de Amazon EC2 utilizan el prefijo “i-”.

Un nodo administrado es cualquier máquina configurada para Systems Manager. Anteriormente, todos los nodos administrados se denominaban instancias administradas. El término instancia ahora refiere únicamente a las instancias de EC2. El comando deregister-managed-instance se nombró antes de este cambio de terminología.

Para obtener más información, consulte Trabajo con nodos administrados.

Acerca de las capas de instancia

Systems Manager ofrece un nivel de instancias estándar y un nivel de instancias avanzadas para los nodos administrados que no son de EC2 en el entorno híbrido y multinube. El nivel de instancias estándar le permite registrar un máximo de 1000 máquinas activadas de manera híbrida por Cuenta de AWS por Región de AWS. Si tiene que registrar más de 1000 máquinas que no son de EC2 en una única cuenta y región, utilice el nivel de instancias avanzadas. Las instancias avanzadas también le permiten conectarse a las máquinas que no son de EC2 mediante AWS Systems Manager Session Manager. Session Manager proporciona acceso mediante el intérprete de comandos interactivo de los nodos administrados.

Para obtener más información, consulte Configuración de los niveles de instancias.

Temas

- Creación del rol de servicio de IAM requerido para Systems Manager en entornos híbridos y multinube
- Creación de una activación híbrida para registrar nodos con Systems Manager
- Instalar SSM Agent en nodos de Linux híbridos
- Instalación de SSM Agent en nodos híbridos de Windows Server

Creación del rol de servicio de IAM requerido para Systems Manager en entornos híbridos y multinube

Los equipos que no son de EC2 (Amazon Elastic Compute Cloud) que se encuentran en un entorno híbrido y multinube necesitan un rol de servicio de AWS Identity and Access Management (IAM) para comunicarse con el servicio AWS Systems Manager. El rol concede AWS Security Token Service (AWS STS) `AssumeRole` confianza en el servicio de Systems Manager. Solo tiene que crear un rol de servicio para un entorno híbrido y multinube una vez para cada Cuenta de AWS. Sin embargo, puede elegir crear varios roles de servicio para distintas activaciones híbridas si los equipos del entorno híbrido y multinube requieren permisos distintos.

Los siguientes procedimientos describen cómo crear el rol de servicio necesario mediante la consola de Systems Manager o la herramienta de la línea de comandos que prefiera.

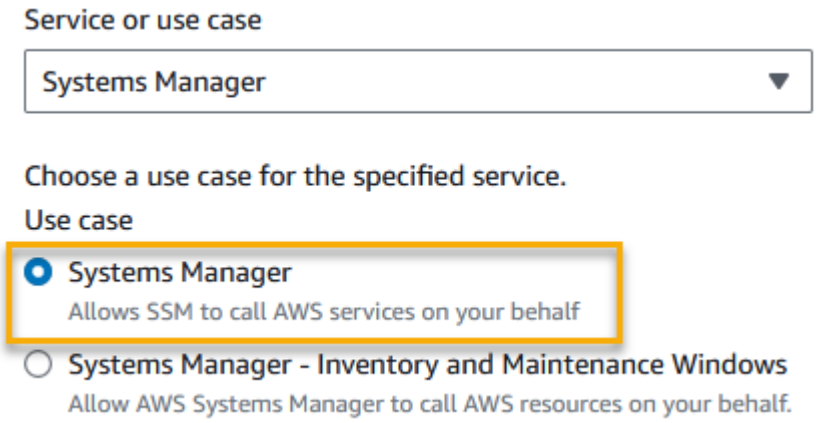
Uso de AWS Management Console para crear un rol de servicio de IAM para activaciones híbridas de Systems Manager

Utilice el siguiente procedimiento para crear un rol de servicio para activación híbrida. Este procedimiento utiliza la política `AmazonSSMManagedInstanceCore` para la funcionalidad principal de Systems Manager. En función del caso de uso, es posible que tenga que agregar políticas adicionales al rol de servicio para las máquinas en las instalaciones para poder acceder a otras herramientas de Systems Manager o Servicios de AWS. Por ejemplo, sin acceso a los buckets de Amazon Simple Storage Service (Amazon S3) requeridos administrados por AWS, las operaciones de aplicación de revisiones de Patch Manager fallan.

Para crear un rol de servicio de (consola)

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, seleccione Roles y luego seleccione Crear rol.
3. En Select trusted entity (Seleccionar entidad de confianza), realice las siguientes elecciones:
    1. En Tipo de entidad de confianza, elija Servicio de AWS.
    2. En Casos de uso de otros Servicios de AWS, elija Systems Manager.
    3. Elija Systems Manager.

    La siguiente imagen resalta la ubicación de la opción Systems Manager.



4. Elija Siguiente.
5. En la página Agregar permisos, haga lo siguiente:

- Utilice el campo Search (Buscar) para localizar la política AmazonSSMManagedInstanceCore. Seleccione la casilla de verificación situada junto a su nombre, como se muestra en la siguiente ilustración.



> **Note**
>
> La consola conserva la selección aunque busque otras políticas.

- Si ha creado una política de bucket de S3 personalizada en el procedimiento (Opcional) Crear una política personalizada para el acceso al bucket de S3, búsquela y seleccione la casilla de verificación situada junto a su nombre.
- Si tiene previsto unir equipos que no sean de EC2 a una instancia de Active Directory administrada por AWS Directory Service, busque AmazonSSMDirectoryServiceAccess y seleccione la casilla de verificación situada junto a su nombre.
- Si tiene previsto utilizar EventBridge o los Registros de CloudWatch para administrar o supervisar el nodo administrado, busque CloudWatchAgentServerPolicy y seleccione la casilla de verificación situada junto a su nombre.

6. Elija Siguiente.
7. En Nombre del rol, ingrese un nombre para el rol de servidor de IAM nuevo, por ejemplo, **`SSMServerRole`**.

> **Note**
>
> Anote el nombre del rol. Elegirá este rol cuando registre equipos nuevos que desee administrar mediante Systems Manager.

8. (Opcional) En Descripción, actualice la descripción de este rol de servidor de IAM.
9. (Opcional) En Tags (Etiquetas), agregue uno o varios pares de valor etiqueta-clave para organizar, realizar un seguimiento o controlar el acceso a este rol.
10. Elija Create role. El sistema le devuelve a la página Roles.

Uso de AWS CLI para crear un rol de servicio de IAM para activaciones híbridas de Systems Manager

Utilice el siguiente procedimiento para crear un rol de servicio para activación híbrida. Este procedimiento utiliza la política `AmazonSSMManagedInstanceCore` de la funcionalidad principal de Systems Manager. En función del caso de uso, es posible que tenga que agregar políticas adicionales al rol de servicio para las máquinas que no sean de EC2 en un entorno híbrido y multinube para poder acceder a otras herramientas u otros Servicios de AWS.

Requisito de política de bucket de S3

Si cualquiera de los siguientes casos es correcto, debe crear una política de permiso de IAM personalizada para los buckets de Amazon Simple Storage Service (Amazon S3) antes de completar este procedimiento:

- Caso 1: está utilizando un punto de conexión de VPC para conectar de forma privada su VPC a Servicios de AWS y servicios de punto de conexión de VPC con tecnología de AWS PrivateLink.
- Caso 2: tiene previsto utilizar un bucket de Amazon S3 que ha creado como parte de sus operaciones de Systems Manager para, por ejemplo, almacenar salidas de comandos de Run Command o sesiones de Session Manager en un bucket de S3. Antes de continuar, siga los pasos en Creación de una política de bucket de S3 personalizada para un perfil de instancias. La información acerca de las políticas del bucket de S3 que aparece en este tema también se aplica a su rol de servicio.

AWS CLI

Para crear un rol de servicio de IAM para un entorno híbrido y multinube (AWS CLI)

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. En el equipo local, cree un archivo de texto con un nombre como `SSMService-Trust.json` con la siguiente política de confianza. Asegúrese de guardar el archivo con la extensión `.json`. Asegúrese de especificar la Cuenta de AWS y la Región de AWS en el ARN en el que creó la activación híbrida. Reemplace los *`valores del marcador de posición`* de los campos de ID de cuenta y región con su propia información.

JSON

```
{
   "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. Abra la AWS CLI, y en el directorio en el que creó el archivo JSON, ejecute el comando create-role para crear el rol de servicio. Este ejemplo crea un rol llamado `SSMServiceRole`. Puede elegir otro nombre si lo prefiere.

   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. Ejecute el comando attach-role-policy de la siguiente forma para permitir que la función de servicio que acaba de crear genere un token de sesión. El token de sesión concede permiso al nodo administrado para ejecutar comandos mediante Systems Manager.

> **Note**
>
> Las políticas que agrega a un perfil de servicio para nodos administrados en un entorno híbrido y multinube son las mismas políticas utilizadas para crear un perfil de instancia para instancias de Amazon Elastic Compute Cloud (Amazon EC2). Para obtener más información sobre las políticas de AWS utilizadas en los siguientes comandos, consulte Configuración de permisos de instancia requeridos para Systems Manager.

(Obligatorio) Ejecute el siguiente comando para permitir que un nodo administrado utilice la funcionalidad principal del servicio de AWS Systems Manager.

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 ha creado una política de bucket de S3 personalizada para su rol de servicio, ejecute el siguiente comando para permitir el acceso de AWS Systems Manager Agent (SSM Agent) a los buckets que especificó en la política. Sustituya el *account-id* y el *amzn-s3-demo-bucket* con el ID de su Cuenta de AWS y el nombre de su bucket.

Linux & macOS

```
aws iam attach-role-policy \
    --role-name SSMServiceRole \
```

```
    --policy-arn arn:aws:iam::account-id:policy/amzn-s3-demo-bucket
```

Windows

```
aws iam attach-role-policy ^
    --role-name SSMServiceRole ^
    --policy-arn arn:aws:iam::account-id:policy/amzn-s3-demo-bucket
```

(Opcional) Ejecute el siguiente comando para permitir a SSM Agent el acceso a AWS Directory Service en su nombre para las solicitudes de unión al dominio por parte del nodo administrado. El rol de servicio solo necesita esta política si une los nodos a un directorio de 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
```

(Opcional) Ejecute el siguiente comando para permitir que el agente de CloudWatch se ejecute en los nodos administrados. Este comando permite la lectura de información en un nodo y su escritura en CloudWatch. Su perfil de servicio solo precisa de esta política si hace uso de servicios, como Amazon EventBridge o Registros de Amazon CloudWatch.

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

Tools for PowerShell

Para crear un rol de servicio de IAM para un entorno híbrido y multinube (AWS Tools for Windows PowerShell)

1. Instale y configure Herramientas de AWS para PowerShell (Herramientas para Windows PowerShell), si aún no lo ha hecho.

   Para obtener más información, consulte Instalación de Herramientas de AWS para PowerShell.

2. En el equipo local, cree un archivo de texto con un nombre como `SSMService-Trust.json` con la siguiente política de confianza. Asegúrese de guardar el archivo con la extensión `.json`. Asegúrese de especificar la Cuenta de AWS y la Región de AWS en el ARN en el que creó la activación híbrida.

   JSON

```
{
    "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-1:123456789012:*"
                }
            }
        }
    ]
}
```

3. Abra PowerShell en modo administrativo y, en el directorio en el que creó el archivo JSON, ejecute New-IAMRole como se indica a continuación para crear una función de servicio. Este ejemplo crea un rol llamado `SSMServiceRole`. Puede elegir otro nombre si lo prefiere.

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

4. Utilice Register-IAMRolePolicy de la siguiente forma para permitir que el rol de servicio que ha creado genere un token de sesión. El token de sesión concede permiso al nodo administrado para ejecutar comandos mediante Systems Manager.

> **Note**
>
> Las políticas que agrega a un perfil de servicio para los nodos administrados en un entorno híbrido y multinube son las mismas políticas utilizadas para crear un perfil de instancia para instancias de EC2. Para obtener más información sobre las políticas de AWS utilizadas en los siguientes comandos, consulte Configuración de permisos de instancia requeridos para Systems Manager.

(Obligatorio) Ejecute el siguiente comando para permitir que un nodo administrado utilice la funcionalidad principal del servicio de AWS Systems Manager.

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

Si ha creado una política de bucket de S3 personalizada para su rol de servicio, ejecute el siguiente comando para permitir el acceso de SSM Agent a los buckets que especificó en la política. Sustituya el *`account-id`* y el *`my-bucket-policy-name`* con el ID de su Cuenta de AWS y el nombre de su bucket.

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

(Opcional) Ejecute el siguiente comando para permitir a SSM Agent el acceso a AWS Directory Service en su nombre para las solicitudes de unión al dominio por parte del nodo administrado. El rol de servidor solo necesita esta política si une los nodos a un directorio de Microsoft AD.

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

(Opcional) Ejecute el siguiente comando para permitir que el agente de CloudWatch se ejecute en los nodos administrados. Este comando permite la lectura de información en un nodo y su escritura en CloudWatch. Su perfil de servicio solo precisa de esta política si hace uso de servicios, como Amazon EventBridge o Registros de Amazon CloudWatch.

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

Siga en Creación de una activación híbrida para registrar nodos con Systems Manager.

Creación de una activación híbrida para registrar nodos con Systems Manager

Para configurar equipos distintos a las instancias de Amazon Elastic Compute Cloud (EC2) como nodos administrados para un entorno híbrido y multinube, cree y aplique una activación híbrida. Después de completar correctamente la activación, recibirá inmediatamente un código y un ID de activación en la parte superior de la página de la consola. Puede especificar esta combinación de código e ID cuando instale AWS Systems Manager SSM Agent en equipos que no sean de EC2 para su entorno híbrido y multinube. La combinación de código e ID proporciona un acceso seguro al servicio de Systems Manager desde sus nodos administrados.

> **Important**
>
> Systems Manager regresa inmediatamente el código e ID de activación a la consola o la ventana de comandos, en función de cómo haya creado la activación. Copie esta información

y guárdela en un lugar seguro. Si sale de la consola o cierra la ventana de comandos, podría perder esta información. Si la pierde, debe crear una nueva activación.

Acerca del vencimiento de la activación

Un vencimiento de la activación es un intervalo de tiempo en el que se pueden registrar máquinas locales en Systems Manager. Una activación que ha vencido no tiene ningún impacto en los servidores ni en las máquinas virtuales que haya registrado en Systems Manager. Si una activación ha vencido, no se podrán registrar más servidores ni máquinas virtuales en Systems Manager mediante esa activación específica. Solo es necesario crear una nueva.

Cada servidor y máquina virtual en las instalaciones que ya haya registrado permanecen registrados como un nodo administrado de Systems Manager hasta que anule el registro explícitamente. Puede anular el registro de un nodo administrado que no sea de EC2 de las siguientes maneras:

- Utilice la pestaña Nodos administrados en Fleet Manager en la consola de Systems Manager.
- Utilice el comando de la AWS CLI deregister-managed-instance.
- Utilice la acción de la API DeregisterManagedInstance.

Para obtener más información, consulte los temas siguientes:

- Anulación del registro y nuevo registro de un nodo administrado (Linux)
- Anulación del registro y nuevo registro de un nodo administrado (Windows Server)

Acerca de los nodos administrados

Un nodo administrado es cualquier máquina configurada para AWS Systems Manager. AWS Systems Manager admite instancias de Amazon Elastic Compute Cloud (Amazon EC2), dispositivos periféricos y servidores o VM en las instalaciones, incluidas VM de otros entornos en la nube. Anteriormente, todos los nodos administrados se denominaban instancias administradas. El término instancia ahora refiere únicamente a las instancias de EC2. El comando deregister-managed-instance se nombró antes de este cambio de terminología.

Acerca de las etiquetas de activación

Si crea una activación mediante la AWS Command Line Interface (AWS CLI) o las AWS Tools for Windows PowerShell, puede especificar etiquetas. Las etiquetas son metadatos opcionales que

usted asigna a un recurso. Las etiquetas permiten clasificar los recursos de diversas maneras, por ejemplo, según la finalidad, el propietario o el entorno. Este es un ejemplo de un comando de la AWS CLI que se puede ejecutar en el este de EE. UU. (Ohio) en un equipo Linux local que incluye etiquetas opcionales.

```
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 especifica etiquetas al crear una activación, esas etiquetas se asignarán automáticamente a sus nodos administrados cuando los active.

No se pueden añadir ni eliminar etiquetas de una activación existente. Si no desea asignar etiquetas automáticamente a sus servidores y máquinas virtuales locales mediante una activación, puede añadirles etiquetas más adelante. En concreto, puede etiquetar los servidores locales y las máquinas virtuales después de que se conecten a Systems Manager por primera vez. Después de conectarse, se les asigna un ID de nodo administrado que se muestra en la consola de Systems Manager con un ID que lleva el prefijo “mi-”.

> **Note**
>
> No puede asignar etiquetas a una activación si se crea mediante la consola de Systems Manager. Para crearla, utilice la AWS CLI o Tools for Windows PowerShell.

Si ya no desea administrar un servidor local o una máquina virtual mediante Systems Manager, puede anular el registro. Para obtener más información, consulta Anulación del registro de nodos administrados en un entorno híbrido y multinube.

Temas

- Uso de la AWS Management Console para crear una activación para registrar nodos administrados con Systems Manager
- Uso de la línea de comandos para crear una activación para registrar nodos administrados con Systems Manager

Uso de la AWS Management Console para crear una activación para registrar nodos administrados con Systems Manager

Para crear una activación de nodo administrado

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Hybrid Activations (Activaciones híbridas).
3. Elija Create activation (Crear activación).

   -o bien-

   Si va a acceder a Hybrid Activations (Activaciones híbridas) por primera vez en la Región de AWS actual, elijaCreate an Activation (Crear una activación).
4. (Opcional) En el campo Activation description (Descripción de la activación), ingrese una descripción para esta activación. Le recomendamos que ingrese una descripción si tiene previsto activar un gran número de servidores y máquinas virtuales.
5. En Instance limit (Límite de instancias), especifique el número total de nodos que desea registrar con AWS como parte de esta activación. El valor predeterminado es 1 instancia.
6. En IAM role (Rol de IAM), elija una opción de rol de servicio que permita que los servidores y las máquinas virtuales se comuniquen con AWS Systems Manager en la nube:

   - Opción 1: elija Use the default role created by the system (Utilizar el rol predeterminado creado por el sistema) para utilizar un rol y una política administrada proporcionada por AWS.
   - Opción 2: elija Select an existing custom IAM role that has the required permissions (Seleccionar un rol de IAM personalizado existente que tenga los permisos requeridos) para utilizar el rol personalizado opcional que ha creado anteriormente. Este rol debe tener una política de relación de confianza que especifique `"Service": "ssm.amazonaws.com"`. Si su rol de IAM no especifica este principio en una política de relación de confianza, recibirá el siguiente error:

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

     Para obtener más información sobre la creación de este rol, consulte Creación del rol de servicio de IAM requerido para Systems Manager en entornos híbridos y multinube.

7. En Activation expiry date (Fecha de vencimiento de la activación), especifique una fecha de vencimiento para la activación. La fecha de vencimiento debe ser en el futuro y no más de 30 días en el futuro. El valor de predeterminado es 24 horas.

   > **Note**
   >
   > Si desea registrar nodos administrados adicionales después de la fecha de vencimiento, debe crear una nueva activación. La fecha de vencimiento no afecta los nodos registrados y en ejecución.

8. (Opcional) En el campo Default instance name (Nombre de instancia predeterminado), especifique un valor de nombre identificativo para mostrarlo en todos los nodos administrados asociados a esta activación.
9. Elija Create activation (Crear activación). Systems Manager regresa inmediatamente el ID y el código de activación a la consola.

Uso de la línea de comandos para crear una activación para registrar nodos administrados con Systems Manager

En el siguiente procedimiento se describe cómo utilizar la AWS Command Line Interface (AWS CLI) (en Linux o Windows Server) o Herramientas de AWS para PowerShell para crear una activación de nodo administrado.

Para crear una activación

1. Si aún no lo ha hecho, instale y configure la AWS CLI o las Herramientas de AWS para PowerShell.

   Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.
2. Ejecute el siguiente comando para crear una activación.

   > **Note**
   >
   > - En el siguiente comando, reemplace *region* con su propia información. Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de

conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

- El rol que especifique para el parámetro *iam-role* debe tener una política de relación de confianza que especifique `"Service": "ssm.amazonaws.com"`. Si su rol de AWS Identity and Access Management (IAM) no especifica este principio en una política de relación de confianza, recibirá el siguiente error:

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

  Para obtener más información sobre la creación de este rol, consulte Creación del rol de servicio de IAM requerido para Systems Manager en entornos híbridos y multinube.

- Para `--expiration-date`, proporcione una fecha en formato de marca temporal, como `"2021-07-07T00:00:00"`, para cuando el código de activación llegue a su vencimiento. Puede especificar una fecha con hasta 30 días de antelación. Si no proporciona una fecha de vencimiento, el código de activación se vencerá en 24 horas.

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"}
```

A continuación se muestra un ejemplo.

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 la activación se crea correctamente, el sistema devuelve inmediatamente un código y un ID de activación.

Instalar SSM Agent en nodos de Linux híbridos

En este tema se describe cómo instalar AWS Systems Manager SSM Agent en máquinas Linux que no son EC2 (Amazon Elastic Compute Cloud) en un entorno híbrido y multinube. Para obtener información acerca de la instalación de SSM Agent en instancias de EC2 para Linux, consulte Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux.

Antes de comenzar, localice el código y el ID de activación que se generaron durante el proceso de activación híbrida, tal y como se describe en Creación de una activación híbrida para registrar nodos con Systems Manager. Deberá especificar el código y el ID en el siguiente procedimiento.

Instalación de SSM Agent en máquinas que no son EC2 en un entorno híbrido y multinube

1. Inicie sesión en un servidor o una máquina virtual del entorno híbrido y multinube.
2. Si utiliza un proxy HTTP o HTTPS, debe establecer las variables de entorno `http_proxy` o `https_proxy` en la sesión de shell actual. Si no utiliza un proxy, puede omitir este paso.

   Ingrese los siguientes comandos en la línea de comandos en el caso de un servidor proxy HTTP:

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

   Ingrese los siguientes comandos en la línea de comandos en el caso de un servidor proxy HTTPS:

   ```
   export http_proxy=http://hostname:port
   export https_proxy=https://hostname:port
   ```
3. Copie y pegue uno de los siguientes bloques de comandos en SSH. Reemplace los valores de marcador por el código y el ID de activación que se generaron durante el proceso de activación

híbrido y por el identificador de la Región de AWS que desea descargar SSM Agent, y luego presione `Enter`.

> **Important**
>
> Tenga en cuenta los siguientes detalles importantes:
>
> - El uso de `ssm-setup-cli` en instalaciones que no son de EC2 maximiza la seguridad de la instalación y la configuración de Systems Manager.
> - `sudo` no es necesario si es un usuario raíz.
> - Descarga `ssm-setup-cli` desde Región de AWS en el mismo lugar donde se creó la activación híbrida.
> - `ssm-setup-cli` admite una opción `manifest-url` que determina la fuente desde la que se descarga el agente. No especifique un valor para esta opción a menos que la organización lo requiera.
> - Utilice únicamente el enlace de descarga proporcionado para `ssm-setup-cli` cuando registre instancias. No debe almacenar `ssm-setup-cli` por separado para su uso futuro.
> - Puede utilizar el script que se proporciona aquí para validar la firma de `ssm-setup-cli`.

`region` representa el identificador de Región de AWS compatible con AWS Systems Manager, como `us-east-2` para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

Además, `ssm-setup-cli` incluye las siguientes opciones:

- `version`: los valores válidos son `latest` y `stable`.
- `downgrade`: permite el cambio del SSM Agent a una versión anterior. Especifique `true` si desea instalar una versión anterior del agente.
- `skip-signature-validation`: omite la validación de la firma durante la descarga e instalación del agente.

RHEL 6.x y 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.x, Oracle Linux, CentOS 7.x y 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 y 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 (anteriormente 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 Server

- Uso de paquetes .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"
  ```

- Uso de paquetes Snap

  No es necesario especificar una URL para la descarga, ya que el comando `snap` descarga automáticamente el agente en la tienda de aplicaciones de Snap en https://snapcraft.io.

  En Ubuntu Server 20.10 STR y 20.04, 18.04 y 16.04 LTS, los archivos del instalador de SSM Agent, incluidos los archivos binarios y de configuración del agente, se almacenan en el siguiente directorio: `/snap/amazon-ssm-agent/current/`. Si realiza cambios en cualquiera de los archivos de configuración de este directorio, debe copiar estos archivos desde el directorio `/snap` al directorio `/etc/amazon/ssm/`. Los archivos de registros y bibliotecas no han cambiado (`/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

El canal candidato en el almacén de Snap contiene la versión más reciente de SSM Agent; no el canal estable. Si desea realizar un seguimiento de información de la versión de SSM Agent en el canal candidato, ejecute el siguiente comando en los nodos administrados de 64 bits de Ubuntu Server 18.04 y 16.04 LTS.

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

El comando se descarga e instala SSM Agent en la máquina activada de manera híbrida en su entorno híbrido y multinube. El comando detiene SSM Agent y, a continuación, registra la máquina virtual en el servicio de Systems Manager. El equipo ahora es un nodo administrado. Las instancias de Amazon EC2 configuradas para Systems Manager también son nodos administrados. Sin embargo, en la consola de Systems Manager, sus nodos activados de manera híbrida se distinguen de las instancias de Amazon EC2 con el prefijo “mi-”.

Siga en Instalación de SSM Agent en nodos híbridos de Windows Server.

Configuración de la rotación automática de clave privada

Para reforzar su posición de seguridad, puede configurar AWS Systems Manager Agent (SSM Agent) para rotar automáticamente la clave privada del entorno híbrido y multinube. Puede acceder a esta característica mediante la versión 3.0.1031.0 o posterior de SSM Agent. Active esta característica siguiendo el procedimiento que se describe a continuación.

Para configurar SSM Agent para rotar la clave privada del entorno híbrido y multinube

1. Vaya a `/etc/amazon/ssm/` en un equipo Linux o a `C:\Program Files\Amazon\SSM` para un equipo Windows.
2. Copie los contenidos de `amazon-ssm-agent.json.template` en un archivo nuevo denominado `amazon-ssm-agent.json`. Guarde `amazon-ssm-agent.json` en el mismo directorio donde se encuentra `amazon-ssm-agent.json.template`.

3. Encuentre`Profile`, `KeyAutoRotateDays`. Ingrese el número de días que desea entre las rotaciones automáticas de clave privada.
4. Reinicie SSM Agent.

Cada vez que cambie la configuración, reinicie SSM Agent.

Puede personalizar otras características de SSM Agent mediante el mismo procedimiento. Para ver la lista actualizada de las propiedades de configuración disponibles y sus valores predeterminados, consulte Definiciones de propiedades de configuración.

Anulación del registro y nuevo registro de un nodo administrado (Linux)

Puede anular el registro de un nodo administrado activado de manera híbrida mediante una llamada a la operación de la API DeregisterManagedInstance desde la AWS CLI o desde Herramientas para Windows PowerShell. A continuación, se muestra un ejemplo de comando de la CLI:

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

Para eliminar el resto de la información de registro del agente, elimine la clave `IdentityConsumptionOrder` del archivo `amazon-ssm-agent.json`. A continuación, en función del tipo de instalación, ejecute uno de los siguientes comandos.

En nodos de Ubuntu Server en los que se instaló SSM Agent mediante paquetes de Snap:

```
sudo /snap/amazon-ssm-agent/current/amazon-ssm-agent -register -clear
```

En todas las demás instalaciones de Linux:

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

Para volver a registrar un nodo administrado en una máquina que no es de EC2 Linux

Puede volver a registrar una máquina después de anular el registro. Sin embargo, debe utilizar un código y un ID de activación diferentes a los utilizados anteriormente para registrarla.

1. Conéctese a su máquina.
2. Ejecute el siguiente comando. Asegúrese de sustituir los valores de marcador por el código y el ID de activación que se generan cuando crea una activación de nodo administrado y por el identificador de la región de la que desea descargar el SSM Agent.

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

Solución de problemas de instalación de SSM Agent en máquinas Linux que no son de EC2

Utilice la siguiente información como ayuda para solucionar problemas de instalación de SSM Agent en máquinas Linux activadas de manera híbrida en un entorno híbrido y multinube.

Recibe el error DeliveryTimedOut

Problema: cuando se configura una máquina en una Cuenta de AWS como un nodo administrado para una Cuenta de AWS separada, recibe `DeliveryTimedOut` después de ejecutar los comandos para instalar SSM Agent en la máquina de destino.

Solución: `DeliveryTimedOut` es el código de respuesta esperado para este escenario. El comando para instalar SSM Agent en el nodo de destino cambia el ID de nodo del nodo de origen. Debido a que el ID de nodo ha cambiado, el nodo de origen no puede responder al nodo de destino que el comando falló, se completó o agotó el tiempo de espera durante la ejecución.

No se pueden cargar las asociaciones de nodos

Problema: después de ejecutar los comandos de instalación, verá el siguiente error en los registros de errores de SSM Agent:

```
Unable to load instance associations, unable to retrieve
associations unable to retrieve associations error occurred in
RequestManagedInstanceRoleToken: MachineFingerprintDoesNotMatch:
Fingerprint doesn't match
```

Se muestra este error cuando el ID del equipo no persiste después de su reinicio.

Solución: para solucionar este problema, ejecute el siguiente comando. Este comando obliga a que el ID del equipo persista después de su reinicio.

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

Instalación de SSM Agent en nodos híbridos de Windows Server

En este tema se describe cómo instalar AWS Systems Manager SSM Agent en equipos Windows Server para un entorno híbrido y multinube. Para obtener información acerca de la instalación de SSM Agent en instancias de EC2 para Windows Server, consulte Cómo instalar y desinstalar de forma manual SSM Agent en instancias de EC2 para Windows Server.

Antes de comenzar, localice el código y el ID de activación que se generaron durante el proceso de activación híbrida, tal y como se describe en Creación de una activación híbrida para registrar nodos con Systems Manager. Deberá especificar el código y el ID en el siguiente procedimiento.

Instalación de SSM Agent en equipos Windows Server que no sean de EC2 en un entorno híbrido y multinube

1. Inicie sesión en un servidor o una máquina virtual del entorno híbrido y multinube.
2. Si utiliza un proxy HTTP o HTTPS, debe establecer las variables de entorno `http_proxy` o `https_proxy` en la sesión de shell actual. Si no utiliza un proxy, puede omitir este paso.

   Para un servidor proxy HTTP, configure esta variable:

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

   Para un servidor proxy HTTPS, configure esta variable:

   ```
   http_proxy=http://hostname:port
   https_proxy=https://hostname:port
   ```
3. Abra Windows PowerShell en modo (administrativo) con permisos elevados.
4. Copie y pegue el siguiente bloque de comandos en Windows PowerShell. Reemplace cada *`example resource placeholder`* con su propia información. Por ejemplo, el código de activación y el ID de activación generados cuando se crea una activación híbrida y con el identificador de la Región de AWS desde la que desea descargar SSM Agent.

   > **Important**
   >
   > Tenga en cuenta los siguientes detalles importantes:

- El uso de `ssm-setup-cli` en instalaciones que no son de EC2 maximiza la seguridad de la instalación y la configuración de Systems Manager.
- `ssm-setup-cli` admite una opción `manifest-url` que determina la fuente desde la que se descarga el agente. No especifique un valor para esta opción a menos que la organización lo requiera.
- Puede utilizar el script que se proporciona aquí para validar la firma de `ssm-setup-cli`.
- Utilice únicamente el enlace de descarga proporcionado para `ssm-setup-cli` cuando registre instancias. No debe almacenar `ssm-setup-cli` por separado para su uso futuro.

`region` representa el identificador de Región de AWS compatible con AWS Systems Manager, como `us-east-2` para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

Además, `ssm-setup-cli` incluye las siguientes opciones:

- `version`: los valores válidos son `latest` y `stable`.
- `downgrade`: revierte el agente a una versión anterior.
- `skip-signature-validation`: omite la validación de la firma durante la descarga e instalación del agente.

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. Pulse `Enter`.

> **Note**
>
> Si el comando falla, verifique que está ejecutando la última versión de Herramientas de AWS para PowerShell.

El comando hace lo siguiente:

- Descarga e instala SSM Agent en el equipo.
- Registra la máquina en el servicio de Systems Manager.
- Devuelve una respuesta a la solicitud similar a la siguiente:

```
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
```

El equipo ahora es un nodo administrado. Ahora, estos nodos administrados están identificados con el prefijo "mi-". Puede ver los nodos administrados en la página Nodos administrados en Fleet Manager, con el comando de la AWS CLI describe-instance-information o con el comando de la API DescribeInstanceInformation.

Configuración de la rotación automática de clave privada

Para reforzar su posición de seguridad, puede configurar AWS Systems Manager Agent (SSM Agent) para rotar de forma automática la clave privada de un entorno híbrido y multinube. Puede acceder a esta característica mediante la versión 3.0.1031.0 o posterior de SSM Agent. Active esta característica siguiendo el procedimiento que se describe a continuación.

Para configurar SSM Agent para rotar la clave privada del entorno híbrido y multinube

1. Vaya a `/etc/amazon/ssm/` en un equipo Linux o a `C:\Program Files\Amazon\SSM` para un equipo Windows Server.
2. Copie los contenidos de `amazon-ssm-agent.json.template` en un archivo nuevo denominado `amazon-ssm-agent.json`. Guarde `amazon-ssm-agent.json` en el mismo directorio donde se encuentra `amazon-ssm-agent.json.template`.
3. Encuentre`Profile`, `KeyAutoRotateDays`. Ingrese el número de días que desea entre las rotaciones automáticas de clave privada.
4. Reinicie SSM Agent.

Cada vez que cambie la configuración, reinicie SSM Agent.

Puede personalizar otras características de SSM Agent mediante el mismo procedimiento. Para ver la lista actualizada de las propiedades de configuración disponibles y sus valores predeterminados, consulte Definiciones de propiedades de configuración.

Anulación del registro y nuevo registro de un nodo administrado (Windows Server)

Puede anular el registro de un nodo administrado mediante una llamada a la operación de la API DeregisterManagedInstance desde la AWS CLI o desde Tools for Windows PowerShell. A continuación, se muestra un ejemplo de comando de la CLI:

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

Para eliminar el resto de la información de registro del agente, elimine la clave `IdentityConsumptionOrder` del archivo `amazon-ssm-agent.json`. A continuación, ejecute el siguiente comando:

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

Para volver a registrar un nodo administrado en un equipo híbrido Windows Server

Puede volver a registrar una máquina después de anular el registro. Sin embargo, debe utilizar un código y un ID de activación diferentes a los utilizados anteriormente para registrarla.

1. Conéctese a su máquina.
2. Ejecute el siguiente comando. Asegúrese de sustituir los valores de marcador por el código y el ID de activación que se generan cuando crea una activación híbrida y por el identificador de la región desde la que desea descargar el 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"
```

Administración de dispositivos periféricos con Systems Manager

En esta sección se describen las tareas de configuración que la cuenta y los administradores de sistemas realizan para habilitar la configuración y la administración de dispositivos de núcleo de AWS IoT Greengrass. Después de completar estas tareas, los usuarios a los que el administrador de Cuenta de AWS les ha concedido permisos pueden utilizar AWS Systems Manager para configurar y administrar los dispositivos de núcleo de AWS IoT Greengrass.

> **Note**
>
> - SSM Agent para AWS IoT Greengrass no es compatible en macOS ni en Windows 10. No puede utilizar las herramientas de Systems Manager para administrar y configurar dispositivos de periferia que utilizan estos sistemas operativos.
> - Systems Manager también admite dispositivos de borde que no están configurados como dispositivos de núcleo de AWS IoT Greengrass. Para utilizar Systems Manager para administrar dispositivos de núcleo de AWS IoT y dispositivos de periferia que no sean de AWS, debe configurarlos mediante una activación híbrida Para obtener más información, consulte Administrador de nodos en entornos híbridos y multinube con Systems Manager.
> - Para utilizar Session Manager y la aplicación de revisiones de aplicaciones de Microsoft con los dispositivos de borde, debe habilitar el nivel de instancias avanzadas. Para obtener más información, consulte Activación del nivel de instancias avanzadas.

Antes de empezar

Compruebe que los dispositivos de borde cumplen los siguientes requisitos.

- Los dispositivos de borde deben cumplir los requisitos para configurarse como dispositivos de núcleo de AWS IoT Greengrass. Para obtener más información, consulte Configuración de dispositivos de núcleo de AWS IoT Greengrass en la Guía para desarrolladores de AWS IoT Greengrass Version 2.
- Los dispositivos de borde deben ser compatibles con el agente de AWS Systems Manager (SSM Agent). Para obtener más información, consulte Sistemas operativos compatibles con Systems Manager.
- Los dispositivos de borde deben poder comunicarse con el servicio de Systems Manager en la nube. Systems Manager no admite dispositivos de borde desconectados.

Acerca de la configuración de dispositivos de borde

La configuración de dispositivos de AWS IoT Greengrass para Systems Manager implica los siguientes procesos.

> **Note**
>
> Para obtener información sobre cómo desinstalar SSM Agent de un dispositivo perimetral, consulte Desinstalar AWS Systems Manager Agent en la AWS IoT Greengrass Version 2Guía para desarrolladores.

Creación de un rol de servicio de IAM para dispositivos periféricos

Los dispositivos de núcleo de AWS IoT Greengrass requieren un rol de servicio de AWS Identity and Access Management (IAM) para comunicarse con AWS Systems Manager. El rol concede AWS Security Token Service (AWS STS) AssumeRole confianza en el servicio de Systems Manager. Solo tiene que crear un rol de servicio una vez por cada Cuenta de AWS. Especificará este rol para el parámetro `RegistrationRole` cuando configure e implemente el componente de SSM Agent a los dispositivos de AWS IoT Greengrass. Si ya creó este rol mientras configuraba nodos que no son de EC2 para un entorno híbrido y multinube, puede omitir este paso.

> **Note**
>
> A los usuarios de la empresa u organización que van a utilizar Systems Manager en los dispositivos de borde se les debe conceder permiso en IAM para llamar a la API de Systems Manager.

Requisito de política de bucket de S3

Si cualquiera de los siguientes casos es correcto, debe crear una política de permiso de IAM personalizada para los buckets de Amazon Simple Storage Service (Amazon S3) antes de completar este procedimiento:

- Caso 1: está utilizando un punto de conexión de VPC para conectar de forma privada su VPC a Servicios de AWS y servicios de punto de conexión de VPC compatibles gestionados por AWS PrivateLink.
- Caso 2: planea usar un bucket S3 que crea como parte de sus operaciones de Administrador de sistemas, como para almacenar la salida para comandos Run Command o sesiones Session Manager en un bucket S3. Antes de continuar, siga los pasos en Creación de una política de bucket de S3 personalizada para un perfil de instancias. La información acerca de las políticas del bucket de S3 que aparece en este tema también se aplica a su rol de servicio.

> **Note**
>
> Si los dispositivos están protegidos por un firewall y planea utilizar Patch Manager, el firewall debe permitir el acceso al punto de conexión de la base de referencia de revisiones `arn:aws:s3:::patch-baseline-snapshot-`*`region`*`/*`.
> *`region`* representa el identificador de Región de AWS compatible con AWS Systems Manager, como `us-east-2` para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

AWS CLI

Para crear un rol de servicio de IAM para un entorno AWS IoT Greengrass (AWS CLI)

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. En el equipo local, cree un archivo de texto con un nombre como `SSMService-Trust.json` con la siguiente política de confianza. Asegúrese de guardar el archivo con la extensión `.json`.

   > **Note**
   >
   > Anote el nombre. Lo especificará cuando implemente SSM Agent en los dispositivos de núcleo de AWS IoT Greengrass.

   JSON

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

```
        },
        "Action": "sts:AssumeRole"
    }
}
```

3. Abra la AWS CLI, y en el directorio en el que creó el archivo JSON, ejecute el comando create-role para crear el rol de servicio. Reemplace cada *example resource placeholder* con su propia información.

   Linux y 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. Ejecute el comando attach-role-policy de la siguiente forma para permitir que la función de servicio que acaba de crear genere un token de sesión. El token de sesión concede permiso a los dispositivos de borde para ejecutar comandos mediante Systems Manager.

   > **Note**
   >
   > Las políticas que agrega a un perfil de servicio para los dispositivos de borde son las mismas políticas que las utilizadas para crear un perfil de instancias en instancias de Amazon Elastic Compute Cloud (Amazon EC2). Para obtener más información sobre las políticas de IAM utilizadas en los siguientes comandos, consulte Configuración de permisos de instancia requeridos para Systems Manager.

   (Obligatorio) Ejecute el siguiente comando para permitir que un dispositivo de borde utilice la funcionalidad principal del servicio de AWS Systems Manager.

   Linux y 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 ha creado una política de bucket de S3 personalizada para su rol de servicio, ejecute el siguiente comando para permitir el acceso de AWS Systems Manager Agent (SSM Agent) a los buckets que especificó en la política. Sustituya *account_ID* y *my_bucket_policy_name* con el ID de la Cuenta de AWS y el nombre del bucket.

Linux y 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
```

(Opcional) Ejecute el siguiente comando para permitir que SSM Agent acceda a AWS Directory Service en su nombre para las solicitudes de unión al dominio desde los dispositivos de borde. El rol de servicio solo necesita esta política si une los dispositivos de borde a un directorio de Microsoft AD.

Linux y 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
```

(Opcional) Ejecute el siguiente comando para permitir que el agente de CloudWatch se ejecute en los dispositivos de borde. Este comando permite la lectura de información en un dispositivo y su escritura en CloudWatch. El rol de servicio solo precisa de esta política si hace uso de servicios, como Amazon EventBridge o los Registros de Amazon CloudWatch.

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

Tools for PowerShell

Para crear un rol de servicio de IAM para un entorno AWS IoT Greengrass (AWS Tools for Windows PowerShell)

1. Instale y configure Herramientas de AWS para PowerShell (Herramientas para Windows PowerShell), si aún no lo ha hecho.

   Para obtener más información, consulte Instalación de Herramientas de AWS para PowerShell.

2. En el equipo local, cree un archivo de texto con un nombre como `SSMService-Trust.json` con la siguiente política de confianza. Asegúrese de guardar el archivo con la extensión `.json`.

   > **Note**
   >
   > Anote el nombre. Lo especificará cuando implemente SSM Agent en los dispositivos de núcleo de AWS IoT Greengrass.

   JSON

   ```
   {
   ```

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

3. Abra PowerShell en modo administrativo y, en el directorio en el que creó el archivo JSON, ejecute New-IAMRole como se indica a continuación para crear una función de servicio.

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

4. Utilice Register-IAMRolePolicy de la siguiente forma para permitir que el rol de servicio que ha creado genere un token de sesión. El token de sesión concede permiso a los dispositivos de borde para ejecutar comandos mediante Systems Manager.

> **Note**
>
> Las políticas que agrega a un rol de servicio para los dispositivos de borde en un entorno de AWS IoT Greengrass son las mismas políticas que las utilizadas para crear un perfil de instancias en instancias EC2. Para obtener más información sobre las políticas de AWS utilizadas en los siguientes comandos, consulte Configuración de permisos de instancia requeridos para Systems Manager.

(Obligatorio) Ejecute el siguiente comando para permitir que un dispositivo de borde utilice la funcionalidad principal del servicio de AWS Systems Manager.

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

Si ha creado una política de bucket de S3 personalizada para su rol de servicio, ejecute el siguiente comando para permitir el acceso de SSM Agent a los buckets que especificó en

la política. Sustituya *account_ID* y *my_bucket_policy_name* con el ID de la Cuenta de AWS y el nombre del bucket.

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

(Opcional) Ejecute el siguiente comando para permitir que SSM Agent acceda a AWS Directory Service en su nombre para las solicitudes de unión al dominio desde los dispositivos de borde. El rol de servicio solo necesita esta política si une los dispositivos de borde a un directorio de Microsoft AD.

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

(Opcional) Ejecute el siguiente comando para permitir que el agente de CloudWatch se ejecute en los dispositivos de borde. Este comando permite la lectura de información en un dispositivo y su escritura en CloudWatch. El rol de servicio solo precisa de esta política si hace uso de servicios, como Amazon EventBridge o los Registros de Amazon CloudWatch.

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

Configuración de dispositivos periféricos para AWS IoT Greengrass

Configure los dispositivos de borde como dispositivos de núcleo de AWS IoT Greengrass. El proceso de configuración implica verificar los sistemas operativos compatibles y los requisitos del sistema, así como instalar y configurar el software AWS IoT Greengrass Core en los dispositivos. Para obtener más información, consulte Configuración de dispositivos de núcleo de AWS IoT Greengrass en la Guía para desarrolladores de AWS IoT Greengrass Version 2.

Actualización del rol de intercambio de tokens de AWS IoT Greengrass e instalación de SSM Agent en dispositivos periféricos

El último paso para instalar y configurar los dispositivos básicos de AWS IoT Greengrass para Systems Manager consiste en actualizar el rol de servicio de dispositivo de AWS IoT Greengrass AWS Identity and Access Management (IAM), también denominado rol de intercambio de tokens, e implementar AWS Systems Manager Agent (SSM Agent) en los dispositivos de AWS IoT Greengrass. Para obtener información sobre estos procesos, consulte Instalación de AWS Systems Manager Agent en la Guía para desarrolladores de AWS IoT Greengrass Version 2.

Después de implementar SSM Agent en los dispositivos, AWS IoT Greengrass registra automáticamente los dispositivos con Systems Manager. No es necesario un registro adicional. Puede empezar a utilizar las herramientas de Systems Manager para acceder, administrar y configurar los dispositivos de AWS IoT Greengrass.

> **Note**
>
> Los dispositivos de borde deben poder comunicarse con el servicio de Systems Manager en la nube. Systems Manager no admite dispositivos de borde desconectados.

Creación de un administrador delegado de AWS Organizations para Systems Manager

Al configurar una organización en AWS Organizations, se asigna una cuenta de administración para realizar todas las tareas administrativas para todos Servicios de AWS. El usuario de la cuenta de administración solo puede asignar una cuenta de administrador delegado para que Systems Manager realice tareas administrativas para Change Manager, Explorer, y OpsCenter. AWS Organizations es un servicio de administración de cuentas que puede usar para crear una organización y asignar Cuentas de AWS para administrar estas cuentas de forma centralizada. Para obtener más información sobre AWS Organizations, consulte AWS Organizations en la Guía del usuario de AWS Organizations.

Change Manager, Explorer, y OpsCenter, herramientas de AWS Systems Manager, funcionan con AWS Organizations para llevar a cabo tareas en todas las cuentas de los miembros de su organización. Solo puede asignar un administrador delegado para todas las herramientas de

Systems Manager. La cuenta de administrador delegado debe ser un miembro de la unidad organizativa a la que esté asignada.

Temas

- Uso de un administrador delegado con Change Manager
- Uso de un administrador delegado con Explorer
- Uso de un administrador delegado con OpsCenter
- Uso de un administrador delegado con Quick Setup

Uso de un administrador delegado con Change Manager

Change Manager es un marco empresarial de administración de cambios con el que se pueden solicitar, aprobar, implementar e informar los cambios operativos de la configuración y la infraestructura de la aplicación.

Si utilizas Change Manager en una organización, asigna una cuenta de administrador delegada para administrar plantillas de cambios, aprobaciones e informes para todas las cuentas de miembros. Mediante Quick Setup, puede configurar Change Manager para usarlo con una organización y seleccionar la cuenta de administrador delegado. La cuenta de administrador delegado no es necesaria si Change Manager se utiliza solo con una única Cuenta de AWS.

De forma predeterminada, Change Manager muestra todas las tareas relacionadas con los cambios en la cuenta de administrador delegado. Para obtener instrucciones sobre cómo configurar un administrador delegado durante la configuración de Change Manager de una organización, consulte Configuración de Change Manager para una organización (cuenta de administración).

> **Important**
>
> Si utiliza Change Manager en toda una organización, se recomienda efectuar siempre los cambios desde la cuenta de administrador delegado. Si bien es posible realizar cambios desde otras cuentas de la organización, esos cambios no se notificarán ni se podrán ver desde la cuenta de administrador delegado.

Uso de un administrador delegado con Explorer

Explorer es un panel de operaciones personalizable que muestra una vista agregada de los datos de operaciones (OpsData) de sus Cuentas de AWS y en todas las Regiones de AWS.

Puede configurar una cuenta de administrador delegado para Systems Manager para agregar datos de Explorer de múltiples regiones y cuentas mediante el uso de la sincronización de datos de recursos con AWS Organizations. Un administrador delegado puede buscar, filtrar y agregar datos de Explorer mediante las teclas AWS Management Console, AWS Command Line Interface (AWS CLI) o AWS Tools for Windows PowerShell.

Un administrador delegado mejora la seguridad de Explorer al limitar el número de administradores que pueden crear o eliminar varias cuentas y sincronizar los datos de recursos de la región en una sola Cuenta de AWS.

Puede sincronizar los datos de las operaciones entre todas las Cuentas de AWS de su organización mediante Explorer. Para obtener información sobre cómo asignar un administrador delegado desde Explorer, consulte Configurar un administrador delegado para Explorer.

Uso de un administrador delegado con OpsCenter

OpsCenter proporciona una ubicación central donde los ingenieros de operaciones y los profesionales de TI pueden administrar los elementos de trabajo operativos (OpsItems) relacionados con los recursos de AWS. Si desea utilizar OpsCenter para gestionar, de forma centralizada, todas las cuentas OpsItems, debe configurar la organización en AWS Organizations.

Si usa Quick Setup para OpsCenter, puede asignar una cuenta de administrador delegado y configurar OpsCenter para administrar OpsItems de forma centralizada. Para obtener más información, consulte (Opcional) Configuración de OpsCenter para administrar OpsItems en todas las cuentas mediante Quick Setup.

Uso de un administrador delegado con Quick Setup

Quick Setup es una herramienta de Systems Manager que lo ayuda a configurar servicios y características de AWS utilizados con frecuencia mediante las prácticas recomendadas. Puede configurar una cuenta de administrador delegado para que Quick Setup lo ayude a implementar y administrar las configuraciones en todas las cuentas y regiones mediante AWS Organizations. Un administrador delegado de Quick Setup puede crear, actualizar, ver y eliminar los recursos del administrador de configuración de la organización. Systems Manager registra un administrador delegado de Quick Setup como parte del proceso de configuración de la experiencia de consola

integrada. Para obtener más información, consulte Configuración de la consola unificada de Systems Manager para una organización.

Configuración general para AWS Systems Manager

Si aún no lo ha hecho, regístrese para obtener una Cuenta de AWS y cree un usuario administrativo.

Cómo crear una Cuenta de AWS

Si no dispone de una Cuenta de AWS, siga estos pasos para crear una.

Procedimiento para registrarse en Cuenta de AWS

1. Abra https://portal.aws.amazon.com/billing/signup.
2. Siga las instrucciones que se le indiquen.

   Parte del procedimiento de registro consiste en recibir una llamada telefónica o mensaje de texto e indicar un código de verificación en el teclado del teléfono.

   Al registrarse en una Cuenta de AWS, se crea un Usuario raíz de la cuenta de AWS. El usuario raíz tendrá acceso a todos los Servicios de AWS y recursos de esa cuenta. Como práctica recomendada de seguridad, asigne acceso administrativo a un usuario y utilice únicamente el usuario raíz para realizar tareas que requieren acceso de usuario raíz.

AWS le enviará un correo electrónico de confirmación cuando complete el proceso de registro. Puede ver la actividad de la cuenta y administrarla en cualquier momento entrando en https://aws.amazon.com/ y seleccionando Mi cuenta.

Creación de un usuario con acceso administrativo

Después de registrarse para obtener una Cuenta de AWS, proteja su Usuario raíz de la cuenta de AWS, habilite AWS IAM Identity Center y cree un usuario administrativo para no usar el usuario raíz en las tareas cotidianas.

Protección de Usuario raíz de la cuenta de AWS

1. Inicie sesión en AWS Management Console como propietario de la cuenta; para ello, elija Usuario raíz e introduzca el correo electrónico de su Cuenta de AWS. En la siguiente página, escriba su contraseña.

Para obtener ayuda para iniciar sesión con el usuario raíz, consulte Iniciar sesión como usuario raíz en la Guía del usuario de AWS Sign-In.

2. Active la autenticación multifactor (MFA) para el usuario raíz.

   Para obtener instrucciones, consulte Habilitación de un dispositivo MFA virtual para su usuario raíz de la Cuenta de AWS (consola) en la Guía del usuario de IAM.

Creación de un usuario con acceso administrativo

1. Activar IAM Identity Center.

   Consulte las instrucciones en Activar AWS IAM Identity Center en la Guía del usuario de AWS IAM Identity Center.

2. En IAM Identity Center, conceda acceso administrativo a un usuario.

   Para ver un tutorial sobre cómo usar Directorio de IAM Identity Center como origen de identidad, consulte Configuración del acceso de los usuarios con el Directorio de IAM Identity Center predeterminado en la Guía del usuario de AWS IAM Identity Center.

Inicio de sesión como usuario con acceso de administrador

- Para iniciar sesión con el usuario de IAM Identity Center, use la URL de inicio de sesión que se envió a la dirección de correo electrónico cuando creó el usuario de IAM Identity Center.

  Para obtener ayuda para iniciar sesión con un usuario de IAM Identity Center, consulte Inicio de sesión en el portal de acceso de AWS en la Guía del usuario de AWS Sign-In.

Concesión de acceso a usuarios adicionales

1. En IAM Identity Center, cree un conjunto de permisos que siga la práctica recomendada de aplicar permisos de privilegios mínimos.

   Para conocer las instrucciones, consulte Create a permission set en la Guía del usuario de AWS IAM Identity Center.

2. Asigne usuarios a un grupo y, a continuación, asigne el acceso de inicio de sesión único al grupo.

Para conocer las instrucciones, consulte Add groups en la Guía del usuario de AWS IAM Identity Center.

Configuración de AWS Systems Manager

En los siguientes temas se describe cómo configurar la consola unificada de AWS Systems Manager para organizaciones de AWS Organizations y Cuentas de AWS individuales.

Temas

- Configuración del acceso a la consola de Systems Manager
- Configuración de la consola unificada de Systems Manager para una organización
- Configuración de la consola unificada de Systems Manager para una sola cuenta y región
- Deshabilitación de la consola unificada de Systems Manager

Configuración del acceso a la consola de Systems Manager

Para usar AWS Systems Manager en la AWS Management Console, debe tener configurados los permisos correctos.

Para obtener más información sobre cómo crear políticas de AWS Identity and Access Management y adjuntarlas a identidades de IAM, consulte Creación de políticas de IAM en la Guía del usuario de IAM.

Política de incorporación de Systems Manager

Puede crear una política de IAM como la que se muestra en el siguiente ejemplo y adjuntarla a sus identidades de IAM. Esta política concede acceso total para incorporarse a Systems Manager y configurarlo.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm-quicksetup`: permite a las entidades principales acceder a todas las acciones de Quick Setup de AWS Systems Manager.
- `ssm`: permite a las entidades principales acceder a Automatización de Systems Manager y Resource Explorer.
- `organizations`: permite a las entidades principales leer la estructura de una organización en AWS Organizations y coordinar a los administradores delegados cuando se incorporan a Systems Manager como una organización.

- `cloudformation`: permite a las entidades principales crear y administrar sus pilas de Quick Setup.
- `iam`: permite a las entidades principales administrar los roles y políticas de IAM necesarios para la incorporación de Systems Manager.

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
     {
       "Sid": "QuickSetupActions",
       "Effect": "Allow",
       "Action": [
         "ssm-quicksetup:*"
       ],
       "Resource": "*"
     },

     {
       "Sid": "SsmReadOnly",
       "Effect": "Allow",
       "Action": [
         "ssm:DescribeAutomationExecutions",
         "ssm:GetAutomationExecution",
         "ssm:ListAssociations",
         "ssm:DescribeAssociation",
         "ssm:ListDocuments",
         "ssm:ListResourceDataSync",
         "ssm:DescribePatchBaselines",
         "ssm:GetPatchBaseline",
         "ssm:DescribeMaintenanceWindows",
         "ssm:DescribeMaintenanceWindowTasks"
       ],
       "Resource": "*"
     },
     {
       "Sid": "SsmDocument",
       "Effect": "Allow",
       "Action": [
         "ssm:GetDocument",
```

```
        "ssm:DescribeDocument"
      ],
      "Resource": [
        "arn:aws:ssm:*:*:document/AWSQuickSetupType-*",
        "arn:aws:ssm:*:*:document/AWS-EnableExplorer"
      ]
    },
    {
      "Sid": "SsmEnableExplorer",
      "Effect": "Allow",
      "Action": "ssm:StartAutomationExecution",
      "Resource": "arn:aws:ssm:*:*:automation-definition/AWS-EnableExplorer:*"
    },
    {
      "Sid": "SsmExplorerRds",
      "Effect": "Allow",
      "Action": [
        "ssm:GetOpsSummary",
        "ssm:CreateResourceDataSync",
        "ssm:UpdateResourceDataSync"
      ],
      "Resource": "arn:aws:ssm:*:*:resource-data-sync/AWS-QuickSetup-*"
    },
    {
      "Sid": "OrgsReadOnly",
      "Effect": "Allow",
      "Action": [
        "organizations:DescribeAccount",
        "organizations:DescribeOrganization",
        "organizations:ListDelegatedAdministrators",
        "organizations:ListRoots",
        "organizations:ListParents",
        "organizations:ListOrganizationalUnitsForParent",
        "organizations:DescribeOrganizationalUnit",
        "organizations:ListAWSServiceAccessForOrganization"
      ],
      "Resource": "*"
    },
    {
      "Sid": "OrgsAdministration",
      "Effect": "Allow",
      "Action": [
        "organizations:EnableAWSServiceAccess",
        "organizations:RegisterDelegatedAdministrator",
```

```
        "organizations:DeregisterDelegatedAdministrator"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "organizations:ServicePrincipal": [
            "ssm.amazonaws.com",
            "ssm-quicksetup.amazonaws.com",
            "member.org.stacksets.cloudformation.amazonaws.com",
            "resource-explorer-2.amazonaws.com"
          ]
        }
      }
    },
    {
      "Sid": "CfnReadOnly",
      "Effect": "Allow",
      "Action": [
        "cloudformation:ListStacks",
        "cloudformation:DescribeStacks",
        "cloudformation:ListStackSets",
        "cloudformation:DescribeOrganizationsAccess"
      ],
      "Resource": "*"
    },
    {
      "Sid": "OrgCfnAccess",
      "Effect": "Allow",
      "Action": [
        "cloudformation:ActivateOrganizationsAccess"
      ],
      "Resource": "*"
    },
    {
      "Sid": "CfnStackActions",
      "Effect": "Allow",
      "Action": [
        "cloudformation:CreateStack",
        "cloudformation:DeleteStack",
        "cloudformation:DescribeStackResources",
        "cloudformation:DescribeStackEvents",
        "cloudformation:GetTemplate",
        "cloudformation:RollbackStack",
        "cloudformation:TagResource",
```

```
        "cloudformation:UntagResource",
        "cloudformation:UpdateStack"
      ],
      "Resource": [
        "arn:aws:cloudformation:*:*:stack/StackSet-AWS-QuickSetup-*",
        "arn:aws:cloudformation:*:*:stack/AWS-QuickSetup-*",
        "arn:aws:cloudformation:*:*:type/resource/*"
      ]
    },
    {
      "Sid": "CfnStackSetActions",
      "Effect": "Allow",
      "Action": [
        "cloudformation:CreateStackInstances",
        "cloudformation:CreateStackSet",
        "cloudformation:DeleteStackInstances",
        "cloudformation:DeleteStackSet",
        "cloudformation:DescribeStackInstance",
        "cloudformation:DetectStackSetDrift",
        "cloudformation:ListStackInstanceResourceDrifts",
        "cloudformation:DescribeStackSet",
        "cloudformation:DescribeStackSetOperation",
        "cloudformation:ListStackInstances",
        "cloudformation:ListStackSetOperations",
        "cloudformation:ListStackSetOperationResults",
        "cloudformation:TagResource",
        "cloudformation:UntagResource",
        "cloudformation:UpdateStackSet"
      ],
      "Resource": [
        "arn:aws:cloudformation:*:*:stackset/AWS-QuickSetup-*",
        "arn:aws:cloudformation:*:*:type/resource/*",
        "arn:aws:cloudformation:*:*:stackset-target/AWS-QuickSetup-*:*"
      ]
    },
    {
      "Sid": "ValidationReadonlyActions",
      "Effect": "Allow",
      "Action": [
        "iam:ListRoles",
        "iam:GetRole"
      ],
      "Resource": "*"
    },
```

```
    {
      "Sid": "IamRolesMgmt",
      "Effect": "Allow",
      "Action": [
        "iam:CreateRole",
        "iam:DeleteRole",
        "iam:GetRole",
        "iam:AttachRolePolicy",
        "iam:DetachRolePolicy",
        "iam:GetRolePolicy",
        "iam:ListRolePolicies"
      ],
      "Resource": [
        "arn:aws:iam::*:role/AWS-QuickSetup-*",
        "arn:aws:iam::*:role/service-role/AWS-QuickSetup-*"
      ]
    },
    {
      "Sid": "IamPassRole",
      "Effect": "Allow",
      "Action": [
        "iam:PassRole"
      ],
      "Resource": [
        "arn:aws:iam::*:role/AWS-QuickSetup-*",
        "arn:aws:iam::*:role/service-role/AWS-QuickSetup-*"
      ],
      "Condition": {
        "StringEquals": {
          "iam:PassedToService": [
            "ssm.amazonaws.com",
            "ssm-quicksetup.amazonaws.com",
            "cloudformation.amazonaws.com"
          ]
        }
      }
    },
    {
      "Sid": "IamRolesPoliciesMgmt",
      "Effect": "Allow",
      "Action": [
        "iam:AttachRolePolicy",
        "iam:DetachRolePolicy"
      ],
```

```
      "Resource": [
        "arn:aws:iam::*:role/AWS-QuickSetup-*",
        "arn:aws:iam::*:role/service-role/AWS-QuickSetup-*"
      ],
      "Condition": {
        "ArnEquals": {
          "iam:PolicyARN": [
            "arn:aws:iam::aws:policy/
AWSSystemsManagerEnableExplorerExecutionPolicy",
            "arn:aws:iam::aws:policy/AWSQuickSetupSSMDeploymentRolePolicy"
          ]
        }
      }
    },
    {
      "Sid": "CfnStackSetsSLR",
      "Effect": "Allow",
      "Action": [
        "iam:CreateServiceLinkedRole"
      ],
      "Resource": [
        "arn:aws:iam::*:role/aws-service-role/
stacksets.cloudformation.amazonaws.com/
AWSServiceRoleForCloudFormationStackSetsOrgAdmin",
        "arn:aws:iam::*:role/aws-service-role/ssm.amazonaws.com/
AWSServiceRoleForAmazonSSM",
        "arn:aws:iam::*:role/aws-service-role/
accountdiscovery.ssm.amazonaws.com/AWSServiceRoleForAmazonSSM_AccountDiscovery",
        "arn:aws:iam::*:role/aws-service-role/ssm-quicksetup.amazonaws.com/
AWSServiceRoleForSSMQuickSetup",
        "arn:aws:iam::*:role/aws-service-role/resource-explorer-2.amazonaws.com/
AWSServiceRoleForResourceExplorer"
      ]
    }
  ]
}
```

Política de operadores de la consola de AWS Systems Manager

Puede crear una política de IAM como la que se muestra en el siguiente ejemplo y adjuntarla a sus identidades de IAM. Esta política otorga acceso completo para operar Systems Manager y permite que Systems Manager ejecute documentos de Automatización para el diagnóstico y la corrección.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite a las entidades principales acceder a todas las API de Systems Manager.
- `ssm-quicksetup`: permite a las entidades principales administrar sus configuraciones de Quick Setup.
- `ec2`: permite a Systems Manager determinar sus Regiones de AWS habilitadas y el estado de la instancia de Amazon EC2.
- `cloudformation`: permite a las entidades principales leer sus pilas de Quick Setup.
- `organizations`: permite a las entidades principales leer la estructura de una organización en AWS Organizations y coordinar a los administradores delegados cuando se incorporan a Systems Manager como una organización.
- `s3`: permite a las entidades principales ennumerar y obtener objetos de un bucket de Amazon S3 para su diagnóstico, que se crea durante el proceso de incorporación de Systems Manager.
- `iam:PassRole`: permite a las entidades principales transferir los roles que van a asumir a Systems Manager cuando ejecutan automatizaciones de diagnóstico y reparación de los nodos no administrados.
- `iam:GetRole`: permite a las entidades principales obtener información específica sobre los roles de Quick Setup cuando trabajan en Systems Manager.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:*",
        "ssm-quicksetup:*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowEC2DescribeActions",
      "Effect": "Allow",
```

```
      "Action": [
        "ec2:DescribeInstanceStatus",
        "ec2:DescribeInstances",
        "ec2:DescribeRegions"
      ],
      "Resource": "*"
    },
    {
      "Sid": "CfnAccess",
      "Effect": "Allow",
      "Action": [
        "cloudformation:ListStacks",
        "cloudformation:ListStackSets",
        "cloudformation:ListStackInstances",
        "cloudformation:ListStackSetOperations",
        "cloudformation:ListStackSetOperationResults",
        "cloudformation:DescribeStacks",
        "cloudformation:DescribeStackSet",
        "cloudformation:DescribeStackSetOperation",
        "cloudformation:DescribeOrganizationsAccess",
        "cloudformation:DescribeStackInstance",
        "cloudformation:DetectStackSetDrift",
        "cloudformation:ListStackInstanceResourceDrifts"
      ],
      "Resource": "*"
    },
    {
      "Sid": "OrgsReadOnly",
      "Effect": "Allow",
      "Action": [
        "organizations:DescribeAccount",
        "organizations:DescribeOrganization",
        "organizations:ListDelegatedAdministrators",
        "organizations:ListRoots",
        "organizations:ListParents",
        "organizations:ListOrganizationalUnitsForParent",
        "organizations:DescribeOrganizationalUnit",
        "organizations:ListAWSServiceAccessForOrganization"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowKMSOperations",
      "Effect": "Allow",
```

```
      "Action": [
        "kms:Decrypt",
        "kms:GenerateDataKey"
      ],
      "Resource": "arn:aws:kms:*:*:key/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/SystemsManagerManaged": "true"
        },
        "ArnLike": {
          "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::do-not-delete-ssm-
diagnosis-*"
        },
        "StringLike": {
          "kms:ViaService": "s3.*.amazonaws.com"
        },
        "Bool": {
          "aws:ViaAWSService": "true"
        }
      }
    },
    {
      "Sid": "AllowReadS3BucketFromOrganization",
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Resource": "arn:aws:s3:::do-not-delete-ssm-diagnosis*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceOrgId": "${aws:PrincipalOrgId}"
        }
      }
    },
    {
      "Sid": "AllowReadS3BucketFromSingleAccount",
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Resource": "arn:aws:s3:::do-not-delete-ssm-diagnosis*",
      "Condition": {
```

```
        "StringEquals": {
          "aws:ResourceAccount": "${aws:PrincipalAccount}"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": [
        "arn:aws:iam::*:role/AWS-SSM-DiagnosisAdminRole*",
        "arn:aws:iam::*:role/AWS-SSM-DiagnosisExecutionRole*",
        "arn:aws:iam::*:role/AWS-SSM-RemediationAdminRole*",
        "arn:aws:iam::*:role/AWS-SSM-RemediationExecutionRole*"
      ],
      "Condition": {
        "StringEquals": {
          "iam:PassedToService": "ssm.amazonaws.com"
        }
      }
    },
    {
      "Sid": "IamReadOnly",
      "Effect": "Allow",
      "Action": "iam:GetRole",
      "Resource": [
        "arn:aws:iam::*:role/AWS-QuickSetup-*",
        "arn:aws:iam::*:role/service-role/AWS-QuickSetup-*"
      ]
    }
  ]
}
```

Política de solo lectura para operadores de la consola de AWS Systems Manager

Puede crear una política de IAM como la que se muestra en el siguiente ejemplo y adjuntarla a sus identidades de IAM. Esta política otorga acceso de solo lectura a Systems Manager.

- `ssm`: permite a las entidades principales acceder a las API de solo lectura de Systems Manager.
- `ssm-quicksetup`: permite a las entidades principales leer sus configuraciones de Quick Setup.
- `cloudformation`: permite a las entidades principales leer sus pilas de Quick Setup.

- `iam:GetRole`: permite a las entidades principales obtener información específica sobre los roles de Quick Setup cuando usan Systems Manager.
- `ec2:DescribeRegions`: permite que Systems Manager determine sus Regiones de AWS habilitadas.
- `organizations`: permite a las entidades principales leer la estructura de una organización en AWS Organizations cuando se incorporan a Systems Manager como organización.
- `s3`: permite a las entidades principales ennumerar y obtener objetos de un bucket de Amazon S3 que se crea durante el proceso de incorporación de Systems Manager.

Detalles de los permisos

Esta política incluye los siguientes permisos.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:Describe*",
        "ssm:Get*",
        "ssm:List*",
        "ssm-quicksetup:List*",
        "ssm-quicksetup:Get*",
        "cloudformation:Describe*",
        "cloudformation:Get*",
        "cloudformation:List*",
        "iam:GetRole",
        "ec2:DescribeRegions",
        "organizations:Describe*",
        "organizations:List*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowKMSOperations",
      "Effect": "Allow",
      "Action": [
```

```
        "kms:Decrypt"
      ],
      "Resource": "arn:aws:kms:*:*:key/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/SystemsManagerManaged": "true"
        },
        "ArnLike": {
          "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::do-not-delete-ssm-
diagnosis-*"
        },
        "StringLike": {
          "kms:ViaService": "s3.*.amazonaws.com"
        },
        "Bool": {
          "aws:ViaAWSService": "true"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Resource": "arn:aws:s3:::do-not-delete-ssm-diagnosis*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceOrgId": "${aws:PrincipalOrgId}"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Resource": "arn:aws:s3:::do-not-delete-ssm-diagnosis*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "${aws:PrincipalAccount}"
        }
      }
```

```
    }
  ]
}
```

Configuración de la consola unificada de Systems Manager para una organización

Use la AWS Management Console para configurar la experiencia de consola unificada de Systems Manager con solo unos pocos clics. Para configurar Systems Manager para una organización de AWS Organizations, debe tener acceso a la cuenta de administración de su organización y a otra cuenta de la organización para usarla como administrador delegado. El acceso a la cuenta de administración solo es necesario para activar o desactivar Systems Manager. Para administrar los nodos, utilizará la cuenta de administrador delegado.

Requisitos previos

Al administrar los nodos de una organización, Systems Manager utiliza varios servicios dependientes para configurar y mejorar la funcionalidad de la consola unificada. Como resultado, Systems Manager debe habilitar el acceso de confianza y registrar una cuenta de administrador delegado para los siguientes servicios:

- AWS CloudFormation: implementa los recursos necesarios de Systems Manager en sus cuentas.
- Explorador de recursos de AWS: búsqueda y filtrado de instancias de EC2 en sus cuentas.
- AWS Systems Manager Explorer: supervisa y soluciona problemas del estado de los recursos implementados de Systems Manager en sus cuentas.
- AWS Systems Manager Quick Setup: implementa las configuraciones de Quick Setup necesarias de Systems Manager en sus cuentas.

Antes de empezar, asegúrese de no haber superado ya la cuota de administradores delegados en ninguno de estos servicios dependientes. De lo contrario, no podrá registrar las cuentas de administrador delegado necesarias para habilitar Systems Manager. Al activar Systems Manager para una organización, se incluyen todas las cuentas de la organización. En este momento, no hay ninguna disposición que permita excluir cuentas del proceso de configuración. Al activar Systems Manager, puede elegir las Regiones de AWS que desee incluir. Solo se pueden seleccionar las regiones que actualmente admiten la consola unificada de Systems Manager. Para obtener más

información sobre las regiones en las que está disponible la experiencia de consola, consulte Regiones de AWS admitidas.

Recursos de la consola unificada

El proceso de configuración de la consola unificada de Systems Manager cumple muchos requisitos previos por usted. En función de las características que elija configurar, esto incluye habilitar la configuración de administración de hosts predeterminada para proporcionar los permisos de IAM necesarios a sus nodos y más. A continuación se muestra una lista detallada de los recursos creados por Systems Manager para la consola unificada. En función de las características que elija configurar, es posible que algunos recursos no se creen.

Vistas administradas por Explorador de recursos de AWS

- `AWSManagedViewForSSM`: permite a Systems Manager acceder a la información de recursos indexada por Resource Explorer para su organización. Solo Systems Manager puede actualizar o eliminar estas vistas administradas. Esto significa que si quiere eliminar las vistas administradas o desactivar el Resource Explorer, debe deshabilitar la consola unificada. Para obtener más información sobre cómo deshabilitar la consola unificada, consulte Deshabilitación de la consola unificada de Systems Manager. Para obtener más información sobre las vistas administradas, consulte AWS Managed Views en la Guía del usuario de Resource Explorer.

  > **Note**
  >
  > Si creó un índice agregador de Resource Explorer en una región distinta a la de su región local, Systems Manager degradará el índice actual. Posteriormente, Systems Manager promocionará el índice local de su región local como el nuevo índice agregador. Durante este periodo, solo se muestran los nodos de su región local. Este proceso puede tardar hasta 24 horas en completarse.

Roles de IAM

- `RoleForOnboardingAutomation`: permite a Systems Manager administrar los recursos durante el proceso de configuración. Para obtener más información sobre la política, consulte AWSQuickSetupSSMManageResourcesExecutionPolicy.

- `RoleForLifecycleManagement`: permite a Lambda administrar el ciclo de vida de los recursos creados por el proceso de configuración. Para obtener más información sobre la política, consulte AWSQuickSetupSSMLifecycleManagementExecutionPolicy.
- `RoleForAutomation`: un rol de servicio para Automatización de Systems Manager que se debe asumir para ejecutar los manuales de procedimientos. Para obtener más información, consulte Cree los roles de servicio para la automatización mediante la consola.
- `AWSSSMDiagnosisAdminRole`: un rol administrativo que se utiliza para iniciar automatizaciones que utilizan manuales de procedimientos de diagnóstico. Para obtener más información sobre las políticas, consulte AWS-SSM-DiagnosisAutomation-AdministrationRolePolicy, AWS-SSM-Automation-DiagnosisBucketPolicy y AWS-SSM-DiagnosisAutomation-OperationalAccountAdministrationRolePolicy.
- `AWSSSMDiagnosisExecutionRole`: un rol de ejecución de la automatización para el manual de procedimientos de diagnóstico. Para obtener más información sobre las políticas, consulte AWS-SSM-DiagnosisAutomation-ExecutionRolePolicy y AWS-SSM-Automation-DiagnosisBucketPolicy.
- `AWSSSMRemediationAdminRole`: un rol administrativo que se utiliza para iniciar automatizaciones que utilizan manuales de procedimientos de corrección. Para obtener más información sobre las políticas, consulte AWS-SSM-RemediationAutomation-AdministrationRolePolicy, AWS-SSM-Automation-DiagnosisBucketPolicy y AWS-SSM-RemediationAutomation-OperationalAccountAdministrationRolePolicy.
- `AWSSSMRemediationExecutionRole`: un rol de ejecución de la automatización para el manual de procedimientos de corrección. Para obtener más información acerca de las políticas, consulte AWS-SSM-RemediationAutomation-ExecutionRolePolicy y AWS-SSM-Automation-DiagnosisBucketPolicy.
- `ManagedInstanceCrossAccountManagementRole`: permite a Systems Manager recopilar información de nodos administrados entre cuentas.

Asociaciones de State Manager

- `EnableDHMCAssociation`: se ejecuta a diario y garantiza que se haya habilitado Configuración predeterminada de administración de hosts.
- `SystemAssociationForEnablingExplorer`: se ejecuta a diario y se asegura de que Explorer esté activado. Explorer se usa para sincronizar datos de los nodos administrados.
- `EnableAREXAssociation`: se ejecuta a diario y garantiza que Explorador de recursos de AWS esté habilitado. Resource Explorer se usa para determinar las instancias de Amazon EC2 de su organización que no administra Systems Manager.

- `SSMAgentUpdateAssociation`: se ejecuta cada 14 días y garantiza que la última versión disponible de SSM Agent esté instalada en los nodos administrados.
- `SystemAssociationForInventoryCollection`: se ejecuta cada 12 horas y recopila datos de inventario de los nodos administrados.

Buckets de S3

- `DiagnosisBucket`: almacena los datos recopilados durante la ejecución del manual de procedimientos.

Funciones de Lambda

- `SSMLifecycleOperatorLambda`: permite a las entidades principales acceder a todas las acciones de Quick Setup de AWS Systems Manager.
- `SSMLifecycleResource`: recurso personalizado para ayudar a administrar el ciclo de vida de los recursos creados por el proceso de configuración.

Además, una vez finalizado el proceso de configuración, puede seleccionar la tarea de nodos Diagnosticar y corregir para aplicar automáticamente las correcciones a los nodos que no se notifican como administrados por Systems Manager. Esto puede incluir la identificación de problemas, como, por ejemplo, problemas de conectividad de red con los puntos de conexión de Systems Manager, etc. Para obtener más información, consulte Diagnóstico y corrección.

Configurar la consola unificada

Configuración de la consola de Systems Manager para una organización

1. Inicie sesión en la cuenta de administración de su organización.
2. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
3. Ingrese el ID de la cuenta que quiera registrar como administrador delegado.
4. Una vez que la cuenta de administrador delegado se registre correctamente, inicie sesión en la cuenta de administrador delegado que acaba de registrar y vuelva a la consola de Systems Manager para terminar de configurar Systems Manager.
5. Seleccione Activar Systems Manager.

6. En la sección Región local, determine la región en la que desea que Systems Manager agregue los datos de los nodos. De forma predeterminada, Systems Manager selecciona la región que utiliza actualmente. Para elegir una región local diferente, cambie la consola a la región que desea usar antes de configurar Systems Manager. Los datos de los nodos se replican en todas las cuentas y regiones de su organización y se almacenan en la región local. La región que seleccione no se podrá cambiar después de que se configure Systems Manager. Para usar una región diferente como región local de su organización, debe deshabilitar la consola unificada y volver a completar el proceso de configuración. Si su organización usa IAM Identity Center, debe seleccionar la misma región en la que lo configuró como región local.
7. En la sección Regiones, seleccione la región en la que desea habilitar Systems Manager.
8. En la sección Configuraciones de características, elija las opciones que desee permitir para la configuración:

   Habilitar la configuración de administración de host predeterminada (DHMC)

   Permite que Systems Manager configure la DHMC. Esta característica permite que Systems Manager utilice un rol de IAM para garantizar que todas las instancias de Amazon EC2 de la cuenta y la región tengan los permisos necesarios para que Systems Manager las administre. También puede especificar la frecuencia de la corrección de desviaciones. La desviación de la configuración se produce cada vez que un usuario realiza algún cambio en un servicio o característica que entra en conflicto con las selecciones realizadas a través de la configuración. Systems Manager comprueba si hay desviaciones en la configuración e intenta corregirlas en función de la frecuencia especificada. Debe especificar un valor comprendido entre 1 y 31 días. Si ya configuró la DHMC en una región, Systems Manager no cambia el rol de IAM que había seleccionado. Para obtener más información acerca de la DHMC, consulte Administración automática de instancias EC2 con la configuración de administración de hosts predeterminada.

   La DHMC permite administrar las instancias de Amazon EC2 sin tener que crear manualmente un perfil de instancia de AWS Identity and Access Management (IAM). Le recomendamos que elija esta opción para asegurarse de que las instancias de EC2 tengan los permisos necesarios para que Systems Manager las administre.

   Habilitar la recopilación de metadatos de inventario

   Puede configurar Systems Manager para que recopile los siguientes tipos de metadatos de los nodos:

   - Componentes de AWS: controlador de EC2, agentes, versiones y más.

- Aplicaciones: nombres de aplicaciones, editores, versiones y más.
- Detalles del nodo: nombre del sistema, nombre del sistema operativo (OS), versión del OS, último inicio, DNS, dominio, grupo de trabajo, arquitectura del OS y más.
- Configuración de red: dirección IP, dirección MAC, DNS, gateway, máscara de subred y más.
- Servicios: nombre, nombre de visualización, estado, servicios dependientes, tipo de servicio, tipo de inicio y más (solo en nodos de Windows Server).
- Roles de Windows: nombre, nombre de visualización, ruta, tipo de característica, estado instalado y más (solo en nodos de Windows Server).
- Actualizaciones de Windows: ID de hotfix, instalado por, fecha de instalación y más (solo en nodos de Windows Server).

Especifique la frecuencia con la que se recopila el inventario. Debe especificar un valor comprendido entre 1 y 744 horas. Para obtener más información sobre Inventario, una herramienta de AWS Systems Manager, consulte Inventario de AWS Systems Manager.

Habilitar las actualizaciones automáticas del agente de Systems Manager (SSM)

Se permite en Systems Manager la comprobación de una nueva versión del agente con la frecuencia especificada. El valor de la frecuencia debe estar comprendido entre 1 y 31 días. Si hay una nueva versión, Systems Manager actualiza automáticamente el agente en el nodo administrado a la versión más reciente publicada. Systems Manager no instala el agente en instancias en las que aún no esté presente. Para obtener información sobre las AMIs que SSM Agent tiene preinstaladas, consulte Búsqueda de AMIs con SSM Agent preinstalado.

Le recomendamos que elija esta opción para asegurarse de que los nodos siempre ejecuten la versión más actualizada de SSM Agent. Para obtener más información acerca de SSM Agent, incluida información sobre cómo instalar manualmente el agente, consulte Uso de SSM Agent.

9. Elija Enviar.

Según el tamaño de la organización, puede llevar mucho tiempo configurar la experiencia de consola unificada de Systems Manager.

Configuración de la consola unificada de Systems Manager para una sola cuenta y región

Para configurar la experiencia de consola unificada de Systems Manager para una sola Cuenta de AWS y Región de AWS no necesita usar Organizations ni registrar una cuenta de administrador delegado. El proceso de configuración de la experiencia de la consola de Systems Manager cumple muchos requisitos previos por usted. En función de las características que elija configurar, esto incluye habilitar la configuración de administración de hosts predeterminada para proporcionar los permisos de IAM necesarios a sus nodos y más. A continuación se muestra una lista detallada de los recursos creados por Systems Manager para la consola unificada.

Recursos de la consola unificada

En función de las características que elija configurar, es posible que algunos recursos no se creen.

Roles de IAM

- `RoleForOnboardingAutomation`: permite a Systems Manager administrar los recursos durante el proceso de configuración. Para obtener más información sobre la política, consulte AWSQuickSetupSSMManageResourcesExecutionPolicy.
- `RoleForLifecycleManagement`: permite a Lambda administrar el ciclo de vida de los recursos creados por el proceso de configuración. Para obtener más información sobre la política, consulte AWSQuickSetupSSMLifecycleManagementExecutionPolicy.
- `RoleForAutomation`: un rol de servicio para Automatización de Systems Manager que se debe asumir para ejecutar los manuales de procedimientos. Para obtener más información, consulte Cree los roles de servicio para la automatización mediante la consola.
- `AWSSSMDiagnosisAdminRole`: un rol de ejecución de la automatización para el manual de procedimientos de diagnóstico. Para obtener más información sobre las políticas, consulte AWS-SSM-DiagnosisAutomation-AdministrationRolePolicy, AWS-SSM-Automation-DiagnosisBucketPolicy y AWS-SSM-DiagnosisAutomation-OperationalAccountAdministrationRolePolicy.
- `AWSSSMRemediationAdminRole`: un rol de ejecución de la automatización para el manual de procedimientos de corrección. Para obtener más información sobre las políticas, consulte AWS-SSM-RemediationAutomation-AdministrationRolePolicy, AWS-SSM-Automation-DiagnosisBucketPolicy y AWS-SSM-RemediationAutomation-OperationalAccountAdministrationRolePolicy.

- `ManagedInstanceCrossAccountManagementRole`: permite a Systems Manager recopilar información de nodos administrados entre cuentas.

Asociaciones de State Manager

- `EnableDHMCAssociation`: se ejecuta a diario y garantiza que se haya habilitado Configuración predeterminada de administración de hosts.
- `SystemAssociationForEnablingExplorer`: se ejecuta a diario y se asegura de que Explorer esté activado. Explorer se usa para sincronizar datos de los nodos administrados.
- `EnableAREXAssociation`: se ejecuta a diario y garantiza que Explorador de recursos de AWS esté habilitado. Resource Explorer se usa para determinar las instancias de Amazon EC2 de su organización que no administra Systems Manager.
- `SSMAgentUpdateAssociation`: se ejecuta cada 14 días y garantiza que la última versión disponible de SSM Agent esté instalada en los nodos administrados.
- `SystemAssociationForInventoryCollection`: se ejecuta cada 12 horas y recopila datos de inventario de los nodos administrados.

Buckets de S3

- `DiagnosisBucket`: almacena los datos recopilados durante la ejecución del manual de procedimientos.

Funciones de Lambda

- `SSMLifecycleOperatorLambda`: permite a las entidades principales acceder a todas las acciones de Quick Setup de AWS Systems Manager.
- `SSMLifecycleResource`: recurso personalizado para ayudar a administrar el ciclo de vida de los recursos creados por el proceso de configuración.

Además, una vez finalizado el proceso de configuración, puede seleccionar la tarea de nodos Diagnosticar y corregir para aplicar automáticamente las correcciones a los nodos que no se notifican como administrados por Systems Manager. Esto puede incluir la identificación de problemas, como, por ejemplo, problemas de conectividad de red con los puntos de conexión de Systems Manager, etc.

Configurar la consola unificada

Configuración de Systems Manager para una sola cuenta y región

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. Seleccione Activar Systems Manager.
3. En la sección Configuraciones de características, elija las opciones que desee permitir para la configuración:

   Habilitar la configuración de administración de host predeterminada (DHMC)

   Permite que Systems Manager configure la DHMC. Esta característica permite que Systems Manager utilice un rol de IAM para garantizar que todas las instancias de Amazon EC2 de la cuenta y la región tengan los permisos necesarios para que Systems Manager las administre. También puede especificar la frecuencia de la corrección de desviaciones. La desviación de la configuración se produce cada vez que un usuario realiza algún cambio en un servicio o característica que entra en conflicto con las selecciones realizadas a través de la configuración. Systems Manager comprueba si hay desviaciones en la configuración e intenta corregirlas en función de la frecuencia especificada. Debe especificar un valor comprendido entre 1 y 31 días. Si ya configuró la DHMC en una región, Systems Manager no cambia el rol de IAM que había seleccionado. Para obtener más información acerca de la DHMC, consulte Administración automática de instancias EC2 con la configuración de administración de hosts predeterminada.

   La DHMC permite administrar las instancias de Amazon EC2 sin tener que crear manualmente un perfil de instancia de AWS Identity and Access Management (IAM). Le recomendamos que elija esta opción para asegurarse de que las instancias de EC2 tengan los permisos necesarios para que Systems Manager las administre.

   Habilitar la recopilación de metadatos de inventario

   Puede configurar Systems Manager para que recopile los siguientes tipos de metadatos de los nodos:

   - Componentes de AWS: controlador de EC2, agentes, versiones y más.
   - Aplicaciones: nombres de aplicaciones, editores, versiones y más.

- Detalles del nodo: nombre del sistema, nombre del sistema operativo (OS), versión del OS, último inicio, DNS, dominio, grupo de trabajo, arquitectura del OS y más.
- Configuración de red: dirección IP, dirección MAC, DNS, gateway, máscara de subred y más.
- Servicios: nombre, nombre de visualización, estado, servicios dependientes, tipo de servicio, tipo de inicio y más (solo en nodos de Windows Server).
- Roles de Windows: nombre, nombre de visualización, ruta, tipo de característica, estado instalado y más (solo en nodos de Windows Server).
- Actualizaciones de Windows: ID de hotfix, instalado por, fecha de instalación y más (solo en nodos de Windows Server).

Especifique la frecuencia con la que se recopila el inventario. Debe especificar un valor comprendido entre 1 y 744 horas. Para obtener más información sobre Inventario, una herramienta de AWS Systems Manager, consulte Inventario de AWS Systems Manager.

Habilitar las actualizaciones automáticas del agente de Systems Manager (SSM)

Se permite en Systems Manager la comprobación de una nueva versión del agente con la frecuencia especificada. El valor de la frecuencia debe estar comprendido entre 1 y 31 días. Si hay una nueva versión, Systems Manager actualiza automáticamente el agente en el nodo administrado a la versión más reciente publicada. Systems Manager no instala el agente en instancias en las que aún no esté presente. Para obtener información sobre las AMIs que SSM Agent tiene preinstaladas, consulte Búsqueda de AMIs con SSM Agent preinstalado.

Le recomendamos que elija esta opción para asegurarse de que los nodos siempre ejecuten la versión más actualizada de SSM Agent. Para obtener más información acerca de SSM Agent, incluida información sobre cómo instalar manualmente el agente, consulte Uso de SSM Agent.

4. Elija Enviar.

Deshabilitación de la consola unificada de Systems Manager

Para deshabilitar la consola unificada de Systems Manager en una organización, necesitará acceder a la cuenta que registró como administrador delegado de Systems Manager. Tras iniciar sesión en la cuenta de administrador delegado de su organización, puede deshabilitar la configuración en la sección Configuración de la consola unificada. Cuando deshabilita la configuración de la consola unificada de su organización, Systems Manager elimina los recursos creados, incluidas las

vistas administradas de Resource Explorer, durante el proceso de configuración. Al deshabilitar la configuración de su organización, no se revoca el acceso de confianza ni se anula el registro de las cuentas de administrador delegado de los servicios dependientes. En el siguiente procedimiento se describe cómo se deshabilita la configuración de la consola unificada.

Desactivación de la configuración de la consola unificada de Systems Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. Seleccione Configuración en el panel de navegación.
3. Seleccione Deshabilitar. Debe confirmar que desea deshabilitar la configuración de su organización. Esta acción elimina los recursos creados para la consola unificada y no se puede deshacer.

Si no puede acceder a la cuenta de administrador delegado de Systems Manager y desea deshabilitar la configuración de la consola unificada, también puede hacerlo desde la cuenta de administración de la organización. Con la AWS CLI o el SDK, llame a la operación de la API `DeleteConfigurationManager` y transfiera el valor `ManagerArn` de la configuración de la organización a su cuenta. El formato del ARN del administrador que se usa para configurar la consola unificada es el siguiente:

`arn:aws:ssm-quicksetup:`*`account-id`*`:configuration-manager/`*`configuration-manager-id`*.

> **Note**
>
> Si no conoce el valor de *`configuration-manager-id`*, llame a la acción de la API `ListConfigurationManagers` y filtre los resultados con el tipo `AWSQuickSetupType-SSM`.

Ejecución de tareas de administración de nodos con AWS Systems Manager

En los siguientes temas se describe cómo hacer las tareas de nodos comunes mediante la consola de AWS Systems Manager unificada para una organización de AWS Organizations y Cuentas de AWS individuales.

Temas

- Revisión de la información sobre los nodos
- Exploración de nodos
- Acceso a los nodos justo a tiempo mediante Systems Manager
- Diagnóstico y corrección
- Ajuste de la configuración de Systems Manager

Revisión de la información sobre los nodos

Puede obtener información sobre el estado general de los nodos administrados y las instancias de EC2 no administradas de su organización o cuenta mediante la consola unificada de Systems Manager.

Systems Manager proporciona una información general visual de los nodos administrados y las instancias de EC2 que aún no administra Systems Manager. (Un nodo administrado es cualquier equipo configurado para su uso con Systems Manager en entornos híbridos y multinube. Para obtener información acerca de los tipos de equipos compatibles, consulte Tipos de equipos compatibles con entornos híbridos y multinube.)

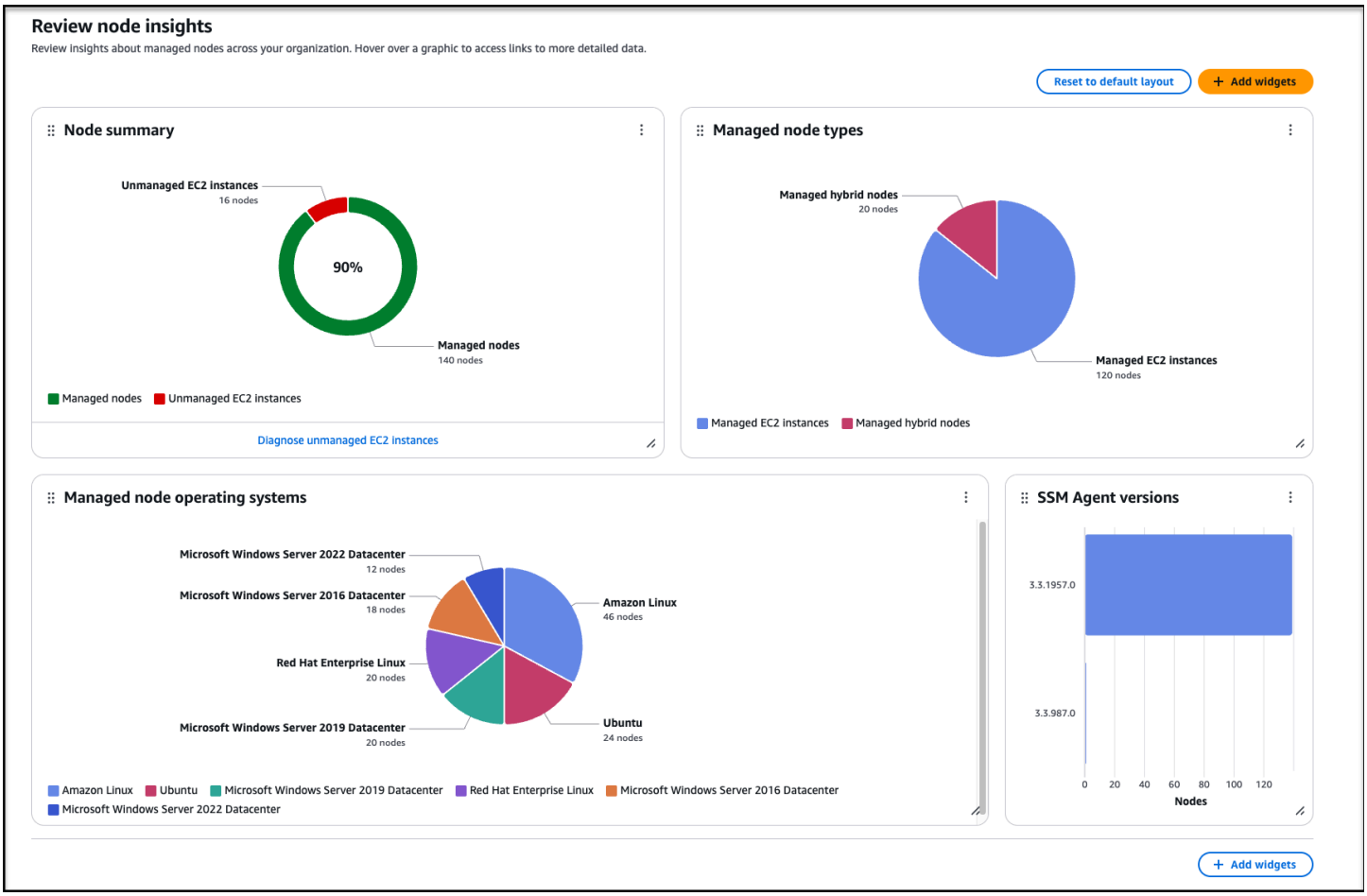
Esta información general se proporciona a través de cuadros de informes individuales, denominados widgets, que incluyen gráficos circulares interactivos y otros gráficos.

Antes de empezar

Para revisar la información sobre los nodos, debe incorporar antes su organización o cuenta a la consola unificada de Systems Manager. Para obtener más información, consulte Configuración de AWS Systems Manager.

Tras la incorporación, abra la consola de Systems Manager y elija Revisión de información de nodos.

En la siguiente imagen se muestran algunos de los cuadros de informe individuales, llamados widgets, que están disponibles en la página Revisión de información de nodos.



En la pantalla se admiten widgets que le proporcionan la siguiente información.

Resumen de nodos

Indica cuántas instancias de EC2 de su organización o cuenta no son nodos administrados actualmente y cuántos nodos administrados hay en la flota de su organización o cuenta.

¿Qué es una instancia no administrada?

Cuando detiene una instancia EC2 administrada, aparece como "No administrada" en la consola de Systems Manager. Este es el comportamiento esperado porque SSM Agent no tiene una conexión activa en el servicio.

> **Note**
>
> Esto es diferente de la forma en que AWS Config define una instancia como no administrada. Si una instancia se detiene en este momento, AWS Config informa cuál era el estado de la instancia la última vez que se estableció una conexión de "latido" entre SSM Agent en la instancia y el servicio Systems Manager.

Cuando la instancia se reinicia, se vuelve a conectar automáticamente al servicio Systems Manager y su estado en la consola unificada se restablece a "Administrada" en cinco minutos. No se requiere ninguna intervención manual y todas la configuración de Systems Manager de la instancia se conservan durante el ciclo de Detención/Inicio.

Sin embargo, si la instancia sigue sin aparecer como "Administrada" varios minutos después de iniciarse, es probable que no esté configurada correctamente para la administración de Systems Manager. En este caso, recomendamos ejecutar un diagnóstico para identificar la razón por la que la instancia permanece en un estado no administrado. Para obtener más información, consulte Diagnóstico y corrección de instancias de Amazon EC2 no administradas en Systems Manager.

Si el análisis de diagnóstico no puede determinar el problema, consulte los siguientes temas para comprobar que se cumplen todos los requisitos para SSM Agent, los roles AWS Identity and Access Management (IAM) y los requisitos previos de Systems Manager:

- Solución de problemas de SSM Agent
- Configuración de permisos de instancia requeridos para Systems Manager
- Solución de problemas de disponibilidad de nodos administrados

Tipos de nodos administrados

Indica cuántos nodos administrados de su flota son instancias de EC2 y cuántos son de otros tipos de servidores, lo que incluye servidores en sus propias instalaciones (servidores en las instalaciones), dispositivos principales de AWS IoT Greengrass y dispositivos de periferia de AWS IoT y que no son de AWS, además de máquinas virtuales (VM), lo que incluye VM en otros entornos en la nube. Puede pasar el ratón sobre el gráfico Tipos de nodos para acceder a los enlaces con más información en la página Exploración de nodos.

Para obtener más información acerca de la compatibilidad de AWS con entornos híbridos y multinube, consulte Soluciones de AWS para entornos híbridos y multinube.

Versiones de SSM Agent

Proporciona información acerca de las instalaciones de AWS Systems Manager Agent (SSM Agent) en su flota. SSM Agent es un software de Amazon que se ejecuta en los nodos administrados. SSM Agent permite que Systems Manager actualice, administre y configure estos recursos. El agente procesa las solicitudes del servicio de Systems Manager en la Nube de AWS y, a continuación, las ejecuta como se especifica en la solicitud.

En el caso de los nodos administrados de su flota, este widget informa sobre las versiones de SSM Agent de su flota, desde la más reciente hasta la más antigua. Puede pasar el ratón sobre el gráfico Versiones deSSM Agent para acceder a los enlaces con más información en la página Exploración de nodos.

Para obtener más información acerca de SSM Agent, consulte Uso de SSM Agent.

Sistemas operativos de nodos administrados

Proporciona un desglose del porcentaje de cada sistema operativo en los nodos administrados de su flota. Puede pasar el ratón sobre el gráfico Nodos administrados por sistemas operativos para acceder a los enlaces con más información en la página Exploración de nodos.

Puede personalizar el diseño de los widgets en la página Revisión de información de nodos mediante una funcionalidad de arrastrar y soltar widgets, además de eliminar widgets y agregarlos a la pantalla.

Use la información de los siguientes temas para trabajar con los widgets de información de nodos de Systems Manager.

Temas

- Adición o eliminación de widgets de la página Revisión de información de nodos
- Reorganización de los widgets en la página Revisión de información de nodos

Adición o eliminación de widgets de la página Revisión de información de nodos

Para personalizar el diseño de la página Revisión de información de nodos de Systems Manager, agregue y elimine widgets.

> **Note**
>
> De manera predeterminada, la página muestra todos los widgets disponibles.

Adición o eliminación de widgets de la página Revisión de información de nodos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación izquierdo, elija Revisión de información de nodos.
3. Para eliminar un widget de la pantalla, haga lo siguiente:
   a. Elija el menú Más opciones () para el widget.
   b. Elija Eliminar widget.
4. Para agregar un widget a la pantalla, haga lo siguiente:
   a. Elija Añadir widget.
   b. En el panel Agregar widgets, haga clic y mantenga pulsado el controlador de arrastre () del widget para agregarlo a la pantalla.
   c. Arrastre el widget y suéltelo en el panel principal.

Reorganización de los widgets en la página Revisión de información de nodos

Puede personalizar el diseño de la página Revisión de información de nodos al reorganizar los widgets.

Reorganización de los widgets de la página Revisión de información de nodos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Revisión de información de nodos.

3. Para personalizar el diseño del widget, elija el widget que quiere mover. Haga clic y mantenga presionado el controlador de arrastre ( ) y, a continuación, arrástrelo a su nueva ubicación.
4. Repita este proceso para cada widget que desee cambiar de posición.

Si decide que no le gusta el nuevo diseño, elija Restablecer el diseño predeterminado para mover todos los widgets a su ubicación original.

Exploración de nodos

Puede utilizar la página Explorar nodos de Systems Manager para revisar los detalles de los nodos administrados de su organización o cuenta de acuerdo con los criterios que especifique en los filtros. También puede utilizar la integración de Systems Manager con Amazon Q Developer (Amazon Q), una solución de IA generativa de AWS, para hacer búsquedas mediante petiociones de texto.

Antes de empezar

Para utilizar la característica Explorar nodos, debe incorporar antes su organización o cuenta a la consola unificada de Systems Manager. Para obtener más información, consulte Configuración de la consola unificada de Systems Manager para una organización.

Tras la incorporación, abra la consola de Systems Manager y seleccione Explorar nodos.

> **Note**
>
> Si creó un índice agregador de Resource Explorer en una región distinta a la de su región local, Systems Manager degradará el índice actual. Posteriormente, Systems Manager promocionará el índice local de su región local como el nuevo índice agregador. Durante este periodo, solo se muestran los nodos de su región local. Este proceso puede tardar hasta 24 horas en completarse.

Temas

- Exploración de nodos mediante filtros de consola
- Exploración de nodos mediante peticiones de texto en Amazon Q

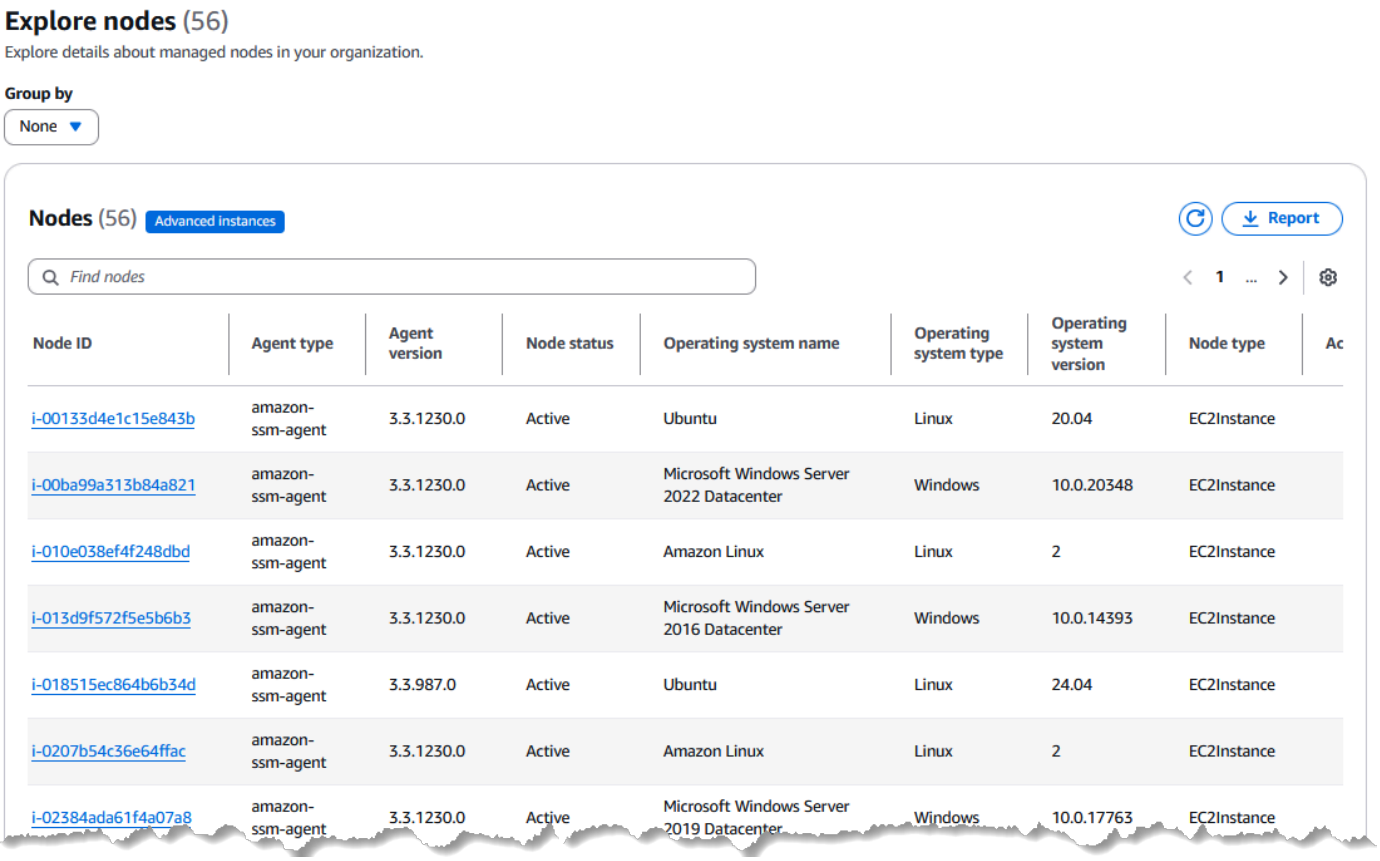
- Visualización de los detalles de los nodos individuales y realización de acciones en un nodo
- Descarga o exportación de un informe de nodos administrados
- Administración del contenido y de la apariencia de los informes de nodos

Exploración de nodos mediante filtros de consola

En la consola de Systems Manager, puede agrupar los nodos administrados de la siguiente manera:

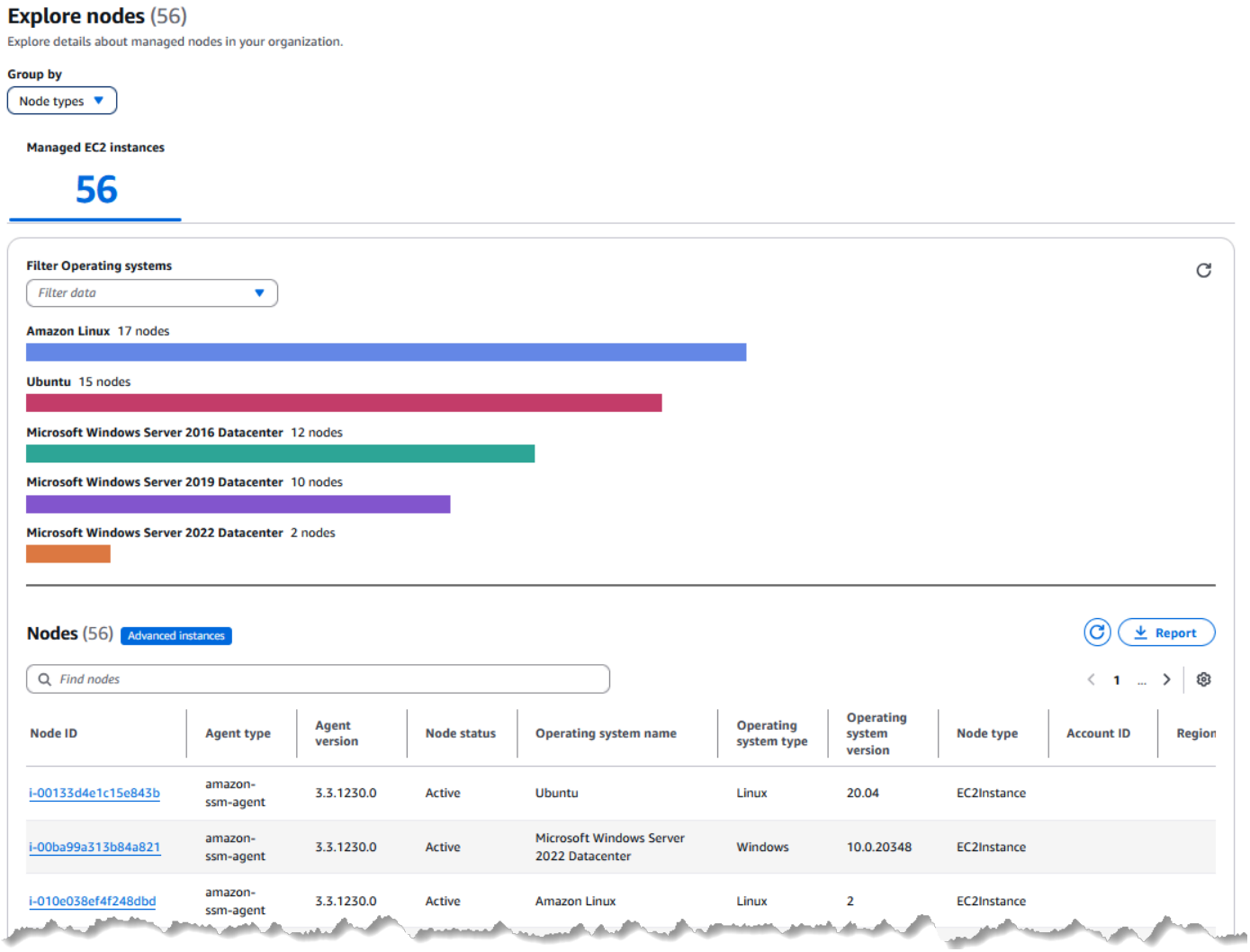
All nodes (No filter)

Lista de todos los nodos administrados de la organización o de la cuenta.



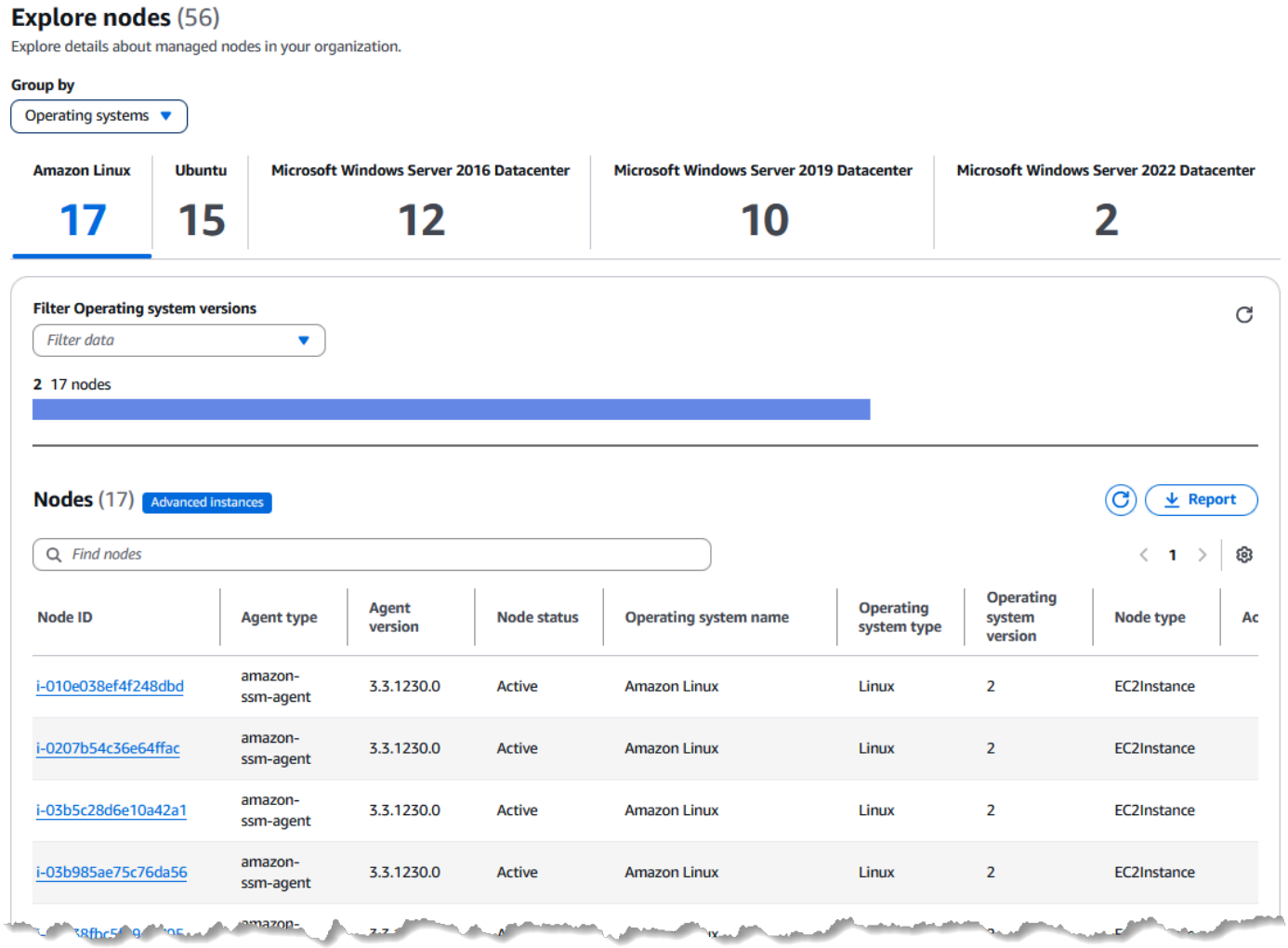
Node types

Proporciona pestañas para ver los datos por separado de las instancias de Amazon Elastic Compute Cloud (Amazon EC2) y de otros tipos de máquinas, además de servidores en sus propias instalaciones (servidores en las instalaciones), dispositivos principales de AWS IoT Greengrass y dispositivos de periferia de AWS IoT y que no son de AWS, así como máquinas virtuales (VM), incluidas las que están en otros entornos en la nube.



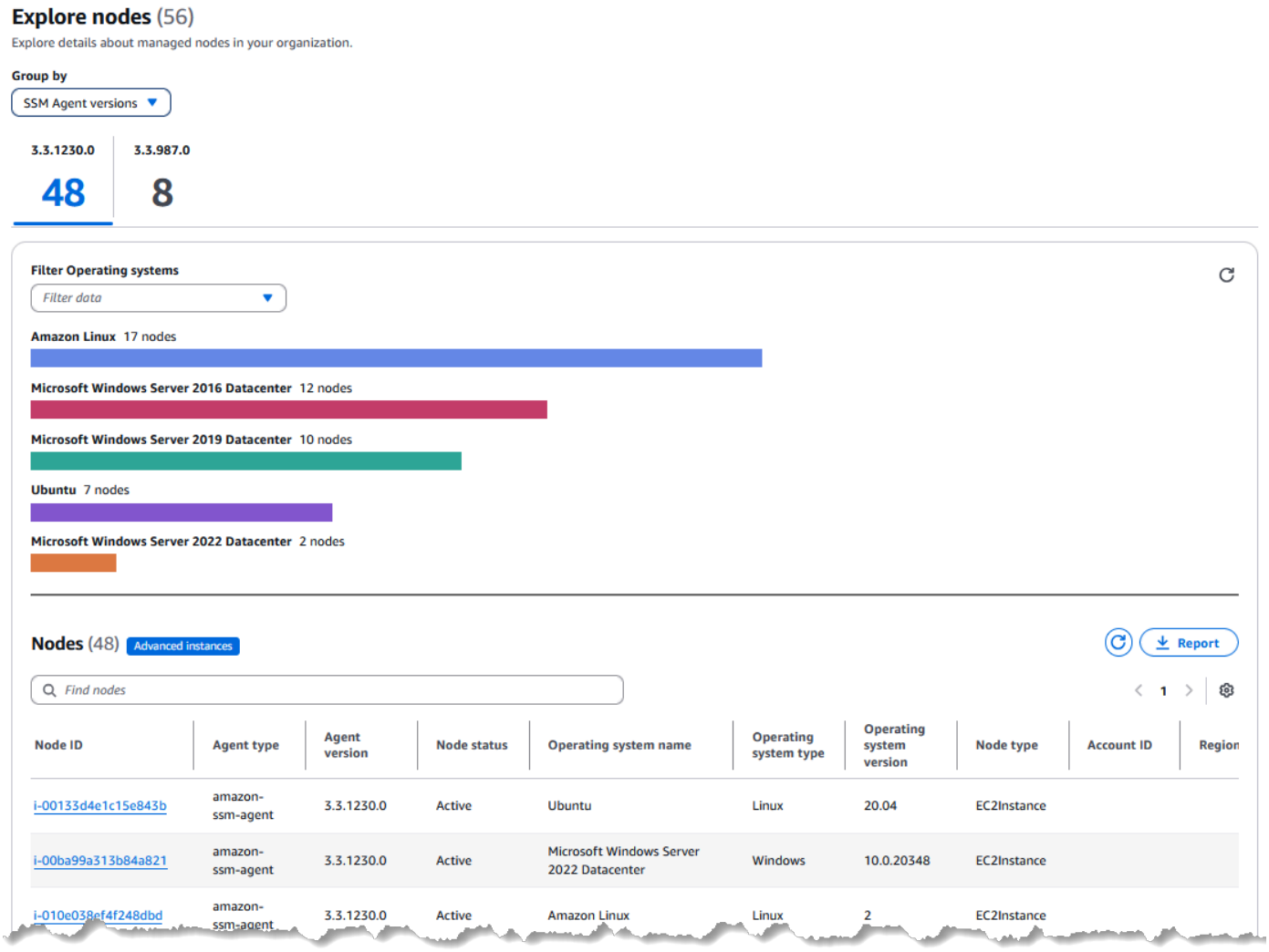
Operating systems

Proporciona una pestaña para cada tipo de sistema operativo de la organización o cuenta, como Amazon Linux y Microsoft Windows Server 2022 Datacenter. En cada pestaña, puede filtrar aún más la lista si selecciona solo versiones específicas de los sistemas operativos, como Amazon Linux 2 y Amazon Linux 2023.



SSM Agent versions

Proporciona una pestaña para cada versión de SSM Agent instalada en los nodos administrados de su flota. En cada pestaña, puede filtrar aún más la lista si selecciona solo versiones específicas de los sistemas operativos, como Amazon Linux y Microsoft Windows Server 2022 Datacenter.



Además, en cada una de estas vistas, puedes refinar aún más la lista de nodos notificados si elige ver solo los nodos de una propiedad determinada, como el estado del nodo, el ID de la Cuenta de AWS, el ID de la unidad organizativa, etc.

Puede personalizar la visualización del informe al elegir qué columnas de datos disponibles se mostrarán en la página Explorar nodos. También puede descargar informes en formatos `CSV` y `JSON` o exportarlos a Amazon S3 en formato `CSV`.

Temas

- Elección de una vista de filtros para los resúmenes de nodos administrados

Elección de una vista de filtros para los resúmenes de nodos administrados

La página Explorar nodos de Systems Manager le permite ver datos agregados sobre su flota de acuerdo con una serie de vistas de filtros disponibles.

Elección de una vista de filtros para los resúmenes de nodos administrados

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Explorar nodos.
3. En Vista de filtros, seleccione una de las opciones de filtro y, si lo desea, restrinja aún más el informe:

   - Nodos administrados: en el cuadro de búsqueda (), puede seleccionar una propiedad y un delimitador, como `Node type = Managed EC2 instances`.
   - Sistemas operativos: en la lista Filtrado de versiones de sistemas operativos, puede seleccionar un número de versión del sistema operativo. En el cuadro de búsqueda (), puede seleccionar una propiedad y un delimitador, como `Node type = Managed EC2 instances`.
   - Versiones de SSM Agent: en la lista Filtrado de sistemas operativos, puede seleccionar un número de versión del sistema operativo. En el cuadro de búsqueda (), puede seleccionar una propiedad y un delimitador, como `Node type = Managed EC2 instances`.
   - Tipos de nodos: en la lista Filtrado de sistemas operativos, puede seleccionar un nombre del sistema operativo. En el cuadro de búsqueda (), puede seleccionar una propiedad y un delimitador, como `Node type = Managed EC2 instances`.

Si lo desea, puede filtrar la lista. A continuación, para ver los detalles de un nodo administrado específico, elija el ID en la columna ID de nodo. Desde esa vista detallada, puede realizar una serie de acciones en el nodo.

Exploración de nodos mediante peticiones de texto en Amazon Q

Gracias a la integración de Systems Manager con Amazon Q Developer, puede usar peticiones de texto para ver la información creada por la IA generativa sobre los nodos administrados.

Amazon Q Developer es un asistente conversacional con tecnología de IA generativa que resulta útil a la hora de comprender, crear, ampliar y utilizar aplicaciones de AWS. A fin de acelerar la creación en AWS, el modelo que hay tras Amazon Q se ha optimizado con contenidos de AWS de alta calidad para generar respuestas más completas, con un mayor volumen de referencias y con las que sea posible tomar más medidas. Para obtener más información, consulte What is Amazon Q Developer? en la Guía del usuario de Amazon Q Developer.

La integración entre Systems Manager y Amazon Q le permite obtener rápidamente visibilidad y control sobre entornos grandes y distribuidos en varias regiones y Cuentas de AWS. Puede utilizar las consultas en lenguaje natural para buscar rápidamente los datos de nodos y, a continuación, identificar los problemas y tomar medidas con mayor rapidez.

Cuando formula una pregunta en lenguaje natural sobre nodos o instancias administrados, Amazon Q utiliza la acción `ListNodes` de Systems Manager y crea filtros basados en la entrada de texto para recuperar los resultados.

Por ejemplo, supongamos que proporciona a Amazon Q la siguiente petición:

`List my managed nodes running Red Hat Enterprise Linux 9.2`

Amazon Q determina qué filtros incluir en una solicitud y, a continuación, ejecuta una consulta similar a la siguiente:

```
aws ssm list-nodes \
    --filters Key=PlatformName,Values='Red Hat Enterprise Linux',Type=Equal
 Key=PlatformVersion,Values=9.2,Type=Equal
```

Posteriormente, Amazon Q genera un informe sobre las instancias de Red Hat Enterprise Linux de su cuenta, en el que se incluye información como el número de instancias, sus ID y sus regiones.

También puede ver un resumen en JSON de los detalles de cada instancia, así como abrir un enlace para ver las listas completas de nodos administrados o instancias de EC2 en la página Explorar nodos de Systems Manager. En Explorar de nodos se muestran los resultados que coinciden con los criterios de filtro que incluyó en la petición. Ahí puede cambiar o refinar los filtros de su solicitud, tal y como se describe en Exploración de nodos.

Temas

- Información sobre cómo crear peticiones efectivas para preguntar a Amazon Q sobre su flota
- Exploración de los nodos administrados con Amazon Q

Información sobre cómo crear peticiones efectivas para preguntar a Amazon Q sobre su flota

Cuanto mejor sea la calidad de la pregunta o petición que le dé a Amazon Q, mejor será el resultado que le proporcionará.

Consejos para las peticiones de las consultas

Tenga en cuenta los siguientes consejos cuando haga consultas a Amazon Q sobre su flota:

1. Para mejorar la precisión de los resultados, use los términos “nodos administrados” e “instancias administradas” en sus solicitudes en lugar de simplemente “nodos” e “instancias”.
2. Para consultar los resultados en varias cuentas que forman parte de una organización, tal como están configuradas en AWS Organizations, debe iniciar sesión en la cuenta de administrador delegado de la región local designada.
3. En la cuenta de administrador delegado, utilice términos para que Amazon Q comprenda que hace preguntas sobre nodos e instancias en toda la organización. Para ello, utilice específicamente términos como “en mi organización” o “en mi cuenta 123456789012”.

Temas

- Ejemplos de preguntas para Amazon Q
- Nombres y versiones de sistemas operativos compatibles con las peticiones

Ejemplos de preguntas para Amazon Q

En la siguiente tabla, incluimos ejemplos de preguntas que muestran algunas formas de crear consultas de Amazon Q que generen mejores resultados.

También ofrecemos ejemplos de los filtros que Amazon Q aplicará al ejecutar el comando ListNodes, que se generan a partir del contenido de su petición.

Ejemplo de pregunta en lenguaje natural	Filtros aplicados de Amazon Q
Show me my Windows managed nodes.	`PlatformType = Windows`
List my managed instances in account 123456789012.	`AccountId = 123456789012`
Show me all managed nodes running Amazon Linux 1 across my organization.	`PlatformName = Amazon Linux` `PlatformVersion = 1`
Show me all managed instances running Microsoft Windows Server 2019 Datacenter in my organization.	`PlatformName = Microsoft Windows Server 2019 Datacenter`
Can you show me all managed nodes with SSM Agent version 3.3.1142.0?	`AgentType = amazon-ssm-agent` `AgentVersion = 3.3.1142.0`

List all Amazon Linux 2 managed instances in account 123456789012 that have SSM Agent version 3.3.1230.0.	`PlatformName = Amazon Linux`
	`PlatformVersion = 2`
	`AccountId = 123456789012`
	`AgentType = amazon-ssm-agent`
	`AgentVersion = 3.3.1230.0`

What Microsoft Windows Server 2008 R2 Enterprise managed nodes are running in the eu-central-1 region across my entire organization?	`PlatformName = Microsoft Windows Server 2008 R2 Enterprise` `Region = eu-central-1`

Show me all managed instances running CentOS Linux 7 in ou-d6ty-gxdma6vm.	`PlatformName = CentOS Linux`
	`PlatformVersion = 7`
	`OrganizationalUnitId = ou-d6ty-gxdma6vm`

Ejemplo de pregunta en lenguaje natural	Filtros aplicados de Amazon Q
What Ubuntu managed instances are in account 123456789012?	`PlatformName = Ubuntu` `AccountId = 123456789012`
List my Linux managed instances.	`PlatformType = Linux`
Find my macOS managed nodes.	`PlatformType = macOS`
Show me all Amazon Linux managed nodes in my org.	`PlatformName = Amazon Linux`
List managed nodes running Amazon Linux 2 .	`PlatformName = Amazon Linux` `PlatformVersion = 2`

List the managed nodes with Ubuntu 16.04 in account 123456789012.	`PlatformName = Ubuntu`
	`PlatformVersion = 16.04`
	`AccountId = 123456789012`

Find all managed nodes that have an SSM Agent version that is not 3.3.987.0.	`AgentType = amazon-ssm-agent` `AgentVersion != 3.3.987.0`
List all managed instances that are not running a Linux operating system.	`PlatformType != Linux`

Nombres y versiones de sistemas operativos compatibles con las peticiones

Cuando pregunte a Amazon Q acerca de los nodos administrados de su cuenta, será útil proporcionar el nombre de un sistema operativo tal y como aparece en Systems Manager. También puede proporcionar números de versión para reducir aún más los resultados. Por ejemplo, como se muestra en las tablas siguientes, puede solicitar resultados específicos sobre **`macOS 14.5`**,

`Microsoft Windows Server 2019 Datacenter` y **`AlmaLinux 9.2 through 9.4`**, por citar solo algunos ejemplos.

Es posible que estas listas no sean exhaustivas y se ofrecen solo como ejemplos.

macOS

nombre de la plataforma	Números de versión
macOS	11.6.7, 11.7.10, 12.6.6, 12.7.6, 13.2, 13.4, 13.7, 14.1, 14.5, 14.6.1, 15.0

Windows

Versiones	Números de versión
Microsoft Windows Server 2008 Enterprise	6.0.6003
Microsoft Windows Server 2008 R2 Datacenter	6.1.7601
Microsoft Windows Server 2008 R2 Enterprise	6.1.7601
Microsoft Windows Server 2008 R2 Standard	6.1.7601
Microsoft Windows Server 2012 R2 Datacenter	6.3.9600
Microsoft Windows Server 2012 R2 Standard	6.3.9600
Microsoft Windows Server 2012 Standard	6.2.9200
Microsoft Windows Server 2016 Datacenter	N/D
Microsoft Windows Server 2016 Standard	10.0.14393
Microsoft Windows Server 2019 Datacenter	N/D
Microsoft Windows Server 2019 Standard	N/D
Microsoft Windows Server 2022 Datacenter	N/D
Microsoft Windows Server 2022 Standard	10.0.20348

Linux

Nombres de plataformas	Números de versión
AlmaLinux	8.10, 9.2, 9.3, 9.4
Amazon Linux 2	2.0 y versiones posteriores
Amazon Linux 2023	2023.0.20230315.0 y versiones posteriores
Amazon Linux	2015.03, 2015.09, 2016.03, 2016.09, 2017.03, 2017.09, 2018.03
Bottlerocket	1.14.3, 1.16.1, 1.18.0, 1.19.1, 1.19.2, 1.19.5, 1.20.0, 1.20.1, 1.20.2, 1.20.3, 1.20.5, 1.21.1, 1.23.0, 1.24.0, 1.24.1, 1.25.0, 1.26.1,
CentOS	6.7, 6.8, 6.9 (final), 6.10, 6.10 (final)
CentOS Linux	7, 7.2.1511, 7.2.1511 (núcleo), 7.3.1611, 7.3.1611 (núcleo), 7.4.1708, 7.5.1804, 7.5.1804 (núcleo), 7.6.1810, 7.7.1908, 7.8.2003, 7.9.2009, 7.9.2009 (núcleo), 8.2.2004, 8.5.2111
CentOS Stream	8, 9
Debian GNU/Linux	8 a 12
Servidor de Oracle Linux	7.8, 8.2, 8.3, 8.8, 8.9, 8.10, 9.4
Red Hat Enterprise Linux	8.2, 8.3, 8.4, 8.5, 8.6, 8.7, 8.8, 8.9, 8.10, 9.2, 9.3, 9.4
Servidor de Red Hat Enterprise Linux	6.9 (Santiago), 6.10 (Santiago), 7.3, 7.6, 7.7, 7.8,7.9
Rocky Linux	8.6, 8.7, 8.8, 8.9, 8.10, 9.1, 9.2, 9.3, 9.4
SLES	12.4, 15, 15.1, 15.2, 15.3, 15.4, 15.5
Ubuntu	14.04, 16.04, 18.04, 20.04, 22.04, 24.04

Exploración de los nodos administrados con Amazon Q

La integración de Systems Manager con Amazon Q Developer le permite hacer preguntas sobre los nodos administrados de su flota desde cualquier lugar en la AWS Management Console donde está disponible la interfaz de Amazon Q.

Para obtener más información sobre cómo interactuar con Amazon Q, consulte Chatting with Amazon Q Developer about AWS en la Guía del usuario de Amazon Q Developer.

Exploración de los nodos administrados con Amazon Q

1. Desde cualquier lugar de la AWS Management Console, elija el icono de Amazon Q ().
2. En el campo de peticiones de la parte inferior del panel de Amazon Q, haga una pregunta sobre los nodos administrados de su cuenta u organización.

   > **Tip**
   >
   > Para obtener consejos sobre cómo crear peticiones eficaces, consulte la información de Información sobre cómo crear peticiones efectivas para preguntar a Amazon Q sobre su flota.

3. Examine la información sobre nodos específicos o elija Abrir la consola de AWS Systems Manager para seguir explorando.

Visualización de los detalles de los nodos individuales y realización de acciones en un nodo

En una lista de la página Explorar nodos de Systems Manager, puede seleccionar un nodo individual para ver detalles completos sobre el equipo o realizar diversas acciones en el nodo. La página General de la página de detalles presenta información completa sobre el nodo.

Visualización de los detalles de cada nodo y realización de acciones en un nodo

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Explorar nodos.

3. (Opcional) Siga los pasos que se indican en Elección de una vista de filtros para los resúmenes de nodos administrados para refinar la lista de nodos administrados que se muestran en su organización o cuenta.
4. En la columna ID de nodo, seleccione el ID vinculado de un nodo.
5. Para ver más detalles sobre el nodo, en el panel de navegación izquierdo, en la lista Propiedades, elija una propiedad para ver más información acerca de los siguientes elementos:
   - Etiquetas: muestra una lista de etiquetas aplicadas al nodo. También puede agregar o eliminar etiquetas.
   - Inventario: elija un tipo de inventario, como AWS:Application o AWS:Network, para ver los detalles del inventario del nodo.
   - Asociaciones: consulte los detalles de todas las asociaciones de State Manager aplicadas al nodo, lo que incluye los detalles como el estado y el nombre del documento de SSM asociado.
   - Revisiones: consulte información resumida sobre las revisiones y el estado de las revisiones del nodo.
   - Cumplimiento de la configuración: consulte los detalles de cumplimiento del nodo, como el estado de cumplimiento y la gravedad del problema de cumplimiento.
6. Para realizar acciones en el nodo, utilice las siguientes opciones del menú Acciones del nodo:

   > **Note**
   >
   > Estas acciones solo están disponibles para los nodos administrados de la región y Cuenta de AWS en la que trabaja actualmente. En el caso de los nodos administrados a los que pueda tener acceso en otras cuentas o regiones, puede acceder a una lista Propiedades.

   - Conectar, iniciar sesión de terminal: conéctese al nodo mediante AWS Systems Manager Session Manager.
   - Herramientas
     - Ver el sistema de archivos: examine el contenido de la estructura de directorios del nodo. Agregue directorios, cámbieles el nombre y elimínelos. Corte o copie y pegue archivos.
     - Ver contadores de rendimiento: consulte la información de rendimiento del nodo, como el uso de la CPU, el tráfico de red y otros tipos de uso.

- Procesos administrados: consulte la información sobre el uso de los recursos en el nodo. Inicie o detenga los procesos en el nodo.
- Administre usuarios y grupos: vea, agregue o elimine cuentas de usuario y grupos de usuarios en el nodo.
- Ejecutar el comando run: utilice AWS Systems Manager Run Command para administrar la configuración del nodo. Run Command utiliza documentos de Systems Manager para hacer cambios bajo demanda, como actualizar aplicaciones o ejecutar scripts de intérprete de comandos de Linux y comandos de Windows PowerShell.
- Nodos de revisiones: utilice la característica Aplicar revisiones ahora de AWS Systems Manager Patch Manager para ejecutar una operación de aplicación de revisión bajo demanda en el nodo desde la consola.

> **Note**
>
> Las tareas anteriores también se pueden iniciar desde el menú Herramientas del menú de navegación de la izquierda.

- Configuración de nodo
  - Agregar etiquetas: aplique al nodo los pares clave-valor de etiqueta adicionales.
  - Restablecer la contraseña de usuario del nodo: establezca una nueva contraseña para un usuario específico del nodo.
  - Modificar el rol de IAM: cambie el rol de IAM asociado al nodo. Cree un nuevo rol de IAM para adjuntarlo al nodo.

Descarga o exportación de un informe de nodos administrados

Puede utilizar la característica Explorar nodos de Systems Manager para ver las listas filtradas o sin filtrar de los nodos administrados de su organización o cuenta de AWS en la consola de Systems Manager. En los casos en los que quiera ver los datos sin conexión o procesarlos en otra aplicación, puede guardar el informe como un archivo `CSV` o `JSON`.

Según el tamaño del informe, se le solicitará que descargue el informe en su equipo local o que lo exporte a un bucket de Amazon S3. Los informes se guardan en buckets de S3 solo en formato `CSV`.

Descarga o exportación de un informe de nodos administrados

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Explorar nodos.
3. (Opcional) Siga los pasos que se indican en Elección de una vista de filtros para los resúmenes de nodos administrados para refinar la lista de nodos administrados que se muestran en su organización o cuenta.
4. Elija Informe
   ( ).
5. Si aparece el cuadro de diálogo Descargar informe, haga lo siguiente:
   a. En Nombre de archivo, asigne un nombre al archivo. Se recomienda especificar un nombre que represente el ámbito del informe, como `all-organization-nodes` o `ec2-instances-out-of-date-agent`.
   b. En Columnas incluidas, especifique si desea incluir columnas de todos los detalles de los nodos disponibles o solo las que haya seleccionado para su visualización actual.

      > **Tip**
      >
      > Para obtener información acerca de la administración de las columnas de la visualización de los informes, consulte Administración del contenido y de la apariencia de los informes de nodos.

   c. En Formato de archivo, seleccione CSV o JSON, según cómo vaya a utilizar el archivo.
   d. En Encabezado de la hoja de cálculo, para incluir una fila de encabezados de columnas en un archivo CSV, seleccione Incluir fila de nombres de columnas.
   e. Elija Descargar.

   El informe se guarda en la ubicación de descarga predeterminada según la configuración del navegador.
6. Si se muestra el cuadro de diálogo Exportar a Amazon S3, haga lo siguiente:
   a. En URI de S3, introduzca el URI del bucket en el que se va a exportar el informe.

> **Tip**
>
> Para ver una lista de los buckets de la consola de Amazon S3, elija Ver. Para seleccionar de una lista de buckets de su cuenta, elija Explorar S3.

b. En Método de autorización, especifique el rol de servicio que se va a utilizar para proporcionar los permisos de exportación del informe al bucket.

   Si decide dejar que Systems Manager cree el rol de manera automática, este proporcionará todos los permisos y las instrucciones de confianza necesarios para la operación.

   Si quiere usar o crear su propio rol, el rol debe incluir los permisos y las instrucciones de confianza necesarios. Para obtener información acerca de la creación de esta función, consulta Creación de un rol de servicio personalizado para exportar informes de diagnóstico a S3.

c. Elija Enviar.

Creación de un rol de servicio personalizado para exportar informes de diagnóstico a S3

Al ver listas filtradas o sin filtrar de nodos administrados para su organización o cuenta de AWS en la página Explorar nodos de Systems Manager, puede exportar la lista como un informe a un bucket de Amazon S3 como archivo `CSV`.

Para ello, debe especificar un rol de servicio con los permisos necesarios y la política de confianza para la operación. Puede elegir que Systems Manager cree el rol de manera automática durante el proceso de descarga del informe. Si lo desea, puede crear por cuenta propia el rol y la política requerida.

Creación de un rol de servicio personalizado para exportar informes de diagnóstico a S3

1. Siga los pasos que se indican en la sección Creación de políticas mediante el editor JSON de la Guía del usuario de IAM.

   - Utilice lo siguiente para el contenido de la política y asegúrese de sustituir los *`valores de los marcadores de posición`* por su información propia.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:PutObject"
      ],
      "Resource": "arn:aws:s3:::s3-bucket-name/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "account-id"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketAcl",
        "s3:ListBucket",
        "s3:PutLifecycleConfiguration",
        "s3:GetLifecycleConfiguration"
      ],
      "Resource": "arn:aws:s3:::s3-bucket-name",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "account-id"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "ssm:ListNodes"
      ],
      "Resource": "*"
    }
  ]
}
```

- Dele un nombre a la política para reconocerla fácilmente en el siguiente paso.

2. Siga los pasos de Creación de un rol de IAM mediante políticas de confianza personalizadas (consola) de la Guía del usuario de IAM.

   - En el paso 4, introduzca la siguiente política de confianza y asegúrese de sustituir los *valores de los marcadores de posición* por su información propia.

     JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "SSMAssumeRole",
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "account-id"
        }
      }
    }
  ]
}
```

3. En el paso 10, elija Paso 2: Agregar permisos y seleccione el nombre de la política que creó en el paso anterior.

Después de crear el rol, siga los pasos que se indican en Descarga o exportación de un informe de nodos administrados para seleccionarlo.

Administración del contenido y de la apariencia de los informes de nodos

Puede utilizar la característica Explorar nodos de Systems Manager para ver las listas filtradas o sin filtrar de los nodos administrados de su organización o cuenta de AWS en la consola de Systems Manager. Puede elegir entre más de una docena de campos para incluirlos en las listas de nodos, como ID de nodo, Nombre del sistema operativo, Región, etc. También puede reordenar las columnas de las listas e informes y cambiar la forma en que se muestra la lista en la consola.

Administración del contenido y la apariencia de los informes de nodos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Explorar nodos.
3. En el área de Nodos, elija el icono de engranaje de preferencias ( ).
4. En el cuadro de diálogo Preferencias, haga lo siguiente:
   a. En Tamaño de página, elija cuántas filas se incluyen en cada pantalla de visualización de la consola: 10, 25 o 50.
   b. En Ajustar líneas, seleccione la casilla para mostrar todo el contenido de una celda con el ancho de columna disponible.
   c. En Filas rayadas, seleccione la casilla para mostrar filas alternantes de fondos claros y sombreados.
   d. En Seleccionar contenido visible, haga lo siguiente:
      - Active o desactive las columnas individuales para los informes y la visualización de la lista.
      - Para cambiar el orden de las columnas, mantenga pulsado el controlador de arrastre ( ) del nombre de una columna y arrástrelo hacia arriba o hacia abajo en la lista.
5. Elija Confirmar.

Acceso a los nodos justo a tiempo mediante Systems Manager

Con Systems Manager, mejora la seguridad de sus nodos al permitir el acceso justo a tiempo. El acceso a los nodos justo a tiempo permite a los usuarios solicitar un acceso temporal y con un límite de tiempo a los nodos, que usted solo puede aprobar cuando sea realmente necesario. Así, se elimina la necesidad de proporcionar un acceso permanente a los nodos que se administran mediante políticas de IAM. Además, Systems Manager proporciona la grabación de sesiones con el protocolo para escritorios remotos (RDP) en los nodos de Windows Server para que pueda respetar los requisitos de cumplimiento, realizar análisis de causas raíz, entre otras tareas. Para utilizar el acceso a los nodos justo a tiempo, debe configurar la consola unificada de Systems Manager.

Con el acceso a los nodos justo a tiempo, puede crear políticas de IAM detalladas para garantizar que solo los usuarios a los que usted autorice puedan enviar solicitudes de acceso a sus nodos. A continuación, creará políticas de aprobación que definen las aprobaciones necesarias para conectarse a los nodos. En el caso del acceso a los nodos justo a tiempo, existen políticas de aprobación automática y políticas de aprobación manual. Una política de aprobación automática define a qué nodos se pueden conectar los usuarios automáticamente. Las políticas de aprobación manual definen el número y los niveles de aprobaciones manuales que se deben proporcionar para acceder a los nodos especificados. Además, puede crear una política de denegación de acceso. Una política de denegación de acceso impide explícitamente la aprobación automática de las solicitudes de acceso a los nodos especificados. Una política de denegación de acceso se aplica a todas las cuentas de una organización de AWS Organizations. Las políticas de aprobación automática y manual se aplican solo a las Cuentas de AWS y Regiones de AWS en las que se crearon.

Cuando un usuario intenta conectarse a un nodo, se le solicita que escriba un motivo para hacerlo. A continuación, se evalúan sus políticas de aprobación. Según sus políticas, los usuarios se conectan automáticamente al nodo de destino o Systems Manager crea automáticamente una solicitud de aprobación manual en nombre del solicitante. A continuación, se notifica sobre la solicitud de acceso a los aprobadores especificados en la política de aprobación manual que se aplica al nodo, y estos pueden aprobar o denegar la solicitud. Los aprobadores y los solicitantes pueden recibir notificaciones por correo electrónico o a través de Amazon Q Developer en la integración de aplicaciones de chat con Slack o Microsoft Teams. Systems Manager solo concede acceso a los nodos solicitados cuando los aprobadores especificados proporcionan todas las aprobaciones necesarias. Una vez recibidas todas las aprobaciones necesarias, el usuario puede iniciar tantas sesiones en el nodo como necesite durante el periodo de acceso especificado en la política de aprobación. Systems Manager no finaliza automáticamente las sesiones de acceso a los nodos justo a tiempo. Como práctica recomendada, especifique valores para las preferencias duración máxima de la sesión y tiempo de espera de sesión inactiva. Estas preferencias impiden que los usuarios permanezcan conectados a los nodos después del periodo de acceso aprobado.

Recomendamos utilizar una combinación de políticas de aprobación para poder proteger los nodos que contienen datos críticos y, al mismo tiempo, permitir que los usuarios se conecten a los nodos menos críticos sin intervención. Por ejemplo, puede requerir aprobaciones manuales para las solicitudes de acceso a los nodos de base de datos y aprobar automáticamente las sesiones para los nodos no persistentes de la capa de presentación.

Systems Manager admite el acceso a los nodos justo a tiempo para los usuarios federados con IAM Identity Center o IAM. Cuando un usuario federado envía una solicitud de acceso, especifica el nodo de destino y el motivo por el que necesita conectarse. Systems Manager compara la identidad del

usuario con los parámetros definidos en las políticas de aprobación de la organización. Cuando se cumplen las condiciones de la política de aprobación automática o los aprobadores proporcionan las aprobaciones manualmente, el solicitante puede conectarse al nodo de destino. Cuando un usuario intenta conectarse a un nodo aprobado, Systems Manager crea y usa un token temporal para establecer la sesión.

Dado que el servicio Systems Manager gestiona la autenticación de las solicitudes de acceso y el establecimiento de sesiones, no es necesario utilizar políticas de IAM para gestionar el acceso a los nodos. Al utilizar el acceso a los nodos justo a tiempo, con Systems Manager, su organización puede acercarse a una actitud de cero privilegios permanentes, ya que solo necesita permitir que los usuarios creen solicitudes de acceso en lugar de que inicien sesiones con permisos persistentes en sus nodos. Para que pueda respetar los requisitos de cumplimiento, Systems Manager conserva todas las solicitudes de acceso durante 1 año. Systems Manager también emite eventos de EventBridge para el acceso a los nodos justo a tiempo en caso de que fallen las solicitudes de acceso y las actualizaciones de estado de las solicitudes de acceso que requieren aprobaciones manuales. Para obtener más información, consulte Monitoreo de eventos de Systems Manager con Amazon EventBridge.

Temas

- Cómo configurar acceso justo a tiempo con Systems Manager
- Cómo iniciar una sesión de acceso a los nodos justo a tiempo
- Cómo administrar las solicitudes de acceso justo a tiempo
- Cambio al acceso a los nodos justo a tiempo desde Session Manager
- Cómo deshabilitar el acceso justo a tiempo con Systems Manager
- Preguntas frecuentes sobre el acceso a los nodos justo a tiempo

Cómo configurar acceso justo a tiempo con Systems Manager

Configurar el acceso a los nodos justo a tiempo con Systems Manager implica varios pasos. En primer lugar, debe elegir los destinos en los que desee configurar el acceso a los nodos justo a tiempo. Los destinos consisten en unidades organizativas (OU) de AWS Organizations y Regiones de AWS. De forma predeterminada, los mismos destinos que eligió al configurar la consola unificada de Systems Manager se seleccionan para el acceso a los nodos justo a tiempo. Puede elegir configurar el acceso a los nodos justo a tiempo para todos los mismos destinos o para un subconjunto de los destinos que especificó al configurar la consola unificada de Systems Manager.

No es posible añadir nuevos destinos que no se seleccionaron al configurar la consola unificada de Systems Manager.

A continuación, creará políticas de aprobación para determinar cuándo se requiere una aprobación manual para conectarse a los nodos y cuándo se aprueba automáticamente. Cada cuenta de la organización administra las políticas de aprobación. También puede compartir una política desde la cuenta de administrador delegado para denegar explícitamente la aprobación automática de conexiones a nodos específicos.

> **Note**
>
> La configuración del acceso a los nodos justo a tiempo no afecta las políticas de IAM o preferencias existentes que haya configurado para Session Manager. Debe eliminar de las políticas de IAM los permisos para acciones de Session Manager, como `StartSession`, a fin de garantizar que solo se utilice el acceso a los nodos justo a tiempo cuando los usuarios intenten conectarse a los nodos. Tras configurar el acceso a los nodos justo a tiempo, le recomendamos que pruebe las políticas de aprobación con un subconjunto de usuarios y nodos para verificar que funcionen como se desee antes de eliminar los permisos para Session Manager.

Soporte de autenticación

Tenga en cuenta los siguientes detalles sobre el soporte de autenticación utilizado para el acceso a nodos justo a tiempo:

- El acceso a nodos justo a tiempo no admite el tipo de autenticación de inicio de sesión único cuando se conecta a instancias de Windows Server con escritorio remoto.
- Solo se admiten las credenciales de `AssumeRole` de AWS Security Token Service (AWS STS) de seguridad temporales. Para obtener más información, consulte los siguientes temas de la guía del usuario de IAM:
- • Credenciales de seguridad temporales en IAM
  - Comparación de credenciales de AWS STS
  - Solicitud de credenciales de seguridad temporales

En las siguientes políticas de IAM, se describen los permisos necesarios para administrar y permitir que los usuarios creen solicitudes de acceso a los nodos justo a tiempo con Systems Manager. Tras

comprobar que dispone de los permisos necesarios para utilizar el acceso a los nodos justo a tiempo con Systems Manager, puede continuar con el proceso de configuración. Reemplace cada *example resource placeholder* con su propia información.

Política de IAM para habilitar el acceso a los nodos justo a tiempo

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "QuickSetupConfigurationManagers",
            "Effect": "Allow",
            "Action": [
                "ssm-quicksetup:CreateConfigurationManager",
                "ssm-quicksetup:DeleteConfigurationManager",
                "ssm-quicksetup:GetConfiguration",
                "ssm-quicksetup:GetConfigurationManager",
                "ssm-quicksetup:GetServiceSettings",
                "ssm-quicksetup:ListConfigurationManagers",
                "ssm-quicksetup:ListConfigurations",
                "ssm-quicksetup:ListQuickSetupTypes",
                "ssm-quicksetup:ListTagsForResource",
                "ssm-quicksetup:TagResource",
                "ssm-quicksetup:UntagResource",
                "ssm-quicksetup:UpdateConfigurationDefinition",
                "ssm-quicksetup:UpdateConfigurationManager",
                "ssm-quicksetup:UpdateServiceSettings"
            ],
            "Resource": "*"
        },
        {
            "Sid": "QuickSetupDeployments",
            "Effect": "Allow",
            "Action": [
                "cloudformation:DescribeStackSetOperation",
                "cloudformation:ListStacks",
                "cloudformation:DescribeStacks",
                "cloudformation:DescribeStackResources",
                "cloudformation:ListStackSetOperations",
                "cloudformation:ListStackInstances",
                "cloudformation:DescribeStackSet",
```

```
                "cloudformation:ListStackSets",
                "cloudformation:DescribeStackInstance",
                "cloudformation:DescribeOrganizationsAccess",
                "cloudformation:ActivateOrganizationsAccess",
                "cloudformation:GetTemplate",
                "cloudformation:ListStackSetOperationResults",
                "cloudformation:DescribeStackEvents",
                "cloudformation:UntagResource",
                "ssm:DescribeAutomationExecutions",
                "ssm:GetAutomationExecution",
                "ssm:ListAssociations",
                "ssm:DescribeAssociation",
                "ssm:GetDocument",
                "ssm:ListDocuments",
                "ssm:DescribeDocument",
                "ssm:GetOpsSummary",
                "organizations:DeregisterDelegatedAdministrator",
                "organizations:DescribeAccount",
                "organizations:DescribeOrganization",
                "organizations:ListDelegatedAdministrators",
                "organizations:ListRoots",
                "organizations:ListParents",
                "organizations:ListOrganizationalUnitsForParent",
                "organizations:DescribeOrganizationalUnit",
                "organizations:ListAWSServiceAccessForOrganization",
                "iam:ListRoles",
                "iam:ListRolePolicies",
                "iam:GetRole",
                "iam:CreatePolicy",
                "cloudformation:TagResource"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudformation:RollbackStack",
                "cloudformation:CreateStack",
                "cloudformation:UpdateStack",
                "cloudformation:DeleteStack"
            ],
            "Resource": [
                "arn:aws:cloudformation:*:*:stack/StackSet-AWS-QuickSetup-
JITNA*",
```

```
                "arn:aws:cloudformation:*:*:stack/AWS-QuickSetup-*",
                "arn:aws:cloudformation:*:*:type/resource/*",
                "arn:aws:cloudformation:*:*:stack/StackSet-SSMQuickSetup"
            ]
        },
        {
            "Sid": "StackSetOperations",
            "Effect": "Allow",
            "Action": [
                "cloudformation:CreateStackSet",
                "cloudformation:UpdateStackSet",
                "cloudformation:DeleteStackSet",
                "cloudformation:DeleteStackInstances",
                "cloudformation:CreateStackInstances",
                "cloudformation:StopStackSetOperation"
            ],
            "Resource": [
                "arn:aws:cloudformation:*:*:stackset/AWS-QuickSetup-JITNA*",
                "arn:aws:cloudformation:*:*:type/resource/*",
                "arn:aws:cloudformation:*:*:stackset-target/AWS-QuickSetup-
JITNA*:*"
            ]
        },
        {
            "Sid": "IamRolesMgmt",
            "Effect": "Allow",
            "Action": [
                "iam:CreateRole",
                "iam:DeleteRole",
                "iam:GetRole",
                "iam:AttachRolePolicy",
                "iam:PutRolePolicy",
                "iam:DetachRolePolicy",
                "iam:GetRolePolicy",
                "iam:ListRolePolicies"
            ],
            "Resource": [
                "arn:aws:iam::*:role/AWS-QuickSetup-JITNA*",
                "arn:aws:iam::*:role/service-role/AWS-QuickSetup-JITNA*"
            ]
        },
        {
            "Sid": "IamPassRole",
            "Effect": "Allow",
```

```
            "Action": [
                "iam:PassRole"
            ],
            "Resource": [
                "arn:aws:iam::*:role/AWS-QuickSetup-JITNA*",
                "arn:aws:iam::*:role/service-role/AWS-QuickSetup-JITNA*"
            ],
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
                        "ssm.amazonaws.com",
                        "ssm-quicksetup.amazonaws.com",
                        "cloudformation.amazonaws.com"
                    ]
                }
            }
        },
        {
            "Sid": "SSMAutomationExecution",
            "Effect": "Allow",
            "Action": "ssm:StartAutomationExecution",
            "Resource": "arn:aws:ssm:us-east-1:111122223333:automation-
definition/AWS-EnableExplorer:*"
        },
        {
            "Sid": "SSMAssociationPermissions",
            "Effect": "Allow",
            "Action": [
                "ssm:DeleteAssociation",
                "ssm:CreateAssociation",
                "ssm:StartAssociationsOnce"
            ],
            "Resource": "arn:aws:ssm:us-east-1:111122223333:association/*"
        },
        {
            "Sid": "SSMResourceDataSync",
            "Effect": "Allow",
            "Action": [
                "ssm:CreateResourceDataSync",
                "ssm:UpdateResourceDataSync"
            ],
            "Resource": "arn:aws:ssm:us-east-1:111122223333:resource-data-sync/
AWS-QuickSetup-*"
        },
```

```
        {
            "Sid": "ListResourceDataSync",
            "Effect": "Allow",
            "Action": [
                "ssm:ListResourceDataSync"
            ],
            "Resource": "*"
        },
        {
            "Sid": "CreateServiceLinkedRoles",
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole"
            ],
            "Condition": {
                "StringEquals": {
                    "iam:AWSServiceName": [
                        "accountdiscovery.ssm.amazonaws.com",
                        "ssm.amazonaws.com",
                        "ssm-quicksetup.amazonaws.com",
                        "stacksets.cloudformation.amazonaws.com"
                    ]
                }
            },
            "Resource": "*"
        },
        {
            "Sid": "CreateStackSetsServiceLinkedRole",
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole"
            ],
            "Resource": "arn:aws:iam::*:role/aws-
service-role/stacksets.cloudformation.amazonaws.com/
AWSServiceRoleForCloudFormationStackSetsOrgAdmin"
        },
        {
            "Sid": "AllowSsmJitnaPoliciesCrudOperations",
            "Effect": "Allow",
            "Action": [
                "ssm:CreateDocument",
                "ssm:UpdateDocument",
                "ssm:UpdateDocumentDefaultVersion",
                "ssm:GetDocument",
```

```
                "ssm:DescribeDocument",
                "ssm:DeleteDocument"
            ],
            "Resource": [
                "arn:aws:ssm:us-east-1:111122223333:document/SSM-
JustInTimeAccessDenyAccessOrgPolicy"
            ],
            "Condition": {
                "StringEquals": {
                    "ssm:DocumentType": [
                        "AutoApprovalPolicy"
                    ]
                }
            }
        },
        {
            "Sid": "AllowAccessRequestOpsItemOperations",
            "Effect": "Allow",
            "Action": [
                "ssm:GetOpsItem",
                "ssm:DescribeOpsItems",
                "ssm:GetOpsSummary",
                "ssm:DeleteOpsItem",
                "ssm:ListOpsItemEvents"
            ],
            "Resource": "*"
        },
        {
            "Sid": "IdentityCenterPermissions",
            "Effect": "Allow",
            "Action": [
                "sso:DescribeRegisteredRegions",
                "sso:ListDirectoryAssociations",
                "identitystore:GetUserId",
                "identitystore:DescribeUser",
                "identitystore:DescribeGroup",
                "identitystore:ListGroupMembershipsForMember"
            ],
            "Resource": "*"
        }
    ]
}
```

Política de IAM para configurar el acceso a los nodos justo a tiempo

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowSsmJitnaPoliciesCrudOperations",
            "Effect": "Allow",
            "Action": [
                "ssm:CreateDocument",
                "ssm:UpdateDocument",
                "ssm:UpdateDocumentDefaultVersion",
                "ssm:GetDocument",
                "ssm:DescribeDocument",
                "ssm:DeleteDocument"
            ],
            "Resource": [
                "arn:aws:ssm:us-east-1:111122223333:document/*"
            ],
            "Condition": {
                "StringEquals": {
                    "ssm:DocumentType": [
                        "ManualApprovalPolicy",
                        "AutoApprovalPolicy"
                    ]
                }
            }
        },
        {
            "Sid": "AllowSsmJitnaPoliciesListOperations",
            "Effect": "Allow",
            "Action": [
                "ssm:ListDocuments",
                "ssm:ListDocumentVersions"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
```

```
            "Resource": "arn:aws:iam::111122223333:role/SSM-
JustInTimeAccessTokenRole",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
                        "justintimeaccess.ssm.amazonaws.com"
                    ]
                }
            }
        },
        {
            "Sid": "AllowAccessRequestOpsItemOperations",
            "Effect": "Allow",
            "Action": [
                "ssm:GetOpsItem",
                "ssm:DescribeOpsItems",
                "ssm:GetOpsSummary",
                "ssm:DeleteOpsItem",
                "ssm:ListOpsItemEvents"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowSessionManagerPreferencesOperation",
            "Effect": "Allow",
            "Action": [
                "ssm:CreateDocument",
                "ssm:GetDocument",
                "ssm:DescribeDocument",
                "ssm:UpdateDocument",
                "ssm:DeleteDocument"
            ],
            "Resource": "arn:aws:ssm:us-east-1:111122223333:document/SSM-
SessionManagerRunShell",
            "Condition": {
                "StringEquals": {
                    "ssm:DocumentType": "Session"
                }
            }
        },
        {
            "Sid": "AllowSessionManagerOperations",
            "Effect": "Allow",
            "Action": [
```

```
                "ssm:DescribeSessions",
                "ssm:GetConnectionStatus",
                "ssm:TerminateSession"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowRDPConnectionRecordingOperations",
            "Effect": "Allow",
            "Action": [
                "ssm-guiconnect:UpdateConnectionRecordingPreferences",
                "ssm-guiconnect:GetConnectionRecordingPreferences",
                "ssm-guiconnect:DeleteConnectionRecordingPreferences"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowRDPConnectionRecordingKmsOperation",
            "Effect": "Allow",
            "Action": [
                "kms:CreateGrant"
            ],
            "Resource": "arn:aws:kms:us-east-1:111122223333:key/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/SystemsManagerJustInTimeNodeAccessManaged":
"true"
                },
                "StringLike": {
                    "kms:ViaService": "ssm-guiconnect.*.amazonaws.com"
                },
                "Bool": {
                    "aws:ViaAWSService": "true"
                }
            }
        },
        {
            "Sid": "AllowFleetManagerOperations",
            "Effect": "Allow",
            "Action": [
                "ssm-guiconnect:GetConnection",
                "ssm-guiconnect:ListConnections"
            ],
            "Resource": "*"
```

```
        },
        {
            "Sid": "SNSTopicManagement",
            "Effect": "Allow",
            "Action": [
                "sns:CreateTopic",
                "sns:SetTopicAttributes"
            ],
            "Resource": [
                "arn:aws:sns:us-east-1:111122223333:SSM-JITNA*"
            ]
        },
        {
            "Sid": "SNSListTopics",
            "Effect": "Allow",
            "Action": [
                "sns:ListTopics"
            ],
            "Resource": "*"
        },
        {
            "Sid": "EventBridgeRuleManagement",
            "Effect": "Allow",
            "Action": [
                "events:PutRule",
                "events:PutTargets"
            ],
            "Resource": [
                "arn:aws:events:us-east-1::rule/SSM-JITNA*"
            ]
        },
        {
            "Sid": "ChatbotSlackManagement",
            "Effect": "Allow",
            "Action": [
                "chatbot:CreateSlackChannelConfiguration",
                "chatbot:UpdateSlackChannelConfiguration",
                "chatbot:DescribeSlackChannelConfigurations",
                "chatbot:DescribeSlackWorkspaces",
                "chatbot:DeleteSlackChannelConfiguration",
                "chatbot:RedeemSlackOauthCode",
                "chatbot:DeleteSlackWorkspaceAuthorization",
                "chatbot:GetSlackOauthParameters"
            ],
```

```
            "Resource": "*"
        },
        {
            "Sid": "ChatbotTeamsManagement",
            "Effect": "Allow",
            "Action": [
                "chatbot:ListMicrosoftTeamsChannelConfigurations",
                "chatbot:CreateMicrosoftTeamsChannelConfiguration",
                "chatbot:UpdateMicrosoftTeamsChannelConfiguration",
                "chatbot:ListMicrosoftTeamsConfiguredTeams",
                "chatbot:DeleteMicrosoftTeamsChannelConfiguration",
                "chatbot:RedeemMicrosoftTeamsOauthCode",
                "chatbot:DeleteMicrosoftTeamsConfiguredTeam",
                "chatbot:GetMicrosoftTeamsOauthParameters",
                "chatbot:TagResource"
            ],
            "Resource": "*"
        },
        {
            "Sid": "SSMEmailSettings",
            "Effect": "Allow",
            "Action": [
                "ssm:UpdateServiceSetting",
                "ssm:GetServiceSetting"
            ],
            "Resource": [
                "arn:aws:ssm:us-east-1:111122223333:servicesetting/ssm/access-
request/email-role-mapping",
                "arn:aws:ssm:us-east-1:111122223333:servicesetting/ssm/access-
request/enabled-email-notifications"
            ]
        },
        {
            "Sid": "AllowViewingJitnaCloudWatchMetrics",
            "Effect": "Allow",
            "Action": [
                "cloudwatch:GetMetricData",
                "cloudwatch:GetMetricStatistics",
                "cloudwatch:ListMetrics"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "cloudwatch:namespace": "AWS/SSM/JustInTimeAccess"
```

```
                }
            }
        },
        {
            "Sid": "QuickSetupConfigurationManagers",
            "Effect": "Allow",
            "Action": [
                "ssm-quicksetup:ListConfigurationManagers",
                "ssm-quicksetup:ListConfigurations",
                "ssm-quicksetup:ListQuickSetupTypes",
                "ssm-quicksetup:GetConfiguration",
                "ssm-quicksetup:GetConfigurationManager"
            ],
            "Resource": "*"
        },
        {
            "Sid": "QuickSetupDeployments",
            "Effect": "Allow",
            "Action": [
                "cloudformation:ListStacks",
                "cloudformation:DescribeStacks",
                "organizations:DescribeOrganization",
                "organizations:ListDelegatedAdministrators"
            ],
            "Resource": "*"
        },
        {
            "Sid": "ManualPolicy",
            "Effect": "Allow",
            "Action": [
                "sso:DescribeRegisteredRegions",
                "ssm:GetServiceSetting",
                "iam:ListRoles"
            ],
            "Resource": "*"
        },
        {
            "Sid": "SessionPreference",
            "Effect": "Allow",
            "Action": [
                "s3:ListAllMyBuckets"
            ],
            "Resource": "*"
        },
```

```
        {
            "Sid": "AllowIamListForKMS",
            "Effect": "Allow",
            "Action": [
                "iam:ListUsers"
            ],
            "Resource": "arn:aws:iam::111122223333:user/*"
        },
        {
            "Sid": "KMSPermission",
            "Effect": "Allow",
            "Action": [
                "kms:TagResource",
                "kms:ListAliases",
                "kms:CreateAlias"
            ],
            "Resource": "*"
        },
        {
            "Sid": "KMSCreateKey",
            "Effect": "Allow",
            "Action": [
                "kms:CreateKey"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/SystemsManagerJustInTimeNodeAccessManaged":
"true"
                },
                "ForAllValues:StringEquals": {
                    "aws:TagKeys": [
                        "SystemsManagerJustInTimeNodeAccessManaged"
                    ]
                }
            }
        },
        {
            "Sid": "AllowIamRoleForChatbotAction",
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::111122223333:role/role name",
```

```
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
                        "chatbot.amazonaws.com"
                    ]
                }
            }
        },
        {
            "Sid": "AllowIamServiceRoleForChat",
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole"
            ],
            "Resource": "arn:aws:iam::111122223333:role/aws-service-role/
management.chatbot.amazonaws.com/AWSServiceRoleForAWSChatbot"
        },
        {
            "Sid": "CloudWatchLogs",
            "Effect": "Allow",
            "Action": [
                "logs:DescribeLogGroups"
            ],
            "Resource": "arn:aws:logs:*:111122223333:log-group::log-stream:"
        },
        {
            "Sid": "IdentityStorePermissions",
            "Effect": "Allow",
            "Action": [
                "sso:ListDirectoryAssociations",
                "identitystore:GetUserId",
                "sso-directory:SearchUsers",
                "sso-directory:SearchGroups",
                "identitystore:DescribeGroup",
                "identitystore:DescribeUser"
            ],
            "Resource": "*"
        }
    ]
}
```

Política de IAM para los aprobadores de solicitudes de acceso

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowAccessRequestDescriptions",
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeOpsItems",
                "ssm:GetOpsSummary",
                "ssm:ListOpsItemEvents"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowGetSpecificAccessRequest",
            "Effect": "Allow",
            "Action": [
                "ssm:GetOpsItem"
            ],
            "Resource": "arn:aws:ssm:us-east-1:111122223333:opsitem/*"
        },
        {
            "Sid": "AllowApprovalRejectionSignal",
            "Effect": "Allow",
            "Action": [
                "ssm:SendAutomationSignal"
            ],
            "Resource": "arn:aws:ssm:*:*:automation-execution/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/SystemsManagerJustInTimeNodeAccessManaged":
 "true"
                }
            }
        },
        {
            "Sid": "QuickSetupConfigurationManagers",
            "Effect": "Allow",
            "Action": [
```

```
                "ssm-quicksetup:ListConfigurationManagers",
                "ssm-quicksetup:ListConfigurations",
                "ssm-quicksetup:GetConfigurationManager",
                "ssm-quicksetup:ListQuickSetupTypes",
                "ssm-quicksetup:GetConfiguration"
            ],
            "Resource": "*"
        },
        {
            "Sid": "QuickSetupDeployments",
            "Effect": "Allow",
            "Action": [
                "cloudformation:ListStacks",
                "cloudformation:DescribeStacks",
                "organizations:DescribeOrganization",
                "organizations:ListDelegatedAdministrators"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowSsmJitnaPoliciesCrudOperations",
            "Effect": "Allow",
            "Action": [
                "ssm:GetDocument",
                "ssm:DescribeDocument"
            ],
            "Resource": [
                "arn:aws:ssm:us-east-1:111122223333:document/*"
            ],
            "Condition": {
                "StringEquals": {
                    "ssm:DocumentType": [
                        "ManualApprovalPolicy",
                        "AutoApprovalPolicy"
                    ]
                }
            }
        },
        {
            "Sid": "AllowSsmJitnaPoliciesListOperations",
            "Effect": "Allow",
            "Action": [
                "ssm:ListDocuments",
                "ssm:ListDocumentVersions"
```

```
            ],
            "Resource": "*"
        },
        {
            "Sid": "IDCPermissions",
            "Effect": "Allow",
            "Action": [
                "sso:DescribeRegisteredRegions",
                "sso:ListDirectoryAssociations",
                "identitystore:GetUserId",
                "identitystore:DescribeUser",
                "identitystore:DescribeGroup",
                "identitystore:ListGroupMembershipsForMember"
            ],
            "Resource": "*"
        }
    ]
}
```

Política de IAM para usuarios con acceso a los nodos justo a tiempo

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowJITNAOperations",
            "Effect": "Allow",
            "Action": [
                "ssm:StartAccessRequest",
                "ssm:GetAccessToken"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowOpsItemCreationAndRetrieval",
            "Effect": "Allow",
            "Action": [
                "ssm:CreateOpsItem",
                "ssm:GetOpsItem"
            ],
```

```
            "Resource": "arn:aws:ssm:*:*:opsitem/*"
        },
        {
            "Sid": "AllowListAccessRequests",
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeOpsItems",
                "ssm:GetOpsSummary",
                "ssm:ListOpsItemEvents",
                "ssm:DescribeSessions"
            ],
            "Resource": "*"
        },
        {
            "Sid": "RequestManualApprovals",
            "Action": "ssm:StartAutomationExecution",
            "Effect": "Allow",
            "Resource": "arn:aws:ssm:*:*:document/*",
            "Condition": {
                "StringEquals": {
                    "ssm:DocumentType": "ManualApprovalPolicy"
                }
            }
        },
        {
            "Sid": "StartManualApprovalsAutomationExecution",
            "Effect": "Allow",
            "Action": "ssm:StartAutomationExecution",
            "Resource": "arn:aws:ssm:*:*:automation-execution/*"
        },
        {
            "Sid": "AllowManualApprovalAutomationExecutionTagging",
            "Effect": "Allow",
            "Action": [
                "ssm:AddTagsToResource"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:automation-execution/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/SystemsManagerJustInTimeNodeAccessManaged":
"true"
                },
```

```
                "ForAllValues:StringEquals": {
                    "aws:TagKeys": [
                        "SystemsManagerJustInTimeNodeAccessManaged"
                    ]
                }
            }
        },
        {
            "Sid": "CancelAccessRequestManualApproval",
            "Effect": "Allow",
            "Action": "ssm:StopAutomationExecution",
            "Resource": "arn:aws:ssm:*:*:automation-execution/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/SystemsManagerJustInTimeNodeAccessManaged":
"true"
                }
            }
        },
        {
            "Sid": "DescribeEC2Instances",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeTags",
                "ec2:GetPasswordData"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowListSSMManagedNodesAndTags",
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeInstanceInformation",
                "ssm:ListTagsForResource"
            ],
            "Resource": "*"
        },
        {
            "Sid": "QuickSetupConfigurationManagers",
            "Effect": "Allow",
            "Action": [
                "ssm-quicksetup:ListConfigurationManagers",
                "ssm-quicksetup:GetConfigurationManager",
```

```
                "ssm-quicksetup:ListConfigurations",
                "ssm-quicksetup:ListQuickSetupTypes",
                "ssm-quicksetup:GetConfiguration"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowSessionManagerOperations",
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeSessions",
                "ssm:GetConnectionStatus"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowRDPOperations",
            "Effect": "Allow",
            "Action": [
                "ssm-guiconnect:ListConnections",
                "ssm:GetConnectionStatus"
            ],
            "Resource": "*"
        },
        {
            "Sid": "QuickSetupDeployments",
            "Effect": "Allow",
            "Action": [
                "cloudformation:ListStacks",
                "cloudformation:DescribeStacks",
                "organizations:DescribeOrganization",
                "organizations:ListDelegatedAdministrators"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowSsmJitnaPoliciesReadOnly",
            "Effect": "Allow",
            "Action": [
                "ssm:GetDocument",
                "ssm:DescribeDocument"
            ],
            "Resource": [
                "arn:aws:ssm:*:111122223333:document/*"
```

```
            ],
            "Condition": {
                "StringEquals": {
                    "ssm:DocumentType": [
                        "ManualApprovalPolicy",
                        "AutoApprovalPolicy"
                    ]
                }
            }
        },
        {
            "Sid": "AllowSsmJitnaPoliciesListOperations",
            "Effect": "Allow",
            "Action": [
                "ssm:ListDocuments",
                "ssm:ListDocumentVersions"
            ],
            "Resource": "*"
        },
        {
            "Sid": "ExploreNodes",
            "Effect": "Allow",
            "Action": [
                "ssm:ListNodesSummary",
                "ssm:ListNodes",
                "ssm:DescribeInstanceProperties"
            ],
            "Resource": "*"
        },
        {
            "Sid": "IdentityStorePermissions",
            "Effect": "Allow",
            "Action": [
                "sso:DescribeRegisteredRegions",
                "sso:ListDirectoryAssociations",
                "identitystore:GetUserId",
                "identitystore:DescribeUser",
                "identitystore:DescribeGroup"
            ],
            "Resource": "*"
        }
    ]
}
```

> **Note**
>
> Para restringir el acceso a las operaciones de la API que crean, actualizan o eliminan políticas de aprobación, utilice la clave de condición `ssm:DocumentType` para los tipos de documento `AutoApprovalPolicy` y `ManualApprovalPolicy` . Las operaciones `StartAccessRequest` y `GetAccessToken` de la API no admiten las siguientes claves de contexto globales:
>
> - aws:ViaAwsService
> - aws:MultiFactorAuthPresent
> - aws:SourceVpce
> - aws:UserAgent

Para obtener más información sobre las claves de contexto de condición para Systems Manager, consulte Claves de condición para AWS Systems Manager en la Referencia de autorizaciones de servicio.

En el siguiente procedimiento, se describe cómo completar el primer paso de la configuración del acceso a los nodos justo a tiempo.

Cómo configurar el acceso a los nodos justo a tiempo

1. Inicie sesión en la cuenta de administrador delegado de Systems Manager de la organización.
2. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
3. En el panel de navegación, seleccione Acceso a nodos justo a tiempo.
4. Seleccione Habilitar la nueva experiencia.
5. Elija las regiones donde desee habilitar el acceso a los nodos justo a tiempo. De forma predeterminada, las mismas regiones que eligió al configurar la consola unificada de Systems Manager se seleccionan para el acceso a los nodos justo a tiempo. No es posible añadir nuevas regiones que no se seleccionaron al configurar la consola unificada de Systems Manager.
6. Seleccione Habilitar el acceso a los nodos justo a tiempo.

No se aplica ningún cargo por usar el acceso a los nodos justo a tiempo durante los 30 días posteriores a la activación de la característica. Después del periodo de prueba de 30 días, se aplicará un cargo por utilizarla. Para más información, consulte Precios de AWS Systems Manager.

Cómo crear políticas de aprobación para sus nodos

Las políticas de aprobación definen qué aprobaciones necesitan los usuarios para acceder a un nodo. Dado que el acceso a los nodos justo a tiempo elimina la necesidad de contar con permisos permanentes para acceder a los nodos mediante políticas de IAM, usted debe crear políticas de aprobación para permitir el acceso. Si no hay políticas de aprobación que se apliquen a un nodo, los usuarios no podrán solicitar el acceso.

En el acceso a los nodos justo a tiempo, hay tres tipos de políticas: aprobación automática, denegación de acceso y aprobación manual.

Tipos de políticas de acceso a los nodos justo a tiempo

- Una política de aprobación automática define a qué nodos se pueden conectar los usuarios automáticamente.
- Las políticas de aprobación manual definen el número y los niveles de aprobaciones manuales que se deben proporcionar para acceder a los nodos especificados.
- Una política de denegación de acceso impide explícitamente la aprobación automática de las solicitudes de acceso a los nodos especificados.

Una política de denegación de acceso se aplica a todas las cuentas de una organización de AWS Organizations. Por ejemplo, puede denegar explícitamente las aprobaciones automáticas para el grupo `Intern` a los nodos etiquetados con la clave `Production`. Las políticas de aprobación automática y manual se aplican solo a las Cuentas de AWS y Regiones de AWS en las que se crearon. Cada cuenta de miembro de su organización administra sus propias políticas de aprobación. Las políticas de aprobación se evalúan en el siguiente orden:

1. Denegación de acceso
2. Aprobación automática
3. Manual

Aunque solo puede tener una política de denegación de acceso por organización y una política de aprobación automática por cuenta y región, es probable que tenga varias políticas de aprobación

manual en una cuenta. Al evaluar las políticas de aprobación manual, el acceso a los nodos justo a tiempo siempre favorece la política más específica para un nodo. Las políticas de aprobación manual se evalúan en el siguiente orden:

1. Destino específico según la etiqueta
2. Todos los destinos de los nodos

Por ejemplo, tiene un nodo etiquetado con la clave `Demo`. En la misma cuenta, tiene una política de aprobación manual que se dirige a todos los nodos y requiere una aprobación de un nivel. También tiene una política de aprobación manual que requiere dos aprobaciones de dos niveles para los nodos etiquetados con la clave `Demo`. Systems Manager aplica la política que dirige la etiqueta `Demo` al nodo, ya que es más específica que la política que se dirige a todos los nodos. Esto le permite crear una política general para todos los nodos de la cuenta, lo que garantiza que los usuarios puedan enviar solicitudes de acceso y, al mismo tiempo, le permite crear políticas más detalladas según sea necesario.

Según su organización, es posible que se apliquen varias etiquetas a los nodos. En esta situación, si se aplican varias políticas de aprobación manual a un nodo, las solicitudes de acceso fallan. Por ejemplo, un nodo está etiquetado con las claves `Production` y `Database`. En la misma cuenta, tiene una política de aprobación manual que se aplica a los nodos etiquetados con la clave `Production` y otra política de aprobación manual que se aplica a los nodos etiquetados con la clave `Database`. Esto provoca un conflicto en el nodo que está etiquetado con ambas claves, y las solicitudes de acceso fallan. Systems Manager redirige al usuario a la solicitud fallida. Allí, verá los detalles sobre las políticas y etiquetas en conflicto para poder realizar los ajustes necesarios si dispone de los permisos requeridos. De lo contrario, puede notificar a un colega en la organización que tenga los permisos necesarios para modificar las políticas. Los conflictos de políticas que provocan solicitudes de acceso fallidas emiten eventos de EventBridge, lo que le ofrece flexibilidad para crear sus propios flujos de trabajo de respuesta. Además, Systems Manager les envía, a los destinatarios especificados, notificaciones por correo electrónico en caso de conflictos de políticas que provoquen solicitudes de acceso fallidas. Para obtener más información acerca de la configuración de notificaciones por correo electrónico para políticas en conflicto, consulte Cómo configurar las notificaciones para las solicitudes de acceso justo a tiempo.

En una política de denegación de acceso, se utiliza el lenguaje de políticas de Cedar para definir, de forma explícita, a qué nodos no se pueden conectar automáticamente los usuarios en la organización. Esta política se crea y se comparte desde la cuenta de administrador delegado de

su organización. La política de denegación de acceso tiene prioridad sobre todas las políticas de aprobación automática. Solo puede tener una política de denegación de acceso por organización.

En una política de aprobación automática, se utiliza el lenguaje de políticas de Cedar para definir qué usuarios pueden conectarse automáticamente a los nodos especificados sin aprobación manual. Si una solicitud de acceso se aprueba automáticamente, la duración del acceso es de 1 hora. Este valor no se puede cambiar. Solo puede tener una política de aprobación automática por cuenta y región.

En una política de aprobación manual, se especifica la duración del acceso, cuántos niveles de aprobación se requieren, el número de aprobadores necesarios por nivel y los nodos a los que pueden aprobar las solicitudes de acceso justo a tiempo. La duración del acceso para una política de aprobación manual debe estar entre 1 y 336 horas. Si especifica varios niveles de aprobaciones, las aprobaciones de la solicitud de acceso se procesan de un nivel a la vez. Esto significa que todas las aprobaciones que necesite para un nivel deben proporcionarse antes de que el proceso de aprobación pase a los niveles siguientes. Si especifica varias etiquetas en una política de aprobación manual, se evalúan como instrucciones `or`, no como instrucciones `and`. Por ejemplo, si crea una política de aprobación manual que incluye las etiquetas `Application`, `Web` y `Test`, la política se aplica a cualquier nodo que esté etiquetado con una de esas claves. La política no se aplica solo a los nodos que están etiquetados con las tres claves.

Recomendamos utilizar una combinación de políticas manuales con la política de aprobación automática para ayudar a proteger los nodos que tienen datos críticos y, al mismo tiempo, permitir que los usuarios se conecten a los nodos menos críticos sin intervención. Por ejemplo, puede requerir aprobaciones manuales para las solicitudes de acceso a los nodos de base de datos y aprobar automáticamente las sesiones para los nodos no persistentes de la capa de presentación.

En los siguientes procedimientos, se describe cómo crear políticas de aprobación para el acceso a los nodos justo a tiempo.

Temas

- Cómo crear políticas de aprobación manual sobre el acceso a los nodos justo a tiempo
- Estructura de instrucciones y operadores integrados para políticas de aprobación automática y denegación de acceso
- Cómo crear una política de aprobación automática sobre el acceso a los nodos justo a tiempo
- Cómo crear una política de denegación sobre el acceso a los nodos justo a tiempo
- Cómo crear políticas de aprobación para el acceso a los nodos justo a tiempo con Amazon Q

Cómo crear políticas de aprobación manual sobre el acceso a los nodos justo a tiempo

En el siguiente procedimiento, se describe cómo crear políticas de aprobación manual. Systems Manager le permite crear hasta 50 políticas de aprobación manual por Cuenta de AWS y Región de AWS.

Cómo crear una política de aprobación manual

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, seleccione Administrar acceso a nodos.
3. En la sección Detalles de la política del paso Crear política de aprobación manual, introduzca un nombre y una descripción para la política de aprobación.
4. Introduzca un valor en Duración del acceso. Esta es la cantidad máxima de tiempo que un usuario puede tener la sesión iniciada en un nodo después de que se apruebe una solicitud de acceso. Este valor debe estar entre 1 y 336 horas.
5. En la sección Destinos de los nodos, introduzca los pares clave-valor de etiquetas asociados a los nodos sobre los que desea que se aplique la política. Si ninguna de las etiquetas especificadas en la política está asociada a un nodo, entonces esta no se aplica.
6. En la sección Aprobadores de solicitudes de acceso, introduzca los usuarios o grupos que desee que puedan aprobar las solicitudes de acceso a los destinos de los nodos en la política. Los aprobadores de las solicitudes de acceso pueden ser usuarios y grupos de IAM Identity Center o roles de IAM. Puede especificar hasta 5 aprobadores por nivel y hasta 5 niveles de aprobadores.
7. Seleccione Crear política de aprobación manual.

Estructura de instrucciones y operadores integrados para políticas de aprobación automática y denegación de acceso

En la siguiente tabla, se muestra la estructura de las políticas de aprobación automática y denegación de acceso.

Componente	Sintaxis
effect	`permit | forbid`
scope	`(principal, action, resource)`

Componente	Sintaxis
cláusula de condición	`when { principal or resource has attribute name };`

Componentes de política

Una política de aprobación automática o denegación de acceso contiene los siguientes componentes:

- Efecto: `permit` (permitir) o `forbid` (denegar) el acceso.
- Alcance: las entidades principales, las acciones y los recursos a los que se aplica el efecto. Puede dejar el alcance de Cedar sin definir si no identifica entidades principales, acciones o recursos específicos. En este caso, la política se aplica a todas las entidades principales, acciones y recursos posibles. Para el acceso a los nodos justo a tiempo, `action` siempre es `AWS::SSM::Action::"getTokenForInstanceAccess"`.
- Cláusula de condición: el contexto en el que se aplica el efecto.

Comentarios

Puede incluir comentarios en sus políticas de . Los comentarios se definen como una línea que comienza por `//` y termina con un carácter de nueva línea.

En el siguiente ejemplo se muestran comentarios en una política.

```
// Allows users in the Engineering group from the Platform org to automatically connect
 to nodes tagged with Engineering and Production keys.
permit (
    principal in AWS::IdentityStore::Group::"d4q81745-r081-7079-d789-14da1EXAMPLE",
    action == AWS::SSM::Action::"getTokenForInstanceAccess",
    resource
)
when {
    principal has organization && resource.hasTag("Engineering") &&
 resource.hasTag("Production") && principal.organization == "Platform"
};
```

Cláusulas múltiples

Puede utilizar más de una cláusula de condición en una declaración de política mediante el operador &&.

```
// Allow access if node has tag where the tag key is Environment
// & tag value is Development

permit(principal, action == AWS::SSM::getTokenForInstanceAccess, resource)
when {
    resource.hasTag("Environment") &&
    resource.getTag("Environment") == "Development"
};
```

Caracteres reservados

El siguiente ejemplo muestra cómo escribir una política si una propiedad de contexto utiliza un `:` (punto y coma), que es un carácter reservado en el lenguaje de la política.

```
permit (
    principal,
    action == AWS::SSM::Action::"getTokenForInstanceAccess",
    resource
)
when {
    principal has employeeNumber && principal.employeeNumber like "E-1*" &&
 resource.hasTag("Purpose") && resource.getTag("Purpose") == "Testing"
}
```

Para ver otros ejemplos, consulte Ejemplo de instrucciones de políticas.

Esquema de acceso a los nodos justo a tiempo

A continuación, se muestra el esquema Cedar para el acceso a los nodos justo a tiempo.

```
namespace AWS::EC2 {
    entity Instance tags String;
}


namespace AWS::IdentityStore {
```

```
    entity Group;

    entity User in [Group] {
    employeeNumber?: String,
    costCenter?: String,
    organization?: String,
    division?: String,
    };

}


namespace AWS::IAM {

    entity Role;

    type AuthorizationContext = {
        principalTags: PrincipalTags,
    };

    entity PrincipalTags tags String;
}

namespace AWS::SSM {

    entity ManagedInstance tags String;

    action "getTokenForInstanceAccess" appliesTo {
    principal: [AWS::IdentityStore::User],
    resource: [AWS::EC2::Instance, AWS::SSM::ManagedInstance],
    context: {
        "iam": AWS::IAM::AuthorizationContext
        }
    };
}
```

Operadores integrados

Al crear el contexto de una política de aprobación automática o denegación de acceso con varias condiciones, puede utilizar el operador && para añadir condiciones adicionales. También hay muchos otros operadores integrados que puede utilizar para añadir un poder de expresión adicional a las condiciones de su política. La siguiente tabla contiene todos los operadores integrados como referencia.

Operador	Tipos y sobrecargas	Descripción
!	Booleano → Booleano	Not lógico.
==	any → any	Igualdad. Funciona con argumentos de cualquier tipo, incluso si los tipos no coinciden. Los valores de diferentes tipos nunca son iguales entre sí.
!=	any → any	Desigualdad; exactamente lo contrario de la igualdad (ver arriba).
<	(long, long) → Booleano	Entero largo menor que.
<=	(long, long) → Booleano	Entero largo menor que o igual a.
>	(long, long) → Booleano	Entero largo mayor que.
>=	(long, long) → Booleano	Entero largo mayor que o igual a.
in	(entity, entity) → Booleano	Pertenencia jerárquica (reflexiva: A en A siempre es verdadera).
	(entidad, conjunto (entidad)) → Booleano	Pertenencia jerárquica: A en [B, C, ...] es verdadero si (A y B) || (A en C) || ... es un error si el conjunto no contiene una entidad.
&&	(Boolean, Boolean) → Booleano	Lógico y (cortocircuito).

Operador	Tipos y sobrecargas	Descripción
||	(Boolean, Boolean) → Booleano	Lógico o (cortocircuito).
.exists()	entity → Booleano	Existencia de la entidad.
has	(entity, attribute) → Booleano	Operador de infijo. `e has f` comprueba si el registro o la entidad `e` tienen un enlace para el atributo `f`. Devuelve `false` si `e` no existe o si `e` existe pero no tiene el atributo `f`. Los atributos se pueden expresar como identificadores o cadenas literales.
like	(string, string) → Booleano	Operador de infijo. `t like p` comprueba si el texto `t` coincide con el patrón `p`, que puede incluir caracteres comodín `*` que coincidan con 0 o más caracteres. Para que coincida con un carácter estrella literal en `t`, puede utilizar la secuencia especial de caracteres de escape `\*` en `p`.
.hasTag()	(entity, string) → Booleano	Comprueba si la entidad tiene aplicada la etiqueta especificada.
.getTag()	(entity, string) → Booleano	Devuelve el valor de la clave de etiqueta especificada.
.contains()	(set, any) → Booleano	Establecer pertenencia (es B un elemento de A).

Operador	Tipos y sobrecargas	Descripción
.containsAll()	(set, set) → Booleano	Comprueba si el conjunto A contiene todos los elementos del conjunto B.
.containsAny()	(set, set) → Booleano	Comprueba si el conjunto A contiene alguno de los elementos del conjunto B.

Ejemplo de instrucciones de políticas

A continuación, se muestran ejemplos de instrucciones de políticas.

```
// Users assuming IAM roles with a principal tag of "Elevated" can automatically access
 nodes tagged with the "Environment" key when the value equals "prod"
permit(principal, action == AWS::SSM::getTokenForInstanceAccess, resource)
when {
    // Verify IAM role principal tag
    context.iam.principalTags.getTag("AccessLevel") == "Elevated" &&

    // Verify the node has a tag with "Environment" tag key and a tag value of "prod"
    resource.hasTag("Environment") &&
    resource.getTag("Environment") == "prod"
};
```

```
// Identity Center users in the "Contractor" division can automatically access nodes
 tagged with the "Environment" key when the value equals "dev"
permit(principal, action == AWS::SSM::getTokenForInstanceAccess, resource)
when {
    // Verify that the user is part of the "Contractor" division
    principal.division == "Contractor" &&

    // Verify the node has a tag with "Environment" tag key and a tag value of "dev"
    resource.hasTag("Environment") &&
    resource.getTag("Environment") == "dev"
};
```

```
// Identity Center users in a specified group can automatically access nodes tagged
 with the "Environment" key when the value equals "Production"
permit(principal in AWS::IdentityStore::Group::"d4q81745-r081-7079-d789-14da1EXAMPLE",
    action == AWS::SSM::getTokenForInstanceAccess,
    resource)
when {
    resource.hasTag("Environment") &&
    resource.getTag("Environment") == "Production"
};
```

Cómo crear una política de aprobación automática sobre el acceso a los nodos justo a tiempo

En las políticas de aprobación automática, se utiliza el lenguaje de políticas Cedar para definir qué usuarios pueden conectarse automáticamente a los nodos especificados sin aprobación manual. Una política de aprobación automática contiene varias instrucciones `permit` que especifican `principal` y `resource`. Cada instrucción incluye una cláusula `when` que define las condiciones de la aprobación automática.

A continuación, se muestra un ejemplo de política de aprobación automática.

```
permit (
    principal in AWS::IdentityStore::Group::"e8c17310-e011-7089-d989-10da1EXAMPLE",
    action == AWS::SSM::Action::"getTokenForInstanceAccess",
    resource
)
when {
    principal has costCenter && resource.hasTag("CostCenter") && principal.costCenter
 == resource.getTag("CostCenter")
};

permit (
    principal in AWS::IdentityStore::Group::"d4q81745-r081-7079-d789-14da1EXAMPLE",
    action == AWS::SSM::Action::"getTokenForInstanceAccess",
    resource
)
when {
    principal has organization && resource.hasTag("Engineering") &&
 resource.hasTag("Production") && principal.organization == "Platform"
};

permit (
    principal,
```

```
    action == AWS::SSM::Action::"getTokenForInstanceAccess",
    resource
)
when {
    principal has employeeNumber && principal.employeeNumber like "E-1*" &&
 resource.hasTag("Purpose") && resource.getTag("Purpose") == "Testing"
};
```

En el siguiente procedimiento, se describe cómo crear una política de aprobación automática sobre el acceso a los nodos justo a tiempo. Si una solicitud de acceso se aprueba automáticamente, la duración del acceso es de 1 hora. Este valor no se puede cambiar. Solo puede tener una política de aprobación automática por Cuenta de AWS y Región de AWS. Para obtener más información acerca de cómo crear instrucciones de políticas, consulte Estructura de instrucciones y operadores integrados para políticas de aprobación automática y denegación de acceso.

Cómo crear una política de aprobación automática

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, seleccione Administrar acceso a nodos.
3. En la pestaña Políticas de aprobación, seleccione Crear política de aprobación automática.
4. Introduzca la instrucción de la política de aprobación automática en la sección Instrucción de la política. Puede usar las Instrucciones de ejemplo proporcionadas que le ayudarán a crear la política.
5. Seleccione Crear política de aprobación automática.

Cómo crear una política de denegación sobre el acceso a los nodos justo a tiempo

Las políticas de denegación de acceso utilizan el lenguaje de políticas Cedar para definir a qué nodos no se pueden conectar automáticamente los usuarios sin una aprobación manual. Una política de denegación de acceso contiene varias instrucciones `forbid` que especifican la `principal` y el `resource`. Cada instrucción incluye una cláusula `when` que define las condiciones para denegar explícitamente la aprobación automática.

A continuación, se muestra un ejemplo de política de denegación de acceso.

```
forbid (
    principal in AWS::IdentityStore::Group::"e8c17310-e011-7089-d989-10da1EXAMPLE",
```

```
    action == AWS::SSM::Action::"getTokenForInstanceAccess",
    resource
)
when {
    resource.hasTag("Environment") && resource.getTag("Environment") == "Production"
};

forbid (
    principal,
    action == AWS::SSM::Action::"getTokenForInstanceAccess",
    resource
)
when {
    principal has division && principal.division != "Finance" &&
 resource.hasTag("DataClassification") && resource.getTag("DataClassification") ==
 "Financial"
};


forbid (
    principal,
    action == AWS::SSM::Action::"getTokenForInstanceAccess",
    resource
)
when {

    principal has employeeNumber && principal.employeeNumber like "TEMP-*" &&
 resource.hasTag("Criticality") && resource.getTag("Criticality") == "High"
};
```

En el siguiente procedimiento, se describe cómo crear una política de denegación de acceso sobre los nodos justo a tiempo. Para obtener más información acerca de cómo crear instrucciones de políticas, consulte Estructura de instrucciones y operadores integrados para políticas de aprobación automática y denegación de acceso.

> **Note**
>
> Observe la siguiente información.
>
> - Puede crear políticas de denegación de acceso mientras está conectado a la cuenta de administración de AWS o a la cuenta de administrador delegado. La organización de AWS Organizations solo puede tener una política de denegación de acceso.

- El acceso a los nodos justo a tiempo utiliza AWS Resource Access Manager (AWS RAM) para compartir su política de denegación de acceso con las cuentas de miembro de su organización. Si desea compartir su política de denegación de acceso con las cuentas de miembro de su organización, debe habilitar el uso compartido de recursos desde la cuenta de administración de su organización. Para obtener más información, consulte Habilitar el uso compartido de recursos dentro de AWS Organizations en la Guía del usuario de AWS RAM.

Cómo crear una política de denegación de acceso

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, seleccione Administrar acceso a nodos.
3. En la pestaña Políticas de aprobación, seleccione Crear política de denegación de acceso.
4. Introduzca su instrucción para la política de denegación de acceso en la sección Instrucción de la política. Puede usar las Instrucciones de ejemplo proporcionadas que le ayudarán a crear la política.
5. Seleccione Crear política de denegación de acceso.

Cómo crear políticas de aprobación para el acceso a los nodos justo a tiempo con Amazon Q

El uso de Amazon Q Developer para la línea de comandos proporciona orientación y soporte en varios aspectos del desarrollo de software. En el caso del acceso a los nodos justo a tiempo, Amazon Q le permite crear políticas de aprobación mediante la generación y actualización del código de las políticas, el análisis de las instrucciones de las políticas y mucho más. A continuación, se describe cómo crear políticas de aprobación mediante Amazon Q para la línea de comandos.

Cómo identificar su caso de uso

El primer paso para crear políticas de aprobación es definir claramente su caso de uso. Por ejemplo, en su organización, es posible que desee aprobar automáticamente las solicitudes de acceso a los nodos con una etiqueta `Environment:Testing`. También puede denegar explícitamente las aprobaciones automáticas a los nodos con una etiqueta `Environment:Production` si el ID de un empleado comienza por `TEMP`. En el caso de los nodos con una etiqueta `Tier:Database`, es posible que desee requerir dos niveles de aprobaciones manuales.

En cualquier situación, es posible que prefiera una política o condición en lugar de otra. Por lo tanto, recomendamos que defina claramente los comportamientos de la política que desee a fin de determinar qué instrucciones se ajustan mejor a su caso de uso y sus preferencias.

Cómo configurar el entorno de desarrollo.

Instale Amazon Q para la línea de comandos donde desee crear las políticas de aprobación. Para obtener información sobre la instalación de Amazon Q para la línea de comandos, consulte Instalación de Amazon Q para la línea de comandos en la Guía del usuario de Amazon Q Developer.

También recomendamos instalar el servidor MCP para obtener la documentación de AWS. Este servidor MCP conecta Amazon Q para la línea de comandos con los recursos de documentación más actuales. Para obtener información sobre el uso de MCP con Amazon Q para la línea de comandos, consulte Using MCP with Amazon Q Developer en la Guía del usuario de Amazon Q Developer.

Para obtener más información sobre el servidor MCP de la documentación de AWS, consulte AWS Documentation MCP Server.

Si aún no lo ha hecho, instale y configure la AWS CLI. Para obtener más información, consulte Instalación o actualización de la última versión de AWS CLI.

Cómo desarrollar el contenido de la política de aprobación

Luego de identificar el caso de uso y configurar el entorno, ya puede desarrollar el contenido de las políticas. Su caso de uso y sus preferencias determinarán en gran medida los tipos de políticas de aprobación e instrucciones que utilice.

Si no sabe cómo utilizar una política concreta o necesita más información sobre el esquema de una política, consulte Cómo crear políticas de aprobación para sus nodos y los temas siguientes. En estos temas, se explica en detalle cómo se evalúan las políticas y se proporcionan ejemplos prácticos de instrucciones.

En el siguiente procedimiento, se describe cómo crear políticas de aprobación con Amazon Q para la línea de comandos.

> **Note**
>
> Amazon Q Developer utiliza IA generativa. Es posible que necesite verificar las respuestas. Consulte Política de IA responsable de AWS.

Cómo crear políticas de aprobación mediante Amazon Q para la línea de comandos

1. Abra la terminal e inicie una sesión con Amazon Q.

```
$ q chat
```

2. Introduzca una petición y describa qué tipo de instrucción de política desee crear. Por ejemplo, a partir de la situación anterior para una política de aprobación, podría introducir la siguiente petición:

```
Create a Cedar policy statement that allows a principal to use the
 “getTokenForInstanceAccess" action for a resource
that has a tag key “Environment” and tag value of “Testing”. Use the following
 example statement for reference to verify the syntax.

permit (
    principal in AWS::IdentityStore::Group::"e8c17310-e011-7089-d989-10da1EXAMPLE",
    action == AWS::SSM::Action::"getTokenForInstanceAccess",
    resource
)
when {
    principal has organization && resource.hasTag("Engineering") &&
 resource.hasTag("Production") && principal.organization == "Platform"
};

The Cedar policy statement you create must be valid. Validate the Cedar statement
 using the AWS documentation provided for just-in-time node access policy statement
 structure and make sure you abide by the schema.
```

3. Al introducir esta petición de ejemplo, se obtuvo el siguiente resultado:

```
Based on the AWS documentation for Cedar policy statements and the examples
 provided, I'll create a valid Cedar policy
statement that allows a principal to use the "getTokenForInstanceAccess" action for
 resources with the tag key "Environment" and tag value "Testing".

Here's the Cedar policy statement:

permit (
    principal,
    action == AWS::SSM::Action::"getTokenForInstanceAccess",
    resource
)
```

```
when {
    resource.hasTag("Environment") &&
    resource.getTag("Environment") == "Testing"
};


This policy statement:
1. Uses the permit effect to allow access
2. Applies to any principal (since no specific principal is defined)
3. Specifically targets the "getTokenForInstanceAccess" action
4. Applies to resources that have a tag with key "Environment" and value "Testing"
5. Uses the proper syntax for the condition clause with the hasTag() and getTag()
 operators

The statement follows the Cedar policy language structure as documented in the AWS
 Systems Manager documentation for just-in-time node access policies.
```

4. Pegue la instrucción de la política en un nuevo archivo Cedar para validar la sintaxis. También puede usar el área de pruebas de Cedar para validar la instrucción.
5. Tras validar la instrucción, guarde el archivo Cedar.
6. Desde la terminal, ejecute el siguiente comando de AWS CLI para crear un documento SSM con el contenido de la política. Antes de utilizar las políticas de aprobación en un entorno de producción, pruébelas con un subconjunto de identidades y nodos en una Cuenta de AWS y Región de AWS.

> **Note**
>
> En el caso de una política de aprobación automática, el nombre del documento debe ser `SSM-JustInTimeAccessAutoApprovalPolicy`. Solo puede haber una política de aprobación automática por Cuenta de AWS y Región de AWS. En el caso de una política de denegación de acceso, el nombre del documento debe ser `SSM-JustInTimeAccessDenyAccessOrgPolicy`. Solo puede haber una política de denegación de acceso por organización de AWS Organizations, y la política debe crearse en la cuenta de administrador delegado de Systems Manager. Las restricciones de denominación de las políticas de aprobación manual son las mismas que las de otros documentos SSM. Para obtener más información, consulte CreateDocument.

Linux & macOS

```
aws ssm create-document \
    --content file://path/to/file/policyContent.cedar \
    --name "SSM-JustInTimeAccessAutoApprovalPolicy" \
    --document-type "AutoApproval"
```

Windows

```
aws ssm create-document ^
    --content file://C:\path\to\file\policyContent.cedar ^
    --name "SSM-JustInTimeAccessAutoApprovalPolicy" ^
    --document-type "AutoApproval"
```

PowerShell

```
$cedar = Get-Content -Path "C:\path\to\file\policyContent.cedar" | Out-String
New-SSMDocument `
    -Content $cedar `
    -Name "SSM-JustInTimeAccessAutoApprovalPolicy" `
    -DocumentType "AutoApproval"
```

Cómo actualizar las preferencias de sesión de acceso a los nodos justo a tiempo

Con el acceso a los nodos justo a tiempo, puede especificar las preferencias generales de sesión y registro en cada Cuenta de AWS y Región de AWS de su organización. Como alternativa, puede usar AWS CloudFormation StackSets para crear un documento de preferencias de la sesión en varias cuentas y regiones a fin de tener preferencias coherentes. Para obtener información sobre el esquema de los documentos de preferencias de la sesión, consulte Esquema del documento de Session.

Para fines de registro, recomendamos utilizar la opción de transmisión con Registros de Amazon CloudWatch. Esta característica permite enviar un flujo continuo de registros de datos de sesión a los Registros de CloudWatch. Los detalles esenciales, como los comandos que un usuario ha ejecutado en una sesión, el ID del usuario que ejecutó los comandos y las marcas de tiempo para el momento en que los datos de sesión se transmiten a los Registros de CloudWatch, se incluyen

cuando se efectúa el streaming de los datos de la sesión. Con el streaming de los datos de la sesión, los registros tienen formato JSON para ayudarlo a integrarse a sus soluciones de registro existentes.

Systems Manager no finaliza automáticamente las sesiones de acceso a los nodos justo a tiempo. Como práctica recomendada, especifique los valores para configurar la duración máxima de la sesión y el tiempo de espera de la sesión inactiva. Con estos valores, podrá evitar que un usuario permanezca conectado a un nodo durante más tiempo que el periodo aprobado en una solicitud de acceso. En el siguiente procedimiento, se describe cómo actualizar las preferencias de la sesión para el acceso a los nodos justo a tiempo.

> **Important**
>
> Debe etiquetar las claves de AWS KMS utilizadas para el cifrado de Session Manager y el registro RDP en el acceso justo a tiempo al nodo con la clave de etiqueta `SystemsManagerJustInTimeNodeAccessManaged` y el valor de la etiqueta `true`. Para obtener más información acerca del etiquetado de claves KMS, consulte Etiquetas en AWS KMS en la Guía para desarrolladores de AWS Key Management Service.

Cómo actualizar las preferencias de sesión

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. Seleccione Configuración en el panel de navegación.
3. Seleccione la pestaña Acceso a los nodos justo a tiempo.
4. En la sección Preferencias de la sesión, seleccione Editar.
5. Actualice las preferencias generales y de registro según sea necesario, y seleccione Guardar.

Cómo configurar las notificaciones para las solicitudes de acceso justo a tiempo

Puede configurar Systems Manager para que envíe notificaciones a las direcciones de correo electrónico o al cliente de chat para los aprobadores y el solicitante cuando un usuario cree una solicitud de acceso a los nodos justo a tiempo. La notificación contiene el motivo de la solicitud de acceso proporcionado por el solicitante, la Cuenta de AWS, la Región de AWS, el estado de la solicitud y el ID del nodo de destino. Actualmente, Systems Manager admite los clientes Slack y Microsoft Teams mediante la integración con Amazon Q Developer en aplicaciones de chat. Cuando

se utilizan notificaciones a través de clientes de chat, los aprobadores de las solicitudes de acceso pueden interactuar directamente con estas solicitudes. Esto elimina la necesidad de iniciar sesión en la consola para tomar medidas en relación con las solicitudes de acceso.

Antes de empezar

Antes de configurar un cliente de chat para las notificaciones de acceso a los nodos justo a tiempo, tenga en cuenta el siguiente requisito:

- Si utiliza roles de IAM para gestionar las identidades de los usuarios en la cuenta, debe asociar manualmente las direcciones de correo electrónico de los aprobadores o solicitantes que posean el rol asociado y a quienes desee enviar notificaciones. De lo contrario, no se podrá notificar por correo electrónico a los destinatarios previstos.

En los siguientes procedimientos, se describe cómo configurar las notificaciones para las solicitudes de acceso a los nodos justo a tiempo.

Cómo configurar un cliente de chat para las notificaciones de acceso a los nodos justo a tiempo

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. Seleccione Configuración en el panel de navegación.
3. Seleccione la pestaña Acceso a los nodos justo a tiempo.
4. En la sección Chat, seleccione Configurar nuevo cliente.
5. En el menú desplegable Seleccionar tipo de cliente, elija el tipo de cliente de chat que desee configurar y seleccione Siguiente.
6. Se le solicitará que permita que Amazon Q Developer en aplicaciones de chat acceda al cliente de chat que utiliza. Seleccione Permitir.
7. En la sección Configurar canal, introduzca la información del canal del cliente de chat y seleccione los tipos de notificaciones que desee recibir.
8. Si está configurando las notificaciones de Slack, invite a "@Amazon Q" a todos los canales de Slack en los que se estén configurando las notificaciones.
9. Seleccione Configurar canal.

> **Note**
>
> Para poder aprobar o rechazar las solicitudes de acceso directamente desde un canal de Slack, asegúrese de que el rol de IAM configurado con el canal de Slack tenga permisos `ssm:SendAutomationSignal` y una política de confianza que incluya el chatbot:

```
{
            "Effect": "Allow",
            "Principal": {
                "Service": "chatbot.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
}
```

Cómo configurar las notificaciones por correo electrónico para las notificaciones de acceso a los nodos justo a tiempo

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. Seleccione Configuración en el panel de navegación.
3. Seleccione la pestaña Acceso a los nodos justo a tiempo.
4. En la sección Correo electrónico, seleccione Editar.
5. Seleccione Añadir correos electrónicos, elija el rol de IAM al que desee asociar manualmente las direcciones de correo electrónico.
6. Ingrese su dirección de correo electrónico en el campo Dirección de correo electrónico. Siempre que se cree una solicitud de acceso que requiera la aprobación del rol de IAM que especificó, se notificará a las direcciones de correo electrónico asociadas a ese rol.
7. Seleccione Añadir dirección de correo electrónico.

Cómo grabar conexiones RDP

El acceso a los nodos justo a tiempo incluye la posibilidad de grabar las conexiones con el protocolo para escritorios remotos (RDP) realizadas con los nodos de Windows Server. Para grabar las conexiones RDP, se necesita un bucket de S3 y una clave de AWS Key Management Service (AWS KMS) administrada por el cliente. La AWS KMS key se usa para cifrar temporalmente los datos

de la grabación mientras se generan y almacenan en los recursos de Systems Manager. La clave administrada por el cliente debe ser una clave simétrica con un uso de cifrado y descifrado. Puede usar una clave multirregional para la organización o debe crear una clave administrada por el cliente en cada región en la que haya habilitado el acceso a los nodos justo a tiempo.

Si habilitó el cifrado de KMS en el bucket de S3 en el que almacena las grabaciones, debe dar acceso a la clave administrada por el cliente. Esta clave se utiliza para el cifrado del bucket en la entidad principal de servicio `ssm-guiconnect`. Esta clave administrada por el cliente puede ser diferente a la que especifique en la configuración de grabación, y debe incluir para que se requiere el permiso `kms:CreateGrant` para establecer conexiones.

Configuración del cifrado de bucket de S3 para las grabaciones RDP

Las grabaciones de su conexión se almacenan en el bucket de S3 que especifique al habilitar la grabación RDP.

Si utiliza una clave de KMS como mecanismo de cifrado predeterminado para el bucket de S3 (SSE-KMS), debe permitir que la entidad principal de servicio `ssm-guiconnect` acceda a la acción `kms:GenerateDataKey` en la clave. Recomendamos usar una clave administrada por el cliente al usar el cifrado de SSE-KMS con un bucket de S3. Esto se debe a que puede actualizar la política de claves asociada para una clave administrada por el cliente. No puede actualizar las políticas de claves para Claves administradas por AWS.

> **Important**
>
> Debe etiquetar las claves de AWS KMS utilizadas para el cifrado de Session Manager y el registro de RDP en el acceso al nodo justo a tiempo con la clave de etiqueta `SystemsManagerJustInTimeNodeAccessManaged` y el valor de etiqueta `true`. Para obtener más información acerca del etiquetado de claves KMS, consulte Etiquetas en AWS KMS en la Guía para desarrolladores de AWS Key Management Service.

Utilice la siguiente política de claves administrada por el cliente para permitir que el servicio `ssm-guiconnect` acceda a la clave KMS para el almacenamiento en S3. Para obtener información sobre la actualización de una clave administrada por el cliente, consulte Cambio de una política de claves en la Guía para desarrolladores de AWS Key Management Service.

Reemplace cada *`example resource placeholder`* con su propia información:

- *account-id* representa el ID de Cuenta de AWS que inicia la conexión.
- *region* representa la Región de AWS donde está ubicado el bucket de S3. (Puede usar * si el bucket recibirá grabaciones de varias regiones. Ejemplo: `s3.*.amazonaws.com`.)

> **Note**
>
> Puede usar `aws:SourceOrgID` en la política en lugar de `aws:SourceAccount` si la cuenta pertenece a una organización en AWS Organizations.

```
{
    "Sid": "Allow the GUI Connect service principal to access S3",
    "Effect": "Allow",
    "Principal": {
        "Service": "ssm-guiconnect.amazonaws.com"
    },
    "Action": [
        "kms:GenerateDataKey*"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": "account-id"
        },
        "StringLike": {
            "kms:ViaService": "s3.region.amazonaws.com"
        }
    }
}
```

Cómo configurar permisos de IAM para grabar conexiones RDP

Además de los permisos de IAM necesarios del acceso a los nodos justo a tiempo, el usuario o el rol que utilice deben tener los siguientes permisos en función de la tarea que deba realizar.

Permisos para configurar la grabación de conexiones

Para configurar la grabación de conexiones RDP, se necesitan los siguientes permisos:

- `ssm-guiconnect:UpdateConnectionRecordingPreferences`

- `ssm-guiconnect:GetConnectionRecordingPreferences`
- `ssm-guiconnect:DeleteConnectionRecordingPreferences`
- `kms:CreateGrant`

Permisos para iniciar conexiones

Para establecer conexiones RDP con acceso a los nodos justo a tiempo, se necesitan los siguientes permisos:

- `ssm-guiconnect:CancelConnection`
- `ssm-guiconnect:GetConnection`
- `ssm-guiconnect:StartConnection`
- `kms:CreateGrant`

Antes de empezar

Para almacenar las grabaciones de las conexiones, primero debe crear un bucket de S3 y añadir la siguiente política de bucket. Reemplace cada *`example resource placeholder`* con su propia información.

Para obtener información sobre cómo agregar una política de bucket, consulte Cómo agregar una política de bucket mediante la consola de Amazon S3 en la Guía del usuario de Amazon Simple Storage Service.

JSON

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

```
            "Action": "s3:PutObject",
            "Resource": [
                "arn:aws:s3:::",
                "arn:aws:s3:::/*"
            ],
            "Condition":{
            "StringEquals":{
                "aws:SourceAccount":"111122223333"
                }
            }
        }
    ]
}
```

Habilitación y configuración de la grabación de conexiones RDP

En el siguiente procedimiento, se describe cómo habilitar y configurar la grabación de conexiones RDP.

Para habilitar y configurar la grabación de conexiones RDP

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. Seleccione Configuración en el panel de navegación.
3. Seleccione la pestaña Acceso a los nodos justo a tiempo.
4. En la sección Grabación RDP, seleccione Habilitar grabación RDP.
5. Elija el bucket de S3 en el que desee cargar las grabaciones de las sesiones.
6. Elija la clave administrada por el cliente que desee utilizar para cifrar temporalmente los datos de la grabación mientras se generan y almacenan en los recursos de Systems Manager. Puede ser una clave administrada por el cliente diferente a la que utiliza para cifrar el bucket.
7. Seleccione Guardar.

Valores de estado de la grabación de conexiones RDP

Los valores de estado válidos para las grabaciones de conexiones RDP son los siguientes:

- `Recording`: la conexión está en proceso de grabación.
- `Processing`: el video se está procesando una vez finalizada la conexión.

- `Finished`: estado terminal exitoso; el video de grabación de la conexión se procesó correctamente y se cargó en el bucket especificado.
- `Failed`: estado terminal fallido. La conexión no se grabó correctamente.
- `ProcessingError`: se produjeron uno o más errores o fallos intermedios durante el procesamiento del vídeo. Las posibles causas incluyen fallos de dependencia del servicio o falta de permisos debido a una configuración incorrecta en el bucket de S3 especificado para almacenar las grabaciones. El servicio sigue intentando procesar cuando la grabación se encuentra en este estado.

> **Note**
>
> `ProcessingError` puede deberse a que la entidad principal de servicio `ssm-guiconnect` no tenga permiso para cargar objetos en el bucket de S3 una vez establecida la conexión. Otra posible causa es la falta de permisos de KMS en la clave de KMS que se utiliza para el cifrado del bucket de S3.

Cómo modificar destinos

Cuando configura el acceso a los nodos justo a tiempo, debe elegir los destinos en los que desee hacerlo. Los destinos consisten en unidades organizativas (OU) de AWS Organizations y Regiones de AWS. De forma predeterminada, los mismos destinos que eligió al configurar la consola unificada de Systems Manager se seleccionan para el acceso a los nodos justo a tiempo. Puede elegir configurar el acceso a los nodos justo a tiempo para todos los mismos destinos o para un subconjunto de los destinos que especificó al configurar la consola unificada de Systems Manager. No es posible añadir nuevos destinos que no se seleccionaron al configurar la consola unificada de Systems Manager. Puede cambiar los destinos que seleccionó después de configurar el acceso a los nodos justo a tiempo.

En el siguiente procedimiento, se describe cómo modificar los destinos para el acceso a los nodos justo a tiempo.

Cómo modificar destinos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. Seleccione Configuración en el panel de navegación.

3. Seleccione la pestaña Acceso a los nodos justo a tiempo.
4. En la sección Objetivos, seleccione Editar.
5. Seleccione las Unidades organizativas y las Regiones en las que desee usar el acceso a los nodos justo a tiempo.
6. Seleccione Guardar.

Cómo cambiar proveedores de identidades

De forma predeterminada, el acceso a los nodos justo a tiempo utiliza IAM como proveedor de identidades. Después de habilitar el acceso a los nodos justo a tiempo, los clientes que utilicen la consola unificada con una organización pueden modificar esta configuración para usar IAM Identity Center. El acceso a los nodos justo a tiempo no admite IAM Identity Center como proveedor de identidades cuando se configura para una sola cuenta y región.

En el siguiente procedimiento, se describe cómo modificar el proveedor de identidades para el acceso a los nodos justo a tiempo.

Cómo modificar proveedores de identidades

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. Seleccione Configuración en el panel de navegación.
3. Seleccione la pestaña Acceso a los nodos justo a tiempo.
4. En la sección Identidad del usuario, seleccione Editar.
5. Elija AWS IAM Identity Center.
6. Seleccione Guardar.

Cómo iniciar una sesión de acceso a los nodos justo a tiempo

Tras habilitar y configurar el acceso a los nodos justo a tiempo y configurar las preferencias de sesión y notificación, los usuarios pueden iniciar las sesiones de acceso a los nodos justo a tiempo. Puede iniciar las sesiones mediante el acceso a los nodos justo a tiempo desde la consola de Systems Manager o desde la AWS Command Line Interface mediante el complemento Session Manager. Las sesiones de acceso a los nodos justo a tiempo se pueden iniciar en los nodos de la misma cuenta y región. En los siguientes procedimientos, se describe cómo iniciar sesiones con acceso a los nodos justo a tiempo.

> **Note**
>
> Si los usuarios utilizaban anteriormente Session Manager para conectarse a los nodos, debe eliminar los permisos de Session Manager; por ejemplo, `ssm:StartSession` de las políticas de IAM para iniciar sesiones mediante el acceso a los nodos justo a tiempo. De lo contrario, cuando se conecten a los nodos, seguirán utilizando Session Manager.

Cómo iniciar una sesión con acceso a los nodos justo a tiempo mediante la consola

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Explorar nodos.
3. Seleccione el nodo al que desee conectarse.
4. En el menú desplegable Acciones, seleccione Conectar.

Si las políticas de aprobación de su organización no le permiten conectarse automáticamente al nodo, deberá enviar una solicitud de acceso. Después de completar la información necesaria y enviar la solicitud de acceso, podrá iniciar sesiones en el nodo una vez que se hayan recibido todas las aprobaciones necesarias.

Cómo iniciar una sesión con acceso a los nodos justo a tiempo mediante la AWS CLI

1. Ejecute el siguiente comando para iniciar el flujo de trabajo de la solicitud de acceso y asegúrese de sustituir los *valores del marcador de posición* con sus datos.

   ```
   aws ssm start-access-request \
       --targets  Key=InstanceIds,Values=i-02573cafcfEXAMPLE
       --reason "Troubleshooting networking performance issue"
   ```

   Según las políticas de aprobación de su organización, se conectará automáticamente al nodo o se iniciará el proceso de aprobación manual. En el caso de las solicitudes que requieren aprobaciones manuales, anote el ID de la solicitud de acceso que aparece en la respuesta.
2. Espere a que se realicen todas las aprobaciones necesarias.
3. Una vez que se hayan realicen todas las aprobaciones necesarias, ejecute el siguiente comando para obtener un token de acceso que contenga las credenciales temporales. Reemplace los *valores de marcador de posición* con su información.

```
aws ssm get-access-token \
    --access-request-id oi-12345abcdef
```

Anote el token de acceso devuelto en la respuesta.

4. Ejecute el siguiente comando para utilizar la credencial temporal en la AWS CLI y asegúrese de sustituir los *valores del marcador de posición* con sus datos.

```
export AWS_SESSION_TOKEN=AQoDYXdzEJr...<remainder of session token>
```

5. Ejecute el siguiente comando para iniciar una sesión en el nodo y asegúrese de sustituir los *valores del marcador de posición* con sus datos.

```
aws ssm start-session \
    --target i-02573cafcfEXAMPLE
```

Cómo administrar las solicitudes de acceso justo a tiempo

Para aumentar la visibilidad en toda la organización, Systems Manager replica las solicitudes de acceso en la cuenta de administrador delegado de la organización. Para que pueda respetar los requisitos de cumplimiento, Systems Manager conserva todas las solicitudes de acceso durante 1 año. En los siguientes temas, se describe cómo administrar las solicitudes de acceso a los nodos justo a tiempo. Esta información está destinada a los aprobadores de solicitudes de acceso. Antes de empezar, le recomendamos revisar las políticas de IAM y asegurarse de que dispone de los permisos necesarios para administrar el acceso a los nodos justo a tiempo. Para obtener más información, consulte Cómo configurar acceso justo a tiempo con Systems Manager.

Temas

- Cómo aprobar y denegar solicitudes de acceso a los nodos justo a tiempo

Cómo aprobar y denegar solicitudes de acceso a los nodos justo a tiempo

Los aprobadores de solicitudes de acceso pueden aprobar o denegar las solicitudes de acceso a los nodos justo a tiempo desde la consola unificada de Systems Manager o mediante la herramienta de línea de comandos que prefieran. Esta información está destinada a los aprobadores de solicitudes de acceso. Si no dispone de los permisos necesarios para aprobar o rechazar solicitudes de acceso,

póngase en contacto con su administrador. En los siguientes procedimientos, se describe cómo aprobar o denegar las solicitudes de acceso a los nodos justo a tiempo.

Cómo aprobar y denegar solicitudes de acceso a los nodos justo a tiempo mediante la consola

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, seleccione Administrar acceso a nodos.
3. Seleccione la pestaña Solicitudes de acceso.
4. Seleccione el botón Solicitudes para mí.
5. Seleccione la casilla de verificación situada junto a la solicitud de acceso que desee aprobar o denegar.
6. Seleccione Aprobar o Rechazar.

Tras aprobar una solicitud de acceso, puede revocar su aprobación en cualquier momento si selecciona Revocar.

Cómo aprobar y denegar solicitudes de acceso a los nodos justo a tiempo mediante la línea de comandos

1. Anote el ID de la solicitud de acceso que aparece en la notificación. Por ejemplo, *oi-12345abcdef*.
2. Ejecute el siguiente comando para obtener los detalles del flujo de trabajo de la aprobación de la solicitud de acceso y asegúrese de sustituir los *valores del marcador de posición* con sus datos.

   ```
   aws ssm get-ops-item \
       --ops-item-id oi-12345abcdef
   ```

   Anote el valor de `automationExecutionId` en el campo `/aws/accessrequest` correspondiente a `OperationalData`. Por ejemplo, *9231944f-61c6-40be-8bce-8ee2bEXAMPLE*.
3. Ejecute el siguiente comando para aprobar o denegar la solicitud de acceso. Utilice el tipo de señal `Approve` para aprobar la solicitud y `Deny` para denegarla. Asegúrese de reemplazar los *valores de marcador de posición* con sus datos.

```
aws ssm send-automation-signal \
    --automation-execution-id 9231944f-61c6-40be-8bce-8ee2bEXAMPLE \
    --signal-type "Approve"
```

Cambio al acceso a los nodos justo a tiempo desde Session Manager

Al habilitar el acceso a los nodos justo a tiempo, Systems Manager no realiza ningún cambio en los recursos existentes para Session Manager. Esto garantiza que no haya interrupciones en su entorno existente y que los usuarios puedan seguir iniciando sesión mientras usted crea y valida las políticas de aprobación. Una vez que desee probar las políticas de aprobación, debe modificar las políticas de IAM existentes para completar la transición al acceso a los nodos justo a tiempo. Esto implica añadir a las identidades los permisos necesarios para el acceso a los nodos justo a tiempo y eliminar el permiso para el funcionamiento de la API de `StartSession` para Session Manager. Recomendamos probar las políticas de aprobación con un subconjunto de identidades y nodos en una Cuenta de AWS y Región de AWS.

Para obtener más información acerca de los permisos necesarios para el acceso a los nodos justo a tiempo, consulte Cómo configurar acceso justo a tiempo con Systems Manager.

Para obtener más información sobre cómo modificar y asignar los permisos a una entidad principal de IAM, consulte Adición y eliminación de permisos de identidad de IAM en la Guía del usuario de IAM.

A continuación, se detalla un método para pasar al acceso a los nodos justo a tiempo desde Session Manager.

Pasar de Session Manager al acceso a los nodos justo a tiempo requiere de una planificación y pruebas cuidadosas para garantizar una transición fluida sin interrumpir sus operaciones. En las siguientes secciones, se describe cómo puede completar este proceso.

Requisitos previos

Antes de comenzar, asegúrese de haber realizado las siguientes tareas:

- Configuró la consola unificada de Systems Manager.
- Verificó que tiene permisos para modificar las políticas de IAM en la cuenta.
- Identificó todas las políticas y roles de IAM que actualmente otorgan permisos de Session Manager.

- Documentó la configuración actual de Session Manager, incluidas las preferencias de sesión y los ajustes de registro.

Evaluación

Evalúe el entorno actual y realice las siguientes acciones para describir los comportamientos de aprobación deseados:

1. Haga un inventario de los nodos: identifique todos los nodos a los que los usuarios acceden actualmente a través de Session Manager.
2. Identifique los patrones de acceso de los usuarios: documente qué usuarios o roles necesitan acceder a qué nodos y en qué circunstancias.
3. Trace los flujos de trabajo de aprobación: determine quién debe aprobar las solicitudes de acceso para los distintos tipos de nodos.
4. Revise la estrategia de etiquetado: asegúrese de que los nodos estén correctamente etiquetados para respaldar las políticas de aprobación planificadas.
5. Audite las políticas de IAM existentes: identifique todas las políticas que incluyen permisos de Session Manager.

Planificación

Estrategia por fases

Al pasar de Session Manager al acceso a los nodos justo a tiempo, recomendamos utilizar un enfoque por fases, como el siguiente:

1. Fase 1, ajustes y configuración: habilite el acceso a los nodos justo a tiempo sin modificar los permisos de Session Manager existentes.
2. Fase 2, desarrollo de políticas: cree y pruebe políticas de aprobación para los nodos.
3. Fase 3, migración piloto: modifique un pequeño grupo de nodos no críticos, usuarios o roles de Session Manager para que admitan el acceso a los nodos justo a tiempo.
4. Fase 4, migración completa: migre gradualmente todos los nodos, usuarios o roles restantes.

Consideraciones sobre el cronograma

Considere los siguientes factores al crear su cronograma para pasar de Session Manager al acceso a los nodos justo a tiempo:

- Dedique tiempo a la capacitación de los usuarios y a su adaptación al nuevo flujo de trabajo de aprobación.
- Programe las migraciones durante los periodos de menor actividad operativa.
- Incluya tiempo adicional para la resolución de problemas y los ajustes.
- Planifique un periodo de funcionamiento paralelo en el que ambos sistemas estén disponibles.

Pasos para la implementación

Fase 1: ajustes y configuración

1. Habilite el acceso a los nodos justo a tiempo en la consola de Systems Manager. Para ver los pasos detallados, consulte Cómo configurar acceso justo a tiempo con Systems Manager.
2. Configure las preferencias de sesión para el acceso a los nodos justo a tiempo a fin de que coincidan con los ajustes actuales de Session Manager. Para obtener más información, consulte Cómo actualizar las preferencias de sesión de acceso a los nodos justo a tiempo.
3. Configure las preferencias de notificación para las solicitudes de acceso. Para obtener más información, consulte Cómo configurar las notificaciones para las solicitudes de acceso justo a tiempo.
4. Si utiliza conexiones con el protocolo para escritorios remotos (RDP) a nodos de Windows Server, configure la grabación RDP. Para obtener más información, consulte Cómo grabar conexiones RDP.

Fase 2: desarrollo de políticas

1. Cree políticas de IAM para los administradores del acceso a los nodos justo a tiempo y los usuarios.
2. Desarrolle políticas de aprobación basadas en sus requisitos de seguridad y caso de uso.
3. Pruebe las políticas en un entorno que no sea de producción para asegurarse de que funcionen como se espera.

Fase 3: migración piloto

1. Seleccione un pequeño grupo de usuarios y nodos no críticos para la prueba piloto.
2. Cree nuevas políticas de IAM para los usuarios piloto que incluyan permisos de acceso a los nodos justo a tiempo.

3. Elimine los permisos de Session Manager (`ssm:StartSession`) de las políticas de IAM de los usuarios piloto.
4. Capacite a los usuarios piloto sobre el nuevo flujo de trabajo de solicitudes de acceso.
5. Supervise el piloto para detectar problemas y recopile comentarios.
6. Ajuste las políticas y los procedimientos en función de los resultados del piloto.

Ejemplo de modificación de la política de IAM para los usuarios piloto

Política original con permisos de Session Manager:

JSON

```
{
                    "Version": "2012-10-17",
                    "Statement": [
                        {
                            "Effect": "Allow",
                            "Action": [
                                "ssm:StartSession",
                                "ssm:ResumeSession",
                                "ssm:TerminateSession"
                            ],
                            "Resource": "*"
                        }
                    ]
                }
```

Política modificada para el acceso a los nodos justo a tiempo:

JSON

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

```
                "ssm:GetAccessToken",
                "ssm:ResumeSession",
                "ssm:TerminateSession"
            ],
            "Resource": "*"
        }
    ]
}
```

Fase 4: migración completa

Desarrolle un cronograma para migrar los usuarios y nodos restantes en lotes.

Metodología de pruebas

Durante todo el proceso de migración, realice las siguientes pruebas:

- Validación de las políticas: verifique que las políticas de aprobación se apliquen correctamente a los nodos y usuarios previstos.
- Flujo de trabajo de la solicitud de acceso: pruebe el flujo de trabajo completo, desde la solicitud de acceso hasta el establecimiento de la sesión, tanto para situaciones de aprobación automática como de aprobación manual.
- Notificaciones: verifique que los aprobadores reciban las notificaciones a través de los canales configurados (correo electrónico, Slack, Microsoft Teams).
- Registro y supervisión: verifique que los registros de las sesiones y las solicitudes de acceso se capturan y almacenan correctamente.

Prácticas recomendadas para una migración exitosa

- Comunicación a tiempo y frecuente: informe a los usuarios sobre el cronograma de migración y las ventajas del acceso a los nodos justo a tiempo.
- Comience con sistemas no críticos: inicie la migración con entornos de desarrollo o prueba antes de pasar a la producción.
- Documente todo: mantenga registros detallados de las políticas de aprobación, los cambios en las políticas de IAM y los ajustes en la configuración.
- Supervise y ajuste: supervise continuamente las solicitudes de acceso y los flujos de trabajo de aprobación, y ajuste las políticas según sea necesario.

- Establezca la gobernanza: cree un proceso para revisar y actualizar periódicamente las políticas de aprobación a medida que cambie el entorno.

Cómo deshabilitar el acceso justo a tiempo con Systems Manager

En el siguiente procedimiento, se describe cómo deshabilitar el acceso a los nodos justo a tiempo. Después de deshabilitar el acceso a los nodos justo a tiempo, es posible que los usuarios de la organización no puedan conectarse a los nodos a menos que ya haya implementado otros métodos de conexión.

Cómo deshabilitar el acceso a los nodos justo a tiempo

1. Inicie sesión en la cuenta de administrador delegado de Systems Manager de la organización.
2. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
3. Seleccione Configuración en el panel de navegación.
4. En la pestaña Acceso a los nodos justo a tiempo, seleccione Deshabilitar.

Preguntas frecuentes sobre el acceso a los nodos justo a tiempo

¿Cómo puedo pasar de Session Manager al acceso a los nodos justo a tiempo?

Tras configurar la consola unificada y habilitar el acceso a los nodos justo a tiempo, debe modificar las políticas de IAM existentes para completar la transición al acceso a los nodos justo a tiempo. Esto incluye añadir los permisos necesarios del acceso a los nodos justo a tiempo y eliminar el permiso del funcionamiento de la API de `StartSession` para Session Manager. Para obtener más información acerca de las políticas de IAM sobre el acceso a los nodos justo a tiempo, consulte Cómo configurar acceso justo a tiempo con Systems Manager.

¿Tengo que configurar la consola unificada para utilizar el acceso a los nodos justo a tiempo?

Sí, la configuración de la consola unificada es un requisito previo del acceso a los nodos justo a tiempo. Sin embargo, después de configurar la consola unificada y habilitar el acceso a los nodos justo a tiempo, existen varios métodos para conectarse a ellos. Por ejemplo, puede iniciar sesiones de acceso a los nodos justo a tiempo desde la consola de Amazon EC2 y la AWS CLI. Para obtener

más información sobre la configuración de la consola unificada, consulte Configuración de la consola unificada de Systems Manager para una organización.

¿El acceso a los nodos justo a tiempo conlleva gastos?

Systems Manager ofrece una prueba gratuita de 30 días para el acceso a los nodos justo a tiempo. Una vez finalizada la prueba, el acceso a los nodos justo a tiempo conlleva gastos. Para más información, consulte Precios de AWS Systems Manager.

¿Cuál es la prioridad de las políticas de aprobación sobre el acceso a los nodos justo a tiempo?

Las políticas de aprobación se evalúan en el siguiente orden:

1. Denegación de acceso
2. Aprobación automática
3. Manual

¿Cómo se evalúan las políticas de aprobación manual?

El acceso a los nodos justo a tiempo siempre favorece la política más específica para un nodo. Las políticas de aprobación manual se evalúan en el siguiente orden:

1. Destino específico según la etiqueta
2. Todos los destinos de los nodos

¿Qué ocurre si no hay una política de aprobación que se aplique a un nodo?

Para conectarse a un nodo mediante el acceso justo a tiempo, se le debe aplicar una política de aprobación. Si no hay políticas de aprobación que se apliquen a un nodo, los usuarios no podrán solicitar el acceso.

¿Varias políticas de aprobación pueden estar dirigidas a una etiqueta?

Una etiqueta solo puede ser el destino una vez en las políticas de aprobación.

¿Qué ocurre si se aplican varias políticas de aprobación manual a un nodo debido a la superposición de etiquetas?

Cuando se aplican varias políticas de aprobación manual a un nodo, se genera un conflicto, y los usuarios no pueden solicitar acceso. Tenga esto en cuenta al crear las políticas de aprobación manual, ya que algunas instancias pueden tener varias etiquetas según su caso.

¿Puedo usar el acceso a los nodos justo a tiempo para solicitar acceso e iniciar sesiones en nodos de todas las cuentas y regiones?

El acceso a los nodos justo a tiempo permite solicitar acceso e iniciar sesiones en nodos de la misma cuenta y región que el solicitante.

¿Puedo usar el acceso a los nodos justo a tiempo para solicitar acceso e iniciar sesiones en los nodos registrados con una activación híbrida?

Sí, el acceso a los nodos justo a tiempo permite solicitar acceso e iniciar sesiones en los nodos registrados con una activación híbrida. El nodo debe estar registrado en la misma cuenta y región que el solicitante.

Diagnóstico y corrección

Gracias a la consola unificada de Systems Manager, puede identificar problemas en la flota en una sola operación de diagnóstico. En el caso de las organizaciones, puede intentar corregir todos los destinos o solo algunos de ellos mediante una única operación de Automatización. En el caso de una organización, como administrador de cuentas delegado, puede seleccionar destinos en todas las cuentas y regiones. Si trabaja en una sola cuenta, puede seleccionar destinos en una sola región a la vez.

Systems Manager puede diagnosticar y ayudar a corregir varios tipos de errores de implementación, así como configuraciones desviadas. Systems Manager también puede identificar las instancias de Amazon Elastic Compute Cloud (Amazon EC2) en su cuenta u organización que no puede tratar como nodos administrados. El proceso de diagnóstico de instancias de EC2 puede identificar problemas relacionados con configuraciones incorrectas en una nube privada virtual (VPC), en una configuración de sistema de nombres de dominio (DNS) o en un grupo de seguridad de Amazon Elastic Compute Cloud (Amazon EC2).

> **Note**
>
> Systems Manager admite instancias de EC2 y otros tipos de equipos en un entorno híbrido y multinube como nodos administrados. Para ser un nodo administrado, AWS Systems Manager Agent (SSM Agent) debe estar instalado en un equipo y Systems Manager debe tener permiso para realizar acciones en el equipo.
> En el caso de las instancias de EC2, este permiso se puede proporcionar por cuenta mediante un rol de AWS Identity and Access Management (IAM) o por instancia mediante un perfil de instancia. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager.
> En el caso de los equipos que no son de EC2, este permiso se proporciona mediante un rol de servicio de IAM. Para obtener más información, consulte Creación del rol de servicio de IAM requerido para Systems Manager en entornos híbridos y multinube.

Antes de empezar

Para utilizar la característica Diagnosticar y corregir para detectar instancias de EC2 no administradas, antes debe incorporar su organización o cuenta a la consola unificada de Systems Manager. Durante este proceso, debe elegir la opción de crear los roles de IAM y las políticas administradas necesarias para estas operaciones. Para obtener más información, consulte Configuración de la consola unificada de Systems Manager para una organización.

Utilice los siguientes temas para identificar y corregir algunos tipos comunes de implementaciones con errores, configuraciones desviadas e instancias de EC2 no administradas.

Temas

- Diagnóstico y corrección de implementaciones con error
- Diagnóstico y corrección de configuraciones desviadas
- Diagnóstico y corrección de instancias de Amazon EC2 no administradas en Systems Manager
- Tipos de impacto de la corrección de las acciones del manual de procedimientos
- Visualización del progreso de la ejecución y el historial de las correcciones en Systems Manager

Diagnóstico y corrección de implementaciones con error

Systems Manager puede diagnosticar y, después, ayudar a corregir los siguientes tipos de implementaciones con error:

- Configuración básica para las cuentas de los miembros de la organización
- Configuración básica para la cuenta de administrador delegado
- Configuración básica de su cuenta

Utilice el siguiente procedimiento para intentar corregir estos tipos de problemas.

Diagnóstico y corrección de las implementaciones con error

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Diagnosticar y corregir.
3. Elija la pestaña Problemas de implementaciones.
4. En la sección Implementaciones con error, revise la lista de resultados de las implementaciones con error.
5. En la columna Paso de configuración, elija el nombre de un resultado para revisar los detalles adicionales acerca del problema. Por ejemplo: Configuración básica para las cuentas de los miembros de la organización.
6. En la página de detalles de esa implementación con errores, se puede ver una lista de cuentas y cuántas regiones de cada una de ellas han sufrido errores en la implementación.
7. Seleccione un ID de cuenta para ver información sobre el motivo de los errores de esa cuenta.
8. En el área Regiones con errores, examine la información proporcionada en Motivo del estado. Esta información puede indicar el motivo de la implementación con errores, lo que podría proporcionar información sobre los cambios de configuración que deben hacerse.
9. Si desea volver a intentar la implementación sin hacer cambios en la configuración, elija Volver a implementar.

Diagnóstico y corrección de configuraciones desviadas

Systems Manager puede diagnosticar y, después, ayudar a corregir los siguientes tipos de configuraciones desviadas:

- Configuración básica para las cuentas de los miembros de la organización
- Configuración básica para la cuenta de administrador delegado
- Configuración básica de su cuenta

Utilice el siguiente procedimiento para intentar corregir este tipo de configuraciones desviadas.

Diagnóstico y corrección de las configuraciones desviadas

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Diagnosticar y corregir.
3. Elija la pestaña Problemas de implementaciones.
4. En la sección Implementaciones desviadas, consulte la lista de resultados de implementaciones con errores.

   -o bien-

   Para ejecutar un nuevo diagnóstico, elija Detectar desviaciones.
5. En la columna Paso de configuración, elija el nombre de un resultado para revisar los detalles adicionales acerca del problema. Por ejemplo: Configuración básica para las cuentas de los miembros de la organización.
6. En la página de detalles de esa implementación con errores, se puede ver una lista de cuentas y cuántas regiones de cada una de ellas han sufrido cambios de configuración.
7. Seleccione un ID de cuenta para ver la información sobre el motivo de los desvíos de configuración de esa cuenta.
8. En el área Recursos desviados, la columna Recurso muestra los nombres de los recursos que se han visto afectados. La columna Tipo de desviación indica si el recurso se modificó o se eliminó.
9. Para volver a implementar la configuración deseada, elija Volver a implementar.

Diagnóstico y corrección de instancias de Amazon EC2 no administradas en Systems Manager

Para ayudar a administrar las instancias de Amazon Elastic Compute Cloud (Amazon EC2) con Systems Manager, puede utilizar la consola de Systems Manager para hacer lo siguiente:

1. Ejecute un proceso de diagnóstico manual o programado para identificar las instancias de EC2 de su cuenta u organización que actualmente no administra Systems Manager.
2. Identifique los problemas de red o de otro tipo que impiden que Systems Manager administre las instancias.

3. Lleve a cabo una ejecución de Automatización para solucionar el problema de manera automática o acceda a la información para solucionar el problema de forma manual.

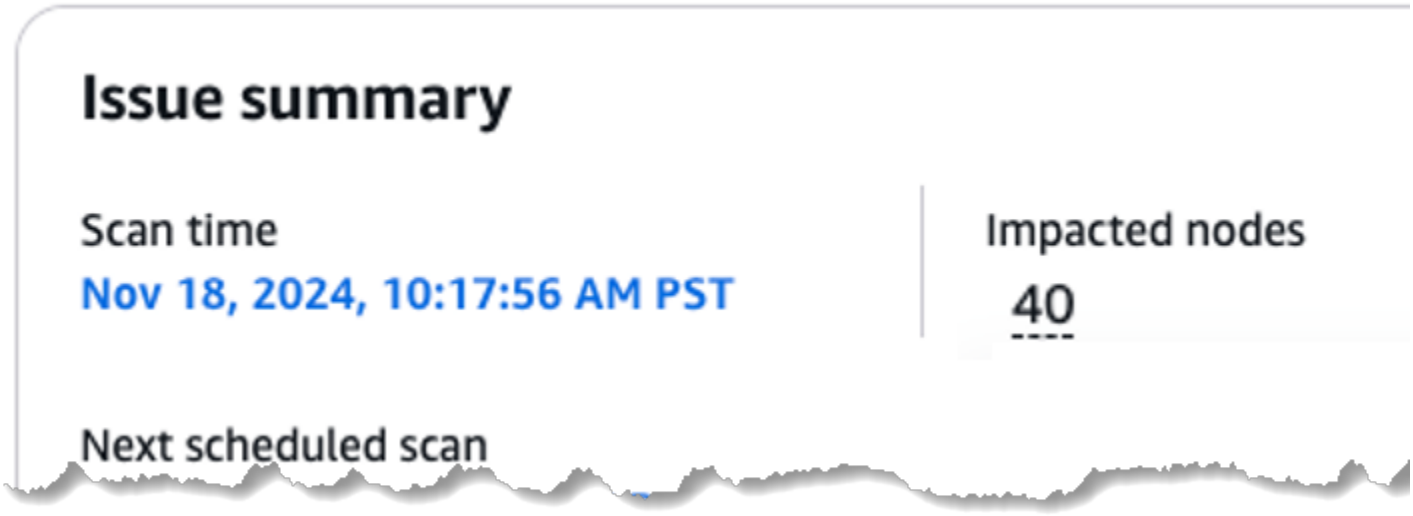
Utilice la información de los temas siguientes para diagnosticar y corregir los problemas que impiden que Systems Manager administre las instancias de EC2.

Cómo contabiliza Systems Manager los nodos afectados en la lista “Problemas de instancias de EC2 no administradas”

El número de nodos declarados como no administrados en la pestaña Problemas de instancias de EC2 no administradas representa el número total de instancias con cualquiera de los siguientes valores de estado en el momento del análisis de diagnóstico:

- `Running`
- `Stopped`
- `Stopping`

Este número se indica como Nodos afectados en el área Resumen del problema. En la siguiente imagen, este número de nodos afectados que Systems Manager no administra actualmente es `40`.



A diferencia del informe sobre instancias de EC2 no administradas que se muestra en la página Revisión de información de nodos, este recuento de instancias de EC2 no es dinámico. Representa los resultados obtenidos durante el último análisis de diagnóstico notificado y se muestra como el valor del Tiempo de análisis. Por lo tanto, recomendamos ejecutar un análisis de diagnóstico para las

instancias de EC2 no administradas de forma periódica para mantener actualizada dicha cantidad notificada de nodos afectados.

Para obtener información sobre la cantidad de instancias no administradas en la página Revisar información sobre los nodos, consulte ¿Qué es una instancia no administrada? en el tema Revisión de la información sobre los nodos.

Temas

- Categorías de problemas de instancias de EC2 no administradas que se pueden diagnosticar
- Ejecución de un diagnóstico y una corrección opcional para instancias de EC2 no administradas
- Programación de un análisis periódico para instancias de EC2 no administradas

Categorías de problemas de instancias de EC2 no administradas que se pueden diagnosticar

En este tema se muestran las principales categorías de problemas de administración de EC2 y los problemas específicos de cada categoría que Systems Manager puede ayudar a diagnosticar y corregir. Tenga en cuenta que, para algunos de los problemas, Systems Manager puede identificar el problema, pero no proporcionar una corrección automática. En esos casos, la consola de Systems Manager lo dirige a la información que será útil para resolver un problema de forma manual.

El proceso de diagnóstico examina cada grupo de instancias de EC2 a la vez según la nube privada virtual (VPC) a la que pertenezcan.

Tipos de problema

- Categoría de problema: configuración de grupos de seguridad y comunicaciones HTTPS
- Categoría de problema: configuración de DNS o nombre de host DNS
- Categoría de problema: configuración de punto de conexión de VPC

Categoría de problema: configuración de grupos de seguridad y comunicaciones HTTPS

Una operación de diagnóstico podría detectar que SSM Agent no puede comunicarse con el servicio de Systems Manager a través de HTTPS. En esos casos, puede optar por ejecutar un manual de procedimientos de Automatización que intente actualizar los grupos de seguridad adjuntos a las instancias.

> **Note**
>
> En ocasiones, es posible que Systems Manager no pueda corregir automáticamente estos problemas, pero puede editar manualmente los grupos de seguridad afectados.

Tipos de problema admitidos

- Grupo de seguridad de instancias: no se permite el tráfico saliente en el puerto 443
- Grupo de seguridad del punto de conexión de VPC de **`ssm`**: no se permite el tráfico entrante en el puerto 443
- Grupo de seguridad del punto de conexión de VPC de **`ssmmessages`**: no se permite el tráfico entrante en el puerto 443
- Grupo de seguridad del punto de conexión de VPC de **`ec2messages`**: no se permite el tráfico entrante en el puerto 443

Para obtener más información, consulte Verificación de las reglas de entrada en los grupos de seguridad de punto de conexión en el tema Solución de problemas de SSM Agent.

Categoría de problema: configuración de DNS o nombre de host DNS

Una operación de diagnóstico podría detectar que el sistema de nombres de dominio (DNS) o los nombres de host DNS no están configurados correctamente para la VPC. En esos casos, puede optar por ejecutar un manual de procedimientos de Automatización que intente habilitar los atributos `enableDnsSupport` y `enableDnsHostnames` de la VPC afectada.

Tipos de problema admitidos

- La compatibilidad con DNS está deshabilitada en una VPC.
- Un nombre de host DNS está deshabilitado en una VPC.

Para obtener más información, consulte Compruebe los atributos relacionados con el DNS de la VPC en el tema Solución de problemas de SSM Agent.

Categoría de problema: configuración de punto de conexión de VPC

Una operación de diagnóstico podría detectar que los puntos de conexión de VPC no están configurados correctamente para la VPC.

Si los puntos de conexión de VPC requeridos por SSM Agent no existen, Systems Manager intenta ejecutar un manual de procedimientos de Automatización para crear los puntos de conexión de VPC y asociarlos a una subred en cada zona de disponibilidad (AZ) regional correspondiente. Si los puntos de conexión necesarios de la VPC existen, pero no están asociados a una subred en la que se encuentra el problema, el manual de procedimientos asocia los puntos de conexión de VPC a la subred afectada.

> **Note**
>
> Systems Manager no admite la solución de todos los problemas de puntos de conexión de VPC mal configurados. En esos casos, Systems Manager le indica las instrucciones de corrección manual en lugar de ejecutar un manual de procedimientos de Automatización.

Tipos de problema admitidos

- No se encontró ningún punto de conexión `ssm.region.amazonaws.com` para PrivateLink.
- No se encontró ningún punto de conexión `ssmmessages.region.amazonaws.com` para PrivateLink.
- No se encontró ningún punto de conexión `ec2messages.region.amazonaws.com` para PrivateLink.

Tipos de problemas diagnosticables

Systems Manager puede diagnosticar los siguientes tipos de problemas, pero actualmente no hay ningún manual de procedimientos disponible para corregirlos. Para estos problemas, puede editar la configuración manualmente.

- La subred de una instancia no está adjuntada a un punto de conexión `ssm.region.amazonaws.com`.
- La subred de una instancia no está adjuntada a un punto de conexión `ssmmessages.region.amazonaws.com`.
- La subred de una instancia no está adjuntada a un punto de conexión `ec2messages.region.amazonaws.com`.

Para obtener más información, consulte Verificación de la configuración VPC en el tema Solución de problemas de SSM Agent.

Ejecución de un diagnóstico y una corrección opcional para instancias de EC2 no administradas

Obtenga información sobre cómo diagnosticar los problemas relacionados con la red y con la VPC que podrían impedir que Systems Manager administre las instancias de EC2.

La operación de diagnóstico puede detectar y agrupar problemas de los siguientes tipos:

- Problemas de configuración de red: tipos de problemas de red que podrían impedir que las instancias de EC2 se comuniquen con el servicio de Systems Manager en la nube. Es posible que haya operaciones de corrección disponibles para estos problemas. Para obtener más información acerca de los problemas de la configuración de red, consulte Categorías de problemas de instancias de EC2 no administradas que se pueden diagnosticar.
- Problemas no identificados: una lista de resultados para los casos en los que la operación de diagnóstico no pudo determinar por qué las instancias de EC2 no pueden comunicarse con el servicio de Systems Manager en la nube.

Ejecución de un diagnóstico y corrección de las instancias de EC2 no administradas

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Diagnosticar y corregir.
3. Elija la pestaña Problema de instancias de EC2 no administradas.
4. En la sección Resumen del problema, elija Ejecutar un nuevo diagnóstico.

   -o bien-

   Si es la primera vez que diagnostica problemas de instancias de EC2 no administradas, en la sección Diagnosticar instancias de EC2 no administradas, elija Ejecutar.

   > **Tip**
   >
   > Mientras se ejecuta el diagnóstico, elija Ver el progreso o Ver ejecuciones para supervisar el estado actual de la ejecución. Para obtener más información, consulte Visualización del progreso de la ejecución y el historial de las correcciones en Systems Manager.

5. Una vez finalizado el diagnóstico, haga lo siguiente:

- Para los problemas notificados en la sección Problemas no identificados, seleccione el enlace Más información para obtener información sobre cómo resolver el problema.
- En el caso de los problemas descritos en la sección Problemas de configuración de red, continúe con el siguiente paso.

6. En la lista de tipos de búsqueda, en la columna Recomendaciones, para un problema concreto, elija el enlace (por ejemplo, 2 recomendaciones).
7. En el panel Recomendaciones que se abre, elija entre las mitigaciones disponibles:

   - Más información: se abre un tema con información sobre cómo resolver un problema manualmente.
   - Ver el manual de procedimientos: se abre un panel con información sobre el manual de procedimientos de Automatización que puede ejecutar para resolver el problema con las instancias de EC2, así como opciones para generar una vista previa de las acciones que llevaría a cabo el manual de procedimientos. Continúe en el paso siguiente.

8. En el panel del manual de procedimientos, haga lo siguiente:

   a. En Descripción del documento, puede revisar el contenido, que proporciona información general de las medidas que el manual de procedimientos puede tomar para solucionar los problemas de las instancias de EC2 no administradas. Elija Ver los pasos para obtener una vista previa de las acciones individuales que llevaría a cabo el manual de procedimientos.

   b. En Destinos, haga lo siguiente:

      - Si va a administrar las correcciones para una organización, en Cuentas, especifique si este manual de procedimientos se destinará a todas las cuentas o solo a un subconjunto de las cuentas que elija.
      - En Regiones, especifique si este manual de procedimientos se destinará a todas las Regiones de AWS de su cuenta u organización, o solo a un subconjunto de las regiones que elija.

   c. En Vista previa del manual de procedimientos, revise detenidamente la información. En esta información se explica cuál sería el alcance y el impacto que tendría si decidiera ejecutar el manual de procedimientos.

> **Note**
>
> Si decide ejecutarlo, se incurriría en gastos. Revise detenidamente la información de la vista previa antes de decidir si desea continuar.

El contenido de Vista previa del manual de procedimientos proporciona la siguiente información:

- En cuántas regiones se llevaría a cabo la operación del manual de procedimientos.
- (Solo organizaciones) En cuántas unidades organizativas (OU) se ejecutaría la operación.
- Los tipos de medidas que se tomarían y el número de cada una de ellas.

  Los tipos de acción son los siguientes:

  - Mutante: el paso del manual de procedimientos hará cambios en los destinos mediante acciones que crean, modifican o eliminan recursos.
  - No mutante: el paso del manual de procedimientos recuperaría los datos sobre los recursos, pero no los modificaría. Por lo general, esta categoría incluye `Describe*`, `List*`, `Get*` y acciones de API similares de solo lectura.
  - Indeterminado: un paso indeterminado invoca las ejecuciones que hace otro servicio de orquestación, como AWS Lambda, AWS Step Functions o AWS Systems Manager Run Command. Un paso indeterminado también podría llamar a una API de terceros. Automatización de Systems Manager no conoce el resultado de los procesos de orquestación ni de las ejecuciones de API de terceros, por lo que los resultados de los pasos son indeterminados.

d. En este caso, puede elegir una de las siguientes acciones:

   - Detenga y no ejecute el manual de procedimientos.
   - Elija Ejecutar para ejecutar el manual de procedimientos con las opciones que ya ha seleccionado.

Si opta por ejecutar la operación, elija Ver el progreso o Ver ejecuciones para supervisar el estado actual de la ejecución. Para obtener más información, consulte Visualización del progreso de la ejecución y el historial de las correcciones en Systems Manager.

Programación de un análisis periódico para instancias de EC2 no administradas

Puede ejecutar un análisis bajo demanda de instancias de Amazon EC2 en su cuenta u organización que Systems Manager no pueda administrar debido a varios problemas de configuración. También puede programar este análisis para que se haga de forma automática y periódica.

Programación de un análisis periódico para instancias de EC2 no administradas

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Diagnosticar y corregir.
3. Elija la pestaña Problema de instancias de EC2 no administradas.
4. En la sección Diagnóstico de instancias de EC2 no administradas, active la opción Programar un diagnóstico periódico.
5. En Frecuencia del diagnóstico, seleccione si desea ejecutar el diagnóstico una vez al día o una vez a la semana.
6. (Opcional) En Hora de inicio, ingrese una hora, en formato de 24 horas, para que comience el diagnóstico. Por ejemplo, para las 20:15 h, especifique **`20:15`**.

   La hora que ingrese corresponde a su zona horaria local actual.

   Si no especifica una hora, el análisis de diagnóstico se ejecuta de inmediato. Systems Manager también programa el análisis para que se ejecute en el futuro a la hora actual. Si especifica una hora, Systems Manager espera para ejecutar el análisis de diagnóstico a la hora especificada.
7. Elija Ejecutar. El diagnóstico se ejecuta de inmediato, pero también se llevará a cabo según la programación que haya especificado.

Tipos de impacto de la corrección de las acciones del manual de procedimientos

Systems Manager puede ejecutar operaciones de diagnóstico que detectan tipos determinados de implementaciones con errores y configuraciones desviadas, así como tipos determinados de problemas de configuración que impiden que Systems Manager administre las instancias de EC2. Los resultados del diagnóstico podrían incluir recomendaciones para los manuales de procedimientos de Automatización que puede ejecutar para intentar corregir un problema. Para obtener más información sobre estas operaciones de diagnóstico, consulte los siguientes temas:

- Diagnóstico y corrección de implementaciones con error
- Diagnóstico y corrección de configuraciones desviadas
- Diagnóstico y corrección de instancias de Amazon EC2 no administradas en Systems Manager

Cuando Systems Manager identifica un problema que podría solucionarse mediante la ejecución de un manual de procedimientos de Automatización en los recursos afectados, le proporciona una vista previa de la ejecución. La vista previa de la ejecución proporciona información sobre los tipos de cambios que haría la ejecución del manual de procedimientos en los destinos. Esta información incluye cuántos cambios de cada uno de los tres tipos identificó el diagnóstico.

Estos tipos de cambios son los siguientes:

- `Mutating`: un paso del manual de procedimientos hará cambios en los destinos mediante acciones que crean, modifican o eliminan recursos.
- `Non-Mutating`: un paso del manual de procedimientos recuperaría los datos sobre los recursos, pero no los modificaría. Por lo general, esta categoría incluye `Describe*`, `List*`, `Get*` y acciones de API similares de solo lectura.
- `Undetermined`: un paso indeterminado invoca las ejecuciones hechas por otro servicio de orquestación, como AWS Lambda, AWS Step Functions o Run Command, una herramienta de AWS Systems Manager. Un paso indeterminado también podría llamar a una API de terceros o ejecutar un script de Python o PowerShell. Automatización de Systems Manager no puede detectar cuál sería el resultado de los procesos de orquestación o de las ejecuciones de una API de terceros y, por lo tanto, no puede evaluarlos. Los resultados de esos pasos deberían revisarse manualmente para determinar su impacto.

  Consulte la siguiente tabla para obtener información sobre el tipo de impacto de las acciones de Automatización compatibles.

Tipos de impacto de las acciones de corrección compatibles

En la tabla se presenta el tipo de impacto (mutante, no mutante e indeterminado) de las diversas acciones que se pueden incluir en un manual de procedimientos.

Acción[1]	Tipo de impacto
aws:approve	No mutante

Acción[1]	Tipo de impacto
aws:assertAwsResourceProperty	No mutante
aws:branch	No mutante
aws:changeInstanceState	Mutante
aws:copyImage	Mutante
aws:createImage	Mutante
aws:createStack	Mutante
aws:createTags	Mutante
aws:deleteImage	Mutante
aws:deleteStack	Mutante
aws:executeAutomation	Indeterminado
aws:executeAwsApi	Indeterminado
aws:executeScript	Indeterminado
aws:executeStateMachine	Indeterminado
aws:invokeLambdaFunction	Indeterminado
aws:invokeWebhook	Indeterminado
aws:loop	Varía. Depende de las acciones del ciclo.
aws:pause	No mutante
aws:runCommand	Indeterminado
aws:runInstances	Mutante
aws:sleep	No mutante

Acción[1]	Tipo de impacto
aws:updateVariable	Mutante
aws:waitForAwsResourceProperty	No mutante

[1] Para obtener más información acerca de las acciones de Automatización, consulte Referencia de acciones de Automatización de Systems Manager.

Visualización del progreso de la ejecución y el historial de las correcciones en Systems Manager

Puede ver una lista de todas las operaciones de corrección en curso y finalizadas que se hacen mediante la característica Diagnosticar y corregir de Systems Manager.

Los datos de la lista del historial de ejecuciones incluyen los siguientes tipos de información:

- El tipo de ejecución, `Diagnosis` o `Remediation`.
- El estado de la ejecución, como `Success` o `Failed`.
- Las horas en que se inició y finalizó la ejecución.

Visualización del progreso de la ejecución y el historial de las correcciones

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Diagnosticar y corregir.
3. Elija Ver ejecuciones.

> **Tip**
>
> Cuando se está llevando a cabo una ejecución, también puede seleccionar Ver el progreso para abrir la página Historial de ejecuciones.

4. (Opcional) En el cuadro de búsqueda (), introduzca una frase que ayude a reducir la lista de ejecuciones, como **EC2** o **VPC**.

5. (Opcional) Para ver detalles adicionales sobre una ejecución, en la columna Nombre de ejecución, elija un nombre de operación, como AWS-DiagnoseUnmanagedEC2NetworkIssues.

   En el panel de detalles, puede revisar la información sobre todos los pasos intentados durante la operación y sobre todas las entradas y salidas de la ejecución.

Ajuste de la configuración de Systems Manager

Las opciones de las páginas de configuración habilitan y configuran las características de la consola unificada de Systems Manager. Las opciones que se muestran dependen de la cuenta en la que haya iniciado sesión y de si ya ha configurado Systems Manager o no.

> **Note**
>
> Las opciones de la página Configuración no afectan a las herramientas de Systems Manager (anteriormente denominadas capacidades).

Ajustes de la configuración de cuenta

Si Systems Manager está habilitado e inició sesión en una cuenta que no es miembro de Organizations o si el administrador delegado no agregó su cuenta de Organizations a Systems Manager, en la página Configuración de la cuenta se muestra la opción Deshabilitar Systems Manager. Al deshabilitar Systems Manager, este no muestra la consola unificada. Todas las herramientas de Systems Manager siguen en funcionamiento.

Ajustes de la configuración organizativa

En la pestaña Configuración organizativa, la sección Región local muestra la Región de AWS elegida como región de origen durante la configuración. En los entornos de varias cuentas y regiones que usan AWS Organizations, Systems Manager agrega automáticamente los datos de nodos de todas las cuentas y regiones a la región local. Agregar datos de esta manera le permite ver los datos de los nodos de todas las cuentas y regiones en una sola ubicación.

> **Note**
>
> Si desea cambiar la región local, debe deshabilitar Systems Manager y volverlo a habilitar. Para deshabilitar Systems Manager, seleccione Deshabilitar.

La sección Configuración organizativa muestra las unidades organizativas de AWS y Regiones de AWS elegidas durante la configuración. Seleccione Editar para cambiar qué unidades organizativas y regiones muestran los datos de los nodos en Systems Manager. Para obtener más información acerca de la configuración de Systems Manager para Organizations, consulte Configuración de AWS Systems Manager.

Configuraciones de características

La sección Configuraciones de características le permite habilitar y configurar las funciones clave de Systems Manager que mejoran la administración de nodos en toda la organización. Estas características trabajan juntas para proporcionar una administración, supervisión del cumplimiento y mantenimiento automatizados de los nodos gestionados.

Puede configurar estas características durante la configuración inicial de Systems Manager o modificarlas más adelante a través de la página de Configuración. Cada característica se puede activar o desactivar de forma independiente en función de los requisitos de su organización.

Configuración de administración de host predeterminada

La Configuración de Administración de Hosts Predeterminada (DHMC) configura automáticamente las instancias de Amazon Elastic Compute Cloud (Amazon EC2) en su organización para que las administre Systems Manager. Cuando está habilitada, DHMC garantiza que las instancias EC2 nuevas y existentes tengan los permisos AWS Identity and Access Management (IAM) y las configuraciones necesarios para comunicarse con los servicios de Systems Manager.

DHMC proporciona los siguientes beneficios:

- Asignación automática de roles de IAM: garantiza que las instancias EC2 tengan los roles y políticas de IAM necesarios para funcionar como nodos gestionados.
- Corrección de desviaciones: corrige automáticamente las desviaciones de configuración cuando las instancias pierden su estado de nodo gestionado.
- Incorporación simplificada: reduce los pasos de configuración manual para las nuevas instancias.
- Configuración uniforme: mantiene una configuración uniforme en toda su flota de EC2.

Configuración de la frecuencia de corrección de desviaciones

La corrección de desviaciones detecta y corrige automáticamente cuando las instancias EC2 pierden su configuración de nodos gestionados. Puede configurar la frecuencia con la que Systems Manager comprueba y corrige las desviaciones de configuración.

Para administrar la configuración de administración de host predeterminada haga lo siguiente:

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, seleccione Configuración.
3. En la sección Configuraciones de características, busque la configuración de administración de hosts predeterminada.
4. Para habilitar la DHMC, active el conmutador.
5. En cuanto a la frecuencia de corrección de desviaciones, elija la frecuencia con la que desea que Systems Manager compruebe y corrija las desviaciones de configuración:
   - Diariamente: comprueba y corrige las desviaciones una vez al día.
   - Semanalmente: comprueba y corrige las desviaciones una vez por semana.
   - Mensualmente: comprueba y corrige las desviaciones una vez al mes.
6. Seleccione Save.

> **Note**
>
> Al habilitar DHMC, Systems Manager crea las políticas y roles de IAM necesarios en su cuenta. Estos roles permiten que las instancias EC2 se comuniquen con los servicios de Systems Manager. Para obtener más información sobre los roles de IAM creados con DHMC, consulte Administrar instancias de EC2 con Systems Manager.

Recopilación de metadatos de inventario

La recopilación de metadatos de inventario reúne automáticamente información detallada sobre sus nodos gestionados, incluidas aplicaciones instaladas, configuraciones de red, actualizaciones del sistema y otros metadatos del sistema. Esta información le ayuda a mantener el cumplimiento, realizar análisis de seguridad y comprender la composición de su infraestructura.

La recopilación de inventario proporciona las siguientes ventajas:

- Supervisión del cumplimiento: realice un seguimiento del software y las configuraciones instalados para generar informes de conformidad.
- Análisis de seguridad: identifique el software obsoleto y las posibles vulnerabilidades de seguridad.

- Administración de activos: mantenga un inventario actualizado de su infraestructura.
- Capacidades de consulta: utilice los datos recopilados con Amazon Q Developer para consultas en lenguaje natural.

Tipos de datos de inventario recopilados

Cuando la recopilación de metadatos de inventario está habilitada, Systems Manager recopila los siguientes tipos de información de los nodos gestionados:

- Aplicaciones: paquetes de software y aplicaciones instalados
- Configuraciones de red: interfaces de red, direcciones IP y configuración de red
- Actualizaciones del sistema: parches instalados y actualizaciones disponibles
- Propiedades del sistema: especificaciones de hardware, detalles del sistema operativo y configuraciones del sistema
- Servicios: servicios en ejecución y sus configuraciones

Configuración de la frecuencia de recopilación de inventario

Puede configurar la frecuencia con la que Systems Manager recopila metadatos de inventario de los nodos administrados. Una recopilación más frecuente proporciona información más actualizada, pero puede aumentar el uso del servicio de AWS.

Para configurar la recopilación de metadatos de inventario, haga lo siguiente:

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, seleccione Configuración.
3. En la sección Configuraciones de características, localice la Recopilación de metadatos de inventario.
4. Para activar la recopilación de inventario, active el conmutador.
5. En Frecuencia de recopilación, elija la frecuencia con la que desea que Systems Manager recopile los datos de inventario:
    - Diariamente: recopila los datos de inventario una vez al día.
    - Semanalmente: recopila los datos de inventario una vez por semana.
    - Mensualmente: recopila los datos de inventario una vez al mes.

6. Seleccione Save.

> **Important**
>
> La recopilación de inventario requiere que los nodos gestionados tengan los permisos necesarios para recopilar la información del sistema. Asegúrese de que los nodos administrados disponen de los roles y las políticas de IAM adecuados. Para obtener más información acerca de los permisos requeridos, consulte Inventario de AWS Systems Manager.

Actualizaciones de SSM Agent

Las actualizaciones automáticas de SSM Agent garantizan que los nodos gestionados ejecuten la última versión de SSM Agent. Mantener el agente actualizado proporciona acceso a las características más recientes, mejoras de seguridad y correcciones de errores.

Las actualizaciones automáticas de SSM Agent proporcionan las siguientes ventajas:

- Características más recientes: acceso a las nuevas funciones y mejoras de Systems Manager
- Actualizaciones de seguridad: instalación automática de parches y correcciones de seguridad
- Fiabilidad mejorada: correcciones de errores y mejoras en la estabilidad
- Mantenimiento reducido: elimina la necesidad de actualizar los agentes manualmente

Configuración de actualizaciones automáticas del agente

Puede configurar la frecuencia con la que Systems Manager comprueba e instala actualizaciones de SSM Agent en los nodos gestionados. Las actualizaciones periódicas ayudan a garantizar un rendimiento y una seguridad óptimos.

Para configurar actualizaciones de SSM Agent, haga lo siguiente:

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, seleccione Configuración.
3. En la sección Configuraciones de características, busque actualizaciones de SSM Agent.
4. Para activar las actualizaciones automáticas, active el conmutador.

5. En Frecuencia de actualización, elija la frecuencia con la que quiere que Systems Manager compruebe e instale las actualizaciones de los agentes:
   - Diariamente: comprueba si hay actualizaciones una vez al día.
   - Semanalmente: comprueba si hay actualizaciones una vez por semana.
   - Mensualmente: comprueba si hay actualizaciones una vez al mes.
6. Seleccione Save.

Configuración de diagnóstico y corrección

La configuración de Diagnosticar y corregir determina si Systems Manager analiza automáticamente los nodos para garantizar que se puedan comunicar con Systems Manager. Si está habilitada, la característica se ejecuta automáticamente según la programación que se defina. La característica identifica qué nodos no se pueden conectar a Systems Manager y la razón por la que no pueden hacerlo. Esta característica también proporciona manuales de procedimientos recomendados para solucionar los problemas de red y otros errores que impiden que los nodos se configuren como nodos administrados.

Programación de un análisis de diagnóstico periódico

Systems Manager puede diagnosticar y ayudar a corregir varios tipos de errores de implementación, así como configuraciones desviadas. Systems Manager también puede identificar las instancias de Amazon Elastic Compute Cloud (Amazon EC2) en su cuenta u organización que no puede tratar como nodos administrados. El proceso de diagnóstico de instancias de EC2 puede identificar problemas relacionados con configuraciones incorrectas en una nube privada virtual (VPC), en una configuración de sistema de nombres de dominio (DNS) o en un grupo de seguridad de Amazon Elastic Compute Cloud (Amazon EC2).

Para simplificar la tarea de identificar los nodos que no se pueden conectar a Systems Manager, la característica Programar diagnósticos periódicos le permite automatizar un análisis de diagnóstico periódico. Los análisis identifican qué nodos no se pueden conectar a Systems Manager y por qué. Use el siguiente procedimiento para habilitar y configurar un análisis de diagnóstico periódico de los nodos.

Programación de un análisis de diagnóstico periódico

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.

2. En el panel de navegación, elija Configuración y luego elija la pestaña Diagnosticar y corregir.
3. Active la opción Programar diagnósticos periódicos.
4. En Periodo de análisis, elija la frecuencia con la que desea que se ejecute el análisis.
5. (Opcional) En Hora de inicio, ingrese una hora, en formato de 24 horas, para que comience el diagnóstico. Por ejemplo, para las 20:15 h, especifique **`20:15`**.

   La hora que ingrese corresponde a su zona horaria local actual.

   Si no especifica una hora, el análisis de diagnóstico se ejecuta de inmediato. Systems Manager también programa el análisis para que se ejecute en el futuro a la hora actual. Si especifica una hora, Systems Manager espera para ejecutar el análisis de diagnóstico a la hora especificada.
6. Seleccione Save.
7. Una vez finalizado el análisis, seleccione Diagnosticar y corregir en el menú de navegación de la izquierda para ver los detalles.

Para obtener más información sobre la característica Diagnosticar y corregir, consulte Diagnóstico y corrección.

Actualización del cifrado de buckets de S3

Al incorporar Systems Manager, Configuración Rápida crea un bucket de Amazon Simple Storage Service (Amazon S3) en la cuenta de administrador delegado para las configuraciones de AWS Organizations. En las configuraciones de una sola cuenta, el bucket se almacena en la cuenta que se está configurando. Este bucket se utiliza para almacenar los metadatos que se generen durante los análisis de diagnóstico.

Para obtener más información acerca de la configuración de la consola unificada de Systems Manager, consulte Configuración de AWS Systems Manager.

De forma predeterminada, los datos del bucket se cifran mediante una clave de AWS Key Management Service (AWS KMS) que AWS posee y administra en su nombre.

Puede utilizar una clave de AWS KMS diferente para el cifrado de los buckets. Como alternativa, puede utilizar el cifrado del lado del servidor con AWS KMS keys (SSE-KMS) mediante una clave administrada por el cliente (CMK). Para obtener más información, consulta Uso de buckets de Amazon S3 y políticas de buckets para Systems Manager.

Uso de una clave de AWS KMS diferente para el cifrado de buckets de S3

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Configuración y luego elija la pestaña Diagnosticar y corregir.
3. En el área Actualizar el cifrado del bucket de S3, seleccione Editar.
4. Seleccione la casilla de verificación Personalizar la configuración de cifrado (avanzado).
5. En Elija una clave de AWS KMS, elija o ingrese el nombre de recurso de Amazon (ARN) de la clave.

   > **Tip**
   >
   > Elija Crear una clave de AWS KMS para crear una clave nueva.

6. Seleccione Save.

Uso de buckets de Amazon S3 y políticas de buckets para Systems Manager

Durante el proceso de incorporación de AWS Systems Manager, Quick Setup crea un bucket de Amazon Simple Storage Service (Amazon S3) en la cuenta de administrador delegado para la configuración de la organización. En las configuraciones de una sola cuenta, el bucket se almacena en la cuenta que se está configurando.

Puede usar Systems Manager para ejecutar operaciones de diagnóstico en su flota a fin de identificar casos de implementaciones con errores y configuraciones desviadas. Systems Manager también puede detectar casos en los que los problemas de configuración impiden que Systems Manager administre las instancias de EC2 en su cuenta u organización. Los resultados de estas operaciones de diagnóstico se almacenan en este bucket de Amazon S3, que está protegido tanto por un método de cifrado como por una política de bucket de S3. Para obtener información sobre las operaciones de diagnóstico que generan datos en este bucket, consulte Diagnóstico y corrección.

Cambio del método de cifrado del bucket

De manera predeterminada, el bucket de S3 usa el cifrado del lado del servidor con las claves administradas de Amazon S3 (SSE-S3).

En su lugar, puede usar el cifrado del servidor con AWS KMS keys (SSE-KMS) mediante una clave administrada por el cliente (CMK) como alternativa a las claves administradas de Amazon S3, como se explica en Cambio a una clave de AWS KMS administrada por el cliente para cifrar los recursos de S3.

Contenido de la política de buckets

La política de buckets impide que las cuentas de los miembros de una organización se detecten unas a otras. Los permisos de lectura y escritura en el bucket solo están permitidos para los roles de diagnóstico y corrección creados para Systems Manager. El contenido de estas políticas generadas por el sistema se presenta en Políticas de bucket de S3 para la consola unificada de Systems Manager.

> **Warning**
>
> La modificación de la política de buckets predeterminada puede permitir que las cuentas de los miembros de una organización se detecten entre sí o lean los resultados de los diagnósticos en busca de instancias de otra cuenta. Le recomendamos que actúe con extrema precaución si decide modificar esta política.

Temas

- Cambio a una clave de AWS KMS administrada por el cliente para cifrar los recursos de S3
- Políticas de bucket de S3 para la consola unificada de Systems Manager

Cambio a una clave de AWS KMS administrada por el cliente para cifrar los recursos de S3

Durante el proceso de incorporación de la consola unificada de Systems Manager, Quick Setup crea un bucket de Amazon Simple Storage Service (Amazon S3) en la cuenta de administrador delegado. Este bucket se utiliza para almacenar los datos de salida del diagnóstico generados durante la ejecución del manual de procedimientos de correcciones. De manera predeterminada, el bucket usa el cifrado del lado del servidor con las claves administradas de Amazon S3 (SSE-S3).

Puede revisar el contenido de estas políticas en Políticas de bucket de S3 para la consola unificada de Systems Manager.

Sin embargo, puede utilizar el cifrado del lado del servidor con AWS KMS keys (SSE-KMS) mediante una clave administrada por el cliente (CMK) como una alternativa a una AWS KMS key.

Complete las siguientes tareas para configurar Systems Manager para que utilice su CMK.

Tarea 1: adición de una etiqueta a una CMK existente

AWS Systems Manager utiliza su CMK solo si está etiquetada con el siguiente par clave-valor:

- Clave: `SystemsManagerManaged`
- Valor: `true`

Utilice el siguiente procedimiento para proporcionar acceso para cifrar el bucket de S3 con la CMK.

Adición de una etiqueta a la CMK existente

1. Abra la consola de AWS KMS en https://console.aws.amazon.com/kms.
2. En el panel de navegación izquierdo, elija Claves administradas por el cliente.
3. Seleccione la AWS KMS key que desea utilizar con AWS Systems Manager.
4. Seleccione la pestaña Etiquetas y, luego, elija Editar.
5. Seleccione Agregar etiqueta.
6. Haga lo siguiente:
   a. En Tag key (Clave de etiqueta), escriba **`SystemsManagerManaged`**.
   b. En Valor de etiqueta, introduzca **`true`**.
7. Seleccione Save.

Tarea 2: modificación de una política de claves de CMK existente

Utilice el siguiente procedimiento para actualizar la política de claves de KMS de la CMK y permitir que los roles de AWS Systems Manager cifren el bucket de S3 en su nombre.

Modificación de una política de claves de CMK existente

1. Abra la consola de AWS KMS en https://console.aws.amazon.com/kms.
2. En el panel de navegación izquierdo, elija Claves administradas por el cliente.
3. Seleccione la AWS KMS key que desea utilizar con AWS Systems Manager.

4. En la pestaña Política de claves, elija Editar.
5. Agregue la siguiente instrucción JSON al campo `Statement` y sustituya los *valores de los marcadores de posición* por su propia información.

   Asegúrese de agregar todos los ID de Cuenta de AWS incorporados en su organización a AWS Systems Manager en el campo `Principal`.

   Para localizar el nombre correcto del bucket en la consola de Amazon S3, en la cuenta de administrador delegado, localice el bucket en el formato `do-not-delete-ssm-`*operational-account-id*`-`*home-region*`-`*disambiguator*.

```
{
     "Sid": "EncryptionForSystemsManagerS3Bucket",
     "Effect": "Allow",
     "Principal": {
         "AWS": [
             "account-id-1",
             "account-id-2",
             ...
         ]
     },
     "Action": ["kms:Decrypt", "kms:GenerateDataKey"],
     "Resource": "*",
     "Condition": {
         "StringEquals": {
             "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::amzn-s3-demo-bucket"
         },
         "StringLike": {
             "kms:ViaService": "s3.*.amazonaws.com"
         },
         "ArnLike": {
             "aws:PrincipalArn": "arn:aws:iam::*:role/AWS-SSM-*"
         }
     }
 }
```

> **Tip**
>
> Como alternativa, puede actualizar la política de claves de la CMK mediante la clave de condición aws:PrincipalOrgID para conceder a AWS Systems Manager el acceso a la CMK.

Tarea 3: especificación de la CMK en la configuración de Systems Manager

Tras completar las dos tareas anteriores, utilice el siguiente procedimiento para cambiar el cifrado del bucket de S3. Este cambio garantiza que el proceso de configuración de Quick Setup asociado pueda agregar permisos para que Systems Manager acepte la CMK.

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, seleccione Configuración.
3. En la pestaña Diagnosticar y corregir, en la sección Actualizar el cifrado del bucket de S3, seleccione Editar.
4. Seleccione la casilla de verificación Personalizar la configuración de cifrado (avanzado).
5. En el cuadro de búsqueda (), elija el ID de una clave existente o pegue el ARN de una clave existente.
6. Seleccione Save.

Políticas de bucket de S3 para la consola unificada de Systems Manager

En este tema se incluyen las políticas de bucket de Amazon S3 que crea Systems Manager al incorporar una organización o una cuenta única a la consola unificada de Systems Manager.

> **Warning**
>
> La modificación de la política de buckets predeterminada puede permitir que las cuentas de los miembros de una organización se detecten entre sí o lean los resultados de los diagnósticos en busca de instancias de otra cuenta. Le recomendamos que actúe con extrema precaución si decide modificar esta política.

Política de buckets de Amazon S3 para una organización

El bucket de diagnóstico se crea con la siguiente política de buckets predeterminada al incorporar una organización a Systems Manager.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "DenyHTTPRequests",
            "Effect": "Deny",
            "Principal": "*",
            "Action": "s3:*",
            "Resource": [
                "arn:aws:s3:::",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ],
            "Condition": {
                "Bool": {
                    "aws:SecureTransport": "false"
                }
            }
        },
        {
            "Sid": "DenyNonSigV4Requests",
            "Effect": "Deny",
            "Principal": "*",
            "Action": "s3:*",
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "s3:SignatureVersion": "AWS4-HMAC-SHA256"
                }
            }
        },
        {
            "Sid": "AllowAccessLog",
            "Effect": "Allow",
```

```
            "Principal": {
                "Service": "logging.s3.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/access-logs/*",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "operational-account-id"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:s3:::amzn-s3-demo-bucket"
                }
            }
        },
        {
            "Sid": "AllowCrossAccountRead",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/actions/*/
${aws:PrincipalAccount}/*",
            "Condition": {
                "StringEquals": {
                    "aws:PrincipalOrgID": "organization-id"
                }
            }
        },
        {
            "Sid": "AllowCrossAccountWrite",
            "Effect": "Allow",
            "Principal": "*",
            "Action": [
                "s3:PutObject",
                "s3:DeleteObject"
            ],
            "Resource": "arn:aws:s3:::bucket-name/actions/*/
${aws:PrincipalAccount}/*",
            "Condition": {
                "StringEquals": {
                    "aws:PrincipalOrgID": "organization-id"
                },
                "ArnLike": {
                    "aws:PrincipalArn": [
```

```
                        "arn:aws:iam::*:role/AWS-SSM-
DiagnosisExecutionRole-operational-account-id-home-region",
                        "arn:aws:iam::*:role/AWS-SSM-
DiagnosisAdminRole-operational-account-id-home-region",
                        "arn:aws:iam::*:role/AWS-SSM-
RemediationExecutionRole-operational-account-id-home-region",
                        "arn:aws:iam::*:role/AWS-SSM-
RemediationAdminRole-operational-account-id-home-region"
                    ]
                }
            }
        },
        {
            "Sid": "AllowCrossAccountListUnderAccountOwnPrefix",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:ListBucket",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket",
            "Condition": {
                "StringEquals": {
                    "aws:PrincipalOrgID": "organization-id"
                },
                "StringLike": {
                    "s3:prefix": "*/${aws:PrincipalAccount}/*"
                }
            }
        },
        {
            "Sid": "AllowCrossAccountGetConfigWithinOrganization",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:GetEncryptionConfiguration",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket",
            "Condition": {
                "StringEquals": {
                    "aws:PrincipalOrgID": "organization-id"
                }
            }
        }
    ]
}
```

Política de buckets de Amazon S3 para una sola cuenta

El bucket de diagnóstico se crea con la siguiente política de buckets predeterminada al incorporar una sola cuenta a Systems Manager.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyHTTPRequests",
      "Effect": "Deny",
      "Principal": "*",
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket",
        "arn:aws:s3:::amzn-s3-demo-bucket/*"
      ],
      "Condition": {
        "Bool": {
          "aws:SecureTransport": "false"
        }
      }
    },
    {
      "Sid": "DenyNonSigV4Requests",
      "Effect": "Deny",
      "Principal": "*",
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket",
        "arn:aws:s3:::amzn-s3-demo-bucket/*"
      ],
      "Condition": {
        "StringNotEquals": {
          "s3:SignatureVersion": "AWS4-HMAC-SHA256"
        }
      }
    }
  ]
}
```

Cómo utilizar herramientas de AWS Systems Manager

Systems Manager agrupa las herramientas en cuatro categorías: La siguiente documentación describe las distintas herramientas de AWS Systems Manager, cómo configurarlas y utilizarlas. Elija las pestañas de cada categoría para obtener más información sobre cada herramienta.

Herramientas de nodos

Un nodo administrado es cualquier máquina configurada para su uso con Systems Manager en entornos híbridos y multinube.

Compliance

Utilice Compliance (Conformidad) para analizar la flota de nodos administrados en cuanto a la conformidad de revisiones y las incoherencias de configuración. Puede recopilar y agregar datos de varias Cuentas de AWS y Regiones de AWS, y luego desglosarlas en recursos específicos que no sean conformes. De forma predeterminada, Compliance muestra datos de conformidad sobre las revisiones de Patch Manager y las asociaciones de State Manager. También puede personalizar el servicio y crear sus propios tipos de conformidad en función de sus requisitos empresariales o de IT.

Distributor

Utilice Distributor para crear e implementar paquetes en los nodos administrados. Con Distributor, puede empaquetar su propio software (o buscar paquetes de software de agente proporcionados por AWS, como AmazonCloudWatchAgent), para instalar en los nodos administrados de Systems Manager. Después de instalar un paquete por primera vez, puede utilizar Distributor para desinstalar y volver a instalar una nueva versión del paquete o realizar una actualización in situ que agregue archivos nuevos o modificados. Distributor publica recursos, como paquetes de software, en nodos administrados por Systems Manager.

Fleet Manager

Fleet Manager es una experiencia de interfaz de usuario (UI) unificada que lo ayuda a administrar en forma remota sus nodos. Con Fleet Manager, puede ver el estado y el rendimiento de toda la flota desde una sola consola. También puede recopilar datos de dispositivos individuales para realizar tareas comunes de solución de problemas y administración desde la consola. Esto incluye ver el contenido de directorios y archivos, administración del registro de Windows, administración de usuarios del sistema operativo y mucho más.

Hybrid Activations

Para configurar máquinas que no sean de EC2 en su entorno híbrido y multinube como nodos administrados, cree una activación híbrida. Después de completar la activación, recibirá un código de activación y un ID. Esta combinación de ID y código funciona como un ID de acceso de Amazon Elastic Compute Cloud (Amazon EC2) y una clave secreta para proporcionar un acceso seguro al servicio de Systems Manager desde las instancias administradas.

También puede crear una activación para dispositivos de borde si desea administrarlos mediante Systems Manager.

Inventory

Inventory (Inventario) automatiza el proceso de recopilación del inventario de software de los nodos administrados. Puede utilizar Inventory para recopilar metadatos sobre las aplicaciones, los archivos, los componentes, las revisiones y más.

Patch Manager

Utilice Patch Manager para automatizar el proceso de aplicación de revisiones a los nodos administrados con actualizaciones relacionadas con la seguridad y otros tipos de actualizaciones. Puede utilizar Patch Manager para aplicar revisiones a los sistemas operativos y a las aplicaciones. (En Windows Server, la compatibilidad con las aplicaciones se limita a las actualizaciones de las aplicaciones publicadas por Microsoft).

Esta herramienta le permite analizar los nodos administrados para detectar las revisiones faltantes y aplicar dichas revisiones de manera individual o a grandes grupos de nodos administrados mediante el uso de etiquetas. Patch Manager utiliza líneas de base de revisiones, que pueden incluir reglas para la aprobación automática de revisiones a los pocos días de su lanzamiento, y una lista de las revisiones aprobadas y rechazadas. Puede instalar revisiones de seguridad periódicamente si programa la ejecución de la aplicación de revisiones como una tarea de periodo de mantenimiento de Systems Manager o puede aplicar revisiones a los nodos administrados bajo demanda en cualquier momento.

Para los sistemas operativos Linux, puede definir los repositorios que deben utilizarse en las operaciones de aplicación de revisiones como parte de la línea de base de revisiones. Esto permite asegurarse de que las actualizaciones se instalen solo desde repositorios de confianza, independientemente de cuáles de ellos estén configurados en el nodo administrado. Para Linux, también tiene la capacidad de actualizar cualquier paquete en el nodo administrado, no solo aquellos que se hayan clasificado como actualizaciones de seguridad del sistema operativo.

También puede generar informes de revisiones que se envían a un bucket de S3 de su elección. Para un único nodo administrado, los informes incluyen detalles de todas las revisiones de la máquina. Para un informe sobre todos los nodos administrados, solo se proporciona un resumen de cuántas revisiones faltan.

Run Command

Utilice Run Command para administrar de forma remota y segura la configuración de los nodos administrados a escala. Utilice Run Command para realizar cambios bajo demanda, como actualizar aplicaciones o ejecutar scripts de shell de Linux y comandos de Windows PowerShell en un destino definido compuesto por docenas o centenares de nodos administrados.

Session Manager

Utilice Session Manager para administrar los dispositivos de borde y las instancias de Amazon Elastic Compute Cloud (Amazon EC2) a través de un shell interactivo basado en el navegador con un solo clic o con la AWS CLI. Session Manager proporciona una administración de instancias y dispositivos de borde segura y auditable sin la necesidad de abrir los puertos de entrada, mantener anfitriones bastión o administrar claves de SSH. Session Manager también facilita el cumplimiento con las políticas corporativas que requieren acceso controlado a dispositivos de borde e instancias, prácticas de seguridad estrictas y registros completamente auditables con detalles del acceso a los dispositivos de borde e instancias, a la vez que ofrecen a los usuarios finales un acceso multiplataforma sencillo con un solo clic a los dispositivos de borde y a las instancias de EC2. Para utilizar Session Manager, debe habilitar el nivel de instancias avanzadas. Para obtener más información, consulte Activación del nivel de instancias avanzadas.

State Manager

Utilice State Manager para automatizar el proceso de mantener los nodos administrados en un estado definido. Puede utilizar State Manager para garantizar que los nodos administrados arranquen con un software específico durante el inicio, se unan a un dominio de Windows (solo nodos de Windows Server) o se les apliquen revisiones con actualizaciones de software específicas.

Herramientas de administración de cambios

Automation

Utilice Automation para automatizar las tareas comunes de mantenimiento e implementación. Puede utilizar Automation para crear y actualizar Amazon Machine Images (AMIs), aplicar

actualizaciones de controladores y agentes, restablecer contraseñas en instancias de Windows Server, restablecer las claves de SSH en las instancias de Linux y aplicar revisiones del OS o actualizaciones de aplicaciones.

Calendario de cambios

Change Calendar lo ayuda a configurar rangos de fecha y hora cuando las acciones que especifica (por ejemplo, en manuales de procedimientos de Automatización de Systems Manager) pueden realizarse o no en su Cuenta de AWS. En Change Calendar, estos intervalos se denominan eventos. Cuando crea una entrada de Change Calendar, está creando un documento de Systems Manager del tipo `ChangeCalendar`. En Change Calendar, el documento almacena datos iCalendar 2.0 en texto sin formato. Los eventos que añada a la entrada de Change Calendar pasan a formar parte del documento. Puede agregar eventos manualmente en la interfaz de Change Calendar o importar eventos desde un calendario de terceros compatible mediante un archivo `.ics`.

Change Manager

Change Manager es un marco empresarial de administración de cambios con el que se pueden solicitar, aprobar, implementar e informar los cambios operativos de la configuración y la infraestructura de la aplicación. A partir de una sola cuenta de administrador delegado, si utiliza AWS Organizations, puede administrar los cambios a través de varias Cuentas de AWS en varias Regiones de AWS. De forma alternativa, a través de una cuenta local, puede administrar los cambios de una sola Cuenta de AWS. Utilice Change Manager para administrar los cambios tanto en los recursos de AWS como en los recursos locales.

Documents

Un documento de Systems Manager (documento de SSM) define las acciones que Systems Manager realiza. Los tipos de documentos de SSM incluyen documentos de Command, que son utilizados por State Manager y Run Command, y manuales de procedimientos de Automation, que utiliza Automatización de Systems Manager. Systems Manager incluye varias docenas de documentos preconfigurados que puede utilizar especificando los parámetros en tiempo de ejecución. Los documentos se pueden expresar en JSON o YAML y contienen los pasos y los parámetros que se especifican.

Períodos de mantenimiento

Utilice Maintenance Windows a fin de configurar programaciones periódicas para que las instancias administradas ejecuten tareas administrativas, como instalar revisiones y actualizaciones sin interrumpir las operaciones esenciales del negocio.

Configuración Rápida

Utilice Quick Setup para configurar Servicios de AWS y características utilizados con frecuencia mediante las prácticas recomendadas. Puede utilizar Quick Setup en una Cuenta de AWS individual o en varias Cuentas de AWS y Regiones de AWS mediante su integración a AWS Organizations. Quick Setup simplifica la configuración de servicios, incluido Systems Manager, mediante la automatización de tareas comunes o recomendadas. Estas tareas incluyen, por ejemplo, la creación de roles de perfil de instancias de AWS Identity and Access Management (IAM) necesarios y la configuración de prácticas recomendadas operativas, como análisis periódicos de revisiones y recopilación de inventario.

Herramientas de aplicaciones

AppConfig

AppConfig ayuda a crear, administrar e implementar rápidamente configuraciones de aplicaciones e indicadores de características. AppConfig admite implementaciones controladas en aplicaciones de cualquier tamaño. Puede utilizar AppConfig con aplicaciones alojadas en instancias de Amazon EC2, contenedores de AWS Lambda, aplicaciones móviles o dispositivos de borde. Para evitar errores al implementar configuraciones de aplicaciones, AppConfig incluye validadores. Un validador proporciona una verificación sintáctica o semántica para verificar que la configuración que desea implementar funciona según lo previsto. Durante la implementación de una configuración, AppConfig monitorea la aplicación para verificar que la implementación se realiza correctamente. Si el sistema encuentra un error o si la implementación invoca una alarma, AppConfig deshace el cambio para minimizar el impacto para los usuarios de la aplicación.

Application Manager

Application Manager ayuda a los ingenieros de DevOps a investigar y corregir problemas en los recursos de AWS en el contexto de las aplicaciones y los clústeres. En Application Manager, una aplicación es un grupo lógico de recursos de AWS que usted desea que opere como una unidad. Este grupo lógico puede representar diferentes versiones de una aplicación, límites de propiedad para operadores o entornos de desarrollador, por nombrar algunos. La compatibilidad de Application Manager con clústeres de contenedores incluye clústeres de Amazon Elastic Kubernetes Service (Amazon EKS) y Amazon Elastic Container Service (Amazon ECS). Application Manager agrega información de operaciones de múltiples Servicios de AWS y herramientas de Systems Manager a una única de AWS Management Console.

Parameter Store:

Parameter Store proporciona un almacenamiento seguro y jerárquico para administrar los datos de configuración y los secretos comerciales. Puede almacenar datos como contraseñas, cadenas de base de datos, ID de instancia de Amazon Elastic Compute Cloud (Amazon EC2), ID de Amazon Machine Image (AMI) y códigos de licencia como valores de parámetros. Puede almacenar valores como texto sin formato o como datos cifrados. A continuación, puede hacer referencia a los valores utilizando el nombre exclusivo especificado al crear el parámetro.

Herramientas de operaciones

CloudWatch Dashboards

Los Amazon CloudWatch Dashboards (Paneles de Amazon CloudWatch son páginas personalizables en la consola de CloudWatch que puede utilizar para monitorear los recursos en una vista única, incluso aquellos que se reparten entre diferentes Regiones. Puede utilizar los paneles de CloudWatch para crear vistas personalizadas de las métricas y las alarmas para los recursos de AWS.

Explorer

Explorer es un panel de operaciones personalizable que transmite información sobre sus recursos de AWS. Explorer muestra una vista agregada de los datos de operaciones (OpsData) de sus Cuentas de AWS y en todas las Regiones de AWS. En Explorer, OpsData incluye metadatos sobre instancias de Amazon EC2, detalles de conformidad de las revisiones y elementos de trabajo operativos (OpsItems). Explorer proporciona un contexto sobre cómo se distribuyen los OpsItems entre las unidades empresariales o las aplicaciones, cómo se presentan a lo largo del tiempo y cómo varían según la categoría. Puede agrupar y filtrar la información en Explorer para centrarse en los elementos que son relevantes para usted y que requieren que se tomen medidas. Cuando identifique problemas de alta prioridad, puede utilizar OpsCenter, una herramienta de Systems Manager, para ejecutar manuales de procedimientos de Automatización y resolver esos problemas.

Incident Manager

Incident Manager es una consola de administración de incidentes que ayuda a los usuarios a mitigar y recuperarse de incidentes que afectan las aplicaciones alojadas de AWS.

Incident Manager aumenta la resolución de incidentes notificando a los encargados de responder sobre el impacto, resaltando los datos relevantes de solución de problemas y proporcionando

herramientas de colaboración para hacer que los servicios se vuelvan a poner en funcionamiento. Incident Manager también automatiza los planes de respuesta y permite la escalada del equipo de respuesta.

OpsCenter

OpsCenter proporciona una ubicación central donde los ingenieros de operaciones y los profesionales de IT pueden ver, investigar y resolver los elementos de trabajo operativo (OpsItems) relacionados con los recursos de AWS. OpsCenter está diseñado para reducir el tiempo de resolución de problemas que afectan a los recursos de AWS. Esta herramienta de Systems Manager agrega y estandariza OpsItems en todos los servicios, al mismo tiempo que proporciona datos de investigación contextual sobre cada OpsItem, OpsItems relacionados y recursos relacionados. OpsCenter también proporciona manuales de procedimientos de Automatización de Systems Manager que puede utilizar para resolver problemas. Puede especificar datos que se pueden buscar y personalizar para cada OpsItem. También puede ver informes de resumen generados automáticamente sobre OpsItems por estado y origen.

Realización de una tarea de nodos con Systems Manager

Utilice esta explicación para empezar a trabajar con AWS Systems Manager. Aprenderá a lanzar una instancia de Amazon Elastic Compute Cloud (Amazon EC2) que administra Systems Manager y a conectarse a la instancia administrada.

Debido a que Systems Manager es una colección de varias herramientas, ninguna explicación o tutorial puede presentar el servicio completo. En este tutorial, solo se proporciona una introducción a algunas de las herramientas.

Requisitos previos

Antes de comenzar, asegúrese de que ha realizado los pasos que se detallan en Administrar instancias de EC2 con Systems Manager.

Lanzar una instancia mediante el uso de una AMI con SSM Agent preinstalado

Puede lanzar una instancia de Amazon EC2 mediante la AWS Management Console, tal como se describe en el siguiente procedimiento. Este tutorial tiene por objetivo ayudarlo a lanzar su primera instancia administrada rápidamente, por lo que no cubre todas las opciones posibles.

Cómo lanzar una instancia

1. Abra la consola de Amazon EC2 en https://console.aws.amazon.com/ec2/.
2. Desde el panel de la consola de EC2, en el cuadro Launch instance (Lanzar instancia), elija Launch instancia y, luego, elija Launch instance de las opciones que aparecen.
3. En Nombre y etiquetas, ingrese un nombre descriptivo para la instancia en Nombre.
4. En Imágenes de aplicaciones y sistema operativo (imagen de máquina de Amazon), realice lo siguiente:

   a. Elija la pestaña Inicio rápido y, a continuación, seleccione Amazon Linux. Este es el sistema operativo (SO) de la instancia.

   b. En Imagen de máquina de Amazon (AMI), elija una versión HVM de Amazon Linux 2.
5. En Tipo de instancia, desde la lista Tipo de instancia, seleccione la configuración de hardware de la instancia. Seleccione el tipo de instancia de `t2.micro`, que es la opción predeterminada. El tipo de instancia `t2.micro` es apto para el nivel gratuito de AWS. En las Regiones de AWS en las que `t2.micro` no esté disponible, puede usar una instancia `t3.micro` en el nivel gratuito. Para obtener más información, consulte Capa gratuita de AWS.
6. En Par de claves (inicio de sesión), para Nombre del par de claves, elija un par de claves.
7. En Configuración de red, elija Editar. En Nombre del grupo de seguridad, verá que el asistente ha creado y seleccionado un grupo de seguridad. Puede usar este grupo de seguridad o, como opción, puede seleccionar el grupo de seguridad que ha creado anteriormente con los siguientes pasos:

   a. Elija Select existing security group (Seleccionar un grupo de seguridad existente).

   b. Desde Common security groups (Grupos de seguridad comunes), elija el grupo de seguridad de la lista de grupos de seguridad existentes.
8. Si no utiliza la configuración de administración de host predeterminada, expanda la sección Detalles avanzados y, para el perfil de instancia de IAM, elija el perfil de instancia que ha creado al configurarlo en Configuración de permisos de instancia requeridos para Systems Manager.
9. Mantenga las selecciones predeterminadas para los demás ajustes de configuración de la instancia.
10. Revise un resumen de la configuración de la instancia en el panel Resumen. Cuando esté preparado, elija Lanzar instancia.

11. Verá una página de confirmación que le informará que la instancia se está lanzando. Elija View all instances (Ver todas las instancias) para cerrar la página de confirmación y volver a la consola.
12. Puede ver el estado del lanzamiento en la pantalla Instancias. La instancia tarda poco tiempo en lanzarse.
13. Puede que transcurran unos minutos hasta que la instancia se muestre como administrada y esté lista para conectarse. Para comprobar que la instancia haya superado las comprobaciones de estado, puede ver esta información en la columna Comprobación de estado.

Conexión a una instancia administrada mediante Systems Manager

Para conectarse a la instancia administrada

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija el botón situado junto a la instancia a la que desea conectarse.
4. En el menú Acciones del nodo, seleccione Iniciar sesión de terminal.
5. Seleccione Conectar.

Elimine la instancia

Si ha terminado de trabajar con la instancia administrada que ha creado para esta explicación, termínela. Al terminar una instancia, esta se elimina de forma eficaz.

Para terminar la instancia

1. Abra la consola de Amazon EC2 en https://console.aws.amazon.com/ec2/.
2. En el panel de navegación, seleccione Instances (Instancia[s]). En la lista de instancias, seleccione la instancia.
3. Elija Instance state (Estado de la instancia) y Terminate instance (Terminar instancia).
4. Cuando se le indique que confirme, elija Terminate (Terminar).

Amazon EC2 apaga y termina la instancia. Una vez terminada la instancia, permanecerá visible en la consola durante un breve periodo y, a continuación, se eliminará automáticamente la entrada. No puede quitar la instancia terminada de la pantalla de la consola por sí mismo.

Herramientas de nodos de AWS Systems Manager

AWS Systems Manager proporciona las siguientes herramientas para obtener acceso, administrar y configurar los nodos administrados. Un nodo administrado es cualquier máquina configurada para su uso con Systems Manager en un entorno híbrido y multinube.

Temas

- Conformidad de AWS Systems Manager
- AWS Systems Manager Distributor
- AWS Systems Manager Fleet Manager
- Activaciones híbridas de AWS Systems Manager
- Inventario de AWS Systems Manager
- AWS Systems Manager Patch Manager
- AWS Systems Manager Run Command
- AWS Systems Manager Session Manager
- AWS Systems Manager State Manager

Conformidad de AWS Systems Manager

Puede utilizar Cumplimiento, una herramienta de AWS Systems Manager, para analizar la flota de nodos administrados en busca de cumplimiento de revisiones e incoherencias de configuración. Puede recopilar y agregar datos de varias Cuentas de AWS y regiones, y luego desglosarlas en recursos específicos que no sean conformes. De forma predeterminada, Cumplimiento muestra datos de cumplimiento actuales sobre la aplicación de parches en Patch Manager y de asociaciones de State Manager. (Patch Manager y State Manager también son herramientas de AWS Systems Manager). Para comenzar a utilizar Compliance, abra la consola de Systems Manager. En el panel de navegación, elija Compliance.

Se pueden enviar los datos de conformidad de revisiones de Patch Manager a AWS Security Hub. Security Hub ofrece una visión completa de las alertas de seguridad de alta prioridad y el estado de

conformidad. También monitorea el estado de aplicación de revisiones de la flota. Para obtener más información, consulte Integración de Patch Manager con AWS Security Hub.

Compliance ofrece los siguientes beneficios y características adicionales:

- visualización del seguimiento de cambios y el historial de conformidad para los datos de aplicación de parches de Patch Manager y las asociaciones de State Manager mediante AWS Config
- Personalización de Compliance para crear sus propios tipos de conformidad en función de sus requisitos empresariales o de TI.
- Corrija problemas mediante Run Command, otra herramienta de AWS Systems Manager, State Manager o Amazon EventBridge.
- transferencia de datos a Amazon Athena y Amazon QuickSight para generar informes de toda la flota

Compatibilidad con EventBridge

Esta herramienta de Systems Manager se admite como un tipo de evento en las reglas de Amazon EventBridge. Para obtener más información, consulte Monitoreo de eventos de Systems Manager con Amazon EventBridge y Referencia: patrones y tipos de eventos de Amazon EventBridge para Systems Manager.

Integración de Chef InSpec

Systems Manager se integra a Chef InSpec. InSpec es un marco de trabajo de tiempo de ejecución de código abierto que permite crear perfiles de lenguaje natural en GitHub o en Amazon Simple Storage Service (Amazon S3). A continuación, puede utilizar Systems Manager para ejecutar análisis de conformidad y ver cuáles nodos administrados son conformes y cuáles no. Para obtener más información, consulte Utilización de perfiles de Chef InSpec con la conformidad de Systems Manager.

Precios

Compliance se ofrece sin cargo adicional. Solo pagará por los recursos de AWS que utilice.

Contenido

- Introducción a Compliance
- Cómo configurar permisos para Compliance

- Cómo crear una sincronización de datos de recursos para Compliance
- Detalles sobre el cumplimiento
- Eliminación de una sincronización de datos de recursos para Compliance
- Solución de problemas de conformidad con EventBridge
- Asignar metadatos de conformidad personalizados mediante la AWS CLI

Introducción a Compliance

Para comenzar con Cumplimiento, una herramienta de AWS Systems Manager, complete las siguientes tareas.

Tarea	Para obtener más información
Cumplimiento funciona con los datos de revisiones en Patch Manager y las asociaciones de State Manager. (Patch Manager y State Manager también son herramientas de AWS Systems Manager). Compliance también funciona con tipos de conformidad personalizados en nodos administrados mediante Systems Manager. Verifique que haya completado los requisitos de configuración para las instancias de Amazon Elastic Compute Cloud (Amazon EC2) y las máquinas que no sean de EC2 en un entorno híbrido y multinube.	Configuración de la consola unificada de Systems Manager para una organización
Actualice el rol AWS Identity and Access Management (IAM) que utilizan sus nodos administrados para restringir los permisos de Compliance.	Cómo configurar permisos para Compliance
Si tiene previsto monitorear la conformidad de parches, compruebe que ha configurado Patch Manager. Debe realizar las operaciones de aplicación de parches mediante Patch Manager	AWS Systems Manager Patch Manager

Tarea	Para obtener más información
para que Compliance pueda mostrar los datos de conformidad de parches.	
Si tiene previsto monitorear la conformidad de asociación, verifique que haya creado asociaciones de State Manager. Debe crear asociaciones para que Compliance pueda mostrar los datos de conformidad de asociación.	AWS Systems Manager State Manager
(Opcional) Configure el sistema para ver el seguimiento de cambios y el historial de conformidad.	Visualización del seguimiento de cambios y del historial de Configuration Compliance
(Opcional) Cree tipos de conformidad personalizados.	Asignar metadatos de conformidad personalizados mediante la AWS CLI
(Opcional) Cree una sincronización de datos de recursos para agregar todos los datos de conformidad a un bucket de Amazon Simple Storage Service (Amazon S3) de destino.	Cómo crear una sincronización de datos de recursos para Compliance

Cómo configurar permisos para Compliance

Como práctica recomendada de seguridad, sugerimos que actualice el rol AWS Identity and Access Management (IAM) que utilizan los nodos administrados con los siguientes permisos para restringir la capacidad del nodo de utilizar la acción de la API PutComplianceItems. Esta acción de API registra un tipo de conformidad junto con sus detalles en un recurso designado, como una instancia de Amazon EC2 o un nodo administrado.

Si su nodo es una instancia de Amazon EC2, debe actualizar el perfil de instancia de IAM utilizado por la instancia con los siguientes permisos. Para obtener más información sobre los perfiles de instancia para instancias EC2 administradas por Systems Manager, consulta Configuración de permisos de instancia requeridos para Systems Manager. Para otros tipos de nodos administrados, actualiza el rol de IAM utilizado por el nodo con los siguientes permisos. Para obtener más información, consulte Actualización de los permisos de un rol en la Guía del usuario de IAM.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:PutComplianceItems"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "ec2:SourceInstanceARN": "${ec2:SourceInstanceARN}"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:PutComplianceItems"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "ssm:SourceInstanceARN": "${ssm:SourceInstanceARN}"
                }
            }
        }
    ]
}
```

Cómo crear una sincronización de datos de recursos para Compliance

Puede utilizar la característica sincronización de datos de recursos de AWS Systems Manager para enviar datos de conformidad de todos los nodos administrados a un bucket de Amazon Simple Storage Service (Amazon S3) de destino. Cuando crea la sincronización, puede especificar los nodos administrados de varias Cuentas de AWS, Regiones de AWS y su entorno híbrido y multinube. A continuación, la sincronización de datos de recursos actualiza automáticamente los datos centralizados cuando se recopilan los nuevos datos de conformidad. Con todos los datos

de conformidad almacenados en un bucket de S3 de destino, puede usar servicios como Amazon Athena y Amazon QuickSight para consultar y analizar los datos agregados. La configuración de la sincronización de datos de recursos para Compliance es una operación que se tiene que realizar una vez.

Utilice el siguiente procedimiento para crear una sincronización de datos de recursos para Compliance mediante la AWS Management Console.

Cómo crear y configurar un bucket de S3 para la sincronización de datos de recursos (consola)

1. Abra la consola de Amazon S3 en https://console.aws.amazon.com/s3.
2. Cree un bucket para almacenar sus datos de conformidad agregados. Para obtener más información, consulte Crear un bucket en la Guía del usuario de Amazon Simple Storage Service. Anote el nombre de bucket y la Región de AWS donde lo creó.
3. Abra el bucket, seleccione la pestaña Permissions (Permisos) y, a continuación, elija Bucket Policy (Política de bucket).
4. Copie y pegue la siguiente política de bucket en el editor de políticas. Reemplace amzn-s3-demo-bucke y *Account-ID* por el nombre del bucket de S3 que ha creado y un ID de Cuenta de AWS válido. Si lo desea, reemplace *Bucket-Prefix* por el nombre de un prefijo de Amazon S3 (subdirectorio). Si no ha creado un prefijo, quite *Bucket-Prefix*/ del ARN de la política.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "SSMBucketPermissionsCheck",
            "Effect": "Allow",
            "Principal": {
                "Service": "ssm.amazonaws.com"
            },
            "Action": "s3:GetBucketAcl",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket"
        },
        {
            "Sid": " SSMBucketDelivery",
            "Effect": "Allow",
```

```
            "Principal": {
                "Service": "ssm.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": ["arn:aws:s3:::amzn-s3-demo-bucket/Bucket-Prefix/*/
accountid=Account_ID_number/*"],
            "Condition": {
                "StringEquals": {
                    "s3:x-amz-acl": "bucket-owner-full-control"
                }
            }
        }
    ]
}
```

Cómo crear una sincronización de datos de recursos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija Account management (Administración de cuenta), Resource Data Syncs (Sincronizaciones de datos de recursos) y, a continuación, elija Create resource data sync (Crear sincronización de datos de recursos).
4. En el campo Sync name (Sincronizar nombre) ingrese el nombre de la configuración de sincronización.
5. En el campo Bucket name (Nombre de bucket), ingrese el nombre del bucket de Amazon S3 que ha creado al comienzo de este procedimiento.
6. (Opcional) En el campo Prefijo de bucket, ingrese el nombre de un prefijo de bucket de S3 (subdirectorio).
7. En el campo Región de bucket, elija Esta región si el bucket de S3 que ha creado se encuentra en la Región de AWS actual. Si el bucket se encuentra en otra Región de AWS, elija Otra región e ingrese el nombre de la región.

> **Note**
>
> Si la sincronización y el bucket de S3 de destino se encuentran en regiones diferentes, es posible que esté sujeto a precios de transferencia de datos. Para obtener más información, consulte Precios de Amazon S3.

8. Seleccione Crear.

Detalles sobre el cumplimiento

Cumplimiento, una herramienta de AWS Systems Manager, recopila y genera informes de datos sobre el estado de aplicación de parches en Patch Manager, la aplicación de parches y las asociaciones en State Manager. (Patch Manager y State Manager también son herramientas de AWS Systems Manager). Compliance también notifica sobre los tipos de conformidad personalizados que ha especificado para los nodos administrados. En esta sección, se incluyen detalles sobre cada uno de estos tipos de conformidad y sobre cómo ver los datos de conformidad de Systems Manager. En esta sección, también se incluye información sobre cómo ver el seguimiento de cambios y el historial de cumplimiento.

> **Note**
>
> Systems Manager se integra a Chef InSpec. InSpec es un marco de trabajo de tiempo de ejecución de código abierto que permite crear perfiles de lenguaje natural en GitHub o en Amazon Simple Storage Service (Amazon S3). A continuación, puede utilizar Systems Manager para ejecutar análisis de conformidad y ver cuáles instancias son conformes y cuáles no. Para obtener más información, consulte Utilización de perfiles de Chef InSpec con la conformidad de Systems Manager.

Acerca de la conformidad de parches

Después de utilizar Patch Manager para instalar los parches en las instancias, la información sobre el estado de la conformidad estará disponible inmediatamente en la consola o como respuesta a los comandos de la AWS Command Line Interface (AWS CLI) o a las operaciones de la API de Systems Manager correspondientes.

Para obtener información sobre los valores de estado de conformidad de parches, consulte Valores de estado de conformidad de las revisiones.

Acerca de la conformidad de las asociaciones de State Manager

Después de crear una o varias asociaciones de State Manager, la información sobre el estado de la conformidad estará disponible inmediatamente en la consola o como respuesta a los comandos de la AWS CLI o a las operaciones de la API de Systems Manager correspondientes. Para las asociaciones, Compliance muestra los estados `Compliant` o `Non-compliant` y el nivel de severidad asignados a la asociación, como, por ejemplo, `Critical` o `Medium`.

Cuando State Manager ejecuta una asociación en un nodo administrado, desencadena un proceso de agregación de cumplimiento que actualiza el estado de cumplimiento de todas las asociaciones de ese nodo. El valor `ExecutionTime` de los informes de cumplimiento representa cuando Systems Manager capturó el estado de cumplimiento, no cuándo se ejecutó la asociación en el nodo administrado. Esto significa que varias asociaciones pueden mostrar valores de `ExecutionTime` idénticos incluso si se ejecutaron en momentos diferentes. Para determinar los tiempos reales de ejecución de la asociación, consulte el historial de ejecución de la asociación mediante el comando de la AWS CLI describe-association-execution-targets o consultando los detalles de la ejecución en la consola.

Acerca de la conformidad personalizada

Puede asignar metadatos de conformidad a un nodo administrado. Estos metadatos se pueden agregar a otros datos de conformidad para elaborar informes de conformidad. Por ejemplo, supongamos que su negocio ejecuta las versiones 2.0, 3.0 y 4.0 del software X en sus nodos administrados. La empresa desea estandarizar la versión 4.0, lo que significa que las instancias que ejecutan las versiones 2.0 y 3.0 no son conformes. Puede utilizar la operación PutComplianceItems de la API para indicar qué nodos administrados ejecutan versiones anteriores del software X. Solo puede asignar los metadatos de conformidad mediante la AWS CLI, AWS Tools for Windows PowerShell o los SDK. El siguiente comando de muestra de la CLI asigna metadatos de conformidad a una instancia administrada y especifica el tipo de conformidad en el formato requerido `Custom:`. Reemplace cada *`example resource placeholder`* con su propia información.

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**
>
> El parámetro `ResourceType` solo es compatible con `ManagedInstance`. Si agrega conformidad personalizada a un dispositivo de núcleo de AWS IoT Greengrass administrado, debe especificar un `ResourceType` de `ManagedInstance`.

Los administradores de conformidad pueden ver resúmenes o crear informes sobre qué nodos son conformes o no lo son. Puede asignar un máximo de 10 tipos de conformidad personalizados distintos a un nodo administrado.

Para ver un ejemplo de cómo crear un tipo de conformidad personalizada y ver los datos de conformidad, consulte Asignar metadatos de conformidad personalizados mediante la AWS CLI.

Visualización de los datos de conformidad actuales

En esta sección, se describe cómo ver los datos de conformidad en la consola de Systems Manager y mediante la AWS CLI. Para obtener información sobre cómo ver el seguimiento de cambios y el historial de cumplimiento de los parches y las asociaciones, consulte Visualización del seguimiento de cambios y del historial de Configuration Compliance.

Temas

- Visualización de los datos de conformidad actuales (consola)
- Visualización de los datos de conformidad actuales (AWS CLI)

Visualización de los datos de conformidad actuales (consola)

Utilice el siguiente procedimiento para ver los datos de conformidad en la consola de Systems Manager.

Para ver los informes de conformidad actuales en la consola de Systems Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Compliance.
3. En la sección Compliance dashboard filtering (Filtrado de paneles de conformidad), elija una opción para filtrar los datos de conformidad. La sección Compliance resources summary (Resumen de recursos de conformidad) muestra recuentos de datos de conformidad según el filtro que eligió.
4. Para obtener más información sobre un recurso, desplácese hacia abajo hasta la sección Details overview for resources (Información general de los detalles de los recursos) y elija el ID de un nodo administrado.
5. En la página de detalles Instance ID (ID de instancia) o Name (Nombre), elija la pestaña Configuration compliance (Conformidad de configuración) para ver un informe detallado de conformidad de configuración del nodo administrado.

> **Note**
>
> Para obtener información sobre cómo solucionar los problemas de conformidad, consulte Solución de problemas de conformidad con EventBridge.

Visualización de los datos de conformidad actuales (AWS CLI)

Puede ver resúmenes de los datos de conformidad para la aplicación de parches, las asociaciones y los tipos de conformidad personalizados en la AWS CLI por medio de los siguientes comandos de la AWS CLI.

list-compliance-summaries

devuelve el recuento de resumen de los estados de asociación que son conformes y no conformes según el filtro que especifique. (API: ListComplianceSummaries)

list-resource-compliance-summaries

Devuelve un recuento de resumen de nivel de recurso. El resumen incluye información sobre los estados conformes y no conformes, así como recuentos detallados de gravedad de elemento de conformidad, según los criterios de filtro que especifique. (API: ListResourceComplianceSummaries)

Puede ver los datos de conformidad adicionales para la aplicación de parches por medio de los siguientes comandos de la AWS CLI.

describe-patch-group-state

muestra el estado de conformidad de parches agregados de alto nivel correspondiente a un grupo de parches. (API: DescribePatchGroupState)

describe-instance-patch-states-for-patch-group

devuelve el estado de parche de alto nivel de las instancias en el grupo de parches especificados. (API: DescribeInstancePatchStatesForPatchGroup)

> **Note**
>
> Para ver una ilustración de cómo configurar la aplicación de parches y los detalles de conformidad de los parches; mediante la AWS CLI, consulte Tutorial: implementación de revisiones en un entorno de servidores mediante la AWS CLI.

Visualización del seguimiento de cambios y del historial de Configuration Compliance

Systems Manager Compliance muestra datos de conformidad actuales sobre la aplicación de revisiones y las asociaciones para los nodos administrados. Puede ver el seguimiento de cambios y el historial de conformidad de la aplicación de parches y las asociaciones mediante AWS Config. AWS Config proporciona una vista detallada de la configuración de los recursos de AWS de su Cuenta de AWS. Esto incluye cómo se relacionan los recursos entre sí y cómo se han configurado en el pasado, para que pueda ver cómo las configuraciones y las relaciones cambian a lo largo del tiempo. Para ver el seguimiento de cambios y el historial de conformidad de la aplicación de parches y las asociaciones, debe habilitar los siguientes recursos en AWS Config:

- `SSM:PatchCompliance`

- `SSM:AssociationCompliance`

Para obtener información sobre cómo elegir y configurar estos recursos específicos en AWS Config, consulte Selección de los recursos que debe registrar AWS Config en la Guía del desarrollador de AWS Config.

> **Note**
>
> Para obtener información sobre precios de AWS Config, consulte precios.

Eliminación de una sincronización de datos de recursos para Compliance

Si ya no desea utilizar AWS Systems Manager Compliance para ver los datos de conformidad, se recomienda que elimine las sincronizaciones de datos de recursos utilizadas para la recopilación de datos de Compliance.

Para eliminar una sincronización de datos de recursos de Compliance

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija Account management (Administración de cuentas) y, a continuación, Resource data sync (Sincronizaciones de datos de recursos).
4. Elija una sincronización en la lista.

   > **Important**
   >
   > Asegúrese de elegir la sincronización utilizada para Compliance. Systems Manager admite la sincronización de datos de recursos para varias herramientas. Si elige una sincronización incorrecta, podría interrumpir la agregación de datos para Systems Manager Explorer o Systems Manager Inventory.

5. Elija Eliminar.
6. Elimine el bucket de Amazon Simple Storage Service (Amazon S3) en el que se almacenaron los datos. Para obtener información acerca de cómo se elimina un bucket de S3, consulte Eliminación de un bucket.

Solución de problemas de conformidad con EventBridge

Puede solucionar rápidamente los problemas de cumplimiento de revisiones y asociaciones mediante Run Command, una herramienta de AWS Systems Manager. Puede dirigirse a una instancia o a ID o etiquetas de dispositivos de núcleo de AWS IoT Greengrass y ejecutar el documento `AWS-RunPatchBaseline` o el documento `AWS-RefreshAssociation`. Si actualizar la asociación o volver a ejecutar la base de referencia de revisiones no soluciona el problema de conformidad, debe investigar sus asociaciones, bases de referencia de revisiones o configuraciones de instancias para comprender por qué las operaciones de Run Command no resolvieron el problema.

Para obtener más información sobre la aplicación de parches, consulte AWS Systems Manager Patch Manager y Documento de comandos SSM para aplicar revisiones: `AWS-RunPatchBaseline`.

Para obtener más información sobre las asociaciones, consulte Trabajo con asociaciones en Systems Manager.

Para obtener más información sobre cómo ejecutar un comando, consulte AWS Systems Manager Run Command.

Especificación de Compliance como el destino de un evento de EventBridge

También puede configurar Amazon EventBridge para que realice una acción en respuesta a los eventos de Systems Manager Compliance. Por ejemplo, si uno o varios nodos administrados no pueden instalar actualizaciones de revisiones críticas o ejecutar una asociación que instala software antivirus, puede configurar EventBridge para ejecutar el documento `AWS-RunPatchBaseline` o el documento `AWS-RefreshAssocation` cuando se produzca el evento de Compliance.

Utilice el siguiente procedimiento para configurar Compliance como el destino de un evento de EventBridge.

Para configurar Compliance como el destino de un evento de EventBridge (consola)

1. Abra la consola de Amazon EventBridge en https://console.aws.amazon.com/events/.
2. En el panel de navegación, seleccione Reglas.
3. Elija Creación de regla.
4. Escriba un nombre y una descripción para la regla.

Una regla no puede tener el mismo nombre que otra regla de la misma Región de AWS y del mismo bus de eventos.

5. En Bus de eventos, seleccione el bus de eventos que desea asociar a esta regla. Si desea que esta regla responda a eventos coincidentes procedentes de su propia Cuenta de AWS, seleccione default (predeterminado). Cuando un Servicio de AWS en su cuenta emite un evento, siempre va al bus de eventos predeterminado de su cuenta.
6. En Tipo de regla, elija Regla con un patrón de evento.
7. Elija Siguiente.
8. En Origen del evento, elija Eventos de AWS o eventos de socios de EventBridge.
9. En la sección Event pattern (Patrón de eventos), elija Event pattern form (Formulario de patrón de eventos).
10. Para Event source (origen de eventos), elija AWSservices (servicios).
11. En AWS service (Servicio de), elija Systems Manager.
12. En el campo Event Type (Tipo de evento), elija Configuration Compliance (Compliance de configuración).
13. En Specific detail type(s) (Tipos de detalle específicos), elija Configuration Compliance State Change (Cambio de estado en la conformidad de la configuración).
14. Elija Siguiente.
15. En Tipos de destino, seleccione Servicio de AWS.
16. En Select a target (Seleccione un destino), elija Systems Manager Run Command.
17. En la lista Document (Documento), elija un documento de Systems Manager (documento de SSM) que se ejecutará cuando se invoque el objetivo. Por ejemplo, elija `AWS-RunPatchBaseline` en el caso de un evento de parche no conforme o elija `AWS-RefreshAssociation` en el caso de un evento de asociación no conforme.
18. Especifique la información del resto de los campos y los parámetros.

> **Note**
>
> Los campos y los parámetros obligatorios tienen un asterisco (*) junto a su nombre. Para crear un destino, debe especificar un valor para cada uno de los parámetros o campos obligatorios. Si no lo hace, el sistema crea la regla, pero esta no se ejecuta.

19. Elija Siguiente.

20. (Opcional) Introduzca una o varias etiquetas para la regla. Para obtener más información, consulte Etiquetado de los recursos de Amazon EventBridge en la Guía del usuario de Amazon EventBridge.
21. Elija Siguiente.
22. Revise los detalles de la regla y seleccione Creación de regla.

Asignar metadatos de conformidad personalizados mediante la AWS CLI

El siguiente procedimiento lo guía a través del proceso de utilizar la AWS Command Line Interface (AWS CLI) para el llamado a la operación PutComplianceItems de la API de AWS Systems Manager para asignar los metadatos de conformidad personalizados a un recurso. También puede utilizar esta operación de la API para asignar manualmente metadatos de conformidad de revisiones o asociaciones a un nodo administrado, tal como se muestra en la siguiente explicación. Para obtener más información sobre la conformidad personalizada, consulte Acerca de la conformidad personalizada.

Para asignar metadatos de conformidad personalizados a una instancia administrada (AWS CLI)

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.
2. Ejecute el siguiente comando para asignar metadatos de conformidad personalizados a un nodo administrado. Reemplace cada *example resource placeholder* con su propia información. El parámetro `ResourceType` solo admite un valor de `ManagedInstance`. Especifique este valor incluso si asigna metadatos de conformidad personalizados a un dispositivo de núcleo de AWS IoT Greengrass administrado.

   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. Repita el paso anterior para asignar metadatos de conformidad personalizados adicionales a uno o más nodos. También puede asignar manualmente los metadatos de conformidad de revisión o asociación a los nodos administrados mediante los comandos siguientes:

   Metadatos de conformidad de asociación

   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
   ```

   Metadatos de conformidad de parche

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. Ejecute el siguiente comando para ver una lista de los elementos de conformidad específicos de un nodo administrado. Utilice filtros para desglosar los datos de conformidad específicos.

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
```

En los siguientes ejemplos se muestra cómo utilizar este comando con filtros.

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. Ejecute el siguiente comando para ver un resumen de los estados de conformidad. Utilice filtros para desglosar los datos de conformidad específicos.

```
aws ssm list-resource-compliance-summaries --filters One or more filters.
```

En los siguientes ejemplos se muestra cómo utilizar este comando con filtros.

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. Ejecute el siguiente comando para ver un recuento de resumen de los recursos conformes y no conformes correspondientes a un tipo de conformidad. Utilice filtros para desglosar los datos de conformidad específicos.

```
aws ssm list-compliance-summaries --filters One or more filters.
```

En los siguientes ejemplos se muestra cómo utilizar este comando con filtros.

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 Distributor

Distributor, una herramienta de AWS Systems Manager, lo ayuda a empaquetar y publicar software en nodos administrados de AWS Systems Manager. Puede empaquetar y publicar su propio software o utilizar Distributor para buscar y publicar paquetes de software de agente proporcionados por AWS, como AmazonCloudWatchAgent, o paquetes de terceros como Trend Micro.La publicación de un paquete anuncia versiones específicas del documento del paquete en nodos administrados que identifica mediante ID de nodo, ID de Cuenta de AWS, etiquetas o una Región de AWS. Para comenzar a utilizar Distributor, abra la consola de Systems Manager. En el panel de navegación, elija Distributor.

Después de crear un paquete en Distributor, puede instalar el paquete de una de las siguientes maneras:

- Una vez mediante AWS Systems Manager Run Command
- De forma programada mediante AWS Systems Manager State Manager

> **Important**
>
> Los paquetes distribuidos por vendedores externos no son gestionados por AWS y son publicados por el vendedor del paquete. Le recomendamos ejercer su propia diligencia debida adicional para garantizar la conformidad de los controles de seguridad internos. La seguridad es una responsabilidad compartida entre AWS y usted. Se describe como el modelo de responsabilidad compartida. Para obtener más información, consulte el Modelo de responsabilidad compartida.

¿Cómo puede Distributor beneficiar a mi organización?

Distributor ofrece las ventajas siguientes:

- Un paquete, muchas plataformas

  Al crear un paquete en Distributor, el sistema crea un documento de AWS Systems Manager (documento de SSM). Puede adjuntar archivos .zip a este documento. Al ejecutar Distributor, el sistema procesa las instrucciones del documento de SSM e instala el paquete de software en el archivo .zip de los destinos especificados. Distributor admite varios sistemas operativos, incluidos Windows, Ubuntu Server, Debian Server y Red Hat Enterprise Linux. Para obtener más

información acerca de las plataformas admitidas, consulte Plataformas de paquetes y arquitecturas admitidas.

- Control de acceso de paquetes en los grupos de instancias administradas

  Puede utilizar Run Command o State Manager para controlar cuáles de los nodos administrados obtienen un paquete y qué versión de ese paquete. Run Command y State Manager son herramientas de AWS Systems Manager. Los nodos administrados se pueden agrupar por ID de instancia o dispositivo, números de Cuenta de AWS, etiquetas o Regiones de AWS. Puede utilizar las asociaciones de State Manager para ofrecer diferentes versiones de un paquete a diferentes grupos de instancias.

- Muchos paquetes de agente de AWS incluidos y listos para usar

  Distributor incluye muchos paquetes de agente de AWS que están listos para implementar en los nodos administrados. Busque los paquetes que publicó Distributor `Packages` en la página de lista `Amazon`. Entre los ejemplos se incluyen `AmazonCloudWatchAgent` y `AWSPVDriver`.

- Automatizar la implementación

  Para mantener su entorno actual, utilice State Manager para programar paquetes para la implementación automática en los nodos de destino cuando dichos nodos se lancen por primera vez.

¿Quién debe utilizar Distributor?

- Cualquier cliente de AWS que desee crear paquetes de software nuevos o implementar paquetes existentes, incluidos paquetes publicados de AWS, en varios nodos administrados de Systems Manager al mismo tiempo.
- Los desarrolladores de software que creen paquetes de software.
- Los administradores que sean responsables de mantener los nodos administrados de Systems Manager actualizados con la versión más reciente de los paquetes de software.

¿Cuáles son las características de Distributor?

- Implementación de paquetes en las instancias de Windows y Linux

  Con Distributor, puede implementar paquetes de software en instancias de Amazon Elastic Compute Cloud (Amazon EC2) y dispositivos de núcleo de AWS IoT Greengrass para Linux y

Windows Server. Para ver una lista de los tipos de sistemas operativos de instancia admitidos, consulte the section called “Plataformas de paquetes y arquitecturas admitidas”.

> **Note**
>
> Distributor no es compatible con el sistema operativo macOS.

- Implementar paquetes una vez o mediante programación automatizada

  Puede elegir implementar paquetes una vez, en una programación periódica o cuando la versión del paquete predeterminado cambie a una versión diferente.
- Reinstale completamente los paquetes o realice actualizaciones in situ

  Para instalar una nueva versión del paquete, puede desinstalar completamente la versión actual e instalar una nueva en su lugar, o solo actualizar la versión actual con componentes nuevos y actualizados, según el script de actualización que proporcione. La aplicación del paquete no está disponible durante una reinstalación, pero puede permanecer disponible durante una actualización in situ. Las actualizaciones in situ son especialmente útiles para aplicaciones de supervisión de seguridad u otros casos en los que necesita evitar el tiempo de inactividad de las aplicaciones.
- Acceso de la consola, CLI, PowerShell y SDK a las capacidades de Distributor

  Puede trabajar con Distributor mediante la consola de Systems Manager, AWS Command Line Interface (AWS CLI), Herramientas de AWS para PowerShell o el AWS SDK de su elección.
- Control de acceso de IAM

  Mediante el uso de políticas (IAM) de AWS Identity and Access Management, puede controlar qué miembros de su organización pueden crear, actualizar, implementar o eliminar paquetes o versiones de paquete. Por ejemplo, es posible que desee dar permisos de administrador para implementar paquetes, pero no para cambiar paquetes o crear nuevas versiones de paquete.
- Compatibilidad con la capacidad de registro y la auditoría

  Puede auditar y registrar las acciones de usuarios de Distributor en su Cuenta de AWS a través de la integración con otros Servicios de AWS. Para obtener más información, consulte Auditoría y registro de la actividad de Distributor.

¿Qué es un paquete en Distributor?

Un paquete es una colección de recursos de software instalable o activos que incluye lo siguiente.

- Un archivo .zip de software por plataforma de sistema operativo de destino. Cada archivo .zip debe incluir lo siguiente.
  - Un script de install y un script de uninstall. Los nodos administrados basados en Windows Server requieren scripts de PowerShell (scripts denominados `install.ps1` y `uninstall.ps1`). Los nodos administrados basados en Linux requieren scripts de shell (scripts denominados `install.sh` y `uninstall.sh`). AWS Systems Manager SSM Agent lee y lleva a cabo las instrucciones de los scripts install y uninstall.
  - Un archivo ejecutable. SSM Agent debe encontrar este archivo para instalar el paquete en los nodos administrados de destino.
- Un archivo de manifiesto con formato JSON que describe el contenido del paquete. El manifiesto no se incluye en el archivo .zip, pero se almacena en el mismo bucket de Amazon Simple Storage Service (Amazon S3) que los archivos .zip que conforman el paquete. El manifiesto identifica la versión del paquete y asigna los archivos .zip en el paquete a atributos del nodo administrado de destino, como, por ejemplo, la versión del sistema operativo o la arquitectura. Para obtener más información sobre cómo crear el manifiesto, consulte Paso 2: crear el manifiesto del paquete JSON.

Cuando elija Creación del paquetesimple en la Distributor consola, Distributor genera la instalación y los scripts de desinstalación, resúmenes de archivo y el paquete JSON manifiesto para usted, en función del nombre de archivo del software ejecutable y las plataformas y arquitecturas de destino.

Plataformas de paquetes y arquitecturas admitidas

Puede utilizar Distributor para publicar paquetes en las siguientes plataformas de nodos administrados de Systems Manager. Un valor de versión debe coincidir con la versión de lanzamiento exacta del sistema operativo Amazon Machine Image (AMI) de destino. Para obtener más información sobre cómo determinar esta versión, consulte el paso 4 de Paso 2: crear el manifiesto del paquete JSON.

> **Note**
>
> Systems Manager no admite todos los siguientes sistemas operativos para dispositivos de núcleo de AWS IoT Greengrass. Para obtener más información, consulte Configuración de dispositivos de núcleo de AWS IoT Greengrass en la Guía para desarrolladores de AWS IoT Greengrass Version 2.

Plataforma	Valor de código en el archivo de manifiesto	Arquitectura compatible
AlmaLinux	`almalinux`	x86_64 ARM64
Amazon Linux 1, Amazon Linux 2 y Amazon Linux 2023	`amazon`	x86_64 o x86 ARM64 (Amazon Linux 2 y AL2023, tipos de instancia de A1)
CentOS	`centos`	x86_64 o x86
Debian Server	`debian`	x86_64 o x86
openSUSE	`opensuse`	x86_64 o x86
openSUSE Leap	`opensuseleap`	x86_64 o x86
Oracle Linux	`oracle`	x86_64
Red Hat Enterprise Linux (RHEL)	`redhat`	x86_64 o x86 ARM64 (RHEL 7.6 y posteriores, tipos de instancias A1)
Rocky Linux	`rocky`	x86_64 ARM64
SUSE Linux Enterprise Server (SLES)	`suse`	x86_64 o x86
Ubuntu Server	`ubuntu`	x86_64 o x86 ARM64 (Ubuntu Server 16 y posteriores, tipos de instancias A1)

Plataforma	Valor de código en el archivo de manifiesto	Arquitectura compatible
Windows Server	`windows`	x86_64 o x86

Temas

- Cómo configurar Distributor
- Trabajo con paquetes Distributor
- Auditoría y registro de la actividad de Distributor
- Solución de problemas de AWS Systems ManagerDistributor

Cómo configurar Distributor

Antes de utilizar Distributor, una herramienta de AWS Systems Manager, para crear, administrar e implementar paquetes de software, siga los pasos que se indican a continuación.

Cómo completar los requisitos previos de Distributor

Antes de utilizar Distributor, una herramienta de AWS Systems Manager, asegúrese de que el entorno cumpla los siguientes requisitos.

Requisitos previos de Distributor

Requisito	Descripción
SSM Agent	La versión 2.3.274.0 o posterior de AWS Systems Manager SSM Agent debe estar instalada en los nodos administrados en los que desea implementar o desde los que desea eliminar paquetes. Para instalar o actualizar SSM Agent, consulte Uso de SSM Agent.
AWS CLI	(Opcional) Para utilizar AWS Command Line Interface (AWS CLI) en lugar de la consola de Systems Manager para crear y administrar los

Requisito	Descripción
	paquetes, instale la versión más reciente de AWS CLI en el equipo local. Para obtener más información acerca de cómo instalar o actualizar la CLI, consulte Instalación de la AWS Command Line Interface en la Guía del usuario de AWS Command Line Interface.
Herramientas de AWS para PowerShell	(Opcional) Para utilizar Tools for PowerShel l en lugar de la consola de Systems Manager para crear y administrar los paquetes, instale la versión más reciente de Tools for PowerShell en el equipo local. Para obtener más información acerca de cómo se instala o actualiza Tools for PowerShel l, consulte Instalación de AWS Tools for Windows PowerShell o AWS Tools for PowerShell Core en la Guía del usuario de Herramientas de AWS para PowerShell.

> **Note**
>
> Systems Manager no admite la distribución de paquetes a nodos administrados de Oracle Linux mediante el uso de Distributor.

Cómo verificar o crear un perfil de instancias de IAM con permisos de Distributor

De forma predeterminada, AWS Systems Manager no tiene permiso para realizar acciones en sus instancias. Para conceder acceso debe utilizar un perfil de instancias de IAM AWS Identity and Access Management. Un perfil de instancias es un contenedor que pasa información del rol de IAM a una instancia de Amazon Elastic Compute Cloud (Amazon EC2) al momento del lanzamiento. Este requisito se aplica a los permisos para todas las herramientas de Systems Manager, no solo a Distributor.

> **Note**
>
> Al configurar los dispositivos de borde para ejecutar el software AWS IoT Greengrass Core y SSM Agent, especifique un rol de servicio de IAM que permite a Systems Manager realizar acciones en él. No es necesario configurar dispositivos de borde administrados con un perfil de instancia.

Si ya utiliza otras herramientas de Systems Manager, como Run Command y State Manager, ya se ha adjuntado a las instancias un perfil de instancias con los permisos necesarios para Distributor. La forma más sencilla de garantizar que tiene permisos para realizar tareas de Distributor es asociar la política AmazonSSMManagedInstanceCore a su perfil de instancia. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager.

Cómo controlar el acceso de los usuarios a los paquetes

Cuando utiliza las políticas (IAM) de AWS Identity and Access Management, puede controlar quién puede crear, implementar y administrar paquetes. También puede controlar qué operaciones de la API Run Command y State Manager pueden realizar en los nodos administrados. Igual que Distributor, Run Command y State Manager son herramientas de AWS Systems Manager.

Formato de ARN

Los paquetes definidos por el usuario se asocian con los Nombre de recurso de Amazon (ARN) de documento y tienen el siguiente formato.

```
arn:aws:ssm:region:account-id:document/document-name
```

A continuación se muestra un ejemplo.

```
arn:aws:ssm:us-west-1:123456789012:document/ExampleDocumentName
```

Puede utilizar un par de políticas de IAM predeterminadas proporcionadas por AWS (una para los usuarios finales y otra para los administradores) con el fin de conceder permisos para actividades de Distributor. También puede crear sus propias políticas personalizadas de IAM adecuadas para sus requisitos de permisos.

Para obtener más información acerca del uso de variables en políticas de IAM, consulte Elementos de la política de IAM: variables

Para obtener información acerca de cómo crear políticas y adjuntarlas a usuarios o grupos, consulte Creación de políticas de IAM y Adición y eliminación de políticas de IAM en la Guía del usuario de IAM.

Cómo crear o elegir un bucket de Amazon S3 para almacenar paquetes de Distributor

Cuando crea un paquete con el flujo de trabajo Simple en la consola de AWS Systems Manager, elija un bucket de Amazon Simple Storage Service (Amazon S3) existente en el cual Distributor cargue su software. Distributor es una herramienta de AWS Systems Manager. En el flujo de trabajo Avanzado, debe cargar archivos .zip de su software o recursos en un bucket de Amazon S3 antes de comenzar. Si crea un paquete utilizando los flujos de trabajo simple o avanzado en la consola, o mediante la API, debe tener un bucket de Amazon S3 antes de comenzar a crear el paquete. Como parte del proceso de creación de paquetes, Distributor copia el software y los recursos instalables de este bucket a un almacén interno de Systems Manager. Dado que los activos se copian en un almacén interno, puede eliminar o reasignar el bucket de Amazon S3 cuando la creación del paquete ha finalizado.

Para obtener más información acerca de la creación de un bucket, consulte Creación de un bucket en la Guía de introducción a Amazon Simple Storage Service. Para obtener más información acerca de cómo ejecutar un comando de AWS CLI para crear un bucket, consulte mb en la Referencia de los comandos de la AWS CLI.

Trabajo con paquetes Distributor

Puede utilizar la consola de AWS Systems Manager, las herramientas de línea de comandos de AWS (AWS CLI y Herramientas de AWS para PowerShell) y los AWS SDK para agregar, administrar o implementar paquetes en Distributor. Distributor es una herramienta de AWS Systems Manager. Antes de agregar un paquete a Distributor:

- Cree y comprima los recursos instalables.
- (Opcional) Cree un archivo de manifiesto de JSON para el paquete. Esto no es necesario para utilizar el proceso de creación de paquetes Simple en la consola de Distributor. La creación del paquete simple genera un archivo de manifiesto JSON para usted.

  Puede utilizar la consola de AWS Systems Manager o un editor de texto o JSON para crear el archivo de manifiesto.
- Tenga preparado un bucket de Amazon Simple Storage Service (Amazon S3) para almacenar los recursos o el software instalables. Si está utilizando el proceso de creación de paquetes Avanzado, cargue sus recursos al bucket de Amazon S3 antes de comenzar.

> **Note**
>
> Puede eliminar o reasignar este bucket después de finalizar la creación de su paquete, ya que Distributor mueve el contenido del paquete a un bucket de Systems Manager interno como parte del proceso de creación de paquetes.

Los paquetes publicados por AWS ya están empaquetados y preparados para implementarse. Para implementar un paquete publicado por AWS en nodos administrados, consulte Instalar o actualizar paquetes de Distributor.

Puede compartir paquetes de Distributor entre Cuentas de AWS. Cuando utilice un paquete compartido desde otra cuenta en comandos de AWS CLI, utilice el nombre de recurso de Amazon (ARN) del paquete en lugar del nombre del paquete.

Temas

- Ver paquetes en Distributor
- Crear un paquete en Distributor
- Editar permisos del paquete Distributor en la consola
- Editar etiquetas del paquete Distributor en la consola
- Agregar una versión al paquete de Distributor
- Instalar o actualizar paquetes de Distributor
- Desinstalación de un paquete Distributor
- Eliminar un paquete de Distributor

Ver paquetes en Distributor

Para ver los paquetes disponibles para su instalación, puede utilizar la consola de AWS Systems Manager o la herramienta de línea de comandos de AWS que prefiera. Distributor es una herramienta de AWS Systems Manager. Para acceder a Distributor, abra la consola de AWS Systems Manager y elija Distributor en el panel de navegación izquierdo. Verá todos los paquetes disponibles para usted.

En la siguiente sección se describe cómo ver los paquetes de Distributor mediante la herramienta de línea de comandos que prefiera.

Ver los paquetes mediante la línea de comandos

Esta sección contiene información acerca de cómo utilizar la herramienta de línea de comandos que prefiera para ver los paquetes de Distributor con los comandos proporcionados.

Linux & macOS

Para ver los paquetes con la AWS CLI en Linux

- Para ver todos los paquetes, excepto los compartidos, ejecute el siguiente comando.

```
aws ssm list-documents \
    --filters Key=DocumentType,Values=Package
```

- Para ver todos los paquetes de propiedad de Amazon, ejecute el siguiente comando.

```
aws ssm list-documents \
    --filters Key=DocumentType,Values=Package Key=Owner,Values=Amazon
```

- Para ver todos los paquetes de propiedad de terceros, ejecute el siguiente comando.

```
aws ssm list-documents \
    --filters Key=DocumentType,Values=Package Key=Owner,Values=ThirdParty
```

Windows

Para ver los paquetes con la AWS CLI en Windows

- Para ver todos los paquetes, excepto los compartidos, ejecute el siguiente comando.

```
aws ssm list-documents ^
    --filters Key=DocumentType,Values=Package
```

- Para ver todos los paquetes de propiedad de Amazon, ejecute el siguiente comando.

```
aws ssm list-documents ^
    --filters Key=DocumentType,Values=Package Key=Owner,Values=Amazon
```

- Para ver todos los paquetes de propiedad de terceros, ejecute el siguiente comando.

```
aws ssm list-documents ^
```

```
    --filters Key=DocumentType,Values=Package Key=Owner,Values=ThirdParty
```

PowerShell

Para ver paquetes con Tools for PowerShell

- Para ver todos los paquetes, excepto los compartidos, ejecute el siguiente comando.

```
$filter = New-Object Amazon.SimpleSystemsManagement.Model.DocumentKeyValuesFilter
$filter.Key = "DocumentType"
$filter.Values = "Package"

Get-SSMDocumentList `
    -Filters @($filter)
```

- Para ver todos los paquetes de propiedad de Amazon, ejecute el siguiente comando.

```
$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)
```

- Para ver todos los paquetes de propiedad de terceros, ejecute el siguiente comando.

```
$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)
```

Crear un paquete en Distributor

Para crear un paquete, prepare su software o activos instalables, un archivo por plataforma de sistema operativo. Se requiere al menos un archivo para crear un paquete.

A veces distintas plataformas pueden utilizar el mismo archivo, pero todos los archivos que asocie a su paquete deben incluirse en la sección `Files` del manifiesto. Si está creando un paquete utilizando el flujo de trabajo simple en la consola, el manifiesto se genera para usted. El número máximo de archivos que se pueden asociar a un único documento es 20. El tamaño máximo de cada archivo es de 1 GB. Para obtener más información acerca de las plataformas admitidas, consulte Plataformas de paquetes y arquitecturas admitidas.

Al crear un paquete, el sistema crea un nuevo documento de SSM. Este documento le permite implementar el paquete en nodos administrados.

Solo para fines de demostración, hay un paquete de ejemplo, ExamplePackage.zip, que puede descargar desde nuestro sitio web. El paquete de ejemplo incluye un manifiesto JSON previamente completado y tres archivos.zip que contienen instaladores para PowerShell v7.0.0. Los scripts de instalación y desinstalación no contienen comandos válidos. Aunque debe comprimir cada software instalable y scripts en un archivo .zip para crear un paquete en el flujo de trabajo Avanzado, no debe comprimir los recursos instalables en el flujo de trabajo Simple.

Temas

- Crear un paquete mediante el flujo de trabajo simple
- Crear un paquete mediante el flujo de trabajo avanzado

Crear un paquete mediante el flujo de trabajo simple

Esta sección describe cómo crear un paquete en Distributor mediante la elección del flujo de trabajo de creación de paquete Simple en la consola de Distributor. Distributor es una herramienta de AWS Systems Manager. Para crear un paquete, prepare los archivos de los recursos instalables, un archivo por plataforma de sistema operativo. Se requiere al menos un archivo para crear un paquete. El proceso de creación de paquetes simple genera scripts de instalación y desinstalación, resúmenes de archivos y un manifiesto con formato JSON para usted. El flujo de trabajo Simple gestiona el

proceso de cargar y comprimir sus archivos instalables, así como el proceso de crear un nuevo paquete y un documento SSM asociado. Para obtener más información acerca de las plataformas admitidas, consulte Plataformas de paquetes y arquitecturas admitidas.

Cuando se utiliza el método simple para crear un paquete, Distributor crea scripts `install` y `uninstall` en su nombre. Sin embargo, al crear un paquete para una actualización in-situ, debe proporcionar su propio contenido de script de `update` en la pestaña Update script (Script de actualización). Cuando agrega comandos de entrada para un script de `update`, Distributor incluye este script en el paquete .zip que crea para usted, junto con los scripts `uninstall` y `install`.

> **Note**
>
> Utilice la opción de actualización `In-place` para agregar archivos nuevos o actualizados a una instalación de paquete existente sin desconectar la aplicación asociada.

Crear un paquete mediante el flujo de trabajo simple

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Distributor.
3. En la página de Distributor inicio, elija Crear paquete y, a continuación, haga clic en Simple.
4. En la página Añadir paquete, escriba un nombre para su paquete. Los nombres de paquete solo pueden incluir letras, números, puntos, guiones y guiones bajos. El nombre debe ser lo suficientemente genérico como para aplicarse a todas las versiones de los archivos adjuntos del paquete, pero lo suficientemente concreto como para identificar la finalidad del paquete.
5. (Opcional) En Version Name (Nombre de la versión), escriba un nombre de versión. Los nombres de las versiones pueden tener un máximo de 512 caracteres y no pueden contener caracteres especiales.
6. En Location (Ubicación), seleccione un bucket mediante el nombre y el prefijo del bucket o mediante la dirección URL del bucket.
7. En Upload software (Cargar software), elija Add software (Agregar software), y luego navegue para buscar archivos de software instalables con las extensiones `.rpm`, `.msi` o `.deb`. Si el nombre del archivo contiene espacios, se produce un error en la carga. Puede cargar más de un archivo de software en una única acción.

8. Para la plataforma de destino, verifique que la plataforma del sistema operativo de destino que se muestra para cada archivo instalable sea correcta. Si el sistema operativo que se muestra no es correcto, seleccione el sistema operativo correcto en la lista desplegable.

   En el flujo de trabajo de creación de paquetes simple, debido a que carga cada archivo instalable solo una vez, se requieren pasos adicionales para dar una orden Distributor a un único archivo en múltiples sistemas operativos. Por ejemplo, si carga un archivo de software instalable denominado `Logtool_v1.1.1.rpm`, debe cambiar algunos valores predeterminados en el flujo de trabajo simple para el mismo software en Amazon Linux y Ubuntu sistemas operativos. Al apuntar a varias plataformas, realice una de las siguientes acciones.

   - Utilice el flujo de trabajo avanzado en su lugar, comprima cada archivo instalable en un archivo .zip antes de comenzar y cree manualmente el manifiesto para que un archivo instalable pueda ser dirigido a múltiples plataformas o versiones de sistemas operativos. Para obtener más información, consulte Crear un paquete mediante el flujo de trabajo avanzado.
   - Edite manualmente el archivo de manifiesto en el flujo de trabajo simple para que su archivo .zip esté dirigido a múltiples plataformas o versiones del sistema operativo. Para obtener más información acerca de cómo hacerlo, consulte el final del paso 4 en Paso 2: crear el manifiesto del paquete JSON.

9. En Platform version (Versión de la plataforma), verifique que la versión de la plataforma del sistema operativo que se muestra sea **_any**, una versión de lanzamiento principal seguida de una comodín (7.*) o la versión exacta de lanzamiento del sistema operativo a la que desea que se aplique su software. Para obtener más información sobre cómo especificar una versión de plataforma del sistema operativo, consulte el paso 4 en Paso 2: crear el manifiesto del paquete JSON.

10. En Architecture (Arquitectura), elija la arquitectura de procesador correcta para cada archivo instalable de la lista desplegable. Para obtener más información acerca de las arquitecturas de procesadores compatibles, consulte Plataformas de paquetes y arquitecturas admitidas.

11. (Opcional) Expanda los scripts y revise los scripts que Distributor genera para su software instalable.

12. (Opcional) Para proporcionar un script de actualización para su uso con actualizaciones in situ, expanda Scripts, elija la pestaña Update script (Script de actualización) e introduzca los comandos de script de actualización.

    Systems Manager no genera scripts de actualización en su nombre.

13. Para añadir más archivos de software instalable, seleccione Añadir software. De no ser así, vaya al siguiente paso.
14. (Opcional) Expanda el manifiesto, y revise el manifiesto del paquete JSON que Distributor genera para su software instalable. Si cambió alguna información sobre su software desde que comenzó este procedimiento, como la versión de plataforma o la plataforma de destino, elija Generar manifiesto para mostrar el manifiesto actualizado del paquete.

    Puede editar el manifiesto manualmente si desea dirigir un software instalable en más de un sistema operativo, como se describe en el paso 8. Para obtener más información sobre cómo editar el manifiesto, consulte Paso 2: crear el manifiesto del paquete JSON.
15. Elija Create package (Crear paquete).

Espere que Distributor termine de cargar su software y crear el paquete. Distributor muestra el estado de carga para cada archivo instalable. En función del número y el tamaño de los paquetes que se agregan, esto puede tardar unos minutos. Distributor automáticamente le redirige a la página Detalles del paquete para el nuevo paquete, pero puede elegir abrir esta página usted mismo después de que el software se haya cargado. La página Detalles del paquete no muestra toda la información sobre el paquete hasta que Distributor termine el proceso de creación de paquete. Para detener el proceso de carga y creación del paquete, seleccione Cancelar.

Si Distributor no puede cargar cualquiera de los archivos de instalación de software, muestra un mensaje de Upload failed (Carga fallida). Para volver a intentar la carga, elija Reintentar carga. Para obtener más información acerca de cómo solucionar errores en la creación de paquetes, consulte Solución de problemas de AWS Systems ManagerDistributor.

Crear un paquete mediante el flujo de trabajo avanzado

En esta sección podrá descubrir cómo los usuarios avanzados pueden crear un paquete en Distributor después de cargar recursos instalables comprimidos con scripts de instalación y desinstalación, así como un archivo de manifiesto JSON, en un bucket de Amazon S3.

Para crear un paquete, prepare los archivos .zip de los recursos instalables, un archivo .zip por plataforma de sistema operativo. Se requiere al menos un archivo .zip para crear un paquete. A continuación, cree un manifiesto JSON. El manifiesto incluye indicadores a los archivos de código del paquete. Cuando tenga sus archivos de código necesarios agregados a una carpeta o un directorio y el manifiesto se haya rellenado con los valores correctos, cargue el paquete en un bucket de S3.

Hay disponible un paquete de ejemplo, ExamplePackage.zip, para su descarga desde nuestro sitio web. El paquete de ejemplo incluye un manifiesto JSON completo y tres archivos .zip.

Temas

- Paso 1: crear los archivos ZIP
- Paso 2: crear el manifiesto del paquete JSON
- Paso 3: cargar el paquete y el manifiesto en un bucket de Amazon S3
- Paso 4: añadir un paquete a Distributor

Paso 1: crear los archivos ZIP

La base del paquete es al menos un archivo .zip P de recursos instalables o software. Un paquete incluye un archivo .zip por sistema operativo que desee admitir, a menos que un archivo .zip se pueda instalar en varios sistemas operativos. Por ejemplo, las instancias de Red Hat Enterprise Linux y Amazon Linux normalmente pueden ejecutar los mismos archivos ejecutables .RPM, por lo que debe adjuntar un único archivo .zip en el paquete para admitir ambos sistemas operativos.

Archivos necesarios

Se requieren los elementos siguientes en cada archivo .zip:

- Un script de install y un script de uninstall. Los nodos administrados basados en Windows Server requieren scripts de PowerShell (scripts denominados `install.ps1` y `uninstall.ps1`). Los nodos administrados basados en Linux requieren scripts de shell (scripts denominados `install.sh` y `uninstall.sh`). SSM Agent ejecuta las instrucciones de los scripts install y uninstall.

  Por ejemplo, los scripts de instalación podrían ejecutar un instalador (como .rpm o .msi), podrían copiar archivos o establecer opciones de configuración.
- Un archivo ejecutable, paquetes de instalador (.rpm, .deb, .msi, etc.) y otros scripts o archivos de configuración.

Archivos opcionales

El siguiente elemento es opcional en cada archivo .zip:

- Un script update. Proporcionar un script de actualización le permite utilizar la opción `In-place update` para instalar un paquete. Si desea agregar archivos nuevos o actualizados a una

instalación de paquete existente, la opción `In-place update` no desconecta la aplicación de paquete mientras se realiza la actualización. Los nodos administrados basados en Windows Server requieren un script de PowerShell (script denominado `update.ps1`). Los nodos administrados basados en Linux requieren un script de shell (script denominado `update.sh`). SSM Agent ejecuta las instrucciones del script update.

Para obtener más información acerca de la instalación o actualización de paquetes, consulte Instalar o actualizar paquetes de Distributor.

Para ver ejemplos de archivos .zip, incluidos los scripts de muestra install y uninstall, descargue el paquete de ejemplo, ExamplePackage.zip.

Paso 2: crear el manifiesto del paquete JSON

Después de preparar y comprimir sus archivos instalables, cree un manifiesto JSON. Lo siguiente es una plantilla. Las partes de la plantilla de manifiesto se describen en el procedimiento que se describe en esta sección. Puede utilizar un editor de JSON para crear este manifiesto en un archivo independiente. Si lo prefiere, puede crear el manifiesto en la consola de AWS Systems Manager cuando cree un paquete.

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

Para crear un manifiesto del paquete JSON

1. Añada la versión de esquema al manifiesto. En esta versión, la versión de esquema es siempre `2.0`.

   ```
   { "schemaVersion": "2.0",
   ```

2. Añada una versión del paquete definido por el usuario al manifiesto. También es el valor de Version name (Nombre de versión) que ha especificado al agregar el paquete a Distributor. Pasa a formar parte del documento de AWS Systems Manager que crea Distributor al agregar el paquete. También proporciona este valor como una entrada en el documento `AWS-ConfigureAWSPackage` para instalar una versión del paquete que no sea la última. Un valor `version` puede contener letras, números, guiones bajos, guiones y puntos, y tener un máximo de 128 caracteres de longitud. Le recomendamos que utilice una versión del paquete en lenguaje natural para facilitarle a usted y a los demás administradores la especificación de las versiones de paquete exactas durante la implementación. A continuación se muestra un ejemplo.

   ```
   "version": "1.0.1",
   ```

3. (Opcional). Añada un nombre de editor. A continuación se muestra un ejemplo.

```
"publisher": "MyOrganization",
```

4. Añada paquetes. En la sección `"packages"` se describen las plataformas, las versiones de lanzamiento y las arquitecturas compatibles con los archivos .zip en el paquete. Para obtener más información, consulte Plataformas de paquetes y arquitecturas admitidas.

   El valor *`platform-version`* puede ser el valor de comodín `_any`. Utilícelo para indicar que un archivo .zip admite cualquier versión de la plataforma. También puede especificar una versión de lanzamiento principal seguida de un comodín para que se admitan todas las versiones secundarias, como la 7.*. Si elige especificar un valor de *`platform-version`* para una versión específica del sistema operativo, asegúrese de que coincida con la versión de lanzamiento exacta del sistema operativo AMI de destino. A continuación, se muestran los recursos sugeridos para obtener el valor correcto del sistema operativo.

   - En nodos administrados basados en Windows Server, la versión de lanzamiento está disponible como datos de Windows Management Instrumentation (WMI). Puede ejecutar el siguiente comando del símbolo del sistema para obtener información de la versión y, a continuación, analizar los resultados de `version`.

     ```
     wmic OS get /format:list
     ```

   - En un nodo administrado basado en Linux, obtenga la versión escaneando en primer lugar la versión del sistema operativo (el siguiente comando). Busque el valor de `VERSION_ID`.

     ```
     cat /etc/os-release
     ```

     Si esto no da los resultados que necesita, ejecute el siguiente comando para obtener información de la versión LSB del archivo `/etc/lsb-release` y busque el valor de `DISTRIB_RELEASE`.

     ```
     lsb_release -a
     ```

     Si estos métodos fallan, normalmente puede encontrar la versión según la distribución. Por ejemplo, en Debian Server, puede examinar el archivo `/etc/debian_version` o, en Red Hat Enterprise Linux, el archivo `/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"
        }
      }
    }
  }
```

A continuación se muestra un ejemplo. En este ejemplo, la plataforma del sistema operativo es `amazon`, la versión de lanzamiento compatible es `2016.09`, la arquitectura es `x86_64` y el archivo .zip que es compatible con esta plataforma es `test.zip`.

```
{
    "amazon": {
        "2016.09": {
            "x86_64": {
                "file": "test.zip"
            }
        }
    }
},
```

Puede añadir el valor de comodín `_any` para indicar que el paquete es compatible con todas las versiones del elemento principal. Por ejemplo, para indicar que el paquete es compatible con

cualquier versión de lanzamiento de Amazon Linux, la instrucción del paquete será parecida a la que se muestra a continuación. Puede utilizar el comodín `_any` en los niveles de arquitectura o versión para admitir todas las versiones de una plataforma o todas las arquitecturas de una versión, o todas las versiones y todas las arquitecturas de una plataforma.

```
{
    "amazon": {
        "_any": {
            "x86_64": {
                "file": "test.zip"
            }
        }
    }
},
```

En el siguiente ejemplo se agrega `_any` para mostrar que el primer paquete, `data1.zip`, es compatible con todas las arquitecturas de Amazon Linux 2016.09. El segundo paquete, `data2.zip`, es compatible con todas las versiones de Amazon Linux, pero solo para los nodos administrados con la arquitectura `x86_64`. Las versiones `2016.09` y `_any` son entradas bajo `amazon`. Hay una plataforma (Amazon Linux), pero diferentes versiones, arquitecturas y archivos .zip asociados admitidos.

```
{
    "amazon": {
        "2016.09": {
            "_any": {
                "file": "data1.zip"
            }
        },
        "_any": {
            "x86_64": {
                "file": "data2.zip"
            }
        }
    }
}
```

Puede hacer referencia a un archivo .zip más de una vez en la sección `"packages"` del manifiesto, si el archivo .zip es compatible con más de una plataforma. Por ejemplo, si tiene un archivo .zip que admite versiones Red Hat Enterprise Linux 7.x y Amazon Linux, tendría dos

entradas en la sección `"packages"` apuntando al mismo archivo .zip, como se muestra en el siguiente ejemplo.

```
{
    "amazon": {
        "2018.03": {
            "x86_64": {
                "file": "test.zip"
            }
        }
    },
    "redhat": {
        "7.*": {
            "x86_64": {
                "file": "test.zip"
            }
        }
    }
},
```

5. Añada la lista de archivos .zip que forman parte de este paquete en el paso 4. Cada entrada de archivo requiere el nombre de archivo y el valor hash de la suma de comprobación `sha256`. Los valores de suma de comprobación en el manifiesto deben coincidir con el valor hash `sha256` en los recursos comprimidos para evitar que se produzca un error en la instalación de paquetes.

   Para obtener la suma de comprobación exacta de su instalables, puede ejecutar los siguientes comandos. En Linux, ejecute `shasum -a 256` *`file-name`*`.zip` o `openssl dgst -sha256` *`file-name`*`.zip`. En Windows, ejecute el cmdlet `Get-FileHash -Path` *`path-to-.zip-file`* en PowerShell.

   La sección `"files"` del manifiesto incluye una referencia a cada uno de los archivos .zip del paquete.

```
"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. Una vez que haya añadido la información del paquete, guarde y cierre el archivo de manifiesto.

A continuación se muestra un ejemplo de un manifiesto completado. En este ejemplo, tiene un archivo .zip, `NewPackage_LINUX.zip`, que admite más de una plataforma, pero al que se hace referencia en la sección `"files"` solo una vez.

```
{
    "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"
            }
```

```
        }
    }
}
```

Ejemplo de paquete

Hay disponible un paquete de ejemplo, ExamplePackage.zip, para su descarga desde nuestro sitio web. El paquete de ejemplo incluye un manifiesto JSON completo y tres archivos .zip.

Paso 3: cargar el paquete y el manifiesto en un bucket de Amazon S3

Prepare el paquete; para ello, copie o transfiera todos los archivos .zip a una carpeta o un directorio. Un paquete válido requiere el manifiesto que creó en Paso 2: crear el manifiesto del paquete JSON y todos los archivos .zip identificados en la lista del archivo de manifiesto.

Para cargar el paquete y el manifiesto en Amazon S3

1. Copie o mueva todos los archivos de almacenamiento .zip que ha especificado en el manifiesto a una carpeta o un directorio. No comprima la carpeta ni el directorio al que mueve los archivos del archivo.zip y el archivo de manifiesto.
2. Cree un bucket o elija uno existente. Para obtener más información, consulte Creación de un bucket en la Guía de introducción a Amazon Simple Storage Service. Para obtener más información acerca de cómo ejecutar un comando de AWS CLI para crear un bucket, consulte mb en la Referencia de los comandos de la AWS CLI.
3. Cargar la carpeta o el directorio en el bucket. Para obtener más información, consulte Cargar un objeto en un bucket en la Guía de introducción a Amazon Simple Storage Service. Si planea pegar el manifiesto JSON en la consola AWS Systems Manager, no cargue el manifiesto. Para obtener más información acerca de cómo ejecutar un comando de AWS CLI para cargar los archivos en un bucket, consulte mv en la Referencia de los comandos de la AWS CLI.
4. En la página de inicio del bucket, elija la carpeta o el directorio que se ha cargado. Si cargó sus archivos a una subcarpeta en un bucket, asegúrese de anotar la subcarpeta (también conocido como un prefijo). Necesita el prefijo para agregar el paquete a Distributor.

Paso 4: añadir un paquete a Distributor

Puede utilizar la consola de AWS Systems Manager, las herramientas de línea de comandos de AWS (AWS CLI y Herramientas de AWS para PowerShell) o los AWS SDK para agregar un nuevo paquete a Distributor. Cuando agrega un paquete, se agrega un nuevo documento de SSM. El documento le permite implementar el paquete en nodos administrados.

Temas

- Agregar un destino mediante la consola
- Agregar un paquete mediante la AWS CLI

Agregar un destino mediante la consola

Puede utilizar la consola de AWS Systems Manager para crear un paquete. Tenga listo el nombre del bucket al que se ha cargado el paquete en Paso 3: cargar el paquete y el manifiesto en un bucket de Amazon S3.

Para añadir un paquete a Distributor (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Distributor.
3. En la página de Distributor inicio, elija Crear paquete y, a continuación, haga clic en Avanzado.
4. En la página Añadir paquete, escriba un nombre para su paquete. Los nombres de paquete solo pueden incluir letras, números, puntos, guiones y guiones bajos. El nombre debe ser lo suficientemente genérico como para aplicarse a todas las versiones de los archivos adjuntos del paquete, pero lo suficientemente concreto como para identificar la finalidad del paquete.
5. En Version name (Nombre de versión), escriba el valor exacto de la entrada `version` en su archivo de manifiesto.
6. En S3 bucket name (Nombre del bucket de S3), seleccione el nombre del bucket al que se ha cargado sus archivos .zip y manifiesto en the section called “Paso 3: cargar el paquete y el manifiesto en un bucket de Amazon S3”.
7. En S3 key prefix (Prefijo de clave de S3), escriba la subcarpeta del bucket en el que están almacenados el manifiesto y sus archivos .zip.
8. En Manifest (Manifiesto), elija Extract from package (Extraer del paquete) para utilizar un manifiesto que ha cargado al bucket de Amazon S3 con sus archivos .zip.

   (Opcional) Si no subió su manifiesto JSON al bucket de Amazon S3 donde almacenó sus archivos .zip, elija New manifest (Nuevo manifiesto). Puede crear o pegar todo el manifiesto en el campo de edición de JSON. Para obtener más información sobre cómo crear el manifiesto JSON, consulte Paso 2: crear el manifiesto del paquete JSON.
9. Cuando haya terminado con el manifiesto, elija Crear paquete.

10. Espere a que Distributor cree el paquete desde su archivos .zip y manifiesto. En función del número y el tamaño de los paquetes que se añaden, esto puede tardar unos minutos. Distributor automáticamente le redirige a la página dedetalles del paquete pero puede elegir abrir esta página por usted mismo después de que el software se haya cargado. La página Detalles del paquete no muestra toda la información sobre el paquete hasta que Distributor termine el proceso de creación de paquete. Para detener el proceso de carga y creación del paquete, seleccione Cancelar.

Agregar un paquete mediante la AWS CLI

Puede utilizar la AWS CLI para crear un paquete. Tenga la URL lista desde el bucket en el que haya cargado el paquete en Paso 3: cargar el paquete y el manifiesto en un bucket de Amazon S3.

Para agregar un paquete a Amazon S3 mediante la AWS CLI

1. Para usar la AWS CLI para crear un paquete, ejecute el siguiente comando, reemplazando *package-name* por el nombre del paquete y *path to manifest-file* por la ruta del archivo de manifiesto JSON. amzn-s3-demo-bucket es la URL del bucket Amazon S3 donde está almacenado todo el paquete. Al ejecutar el comando create-document en Distributor, debe especificar el valor `Package` para `--document-type`.

   Si no agregó el archivo de manifiesto al bucket de Amazon S3, el valor del parámetro `--content` es la ruta del archivo al archivo de manifiesto JSON.

   ```
   aws ssm create-document \
       --name "package-name" \
       --content file://path-to-manifest-file \
       --attachments Key="SourceUrl",Values="amzn-s3-demo-bucket" \
       --version-name version-value-from-manifest \
       --document-type Package
   ```

   A continuación se muestra un ejemplo.

   ```
   aws ssm create-document \
       --name "ExamplePackage" \
       --content file://path-to-manifest-file \
       --attachments Key="SourceUrl",Values="https://s3.amazonaws.com/amzn-s3-demo-
   bucket/ExamplePackage" \
       --version-name 1.0.1 \
   ```

```
    --document-type Package
```

2. Compruebe que el paquete se añadió y que muestra el manifiesto de paquete mediante la ejecución del siguiente comando, sustituyendo *package-name* por el nombre de su paquete. Para obtener una versión específica del documento (no la misma que la versión de un paquete), puede agregar el parámetro `--document-version`.

```
aws ssm get-document \
    --name "package-name"
```

Para obtener más información acerca de otras opciones que puede utilizar con el comando create-document, consulte create-document en la sección sobre AWS Systems Manager de la Referencia de comandos de la AWS CLI. Para obtener más información sobre otras opciones que puede usar con el comando get-document, consulte get-document.

Editar permisos del paquete Distributor en la consola

Una vez que haya agregado un paquete a Distributor, una herramienta de AWS Systems Manager, puede editar los permisos del paquete en la consola de Systems Manager. Puede agregar otras Cuentas de AWS a unos permisos de paquete. Los paquetes solo se pueden compartir con otras cuentas en la misma Región de AWS. No se admite el uso compartido entre regiones. De forma predeterminada, los paquetes se establecen en Private (Privado), lo que significa que solo quienes tienen acceso a la Cuenta de AWS del creador del paquete pueden ver la información del paquete y actualizar o eliminar el paquete. Si los permisos de Private (Privado) son aceptables, puede omitir este procedimiento.

> **Note**
>
> Puede actualizar los permisos de los paquetes que se comparten con 20 cuentas o menos.

Para editar permisos del paquete mediante la consola

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Distributor.
3. En la página Packages (Paquetes), elija el paquete del que desea editar los permisos.

4. En la pestaña Packages details (Detalles del paquete), elija Edit permisos (Editar permisos) para cambiar los permisos.
5. En Edit permissions (Editar permisos), elija Shared with specific accounts (Compartido con cuentas específicas).
6. En Shared with specific accounts (Compartido con cuentas específicas), agregue números de Cuenta de AWS, uno a la vez. Cuando haya terminado, elija Save.

Editar etiquetas del paquete Distributor en la consola

Una vez que haya agregado un paquete a Distributor, una herramienta de AWS Systems Manager, puede editar las etiquetas del paquete en la consola de Systems Manager. Estas etiquetas se aplican al paquete y no están conectadas a etiquetas en los nodos administrados en los que desea implementar el paquete. Las etiquetas son pares de claves y valores que distinguen entre mayúsculas y minúsculas y que permiten agrupar y filtrar los paquetes por criterios que son pertinentes para la organización. Si no desea agregar etiquetas, está listo para instalar el paquete o agregar una nueva versión.

Para editar etiquetas del paquete en la consola

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Distributor.
3. En la página Packages (Paquetes), elija el paquete del que desea editar las etiquetas.
4. En la pestaña Package details (Detalles del paquete), en Tags (Etiquetas), elija Edit (Editar).
5. En Add tags (Agregar etiquetas), ingrese una clave de etiqueta o un par de clave de etiqueta y valor, y luego elija Add (Agregar). Repita si desea añadir varias etiquetas. Para eliminar etiquetas, elija X en la etiqueta de la parte inferior de la ventana.
6. Cuando haya terminado de agregar etiquetas a su paquete, elija Save (Guardar).

Agregar una versión al paquete de Distributor

Para agregar una versión del paquete, cree un paquete y, a continuación, utilice Distributor para agregar una versión del paquete mediante la incorporación de una entrada en el documento de AWS Systems Manager (SSM) que ya existe para las versiones anteriores. Distributor es una herramientas de AWS Systems Manager. Para ahorrar tiempo, actualice el manifiesto de una versión anterior del paquete, cambie el valor de la entrada `version` en el manifiesto (por ejemplo, de `Test_1.0`

a `Test_2.0`) y guárdelo como el manifiesto de la nueva versión. El flujo de trabajo simple Añadir versión en la Distributor consola actualiza el archivo de manifiesto por usted.

Una nueva versión del paquete puede:

- Sustituya al menos uno de los archivos instalables asociados a la versión actual.
- Añada nuevos archivos instalables para admitir plataformas adicionales.
- Elimine archivos para cesar la compatibilidad con plataformas específicas.

Una versión más reciente puede utilizar el mismo bucket de Amazon Simple Storage Service (Amazon S3), pero debe tener una URL con otro nombre de archivo mostrado al final. Puede utilizar la consola de Systems Manager o la AWS Command Line Interface (AWS CLI) para agregar la nueva versión. Cargar un archivo instalable con el nombre exacto de un archivo instalable existente en el bucket de Amazon S3 sobrescribe el archivo existente. No se copian archivos instalables de la versión anterior a la nueva versión; debe cargar archivos instalables de la versión anterior para que formen parte de una nueva versión. Una vez que Distributor haya terminado de crear su nueva versión del paquete, puede eliminar o reutilizar el bucket de Amazon S3, ya que Distributor copia su software en un bucket de Systems Manager interno como parte del proceso de control de versiones.

> **Note**
>
> Cada paquete está limitado a un máximo de 25 versiones. Puede eliminar las versiones que ya no sean necesarias.

Temas

- Añadir una versión del paquete mediante la consola
- Añadir una versión del paquete mediante la AWS CLI

Añadir una versión del paquete mediante la consola

Antes de realizar estos pasos, siga las instrucciones de Crear un paquete en Distributor para crear un nuevo paquete de la versión. A continuación, utilice la consola de Systems Manager para agregar una nueva versión del paquete a Distributor.

Añadir una versión de paquete mediante el flujo de trabajo simple

Para añadir una versión del paquete mediante el flujo de trabajo Simple, prepare archivos instalables actualizados o añade archivos instalables para admitir más plataformas y arquitecturas. A continuación, utiliceDistributor para cargar archivos instalables nuevos y actualizados y añada una versión del paquete. El flujo de trabajo simplificado Añadir versión en la Distributor consola actualiza el archivo de manifiesto y documento de SSM asociados por usted.

Para añadir una versión de paquete mediante el flujo de trabajo simple

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Distributor.
3. En la página de inicio de Distributor, elija el paquete en el que desee añadir otra versión.
4. En la página Añadir versión, seleccione simple.
5. En Version Name (Nombre de la versión), escriba un nombre de versión. El nombre de la versión para la nueva versión debe ser diferente de los nombres de versiones anteriores. Los nombres de las versiones pueden tener un máximo de 512 caracteres y no pueden contener caracteres especiales.
6. Para nombre de bucket de S3, elija un bucket de S3 existente de la lista. Este puede ser el mismo bucket que utilizó para almacenar archivos instalables para versiones anteriores, pero los nombres de los archivos instalables deben ser diferentes para evitar sobrescribir los archivos instalables existentes en el bucket.
7. En S3 key prefix (Prefijo de clave de S3), escriba la subcarpeta del bucket donde se almacenan sus recursos instalables.
8. En Upload software (Cargar software), diríjase a los archivos de software instalable que desea adjuntar a la nueva versión. Los archivos instalables de versiones existentes no se copian automáticamente a una nueva versión; debe cargar cualquier archivo instalable de versiones anteriores del paquete si desea que alguno de los mismos archivos instalables forme parte de la nueva versión. Puede cargar más de un archivo de software en una única acción.
9. Para la plataforma de destino, verifique que la plataforma del sistema operativo de destino que se muestra para cada archivo instalable sea correcta. Si el sistema operativo que se muestra no es correcto, seleccione el sistema operativo correcto en la lista desplegable.

   En el flujo de trabajo simple, ya que carga cada archivo una sola vez, se requieren pasos adicionales para dirigirse a un único archivo en varios sistemas operativos. Por ejemplo, si carga

un archivo de software instalable denominado `Logtool_v1.1.1.rpm`, debe cambiar algunos valores predeterminados en el flujo de trabajo simple para dirigir Distributor al mismo software en los sistemas operativos Amazon Linux y Ubuntu. Puede elegir una de las opciones siguientes para evitar esta limitación.

- Utilice el flujo de trabajo avanzado en su lugar, comprima cada archivo instalable en un archivo .zip antes de comenzar, y cree manualmente el manifiesto para que un archivo instalable pueda ser dirigido a varias plataformas o versiones del sistema operativo. Para obtener más información, consulte Añadir una versión de paquete mediante el flujo de trabajo avanzado.
- Edite manualmente el archivo de manifiesto en el flujo de trabajo simple para que su archivo .zip esté dirigido a múltiples plataformas o versiones del sistema operativo. Para obtener más información acerca de cómo hacerlo, consulte el final del paso 4 en Paso 2: crear el manifiesto del paquete JSON.

10. En Platform version (Versión de la plataforma), verifique que la versión de la plataforma del sistema operativo que se muestra sea **_any**, una versión de lanzamiento principal seguida de una comodín (7.*) o la versión exacta de lanzamiento del sistema operativo a la que desea que se aplique su software. Para obtener más información sobre cómo especificar una versión de la plataforma, consulte el paso 4 en Paso 2: crear el manifiesto del paquete JSON.
11. Para Arquitectura, elija la arquitectura de procesador correcta para cada archivo instalable de la lista desplegable. Para obtener más información acerca de las arquitecturas compatibles, consulte Plataformas de paquetes y arquitecturas admitidas.
12. (Opcional) Expanda los scripts y revise los scripts de instalación y desinstalación que Distributor genera para su software instalable.
13. Para añadir más archivos de software instalable a la nueva versión, seleccione Añadir software. De no ser así, vaya al siguiente paso.
14. (Opcional) Expanda el manifiesto, y revise el manifiesto del paquete JSON que Distributor genera para su software instalable. Si cambió cualquier información sobre su software instalable desde que comenzó este procedimiento, como la versión de plataforma o la plataforma de destino, elija Generar manifiesto para mostrar el manifiesto actualizado del paquete.

    Puede editar el manifiesto manualmente si desea dirigirlo a un software instalable en más de un sistema operativo, como se describe en el paso 9. Para obtener más información sobre cómo editar el manifiesto, consulte Paso 2: crear el manifiesto del paquete JSON.
15. Cuando haya terminado de añadir software y revisar la plataforma de destino, la versión y los datos de arquitectura, seleccione Añadir versión.

16. Espere Distributor para terminar la carga de su software y crear la nueva versión del paquete. Distributor muestra el estado de carga para cada archivo instalable. En función del número y el tamaño de los paquetes que se agreguen, esto puede tardar unos minutos. Distributor automáticamente le redirige a la página de detalles del paquete,pero puede elegir abrir esta página por sí mismo después de que el software se hayan cargado. La página Detalles del paquete no muestra toda la información sobre el paquete hasta que Distributor termina de crear la nueva versión del paquete. Para detener la versión del paquete de carga y creación, elija Detener carga.
17. Si Distributor no puede cargar cualquiera de los archivos de instalación de software, muestra un mensaje de Upload failed (Carga fallida). Para volver a intentar la carga, elija Reintentar carga. Para obtener más información sobre cómo solucionar errores de creación de versión del paquete, consulte Solución de problemas de AWS Systems ManagerDistributor.
18. Cuando Distributor haya terminado de crear la nueva versión del paquete, en la página de Detalles del paquete, en la pestaña Versiones, vea la nueva versión en la lista de versiones de paquete disponibles. Establezca una versión predeterminada del paquete; para ello, elija una versión y, a continuación, elija Set default version (Establecer versión predeterminada).

    Si no establece una versión predeterminada, la versión del paquete más reciente es la versión predeterminada.

Añadir una versión de paquete mediante el flujo de trabajo avanzado

Para añadir una versión del paquete, cree un paquete, y después use Distributor para añadir una versión del paquete incorporando una entrada en el documento SSM que ya existe para las versiones anteriores. Para ahorrar tiempo, actualice el manifiesto de una versión anterior del paquete, cambie el valor de la entrada `version` en el manifiesto (por ejemplo, de `Test_1.0` a `Test_2.0`) y guárdelo como el manifiesto de la nueva versión. Debe tener un manifiesto actualizada para añadir una nueva versión del paquete utilizando el flujo de trabajo avanzado.

Para añadir una versión de paquete mediante el flujo de trabajo avanzado

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Distributor.
3. En la página de Distributor inicio, elija el paquete a la que desea añadir otra versión y, a continuación, elija Añadir versión.

4. En Version name (Nombre de versión), escriba el valor exacto que se encuentra en la entrada `version` en su archivo de manifiesto.
5. Para nombre de bucket de S3, elija un bucket de S3 existente de la lista. Este puede ser el mismo bucket que utilizó para almacenar archivos instalables para versiones anteriores, pero los nombres de los archivos instalables deben ser diferentes para evitar sobrescribir los archivos instalables existentes en el bucket.
6. En S3 key prefix (Prefijo de clave de S3), escriba la subcarpeta del bucket donde se almacenan sus recursos instalables.
7. En Manifest (Manifiesto), elija Extract from package (Extraer del paquete) para utilizar un manifiesto que cargó al bucket de S3 con sus archivos .zip.

   (Opcional) Si no cargó el manifiesto JSON revisado al bucket de Amazon S3 donde guardó los archivos .zip, elija New manifest (Nuevo manifiesto). Puede crear o pegar todo el manifiesto en el campo de edición de JSON. Para obtener más información sobre cómo crear el manifiesto JSON, consulte Paso 2: crear el manifiesto del paquete JSON.
8. Cuando haya terminado con el manifiesto, seleccione Agregar versión del paquete.
9. En la página Details (Detalles) del paquete, en la pestaña Versions (Versiones), vea la nueva versión en la lista de versiones de paquete disponibles. Establezca una versión predeterminada del paquete; para ello, elija una versión y, a continuación, elija Set default version (Establecer versión predeterminada).

   Si no establece una versión predeterminada, la versión del paquete más reciente es la versión predeterminada.

Añadir una versión del paquete mediante la AWS CLI

Puede utilizar la AWS CLI para añadir una nueva versión del paquete a Distributor. Antes de ejecutar estos comandos, debe crear una nueva versión del paquete y cargarlo en S3, tal y como se describe al principio de este tema.

Para añadir una versión de paquete mediante la AWS CLI

1. Ejecute el siguiente comando para editar el documento de AWS Systems Manager con una entrada para una nueva versión del paquete. Sustituya *`document-name`* por el nombre de su documento. Sustituya *`amzn-s3-demo-bucket`* con la URL del manifiesto JSON que copió en Paso 3: cargar el paquete y el manifiesto en un bucket de Amazon S3. *`S3-bucket-URL-of-package`* es la URL del bucket de Amazon S3 donde se almacena el paquete

completo. Sustituya *version-name-from-updated-manifest* por el valor de `version` en el manifiesto. Establezca el parámetro `--document-version` en `$LATEST` para que el documento asociado a esta versión del paquete sea la versión más reciente del documento.

```
aws ssm update-document \
    --name "document-name" \
    --content "S3-bucket-URL-to-manifest-file" \
    --attachments Key="SourceUrl",Values="amzn-s3-demo-bucket" \
    --version-name version-name-from-updated-manifest \
    --document-version $LATEST
```

A continuación se muestra un ejemplo.

```
aws ssm update-document \
    --name ExamplePackage \
    --content "https://s3.amazonaws.com/amzn-s3-demo-bucket/ExamplePackage/
manifest.json" \
    --attachments Key="SourceUrl",Values="https://s3.amazonaws.com/amzn-s3-demo-
bucket/ExamplePackage" \
    --version-name 1.1.1 \
    --document-version $LATEST
```

2. Ejecute el siguiente comando para verificar que el paquete se ha actualizado y muestra el manifiesto de paquete. Sustituya *package-name* por el nombre de su paquete y, si lo desea, *document-version* por el número de versión del documento (no coincide con la versión del paquete) que actualizó. Si esta versión del paquete está asociada a la versión más reciente del documento, puede especificar `$LATEST` para el valor del parámetro `--document-version` opcional.

```
aws ssm get-document \
    --name "package-name" \
    --document-version "document-version"
```

Para obtener más información acerca de otras opciones que puede utilizar con el comando update-document, consulte update-document en la sección sobre AWS Systems Manager de la Referencia de comandos de la AWS CLI.

Instalar o actualizar paquetes de Distributor

Puede implementar paquetes en los nodos administrados de AWS Systems Manager mediante Distributor, una herramienta de AWS Systems Manager. Para implementar los paquetes, utilice la AWS Management Console o AWS Command Line Interface (AWS CLI). Puede implementar una versión de un paquete en cada comando. Puede instalar nuevos paquetes o actualizar las instalaciones existentes. Puede elegir implementar una versión específica o elegir implementar siempre la versión más reciente de un paquete para implementación. Le recomendamos utilizar State Manager, una herramienta de AWS Systems Manager, para instalar paquetes. Utilice State Manager para asegurarse de que los nodos administrados siempre ejecutan la versión más actualizada del paquete.

> **Important**
>
> Los paquetes que instale mediante Distribuidor se deben desinstalar únicamente mediante Distribuidor. De lo contrario, Systems Manager podría seguir registrando la aplicación como `INSTALLED` y provocar otros resultados imprevistos.

Preference	Acción de AWS Systems Manager	Más información
Instale o actualice un paquete inmediatamente.	Run Command	• Instalación o actualización de un paquete una vez mediante la consola • Instalación de un paquete una vez mediante la AWS CLI • Actualización de un paquete una vez mediante la AWS CLI
Instale o actualice un paquete de manera programada, de forma que la instalación siempre incluya la versión predeterminada.	State Manager	• Programación de una instalación o actualización de un paquete mediante la consola

Preference	Acción de AWS Systems Manager	Más información
		• Programación de la instalación de un paquete mediante la AWS CLI • Programación de una actualización de paquete mediante la AWS CLI
Instale un paquete automáticamente en nodos administrados nuevos que tengan una etiqueta específica o un conjunto de etiquetas. Por ejemplo, instale el agente de Amazon CloudWatch en nuevas instancias.	State Manager	Una forma de hacerlo es aplicar etiquetas a nodos administrados nuevos y, a continuación, especificar las etiquetas como destinos en su asociación State Manager. State Manager instala automáticamente el paquete de una asociación en los nodos administrados que tienen etiquetas coincidentes. Consulte Cómo comprender los controles de frecuencia y destinos en las asociaciones de State Manager.

Temas

- Instalación o actualización de un paquete una vez mediante la consola
- Programación de una instalación o actualización de un paquete mediante la consola
- Instalación de un paquete una vez mediante la AWS CLI
- Actualización de un paquete una vez mediante la AWS CLI
- Programación de la instalación de un paquete mediante la AWS CLI
- Programación de una actualización de paquete mediante la AWS CLI

Instalación o actualización de un paquete una vez mediante la consola

Puede utilizar la consola AWS Systems Manager para instalar o actualizar un paquete una vez. Al configurar una instalación única, Distributor utiliza AWS Systems Manager Run Command, una herramienta de AWS Systems Manager, para llevar a cabo la instalación.

Para instalar o actualizar un paquete una vez mediante la consola

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Distributor.
3. En la página de inicio de Distributor, elija el paquete que desee instalar.
4. Elija Instalar una vez.

   Este comando abre Run Command con el documento de Command `AWS-ConfigureAWSPackage` y el paquete de Distributor ya seleccionado.
5. En Versión del documento, seleccione la versión del documento `AWS-ConfigureAWSPackage` que desea ejecutar.
6. En Acción, elija Instalar.
7. En Installation type, seleccione una de las siguientes opciones:

   - Desinstalar y reinstalar: el paquete se desinstala por completo y, a continuación, se vuelve a instalar. La aplicación no estará disponible hasta que se complete la reinstalación.
   - In-place update: solo se agregan archivos nuevos o modificados a la instalación existente según las instrucciones que proporcione en un script `update`. La aplicación permanece disponible durante todo el proceso de actualización. Esta opción no es compatible con los paquetes publicados de AWS, excepto el paquete de `AWSEC2Launch-Agent`.
8. En Nombre, compruebe que se ha introducido el nombre del paquete que ha seleccionado.
9. (Opcional) En Versión, escriba el valor de nombre de versión del paquete. Si deja este campo en blanco, Run Command instala la versión predeterminada que ha seleccionado en Distributor.
10. En la sección Destinos, para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos manualmente o especifique un grupo de recursos.

> **Note**
>
> Si no ve un nodo administrado en la lista, consulte Solución de problemas de disponibilidad de nodos administrados.

11. En Otros parámetros:

    - En Comentario, ingrese la información acerca de este comando.
    - En Tiempo de espera (segundos), especifique el número de segundos que tiene que esperar el sistema antes de indicar que se ha producido un error en la ejecución del comando general.

12. En Control de velocidad:

    - En Simultaneidad, especifique un número o un porcentaje de los destinos en los que desea ejecutar el comando al mismo tiempo.

      > **Note**
      >
      > Si seleccionó los destinos mediante la especificación de etiquetas o grupos de recursos y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

    - En Umbral de error, especifique cuándo desea parar la ejecución del comando en los demás destinos después de que haya fallado en un número o un porcentaje de los nodos administrados. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

13. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración

de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

14. En la sección Notificaciones de SNS, seleccione la casilla de verificación Habilitar notificaciones de SNS si desea recibir notificaciones sobre el estado de ejecución de los comandos.

    Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.
15. Cuando esté listo para instalar el paquete, elija Ejecutar.
16. El área Estado del comando informa del progreso de la ejecución. Si el comando sigue en curso, elija el icono de actualización en la esquina superior izquierda de la consola hasta que la columna Estado General o Estado detallado muestre Correcto o Error.
17. En el área Destinos y salidas, elija el botón situado junto a un nombre de nodo administrado y, a continuación, elija Ver resultado.

    La página de salida de comandos muestra los resultados de la ejecución de comandos.
18. (Opcional) Si elige escribir la salida del comando en un bucket de Amazon S3, elija Amazon S3 para ver los datos del registro de salida.

Programación de una instalación o actualización de un paquete mediante la consola

Puede utilizar la consola de AWS Systems Manager para programar la instalación o actualización de un paquete. Al planificar la instalación o actualización de un paquete, Distributor usa AWS Systems Manager State Manager para instalar o actualizar.

Para programar una instalación de paquete mediante la consola

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Distributor.
3. En la página de inicio de Distributor, elija el paquete que desee instalar o actualizar.
4. En Paquete, elija Instalar de manera programada.

Este comando abre State Manager en una nueva asociación que se crea automáticamente.

5. En Nombre, escriba un nombre (por ejemplo, **`Deploy-test-agent-package`**). Esto es opcional, pero recomendable. No se permiten espacios en el nombre.
6. En la lista Documento, el nombre del documento `AWS-ConfigureAWSPackage` ya está seleccionado.
7. En Acción, compruebe que se ha seleccionado Instalar.
8. En Installation type, seleccione una de las siguientes opciones:

   - Desinstalar y reinstalar: el paquete se desinstala por completo y, a continuación, se vuelve a instalar. La aplicación no estará disponible hasta que se complete la reinstalación.
   - In-place update: solo se agregan archivos nuevos o modificados a la instalación existente según las instrucciones que proporcione en un script `update`. La aplicación permanece disponible durante todo el proceso de actualización.
9. En Nombre, compruebe que se ha introducido el nombre del paquete.
10. En Versión, si desea instalar una versión del paquete que no sea la versión más reciente publicada, introduzca el identificador de versión.
11. En Destinos, elija Selecting all managed instances in this account, Especificación de etiquetas o Selección manual de la instancia. Si selecciona como destino recursos mediante el uso de etiquetas, introduzca una clave de etiqueta y un valor de etiqueta en los campos correspondientes.

    > **Note**
    >
    > Para elegir dispositivos de núcleo administrados de AWS IoT Greengrass, elija Selecting all managed instances in this account o Selección manual de la instancia.

12. En Especificar programación, seleccione Según la programación para ejecutar la asociación según un programa habitual o Sin programación para ejecutar la asociación una vez. Para obtener más información sobre estas opciones, consulte Trabajo con asociaciones en Systems Manager. Utilice los controles para crear un programa de `cron` o rate para la asociación.
13. Elija Crear asociación.
14. En la página Asociación pulse el botón situado junto a la asociación que ha creado y, a continuación, elija Aplicar asociación ahora.

State Manager crea y ejecuta inmediatamente la asociación en los destinos especificados. Para obtener más información acerca de los resultados de las asociaciones en ejecución, consulte Trabajo con asociaciones en Systems Manager en esta guía.

Para obtener más información sobre cómo trabajar con las opciones en Opciones avanzadas, Control de velocidad y Opciones de salida, consulte Trabajo con asociaciones en Systems Manager.

Instalación de un paquete una vez mediante la AWS CLI

Puede ejecutar send-command en la AWS CLI para instalar un paquete de Distributor una vez. Si el paquete ya está instalado, la aplicación se desconectará mientras se desinstala el paquete y se instala la nueva versión en su lugar.

Para instalar un paquete una vez mediante la AWS CLI

- Ejecute el siguiente comando en la 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**
>
> El comportamiento predeterminado para `installationType` es `Uninstall and reinstall`. Puede omitir `"installationType":["Uninstall and reinstall"]` de este comando cuando instale un paquete completo.

A continuación se muestra un ejemplo.

```
aws ssm send-command \
    --document-name "AWS-ConfigureAWSPackage" \
    --instance-ids "i-00000000000000" \
    --parameters '{"action":["Install"],"installationType":["Uninstall and
 reinstall"],"name":["ExamplePackage"]}'
```

Para obtener más información acerca de otras opciones que puede utilizar con el comando send-command, consulte send-command en la sección sobre AWS Systems Manager de la Referencia de comandos de la AWS CLI.

Actualización de un paquete una vez mediante la AWS CLI

Puede ejecutar send-command en la AWS CLI para actualizar un paquete de Distributor sin desconectar la aplicación asociada. Solo se reemplazan los archivos nuevos o actualizados del paquete.

Para actualizar un paquete una vez mediante la AWS CLI

- Ejecute el siguiente comando en la 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**
>
> Cuando agregue archivos nuevos o modificados, debe incluir `"installationType": ["In-place update"]` en el comando.

A continuación se muestra un ejemplo.

```
aws ssm send-command \
    --document-name "AWS-ConfigureAWSPackage" \
    --instance-ids "i-02573cafcfEXAMPLE" \
    --parameters '{"action":["Install"],"installationType":["In-place
 update"],"name":["ExamplePackage"]}'
```

Para obtener más información acerca de otras opciones que puede utilizar con el comando send-command, consulte send-command en la sección sobre AWS Systems Manager de la Referencia de comandos de la AWS CLI.

Programación de la instalación de un paquete mediante la AWS CLI

Puede ejecutar create-association en la AWS CLI para instalar un paquete de Distributor de forma programada. El valor de `--name`, el nombre del documento, es siempre `AWS-ConfigureAWSPackage`. El comando siguiente utiliza la clave `InstanceIds` para especificar los nodos administrados de destino. Si el paquete ya está instalado, la aplicación se desconectará mientras se desinstala el paquete y se instala la nueva versión en su lugar.

```
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**
>
> El comportamiento predeterminado para `installationType` es `Uninstall and reinstall`. Puede omitir `"installationType":["Uninstall and reinstall"]` de este comando cuando instale un paquete completo.

A continuación se muestra un ejemplo.

```
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\"]}]
```

Para obtener más información acerca de otras opciones que puede utilizar con el comando create-association, consulte create-association en la sección sobre AWS Systems Manager de la Referencia de comandos de la AWS CLI.

Programación de una actualización de paquete mediante la AWS CLI

Puede ejecutar create-association en la AWS CLI para actualizar un paquete de Distributor de forma programada sin desconectar la aplicación asociada. Solo se reemplazan los archivos nuevos

o actualizados del paquete. El valor de `--name`, el nombre del documento, es siempre `AWS-ConfigureAWSPackage`. El comando siguiente utiliza la clave `InstanceIds` para especificar las instancias de destino.

```
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**
>
> Cuando agregue archivos nuevos o modificados, debe incluir `"installationType": ["In-place update"]` en el comando.

A continuación se muestra un ejemplo.

```
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\"]}]
```

Para obtener más información acerca de otras opciones que puede utilizar con el comando create-association, consulte create-association en la sección sobre AWS Systems Manager de la Referencia de comandos de la AWS CLI.

Desinstalación de un paquete Distributor

Puede utilizar la AWS Management Console o la AWS Command Line Interface (AWS CLI) para desinstalar los paquetes de Distributor de sus nodos administrados de AWS Systems Manager mediante el uso de Run Command. Distributor y Run Command son herramientas de AWS Systems Manager. En esta versión, puede desinstalar una versión de un paquete en cada comando. Puede desinstalar una versión específica o la versión predeterminada.

> **Important**
>
> Los paquetes que instale mediante Distribuidor se deben desinstalar únicamente mediante Distribuidor. De lo contrario, Systems Manager podría seguir registrando la aplicación como `INSTALLED` y provocar otros resultados imprevistos.

Temas

- Desinstalación de un paquete mediante la consola
- Desinstalación de un paquete mediante la AWS CLI

Desinstalación de un paquete mediante la consola

Puede utilizar Run Command en la consola de Systems Manager para desinstalar un paquete una vez. Distributor utiliza AWS Systems Manager Run Command para desinstalar paquetes.

Para desinstalar un paquete mediante la consola

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command.
3. En la página de inicio de Run Command, elija Run command (Ejecutar comando)..
4. Elija el documento de comando `AWS-ConfigureAWSPackage`.
5. Desde Action (Acción), elija Uninstall (Desinstalar).
6. Para Name (Nombre), escriba el nombre del paquete que desea desinstalar.
7. En Targets (Destinos), elija el modo en el que desea dirigirse a los nodos administrados. Puede especificar una clave de etiqueta y valores compartidos por los destinos. Para especificar destinos, también puede elegir atributos, como un ID, una plataforma y la versión de SSM Agent.
8. Puede utilizar las opciones avanzadas para agregar comentarios acerca de la operación, cambiar los valores de Concurrency (Simultaneidad) y Error threshold (Límite de error) en Rate control (Control de velocidad), especificar las opciones de salida o configurar las notificaciones de Amazon Simple Notification Service (Amazon SNS). Para obtener más información, consulte Ejecución de comandos desde la consola en esta guía.
9. Cuando esté listo para desinstalar el paquete, elija Run (Ejecutar) y, a continuación, seleccione View results (Ver resultados).

10. En la lista de comandos, elija el comando `AWS-ConfigureAWSPackage` que ha ejecutado. Si el comando todavía está en curso, elija el icono de actualización de la esquina superior derecha de la consola.
11. Cuando la columna Status (Estado) muestre Success (Correcto) o Failed (Error), elija la pestaña Output (Salida).
12. Elija View output (Ver salida). La página de salida de comandos muestra los resultados de la ejecución de comandos.

Desinstalación de un paquete mediante la AWS CLI

Puede utilizar la AWS CLI para desinstalar un paquete de Distributor de sus nodos administrados mediante Run Command.

Para desinstalar un paquete mediante la AWS CLI

- Ejecute el siguiente comando en la 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)"]}'
```

  A continuación se muestra un ejemplo.

```
aws ssm send-command \
    --document-name "AWS-ConfigureAWSPackage" \
    --instance-ids "i-02573cafcfEXAMPLE" \
    --parameters '{"action":["Uninstall"],"name":["Test-ConfigureAWSPackage"]}'
```

Para obtener más información acerca de otras opciones que puede utilizar con el comando send-command, consulte send-command en la sección sobre AWS Systems Manager de la Referencia de comandos de la AWS CLI.

Eliminar un paquete de Distributor

En esta sección se describe cómo eliminar un paquete. No es posible eliminar la versión de un paquete, solo todo el paquete.

Eliminar un paquete mediante la consola

Puede utilizar la consola de AWS Systems Manager para eliminar un paquete o una versión del paquete de Distributor, una herramienta de AWS Systems Manager. Si elimina un paquete, eliminará todas las versiones de un paquete de Distributor.

Para eliminar un paquete mediante la consola

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Distributor.
3. En la página de inicio de Distributor, seleccione el paquete que desee eliminar.
4. En la página de detalles del paquete, elija Delete package (Eliminar paquete).
5. Cuando se le pida que confirme la eliminación, elija Delete package (Eliminar paquete).

Eliminación de una versión del paquete mediante la consola

Puede utilizar la consola de Systems Manager para eliminar una versión del paquete de Distributor.

Para eliminar una versión de paquete mediante la consola

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Distributor.
3. En la página de inicio de Distributor, elija el paquete cuya versión desee eliminar.
4. En la página de versiones del paquete, elija la versión que desea eliminar y haga clic en Delete version (Eliminar versión).
5. Cuando le pidan que confirme la eliminación, elija Delete package version (Eliminar versión del paquete).

Eliminación de un paquete mediante la línea de comandos

Puede utilizar la herramienta de línea de comandos que prefiera para eliminar un paquete de Distributor.

Linux & macOS

Para eliminar un paquete mediante la AWS CLI

1. Ejecute el siguiente comando para enumerar los documentos de paquetes específicos. En los resultados de este comando, busque el paquete que desea eliminar.

```
aws ssm list-documents \
    --filters Key=Name,Values=package-name
```

2. Ejecute el siguiente comando para eliminar un paquete. Sustituya *package-name* por el nombre del paquete.

```
aws ssm delete-document \
    --name "package-name"
```

3. Vuelva a ejecutar el comando list-documents para comprobar que el paquete se ha eliminado. El paquete que ha eliminado no debería incluirse en la lista.

```
aws ssm list-documents \
    --filters Key=Name,Values=package-name
```

Windows

Para eliminar un paquete mediante la AWS CLI

1. Ejecute el siguiente comando para enumerar los documentos de paquetes específicos. En los resultados de este comando, busque el paquete que desea eliminar.

```
aws ssm list-documents ^
    --filters Key=Name,Values=package-name
```

2. Ejecute el siguiente comando para eliminar un paquete. Sustituya *package-name* por el nombre del paquete.

```
aws ssm delete-document ^
    --name "package-name"
```

3. Vuelva a ejecutar el comando list-documents para comprobar que el paquete se ha eliminado. El paquete que ha eliminado no debería incluirse en la lista.

```
aws ssm list-documents ^
    --filters Key=Name,Values=package-name
```

PowerShell

Para eliminar un paquete mediante Tools for PowerShell

1. Ejecute el siguiente comando para enumerar los documentos de paquetes específicos. En los resultados de este comando, busque el paquete que desea eliminar.

```
$filter = New-Object
 Amazon.SimpleSystemsManagement.Model.DocumentKeyValuesFilter
$filter.Key = "Name"
$filter.Values = "package-name"

Get-SSMDocumentList `
    -Filters @($filter)
```

2. Ejecute el siguiente comando para eliminar un paquete. Sustituya *package-name* por el nombre del paquete.

```
Remove-SSMDocument `
    -Name "package-name"
```

3. Vuelva a ejecutar el comando Get-SSMDocumentList para comprobar que el paquete se ha eliminado. El paquete que ha eliminado no debería incluirse en la lista.

```
$filter = New-Object
 Amazon.SimpleSystemsManagement.Model.DocumentKeyValuesFilter
$filter.Key = "Name"
$filter.Values = "package-name"

Get-SSMDocumentList `
    -Filters @($filter)
```

Eliminación de una versión del paquete mediante la línea de comandos

Puede utilizar la herramienta de línea de comandos que prefiera para eliminar una versión del paquete de Distributor.

Linux & macOS

Para eliminar una versión de paquete mediante la AWS CLI

1. Ejecute el siguiente comando para mostrar las versiones del paquete. En los resultados, busque la versión del paquete que desea eliminar.

```
aws ssm list-document-versions \
    --name "package-name"
```

2. Ejecute el siguiente comando para eliminar una versión del paquete. Reemplace *package-name* por el nombre del paquete y *version* por el número de versión.

```
aws ssm delete-document \
    --name "package-name" \
    --document-version version
```

3. Ejecute el comando list-document-versions para comprobar que la versión del paquete se ha eliminado. La versión del paquete que eliminó ya no debería encontrarse.

```
aws ssm list-document-versions \
    --name "package-name"
```

Windows

Para eliminar una versión de paquete mediante la AWS CLI

1. Ejecute el siguiente comando para mostrar las versiones del paquete. En los resultados, busque la versión del paquete que desea eliminar.

```
aws ssm list-document-versions ^
    --name "package-name"
```

2. Ejecute el siguiente comando para eliminar una versión del paquete. Reemplace *package-name* por el nombre del paquete y *version* por el número de versión.

```
aws ssm delete-document ^
    --name "package-name" ^
    --document-version version
```

3. Ejecute el comando list-document-versions para comprobar que la versión del paquete se ha eliminado. La versión del paquete que eliminó ya no debería encontrarse.

```
aws ssm list-document-versions ^
    --name "package-name"
```

PowerShell

Para eliminar una versión del paquete mediante Tools for PowerShell

1. Ejecute el siguiente comando para mostrar las versiones del paquete. En los resultados, busque la versión del paquete que desea eliminar.

```
Get-SSMDocumentVersionList `
    -Name "package-name"
```

2. Ejecute el siguiente comando para eliminar una versión del paquete. Reemplace *package-name* por el nombre del paquete y *version* por el número de versión.

```
Remove-SSMDocument `
    -Name "package-name" `
    -DocumentVersion version
```

3. Ejecute el comando Get-SSMDocumentVersionList para comprobar que la versión del paquete se ha eliminado. La versión del paquete que eliminó ya no debería encontrarse.

```
Get-SSMDocumentVersionList `
    -Name "package-name"
```

Para obtener más información acerca de otras opciones que puede utilizar con el comando list-documents, consulte list-documents en la sección sobre AWS Systems Manager de la Referencia de comandos de la AWS CLI. Para obtener más información sobre otras opciones que puede usar con el comando delete-document, consulte delete-document.

Auditoría y registro de la actividad de Distributor

Puede utilizar AWS CloudTrail para auditar la actividad relacionada con Distributor, una herramienta de AWS Systems Manager. Para obtener más información acerca de cómo auditar y registrar opciones de Systems Manager, consulte Registro y monitorización en AWS Systems Manager.

Audite la actividad de Distributor mediante el uso de CloudTrail

CloudTrail captura las llamadas a las API realizadas en la consola de AWS Systems Manager, la AWS Command Line Interface (AWS CLI) y el SDK de Systems Manager. La información puede consultarse en la consola de CloudTrail o almacenarse en un bucket de Amazon Simple Storage Service (Amazon S3). Se utiliza un bucket para todos los registros de CloudTrail de su cuenta.

Los registros de acciones de Run Command y State Manager muestran la actividad de creación de documentos, instalación de paquetes y desinstalación de paquetes. Run Command y State Manager son herramientas de AWS Systems Manager. Para obtener más información acerca de cómo ver y utilizar los registros de CloudTrail de la actividad de Systems Manager, consulte Registro de llamadas a la API de AWS Systems Manager con AWS CloudTrail.

Solución de problemas de AWS Systems ManagerDistributor

La siguiente información puede ayudarlo a solucionar problemas que pueden ocurrir cuando utiliza Distributor, una herramienta de AWS Systems Manager.

Temas

- Instalación de paquete incorrecto con el mismo nombre
- Error: no se pudo recuperar el manifiesto: no se pudo encontrar la versión más reciente del paquete
- Error: no se pudo recuperar el manifiesto: excepción de validación
- El paquete no es compatible (falta la acción de instalación del paquete)
- Error: Error al descargar el manifiesto: el documento con el nombre no existe
- Carga fallida.
- Error: No se pudo encontrar la plataforma: no se encontró ningún manifiesto para la plataforma: Oracle, versión 8.9, arquitectura x86_64

Instalación de paquete incorrecto con el mismo nombre

Problema: ha instalado un paquete, pero Distributor ha instalado un paquete diferente en su lugar.

Causa: durante la instalación, Systems Manager encuentra paquetes publicados por AWS como resultados antes que los paquetes externos definidos por el usuario. Si el nombre del paquete definido por el usuario es el mismo que un nombre de paquete publicado por AWS, el paquete de AWS se instala en lugar de su paquete.

Solución: para evitar este problema, asigne al paquete un nombre algo diferente del nombre de un paquete publicado por AWS.

Error: no se pudo recuperar el manifiesto: no se pudo encontrar la versión más reciente del paquete

Problema: ha recibido un error como el siguiente.

```
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
```

Causa: está utilizando una versión de SSM Agent con Distributor anterior a la versión 2.3.274.0.

Solución: actualice la versión de SSM Agent a la versión 2.3.274.0 o una versión posterior. Para obtener más información, consulte Actualización de SSM Agent mediante Run Command o Tutorial: actualización automática de SSM Agent con AWS CLI.

Error: no se pudo recuperar el manifiesto: excepción de validación

Problema: ha recibido un error como el siguiente.

```
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
```

Causa: está utilizando una versión de SSM Agent con Distributor anterior a la versión 2.3.274.0.

Solución: actualice la versión de SSM Agent a la versión 2.3.274.0 o una versión posterior. Para obtener más información, consulte Actualización de SSM Agent mediante Run Command o Tutorial: actualización automática de SSM Agent con AWS CLI.

El paquete no es compatible (falta la acción de instalación del paquete)

Problema: ha recibido un error como el siguiente.

```
Package is not supported (package is missing install action)
```

Causa: la estructura del directorio del paquete es incorrecta.

Solución: no comprima un directorio principal que contenga el software y los scripts necesarios. En su lugar, cree un archivo de `.zip` con todos los contenidos requeridos directamente en la ruta absoluta. Para comprobar que el archivo de `.zip` se creó correctamente, descomprima el directorio de la plataforma destino y revise la estructura del directorio. Por ejemplo, la ruta absoluta del script de instalación debe ser */*_ExamplePackage_targetPlatform_/install.sh*.

Error: Error al descargar el manifiesto: el documento con el nombre no existe

Problema: ha recibido un error como el siguiente.

```
Failed to download manifest - failed to retrieve package document description:
 InvalidDocument: Document with name filename does not exist.
```

Causa 1: Distributor no puede encontrar el paquete por el nombre del paquete al compartir un paquete de Distributor de otra cuenta.

Solución 1: cuando comparta un paquete de otra cuenta, utilice el Nombre de recurso de Amazon (ARN) completo del paquete y no solo el nombre.

Causa 2: cuando utiliza una VPC, no ha proporcionado el perfil de instancia de IAM con acceso al bucket de S3 administrado de AWS que contiene el documento `AWS-ConfigureAWSPackage` para la Región de AWS de destino.

Solución 2: asegúrese de que el perfil de instancia de IAM proporcione a SSM Agent acceso al bucket de S3 administrado de AWS que contiene el documento `AWS-ConfigureAWSPackage` para la Región de AWS de destino, como se explica en Comunicaciones de SSM Agent con buckets de S3 administrados de AWS.

Carga fallida.

Problema: ha recibido un error como el siguiente.

```
Upload failed. At least one of your files was not successfully uploaded to your S3
 bucket.
```

Causa: el nombre del paquete de software incluye un espacio. Por ejemplo, `Hello World.msi` no se podría cargar.

Error: No se pudo encontrar la plataforma: no se encontró ningún manifiesto para la plataforma: Oracle, versión 8.9, arquitectura x86_64

Problema: ha recibido un error como el siguiente.

```
Failed to find platform: no manifest found for platform: oracle, version 8.9,
 architecture x86_64
```

Causa: el error significa que el manifiesto del paquete JSON no tiene ninguna definición para esa plataforma, en este caso, Oracle Linux.

Solución: descargue el paquete que desee distribuir del sitio web de Deep Security Software de Trend Micro. Cree un paquete de software `.rpm` mediante el proceso de flujo de trabajo simple de Distributor. Establezca los siguientes valores para el paquete y complete la carga del paquete de software con Distributor:

```
Platform version: _any
Target platform: oracle
Architecture: x86_64
```

AWS Systems Manager Fleet Manager

Fleet Manager, una herramienta de AWS Systems Manager, es una experiencia de interfaz de usuario (UI) unificada que lo ayuda a administrar de forma remota los nodos que se ejecutan en AWS o en las instalaciones. Con Fleet Manager, puede ver el estado y el rendimiento de toda la flota de servidores desde una sola consola. También puede recopilar datos de nodos individuales para realizar tareas comunes de solución de problemas y administración desde la consola. Esto incluye la conexión a instancias de Windows mediante el protocolo de escritorio remoto (RDP), la visualización del contenido de carpetas y archivos, la administración del registro de Windows, la administración de usuarios del sistema operativo y más.

Para comenzar a utilizar Fleet Manager, abra la consola de Systems Manager. En el panel de navegación, elija Fleet Manager.

¿Quién debe utilizar Fleet Manager?

Cualquier cliente de AWS que desee tener una forma centralizada de administrar su flota de nodos debería utilizar Fleet Manager.

¿Cómo puede Fleet Manager beneficiar a mi organización?

Fleet Manager ofrece las ventajas siguientes:

- Realice una variedad de tareas comunes de administración de sistemas sin tener que conectarse de manera manual a los nodos administrados.
- Administre nodos que se ejecutan en varias plataformas desde una única consola unificada.
- Administre nodos que ejecutan diferentes sistemas operativos desde una única consola unificada.
- Mejore la eficiencia de la administración de los sistemas.

¿Cuáles son las características de Fleet Manager?

Estas son algunas de las características clave de Fleet Manager:

- Acceso al portal de la base de conocimientos de Red Hat

  Acceda a datos binarios, recursos compartidos de conocimientos y foros de discusión en el portal de la base de conocimientos de Red Hat a través de las instancias de Red Hat Enterprise Linux (RHEL).
- Estado de nodos administrados

  Vea qué instancias administradas están `running` (ejecutándose) y cuáles están `stopped` (detenidas). Para obtener más información acerca de las instancias detenidas, consulte Detención e inicio de una instancia en la Guía del usuario de Amazon EC2. Para dispositivos de núcleo de AWS IoT Greengrass, puede ver cuáles están `online`, `offline` o muestran el estado de `Connection lost`.

  > **Note**
  >
  > Si detuvo la instancia administrada antes del 12 de julio de 2021, no mostrará el marcador `stopped`. Para mostrar el marcador, inicie y detenga la instancia.

- Ver la información de la instancia

  Vea información sobre los datos de carpetas y archivos almacenados en los volúmenes adjuntados en las instancias administradas, los datos de rendimiento sobre las instancias en tiempo real y los datos de registro almacenados en las instancias.
- View edge device information (Ver información de dispositivo de borde)

Vea el nombre del objeto del dispositivo de AWS IoT Greengrass, el estado del ping y la versión de SSM Agent, y mucho más.

- Administración de cuentas y registro

  Administre las cuentas de usuario del sistema operativo (SO) en las instancias y el registro en las instancias de Windows.

- Controlar el acceso a las características

  Controle el acceso a las características de Fleet Manager mediante políticas de AWS Identity and Access Management (IAM). Con estas políticas, puede controlar qué usuarios o grupos de la organización pueden utilizar varias características de Fleet Manager y qué nodos administrados pueden administrar.

Temas

- Configuración de Fleet Manager
- Trabajo con nodos administrados
- Administración automática de instancias EC2 con la configuración de administración de hosts predeterminada
- Conéctese a una instancia administrada de Windows Server mediante Remote Desktop
- Administración de volúmenes de Amazon EBS en instancias administradas
- Acceso al portal de la base de conocimientos de Red Hat
- Solución de problemas de disponibilidad de nodos administrados

Configuración de Fleet Manager

Antes de que los usuarios de su Cuenta de AWS puedan utilizar Fleet Manager, una herramienta de AWS Systems Manager, para supervisar y administrar los nodos administrados, es preciso que se les concedan los permisos necesarios. Además, para que cualquier instancia de Amazon Elastic Compute Cloud (Amazon EC2); los dispositivos de núcleo de AWS IoT Greengrass; y los servidores en las instalaciones, dispositivos de la periferia y máquinas virtuales (VM) sean monitoreadas y administradas mediante Fleet Manager, deben ser nodos administrados de Systems Manager. Un nodo administrado es cualquier máquina configurada para su uso con Systems Manager en entornos híbridos y multinube.

Esto significa que los nodos deben cumplir ciertos requisitos previos y configurarse con el agente de AWS Systems Manager (SSM Agent).

Según el tipo de máquina, consulte uno de los siguientes temas para asegurarse de que sus máquinas cumplen los requisitos de los nodos administrados.

- Instancias de Amazon EC2: Administrar instancias de EC2 con Systems Manager

> **Tip**
>
> También puede utilizar Quick Setup, una herramienta de AWS Systems Manager, para ayudarlo a configurar rápidamente las instancias de Amazon EC2 como instancias administradas en una cuenta individual. Si la empresa u organización utiliza AWS Organizations, también puede configurar instancias en varias unidades organizativas y Regiones de AWS. Para obtener más información acerca del uso de Quick Setup para configurar instancias administradas, consulte Instalar la administración de host de Amazon EC2 mediante Quick Setup.

- Servidores en las instalaciones y de otros tipos en la nube: Administrador de nodos en entornos híbridos y multinube con Systems Manager
- Dispositivos AWS IoT Greengrass (de la periferia): Administración de dispositivos periféricos con Systems Manager

Ejemplos de política

- Control del acceso a Fleet Manager

Control del acceso a Fleet Manager

Para utilizar Fleet Manager, una herramienta de AWS Systems Manager, su usuario o rol de AWS Identity and Access Management (IAM) debe tener los permisos necesarios. Puede crear una política de IAM que proporcione acceso a todas las características de Fleet Manager o modificar su política para conceder acceso a las características que elija. A continuación, conceda estos permisos a los usuarios o identidades en su cuenta.

Tarea 1: crear políticas de IAM para definir los permisos de acceso

Siga uno de los métodos que se proporcionan en el siguiente tópico de la Guía del usuario de IAM para crear un IAM y conceder acceso a Fleet Manager a las identidades (usuarios, roles o grupos de usuarios):

- Defina los permisos de IAM personalizados con políticas administradas por el cliente

Puede utilizar uno de los ejemplos de políticas que se indican a continuación o modificarlas según los permisos que desee conceder. Proporcionamos políticas de ejemplo para el acceso completo a Fleet Manager y el acceso de solo lectura.

Tarea 2: asociar las políticas de IAM a los usuarios para conceder permisos

Una vez que haya creado la política o las políticas de IAM que definen los permisos de acceso a Fleet Manager, use uno de los siguientes procedimientos de la Guía del usuario de IAM para conceder estos permisos a las identidades de su cuenta:

- Adición de permisos de identidad de IAM (consola)
- Adición de permisos de identidad de IAM (AWS CLI)
- Adición de permisos de identidad de IAM (API de AWS)

Temas

- Ejemplo de política para el acceso de administrador de Fleet Manager
- Ejemplo de política para el acceso de solo lectura de Fleet Manager

Ejemplo de política para el acceso de administrador de Fleet Manager

La siguiente política proporciona permisos para todas las características de Fleet Manager. Esto significa que un usuario puede crear y eliminar usuarios y grupos locales, modificar la membresía a un grupo para cualquier grupo local y modificar claves o valores del registro Windows Server. Reemplace cada *example resource placeholder* con su propia información.

JSON

```
{
    "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:us-east-1:111122223333:servicesetting/ssm/
managed-instance/default-ec2-instance-management-role"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
```

```
            ],
            "Resource": "arn:aws:iam::111122223333: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:*:111122223333:instance/*",
                "arn:aws:ssm:*:111122223333:managed-instance/*",
                "arn:aws:ssm:*:111122223333: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"
            ]
        },
        {
            "Sid": "TerminateSession",
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/aws:ssmmessages:session-id": [
                        "${aws:userid}"
                    ]
                }
            }
        }
    ]
}
```

Ejemplo de política para el acceso de solo lectura de Fleet Manager

La siguiente política proporciona permisos para características de Fleet Manager de solo lectura. Reemplace cada *example resource placeholder* con su propia información.

JSON

```
{
    "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:*:111122223333:instance/*",
```

```
                "arn:aws:ssm:*:111122223333:managed-instance/*",
                "arn:aws:ssm:*:111122223333: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"
            ]
        },
        {
            "Sid": "TerminateSession",
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/aws:ssmmessages:session-id": [
                        "${aws:userid}"
                    ]
                }
            }
        }
    ]
}
```

Trabajo con nodos administrados

Un nodo administrado es cualquier máquina configurada para AWS Systems Manager. Puede configurar los siguientes tipos de máquinas como nodos administrados:

- Instancias de Amazon Elastic Compute Cloud (Amazon EC2)
- Servidores en sus propias instalaciones (servidores en las instalaciones)
- Dispositivos de núcleo de AWS IoT Greengrass

- Dispositivos de AWS IoT y periféricos que no sean de AWS
- Máquinas virtuales (VM), incluidas las VM de otros entornos de nube

En la consola de Systems Manager, toda máquina que tenga el prefijo “mi-” se configuró como un nodo administrado mediante una activación híbrida. Los dispositivos de borde muestran el nombre del objeto de AWS IoT.

> **Note**
>
> La única característica compatible con las instancias de macOS es la vista del sistema de archivos.

Acerca de los niveles de instancias de Systems Manager

AWS Systems Manager ofrece un nivel de instancias estándares y un nivel de instancias avanzadas. Ambos admiten nodos administrados en su entorno híbrido y multinube. El nivel de instancias estándar le permite registrar un máximo de 1000 máquinas por Cuenta de AWS por Región de AWS. Si tiene que registrar más de 1000 máquinas en una única cuenta y región, utilice el nivel de instancias avanzadas. Puede crear tantos nodos administrados como desee en el nivel de instancias avanzadas. Todos los nodos administrados configurados para Systems Manager tienen un precio de pago por uso. Para obtener más información acerca de la habilitación del nivel de instancias avanzadas, consulte Activación del nivel de instancias avanzadas. Para obtener más información sobre los precios, consulte Precios de AWS Systems Manager.

Tenga en cuenta la siguiente información adicional acerca del nivel de instancias estándar y el nivel de instancias avanzadas:

- Las instancias avanzadas también permiten conectar a los nodos que no son de EC2 a un entorno híbrido y multinube mediante AWS Systems Manager Session Manager. Session Manager proporciona acceso a las instancias mediante el intérprete de comandos interactivo. Para obtener más información, consulte AWS Systems Manager Session Manager.
- La cuota de instancias estándar también se aplica a las instancias EC2 que utilizan una activación local de Systems Manager (que no es un escenario común).
- Para aplicar revisiones a las aplicaciones publicadas por Microsoft en instancias locales de máquinas virtuales, active el nivel de instancias avanzadas. El uso del nivel de instancias avanzadas conlleva un cargo. No hay ningún cargo adicional por usar revisiones en las

aplicaciones publicadas por Microsoft en instancias de Amazon Elastic Compute Cloud (Amazon EC2). Para obtener más información, consulte Revisiones de aplicaciones publicadas por Microsoft en Windows Server.

Mostrar nodos administrados

Si no ve los nodos administrados que se muestran en la consola, haga lo siguiente:

1. Verifique que la consola esté abierta en la Región de AWS en la que creó los nodos administrados. Puede cambiar Regiones utilizando la lista en la parte superior, que se encuentra en la esquina superior derecha de la consola.
2. Compruebe que los pasos de configuración para los nodos administrados cumplan los requisitos de Systems Manager. Para obtener más información, consulta Configuración de nodos administrados para AWS Systems Manager.
3. En el caso de las máquinas que no son de EC2, compruebe que completó el proceso de activación híbrida. Para obtener más información, consulte Administrador de nodos en entornos híbridos y multinube con Systems Manager.

Tenga en cuenta la siguiente información adicional:

- La consola de Fleet Manager no muestra los nodos de Amazon EC2 que se terminaron.
- Systems Manager requiere referencias de tiempo precisas para realizar operaciones en las máquinas. Si no se establecen correctamente la fecha y la hora de los nodos administrados, es posible que las máquinas no coincidan con la fecha de la firma de las solicitudes de la API. Para obtener más información, consulte Casos de uso y prácticas recomendadas.
- Al crear o editar etiquetas, el sistema puede tardar hasta una hora en mostrar los cambios en el filtro de tabla.
- Cuando el estado de un nodo administrado sea `Connection Lost` durante al menos 30 días, es posible que el nodo deje de aparecer en la lista de la consola de Fleet Manager. Para que vuelva a aparecer en la lista, se debe resolver el problema que haya provocado la pérdida de conexión. Para obtener sugerencias acerca de la solución de problemas, consulte Solución de problemas de disponibilidad de nodos administrados.

Verificar la compatibilidad de Systems Manager en un nodo administrado

AWS Config proporciona reglas administradas de AWS, que son reglas personalizables y predefinidas que AWS Config utiliza para evaluar si las configuraciones de recursos de AWS se ajustan a las prácticas recomendadas. AWS Config se incluye la regla ec2-instance-managed-by-systems-manager. Esta regla verifica si las instancias de Amazon EC2 de su cuenta se administran mediante Systems Manager. Para obtener más información, consulte Reglas administradas de AWS Config.

Aumento de la posición de seguridad en nodos administrados

Para obtener información acerca de cómo aumentar la posición de seguridad frente a comandos de nivel raíz no autorizados en los nodos administrados, consulte Restricción del acceso a los comandos de nivel raíz con SSM Agent.

Anular el registro de nodos administrados

Puede anular el registro de los nodos administrados en cualquier momento. Por ejemplo, si administra varios nodos con el mismo rol de AWS Identity and Access Management (IAM) y nota cualquier tipo de comportamiento malicioso, puede anular el registro de cualquier número de equipos en cualquier momento. (Para volver a registrar la misma máquina, debe utilizar un código de activación híbrida y un ID de activación diferentes a los utilizados anteriormente para registrarla). Para obtener información acerca de cómo anular el registro de los nodos administrados, consulte Anulación del registro de nodos administrados en un entorno híbrido y multinube.

Temas

- Configuración de los niveles de instancias
- Restablecimiento de contraseñas en nodos administrados
- Anulación del registro de nodos administrados en un entorno híbrido y multinube
- Utilizar Fleet Manager para trabajar con sistemas de archivos del sistema operativo
- Monitoreo del rendimiento de nodos administrados
- Trabajo con procesos
- Visualización de registros en nodos administrados
- Utilizar Fleet Manager para administrar cuentas de usuario y grupos del sistema operativo en nodos administrados
- Administración del registro de Windows en nodos administrados

Configuración de los niveles de instancias

En este tema se describen las situaciones en las que debe activar el nivel de instancias avanzadas.

AWS Systems Manager ofrece un nivel de instancias estándar y un nivel de instancias avanzadas para las máquinas que no sean de EC2 en un entorno híbrido y multinube.

Puede registrar hasta 1000 nodos activados de manera híbrida estándar por cada cuenta por Región de AWS sin costo adicional. Sin embargo, para registrar más de 1000 nodos híbridos es necesario activar el nivel de instancias avanzadas. El uso del nivel de instancias avanzadas conlleva un cargo. Para más información, consulte Precios de AWS Systems Manager.

Incluso con menos de 1000 nodos activados de manera híbrida y registrados, otros dos escenarios requieren el nivel de instancias avanzadas:

- Quiere usar Session Manager para conectarse a nodos que no son de EC2.
- Desea aplicar revisiones a aplicaciones (no a sistemas operativos) lanzadas por Microsoft en nodos que no sean de EC2.

> **Note**
>
> No hay ningún cargo por usar revisiones en las aplicaciones lanzadas por Microsoft en instancias de Amazon EC2.

Escenarios detallados de instancias avanzadas

La siguiente información proporciona detalles sobre los tres escenarios en los que debe activar el nivel de instancias avanzadas.

Escenario 1: registro de más de 1000 nodos activados de manera híbrida

Con el nivel de instancias estándar, puede registrar un máximo de 1000 nodos que no sean de EC2 en un entorno híbrido y multinube por Región de AWS en una cuenta específica sin cargo adicional. Si tiene que registrar más de 1000 nodos que no sean de EC2 en una región, debe utilizar el nivel de instancias avanzadas. A continuación, puede activar tantas máquinas para su entorno híbrido y multinube como desee. Las instancias avanzadas se cobran en función del número de nodos avanzados activados como nodos administrados de Systems Manager y de las horas en que se ejecutan esos nodos.

Todos los nodos administrados de Systems Manager que utilicen el proceso de activación descrito en Creación de una activación híbrida para registrar nodos con Systems Manager están por lo tanto sujetos a cargos si se superan los 1000 nodos locales en una región en una cuenta específica.

> **Note**
>
> También puede activar instancias de Amazon Elastic Compute Cloud (Amazon EC2) mediante las activaciones híbridas de Systems Manager y trabajar con ellas como instancias que no son de EC2, por ejemplo, para hacer pruebas. Estos también se considerarían nodos híbridos. Este no es un escenario común.

Escenario 2: Aplicación de revisiones a aplicaciones lanzadas por Microsoft en nodos activados de forma híbrida

El nivel de instancias avanzadas también es obligatorio si desea aplicar revisiones a aplicaciones lanzadas por Microsoft en nodos que no sean de EC2 en un entorno híbrido y multinube. Si activa el nivel de instancias avanzadas para aplicar revisiones a aplicaciones de Microsoft en nodos que no sean de EC2, se cobrarán cargos por todos los nodos en las instalaciones, incluso si tiene menos de 1000.

No hay ningún cargo adicional por usar revisiones en las aplicaciones publicadas por Microsoft en instancias de Amazon Elastic Compute Cloud (Amazon EC2). Para obtener más información, consulte Revisiones de aplicaciones publicadas por Microsoft en Windows Server.

Escenario 3: Conexión a nodos activados de forma híbrida mediante Session Manager

Session Manager proporciona acceso de shell interactivo a las instancias. Para conectarse a nodos administrados activados de manera híbrida mediante Session Manager, primero debe activar el nivel de instancias avanzadas. A continuación, se cobran cargos por todos los nodos activados de forma híbrida, incluso si tiene menos de 1000.

Resumen: ¿cuándo es necesario el nivel de instancias avanzadas?

Use la siguiente tabla para ver cuándo tiene que usar el nivel de instancias avanzadas y en qué situaciones se aplican cargos adicionales.

Escenario	¿Se necesita un nivel de instancias avanzado?	¿Se aplican cargos adicionales?
El número de nodos activados de forma híbrida de mi región en una cuenta específica supera los 1000.	Sí	Sí
Quiero usar Patch Manager para aplicar revisiones a aplicaciones lanzadas por Microsoft en cualquier número de nodos activados de forma híbrida, incluso menos de 1000.	Sí	Sí
Quiero usar Session Manager para conectarme a cualquier número de nodos activados de forma híbrida, incluso menos de 1000.	Sí	Sí
1. El número de nodos activados de forma híbrida de una región en una cuenta específica es de 1000 o menos; y 2. No voy a aplicar revisiones a aplicaciones de Microsoft en ningún nodo activado de forma híbrida; y 3. No me conecto a ningún nodo activado de forma híbrida mediante Session Manager.	No	No

Temas

- Activación del nivel de instancias avanzadas
- Revertir de la capa de instancias avanzadas a la capa de instancias estándar

Activación del nivel de instancias avanzadas

AWS Systems Manager ofrece un nivel de instancias estándar y un nivel de instancias avanzadas para las máquinas que no sean de EC2 en un entorno híbrido y multinube. El nivel de instancias estándar le permite registrar un máximo de 1000 máquinas activadas de forma híbrida por Cuenta de AWS por Región de AWS. También se requiere el nivel de instancias avanzadas para usar Patch Manager a fin de aplicar revisiones a aplicaciones lanzadas por Microsoft en nodos que no sean de EC2 y para conectarse a los mismos mediante Session Manager. Para obtener más información, consulte Activación del nivel de instancias avanzadas.

En esta sección se describe cómo configurar el entorno híbrido y multinube para utilizar el nivel de instancias avanzadas.

Antes de empezar

Revise la información sobre precios sobre el uso de instancias avanzadas. Las instancias avanzadas están disponibles según su uso. Para obtener más información, consulte Precios de AWS Systems Manager.

Configuración de permisos para activar el nivel de instancias avanzadas

Compruebe que tiene permiso en AWS Identity and Access Management (IAM) para cambiar su entorno desde el nivel de instancias estándar al nivel de instancias avanzadas. Para ello, debe tener la política de `AdministratorAccess` IAM adjunta al usuario, grupo o rol, o bien debe tener permiso para cambiar la configuración del servicio de nivel de activación de Systems Manager. La configuración del nivel de activación utiliza las siguientes operaciones de la API:

- GetServiceSetting
- UpdateServiceSetting
- ResetServiceSetting

Utilice el siguiente procedimiento para agregar una política de IAM en línea a una cuenta de usuario. Esta política le permite a un usuario ver la configuración actual del nivel de la instancia administrada.

Esta política también le permite al usuario cambiar o restablecer la configuración actual en la Cuenta de AWS y la Región de AWS especificada.

1. Inicie sesión en la AWS Management Console y abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, seleccione Usuarios.
3. En la lista, seleccione el nombre del usuario en el que incorporar una política.
4. Elija la pestaña Permisos.
5. En la parte derecha de la página, en Permission policies (Políticas de permisos), elija Add inline policy (Añadir política insertada).
6. Seleccione la pestaña JSON.
7. Reemplace el contenido predeterminado por lo siguiente:

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "arn:aws:ssm:us-east-1:111122223333:servicesetting/
ssm/managed-instance/activation-tier"
        }
    ]
}
```

8. Elija Revisar política.

9. En la página Review Policy (Revisar política), en Name (Nombre), escriba un nombre para la política insertada. Por ejemplo: **`Managed-Instances-Tier`**.
10. Seleccione Crear política.

Los administradores pueden especificar permiso de solo lectura al asignar la siguiente política insertada al usuario.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "*"
        }
    ]
}
```

Para obtener más información acerca de la creación y la edición de políticas de IAM, consulte Creación de políticas de IAM en la Guía del usuario de IAM.

Activación del nivel de instancias avanzadas (consola)

En el siguiente procedimiento, se muestra cómo utilizar la consola de Systems Manager para cambiar todos los nodos que no son de EC2 que se agregaron mediante la activación de instancias administradas, en la Cuenta de AWS y la Región de AWS especificadas, para que utilicen el nivel de instancias avanzadas.

Antes de empezar

Verifique que la consola esté abierta en la Región de AWS en la que ha creado las instancias administradas. Puede cambiar Regiones utilizando la lista en la parte superior, que se encuentra en la esquina superior derecha de la consola.

Verifique que haya completado los requisitos de configuración para las instancias de Amazon Elastic Compute Cloud (Amazon EC2) y las máquinas que no sean de EC2 en un entorno híbrido y multinube. Para obtener más información, consulta Configuración de nodos administrados para AWS Systems Manager.

> **Important**
>
> El siguiente procedimiento describe cómo cambiar una configuración de nivel de cuenta. Este cambio se traduce en cargos que se facturan a su cuenta.

Para habilitar el nivel de instancias avanzadas (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija Configuración y, a continuación, elija Cambiar la configuración de la capa de instancia.
4. Revise la información del cuadro de diálogo sobre el cambio de la configuración de la cuenta.
5. Si la aprueba, elija la opción que desee aceptar y, a continuación, elija Cambiar configuración.

El sistema puede tardar varios minutos en completar el proceso de trasladar todas las instancias de la capa de instancias estándar a la capa de instancias avanzadas.

> **Note**
>
> Para obtener información sobre cómo volver a cambiar al nivel de instancias estándar, consulte Revertir de la capa de instancias avanzadas a la capa de instancias estándar.

Activación del nivel de instancias avanzadas (AWS CLI)

En el siguiente procedimiento, se muestra cómo utilizar la AWS Command Line Interface para cambiar todos los servidores locales y las máquinas virtuales que se agregaron mediante la activación de instancias administradas, en la Cuenta de AWS y la Región de AWS especificadas, para que utilicen el nivel de instancias avanzadas.

> **Important**
>
> El siguiente procedimiento describe cómo cambiar una configuración de nivel de cuenta. Este cambio se traduce en cargos que se facturan a su cuenta.

Para activar el nivel de instancias avanzadas mediante la AWS CLI

1. Abra la AWS CLI y ejecute el siguiente comando. Reemplace cada *example resource placeholder* con su propia información.

   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
   ```

   No se obtienen resultados si el comando se ejecuta correctamente.

2. Ejecute el siguiente comando para ver la configuración del servicio actual para nodos administrados en la Cuenta de AWS y la Región de AWS actuales.

   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
```

El comando devuelve información similar a la siguiente.

```
{
    "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"
    }
}
```

Activación del nivel de instancias avanzadas (PowerShell)

En el siguiente procedimiento, se muestra cómo utilizar la AWS Tools for Windows PowerShell para cambiar todos los servidores locales y las máquinas virtuales que se agregaron mediante la activación de instancias administradas, en la Cuenta de AWS y la Región de AWS especificadas, para que utilicen el nivel de instancias avanzadas.

> **Important**
>
> El siguiente procedimiento describe cómo cambiar una configuración de nivel de cuenta. Este cambio se traduce en cargos que se facturan a su cuenta.

Para activar el nivel de instancias avanzadas con PowerShell

1. Abra AWS Tools for Windows PowerShell y ejecute el siguiente comando. Reemplace cada *example resource placeholder* con su propia información.

```
Update-SSMServiceSetting `
    -SettingId "arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
instance/activation-tier" `
    -SettingValue "advanced"
```

   No se obtienen resultados si el comando se ejecuta correctamente.

2. Ejecute el siguiente comando para ver la configuración del servicio actual para nodos administrados en la Cuenta de AWS y la Región de AWS actuales.

```
Get-SSMServiceSetting `
    -SettingId "arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
instance/activation-tier"
```

   El comando devuelve información similar a la siguiente.

```
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
```

El sistema puede tardar varios minutos en completar el proceso de trasladar todos los nodos del nivel de instancias estándar al nivel de instancias avanzadas.

> **Note**
>
> Para obtener información sobre cómo volver a cambiar al nivel de instancias estándar, consulte Revertir de la capa de instancias avanzadas a la capa de instancias estándar.

Revertir de la capa de instancias avanzadas a la capa de instancias estándar

En esta sección se describe cómo cambiar los nodos activados de manera híbrida que se ejecutan en el nivel de instancias avanzadas al nivel de instancias estándar. Esta configuración se aplica a todos los nodos activados de manera híbrida de una Cuenta de AWS y una única Región de AWS.

Antes de empezar

Revise los siguientes detalles importantes.

> **Note**
>
> - No puede volver al nivel de instancia estándar si está ejecutando más de 1000 nodos activados de manera híbrida en la cuenta y en la región. Primero debe anular el registro de nodos hasta que tenga 1000 o menos. Esto también se aplica a las instancias de Amazon Elastic Compute Cloud (Amazon EC2) que utilizan una activación híbrida de Systems Manager (lo que no es una situación común). Para obtener más información, consulte Anulación del registro de nodos administrados en un entorno híbrido y multinube.
> - Después de revertir, no podrá utilizar Session Manager, una herramienta de AWS Systems Manager, para acceder de forma interactiva a los nodos activados de manera híbrida.
> - Después de revertir, no podrá utilizar Patch Manager, una herramienta de AWS Systems Manager, para revisar las aplicaciones lanzadas por Microsoft en nodos activados de manera híbrida.
> - El proceso de revertir todos los nodos activados de manera híbrida al nivel de instancias estándar puede tardar 30 minutos o más en completarse.

En esta sección se describe cómo revertir todos los nodos activados de manera híbrida de una Cuenta de AWS y Región de AWS desde el nivel de instancias avanzadas al nivel de instancias estándar.

Reversión a la capa de instancias estándar (consola)

En el procedimiento siguiente se muestra cómo utilizar la consola de Systems Manager para cambiar todos los nodos activados de manera híbrida en el entorno híbrido y multinube para utilizar el nivel de instancias estándar de la Cuenta de AWS y la Región de AWS especificadas.

Para revertir al nivel de instancias estándar (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Seleccione el menú desplegable Account settings (Configuración de la cuenta) y elija Instance tier settings (Configuración del nivel de instancias).
4. Elija Change account setting (Cambiar la configuración de la cuenta).
5. Revise la información en la ventana emergente sobre cómo cambiar la configuración de la cuenta y, a continuación, si está de acuerdo, elija la opción para aceptar y continuar.

Reversión a la capa de instancias estándar (AWS CLI)

En el procedimiento siguiente se muestra cómo utilizar AWS Command Line Interface para cambiar todos los nodos activados de manera híbrida en el entorno híbrido y multinube para utilizar el nivel de instancias estándar de la Cuenta de AWS y la Región de AWS especificadas.

Para revertir al nivel de instancias estándar utilizando la AWS CLI

1. Abra la AWS CLI y ejecute el siguiente comando. Reemplace cada *example resource placeholder* con su propia información.

   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
   ```

   No se obtienen resultados si el comando se ejecuta correctamente.

2. Ejecute el siguiente comando 30 minutos más tarde para ver la configuración de instancias administradas en la Cuenta de AWS y la Región de AWS actuales.

   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
   ```

   El comando devuelve información similar a la siguiente.

   ```
   {
       "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"
       }
   }
   ```

   El estado cambia a Default (Predeterminado) una vez aprobada la solicitud.

Reversión a la capa de instancias estándar (PowerShell)

En el procedimiento siguiente se muestra cómo utilizar AWS Tools for Windows PowerShell para cambiar todos los nodos activados de manera híbrida en el entorno híbrido y multinube para utilizar el nivel de instancias estándar de la Cuenta de AWS y la Región de AWS especificadas.

Para revertir al nivel de instancias estándar mediante PowerShell

1. Abra AWS Tools for Windows PowerShell y ejecute el siguiente comando.

```
Update-SSMServiceSetting `
    -SettingId "arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
instance/activation-tier" `
    -SettingValue "standard"
```

   No se obtienen resultados si el comando se ejecuta correctamente.

2. Ejecute el siguiente comando 30 minutos más tarde para ver la configuración de instancias administradas en la Cuenta de AWS y la Región de AWS actuales.

```
Get-SSMServiceSetting `
    -SettingId "arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
instance/activation-tier"
```

   El comando devuelve información similar a la siguiente.

```
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
```

   El estado cambia a Default (Predeterminado) una vez aprobada la solicitud.

Restablecimiento de contraseñas en nodos administrados

Puede restablecer la contraseña para cualquier usuario en un nodo administrado. Esto incluye instancias de Amazon Elastic Compute Cloud (Amazon EC2), servidores de núcleo de AWS IoT Greengrass, servidores locales, dispositivos de borde y máquinas virtuales administradas por AWS Systems Manager. La función de restablecimiento de contraseña se basa en Session Manager, una herramienta de AWS Systems Manager. Puede utilizar esta funcionalidad para conectarse a los nodos administrados sin abrir los puertos de entrada, mantener hosts bastión ni administrar claves SSH.

El restablecimiento de contraseña resulta útil cuando un usuario ha olvidado una contraseña, o cuando desea actualizar rápidamente una contraseña sin realizar una conexión RDP o SSH al nodo administrado.

Requisitos previos

Antes de que pueda restablecer la contraseña en un nodo administrado, deben cumplirse los siguientes requisitos:

- El nodo administrado en el que desee cambiar una contraseña debe ser un nodo administrado de Systems Manager. Además, la versión 2.3.668.0 de SSM Agent o una versión posterior debe estar instalada en el nodo administrado. Para obtener información acerca cómo instalar o actualizar SSM Agent, consulte Uso de SSM Agent.
- La funcionalidad de restablecimiento de contraseñas utiliza la configuración de Session Manager, la cual está establecida de forma que su cuenta se conecte al nodo administrado. Por lo tanto, los requisitos previos para utilizar Session Manager deben haberse completado en su cuenta en la Región de AWS actual. Para obtener más información, consulte Configuración de Session Manager.

  > **Note**
  >
  > La compatibilidad de Session Manager con nodos en las instalaciones se proporciona solo para el nivel de instancias avanzadas. Para obtener más información, consulte Activación del nivel de instancias avanzadas.

- El usuario AWS que está cambiando la contraseña debe tener el permiso `ssm:SendCommand` para el nodo administrado. Para obtener más información, consulte Restricción de acceso de Run Command basado en etiquetas.

Restricción del acceso

Puede limitar la capacidad de un usuario para restablecer contraseñas para nodos administrados específicos. Esto se hace mediante políticas basadas en la identidad para la operación Session Manager de `ssm:StartSession` con el documento `AWS-PasswordReset` de SSM. Para obtener más información, consulte Control del acceso de las sesiones de usuario a las instancias.

Cifrado de datos

Active el cifrado completo de AWS Key Management Service (AWS KMS) para los datos de Session Manager a fin de utilizar la opción de restablecimiento de contraseña de los nodos administrados. Para obtener más información, consulte Activación del cifrado de datos de sesión con claves de KMS (consola).

Restablecer una contraseña en un nodo administrado

Puede restablecer una contraseña en un nodo administrado de Systems Manager mediante la consola Fleet Manager de Systems Manager o AWS Command Line Interface (AWS CLI).

Para cambiar la contraseña en un nodo administrado (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija el botón situado al lado del nodo que necesita una nueva contraseña.
4. Seleccione Acciones de instancia, Restablecer la contraseña.
5. En el campo User name (Nombre de usuario), ingrese el nombre del usuario para el que va a cambiar la contraseña. Este puede ser cualquier nombre de usuario con una cuenta en el nodo.
6. Elija Enviar.
7. Siga las instrucciones en la ventana de comandos Introducir nueva contraseña para especificar la nueva contraseña.

> **Note**
>
> Si la versión de SSM Agent en el nodo administrado no admite restablecimientos de contraseñas, se le pedirá que instale una versión compatible con Run Command, una herramienta de AWS Systems Manager.

Para restablecer la contraseña en un nodo administrado (AWS CLI)

1. Para restablecer la contraseña de un usuario en un nodo administrado, ejecute el siguiente comando. Reemplace cada *example resource placeholder* con su propia información.

> **Note**
>
> Para utilizar la AWS CLI para restablecer una contraseña, el complemento Session Manager debe estar instalado en su equipo local. Para obtener más información, consulta Instalación del complemento de Session Manager para la 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. Siga las instrucciones en la ventana de comandos Introducir nueva contraseña para especificar la nueva contraseña.

Solucionar problemas de restablecimiento de contraseña en nodos administrados

Muchos problemas de restablecimiento de contraseña pueden resolverse; para ello, asegúrese de que ha completado los requisitos previos de restablecimiento de contraseña. Para otros problemas, utilice la siguiente información para ayudarle a solucionar problemas para restablecer la contraseña.

Temas

- Nodo administrado no disponible
- SSM Agent no actualizada (consola)
- No se proporcionan las opciones de restablecimiento de contraseña (AWS CLI).
- No hay autorización para ejecutar ssm:SendCommand
- Mensaje de error Session Manager

Nodo administrado no disponible

Problema: desea restablecer la contraseña de un nodo administrado en la página de la consola de Instancias administradas, pero el nodo no se encuentra en la lista.

- Solución: el nodo administrado al que desea conectarse podría no estar configurado para Systems Manager. Para utilizar una instancia de EC2 con Systems Manager, el perfil de instancias de AWS Identity and Access Management (IAM) que concede permiso a Systems Manager para realizar acciones en sus instancias debe estar adjuntado a la instancia. Para obtener información, consulte Configuración de permisos de instancia requeridos para Systems Manager.

  Para utilizar una máquina que no es de EC2 con Systems Manager, debe crear un rol de servicio de IAM que le conceda permiso a Systems Manager para realizar acciones en los nodos administrados. Para obtener más información, consulte Creación del rol de servicio de IAM requerido para Systems Manager en entornos híbridos y multinube. (Solo se ofrece compatibilidad de Session Manager con servidores locales y máquinas virtuales para el nivel de instancias avanzadas. Para obtener más información, consulte Activación del nivel de instancias avanzadas.)

SSM Agent no actualizada (consola)

Problema: un mensaje informa que la versión de SSM Agent no admite la función de restablecimiento de contraseñas.

- Solución: se requiere la versión 2.3.668.0 o una versión posterior de SSM Agent para realizar restablecimientos de contraseñas. En la consola, puede actualizar el agente en el nodo administrado al elegir Update SSM Agent (Actualizar SSM Agente).

  Cada vez que se agregan herramientas nuevas a Systems Manager o se actualizan las herramientas existentes, se lanza una versión actualizada de SSM Agent. No utilizar la versión más reciente del agente puede impedir que el nodo administrado utilice diversas herramientas y características de Systems Manager. Por este motivo, se recomienda automatizar el proceso de mantener SSM Agent actualizado en los equipos. Para obtener más información, consulta Automatización de las actualizaciones de SSM Agent. Suscríbase a la página SSM Agent Release Notes en GitHub para recibir notificaciones sobre las actualizaciones de SSM Agent.

No se proporcionan las opciones de restablecimiento de contraseña (AWS CLI).

Problema: puede conectarse correctamente a un nodo administrado mediante el comando AWS CLI `start-session`. Ha especificado el documento de SSM `AWS-PasswordReset` y ha proporcionado un nombre de usuario válido, pero no se muestran las instrucciones para cambiar la contraseña.

- Solución: la versión de SSM Agent en el nodo administrado no está actualizada. Es necesaria la versión 2.3.668.0 o una versión posterior para realizar restablecimientos de contraseña.

  Cada vez que se agregan herramientas nuevas a Systems Manager o se actualizan las herramientas existentes, se lanza una versión actualizada de SSM Agent. No utilizar la versión más reciente del agente puede impedir que el nodo administrado utilice diversas herramientas y características de Systems Manager. Por este motivo, se recomienda automatizar el proceso de mantener SSM Agent actualizado en los equipos. Para obtener más información, consulta Automatización de las actualizaciones de SSM Agent. Suscríbase a la página SSM Agent Release Notes en GitHub para recibir notificaciones sobre las actualizaciones de SSM Agent.

No hay autorización para ejecutar **`ssm:SendCommand`**

Problema: intenta conectarse a un nodo administrado para cambiar la contraseña, pero recibe un mensaje de error que indica que no cuenta con la autorización para ejecutar `ssm:SendCommand` en el nodo administrado.

- Solución: la política de IAM debe incluir un permiso para ejecutar el comando `ssm:SendCommand`. Para obtener más información, consulta Restricción de acceso de Run Command basado en etiquetas.

Mensaje de error Session Manager

Problema: recibe un mensaje de error relacionado con Session Manager.

- Solución: el soporte para restablecer la contraseña requiere que Session Manager esté configurado correctamente. Para obtener más información, consulte Configuración de Session Manager y Solución de problemas de Session Manager.

Anulación del registro de nodos administrados en un entorno híbrido y multinube

Si ya no desea administrar un servidor local, un dispositivo de borde o una máquina virtual mediante AWS Systems Manager, puede anular el registro. Anular el registro de un nodo activado de manera

híbrida lo elimina de la lista de nodos administrados en Systems Manager. AWS Systems Manager El agente (SSM Agent) que se está ejecutando en el nodo activado de manera híbrida no podrá actualizar su token de autorización porque ya no está registrado. SSM Agent hiberna y reduce su frecuencia de ping a Systems Manager en la nube a una vez por hora. Systems Manager almacena el historial de comandos de un nodo administrado con el registro anulado durante 30 días.

Si desea volver a registrar un servidor en las instalaciones, un dispositivo de periferia o una VM, debe usar un código de activación y un ID de activación diferentes a los que se usaron para registrar la máquina anteriormente. El código y el ID de activación no deben haberse utilizado ya en el número máximo de activaciones especificado cuando se crearon.

El procedimiento siguiente describe cómo anular el registro de un nodo activado de manera híbrida mediante la consola de Systems Manager. Para obtener información acerca de cómo hacer esto mediante la AWS Command Line Interface, consulte deregister-managed-instance.

Para obtener información relacionada, consulte los siguientes temas:

- Anulación del registro y nuevo registro de un nodo administrado (Linux) (Linux)
- Anulación del registro y nuevo registro de un nodo administrado (Windows Server) (Windows Server)

Anulación del registro de un nodo activado de manera híbrida (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Seleccione la casilla de verificación situada junto al nodo administrado cuyo registro desea anular.
4. Seleccione Acciones de nodos, Herramientas, Anular el registro de este nodo administrado.
5. Revise la información del cuadro de diálogo Anular el registro de este nodo administrado. Si está de acuerdo, seleccione Anular registro.

Utilizar Fleet Manager para trabajar con sistemas de archivos del sistema operativo

Puede utilizar Fleet Manager, una herramienta de AWS Systems Manager, para trabajar con el sistema de archivos en los nodos administrados. Al utilizar Fleet Manager, puede visualizar la información sobre los datos de archivos y de directorios almacenados en los volúmenes adjuntados a

los nodos administrados. Por ejemplo, puede ver el nombre, el tamaño, la extensión, el propietario y los permisos de sus directorios y archivos. Se puede obtener una vista previa de hasta 10 000 líneas de datos de archivo como texto desde la consola de Fleet Manager. También puede utilizar esta característica para `tail` (poner en cola) archivos. Cuando se utiliza `tail` para ver los datos del archivo, las últimas 10 líneas del archivo se muestran al inicio. A medida que se escriben nuevas líneas de datos en el archivo, la vista se actualiza en tiempo real. Como resultado, puede revisar los datos de registro desde la consola, lo que puede mejorar la eficiencia de la solución de problemas y la administración de sistemas. Además, puede crear directorios y copiar, cortar, pegar, cambiar el nombre o eliminar archivos y directorios.

Se recomienda que cree copias de seguridad con regularidad o que tome instantáneas de los volúmenes de Amazon Elastic Block Store (Amazon EBS) adjuntados a los nodos administrados. Al copiar, cortar o pegar archivos, se sustituyen los archivos y directorios existentes en la ruta de destino con el mismo nombre que los archivos o directorios nuevos. Pueden producirse problemas graves si reemplaza o modifica los archivos y directorios del sistema. AWS no garantiza que estos problemas se puedan resolver. Modifique los archivos del sistema bajo su propia responsabilidad. El usuario es responsable de todos los cambios en los archivos y directorios, y de asegurarse de tener copias de seguridad. No se puede deshacer la eliminación o la sustitución de archivos y directorios.

> **Note**
>
> Fleet Manager utiliza Session Manager, una herramienta de AWS Systems Manager, para obtener vistas previas de texto y archivos `tail`. Para las instancias de Amazon Elastic Compute Cloud (Amazon EC2), el perfil de instancias adjuntado a las instancias administradas debe proporcionar permisos para que Session Manager utilice esta característica. Para obtener más información acerca de cómo agregar permisos de Session Manager al perfil de instancias, consulte Adición de permisos de Session Manager a un rol de IAM existente.

Temas

- Utilizar Fleet Manager para visualizar el sistema de archivos del sistema operativo
- Utilizar Fleet Manager para obtener una vista previa de los archivos del sistema operativo
- Utilizar Fleet Manager para seguir archivos del sistema operativo
- Utilizar Fleet Manager para copiar, cortar y pegar archivos o directorios del sistema operativo
- Utilizar Fleet Manager para cambiar el nombre de archivos y directorios del sistema operativo

- Utilizar Fleet Manager para eliminar archivos y directorios del sistema operativo
- Utilizar Fleet Manager para crear directorios del sistema operativo
- Utilizar Fleet Manager para cortar, copiar y pegar directorios del sistema operativo

Utilizar Fleet Manager para visualizar el sistema de archivos del sistema operativo

Se puede utilizar Fleet Manager para visualizar el sistema de archivos del sistema operativo en un nodo administrado de Systems Manager.

Para utilizar Fleet Manager para visualizar el sistema de archivos del sistema operativo

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Seleccione el enlace del nodo administrado con el sistema de archivos que desea ver.
4. Elija Herramientas, Sistema de archivos.

Utilizar Fleet Manager para obtener una vista previa de los archivos del sistema operativo

Puede utilizar Fleet Manager para obtener una vista previa de los archivos de texto en un sistema operativo.

Para utilizar Fleet Manager para obtener vistas previas de texto de archivos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Seleccione el enlace del nodo administrado con los archivos de los que desea obtener una vista previa.
4. Elija Herramientas, Sistema de archivos.
5. Seleccione el File name (Nombre de archivo) del directorio que contiene el archivo que desea previsualizar.
6. Elija el botón situado al lado del archivo cuyo contenido desea previsualizar.
7. Elija Acciones, Obtener vista previa como texto.

Utilizar Fleet Manager para seguir archivos del sistema operativo

Puede utilizar Fleet Manager para seguir un archivo en un nodo administrado.

Para seguir archivos del sistema operativo con Fleet Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Seleccione el enlace del nodo administrado con los archivos que desea poner en cola.
4. Elija Herramientas, Sistema de archivos.
5. Seleccione el File name (Nombre de archivo) del directorio que contiene el archivo que desea poner en cola.
6. Elija el botón situado al lado del archivo cuyo contenido desea poner en cola.
7. Elija Acciones, Obtener final de archivos.

Utilizar Fleet Manager para copiar, cortar y pegar archivos o directorios del sistema operativo

Se puede utilizar Fleet Manager para copiar, cortar y pegar archivos del sistema operativo en un nodo administrado.

Para utilizar Fleet Manager para copiar o cortar y pegar archivos o directorios

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Seleccione el enlace del nodo administrado con los archivos que desea copiar, cortar o pegar.
4. Elija Herramientas, Sistema de archivos.
5. Para copiar o cortar un archivo, seleccione la opción File name (Nombre del archivo) del directorio que contiene el archivo que desea copiar o cortar. Para copiar o cortar un directorio, elija el botón situado junto al directorio que desea copiar o cortar y, a continuación, proceda al paso 8.
6. Elija el botón situado al lado del archivo cuyo contenido desea copiar o cortar.
7. En el menú Actions (Acciones), elija Copy (Copiar) o Cut (Cortar).
8. En la vista File system (Sistema de archivos), elija el botón que aparece junto al directorio en el que desea pegar el archivo.

9. En el menú Actions (Acciones), elija Paste (Pegar).

Utilizar Fleet Manager para cambiar el nombre de archivos y directorios del sistema operativo

Puede utilizar Fleet Manager para cambiar el nombre de archivos y directorios en un nodo administrado de la cuenta.

Para cambiar el nombre de los archivos o directorios con Fleet Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Seleccione el enlace del nodo administrado con los archivos o directorios que desea renombrar.
4. Elija Herramientas, Sistema de archivos.
5. Para cambiar el nombre de un archivo, seleccione el File name (Nombre del archivo) del directorio que contiene el archivo que desea cambiar el nombre. Para cambiar el nombre de un directorio, elija el botón situado junto al directorio al que desea cambiar el nombre y, a continuación, continúe con el paso 8.
6. Elija el botón situado al lado del archivo cuyo contenido desea cambiarle el nombre
7. Selecciona Acciones, Cambiar nombre.
8. En Nombre del archivo, escriba el nombre nuevo del archivo y seleccione Cambiar nombre.

Utilizar Fleet Manager para eliminar archivos y directorios del sistema operativo

Puede utilizar Fleet Manager para eliminar archivos y directorios en un nodo administrado de la cuenta.

Para utilizar Fleet Manager para eliminar archivos o directorios

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Seleccione el enlace del nodo administrado con los archivos o directorios que desea eliminar.
4. Elija Herramientas, Sistema de archivos.

5. Para eliminar un archivo, seleccione la File name (Nombre del archivo) del directorio que contiene el archivo que desea eliminar. Para eliminar un directorio, elija el botón que aparece junto al directorio que desea eliminar y, a continuación, vaya al paso 7.
6. Elija el botón situado al lado del archivo cuyo contenido desea eliminar.
7. Elija Acciones, Eliminar.

Utilizar Fleet Manager para crear directorios del sistema operativo

Puede utilizar Fleet Manager para crear directorios en un nodo administrado de la cuenta.

Para utilizar Fleet Manager para crear un directorio

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Seleccione el enlace del nodo administrado en el que desea crear un directorio.
4. Elija Herramientas, Sistema de archivos.
5. Seleccione el File name (Nombre del archivo) del directorio en el que desea crear un nuevo directorio.
6. Seleccione Create directory (Crear directorio).
7. En el campo Nombre del directorio, escriba el nombre del directorio nuevo y seleccione Crear directorio.

Utilizar Fleet Manager para cortar, copiar y pegar directorios del sistema operativo

Puede utilizar Fleet Manager para cortar, copiar y pegar directorios en un nodo administrado en la cuenta.

Para utilizar Fleet Manager para copiar o cortar y pegar directorios

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Seleccione el enlace del nodo administrado con los archivos que desea copiar, cortar o pegar.
4. Elija Herramientas, Sistema de archivos.

5. Elija el botón situado junto al directorio que desea copiar o cortar y, a continuación, proceda al paso 8.
6. En el menú Actions (Acciones), elija Copy (Copiar) o Cut (Cortar).
7. En la vista File system (Sistema de archivos), elija el botón que aparece junto al directorio en el que desea pegar el archivo.
8. En el menú Actions (Acciones), elija Paste (Pegar).

Monitoreo del rendimiento de nodos administrados

Puede utilizar Fleet Manager, una herramienta de AWS Systems Manager, para ver los datos de rendimiento de los nodos administrados en tiempo real. Los datos de rendimiento se recuperan de los contadores de rendimiento.

Los siguientes contadores de rendimiento están disponibles en Fleet Manager:

- Utilización de la CPU
- Utilización de entrada/salida (E/S) del disco
- Tráfico de red
- Uso de memoria

> **Note**
>
> Fleet Manager utiliza Session Manager, una herramienta de AWS Systems Manager, para recuperar datos de rendimiento. Para las instancias de Amazon Elastic Compute Cloud (Amazon EC2), el perfil de instancias adjuntado a las instancias administradas debe proporcionar permisos para que Session Manager utilice esta característica. Para obtener más información acerca de cómo agregar permisos de Session Manager al perfil de instancias, consulte Adición de permisos de Session Manager a un rol de IAM existente.

Para ver los datos de rendimiento con Fleet Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija el botón situado junto al nodo administrado cuyo rendimiento desea monitorear.

4. Elija Ver detalles.
5. Elija Herramientas, Contadores de rendimiento.

Trabajo con procesos

Puede utilizar Fleet Manager, una herramienta de AWS Systems Manager, para trabajar con procesos en los nodos administrados. Mediante el uso de Fleet Manager, puede ver la información de los procesos. Por ejemplo, puede ver la utilización de la CPU y el uso de memoria de los procesos, además de sus controladores y subprocesos. Con Fleet Manager, puede iniciar y terminar procesos desde la consola.

> **Note**
>
> Fleet Manager utiliza Session Manager, una herramienta de AWS Systems Manager, para recuperar los datos del proceso. Para las instancias de Amazon Elastic Compute Cloud (Amazon EC2), el perfil de instancias adjuntado a las instancias administradas debe proporcionar permisos para que Session Manager utilice esta característica. Para obtener más información acerca de cómo agregar permisos de Session Manager al perfil de instancias, consulte Adición de permisos de Session Manager a un rol de IAM existente.

Temas

- Utilizar Fleet Manager para visualizar los detalles sobre los procesos del sistema operativo
- Utilizar Fleet Manager para iniciar un proceso del sistema operativo en un nodo administrado
- Utilizar Fleet Manager para terminar un proceso del sistema operativo

Utilizar Fleet Manager para visualizar los detalles sobre los procesos del sistema operativo

Puede utilizar Fleet Manager para visualizar los detalles sobre los procesos en los nodos administrados.

Para ver los detalles de los procesos con Fleet Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Seleccione el enlace del nodo cuyos procesos desea visualizar.

4. Elija Herramientas, Procesos.

Utilizar Fleet Manager para iniciar un proceso del sistema operativo en un nodo administrado

Puede utilizar Fleet Manager para iniciar un proceso en un nodo administrado.

Para iniciar un proceso con Fleet Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Seleccione el enlace del nodo administrado en el que desea iniciar un proceso.
4. Elija Herramientas, Procesos.
5. Seleccione Start new process (Iniciar un proceso nuevo).
6. En el campo Nombre del proceso o ruta completa, escriba el nombre del proceso o la ruta completa del ejecutable.
7. (Opcional) En el campo Directorio de trabajo, ingrese la ruta del directorio en la que desea que se ejecute el proceso.

Utilizar Fleet Manager para terminar un proceso del sistema operativo

Para utilizar Fleet Manager para terminar un proceso del sistema operativo

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Seleccione el enlace del nodo administrado en el que desea iniciar un proceso.
4. Elija Herramientas, Procesos.
5. Seleccione el botón situado junto al proceso que desea terminar.
6. Elija Acciones, Terminar proceso o Acciones, Terminar árbol de procesos.

> **Note**
>
> Terminar un árbol de procesos también finaliza todos los procesos y aplicaciones que utilizan ese proceso.

Visualización de registros en nodos administrados

Puede utilizar Fleet Manager, una herramienta de AWS Systems Manager, para ver los datos de registro almacenados en los nodos administrados. En los nodos administrados de Windows, puede ver los registros de eventos de Windows y copiar sus detalles desde la consola. Para ayudarlo a buscar eventos, filtre los registros de eventos de Windows por Event level (Nivel de evento), Event ID (ID de evento), Event source (Origen del evento), y Time created (Hora creada). También puede ver otros datos de registro mediante el procedimiento para ver el sistema de archivos. Para obtener más información acerca de cómo ver el sistema de archivos con Fleet Manager, consulte Utilizar Fleet Manager para trabajar con sistemas de archivos del sistema operativo.

Para ver los registros de eventos de Windows con Fleet Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija el botón situado junto al nodo administrado cuyos registros de eventos desea ver.
4. Elija Ver detalles.
5. Elija Herramientas, Registros de eventos de Windows.
6. Elija la opción de Log name (Nombre de registro) que contiene los eventos que desea ver.
7. Elija el botón situado al lado de Log name (Nombre de registro) que desea ver y, a continuación, seleccione View eventos (Ver eventos).
8. Elija el botón situado al lado del evento que desea ver y, a continuación, seleccione View event details (Ver detalles del evento).
9. (Opcional) Seleccione Copy as JSON (Copiar como JSON) para copiar los detalles del evento en el portapapeles.

Utilizar Fleet Manager para administrar cuentas de usuario y grupos del sistema operativo en nodos administrados

Puede utilizar Fleet Manager, una herramienta de AWS Systems Manager, para administrar las cuentas de usuario y los grupos del sistema operativo (SO) de los nodos administrados. Por ejemplo, puede crear y eliminar usuarios y grupos. Además, puede ver detalles como, por ejemplo, pertenencia a grupos, roles de usuario y estado.

> **Important**
>
> Fleet Manager utiliza Run Command y Session Manager, herramientas de AWS Systems Manager, para diversas operaciones de administración de usuarios. Como resultado, un usuario podría conceder permisos a una cuenta de usuario del sistema operativo que no podría conceder de otro modo. Esto es porque AWS Systems Manager Agent (SSM Agent) se ejecuta en las instancias de Amazon Elastic Compute Cloud (Amazon EC2) mediante permisos raíz (Linux) o permisos SYSTEM (Windows Server). Para obtener más información sobre cómo restringir el acceso a los comandos de nivel de raíz mediante SSM Agent, consulte Restricción del acceso a los comandos de nivel raíz con SSM Agent. Para restringir el acceso a esta característica, recomendamos crear políticas de AWS Identity and Access Management (IAM) para los usuarios que solo permiten el acceso a las acciones que defina. Para obtener más información acerca de la creación de políticas de IAM para Fleet Manager, consulte Control del acceso a Fleet Manager.

Temas

- Utilizar Fleet Manager para crear un usuario o grupo del sistema operativo
- Utilizar Fleet Manager para actualizar la pertenencia a un usuario o grupo
- Utilizar Fleet Manager para eliminar un usuario o grupo de sistema operativo

Utilizar Fleet Manager para crear un usuario o grupo del sistema operativo

> **Note**
>
> Fleet Manager utiliza Session Manager para establecer contraseñas para nuevos usuarios. Para las instancias de Amazon EC2, el perfil de instancias adjuntado a las instancias administradas debe proporcionar permisos para que Session Manager utilice esta característica. Para obtener más información acerca de cómo agregar permisos de Session Manager al perfil de instancias, consulte Adición de permisos de Session Manager a un rol de IAM existente.

En lugar de conectarse directamente a un servidor para crear una cuenta de usuario o grupo, puede utilizar la consola de Fleet Manager para realizar las mismas tareas.

Para utilizar Fleet Manager para crear una cuenta de usuario del sistema operativo

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija el botón situado junto al nodo administrado en el que desea crear un nuevo usuario.
4. Elija Ver detalles.
5. Elija Herramientas, Usuarios y grupos.
6. Elija la pestaña Users (Usuarios) y, a continuación, elija Create user (Crear usuario).
7. Ingrese un valor para la propiedad Name (Nombre) del nuevo usuario.
8. (Recomendado) Seleccione la casilla de verificación situada junto a Set password (Establecer la contraseña). Al final del procedimiento, se le pedirá que especifique una contraseña para el nuevo usuario.
9. Seleccione Create user (Crear usuario). Si seleccionó la casilla de verificación para crear una contraseña para el nuevo usuario, se le pedirá que ingrese un valor para la contraseña y seleccione Done (Hecho). Si la contraseña especificada no cumple los requisitos especificados por las políticas locales o de dominio del nodo administrado, se devuelve un error.

Para utilizar Fleet Manager para crear un grupo del sistema operativo

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija el botón situado junto al nodo administrado en el que desea crear un nuevo grupo.
4. Elija Ver detalles.
5. Elija Herramientas, Usuarios y grupos.
6. Elija la pestaña Groups (Grupos) y, a continuación, elija Create group (Crear grupo).
7. Ingrese un valor para la propiedad Name (Nombre) del nuevo grupo.
8. (Opcional) Ingrese un valor para la propiedad Description (Descripción) del nuevo grupo.
9. (Opcional) Seleccione los usuarios que desea agregar a la propiedad Group members (Miembros del grupo) para el nuevo grupo.
10. Seleccione Create group (Crear grupo).

Utilizar Fleet Manager para actualizar la pertenencia a un usuario o grupo

En lugar de conectarse directamente a un servidor para actualizar una cuenta de usuario o grupo, puede utilizar la consola de Fleet Manager para realizar las mismas tareas.

Para utilizar Fleet Manager para agregar una cuenta de usuario del sistema operativo a un nuevo grupo

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija el botón situado junto al nodo administrado en el que existe la cuenta de usuario que desea actualizar.
4. Elija Ver detalles.
5. Elija Herramientas, Usuarios y grupos.
6. Elija la pestaña Users.
7. Elija el botón situado al lado del usuario que desea actualizar.
8. Elija Acciones, Agregar usuario al grupo.
9. Elija el grupo al que desea agregar el usuario en Add to group (Agregar al grupo).
10. Seleccione Add user to group (Agregar usuario al grupo).

Para utilizar Fleet Manager para editar la pertenencia a un grupo del sistema operativo

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija el botón situado junto al nodo administrado en el que existe el grupo que desea actualizar.
4. Elija Ver detalles.
5. Elija Herramientas, Usuarios y grupos.
6. Seleccione la pestaña Groups (Grupos).
7. Elija el botón situado al lado del grupo que desea actualizar.
8. Elija Acciones, Modificar grupo.
9. Elija los usuarios que desea agregar o eliminar en Group members (Miembros del grupo).
10. Seleccione Modify group (Modificar grupo).

Utilizar Fleet Manager para eliminar un usuario o grupo de sistema operativo

En lugar de conectarse directamente a un servidor para eliminar una cuenta de usuario o grupo, puede utilizar la consola de Fleet Manager para realizar las mismas tareas.

Para utilizar Fleet Manager para eliminar una cuenta de usuario del sistema operativo

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija el botón situado junto al nodo administrado en el que existe la cuenta de usuario que desea eliminar.
4. Elija Ver detalles.
5. Elija Usuarios y grupos.
6. Elija la pestaña Users.
7. Elija el botón situado al lado del usuario que desea eliminar.
8. Elija Acciones, Eliminar usuario local.

Utilizar Fleet Manager para eliminar un grupo del sistema operativo

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija el botón situado junto al nodo administrado en el que existe el grupo que desea eliminar.
4. Elija Ver detalles.
5. Elija Herramientas, Usuarios y grupos.
6. Elija la pestaña Groups (Grupos).
7. Elija el botón situado al lado del grupo que desea actualizar.
8. Elija Acciones, Eliminar grupo local.

Administración del registro de Windows en nodos administrados

Puede utilizar Fleet Manager, una herramienta de AWS Systems Manager, para administrar el registro en los nodos administrados de Windows Server. Desde la consola de Fleet Manager puede crear, copiar, actualizar y eliminar entradas y valores del registro.

> **Important**
>
> Recomendamos crear una copia de seguridad del registro o tomar una instantánea del volumen raíz de Amazon Elastic Block Store (Amazon EBS) adjuntado al nodo administrado antes de modificar el registro. Pueden producirse problemas graves si modifica el registro de manera incorrecta. Estos problemas pueden requerir que vuelva a instalar el sistema operativo o que restaure el volumen raíz del nodo desde una instantánea. AWS no garantiza que estos problemas puedan ser resueltos. Modifique el registro por su cuenta y riesgo. Usted es responsable de todos los cambios en el registro y de asegurarse de tener copias de seguridad.

Creación de una clave o entrada del registro de Windows

Para crear una clave del registro de Windows con Fleet Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija el botón situado junto al nodo administrado en el que desea crear una clave de registro.
4. Elija Ver detalles.
5. Elija Herramientas, Registro de Windows.
6. Elija el hive en el que desea crear una nueva clave de registro seleccionando la opción Registry name (Nombre de registro).
7. Elija Crear, Crear clave del registro.
8. Elija el botón situado al lado de la entrada de registro en la que desea crear una nueva clave.
9. Elija Create registry key (Crear clave del registro).
10. Ingrese un valor para la propiedad Name (Nombre) de la nueva clave de registro y seleccione Submit (Enviar).

Para crear una entrada de registro de Windows con Fleet Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.

3. Elija el botón situado al lado de la instancia en la que desea crear una entrada de registro.
4. Elija Ver detalles.
5. Elija Herramientas, Registro de Windows.
6. Elija el hive y la clave de registro siguiente en la que desea crear una nueva entrada de registro seleccionando la propiedad Registry name (Nombre de registro).
7. Elija Crear, Crear entrada del registro.
8. Ingrese un valor en Name (Nombre) para el registro nuevo.
9. Elija la característica Type (Tipo) de valor que desea crear para la entrada de registro. Para obtener más información acerca de los tipos de valores de registro, consulte Tipos de valores de registro.
10. Ingrese un valor para la propiedad Valor (Valor) de la nueva entrada de registro.

Actualización de una entrada de registro de Windows

Para actualizar una entrada de registro de Windows con Fleet Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija el botón situado junto al nodo administrado en el que desea actualizar una entrada del registro.
4. Elija Ver detalles.
5. Elija Herramientas, Registro de Windows.
6. Elija el hive y la clave de registro siguiente que desea actualizar seleccionando la casilla de Registry name (Nombre de registro).
7. Elija el botón situado al lado de la entrada de registro que desea actualizar.
8. Elija Acciones, Actualizar entrada de registro.
9. Ingrese el nuevo valor para Value (Valor) de la entrada del registro.
10. Elija Actualizar.

Eliminación de una entrada o clave de registro de Windows

Para eliminar una clave de registro de Windows con Fleet Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija el botón situado junto al nodo administrado en el que desea eliminar una clave de registro.
4. Elija Herramientas, Registro de Windows.
5. Seleccione el hive y la clave de registro posterior que desea eliminar seleccionando la casilla de Registry name (Nombre de registro).
6. Elija el botón situado al lado de la clave de registro que desea eliminar.
7. Elija Acciones, Eliminar clave de registro.

Para eliminar una entrada de registro de Windows con Fleet Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija el botón situado junto al nodo administrado en el que desea eliminar una entrada de registro.
4. Elija Ver detalles.
5. Elija Herramientas, Registro de Windows.
6. Elija el hive y la clave de registro siguiente que contiene la entrada que desea eliminar seleccionando la casilla de Registry name (Nombre de registro).
7. Elija el botón situado al lado de la entrada de registro que desea eliminar.
8. Elija Acciones, Eliminar entrada de registro.

Administración automática de instancias EC2 con la configuración de administración de hosts predeterminada

La configuración predeterminada de la administración de hosts permite que AWS Systems Manager administre las instancias de Amazon EC2 de forma automática como instancias administradas. Una instancia administrada es una instancia EC2 configurada para usarla con Systems Manager.

Los beneficios de administrar sus instancias con Systems Manager incluyen los siguientes:

- Conéctese a sus instancias de EC2 de forma segura mediante Session Manager.
- Realice escaneos de revisiones automatizados mediante Patch Manager.
- Consulte información detallada sobre sus instancias mediante Systems Manager Inventory.
- Realice un seguimiento y administre las instancias mediante Fleet Manager.
- Mantenga SSM Agent actualizado automáticamente.

Fleet Manager, Inventario, Patch Manager y Session Manager son herramientas de Systems Manager.

Si usa la configuración de administración de host predeterminada, puede administrar las instancias de EC2 sin tener que crear manualmente un perfil de instancia de AWS Identity and Access Management (IAM). En su lugar, la configuración predeterminada de la administración de hosts crea y aplica un rol de IAM predeterminado para garantizar que Systems Manager tenga los permisos de administración de todas las instancias de la Cuenta de AWS y la Región de AWS en la que está activado.

Si los permisos proporcionados no son suficientes para su caso de uso, también puede agregar políticas al rol de IAM predeterminado creado en la configuración de administración de host predeterminada. Como alternativa, si no necesita permisos para todas las capacidades proporcionadas por el rol de IAM predeterminado, puede crear sus propias políticas y roles personalizados. Cualquier cambio realizado en el rol de IAM que elija para la configuración de administración de host predeterminada se aplica a todas las instancias de Amazon EC2 administradas en la región y la cuenta.

Para obtener más información acerca de la política que usa la configuración de administración de host predeterminada, consulte Política administrada por AWS: AmazonSSMManagedEC2InstanceDefaultPolicy.

Implementación del acceso a los privilegios mínimos

Los procedimientos de este tema están pensados para que solo los realicen los administradores. Por lo tanto, recomendamos implementar el acceso con privilegio mínimo para evitar que los usuarios no administrativos configuren o modifiquen la configuración de administración de host predeterminada. Para ver ejemplos de políticas que restringen el acceso a la configuración de administración de host predeterminada, consulte Ejemplos de políticas de privilegio mínimo para la configuración de administración de host predeterminada más adelante en este tema.

> **Important**
>
> El registro de la información de las instancias registradas con la configuración predeterminada de la administración de hosts almacena la información de registro localmente en los directorios `var/lib/amazon/ssm` o `C:\ProgramData\Amazon`. Si se eliminan estos directorios o sus archivos, la instancia no podrá adquirir las credenciales necesarias para conectarse a Systems Manager mediante la Configuración de la administración de hosts predeterminada. En estos casos, debe utilizar un perfil de instancia IAM para brindar los permisos necesarios a la instancia, o bien volver a crearla.

Temas

- Requisitos previos
- Activación de la configuración de la administración de hosts predeterminada
- Desactivación de la configuración predeterminada de la administración de hosts
- Ejemplos de políticas de privilegio mínimo para la configuración de administración de host predeterminada

Requisitos previos

Para utilizar la configuración predeterminada de la administración de hosts en la Región de AWS y la Cuenta de AWS donde se active la característica, se deben cumplir los siguientes requisitos.

- La instancia que se vaya a administrar debe usar Instance Metadata Service versión 2 (IMDSv2).

  La configuración de administración de host predeterminada no admite la versión 1 del servicio de metadatos de instancias. Para obtener información sobre la transición a IMDSv2, consulte Transición al uso de Servicio de metadatos de instancia versión 2 en la Guía del usuario de Amazon EC2.
- Se debe instalar la versión 3.2.582.0 o alguna versión posterior de SSM Agent en la instancia para ser administrada.

  Para obtener información sobre cómo comprobar la versión de SSM Agent instalada en la instancia, consulte Verificación del número de versión de SSM Agent.

  Para obtener información sobre cómo actualizar SSM Agent, consulte Actualización automática de SSM Agent.

- Usted, como administrador que realiza las tareas de este tema, debe tener permisos para las operaciones de las API GetServiceSetting, ResetServiceSetting y UpdateServicesetting. Además, debe tener permisos `iam:PassRole` para el rol de IAM `AWSSystemsManagerDefaultEC2InstanceManagementRole`. La siguiente política de ejemplo concede estos permisos. Reemplace cada *`example resource placeholder`* con su propia información.

  JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting",
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "arn:aws:ssm:us-east-1:111122223333:servicesetting/ssm/
managed-instance/default-ec2-instance-management-role"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::111122223333:role/service-role/
AWSSystemsManagerDefaultEC2InstanceManagementRole",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
                        "ssm.amazonaws.com"
                    ]
                }
            }
        }
    ]
}
```

- Si ya existe un perfil de instancia de IAM asociado a una instancia de EC2 que hay que administrar con Systems Manager, debe quitarle todos los permisos que permitan la operación

`ssm:UpdateInstanceInformation`. SSM Agent intenta usar permisos de perfil de instancia antes de emplear los permisos de la Configuración de la administración de hosts predeterminada. Si permite la operación `ssm:UpdateInstanceInformation` en los perfiles de instancia de IAM, la instancia no utilizará los permisos de la configuración de administración de host predeterminada.

Activación de la configuración de la administración de hosts predeterminada

Puede activar la configuración predeterminada de la administración de hosts desde la consola de Fleet Manager o al utilizar la AWS Command Line Interface o AWS Tools for Windows PowerShell.

Debe activar, una por una, las configuraciones predeterminadas de la administración de hosts en cada región en donde quiera que las instancias de Amazon EC2 se administren mediante esta configuración.

Tras activar la configuración predeterminada de la administración de hosts, es posible que las instancias tarden 30 minutos en utilizar las credenciales del rol que seleccione en el paso 5 del procedimiento que se detalla a continuación.

Activación de la configuración de administración de host predeterminada (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija Administración de cuenta, Configuración de la administración de host predeterminada.
4. Active Habilitar configuración de administración de host predeterminada.
5. Elija el rol de AWS Identity and Access Management (IAM) que se utiliza para habilitar las herramientas de Systems Manager en sus instancias. Recomendamos utilizar el rol predeterminado que proporciona la configuración de administración de host predeterminada. Contiene el conjunto mínimo de permisos necesarios para administrar sus instancias de Amazon EC2 mediante Systems Manager. Si prefiere utilizar un rol personalizado, la política de confianza del rol debe permitir que Systems Manager sea una entidad de confianza.
6. Elija Configurar para completar la configuración.

Activación de la configuración de administración de host predeterminada (línea de comandos)

1. Cree un archivo JSON en su equipo local que contenga la siguiente política de relaciones de confianza.

JSON

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

2. Abra la AWS CLI o las Herramientas para Windows PowerShell y ejecute uno de los siguientes comandos, según el tipo de sistema operativo de su máquina local, a fin de crear un rol de servicio en su cuenta. Reemplace cada *example resource placeholder* con su propia información.

   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. Ejecute el siguiente comando para adjuntar la política administrada `AmazonSSMManagedEC2InstanceDefaultPolicy` al rol recientemente creado. Reemplace cada *example resource placeholder* con su propia información.

   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. Abra la AWS CLI o Herramientas para Windows PowerShell y ejecute el siguiente comando. Reemplace cada *example resource placeholder* con su propia información.

   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"
```

No se obtienen resultados si el comando se ejecuta correctamente.

5. Ejecute el siguiente comando a fin de ver la configuración del servicio actual para la configuración de administración de host predeterminada en la Cuenta de AWS y la Región de AWS actuales.

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"
```

El comando devuelve información similar a la siguiente.

```
{
    "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"
    }
}
```

Desactivación de la configuración predeterminada de la administración de hosts

Puede desactivar la configuración predeterminada de la administración de hosts desde la consola de Fleet Manager o al utilizar la AWS Command Line Interface o AWS Tools for Windows PowerShell.

Debe desactivar, una por una, las configuraciones predeterminadas de la administración de hosts en cada región en donde ya no quiere que las instancias de Amazon EC2 se administren mediante esta configuración. Si se desactiva en una región, no se desactiva en todas las regiones.

Si desactiva la configuración de administración de host predeterminada y no ha adjuntado un perfil de instancia a sus instancias de Amazon EC2 que permita el acceso a Systems Manager, este último dejará de administrarlas.

Desactivación de la configuración de administración de host predeterminada (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija Administración de cuenta, Configuración de la administración de host predeterminada.
4. Desactive Habilitar configuración de administración de host predeterminada.
5. Elija Configurar para deshabilitar la configuración de administración de host predeterminada.

Desactivación de la configuración de administración de host predeterminada (línea de comandos)

- Abra la AWS CLI o Herramientas para Windows PowerShell y ejecute el siguiente comando. Reemplace cada *example resource placeholder* con su propia información.

  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"
```

Ejemplos de políticas de privilegio mínimo para la configuración de administración de host predeterminada

Los siguientes ejemplos de políticas muestran cómo evitar que los miembros de su organización realicen cambios en la configuración de administración de host predeterminada de su cuenta.

Política de control de servicios para AWS Organizations

La siguiente política demuestra cómo evitar que los miembros no administrativos de AWS Organizations actualicen la configuración de administración de host predeterminada. Reemplace cada *example resource placeholder* con su propia información.

JSON

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

Política para entidades principales de IAM

La siguiente política demuestra cómo evitar que sus grupos, funciones o usuarios de IAM en AWS Organizations actualicen su configuración de administración de host predeterminada. Reemplace cada *example resource placeholder* con su propia información.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "ssm:UpdateServiceSetting",
                "ssm:ResetServiceSetting"
            ],
            "Resource": "arn:aws:ssm:us-east-1:111122223333:servicesetting/ssm/
managed-instance/default-ec2-instance-management-role"
        },
        {
            "Effect": "Deny",
            "Action": [
                "iam:AttachRolePolicy",
                "iam:DeleteRole",
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::111122223333:role/service-role/
AWSSystemsManagerDefaultEC2InstanceManagementRole"
        }
    ]
}
```

Conéctese a una instancia administrada de Windows Server mediante Remote Desktop

Puede utilizar Fleet Manager, una herramienta de AWS Systems Manager, para conectarse a sus instancias de Amazon Elastic Compute Cloud (Amazon EC2) de Windows Server mediante el Remote Desktop Protocol (RDP). Fleet Manager El escritorio remoto, que cuenta con tecnología Amazon DCV, brinda una conectividad segura a las instancias de Windows Server directamente

desde la consola de Systems Manager. Puede tener hasta cuatro conexiones simultáneas en una sola ventana del navegador.

La API de escritorio remoto Fleet Manager se denomina AWS Systems Manager GUI Connect. Para obtener información sobre el uso de la API GUI Connect de Systems Manager, consulte la Referencia de la API de AWS Systems Manager GUI Connect.

Actualmente, solo puede utilizar el escritorio remoto con instancias que ejecuten Windows Server 2012 RTM o versiones posteriores. El escritorio remoto solo admite entradas en idioma inglés.

Fleet Manager Remote Desktop es un servicio exclusivo de consola y no admite conexiones de línea de comandos a las instancias administradas. Para conectarte a una instancia administrada de Windows Server a través de un intérprete de comandos, puede usar Session Manager, otra herramienta de AWS Systems Manager. Para obtener más información, consulte AWS Systems Manager Session Manager.

> **Note**
>
> La duración de una conexión RDP no se determina por la duración de sus credenciales AWS Identity and Access Management (IAM). En su lugar, la conexión se mantiene hasta que se alcance la duración máxima de la conexión o el límite de tiempo de inactividad, lo que ocurra primero. Para obtener más información, consulte Duración y simultaneidad de la conexión remota.

Para obtener más información sobre la configuración de los permisos de (IAM) AWS Identity and Access Management para permitir que las instancias interactúen con Systems Manager, consulte Configurar permisos de instancia para Systems Manager.

Temas

- Configuración del entorno
- Configuración de los permisos de IAM para escritorio remoto
- Autenticación de conexiones del escritorio remoto
- Duración y simultaneidad de la conexión remota
- Conexión a un nodo administrado mediante escritorio remoto
- Ver información sobre las conexiones actuales y completadas

Configuración del entorno

Antes de utilizar el escritorio remoto, asegúrese de que el entorno cumple los siguientes requisitos:

- Configuración de nodos administrados

  Asegúrese de que las instancias de Amazon EC2 estén configuradas como nodos administrados en Systems Manager.

- Versión mínima de SSM Agent

  Verifique que los nodos ejecuten la versión 3.0.222.0 o versiones posteriores de SSM Agent. Para obtener información sobre cómo comprobar la versión de agente que se está ejecutando en un nodo, consulte Verificación del número de versión de SSM Agent. Para obtener información acerca cómo instalar o actualizar SSM Agent, consulte Uso de SSM Agent.

- Configuración del puerto de RDP

  Para aceptar conexiones remotas, el servicio Remote Desktop Services de los nodos de Windows Server debe utilizar el puerto de RDP 3389 predeterminado. Esta es la configuración predeterminada en Amazon Machine Images (AMIs) que proporciona AWS. No es necesario abrir de manera explícita ningún puerto de entrada para utilizar el escritorio remoto.

- Versión del módulo PSReadLine para la funcionalidad del teclado

  Para asegurarse de que el teclado funciona correctamente en PowerShell, verifique que los nodos que ejecutan Windows Server 2022 tengan instalada la versión 2.2.2 o versiones posteriores del módulo PSReadLine. Si están ejecutando una versión anterior, puede instalar la versión necesaria mediante los siguientes comandos.

  ```
  Install-PackageProvider -Name NuGet -MinimumVersion 2.8.5.201 -Force
  ```

  Una vez instalado el proveedor de paquetes NuGet, ejecute el siguiente comando.

  ```
  Install-Module `
   -Name PSReadLine `
   -Repository PSGallery `
   -MinimumVersion 2.2.2 -Force
  ```

- Configuración del administrador de sesiones

Para poder utilizar el escritorio remoto, debe completar los requisitos previos de la configuración del administrador de sesiones. Cuando se conecta a una instancia mediante el escritorio remoto, se aplican todas las preferencias de sesión que haya definido para la Cuenta de AWS y la Región de AWS. Para obtener más información, consulte Configuración de Session Manager.

> **Note**
>
> Si registra la actividad del administrador de sesiones mediante Amazon Simple Storage Service (Amazon S3), las conexiones del escritorio remoto generarán el siguiente error en `bucket_name/Port/stderr`. Se espera que aparezca este error, pero se puede omitir sin problemas.
>
> ```
> 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>
> ```

Configuración de los permisos de IAM para escritorio remoto

Además de los permisos de IAM necesarios para Systems Manager y Session Manager, el usuario o rol que utilice debe tener permisos para iniciar conexiones.

Permisos para iniciar conexiones

Para establecer conexiones RDP con instancias de EC2 en la consola, se necesitan los siguientes permisos:

- `ssm-guiconnect:CancelConnection`
- `ssm-guiconnect:GetConnection`
- `ssm-guiconnect:StartConnection`

Permisos para enumerar las conexiones

Para ver listas de conexiones en la consola, se necesita el siguiente permiso:

`ssm-guiconnect:ListConnections`

A continuación, se muestran ejemplos de políticas de IAM que puede adjuntar a un usuario o un rol para permitir distintos tipos de interacción con el escritorio remoto. Reemplace cada *example resource placeholder* con su propia información.

Política estándar para conectarse a instancias de EC2

JSON

```
{
    "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:*:111122223333:instance/*",
                "arn:aws:ssm:*:111122223333:managed-instance/*",
                "arn:aws:ssm:*::document/AWS-StartPortForwardingSession"
            ],
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "aws:CalledVia": "ssm-guiconnect.amazonaws.com"
                }
            }
        },
        {
            "Sid": "GuiConnect",
            "Effect": "Allow",
            "Action": [
                "ssm-guiconnect:CancelConnection",
                "ssm-guiconnect:GetConnection",
                "ssm-guiconnect:StartConnection",
                "ssm-guiconnect:ListConnections"
            ],
            "Resource": "*"
        }
    ]
}
```

Política de conexión a instancias de EC2 con etiquetas específicas

> **Note**
>
> En la siguiente política de IAM, la sección `SSMStartSession` requiere un nombre de recurso de Amazon (ARN) para la acción `ssm:StartSession`. Como se muestra, el ARN que especifique no requiere un ID de Cuenta de AWS. Si especifica un ID de cuenta, Fleet Manager devuelve un `AccessDeniedException`.

La sección `AccessTaggedInstances`, que se encuentra en la parte inferior de la política de ejemplo, también requiere los ARN para `ssm:StartSession`. Para esos ARN, debe especificar los ID de Cuenta de AWS.

JSON

```
{
    "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": {
                "ForAnyValue:StringEquals": {
                    "aws:CalledVia": "ssm-guiconnect.amazonaws.com"
                }
```

```
            }
        },
        {
            "Sid": "AccessTaggedInstances",
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:*:111122223333:instance/*",
                "arn:aws:ssm:*:111122223333:managed-instance/*"
            ],
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/tag key": [
                        "tag value"
                    ]
                }
            }
        },
        {
            "Sid": "GuiConnect",
            "Effect": "Allow",
            "Action": [
                "ssm-guiconnect:CancelConnection",
                "ssm-guiconnect:GetConnection",
                "ssm-guiconnect:StartConnection",
                "ssm-guiconnect:ListConnections"
            ],
            "Resource": "*"
        }
    ]
}
```

Política para que los usuarios de AWS IAM Identity Center se conecten a instancias de EC2

JSON

```
{
    "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:DescribeInstanceInformation",
                "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": {
                "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"
            ]
        },
        {
            "Sid": "GuiConnect",
            "Effect": "Allow",
            "Action": [
                "ssm-guiconnect:CancelConnection",
                "ssm-guiconnect:GetConnection",
                "ssm-guiconnect:StartConnection",
                "ssm-guiconnect:ListConnections"
            ],
            "Resource": "*"
        }
    ]
}
```

Autenticación de conexiones del escritorio remoto

Cuando se establece una conexión remota, puede autenticarse mediante las credenciales de Windows o el par de claves de Amazon EC2 (archivo `.pem`) que está asociado a la instancia. Para obtener información sobre cómo usar los pares de claves, consulte Pares de claves de Amazon EC2 e instancias de Windows en la Guía del usuario de Amazon EC2.

De forma alternativa, si está autenticado en la AWS Management Console con AWS IAM Identity Center, puede conectarse a las instancias sin proporcionar credenciales adicionales. Si desea ver un ejemplo de política para permitir la autenticación de conexiones remotas mediante IAM Identity Center, consulte Configuración de los permisos de IAM para escritorio remoto.

Antes de empezar

Tenga en cuenta las siguientes condiciones para utilizar la autenticación de IAM Identity Center antes de comenzar a conectarse mediante Remote Desktop.

- El escritorio remoto admite la autenticación de IAM Identity Center para nodos en la misma Región de AWS donde se habilitó IAM Identity Center.
- El escritorio remoto admite nombres de usuario de IAM Identity Center que tengan hasta 16 caracteres.
- El escritorio remoto admite nombres de usuario de IAM Identity Center compuestos por caracteres alfanuméricos y los siguientes caracteres especiales: `.` `-` o `_` .

> **Important**
>
> Las conexiones no se realizarán correctamente para los nombres de usuario de IAM Identity Center que contengan los siguientes caracteres: `+` `=` `,`
> IAM Identity Center admite estos caracteres en los nombres de usuario, pero las conexiones de RDP de Fleet Manager no lo hace.
> Además, si un nombre de usuario del Centro de Identidad de IAM contiene uno o más símbolos `@`, Fleet Manager ignora el primer símbolo `@` y todos los caracteres que le siguen, independientemente de que `@` introduzca o no la parte de dominio de una dirección de correo electrónico. Por ejemplo, en el caso del nombre de usuario del Centro de Identidad de IAM `diego_ramirez@example.com`, la parte `@example.com` se ignora y el nombre de usuario para Fleet Manager pasa a ser `diego_ramirez`. Para `diego_r@mirez@example.com`, Fleet Manager ignora `@mirez@example.com` y el nombre de usuario para Fleet Manager pasa a ser `diego_r`.

- Cuando se autentica una conexión mediante IAM Identity Center, el escritorio remoto crea un usuario de Windows local en el grupo de administradores locales de la instancia. Este usuario persiste una vez finalizada la conexión remota.
- El escritorio remoto no permite la autenticación de IAM Identity Center para nodos que son controladores de dominio de Microsoft Active Directory.
- Si bien el escritorio remoto permite utilizar la autenticación de IAM Identity Center para los nodos unidos a un dominio de Active Directory, no se recomienda hacerlo. Este método de autenticación concede permisos administrativos a usuarios que pueden anular los permisos más restrictivos concedidos por el dominio.

Regiones compatibles con la autenticación de IAM Identity Center

Las siguientes Regiones de AWS son compatibles con conexiones de Remote Desktop que utilicen la autenticación de IAM Identity Center:

- Este de EE. UU. (Ohio) (us-east-2)
- Este de EE. UU. (Norte de Virginia) (us-east-1)
- EE. UU. Oeste (Norte de California) (us-west-1)
- Oeste de EE. UU. (Oregón) (us-west-2)
- África (Ciudad del Cabo) (af-south-1)
- Asia-Pacífico (Hong Kong) (ap-east-1)
- Asia Pacífico (Bombay) (ap-south-1)
- Asia-Pacífico (Tokio) (ap-northeast-1)
- Asia-Pacífico (Seúl) (ap-northeast-2)
- Asia Pacific (Osaka) (ap-northeast-3)
- Asia-Pacífico (Singapur) (ap-southeast-1)
- Asia-Pacífico (Sídney) (ap-southeast-2)
- Asia-Pacífico (Yakarta) (ap-southeast-3)
- Canadá (centro) (ca-central-1)
- Europa (Fráncfort) (eu-central-1)
- Europa (Estocolmo) (eu-north-1)
- Europa (Irlanda) (eu-west-1)
- Europa (Londres) (eu-west-2)

- Europa (París) (eu-west-3)
- Israel (Tel Aviv) (il-central-1)
- América del Sur (São Paulo) (sa-east-1)
- UE (Milán) (eu-south-1)
- Medio Oriente (Baréin) (me-south-1)
- AWS GovCloud (EE. UU. Este) (us-gov-east-1)
- AWS GovCloud (EE. UU. Oeste) (us-gov-west-1)

Duración y simultaneidad de la conexión remota

Las siguientes condiciones se aplican a las conexiones activas del escritorio remoto:

- Duración de la conexión

  Una conexión al escritorio remoto dura 60 minutos de forma predeterminada. Para evitar que una conexión se desconecte, puede elegir Renovar sesión antes de que se desconecte para restablecer el temporizador de duración.
- Tiempo de espera de la conexión

  Una conexión del escritorio remoto se desconecta después de haber estado inactiva durante más de 10 minutos.
- Conexiones persistentes

  Después de conectarse a un Windows Server con Escritorio remoto, la conexión se mantiene hasta que se alcance la duración máxima de la conexión (60 minutos) o el límite de tiempo de espera de inactividad (10 minutos). La duración de la conexión no se determina por la duración de sus credenciales AWS Identity and Access Management (IAM). La conexión se mantiene después de que caduquen las credenciales de IAM si no se alcanzan los límites de duración de la conexión. Cuando utilice Escritorio remoto, debe salir de la página del navegador para finalizar la conexión luego de que caduquen sus credenciales de IAM.
- Conexiones simultáneas

  De forma predeterminada, puede tener un máximo de 5 conexiones del escritorio remoto activas a la vez para la misma Cuenta de AWS y la misma Región de AWS. Para solicitar un aumento de cuota de servicio de hasta 25 conexiones simultáneas, consulte Solicitud de aumento de cuota en la Guía del usuario de Service Quotas.

> **Note**
>
> La licencia estándar de Windows Server permite dos conexiones RDP simultáneas. Para admitir más conexiones, debe adquirir licencias de acceso de cliente (CAL) adicionales de Microsoft o licencias de Microsoft Remote Desktop Services de AWS. Para obtener más información acerca de las licencias complementarias, consulte los siguientes temas:
>
> - Licencias de acceso de cliente y licencias de administración en el sitio web de Microsoft
> - Utilice las suscripciones basadas en usuarios de License Manager para los productos de software compatibles en la Guía del usuario de License Manager

Conexión a un nodo administrado mediante escritorio remoto

Soporte para copiar/pegar texto en el navegador

Con los navegadores Google Chrome y Microsoft Edge, puede copiar y pegar texto desde un nodo administrado a su equipo local y desde su equipo local a un nodo administrado al que esté conectado.

Con el navegador Mozilla Firefox, solo puede copiar y pegar texto desde un nodo administrado a su equipo local. No se admite la copia desde el equipo local al nodo administrado.

Para conectarse a un nodo administrado mediante el escritorio remoto de Fleet Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Haga clic en el nodo al que desea conectarse. Puede seleccionar la casilla de verificación o el nombre del nodo.
4. En el menú Acciones de nodos, elija Conectar con escritorio remoto.
5. Elija el Tipo de autenticación preferido. Si selecciona Credenciales de usuario, ingrese el nombre de usuario y la contraseña de una cuenta de usuario de Windows en el nodo al que se está conectando. Si elige Par de claves, puede proporcionar la autenticación mediante uno de los siguientes métodos:

   a. Elija Examinar equipo local si desea seleccionar la clave PEM asociada a la instancia desde su sistema de archivos local.

-o bien-

b. Elija Pegar el contenido del par de claves si desea copiar el contenido del archivo PEM y pegarlo en el campo proporcionado.

6. Seleccione Conectar.
7. Para elegir la resolución de pantalla de preferencia, en el menú Acciones, elija Resoluciones y, luego, seleccione una de las siguientes opciones:

   - Ajuste automático
   - 1920 x 1080
   - 1400 x 900
   - 1366 x 768
   - 800 x 600

   La opción Adaptar automáticamente establece la resolución en función del tamaño de pantalla detectado.

Ver información sobre las conexiones actuales y completadas

Puede utilizar la sección Fleet Manager de la consola de Systems Manager para ver información sobre las conexiones RDP que se han establecido en la cuenta. Puede utilizar un conjunto de filtros para limitar la lista de conexiones que aparecen a un intervalo de tiempo, una instancia específica, el usuario que estableció las conexiones y las conexiones de un estado específico. Además, la consola ofrece pestañas que muestran información sobre todas las conexiones actualmente activas y todas las conexiones anteriores.

Para ver información sobre las conexiones actuales y completadas

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Seleccione Administración de cuentas, Establecer conexión con el escritorio remoto.
4. Elija una de las siguientes pestañas:

   - Conexiones activas
   - Historial de conexiones

5. Para reducir aún más la lista de resultados de conexiones que aparecen, especifique uno o varios filtros en la casilla de búsqueda (). También puede ingresar un término de búsqueda de texto libre.

Administración de volúmenes de Amazon EBS en instancias administradas

Amazon Elastic Block Store (Amazon EBS) proporciona volúmenes de almacenamiento por bloques para su uso con instancias de Amazon Elastic Compute Cloud (EC2). Los volúmenes de EBS se comportan como dispositivos de bloques sin formatear. Puede montar estos volúmenes como dispositivos en sus instancias.

Puede utilizar Fleet Manager, una herramienta de AWS Systems Manager, para administrar los volúmenes de Amazon EBS en las instancias administradas. Por ejemplo, puede inicializar un volumen de EBS, dar formato a una partición y montar el volumen para que esté disponible para su uso.

> **Note**
>
> En la actualidad, Fleet Manager solo admite la administración de volúmenes de Amazon EBS para instancias de Windows Server.

Visualización de los detalles del volumen de EBS

Para ver los detalles del volumen de EBS con Fleet Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija el botón situado junto a la instancia administrada de la que desea ver los detalles del volumen de EBS.
4. Elija Ver detalles.
5. Elija Herramientas, Volúmenes de EBS.
6. Para ver los detalles de un volumen de EBS, elija su ID en la columna ID de volumen.

Inicialización y formato de un volumen de EBS

Para inicializar un volumen de EBS y darle formato con Fleet Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija el botón situado junto a la instancia administrada de la que desea inicializar, dar formato y montar el volumen de EBS. Solo puede inicializar un volumen de EBS si su disco está vacío.
4. Elija Ver detalles.
5. En el menú Herramientas, elija Volúmenes de EBS.
6. Elija el botón situado junto al volumen de EBS que desea inicializar y dar formato.
7. Seleccione Inicializar y dar formato.
8. En Estilo de partición, elija el que desee usar para el volumen de EBS.
9. (Opcional) Elija una letra de unidad para la partición.
10. (Opcional) Escriba un nombre de partición para identificarla.
11. Elija el sistema de archivos que se utilizará para organizar los archivos y los datos almacenados en la partición.
12. Elija Confirmar para poner el volumen de EBS a disposición para su uso. No puede cambiar la configuración de la partición de la AWS Management Console después de confirmarla. Sin embargo, puede usar SSH o RDP para iniciar sesión en la instancia y cambiar la configuración de la partición.

Acceso al portal de la base de conocimientos de Red Hat

Puede usar Fleet Manager, una herramienta de AWS Systems Manager, para acceder al portal de la base de conocimientos si es cliente de Red Hat. Se lo considera cliente de Red Hat si ejecuta instancias de Red Hat Enterprise Linux (RHEL) o utiliza servicios de RHEL en AWS. El portal de la base de conocimientos incluye datos binarios y recursos compartidos de conocimientos y foros de discusión para soporte comunitario que solo están disponibles para clientes con licencia de Red Hat.

Además de los permisos de AWS Identity and Access Management (IAM) requeridos para Systems Manager y Fleet Manager, el usuario o el rol que usa para acceder a la consola debe permitir la acción `rhelkb:GetRhelURL` para acceder al portal de la base de conocimientos.

Para acceder al portal de la base de conocimientos de Red Hat

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija la instancia de RHEL que desea utilizar para conectarse al portal de la base de conocimientos de Red Hat.
4. Seleccione Administración de cuenta, Acceder a la base de conocimientos de Red Hat para abrir la página de la base de conocimientos de Red Hat.

Si utiliza RHEL en AWS para ejecutar cargas de trabajo de RHEL totalmente compatibles, también puede acceder a la base de conocimientos de Red Hat a través del sitio web de Red Hat con sus credenciales de AWS.

Solución de problemas de disponibilidad de nodos administrados

Para varias herramientas de AWS Systems Manager, como Run Command, Distributor y Session Manager, puede elegir seleccionar de forma manual los nodos administrados en los que desea ejecutar la operación. En estos casos, después de especificar que desea elegir los nodos de forma manual, se muestra una lista de los nodos administrados donde puede ejecutar la operación.

Este tema proporciona información para ayudarlo a diagnosticar por qué un nodo administrado que ha confirmado que se está ejecutando no está incluido en las listas de nodos administrados de Systems Manager.

Para que Systems Manager administre un nodo y dicho nodo esté disponible en las listas de nodos administrados, debe cumplir tres requisitos:

- SSM Agent debe estar instalado y ejecutándose en el nodo con un sistema operativo compatible.

> **Note**
>
> Algunas Amazon Machine Images (AMIs) administradas de AWS están configuradas para lanzar instancias con SSM Agent preinstalado. (También puede configurar una AMI personalizada para la preinstalación de SSM Agent.) Para obtener más información, consulte Búsqueda de AMIs con SSM Agent preinstalado.

- Para las instancias de Amazon Elastic Compute Cloud (Amazon EC2), debe adjuntar un perfil de instancias de AWS Identity and Access Management (IAM) a la instancia. El perfil de instancias permite que la instancia se comunique con el servicio de Systems Manager. Si no asigna un perfil de instancias a la instancia, lo registra mediante una activación híbrida, lo que no es un escenario común.
- SSM Agent debe poder conectarse a un punto de enlace de Systems Manager para registrarse en el servicio. A partir de entonces, el nodo administrado debe estar disponible para el servicio, lo que se confirma mediante el envío de una señal cada cinco minutos para verificar el estado de la instancia.
- Cuando el estado de un nodo administrado sea `Connection Lost` durante al menos 30 días, es posible que el nodo deje de aparecer en la lista de la consola de Fleet Manager. Para que vuelva a aparecer en la lista, se debe resolver el problema que haya provocado la pérdida de conexión.

Después de verificar que se está ejecutando un nodo administrado, puede utilizar el siguiente comando para verificar si SSM Agent se ha registrado correctamente con el servicio de Systems Manager. Este comando no devuelve resultados hasta que se haya realizado un registro correcto.

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 el registro se realizó correctamente y el nodo administrado ahora está disponible para las operaciones de Systems Manager, el comando devuelve resultados similares a los siguientes.

```
{
    "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 el registro aún no se ha completado o no se ejecuta de manera correcta, el comando devuelve resultados similares a los siguientes:

```
{
    "InstanceAssociationStatusInfos": []
}
```

Si el comando no devuelve resultados después de 5 minutos, utilice la siguiente información para que lo ayude a solucionar problemas con los nodos administrados.

Temas

- Solución 1: comprobar que el SSM Agent está instalado y ejecutándose en el nodo administrado
- Solución 2: comprobar que se ha especificado un perfil de instancias de IAM para la instancia (solo instancias de EC2)
- Solución 3: verificar la conectividad de los puntos de enlace de servicio
- Solución 4: verificar la compatibilidad con el sistema operativo de destino
- Solución 5: verificar que está trabajando en la misma Región de AWS que la instancia de Amazon EC2

- Solución 6: comprobar la configuración del proxy que aplicó a SSM Agent en el nodo administrado
- Solución 7: instalar un certificado TLS en instancias administradas
- Solución de problemas de disponibilidad de nodos administrados mediante ssm-cli

Solución 1: comprobar que el SSM Agent está instalado y ejecutándose en el nodo administrado

Asegúrese de que la versión más reciente de SSM Agent está instalada y ejecutándose en el nodo administrado.

Para determinar si SSM Agent está instalado y ejecutándose en un nodo administrado, consulte Verificación del estado de SSM Agent e inicio del agente.

Para instalar o reinstalar SSM Agent en un nodo administrado, consulte los siguientes temas:

- Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux
- Cómo instalar SSM Agent en nodos de Linux híbridos
- Cómo instalar y desinstalar de forma manual SSM Agent en instancias de EC2 para Windows Server
- Cómo instalar SSM Agent en nodos de Windows híbridos

Solución 2: comprobar que se ha especificado un perfil de instancias de IAM para la instancia (solo instancias de EC2)

Para instancias de Amazon Elastic Compute Cloud (Amazon EC2), compruebe que la instancia está configurada con un perfil de instancias de AWS Identity and Access Management (IAM) que permite que la instancia se comunique con la API de Systems Manager. Verifique también que su usuario tenga una política de confianza de IAM que le permita comunicarse con la API de Systems Manager.

> **Note**
>
> Los servidores locales, los dispositivos de borde y las máquinas virtuales utilizan un rol de servicio de IAM en lugar de un perfil de instancias. Para obtener más información, consulte Creación del rol de servicio de IAM requerido para Systems Manager en entornos híbridos y multinube.

Para determinar si un perfil de instancias con los permisos necesarios está adjuntado a una instancia de EC2

1. Abra la consola de Amazon EC2 en https://console.aws.amazon.com/ec2/.
2. En el panel de navegación, seleccione Instances (Instancias).
3. Elija la instancia que desea verificar para un perfil de instancias.
4. En la pestaña Descripción en el panel inferior, busque Rol de IAM y elija el nombre del rol.
5. En la página Resumen del rol para el perfil de instancia, en la pestaña Permisos, asegúrese de que `AmazonSSMManagedInstanceCore` se encuentre en la lista de Políticas de permisos.

   Si, en su lugar, se utiliza una política personalizada, asegúrese de que proporcione los mismos permisos que `AmazonSSMManagedInstanceCore`.

   Abrir `AmazonSSMManagedInstanceCore` en la consola

   Para obtener información sobre otras políticas que se pueden asociar a un perfil de instancia para Systems Manager, consulte Configuración de permisos de instancia requeridos para Systems Manager.

Solución 3: verificar la conectividad de los puntos de enlace de servicio

Compruebe que la instancia tenga conectividad con los puntos de enlace de servicio de Systems Manager. Esta conectividad se proporciona creando y configurando puntos de enlace de la VPC para Systems Manager, o permitiendo el tráfico saliente HTTPS (puerto 443) a los puntos de enlace de servicio.

Para las instancias de Amazon EC2, el punto de conexión de servicio de Systems Manager para la Región de AWS se utiliza para registrar la instancia si la configuración de la nube privada virtual (VPC) permite el tráfico saliente. Sin embargo, si la configuración de la VPC en la que se lanzó la instancia no permite el tráfico saliente y no puede cambiar esta configuración para permitir la conectividad con los puntos de enlace de servicio públicos, configure los puntos de enlace de interfaz para la VPC.

Para obtener más información, consulte Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager.

Solución 4: verificar la compatibilidad con el sistema operativo de destino

Compruebe que la operación que eligió se pueda ejecutar en el tipo de nodo administrado que espera ver en la lista. Algunas operaciones de Systems Manager solo pueden dirigirse a instancias de Windows o a instancias de Linux. Por ejemplo, los documentos de Systems Manager (SSM) `AWS-InstallPowerShellModule` y `AWS-ConfigureCloudWatch` solo se puede ejecutar en las instancias de Windows. En la página Ejecutar un comando, si elige cualquiera de estos documentos y selecciona Elegir instancias manualmente, solo se muestran las instancias de Windows que están disponibles para su selección.

Solución 5: verificar que está trabajando en la misma Región de AWS que la instancia de Amazon EC2

Las instancias de Amazon EC2 se crean y están disponibles en Regiones de AWS específicas, como la Región Este de EE. UU. (Ohio) (us-east-2) o la Región Europa (Irlanda) (eu-west-1). Asegúrese de que trabaja en la misma Región de AWS que la instancia de Amazon EC2 con la que desea trabajar. Para obtener más información, consulte Elegir una región en la Introducción a la AWS Management Console.

Solución 6: comprobar la configuración del proxy que aplicó a SSM Agent en el nodo administrado

Verifique que la configuración del proxy que aplicó a SSM Agent en el nodo administrado sea correcta. Si la configuración del proxy es incorrecta, el nodo no puede conectarse a los puntos de conexión de servicio requeridos o es posible que Systems Manager identifique incorrectamente el sistema operativo del nodo administrado. Para obtener más información, consulte Configuración de SSM Agent para utilizar un proxy en nodos de Linux y Configurar el SSM Agent para usar un proxy para las instancias de Windows Server.

Solución 7: instalar un certificado TLS en instancias administradas

Se debe instalar un certificado de Seguridad de la capa de transporte (TLS) en cada instancia administrada que utiliza con AWS Systems Manager. Los Servicios de AWS utilizan estos certificados para cifrar llamadas a otros Servicios de AWS.

Un certificado TLS ya está instalado de forma predeterminada en cada instancia de Amazon EC2 creada a partir de una Amazon Machine Image (AMI). La mayoría de los sistemas operativos modernos incluyen el certificado TLS necesario de las entidades de certificación (CA) de Amazon Trust Services en su almacén de confianza.

Para comprobar si el certificado requerido está instalado en la instancia, ejecute el comando siguiente basado en el sistema operativo de la instancia. Asegúrese de sustituir la parte de la *región* de la URL con la Región de AWS en la que se encuentra la instancia administrada.

Linux & macOS

```
curl -L https://ssm.region.amazonaws.com
```

Windows

```
Invoke-WebRequest -Uri https://ssm.region.amazonaws.com
```

El comando debe devolver un error `UnknownOperationException`. Si en su lugar recibe un mensaje de error SSL/TLS, es posible que el certificado requerido no esté instalado.

Si descubre que los certificados de CA de Amazon Trust Services necesarios no están instalados en sus sistemas operativos principales, en las instancias creadas a partir de AMIs que no han sido suministradas por Amazon o en sus propios servidores locales y máquinas virtuales, debe instalar y permitir un certificado de Amazon Trust Services o usar AWS Certificate Manager (ACM) para crear y administrar certificados para un servicio integrado compatible.

Todas las instancias administradas deben tener instalado uno de los siguientes certificados Transport Layer Security (TLS).

- Amazon Root CA 1
- Starfield Services Root Certificate Authority - G2
- Starfield Class 2 Certificate Authority

Para obtener información sobre ACM, consulte la Guía del usuario de AWS Certificate Manager.

Si los certificados de su entorno informático se administran por medio de un objeto de política de grupo (GPO), es posible que tenga que configurar la política de grupo para que incluya uno de estos certificados.

Para obtener más información acerca de los certificados Root y Starfield de Amazon, consulte la publicación del blog How to Prepare for AWS's Move to Its Own Certificate Authority.

Solución de problemas de disponibilidad de nodos administrados mediante **ssm-cli**

La `ssm-cli` es una herramienta de la línea de comandos independiente incluida en la instalación de SSM Agent. Al instalar SSM Agent 3.1.501.0 o una versión posterior en un equipo, puede ejecutar comandos `ssm-cli` en ese equipo. El resultado de estos comandos lo ayuda a determinar si el equipo cumple con los requisitos mínimos para que AWS Systems Manager administre una instancia de Amazon EC2 o un equipo que no sea de EC2 y que, por lo tanto, se agregue a listas de nodos administrados de Systems Manager. (La versión 3.1.501.0 de SSM Agent se publicó en noviembre de 2021).

Requisitos mínimos

Para que AWS Systems Manager administre una instancia de Amazon EC2 o un equipo que no sea de EC2, y que estos estén disponibles en listas de nodos administrados, se deben cumplir tres requisitos principales:

- SSM Agent debe estar instalado y ejecutándose en un equipo con un sistema operativo compatible.

  Algunas Amazon Machine Images (AMIs) administradas de AWS para EC2 están configuradas para lanzar instancias con SSM Agent preinstalado. (También puede configurar una AMI personalizada para la preinstalación de SSM Agent.) Para obtener más información, consulte Búsqueda de AMIs con SSM Agent preinstalado.
- Debe asociarse al equipo un perfil de instancia de AWS Identity and Access Management (IAM) (para instancias de EC2) o un rol de servicio de IAM (para equipos que no sean de EC2) que proporcione los permisos necesarios para comunicarse con el servicio de Systems Manager.
- SSM Agent debe poder conectarse a un punto de conexión de Systems Manager para registrarse en el servicio. A partir de entonces, el nodo administrado debe estar disponible para el servicio, lo que se confirma mediante el envío de una señal cada cinco minutos para verificar el estado del nodo administrado.

Comandos preconfigurados en **ssm-cli**

Se incluyen comandos preconfigurados que recopilan la información necesaria para ayudarlo a diagnosticar por qué un equipo que ha confirmado que se está ejecutando no está incluido en las listas de nodos administrados de Systems Manager. Estos comandos se ejecutan cuando se especifica la opción `get-diagnostics`.

En el equipo, ejecute el siguiente comando para utilizar `ssm-cli` como ayuda para solucionar problemas de disponibilidad del nodo administrado.

Linux & macOS

```
ssm-cli get-diagnostics --output table
```

Windows

En equipos de Windows Server, debe dirigirse al directorio `C:\Program Files\Amazon\SSM` antes de ejecutar el comando.

```
ssm-cli.exe get-diagnostics --output table
```

PowerShell

En equipos de Windows Server, debe dirigirse al directorio `C:\Program Files\Amazon\SSM` antes de ejecutar el comando.

```
.\ssm-cli.exe get-diagnostics --output table
```

El comando devuelve el resultado en formato de tabla similar al siguiente.

> **Note**
>
> Las comprobaciones de conectividad a los puntos de conexión `ssmmessages`, `s3`, `kms`, `logs` y `monitoring` son para características opcionales adicionales, tales como Session Manager, que pueden registrarse en Amazon Simple Storage Service (Amazon S3) o los Registros de Amazon CloudWatch, y utilizar el cifrado de AWS Key Management Service (AWS KMS).

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
                                      #
#############################################################################################
```

En la siguiente tabla se proporcionan detalles adicionales para cada una de las verificaciones realizadas por `ssm-cli`.

Verificaciones de diagnóstico de **`ssm-cli`**

Check	Detalles
Servicio de metadatos de la instancia de Amazon EC2	Indica si el nodo administrado puede acceder al servicio de metadatos. Una prueba fallida indica un problema de conectividad con `http://169.254.169.254` , que puede deberse a configuraciones de firewall y proxy de la ruta local, el proxy o el sistema operativo (SO).
Registro de instancias híbridas	Indica si SSM Agent está registrado mediante una activación híbrida.
Conectividad al punto de conexión `ssm`	Indica si el nodo puede acceder a los puntos de conexión de servicio de Systems Manager en el puerto TCP 443. Una prueba fallida indica

Check	Detalles
	problemas de conectividad con `https://ssm.` *`region`*`.amazonaws.com` en función de la Región de AWS donde se encuentra el nodo. Los problemas de conectividad se pueden deber a la configuración de la VPC, incluidos grupos de seguridad, listas de control de acceso a la red, tablas de enrutamiento o firewalls y proxies del SO.
Conectividad al punto de conexión `ec2messages`	Indica si el nodo puede acceder a los puntos de conexión de servicio de Systems Manager en el puerto TCP 443. Una prueba fallida indica problemas de conectividad con `https://ec2messages.` *`region`*`.amazonaws.com` en función de la Región de AWS donde se encuentra el nodo. Los problemas de conectividad se pueden deber a la configuración de la VPC, incluidos grupos de seguridad, listas de control de acceso a la red, tablas de enrutamiento o firewalls y proxies del SO.
Conectividad al punto de conexión `ssmmessages`	Indica si el nodo puede acceder a los puntos de conexión de servicio de Systems Manager en el puerto TCP 443. Una prueba fallida indica problemas de conectividad con `https://ssmmessages.` *`region`*`.amazonaws.com` en función de la Región de AWS donde se encuentra el nodo. Los problemas de conectividad se pueden deber a la configuración de la VPC, incluidos grupos de seguridad, listas de control de acceso a la red, tablas de enrutamiento o firewalls y proxies del SO.

Check	Detalles
Conectividad al punto de conexión `s3`	Indica si el nodo puede acceder al punto de conexión de servicio de Amazon Simple Storage Service en el puerto TCP 443. Una prueba fallida indica problemas de conectividad con `https://s3.` *`region`*`.amazonaws.com` en función de la Región de AWS donde se encuentra el nodo. No es necesaria la conectividad a este punto de conexión para que un nodo aparezca en la lista de nodos administrados.
Conectividad al punto de conexión `kms`	Indica si el nodo puede acceder al punto de conexión de servicio de AWS Key Management Service en el puerto TCP 443. Una prueba fallida indica problemas de conectividad con `https://kms.` *`region`*`.amazonaws.com` en función de la Región de AWS donde se encuentra el nodo. No es necesaria la conectividad a este punto de conexión para que un nodo aparezca en la lista de nodos administrados.
Conectividad al punto de conexión `logs`	Indica si el nodo puede acceder al punto de conexión de servicio de Registros de Amazon CloudWatch en el puerto TCP 443. Una prueba fallida indica problemas de conectividad con `https://logs.` *`region`*`.amazonaws.com` en función de la Región de AWS donde se encuentra el nodo. No es necesaria la conectividad a este punto de conexión para que un nodo aparezca en la lista de nodos administrados.

Check	Detalles
Conectividad al punto de conexión `monitoring`	Indica si el nodo puede acceder al punto de conexión de servicio de Amazon CloudWatch en el puerto TCP 443. Una prueba fallida indica problemas de conectividad con `https://monitoring.` *`region`*`.amazonaws.com` en función de la Región de AWS donde se encuentra el nodo. No es necesaria la conectividad a este punto de conexión para que un nodo aparezca en la lista de nodos administrados.
Credenciales de AWS	Indica si SSM Agent tiene las credenciales necesarias en función del perfil de instancia de IAM (para instancias de EC2) o el rol de servicio de IAM (para equipos que no sean de EC2) asociadas al equipo. Una prueba fallida indica que no hay un perfil de instancia de IAM o un rol de servicio de IAM asociado al equipo o que no contiene los permisos necesarios para Systems Manager.
Servicio de agente	Indica si el servicio de SSM Agent se está ejecutando y si se ejecuta como raíz para Linux o macOS, o como SYSTEM para Windows Server. Una prueba fallida indica que el servicio de SSM Agent no se está ejecutando o no se ejecuta como raíz o SYSTEM.
Configuración del proxy	Indica si SSM Agent está configurado para utilizar un proxy.
Estado de imagen de Sysprep (solo Windows)	Indica el estado de `Sysprep` en el nodo. SSM Agent no iniciará en el nodo si el estado de `Sysprep` es un valor distinto a `IMAGE_STATE_COMPLETE` .

Check	Detalles
Versión de SSM Agent	Indica si está instalada la versión más reciente disponible de SSM Agent.

Activaciones híbridas de AWS Systems Manager

Para configurar máquinas que no son de EC2 para su uso con AWS Systems Manager en un entorno híbrido y multinube, debe crear una activación híbrida. Los tipos de máquinas que no son de EC2 y que se admiten como nodos administrados son los siguientes:

- Servidores en sus propias instalaciones (servidores en las instalaciones)
- Dispositivos de núcleo de AWS IoT Greengrass
- Dispositivos de AWS IoT y periféricos que no sean de AWS
- Máquinas virtuales (VM), incluidas las VM de otros entornos de nube

Cuando ejecuta el comando create-activation para iniciar un proceso de activación híbrida, recibe un código de activación y un ID en la respuesta del comando. A continuación, debe incluir el código y el ID de activación junto con el comando para instalar SSM Agent en la máquina, tal y como se describe en el paso 3 de Instalar SSM Agent en nodos de Linux híbridos y el paso 4 de Instalación de SSM Agent en nodos híbridos de Windows Server.

Este proceso de activación se aplica a todos los tipos de máquinas que no son de EC2, excepto a los dispositivos de AWS IoT Greengrass principales. Para obtener información acerca de cómo configurar dispositivos de núcleo de AWS IoT Greengrass para Systems Manager, consulte Administración de dispositivos periféricos con Systems Manager.

> **Note**
>
> Por el momento, no se proporciona soporte para máquinas con macOS que no sean de EC2.

Acerca de los niveles de instancias de Systems Manager

AWS Systems Manager ofrece un nivel de instancias estándares y un nivel de instancias avanzadas. Ambos admiten nodos administrados en su entorno híbrido y multinube. El nivel de instancias estándar le permite registrar un máximo de 1000 máquinas por Cuenta de AWS por Región de

AWS. Si tiene que registrar más de 1000 máquinas en una única cuenta y región, utilice el nivel de instancias avanzadas. Puede crear tantos nodos administrados como desee en el nivel de instancias avanzadas. Todos los nodos administrados configurados para Systems Manager tienen un precio de pago por uso. Para obtener más información acerca de la habilitación del nivel de instancias avanzadas, consulte Activación del nivel de instancias avanzadas. Para obtener más información sobre los precios, consulte Precios de AWS Systems Manager.

Tenga en cuenta la siguiente información adicional acerca del nivel de instancias estándar y el nivel de instancias avanzadas:

- Las instancias avanzadas también permiten conectar a los nodos que no son de EC2 a un entorno híbrido y multinube mediante AWS Systems Manager Session Manager. Session Manager proporciona acceso a las instancias mediante el intérprete de comandos interactivo. Para obtener más información, consulte AWS Systems Manager Session Manager.
- La cuota de instancias estándar también se aplica a las instancias EC2 que utilizan una activación local de Systems Manager (que no es un escenario común).
- Para aplicar revisiones a las aplicaciones publicadas por Microsoft en instancias locales de máquinas virtuales, active el nivel de instancias avanzadas. El uso del nivel de instancias avanzadas conlleva un cargo. No hay ningún cargo adicional por usar revisiones en las aplicaciones publicadas por Microsoft en instancias de Amazon Elastic Compute Cloud (Amazon EC2). Para obtener más información, consulte Revisiones de aplicaciones publicadas por Microsoft en Windows Server.

Inventario de AWS Systems Manager

AWS Systems Manager Inventory ofrece visibilidad en su entorno informático de AWS. Puede utilizar Inventory para recopilar metadatos de los nodos administrados. Puede almacenar estos metadatos en un bucket de Amazon Simple Storage Service (Amazon S3) central y, a continuación, utilizar las herramientas integradas para consultar los datos y determinar rápidamente qué nodos ejecutan el software y las configuraciones requeridas por la política de software, así como los nodos que deben actualizarse. Puede configurar Inventory en todos los nodos administrados mediante un procedimiento de un solo clic. También puede configurar y ver los datos de inventario de varias Regiones de AWS y Cuentas de AWS cuando utiliza Amazon Athena. Para comenzar a utilizar Inventory, abra la consola de Systems Manager. En el panel de navegación, elija Inventory.

Si los tipos de metadatos preconfigurados recopilados por Systems Manager Inventory no se adaptan a sus necesidades, puede crear un inventario personalizado. Un inventario personalizado

es simplemente un archivo JSON con información proporcionada y agregada por usted al nodo administrado en un directorio específico. Cuando Systems Manager Inventory recopila datos, captura estos datos de inventario personalizados. Por ejemplo, si ejecuta un gran centro de datos, puede especificar la ubicación de bastidor de cada servidor como un inventario personalizado. Una vez hecho esto, podrá ver los datos del espacio del bastidor al visualizar otros datos de inventario.

> **Important**
>
> Systems Manager Inventory solo recopila metadatos de los nodos administrados. Inventory no tiene acceso a información o datos privados.

En la siguiente tabla, se muestran los tipos de datos que puede recopilar con Systems Manager Inventory. La tabla también describe las diferentes ofertas para dirigirse a nodos y los intervalos de recopilación que puede especificar.

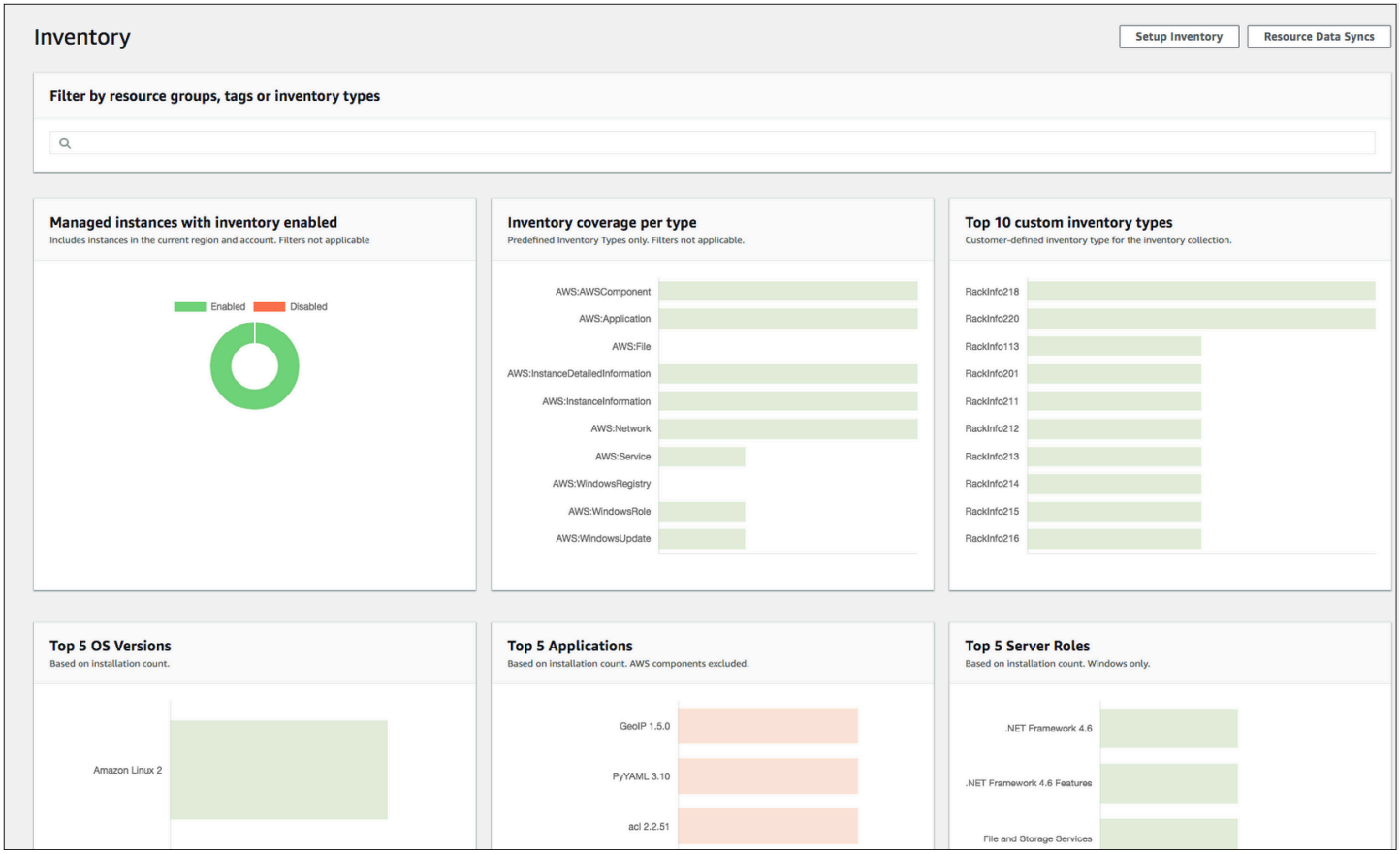
Configuración	Detalles
Tipos de metadatos	Puede configurar Inventory para que recopile los siguientes tipos de datos: • Aplicaciones: nombres de aplicaciones, editores, versiones, etc. • Componentes de AWS: controlador de EC2, agentes, versiones, etc. • Archivos: nombre, tamaño, versión, fecha de instalación, horas de modificación y último acceso, etc. • Configuración de red: dirección IP, dirección MAC, DNS, gateway, máscara de subred, etc. • Actualizaciones de Windows: ID de hotfix, instalado por, fecha de instalación, etc. • Detalles de la instancia: CPUModel, CPUCores, CPUs, CPUSpeedMHz, CPUSockets, CPUHyperThreadEnabled, OSServicePacketc.

Configuración	Detalles
	• Servicios: nombre, nombre de visualización, estado, servicios relacionados, tipo de servicio, tipo de inicio, etc. • Etiquetas: etiquetas asignadas a los nodos. • Registro de Windows: ruta de la clave del registro, nombre de valor, tipo de valor y valor. • Roles de Windows: nombre, nombre de visualización, ruta, tipo de característica, estado de instalación, etc. • Inventario personalizado: metadatos asignados a un nodo administrado tal y como se describe en Uso del inventario personalizado. **Note** Para ver una lista de todos los metadatos que Inventory ha recopilado, consulte Metadatos recopilados por Inventory.
Nodos a los que dirigirse	Puede elegir inventariar todos los nodos administrados de la Cuenta de AWS, seleccionar nodos individualmente o dirigirse a grupos de nodos mediante etiquetas. Para obtener más información sobre cómo recopilar datos de inventario de todos los nodos administrados, consulte Inventario de todos los nodos administrados de la Cuenta de AWS.

Configuración	Detalles
Cuándo recopilar la información	Puede especificar un intervalo de recopilación en términos de minutos, horas y días. El intervalo de recopilación más corto es cada 30 minutos.

> **Note**
>
> Dependiendo de la cantidad de datos recopilados, el sistema puede tardar varios minutos para informar de los datos a la salida especificada. Después de haber recopilado la información, los datos se envían a través de un canal HTTPS seguro a un almacén de AWS de texto sin formato que solo es accesible desde su Cuenta de AWS.

Puede ver los datos en la consola de Systems Manager, en la página Inventory, que incluye diferentes tarjetas predefinidas para ayudarlo a consultar los datos.



> **Note**
>
> Las tarjetas de Inventory filtran automáticamente las instancias administradas de Amazon EC2 con los estados Terminated (Terminado) y Stopped (Detenido). Para los nodos administrados locales y de dispositivos de núcleo de AWS IoT Greengrass, las tarjetas de Inventory filtran automáticamente los nodos con el estado Terminated (Terminado).

Si crea una sincronización de datos de recursos para sincronizar y almacenar todos los datos en un solo bucket de Amazon S3, puede desglosar los datos en la página Inventory Detailed View (Vista detallada de inventario). Para obtener más información, consulte Consulta de datos de Inventory de varias regiones y cuentas.

Compatibilidad con EventBridge

Esta herramienta de Systems Manager se admite como un tipo de evento en las reglas de Amazon EventBridge. Para obtener más información, consulte Monitoreo de eventos de Systems Manager con Amazon EventBridge y Referencia: patrones y tipos de eventos de Amazon EventBridge para Systems Manager.

Contenido

- Más información acerca de Systems Manager Inventory
- Configuración de Systems Manager Inventory
- Configuración de la recopilación de inventario
- Consulta de datos de Inventory de varias regiones y cuentas
- Consulta de una recopilación de inventario mediante filtros
- Agregación de datos de Inventory
- Uso del inventario personalizado
- Visualización del seguimiento de cambios y del historial de Inventory
- Detención de la recopilación de datos y eliminación de datos de inventario
- Asignación de metadatos de inventarios personalizados a un nodo administrado
- Uso de la AWS CLI para configurar la recopilación de datos de inventario
- Explicación: cómo utilizar la sincronización de datos de recursos para agregar datos de inventario
- Solución de problemas con Systems Manager Inventory

Más información acerca de Systems Manager Inventory

Al configurar AWS Systems Manager Inventory, debe especificar el tipo de metadatos que se van a recopilar, los nodos administrados de los que deben recopilarse los metadatos y una programación de recopilación de metadatos. Estas configuraciones se guardan con la Cuenta de AWS como una asociación de AWS Systems Manager State Manager. Una asociación no es más que una configuración.

> **Note**
>
> Inventory solo recopila metadatos. No recopila los datos personales o privados.

Temas

- Metadatos recopilados por Inventory
- Uso del inventario de archivos y del registro de Windows

Metadatos recopilados por Inventory

En el siguiente ejemplo se muestra la lista completa de los metadatos que cada complemento de AWS Systems Manager Inventory recopila.

```
{
    "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: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

- Para `"typeName": "AWS:InstanceInformation"`, `InstanceStatus` puede ser uno de los siguientes: Activo, Conexión perdida, Detenido, Terminado.
- Con el lanzamiento de la versión 2.5, RPM Package Manager sustituye el atributo Serial por Epoch. El atributo Epoch es un entero de incremento monotónico como Serial. Cuando se realiza el inventario mediante el tipo `AWS:Application`, un valor mayor de Epoch

significa una versión más reciente. Si los valores de Epoch son los mismos o están en blanco, utilice el valor del atributo Version o Release para determinar la versión más reciente.

- Algunos metadatos no están disponibles en las instancias de Linux. En concreto, en el caso de “typeName”: “AWS:Network”, los siguientes tipos de metadatos aún no son compatibles con las instancias de Linux. SON compatibles con Windows.
  - { "name": "SubnetMask", "dataType": "STRING"},
  - { "name": "DHCPServer", "dataType": "STRING"},
  - { "name": "DHCPServer", "dataType": "STRING"},
  - { "name": "Gateway", "dataType": "STRING"},

Uso del inventario de archivos y del registro de Windows

AWS Systems Manager Inventory le permite buscar archivos en los sistemas operativos Windows, Linux y macOS, y realizar un inventario de estos. Asimismo puede realizar búsquedas en el registro de Windows e inventariarlo.

Archivos: puede recopilar información de los metadatos de los archivos, como los nombres de los archivos, la hora en que se crearon, la hora de la última modificación y del último acceso y los tamaños de los archivos, por solo citar algunos ejemplos. Para comenzar a recopilar el inventario de archivos, debe especificar una ruta de archivo donde desee realizar el inventario, uno o varios patrones que definan los tipos de archivos que desee inventariar y si la ruta debe atravesarse recursivamente. Systems Manager realiza inventarios de todos los metadatos de los archivos de la ruta especificada que coinciden con el patrón. El inventario de archivos utiliza la entrada de parámetro siguiente.

```
{
"Path": string,
"Pattern": array[string],
"Recursive": true,
"DirScanLimit" : number // Optional
}
```

- Ruta: la ruta del directorio donde desee inventariar los archivos. En el caso de Windows puede utilizar variables de entorno como %PROGRAMFILES% siempre y cuando la variable se asocie a

una única ruta de directorio. Por ejemplo, si utiliza la variable %PATH% que está asociada a varias rutas de directorio, Inventory genera un error.

- Patrón: matriz de patrones para identificar archivos.
- Recursivo: valor booleano que indica si Inventory debe atravesar recursivamente los directorios.
- DirScanLimit: valor opcional que especifica cuántos directorios deben examinarse. Use este parámetro para minimizar el impacto en el rendimiento de los nodos administrados. Inventory examina un máximo de 5.000 directorios.

> **Note**
>
> Inventory recopila metadatos para un máximo de 500 archivos en todas las rutas especificadas.

A continuación, mostramos algunos ejemplos de cómo especificar los parámetros al realizar un inventario de archivos.

- En Linux y macOS, recopile metadatos de los archivos .sh en el directorio `/home/ec2-user` y excluya todos los subdirectorios.

```
[{"Path":"/home/ec2-user","Pattern":["*.sh", "*.sh"],"Recursive":false}]
```

- En Windows recopile metadatos de todos los archivos ".exe" de la carpeta Archivos de programa e incluya los subdirectorios de forma recursiva.

```
[{"Path":"C:\Program Files","Pattern":["*.exe"],"Recursive":true}]
```

- En Windows recopile metadatos de patrones de log específicos.

```
[{"Path":"C:\ProgramData\Amazon","Pattern":["*amazon*.log"],"Recursive":true}]
```

- Limite el número de directorios cuando ejecute una recopilación recursiva.

```
[{"Path":"C:\Users","Pattern":["*.ps1"],"Recursive":true, "DirScanLimit": 1000}]
```

Registro de Windows: puede recopilar claves y valores de registro de Windows. Puede elegir una ruta de clave y recopilar todas las claves y valores recursivamente. También puede recopilar una

clave de registro específica y su valor para una ruta específica. El inventario recopila la ruta de clave, el nombre, el tipo y el valor.

```
{
"Path": string,
"Recursive": true,
"ValueNames": array[string] // optional
}
```

- Ruta: la ruta de acceso a la clave de registro.
- Recursivo: valor booleano que indica si Inventory debe atravesar recursivamente las rutas de registro.
- ValueNames: matriz de nombres de valores para realizar un inventario de claves de registro. Si utiliza este parámetro, Systems Manager solo inventariará los nombres de valor especificados para la ruta especificada.

> **Note**
>
> Inventory recopila un máximo de 250 valores de clave de registro para todas las rutas especificadas.

A continuación, mostramos algunos ejemplos de cómo especificar los parámetros al realizar un inventario del registro de Windows.

- Recopile todas las claves y los valores de una ruta específica de forma recursiva.

```
[{"Path":"HKEY_LOCAL_MACHINE\SOFTWARE\Amazon","Recursive": true}]
```

- Recopile todas las claves y los valores de una ruta específica (la búsqueda recursiva está desactivada).

```
[{"Path":"HKEY_LOCAL_MACHINE\SOFTWARE\Intel\PSIS\PSIS_DECODER", "Recursive": false}]
```

- Recopile una clave específica utilizando la opción `ValueNames`.

```
{"Path":"HKEY_LOCAL_MACHINE\SOFTWARE\Amazon\MachineImage","ValueNames":["AMIName"]}
```

Configuración de Systems Manager Inventory

Antes de utilizar AWS Systems Manager Inventory para recopilar metadatos sobre las aplicaciones, los servicios y los componentes de AWS, entre otros, que se ejecutan en los nodos administrados, se recomienda configurar la sincronización de los datos de recursos para centralizar el almacenamiento de los datos de inventario en un único bucket de Amazon Simple Storage Service (Amazon S3). También le recomendamos que configure el monitoreo de eventos de inventario con Amazon EventBridge. Estos procesos facilitan la visualización y la administración de la recopilación y los datos de inventario.

Temas

- Cómo crear una sincronización de datos de recursos para Inventory
- Uso de EventBridge para monitorear eventos de Inventory

Cómo crear una sincronización de datos de recursos para Inventory

En este tema se describe cómo configurar y configurar la sincronización de datos de recursos para AWS Systems Manager Inventory. Para obtener información acerca de la sincronización de datos de recursos para Systems Manager Explorer, consulte Configuración de Systems Manager Explorer para mostrar datos de varias cuentas y regiones.

Acerca de la sincronización de datos de recursos

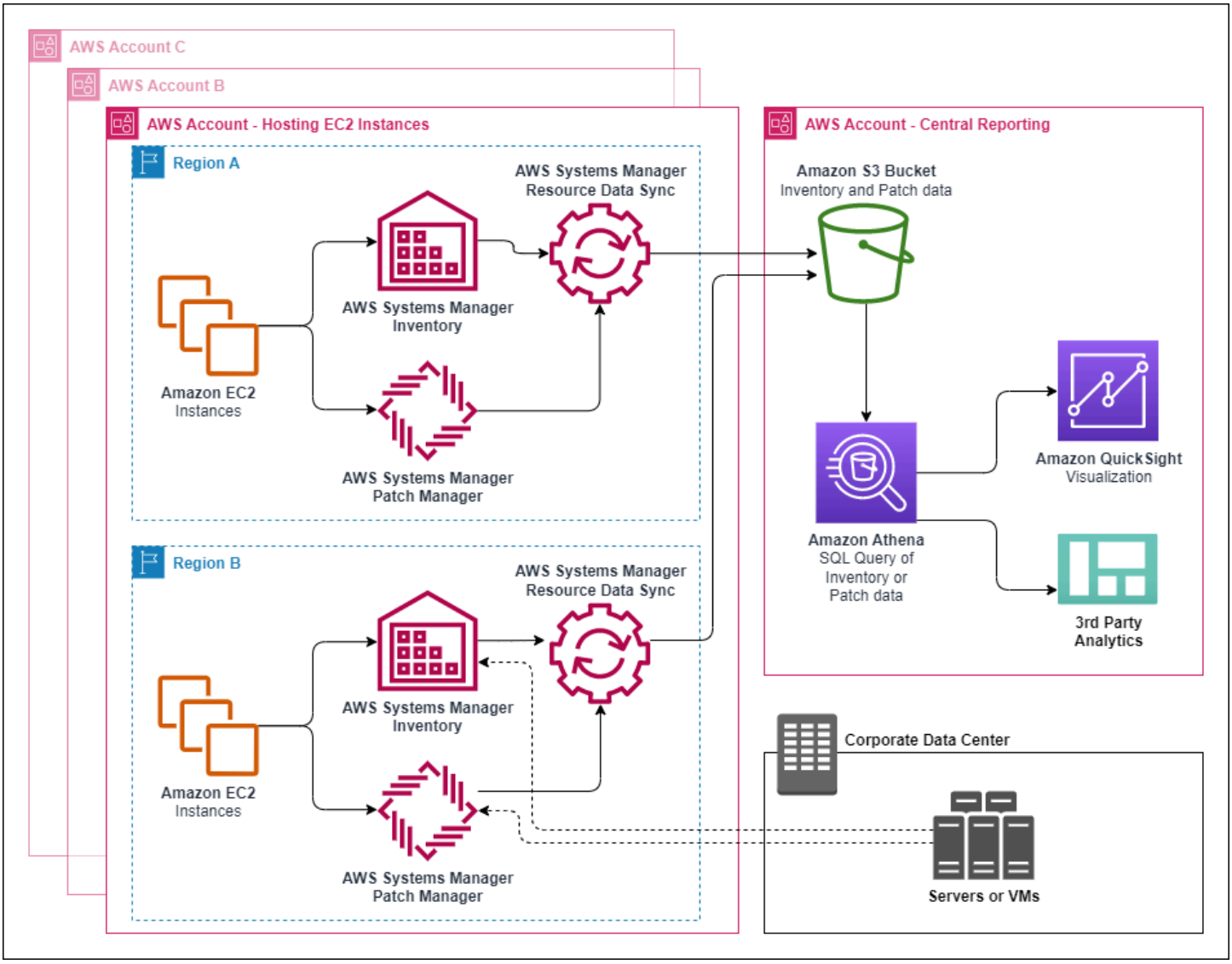
Puede utilizar la sincronización de datos de recursos de Systems Manager para enviar datos de inventario recopilados de todos los nodos administrados a un solo bucket de Amazon Simple Storage Service (Amazon S3). A continuación, la sincronización de datos de recursos actualiza automáticamente los datos centralizados cuando se recopilan los nuevos datos de inventario. Con todos los datos de inventario almacenados en un bucket de Amazon S3 de destino, puede usar servicios como Amazon Athena y Amazon QuickSight para consultar y analizar los datos agregados.

Por ejemplo, digamos que ha configurado inventario para recopilar datos sobre el sistema operativo (SO) y las aplicaciones que se ejecutan en una flota de 150 nodos administrados. Algunos de estos nodos se encuentran en un centro de datos en las instalaciones y otros se ejecutan en Amazon Elastic Compute Cloud (Amazon EC2) en varias Regiones de AWS. Si no ha configurado la sincronización de datos de recursos, tendrá que reunir manualmente la recopilación de datos de inventario para cada nodo administrado o tendrá que crear scripts que recopilen esta información. Después tendrá que portar los datos a una aplicación para poder ejecutar consultas y analizarlas.

Con la sincronización de datos de recursos se lleva a cabo una única operación que sincroniza todos los datos de inventario de todos los nodos administrados. Después de haber creado correctamente la sincronización, Systems Manager crea una base de referencia de todos los datos de inventario y la guarda en el bucket de Amazon S3 de destino. Cuando se recopilen nuevos datos de inventario, Systems Manager actualizará automáticamente los datos en el bucket de Amazon S3. A continuación, puede transferir de forma rápida y rentable los datos a Amazon Athena y Amazon QuickSight.

En el diagrama 1, se muestra cómo la sincronización de datos de recursos agrega los datos de inventario de Amazon EC2 y otros tipos de equipos en un entorno híbrido y multinube a un bucket de Amazon S3 de destino. Dicho diagrama muestra también cómo funciona la sincronización de datos de recursos con varias Cuentas de AWS y Regiones de AWS.

Diagrama 1: sincronización de datos de recursos con varias Cuentas de AWS y Regiones de AWS



Si elimina un nodo administrado, la sincronización de datos de recursos conserva el archivo de inventario del nodo eliminado. En el caso de los nodos en ejecución, la sincronización de datos de recursos, sin embargo, sobrescribe automáticamente los archivos de inventario antiguos cuando se crean y escriben archivos nuevos en el bucket de Amazon S3. Si desea realizar un seguimiento de los cambios de inventario con el paso del tiempo, puede utilizar el servicio AWS Config para realizar un seguimiento del tipo de recurso `SSM:ManagedInstanceInventory`. Para obtener más información, consulte Introducción a AWS Config.

Utilice los procedimientos de esta sección para crear una sincronización de datos de recursos para Inventory mediante las consolas de Amazon S3 y AWS Systems Manager. También puede utilizar AWS CloudFormation para crear o eliminar una sincronización de datos de recursos. Para utilizar AWS CloudFormation, agregue el recurso `AWS::SSM::ResourceDataSync` a la plantilla de AWS CloudFormation. Para obtener información, consulte uno de los siguientes recursos de documentación:

- AWS CloudFormation resource for resource data sync in AWS Systems Manager (blog)
- Uso de plantillas de AWS CloudFormation en la Guía del usuario de AWS CloudFormation

> **Note**
>
> Puede utilizar AWS Key Management Service (AWS KMS) para cifrar los datos de inventario en el bucket de Amazon S3. Si desea ver un ejemplo de cómo crear una sincronización cifrada mediante la AWS Command Line Interface (AWS CLI) y cómo trabajar con los datos centralizados en Amazon Athena y Amazon QuickSight, consulte Explicación: cómo utilizar la sincronización de datos de recursos para agregar datos de inventario.

Antes de empezar

Antes de crear una sincronización de datos de recursos, utilice el siguiente procedimiento para crear un bucket de Amazon S3 central para almacenar datos de inventario agregados. El procedimiento describe cómo asignar una política de bucket que permite a Systems Manager escribir datos de inventario en el bucket desde varias cuentas. Si ya tiene un bucket de Amazon S3 que desea utilizar para agregar datos de inventario para la sincronización de datos de recursos, debe configurar el bucket para que utilice la política en el siguiente procedimiento.

> **Note**
>
> Systems Manager Inventory no puede agregar datos a un bucket de Amazon S3 especificado si ese bucket está configurado para utilizar Object Lock. Compruebe que el bucket de Amazon S3 que cree o elija para la sincronización de datos de recursos no esté configurado para utilizar Object Lock de Amazon S3. Para obtener más información, consulte Cómo funciona Bloqueo de objetos de S3 en la Guía del usuario de Amazon Simple Storage Service.

Cómo crear y configurar un bucket de Amazon S3 para la sincronización de datos de recursos

1. Abra la consola de Amazon S3 en https://console.aws.amazon.com/s3.
2. Cree un bucket para almacenar los datos de Inventory agregados. Para obtener más información, consulte Crear un bucket en la Guía del usuario de Amazon Simple Storage Service. Anote el nombre de bucket y la Región de AWS donde lo creó.
3. Elija la pestaña Permisos y, a continuación, elija Política de bucket.
4. Copie y pegue la siguiente política de bucket en el editor de políticas. Sustituya *amzn-s3-demo-bucket* por el nombre del bucket de S3 que ha creado. Sustituya *account_ID_number* por un número de ID de Cuenta de AWS válido.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "SSMBucketPermissionsCheck",
            "Effect": "Allow",
            "Principal": {
                "Service": "ssm.amazonaws.com"
            },
            "Action": "s3:GetBucketAcl",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket"
        },
        {
            "Sid": " SSMBucketDelivery",
            "Effect": "Allow",
            "Principal": {
```

```
                "Service": "ssm.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket/*/
accountid=account_ID_number/*",
                "arn:aws:s3:::amzn-s3-demo-bucket/*/
accountid=account_ID_number/*",
                "arn:aws:s3:::amzn-s3-demo-bucket/*/
accountid=account_ID_number/*",
                "arn:aws:s3:::amzn-s3-demo-bucket/*/
accountid=account_ID_number/*"
            ],
            "Condition": {
                "StringEquals": {
                    "s3:x-amz-acl": "bucket-owner-full-control",
                    "aws:SourceAccount": "ID_number"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:ssm:*:111122223333:resource-
data-sync/*"
                }
            }
        }
    ]
}
```

5. Guarde los cambios.

Cómo crear una sincronización de datos de recursos para Inventory

Utilice el siguiente procedimiento para crear una sincronización de datos de recursos para Systems Manager Inventory mediante la consola de Systems Manager. Para obtener información acerca de cómo crear una sincronización de datos de recursos utilizando la AWS CLI, consulte Uso de la AWS CLI para configurar la recopilación de datos de inventario.

Cómo crear una sincronización de datos de recursos

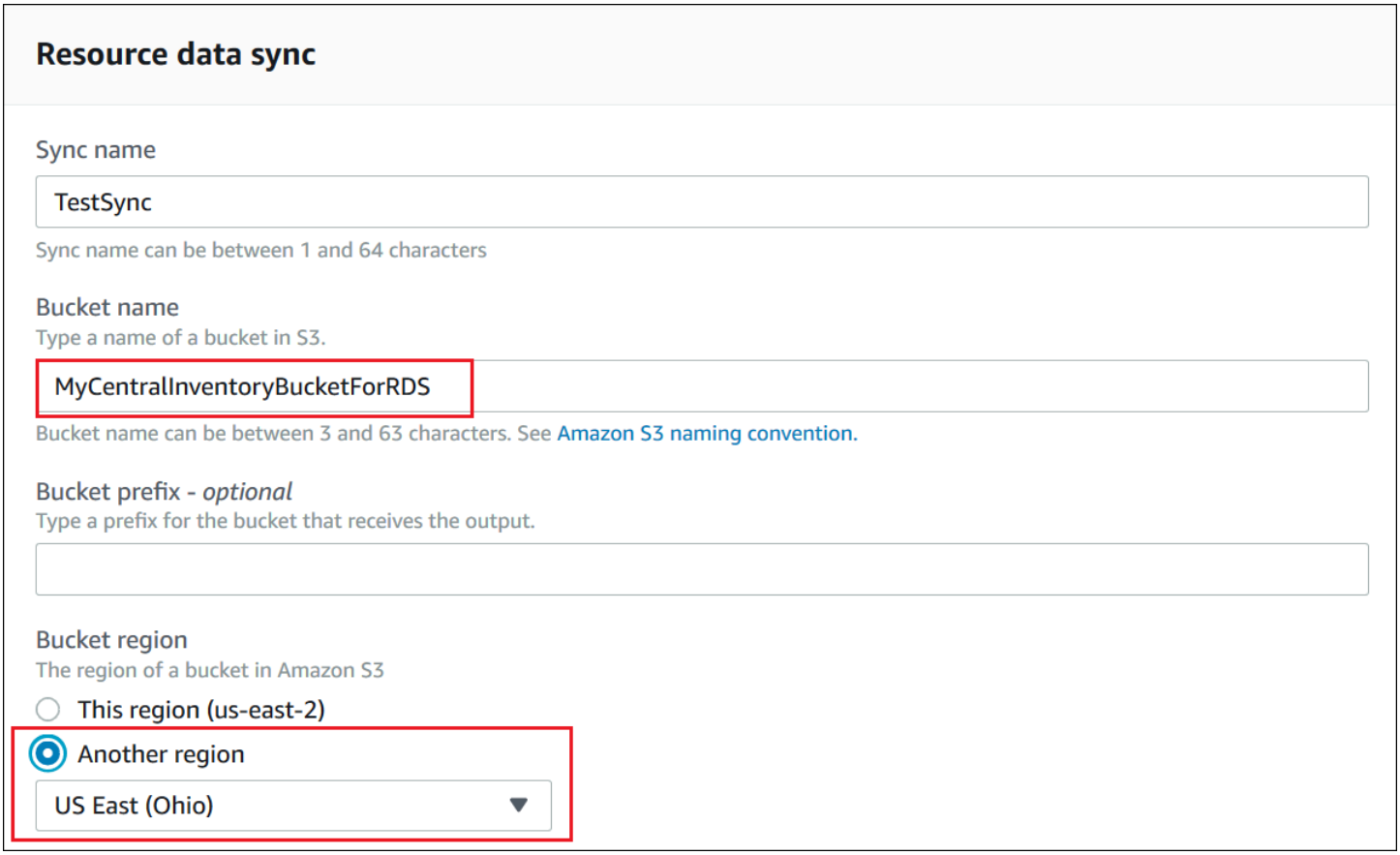
1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. En el menú Administración de cuentas, elija Sincronización de datos de recursos.

4. Elija Crear sincronización de datos de recursos.
5. En el campo Nombre de la sincronización ingrese el nombre de la configuración de sincronización.
6. En el campo Nombre del bucket, ingrese el nombre del bucket de Amazon S3 que creó con el procedimiento Para crear y configurar un bucket de Amazon S3 para la sincronización de datos de recursos.
7. (Opcional) En el campo Prefijo del bucket, ingrese el nombre de un prefijo de bucket de Amazon S3 (subdirectorio).
8. En el campo Región del bucket, elija This region si el bucket de Amazon S3 que ha creado se encuentra en la Región de AWS actual. Si el bucket se encuentra en otra Región de AWS, elija Otra región e ingrese el nombre de la región.

   > **Note**
   >
   > Si la sincronización y el bucket de Amazon S3 de destino se encuentran en regiones diferentes, es posible que esté sujeto a precios de transferencia de datos. Para obtener más información, consulte Precios de Amazon S3.

9. (Opcional) En el campo ARN de la clave del KMS, escriba o pegue un ARN de clave de KMS para cifrar los datos de inventario de Amazon S3.
10. Seleccione Crear.

Para sincronizar los datos de inventario de varias Regiones de AWS, debe crear una sincronización de datos de recursos en cada región. Repita este procedimiento en cada Región de AWS en la que desea recopilar los datos de inventario y enviarlos al bucket de Amazon S3 central. Cuando cree la sincronización en cada región, especifique el bucket de Amazon S3 central en el campo Nombre del bucket. A continuación, utilice la opción Región del bucket para elegir la región en la que ha creado el bucket de Amazon S3 central, tal y como se muestra en la siguiente captura de pantalla. La próxima vez que la asociación se ejecute para recopilar los datos de inventario, Systems Manager almacenará los datos en el bucket de Amazon S3 central.



Cómo crear una sincronización de datos de recursos de inventario para cuentas definidas en AWS Organizations

Puede sincronizar los datos de inventario de Cuentas de AWS definidas en AWS Organizations a un bucket de Amazon S3 central. Después de completar los siguientes procedimientos, los datos de inventario se sincronizan con prefijos de clave individuales de Amazon S3 en el bucket central. Cada prefijo de clave representa un ID de Cuenta de AWS diferente.

Antes de empezar

Antes de comenzar, compruebe que ha preparado y configurado Cuentas de AWS en AWS Organizations. Para obtener más información, consulte en la Guía del usuario de AWS Organizations.

Además, tenga en cuenta que debe crear la sincronización de datos de recursos basada en la organización para cada Región de AWS y Cuenta de AWS definida en AWS Organizations.

Cómo crear un bucket de Amazon S3 central

Utilice el siguiente procedimiento para crear un bucket de Amazon S3 central para almacenar datos de inventario agregados. El procedimiento describe cómo asignar una política de bucket que permite a Systems Manager escribir datos de inventario en el bucket desde el ID de cuenta de AWS Organizations. Si ya tiene un bucket de Amazon S3 que desea utilizar para agregar datos de inventario para la sincronización de datos de recursos, debe configurar el bucket para que utilice la política en el siguiente procedimiento.

Cómo crear y configurar un bucket de Amazon S3 para la sincronización de datos de recursos para varias cuentas definidas en AWS Organizations

1. Abra la consola de Amazon S3 en https://console.aws.amazon.com/s3.
2. Cree un bucket para almacenar los datos de inventario agregados. Para obtener más información, consulte Crear un bucket en la Guía del usuario de Amazon Simple Storage Service. Anote el nombre de bucket y la Región de AWS donde lo creó.
3. Elija la pestaña Permisos y, a continuación, elija Política de bucket.
4. Copie y pegue la siguiente política de bucket en el editor de políticas. Sustituya *amzn-s3-demo-bucket* y *organization-id* por el nombre del bucket de Amazon S3 que ha creado y un ID de cuenta de AWS Organizations válido.

   Si lo desea, reemplace *bucket-prefix* por el nombre de un prefijo de Amazon S3 (subdirectorio). Si no ha creado ningún prefijo, quite *bucket-prefix/* del ARN de la siguiente política.

   JSON

```
{
  "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:::amzn-s3-demo-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:::amzn-s3-demo-bucket/bucket-prefix/*/accountid=*/*"
      ]
    }
  ]
}
```

Cómo crear una sincronización de datos de recursos de inventario para cuentas definidas en AWS Organizations

En el siguiente procedimiento se describe cómo utilizar la AWS CLI para crear una sincronización de datos de recursos para cuentas definidas en AWS Organizations. Debe utilizar la AWS CLI para realizar esta tarea. También debe realizar este procedimiento para cada Región de AWS y Cuenta de AWS definida en AWS Organizations.

Cómo crear una sincronización de datos de recursos para una cuenta definida en AWS Organizations (AWS CLI)

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.
2. Ejecute el siguiente comando para comprobar que no dispone de ninguna otra sincronización de datos de recursos basada en AWS Organizations. Puede tener varias sincronizaciones estándar, incluidas varias sincronizaciones estándar y una sincronización basada en organizaciones. Pero, solo puede disponer de una sincronización de datos de recursos basada en la organización.

   ```
   aws ssm list-resource-data-sync
   ```

   Si el comando regresa otra sincronización de datos de recursos basada en organizaciones, debe eliminarla o elegir no crear una nueva.
3. Ejecute el siguiente comando para crear una sincronización de datos de recursos para una cuenta definida en AWS Organizations. Para amzn-s3-demo-bucket, especifique el nombre del bucket de Amazon S3 creado anteriormente en este tema. Si ha creado un prefijo (subdirectorio) para el bucket, especifique esta información para *prefix-name*.

   ```
   aws ssm create-resource-data-sync --sync-name name --s3-
   destination "BucketName=amzn-s3-demo-bucket,Prefix=prefix-
   name,SyncFormat=JsonSerDe,Region=Región de AWS, for example us-
   east-2,DestinationDataSharing={DestinationDataSharingType=Organization}"
   ```
4. Repita los pasos 2 y 3 para cada Región de AWS y Cuenta de AWS en el que desee sincronizar datos con el bucket de Amazon S3 central.

Cómo administrar sincronizaciones de datos de recursos

Cada Cuenta de AWS puede tener 5 sincronizaciones de datos de recursos por Región de AWS. Puede utilizar la consola de AWS Systems Manager Fleet Manager para administrar las sincronizaciones de datos de recursos.

Cómo visualizar las sincronizaciones de datos de recursos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.

2. En el panel de navegación, elija Fleet Manager.
3. En el menú desplegable Administración de cuenta, elija Sincronizaciones de datos de recursos.
4. Seleccione una sincronización de datos de recursos de la tabla y luego, elija Ver detalles para ver la información de la sincronización de datos de recursos.

Cómo eliminar una sincronización de datos de recursos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. En el menú desplegable Administración de cuenta, elija Sincronizaciones de datos de recursos.
4. Seleccione una sincronización de datos de recursos de la tabla y luego, elija Eliminar.

Uso de EventBridge para monitorear eventos de Inventory

Puede configurar una regla en Amazon EventBridge para crear un evento en respuesta a cambios de estado de recursos de AWS Systems Manager Inventory. EventBridge admite eventos para los siguientes cambios de estado de Inventory. Todos los eventos se envían en la medida de lo posible.

Tipo de inventario personalizado eliminado para una instancia específica: si se configura una regla para monitorear este evento, EventBridge crea un evento cuando se elimina un tipo de inventario personalizado en un nodo administrado específico. EventBridge envía un evento por nodo por tipo de inventario personalizado. A continuación, se muestra un patrón de eventos de ejemplo.

```
{
    "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"
    }
}
```

Evento de tipo de inventario personalizado eliminado para todas las instancias: si se configura una regla para monitorear este evento, EventBridge crea un evento cuando se elimina un tipo de inventario personalizado para todos los nodos administrados. A continuación, se muestra un patrón de eventos de ejemplo.

```
{
    "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"
    }
}
```

Llamada PutInventory con evento de versión de esquema anterior: si se configura una regla para monitorear este evento, EventBridge crea un evento cuando se realiza una llamada PutInventory que utiliza una versión de esquema inferior al esquema actual. Este evento se aplica a todos los tipos de inventario. A continuación, se muestra un patrón de eventos de ejemplo.

```
{
    "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"
    }
}
```

Para obtener información acerca de cómo configurar EventBridge para monitorear estos eventos, consulte Configuración de EventBridge para eventos de Systems Manager.

Configuración de la recopilación de inventario

Esta sección describe cómo configurar la recopilación de AWS Systems Manager Inventory en uno o varios nodos administrados mediante la consola de Systems Manager. Si desea ver un ejemplo de cómo configurar la recopilación de inventario mediante la AWS Command Line Interface (AWS CLI), consulte Uso de la AWS CLI para configurar la recopilación de datos de inventario.

Cuando configure la recopilación de Inventory, comience creando una asociación de AWS Systems Manager State Manager. Systems Manager recopila los datos de inventario cuando se ejecuta la asociación. Si no crea la asociación en primer lugar e intenta invocar el complemento `aws:softwareInventory` mediante, por ejemplo, el uso de AWS Systems Manager Run Command , el sistema regresará el siguiente error: `The aws:softwareInventory plugin can only be invoked via ssm-associate.`

> **Note**
>
> Tenga en cuenta el siguiente comportamiento si crea varias asociaciones de inventario para un nodo administrado:

- A cada nodo se le puede asignar una asociación de inventario que se dirija a todos los nodos (--targets "Key=InstanceIds,Values=*").
- A cada nodo también se le puede asignar una asociación específica que utilice pares clave-valor de etiqueta o un grupo de recursos de AWS.
- Si a un nodo se le asignan varias asociaciones de inventario, el estado muestra Omitido para la asociación que no se ha ejecutado. La asociación que se ha ejecutado más recientemente muestra el estado real de la asociación de inventario.
- Si a un nodo se le asignan varias asociaciones de inventario y cada una utiliza un par clave-valor de etiqueta, esas asociaciones de inventario no se ejecutan en el nodo debido al conflicto de etiqueta. La asociación sigue ejecutándose en nodos que no tienen el conflicto clave-valor de etiqueta.

Antes de empezar

Complete las siguientes tareas antes de configurar la recopilación de inventario.

- Actualice AWS Systems Manager SSM Agent en los nodos que desee inventariar. Si ejecuta la versión más reciente del SSM Agent, se asegurará de poder recopilar metadatos de todos los tipos de inventario admitidos. Para obtener información acerca de cómo actualizar el SSM Agent mediante State Manager, consulte Tutorial: actualización automática de SSM Agent con AWS CLI.
- Verifique que haya completado los requisitos de configuración para las instancias de Amazon Elastic Compute Cloud (Amazon EC2) y las máquinas que no sean de EC2 en un entorno híbrido y multinube. Para obtener más información, consulta Configuración de nodos administrados para AWS Systems Manager.
- Para los nodos de Microsoft Windows, compruebe que el nodo administrado está configurado con Windows PowerShell 3.0 (o posterior). SSM Agent utiliza el `ConvertTo-Json` cmdlet en PowerShell para convertir los datos de inventario de la actualización de Windows al formato requerido.
- (Opcional) Cree una sincronización de datos de recursos para almacenar de forma centralizada los datos de inventario en un bucket de Amazon S3. A continuación, la sincronización de datos de recursos actualiza automáticamente los datos centralizados cuando se recopilan nuevos datos de inventario. Para obtener más información, consulte Explicación: cómo utilizar la sincronización de datos de recursos para agregar datos de inventario.

- (Opcional) Cree un archivo JSON para recopilar el inventario personalizado. Para obtener más información, consulte Uso del inventario personalizado.

Inventario de todos los nodos administrados de la Cuenta de AWS

Puede inventariar fácilmente todos los nodos administrados de la Cuenta de AWS mediante la creación de una asociación de inventario global. Una asociación de inventario global realiza las siguientes acciones:

- Aplica automáticamente la configuración de inventario global (la asociación) a todos los nodos administrados existentes en la Cuenta de AWS. Los nodos administrados que ya tienen una asociación de inventario se omiten cuando se aplica y ejecuta la asociación de inventario global. Si un nodo se omite, en el mensaje de estado detallado aparece `Overridden By Explicit Inventory Association`. Estos nodos se omiten en la asociación global, pero siguen aportando datos sobre el inventario cuando se ejecuta la asignación de inventario asignada.
- Agrega automáticamente a la asociación de inventario global los nuevos nodos creados en la Cuenta de AWS.

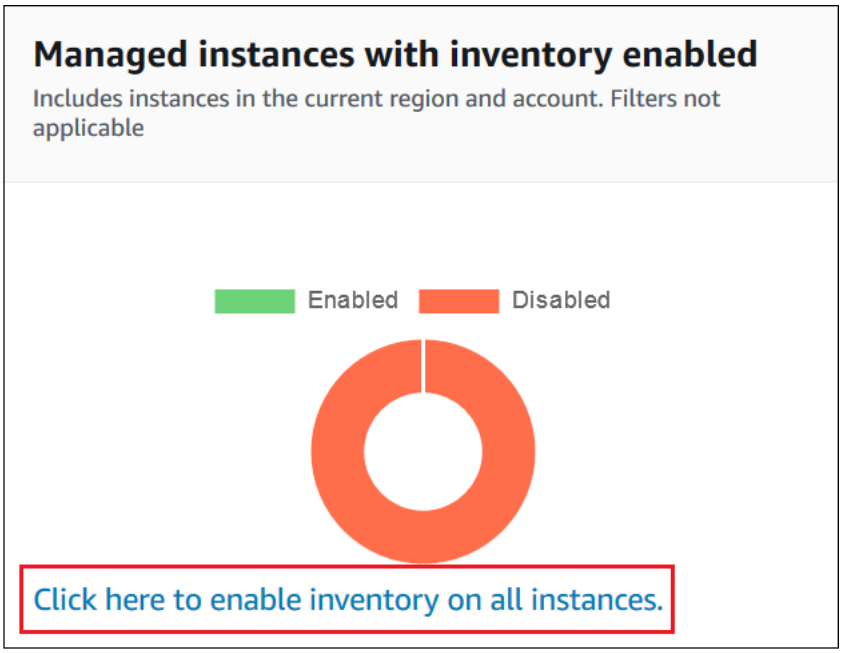
> **Note**
>
> - Si un nodo administrado está configurado para la asociación de inventario global y se asigna una asociación específica a dicho nodo, Systems Manager Inventory deja de dar prioridad a la asociación global y aplica la asociación específica.
> - Las asociaciones de inventario globales están disponibles en la versión 2.0.790.0 y versiones posteriores del SSM Agent. Para obtener más información sobre cómo actualizar el SSM Agent en los nodos, consulte Actualización de SSM Agent mediante Run Command.

Configuración de la recopilación de inventario con un solo clic (consola)

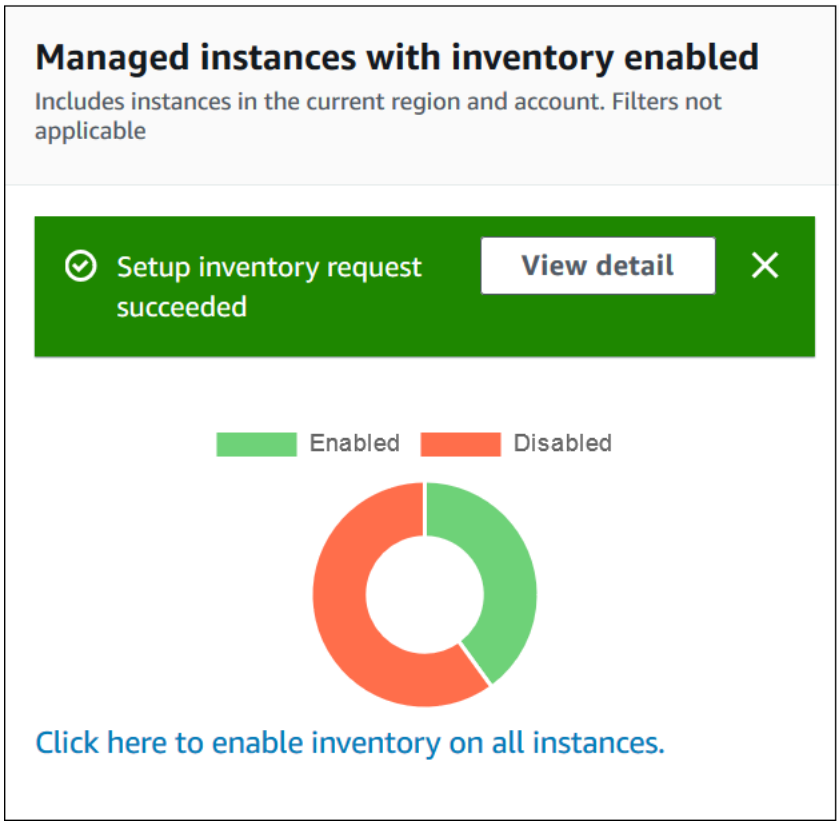
Utilice el siguiente procedimiento para configurar Systems Manager Inventory para todos los nodos administrados en la Cuenta de AWS y en una única Región de AWS.

Para configurar todos los nodos administrados en la región actual para Systems Manager Inventory

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Inventory.
3. En la tarjeta Instancias administradas con el inventario habilitado, elija Click here to enable inventory on all instances.



Si se realiza correctamente, la consola muestra el siguiente mensaje.



En función del número de nodos administrados en la cuenta, es posible que la asociación de inventario global tarde varios minutos en aplicarse. Espere unos minutos y actualice la página. Compruebe que el gráfico cambia para reflejar que el inventario está configurado en todos los nodos administrados.

Configuración de la recopilación mediante el uso de la consola

En esta sección, se incluye información acerca de cómo configurar Systems Manager Inventory para recopilar metadatos de los nodos administrados mediante la consola de Systems Manager. Puede recopilar rápidamente los metadatos de todos los nodos de una Cuenta de AWS específica (y de los futuros nodos que se creen en esa cuenta) o, si lo prefiere, puede recopilar los datos de inventario de manera selectiva mediante el uso de etiquetas o de ID de nodos.

> **Note**
>
> Antes de completar este procedimiento, compruebe si ya existe una asociación de inventario global. Si ya existe una asociación de inventario global, cada vez que lance una instancia nueva, se le aplicará la asociación y se creará un inventario de la instancia nueva.

Para configurar la recopilación de inventario

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Inventory.
3. Elija Setup Inventory.
4. En la sección Destinos, identifique los nodos en los que desea ejecutar esta operación. Para ello, seleccione una de las opciones siguientes.

   - Selecting all managed instances in this account: esta opción selecciona todos los nodos administrados para los que no existe ninguna asociación de inventario. Si elige esta opción, los nodos que ya tengan asociaciones de inventario se omitirán durante la recopilación del inventario y aparecerán con el estado Omitido en los resultados del inventario. Para obtener más información, consulte Inventario de todos los nodos administrados de la Cuenta de AWS.
   - Especificación de una etiqueta: utilice esta opción para especificar una sola etiqueta para identificar los nodos de la cuenta de los que desea recopilar el inventario. Si utiliza una etiqueta, todos los nodos que se creen en el futuro con la misma etiqueta también se incluirán en el inventario. Si hay una asociación de inventario existente en todos los nodos y se utiliza una etiqueta para seleccionar nodos específicos como destino de un inventario distinto, se invalidará la pertenencia del nodo en el grupo de destino All managed instances. Los nodos administrados con la etiqueta especificada se omitirán en las recopilaciones de inventario que se realicen en All managed instances en el futuro.
   - Selección manual de instancias: utilice esta opción para elegir determinados nodos administrados de la cuenta. Si se eligen explícitamente nodos concretos a través de esta opción, se invalidan las asociaciones de inventario del destino All managed instances. El nodo se omitirá en las recopilaciones de inventario que se realicen en All managed instances en el futuro.

   > **Note**
   >
   > Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

5. En la sección Schedule, elija la frecuencia con la que desea que el sistema recopile los metadatos de inventario de los nodos.

6. En la sección Parámetros, utilice las listas para activar o desactivar los diferentes tipos de recopilación de inventario. Para obtener más información acerca de cómo recopilar el inventario de archivos y del registro de Windows, consulte Uso del inventario de archivos y del registro de Windows.
7. En la sección Avanzado, seleccione Sync inventory execution logs to an Amazon S3 bucket si desea almacenar el estado de ejecución de la asociación en un bucket de Amazon S3.
8. Elija Setup Inventory. Systems Manager crea una asociación de State Manager y ejecuta inmediatamente Inventory en los nodos.
9. En el panel de navegación, elija State Manager. Compruebe que se haya creado una nueva asociación que utilice el documento **`AWS-GatherSoftwareInventory`**. La programación de asociación utiliza una expresión rate. Además, verifique que el campo Estado muestre Correcto. Si eligió la opción Sync inventory execution logs to an Amazon S3 bucket, podrá ver los datos de registro en Amazon S3 después de unos minutos. Si desea ver los datos de inventario de un nodo específico, elija Instancias administradas en el panel de navegación.
10. Elija un nodo y, a continuación, elija Ver detalles.
11. En la página de detalles del nodo, elija Inventory. Utilice las listas Inventory type para filtrar el inventario.

Consulta de datos de Inventory de varias regiones y cuentas

AWS Systems Manager Inventory se integra a Amazon Athena para ayudarlo a consultar los datos de inventario de varias Regiones de AWS y Cuentas de AWS. La integración de Athena utiliza la sincronización de datos de recursos, de modo que podrá visualizar los datos de inventario de todos los nodos administrados en la página Detail View (Vista de detalles) en la consola de AWS Systems Manager.

> **Important**
>
> Esta característica utiliza AWS Glue para rastrear los datos del bucket de Amazon Simple Storage Service (Amazon S3) y Amazon Athena para consultar los datos. En función de la cantidad de datos que haya consultado y rastreado, puede se le apliquen cargos por el uso de estos servicios. Con AWS Glue, paga una tarifa por hora, que se factura por segundo, por los rastreadores (detección de datos) y los trabajos de ETL (procesamiento y carga de datos). Con Athena, se le cobra en función del volumen de datos escaneados por cada consulta. Lo animamos a que consulte las directrices de precios de estos servicios antes de

utilizar la integración de Amazon Athena a Systems Manager Inventory. Para obtener más información, consulte Precios de Amazon Athena y Precios de AWS Glue.

Puede ver los datos de inventario en la página Detailed View (Vista detallada) en todas las Regiones de AWS en las que Amazon Athena está disponible. Para ver una lista de las regiones admitidas, consulte Puntos de conexión de Amazon Athena en la Referencia general de Amazon Web Services.

Antes de empezar

La integración de Athena utiliza la sincronización de datos de recursos. Debe preparar y configurar la sincronización de datos de recursos para utilizar esta característica. Para obtener más información, consulte Explicación: cómo utilizar la sincronización de datos de recursos para agregar datos de inventario.

Además, tenga en cuenta que la página Detailed View (Vista detallada) muestra los datos de inventario del propietario del bucket de Amazon S3 central que utiliza la sincronización de datos de recursos. Si no es el propietario del bucket de Amazon S3 central, no verá los datos de inventario en la página Detailed View (Vista detallada).

Configuración del acceso

Antes de que pueda consultar y visualizar los datos de varias cuentas y regiones en la página Vista detallada en la consola de Systems Manager, debe configurar su entidad de IAM con permiso para ver los datos.

Si los datos de inventario se almacenan en un bucket de Amazon S3 que utiliza cifrado de AWS Key Management Service (AWS KMS), también configure la entidad de IAM y el rol de servicio `Amazon-GlueServiceRoleForSSM` para el cifrado de AWS KMS.

Temas

- Configuración de la entidad de IAM para acceder a la página Vista detallada
- (Opcional) Configuración de permisos para ver datos cifrados de AWS KMS

Configuración de la entidad de IAM para acceder a la página Vista detallada

A continuación se describen los permisos mínimos necesarios para ver los datos del inventario en la página Vista Detallada.

La política administrada **AWSQuicksightAthenaAccess**

El siguiente `PassRole` y bloque de permisos necesarios adicionales

JSON

```
{
    "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": [
                "arn:aws:iam::123456789012:role/SSMInventoryGlueRole",
                "arn:aws:iam::123456789012:role/SSMInventoryServiceRole"
            ],
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "glue.amazonaws.com"
                }
            }
        },
        {
            "Sid": "iamRoleCreation",
            "Effect": "Allow",
            "Action": [
                "iam:CreateRole",
                "iam:AttachRolePolicy"
            ],
            "Resource": "arn:aws:iam::111122223333:role/*"
```

```
        },
        {
            "Sid": "iamPolicyCreation",
            "Effect": "Allow",
            "Action": "iam:CreatePolicy",
            "Resource": "arn:aws:iam::111122223333:policy/*"
        }
    ]
}
```

(Opcional) Si el bucket de Amazon S3 que se utiliza para almacenar datos de inventario está cifrado con AWS KMS, también agregue el siguiente bloque a la política.

```
{
    "Effect": "Allow",
    "Action": [
        "kms:Decrypt"
    ],
    "Resource": [
        "arn:aws:kms:Region:account_ID:key/key_ARN"
    ]
}
```

Para dar acceso, agregue permisos a los usuarios, grupos o roles:

- Usuarios y grupos en AWS IAM Identity Center:

  Cree un conjunto de permisos. Siga las instrucciones de Creación de un conjunto de permisos en la Guía del usuario de AWS IAM Identity Center.
- Usuarios gestionados en IAM a través de un proveedor de identidades:

  Cree un rol para la federación de identidades. Siga las instrucciones descritas en Creación de un rol para un proveedor de identidad de terceros (federación) en la Guía del usuario de IAM.
- Usuarios de IAM:
  - Cree un rol que el usuario pueda aceptar. Siga las instrucciones descritas en Creación de un rol para un usuario de IAM en la Guía del usuario de IAM.
  - (No recomendado) Adjunte una política directamente a un usuario o añada un usuario a un grupo de usuarios. Siga las instrucciones descritas en Adición de permisos a un usuario (consola) de la Guía del usuario de IAM.

(Opcional) Configuración de permisos para ver datos cifrados de AWS KMS

Si el bucket de Amazon S3 que se utiliza para almacenar datos de inventario está cifrado con AWS Key Management Service (AWS KMS), configure su entidad de IAM y el rol Amazon-GlueServiceRoleForSSM con permisos `kms:Decrypt` para la clave de AWS KMS.

Antes de empezar

Para proporcionar los permisos `kms:Decrypt` de la clave AWS KMS, agregue el siguiente bloque de políticas a su entidad de IAM:

```
{
    "Effect": "Allow",
    "Action": [
        "kms:Decrypt"
    ],
    "Resource": [
        "arn:aws:kms:Region:account_ID:key/key_ARN"
    ]
}
```

Si aún no lo ha hecho, complete ese procedimiento y agregue los permisos de `kms:Decrypt` para la clave de AWS KMS.

Utilice el siguiente procedimiento para configurar el rol Amazon-GlueServiceRoleForSSM con los permisos de `kms:Decrypt` para la clave de AWS KMS.

Para configurar el rol Amazon-GlueServiceRoleForSSM con los permisos de **`kms:Decrypt`**

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, elija Roles y, a continuación, utilice el campo de búsqueda para ubicar el rol Amazon-GlueServiceRoleForSSM. Se abre la página Resumen.
3. Utilice el campo de búsqueda para buscar el rol Amazon-GlueServiceRoleForSSM. Elija el nombre del rol . Se abre la página Resumen.
4. Elija el nombre del rol . Se abre la página Resumen.
5. Elija Agregar política insertada. Se abre la página Crear política.
6. Seleccione la pestaña JSON.
7. Elimine el texto JSON existente en el editor y, a continuación, copie y pegue la siguiente política en el editor de JSON.

JSON

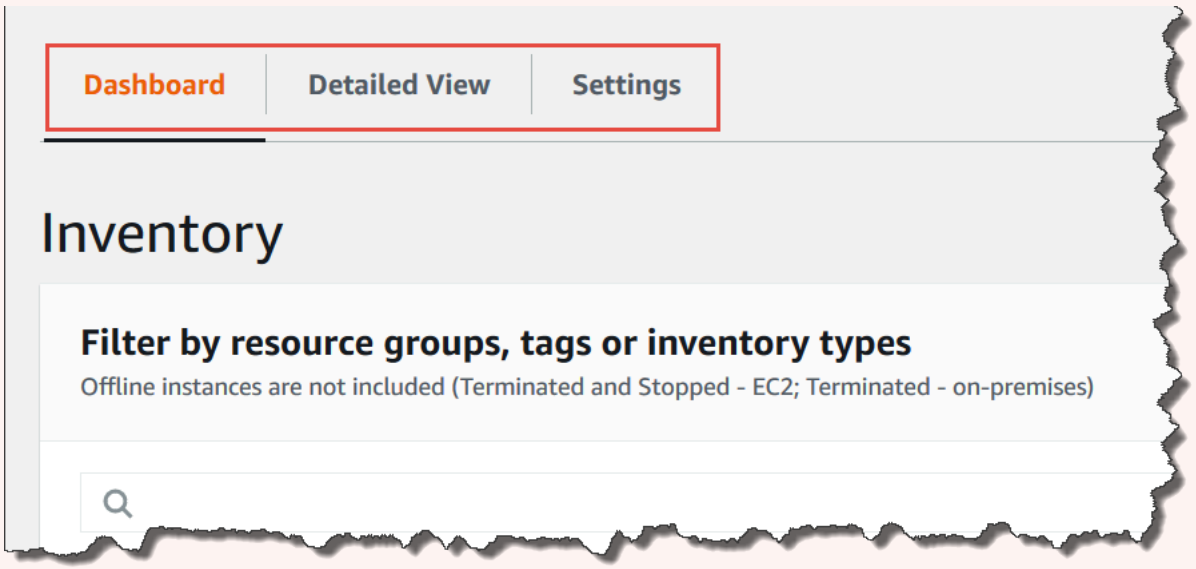
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:us-east-1:111122223333:key/key_ARN"
            ]
        }
    ]
}
```

8. Elija Revisar la política
9. En la página Revisar política, escriba un nombre en el campo Nombre.
10. Seleccione Crear política.

Consulta de datos en la página Detailed Inventory View (Vista de inventario detallado)

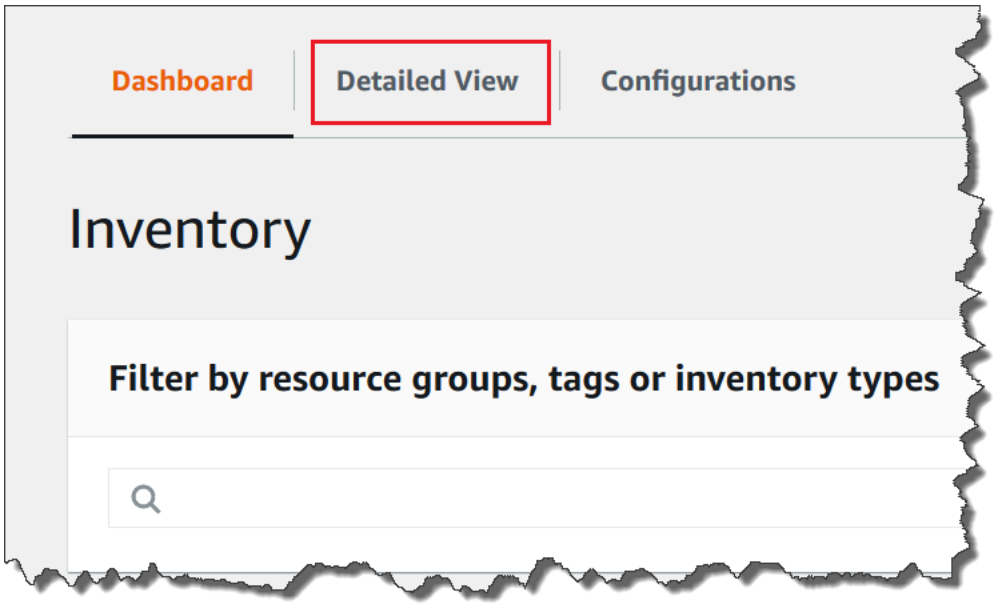
Utilice el siguiente procedimiento para ver los datos de inventario de varias Regiones de AWS y Cuentas de AWS en la página Detailed View (Vista detallada) de Systems Manager Inventory.

> **Important**
>
> La página Detailed View (Vista detallada) de Inventory solo está disponible en las Regiones de AWS que ofrece Amazon Athena. Si las siguientes pestañas no se muestran en la página de Systems Manager Inventory, significa que Athena no está disponible en la región y que no puede utilizar la Detailed View (Vista detallada) para consultar datos.

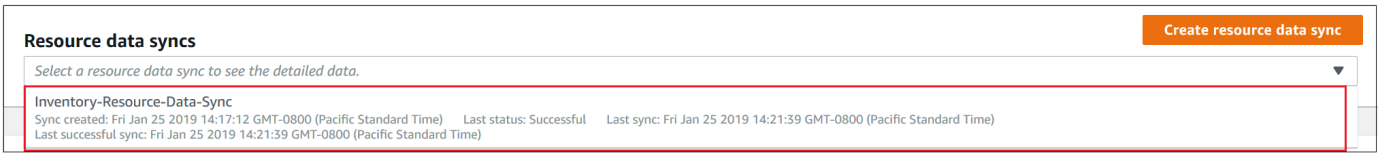


Para ver los datos de inventario de varias regiones y cuentas en la consola de AWS Systems Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Inventory.
3. Elija la pestaña Detailed View (Vista detallada).



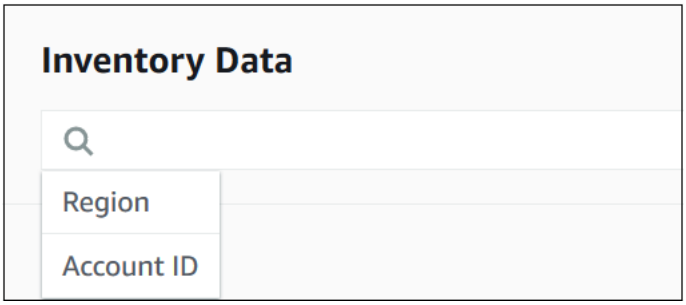
4. Seleccione la sincronización de datos de recursos para la que desea consultar datos.



5. En la lista Inventory Type (Tipo de inventario), seleccione el tipo de datos de inventario que desea consultar y, a continuación, pulse Enter.



6. Para filtrar los datos, elija la barra Filtro y, a continuación, elija una opción de filtro.



Puede utilizar el botón Export to CSV (Exportar a CSV) para ver el conjunto de consultas actual en una aplicación de hoja de cálculo como Microsoft Excel. También puede utilizar los botones Query History (Historial de consultas) y Run Advanced Queries (Ejecutar consultas avanzadas) para ver detalles sobre el historial e interactuar con sus datos en Amazon Athena.

Edición de la programación del rastreador de AWS Glue

AWS Glue rastrea los datos de inventario en el bucket de Amazon S3 central dos veces al día y de forma predeterminada. Si cambia con frecuencia los tipos de datos que se recopilarán en los nodos, es posible que desee rastrear los datos con mayor frecuencia, tal y como se describe en el siguiente procedimiento.

> **Important**
>
> Con AWS Glue, paga una tarifa por hora por Cuenta de AWS, que se factura por segundo, por los rastreadores (detección de datos) y los trabajos de ETL (procesamiento y carga de

datos). Antes de cambiar la programación del rastreador, consulte la página de precios de AWS Glue

Para cambiar la programación del rastreador de datos de inventario

1. Abra la consola de AWS Glue en https://console.aws.amazon.com/glue/.
2. En el panel de navegación, elija Crawlers (Rastreadores).
3. En la lista de rastreadores, elija la opción que aparece junto al rastreador de datos de Systems Manager Inventory. El nombre del rastreador utiliza el formato siguiente:

   `AWSSystemsManager-s3-bucket-name-Region-account_ID`
4. Elija Acción y, a continuación, seleccione Edit crawler (Editar rastreador).
5. En el panel de navegación, seleccione Schedule (Programación).
6. En el campo Expresión Cron, especifique una nueva programación mediante un formato Cron. Para obtener más información acerca del formato Cron, consulte Programaciones basadas en tiempo para trabajos y rastreadores en la Guía para desarrolladores de AWS Glue.

> **Important**
>
> Puede detener el rastreador para dejar de incurrir en gastos de AWS Glue. Si pone en pausa el rastreador o si cambia la frecuencia de tal forma que los datos se rastreen con menos frecuencia, Detailed View (Vista detallada) de Inventory podría mostrar datos que no están actualizados.

Consulta de una recopilación de inventario mediante filtros

Después de recopilar datos de inventario, puede utilizar las características de filtro de AWS Systems Manager para consultar una lista de nodos administrados que cumplan determinados criterios de filtro.

Para consultar nodos en función de filtros de inventario

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Inventory.

3. En la sección Filter by resource groups, tags or inventory types elija el cuadro de filtro. Aparecerá una lista de filtros predefinidos.
4. Elija un atributo que servirá de filtro. Por ejemplo, elija **`AWS:Application`**. Si el sistema se lo solicita, elija un atributo secundario como filtro. Por ejemplo, elija **`AWS:Application.Name`**.
5. Elija un delimitador en la lista. Por ejemplo, elija Begin with. Aparecerá un cuadro de texto en el filtro.
6. Ingrese un valor en el cuadro de texto. Por ejemplo, ingrese Amazon (SSM Agent se llama Amazon SSM Agent).
7. Pulse Enter. El sistema devuelve una lista de nodos administrados que contienen el nombre de una aplicación que comienza por la palabra Amazon.

> **Note**
>
> Puede combinar varios filtros para limitar la búsqueda.

Agregación de datos de Inventory

Después de configurar los nodos administrados de AWS Systems Manager Inventory, puede ver los totales agregados de los datos de Inventory. Por ejemplo, supongamos que ha configurado decenas o cientos de nodos administrados para que recopilen el tipo de inventario `AWS:Application`. La información de esta sección le permite ver un recuento exacto del número de nodos que están configurados para recopilar estos datos.

También puede ver detalles específicos de inventario si efectúa la agregación por un tipo de datos. Por ejemplo, el tipo de inventario `AWS:InstanceInformation` recopila información de la plataforma del sistema operativo con el tipo de datos `Platform`. Al agregar datos según el tipo de datos `Platform`, puede ver rápidamente el número de nodos en los que se ejecuta Windows, en los que se ejecuta Linux y en los que se ejecuta macOS.

En los procedimientos de esta sección, se describe cómo ver los totales agregados de datos de inventario mediante la AWS Command Line Interface (AWS CLI). También puede ver los recuentos agregados preconfigurados en la consola de AWS Systems Manager, en la página Inventario. Estos paneles preconfigurados se denominan Inventory Insights (Información de inventario) y ofrecen la solución a los problemas de configuración de Inventory con un solo clic.

Tenga en cuenta los siguientes detalles importantes sobre los recuentos de agregación de los datos de inventario:

- Si termina un nodo administrado que está configurado para recopilar datos de inventario, Systems Manager conserva los datos durante 30 días y, luego, los elimina. Para los nodos en ejecución, los sistemas eliminan los datos de inventario con más de 30 días de antigüedad. Si necesita almacenar datos de inventario durante más de 30 días, puede usar AWS Config para registrar el historial o para realizar una consulta periódica y cargar los datos en un bucket de Amazon Simple Storage Service (Amazon S3).
- Si un nodo se había configurado previamente para informar sobre un tipo de datos de inventario específico, por ejemplo, `AWS:Network` y, más adelante, se cambia la configuración para dejar de recopilar ese tipo de datos, los totales de agregación siguen mostrando los datos `AWS:Network` hasta que se termine el nodo y hayan pasado 30 días.

Para obtener información sobre cómo configurar y recopilar rápidamente los datos de inventario de todos los nodos en una Cuenta de AWS específica (y de los futuros nodos que podrían crearse en dicha cuenta), consulte Inventario de todos los nodos administrados de la Cuenta de AWS.

Temas

- Agregación de datos de Inventory para ver recuentos de nodos que recopilen determinados tipos de datos
- Agregación de datos de Inventory mediante grupos para ver qué nodos están configurados o no para recopilar un tipo de inventario

Agregación de datos de Inventory para ver recuentos de nodos que recopilen determinados tipos de datos

Puede utilizar la operación GetInventory de la API de AWS Systems Manager para ver los totales agregados de nodos que recopilan uno o varios tipos de inventario y tipos de datos. Por ejemplo, el tipo de inventario `AWS:InstanceInformation` le permite ver el total de sistemas operativos mediante la operación GetInventory de la API con el tipo de datos `AWS:InstanceInformation.PlatformType`. A continuación, se muestra un ejemplo de comando de la AWS CLI y su resultado.

```
aws ssm get-inventory --aggregators "Expression=AWS:InstanceInformation.PlatformType"
```

El sistema devuelve información similar a la siguiente.

```
{
   "Entities":[
      {
         "Data":{
            "AWS:InstanceInformation":{
               "Content":[
                  {
                     "Count":"7",
                     "PlatformType":"windows"
                  },
                  {
                     "Count":"5",
                     "PlatformType":"linux"
                  }
               ]
            }
         }
      }
   ]
}
```

Introducción

Determine los tipos de inventario y los tipos de datos cuyos recuentos desea ver. Puede ver una lista de los tipos de inventario y de datos que admiten la agregación mediante la ejecución del siguiente comando en la AWS CLI.

```
aws ssm get-inventory-schema --aggregator
```

El comando devuelve una lista JSON de tipos de inventario y tipos de datos que admiten la agregación. El campo TypeName muestra los tipos de inventario admitidos. Y el campo Name muestra cada tipo de datos. Por ejemplo, en la siguiente lista, el tipo de inventario `AWS:Application` incluye tipos de datos para `Name` y `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"
                }
            ]
```

```
        }
    ]
}
```

Puede agregar datos para cualquiera de los tipos de inventario indicados mediante un comando con la sintaxis siguiente.

```
aws ssm get-inventory --aggregators "Expression=InventoryType.DataType"
```

Estos son algunos ejemplos.

Ejemplo 1

En este ejemplo, se calcula el total de los roles de Windows que utilizan los nodos.

```
aws ssm get-inventory --aggregators "Expression=AWS:WindowsRole.Name"
```

Ejemplo 2

En este ejemplo, se calcula el total de las aplicaciones instaladas en los nodos.

```
aws ssm get-inventory --aggregators "Expression=AWS:Application.Name"
```

Combinación de varios agregadores

También puede combinar varios tipos de inventario y tipos de datos en un mismo comando para entender mejor los datos. Estos son algunos ejemplos.

Ejemplo 1

En este ejemplo, se calcula el total de los tipos de sistemas operativos que utilizan los nodos. También devuelve el nombre específico de los sistemas operativos.

```
aws ssm get-inventory --aggregators '[{"Expression":
 "AWS:InstanceInformation.PlatformType", "Aggregators":[{"Expression":
 "AWS:InstanceInformation.PlatformName"}]}]'
```

Ejemplo 2

En este ejemplo, se calcula el total de las aplicaciones que se ejecutan en los nodos y la versión específica de cada aplicación.

```
aws ssm get-inventory --aggregators '[{"Expression": "AWS:Application.Name",
 "Aggregators":[{"Expression": "AWS:Application.Version"}]}]'
```

Si lo prefiere, puede crear una expresión de agregación con uno o varios tipos de inventario y tipos de datos en un archivo JSON y llamar al archivo desde la AWS CLI. El JSON del archivo debe utilizar la sintaxis siguiente.

```
[
       {
           "Expression": "string",
           "Aggregators": [
               {
                  "Expression": "string"
               }
           ]
       }
]
```

Debe guardar el archivo con la extensión .json.

A continuación se muestra un ejemplo que utiliza varios tipos de inventario y tipos de datos.

```
[
       {
           "Expression": "AWS:Application.Name",
           "Aggregators": [
               {
                   "Expression": "AWS:Application.Version",
                   "Aggregators": [
                     {
                     "Expression": "AWS:InstanceInformation.PlatformType"
                     }
                   ]
               }
           ]
       }
]
```

Utilice el siguiente comando para llamar al archivo desde la AWS CLI.

```
aws ssm get-inventory --aggregators file://file_name.json
```

El comando devuelve información similar a la siguiente.


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

Agregación de datos de Inventory mediante grupos para ver qué nodos están configurados o no para recopilar un tipo de inventario

Los grupos en Systems Manager Inventory le permiten ver rápidamente el número de nodos administrados que están o que no están configurados para recopilar uno o varios tipos de inventarios. Con los grupos, debe especificar uno o varios tipos de inventario y un filtro que utiliza el operador `exists`.

Por ejemplo, suponga que tiene cuatro nodos administrados configurados para recopilar los siguientes tipos de inventario:

- Nodo 1: `AWS:Application`
- Nodo 2: `AWS:File`
- Nodo 3: `AWS:Application`, `AWS:File`
- Nodo 4: `AWS:Network`

Puede ejecutar el siguiente comando desde la AWS CLI para ver cuántos nodos están configurados para recopilar los tipos `AWS:Application` y `AWS:File inventory`. La respuesta también devuelve el número de nodos que no están configurados para recopilar ambos tipos de inventario.

```
aws ssm get-inventory --aggregators
 'Groups=[{Name=ApplicationAndFile,Filters=[{Key=TypeName,Values=[AWS:Application],Type=Exists}
{Key=TypeName,Values=[AWS:File],Type=Exists}]}]'
```

La respuesta del comando muestra que solo hay un nodo administrado configurado para recopilar los tipos de inventario `AWS:Application` y `AWS:File`.

```
{
   "Entities":[
      {
         "Data":{
            "ApplicationAndFile":{
               "Content":[
                  {
                     "notMatchingCount":"3"
                  },
                  {
                     "matchingCount":"1"
                  }
               ]
```

```
            }
         }
      }
   ]
}
```

> **Note**
>
> Los grupos no devuelven recuentos para los tipos de datos. Además, no se pueden desglosar los resultados para ver los ID de los nodos que están o que no están configurados para recopilar el tipo de inventario.

Si lo prefiere, puede crear una expresión de agregación con uno o varios tipos de inventario en un archivo JSON y llamar al archivo desde la AWS CLI. El JSON del archivo debe utilizar la sintaxis siguiente:

```
{
   "Aggregators":[
      {
         "Groups":[
            {
               "Name":"Name",
               "Filters":[
                  {
                     "Key":"TypeName",
                     "Values":[
                        "Inventory_type"
                     ],
                     "Type":"Exists"
                  },
                  {
                     "Key":"TypeName",
                     "Values":[
                        "Inventory_type"
                     ],
                     "Type":"Exists"
                  }
               ]
            }
         ]
      }
```

```
   ]
}
```

Debe guardar el archivo con la extensión .json.

Utilice el siguiente comando para llamar al archivo desde la AWS CLI.

```
aws ssm get-inventory --cli-input-json file://file_name.json
```

Ejemplos adicionales

Los siguientes ejemplos muestran cómo agregar datos de inventario para ver los nodos administrados que están o que no están configurados para recopilar los tipos de inventario especificados. Estos ejemplos utilizan la AWS CLI. Cada ejemplo incluye un comando completo con filtros que puede ejecutar desde la línea de comandos y un ejemplo de archivo input.json por si prefiere introducir la información en un archivo.

Ejemplo 1

En este ejemplo, se calcula el número de nodos que están o que no están configurados para recopilar los tipos de inventario `AWS:Application` o `AWS:File`.

Ejecute el siguiente comando desde la AWS CLI.

```
aws ssm get-inventory --aggregators
 'Groups=[{Name=ApplicationORFile,Filters=[{Key=TypeName,Values=[AWS:Application,
 AWS:File],Type=Exists}]}]'
```

Si prefiere utilizar un archivo, copie y pegue el siguiente ejemplo en un archivo y guárdelo como input.json.

```
{
   "Aggregators":[
      {
         "Groups":[
            {
               "Name":"ApplicationORFile",
               "Filters":[
                  {
                     "Key":"TypeName",
                     "Values":[
                        "AWS:Application",
```

```
                        "AWS:File"
                     ],
                     "Type":"Exists"
                  }
               ]
            }
         ]
      }
   ]
}
```

Ejecute el siguiente comando desde la AWS CLI.

```
aws ssm get-inventory --cli-input-json file://input.json
```

El comando devuelve información similar a la siguiente.

```
{
   "Entities":[
      {
         "Data":{
            "ApplicationORFile":{
               "Content":[
                  {
                     "notMatchingCount":"1"
                  },
                  {
                     "matchingCount":"3"
                  }
               ]
            }
         }
      }
   ]
}
```

Ejemplo 2

En este ejemplo, se calcula el número de nodos que están o que no están configurados para recopilar los tipos de inventario `AWS:Application`, `AWS:File` y `AWS:Network`.

Ejecute el siguiente comando desde la 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 prefiere utilizar un archivo, copie y pegue el siguiente ejemplo en un archivo y guárdelo como 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"
                }
            ]
        }
    ]
}
```

Ejecute el siguiente comando desde la AWS CLI.

```
aws ssm get-inventory --cli-input-json file://input.json
```

El comando devuelve información similar a la siguiente.

```
{
   "Entities":[
      {
         "Data":{
            "Application":{
               "Content":[
                  {
                     "notMatchingCount":"2"
                  },
                  {
                     "matchingCount":"2"
                  }
               ]
            },
            "File":{
               "Content":[
                  {
                     "notMatchingCount":"2"
                  },
                  {
                     "matchingCount":"2"
                  }
               ]
            },
            "Network":{
               "Content":[
                  {
```

```
                    "notMatchingCount":"3"
                },
                {
                    "matchingCount":"1"
                }
            ]
        }
    }
  }
]
}
```

Uso del inventario personalizado

Puede asignar los metadatos que desee a los nodos mediante la creación de un inventario personalizado de AWS Systems Manager Inventory. Por ejemplo, supongamos que administra un gran número de servidores en bastidores en su centro de datos y que estos servidores se han configurado como nodos administrados de Systems Manager. En la actualidad, guarda información sobre la ubicación de los bastidores de servidores en una hoja de cálculo. Con el inventario personalizado, puede especificar la ubicación de bastidor de cada nodo como metadatos en el nodo. Cuando recopila el inventario mediante Systems Manager, se recopilan los metadatos con otros metadatos de inventario. A continuación, puede transferir todos los metadatos de inventario a un bucket de Amazon S3 central mediante la sincronización de datos de recursos y consultar los datos.

> **Note**
>
> Systems Manager admite un máximo de 20 tipos de inventario personalizados por Cuenta de AWS.

Para asignar el inventario personalizado a un nodo, puede utilizar la operación PutInventory de la API de Systems Manager tal y como se describe en Asignación de metadatos de inventarios personalizados a un nodo administrado. O bien, puede crear un archivo JSON de inventario personalizado y cargarlo en el nodo. En esta sección se describe cómo crear el archivo JSON.

En el siguiente ejemplo, el archivo JSON con inventario personalizado especifica la información de bastidor en relación con un servidor local. Este ejemplo especifica un tipo de datos de inventario personalizado (`"TypeName": "Custom:RackInformation"`), con varias entradas en `Content` que describen los datos.

```
{
    "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"
      }
 }
```

También puede especificar entradas diferentes en la sección `Content`, tal y como se muestra en el siguiente ejemplo.

```
{
"SchemaVersion": "1.0",
"TypeName": "Custom:PuppetModuleInfo",
    "Content": [{
        "Name": "puppetlabs/aws",
        "Version": "1.0"
      },
      {
        "Name": "puppetlabs/dsc",
        "Version": "2.0"
      }
    ]
}
```

El esquema JSON del inventario personalizado requiere las secciones `SchemaVersion`, `TypeName` y `Content`, pero puede definir la información en dichas secciones.

```
{
    "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"
      }
```

```
}
```

El valor de encabezado `TypeName` se limita a 100 caracteres. Además, el valor `TypeName` debe empezar por la palabra en mayúscula `Custom`. Por ejemplo, `Custom:PuppetModuleInfo`. Por lo tanto, los siguientes ejemplos darían lugar a una excepción: `CUSTOM:PuppetModuleInfo`, `custom:PuppetModuleInfo`.

La sección `Content` incluye atributos y *`datos`*. Esos elementos no distinguen entre mayúsculas y minúsculas. No obstante, si define un atributo (por ejemplo: "`Vendor`": "DELL"), deberá hacer referencia a este atributo de forma coherente en los archivos de inventario personalizado. Si especifica "`Vendor`": "DELL" (utilizando una "P" mayúscula en `vendor`) en un archivo y, a continuación, especifica "`vendor`": "DELL" (con una "p" en minúscula en `vendor`) en otro archivo, el sistema devolverá un error.

> **Note**
>
> Debe guardar el archivo con una extensión `.json` y el inventario que defina debe incluir únicamente valores de cadena.

Después de crear el archivo, debe guardarlo en el nodo. La tabla siguiente muestra la ubicación en la que deben guardarse los archivos JSON del inventario personalizado en el nodo.

Sistema operativo	Ruta
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

Si desea ver un ejemplo de cómo utilizar el inventario personalizado, consulte Get Disk Utilization of Your Fleet Using EC2 Systems Manager Custom Inventory Types.

Eliminación de un inventario personalizado

Puede utilizar la operación DeleteInventory de la API para eliminar un tipo de inventario personalizado y los datos asociados a él. Para eliminar todos los datos de un tipo de inventario, tiene que llamar al comando delete-inventory mediante la AWS Command Line Interface (AWS CLI). Para eliminar un tipo de inventario personalizado, tiene que llamar al comando delete-inventory con `SchemaDeleteOption`.

> **Note**
>
> Un tipo de inventario también se denomina un esquema de inventario.

El parámetro `SchemaDeleteOption` incluye las siguientes opciones:

- DeleteSchema: esta opción elimina el tipo personalizado especificado y todos los datos asociados con él. Puede volver a crear el esquema más tarde, si lo desea.
- DisableSchema: si elige esta opción, el sistema desactivará la versión actual, eliminará todos sus datos y omitirá todos los nuevos datos si la versión es anterior o igual a la versión desactivada. Puede volver a habilitar este tipo de inventario si llama a la acción PutInventory para una versión posterior a la versión desactivada.

Para eliminar o desactivar un inventario personalizado mediante la AWS CLI

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.
2. Ejecute el siguiente comando para utilizar la opción `dry-run` con el fin de ver qué datos se eliminarán del sistema. Este comando no elimina ningún dato.

   ```
   aws ssm delete-inventory --type-name "Custom:custom_type_name" --dry-run
   ```

   El sistema devuelve información similar a la siguiente.

   ```
   {
      "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"
}
```

Para obtener información acerca del resumen de eliminaciones del inventario, consulte Explicación del resumen de eliminaciones del inventario.

3. Ejecute el siguiente comando para eliminar todos los datos de un tipo de inventario personalizado.

```
aws ssm delete-inventory --type-name "Custom:custom_type_name"
```

> **Note**
>
> El resultado de este comando no muestra el progreso de la eliminación. Por este motivo, TotalCount y Remaining Count siguen siendo iguales, ya que el sistema no ha eliminado nada todavía. Puede utilizar el comando describe-inventory-deletions para mostrar el progreso de la eliminación, tal y como se describe más adelante en este tema.

El sistema devuelve información similar a la siguiente.

```
{
   "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"
}
```

El sistema elimina todos los datos del tipo de inventario personalizado especificado del servicio Systems Manager Inventory.

4. Ejecute el siguiente comando. El comando realiza las siguientes acciones para la versión actual del tipo de inventario: desactiva la versión actual, elimina todos los datos correspondientes a ella y omite todos los datos nuevos si la versión es inferior o igual a la versión desactivada.

```
aws ssm delete-inventory --type-name "Custom:custom_type_name" --schema-delete-
option "DisableSchema"
```

El sistema devuelve información similar a la siguiente.

```
{
   "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"
}
```

Puede ver un tipo de inventario desactivado mediante el siguiente comando.

```
aws ssm get-inventory-schema --type-name Custom:custom_type_name
```

5. Ejecute el siguiente comando para eliminar un tipo de inventario.

```
aws ssm delete-inventory --type-name "Custom:custom_type_name" --schema-delete-
option "DeleteSchema"
```

El sistema elimina el esquema y todos los datos de inventario del tipo personalizado especificado.

El sistema devuelve información similar a la siguiente.

```
{
   "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"
}
```

Visualización del estado de eliminación

Puede verificar el estado de una operación de eliminación mediante el comando `describe-inventory-deletions` de la AWS CLI. Puede especificar un ID de eliminación para ver el estado de una operación de eliminación específica. Asimismo, puede omitir el ID de eliminación para ver una lista de todas las eliminaciones ejecutadas en los últimos 30 días.

1. Ejecute el siguiente comando para ver el estado de una operación de eliminación. El sistema devolvió el ID de eliminación en el resumen de eliminaciones del inventario.

```
aws ssm describe-inventory-deletions --deletion-id system_generated_deletion_ID
```

   El sistema devuelve el estado más reciente. Puede que la operación de eliminación no haya terminado todavía. El sistema devuelve información similar a la siguiente.

```
{"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 la operación de eliminación se realiza correctamente, `LastStatusMessage` indica: Deletion is successful.

```
{"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. Ejecute el siguiente comando para ver una lista de todas las eliminaciones realizadas en los últimos 30 días.

```
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"
```

Explicación del resumen de eliminaciones del inventario

Para ayudarle a comprender el contenido del resumen de eliminaciones del inventario, considere el siguiente ejemplo. Un usuario asignó el inventario Custom:RackSpace a tres nodos. Los artículos de inventario 1 y 2 utilizan la versión 1.0 del tipo personalizado ("SchemaVersion":"1.0"). El artículo de inventario 3 utiliza la versión 2.0 del tipo personalizado ("SchemaVersion":"2.0").

Inventario personalizado RackSpace 1

```
{
   "CaptureTime":"2018-02-19T10:48:55Z",
```

```
   "TypeName":"CustomType:RackSpace",
   "InstanceId":"i-1234567890",
   "SchemaVersion":"1.0"   "Content":[
      {
         content of custom type omitted
      }
   ]
}
```

Inventario personalizado RackSpace 2

```
{
   "CaptureTime":"2018-02-19T10:48:55Z",
   "TypeName":"CustomType:RackSpace",
   "InstanceId":"i-1234567891",
   "SchemaVersion":"1.0"   "Content":[
      {
         content of custom type omitted
      }
   ]
}
```

Inventario personalizado RackSpace 3

```
{
   "CaptureTime":"2018-02-19T10:48:55Z",
   "TypeName":"CustomType:RackSpace",
   "InstanceId":"i-1234567892",
   "SchemaVersion":"2.0"   "Content":[
      {
         content of custom type omitted
      }
   ]
}
```

El usuario ejecuta el siguiente comando para obtener una vista previa de los datos que se eliminarán.

```
aws ssm delete-inventory --type-name "Custom:RackSpace" --dry-run
```

El sistema devuelve información similar a la siguiente.

```
{
```

```
   "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"
}
```

El usuario ejecuta el siguiente comando para eliminar el inventario Custom:RackSpace.

> **Note**
>
> El resultado de este comando no muestra el progreso de la eliminación. Por este motivo, `TotalCount` y `RemainingCount` siguen siendo iguales, ya que el sistema no ha eliminado nada todavía. Puede utilizar el comando `describe-inventory-deletions` para mostrar el progreso de la eliminación.

```
aws ssm delete-inventory --type-name "Custom:RackSpace"
```

El sistema devuelve información similar a la siguiente.

```
{
```

```
   "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"
}
```

Visualización de acciones de eliminación de inventario en EventBridge

Puede configurar Amazon EventBridge para que cree un evento cada vez que un usuario elimine un inventario personalizado. EventBridge ofrece tres tipos de eventos para las operaciones de eliminación de inventarios personalizados:

- Acción de eliminación de una instancia: indica si el inventario personalizado de un nodo administrado específico se ha eliminado correctamente o no.
- Resumen de la acción de eliminación: muestra un resumen de la acción de eliminación.
- Advertencia de tipo de inventario personalizado desactivado: se genera un evento de advertencia si un usuario llama a la operación PutInventory de la API para una versión de tipo de inventario personalizado que se desactivó anteriormente.

A continuación, se muestran ejemplos de cada evento.

Acción de eliminación de una instancia

```
{
   "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"
   }
}
```

Resumen de la acción de eliminación

```
{
   "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"
   }
}
```

Advertencia de tipo de inventario personalizado desactivado

```
{
   "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"
   }
}
```

Utilice el siguiente procedimiento para crear una regla de EventBridge para las operaciones de eliminación de inventarios personalizados. En este procedimiento se muestra cómo crear una regla que envíe notificaciones de las operaciones de eliminación de inventarios personalizados a un tema de Amazon SNS. Antes de comenzar, compruebe que tiene un tema de Amazon SNS o cree uno nuevo. Para obtener más información, consulte Introducción en la Guía para desarrolladores de Amazon Simple Notification Service.

Para configurar EventBridge para la eliminación de operaciones de inventario

1. Abra la consola de Amazon EventBridge en https://console.aws.amazon.com/events/.
2. En el panel de navegación, seleccione Reglas.
3. Elija Creación de regla.
4. Escriba un nombre y una descripción para la regla.

   Una regla no puede tener el mismo nombre que otra regla de la misma región y del mismo bus de eventos.

5. En Bus de eventos, seleccione el bus de eventos que desea asociar a esta regla. Si desea que esta regla responda a eventos coincidentes procedentes de su propia Cuenta de AWS, seleccione default (predeterminado). Cuando un Servicio de AWS en su cuenta emite un evento, siempre va al bus de eventos predeterminado de su cuenta.
6. En Tipo de regla, elija Regla con un patrón de evento.
7. Elija Siguiente.
8. En Origen del evento, elija Eventos de AWS o eventos de socios de EventBridge.
9. En la sección Event pattern (Patrón de eventos), elija Event pattern form (Formulario de patrón de eventos).
10. Para Event source (origen de eventos), elija AWSservices (servicios).
11. En AWS service (Servicio de), elija Systems Manager.
12. Para Event type (Tipo de evento), elija Inventory (Inventario).
13. En Specific detail type(s) (Tipos de detalles específicos), elija Inventory Resource State Change (Cambio de estado de recursos de inventario).
14. Elija Siguiente.
15. En Target types (Tipos de destino), elija AWS service.
16. En Select a target (Seleccione un destino), elija SNS topic (Tema de SNS) y, a continuación, elija el tema en Topic (Tema).
17. En la sección Additional settings (Ajustes adicionales), en Configure target input (Configurar entrada de destino), verifique que Matched event (Evento coincidente) está seleccionado.
18. Elija Siguiente.
19. (Opcional) Introduzca una o varias etiquetas para la regla. Para obtener más información, consulte Etiquetado de los recursos de Amazon EventBridge en la Guía del usuario de Amazon EventBridge.
20. Elija Siguiente.
21. Revise los detalles de la regla y seleccione Creación de regla.

Visualización del seguimiento de cambios y del historial de Inventory

Puede ver el historial de AWS Systems Manager Inventory y el seguimiento de cambios de todos los nodos administrados mediante AWS Config. AWS Config proporciona una vista detallada de la configuración de los recursos de AWS en su Cuenta de AWS. Esto incluye cómo se relacionan los recursos entre sí y cómo se han configurado en el pasado, para que pueda ver cómo las

configuraciones y las relaciones cambian a lo largo del tiempo. Para ver el seguimiento de cambios y el historial de inventario, debe activar los siguientes recursos en AWS Config:

- SSM:ManagedInstanceInventory
- SSM:PatchCompliance
- SSM:AssociationCompliance
- SSM:FileData

> **Note**
>
> Tenga en cuenta los siguientes detalles importantes acerca del historial y el seguimiento de cambios de Inventory:
>
> - Si usa AWS Config para realizar un seguimiento de los cambios en el sistema, debe configurar Systems Manager Inventory para recopilar los metadatos `AWS:File` y, así, poder ver los cambios de archivos en AWS Config (`SSM:FileData`). Si no lo hace, AWS Config no realizará un seguimiento de los cambios de archivos en el sistema.
> - Cuando activa SSM:PatchCompliance y SSM:AssociationCompliance, puede ver el seguimiento de cambios y el historial de la aplicación de parches de Systems Manager Patch Manager y de la conformidad de las asociaciones de Systems Manage State Manager. Para obtener más información sobre la administración de la conformidad para estos recursos, consulte Detalles sobre el cumplimiento.

En el siguiente procedimiento, se describe cómo activar el registro del seguimiento de cambios y el historial de inventario en AWS Config mediante la AWS Command Line Interface (AWS CLI). Para obtener más información acerca de cómo elegir y configurar estos recursos en AWS Config, consulte Selección de los recursos que debe registrar AWS Config en la Guía para desarrolladores de AWS Config. Para obtener información sobre precios de AWS Config, consulte precios.

Antes de empezar

AWS Config requiere permisos de AWS Identity and Access Management (IAM) para obtener los detalles de configuración de los recursos de Systems Manager. En el siguiente procedimiento, debe especificar el nombre de recurso de Amazon (ARN) de un rol de IAM que concede permisos a AWS Config para los recursos de Systems Manager. Puede adjuntar la política administrada `AWS_ConfigRole` al rol de IAM que va a asignar a AWS Config. Para obtener más información

acerca de este rol, consulte Política administrada de AWS: AWS_Configrole en la Guía para desarrolladores de AWS Config. Para obtener información acerca de cómo crear un rol de IAM y asignar la política administrada por `AWS_ConfigRole` a ese rol, consulte Creación de un rol para delegar permisos a un servicio de Servicio de AWS en la Guía del usuario de IAM.

Para activar el registro del seguimiento de cambios y el historial de inventario en AWS Config

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.
2. Copie y pegue la siguiente muestra de JSON en un archivo de texto sencillo y guárdelo como recordingGroup.json.

```
{
   "allSupported":false,
   "includeGlobalResourceTypes":false,
   "resourceTypes":[
      "AWS::SSM::AssociationCompliance",
      "AWS::SSM::PatchCompliance",
      "AWS::SSM::ManagedInstanceInventory",
      "AWS::SSM::FileData"
   ]
}
```

3. Ejecute el siguiente comando para cargar el archivo recordingGroup.json en AWS Config.

```
aws configservice put-configuration-recorder --configuration-recorder
 name=myRecorder,roleARN=arn:aws:iam::123456789012:role/myConfigRole --recording-
group file://recordingGroup.json
```

4. Ejecute el siguiente comando para empezar a registrar el seguimiento de cambios y el historial de inventario.

```
aws configservice start-configuration-recorder --configuration-recorder-
name myRecorder
```

Después de configurar el historial y el seguimiento de cambios, puede desglosar el historial para ver un nodo administrado específico mediante el botón AWS Config en la consola de Systems Manager. Puede acceder al botón AWS Config desde las páginas Managed Instances (Instancias

administradas) o Inventory (Inventario). En función del tamaño del monitor, es posible que tenga que desplazarse a la parte derecha de la página para ver el botón.

Detención de la recopilación de datos y eliminación de datos de inventario

Si ya no desea utilizar AWS Systems Manager Inventory para ver metadatos sobre los recursos de AWS, puede detener la recopilación de datos y eliminar los datos que ya se han recopilado. Esta sección incluye la siguiente información.

Temas

- Detención de la recopilación de datos
- Eliminación de una sincronización de datos de recursos de inventario

Detención de la recopilación de datos

Cuando configura inicialmente Systems Manager para recopilar datos de inventario, el sistema crea una asociación de State Manager que define la programación y los recursos a partir de los cuales recopilar metadatos. Puede detener la recopilación de datos si elimina cualquier asociaciones de State Manager que utilice el documento `AWS-GatherSoftwareInventory`.

Para eliminar una asociación de inventario

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija State Manager.
3. Elija una asociación que utilice el documento `AWS-GatherSoftwareInventory` y, a continuación, elija Delete (Eliminar).
4. Repita el paso tres para cualquier asociación restante que utilice el documento `AWS-GatherSoftwareInventory`.

Eliminación de una sincronización de datos de recursos de inventario

Si ya no desea utilizar AWS Systems Manager Inventory para ver metadatos sobre los recursos de AWS, se recomienda que elimine las sincronizaciones de datos de recursos que se utilizan para la recopilación de datos de inventario.

Para eliminar una sincronización de datos de recursos de inventario

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Inventory.
3. Elija Resource Data Syncs (Sincronización de datos de recursos).
4. En la lista, elija un nombre.

   > **Important**
   >
   > Asegúrese de elegir la sincronización utilizada para el inventario. Systems Manager admite la sincronización de datos de recursos para varias herramientas. Si elige una sincronización incorrecta, podría interrumpir la agregación de datos para Systems Managero Explorer o Systems Manager Compliance.

5. Elija Delete (Eliminar)
6. Repita estos pasos para cualquier sincronización de datos de recursos restante que desee eliminar.
7. Elimine el bucket de Amazon Simple Storage Service (Amazon S3) en el que se almacenaron los datos. Para obtener información acerca de cómo se elimina un bucket de Amazon S3, consulte Deleting a bucket (Eliminación de un bucket).

Asignación de metadatos de inventarios personalizados a un nodo administrado

El siguiente procedimiento presenta el proceso de utilizar la operación PutInventory de la API de AWS Systems Manager para asignar los metadatos de inventarios personalizados a un nodo administrado. En este ejemplo se asigna información de ubicación de bastidores a un nodo. Para obtener más información acerca del inventario personalizado, consulte Uso del inventario personalizado.

Para asignar metadatos de inventarios personalizados a un nodo

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Ejecute el siguiente comando para asignar la información acerca de la ubicación de bastidores a un nodo.

   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. Ejecute el siguiente comando para ver las entradas de inventario personalizado para este nodo.

   ```
   aws ssm list-inventory-entries --instance-id ID --type-name "Custom:RackInfo"
   ```

   El sistema devuelve información similar a la siguiente.

   ```
   {
       "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. Ejecute el siguiente comando para ver el esquema de inventario personalizado.

   ```
   aws ssm get-inventory-schema --type-name Custom:RackInfo
   ```

   El sistema devuelve información similar a la siguiente.

   ```
   {
       "Schemas": [
   ```

```
        {
            "TypeName": "Custom:RackInfo",
            "Version": "1.0",
            "Attributes": [
                {
                    "DataType": "STRING",
                    "Name": "RackLocation"
                }
            ]
        }
    ]
}
```

Uso de la AWS CLI para configurar la recopilación de datos de inventario

En los siguientes procedimientos se presenta el proceso de configuración de AWS Systems Manager Inventory para la recopilación de metadatos desde los nodos administrados. Cuando configure la recopilación de inventario, empiece creando una asociación de Systems Manager State Manager. Systems Manager recopila los datos de inventario cuando se ejecuta la asociación. Si no crea la asociación en primer lugar e intenta invocar el complemento `aws:softwareInventory` mediante, por ejemplo, el uso de Systems Manager Run Command, el sistema regresará el siguiente error:

`The aws:softwareInventory plugin can only be invoked via ssm-associate.`

> **Note**
>
> Un nodo solo puede tener una única asociación a inventario configurada a la vez. Si configura un nodo con dos o más asociaciones a inventario, la asociación no funciona y no se recopilan los datos de inventario.

Configuración rápida de todos los nodos administrados para Inventory (CLI)

Puede configurar rápidamente todos los nodos administrados en la Cuenta de AWS y en la región actual para la recopilación de datos de inventario. Esto se conoce como creación de una asociación de inventario global. Para crear una asociación de inventario global mediante la AWS CLI, utilice la opción comodín del valor `instanceIds`, tal y como se muestra en el siguiente procedimiento:

Para configurar el inventario para todos los nodos administrados en la Cuenta de AWS y en la región actual (CLI)

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Ejecute el siguiente comando.

   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**
>
> Este comando no permite que Inventory recopile metadatos para los archivos o el registro de Windows. Para inventariar estos tipos de datos, utilice el siguiente procedimiento.

Configuración manual de Inventory en los nodos administrados (CLI)

Utilice el siguiente procedimiento para configurar manualmente AWS Systems Manager Inventory en los nodos administrados mediante ID o etiquetas de nodo.

Para configurar manualmente los nodos administrados para Inventory (CLI)

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Ejecute el siguiente comando para crear una asociación de State Manager que ejecute Systems Manager Inventory en el nodo. Reemplace cada *example resource placeholder* con su propia información. Este comando configura que el servicio se ejecute cada seis horas y que se recopilen los metadatos de configuración de la red, de Windows Update y de las aplicaciones de un nodo.

   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\": \"amzn-s3-demo-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\": \"amzn-s3-demo-bucket\",
    \"OutputS3KeyPrefix\": \"Test\" } }" ^
   --parameters "networkConfig=Enabled,windowsUpdates=Enabled,applications=Enabled"
   ```

   El sistema devuelve información similar a la siguiente.

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

Puede dirigirse a grandes grupos de nodos de destino mediante el parámetro `Targets` con etiquetas de EC2. Consulte el siguiente ejemplo.

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\": \"amzn-s3-demo-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\": \"amzn-s3-demo-bucket\", \"OutputS3KeyPrefix\": \"Test
\" } }" ^
--parameters "networkConfig=Enabled,windowsUpdates=Enabled,applications=Enabled"
```

También puede realizar un inventario de claves de archivos y del registro de Windows en un nodo de Windows Server mediante los tipos de inventario `files` y `windowsRegistry` con expresiones. Para obtener más información acerca de estos tipos de inventario, consulte Uso del inventario de archivos y del registro de 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. Ejecute el siguiente comando para ver el estado de la asociación.

```
aws ssm describe-instance-associations-status --instance-id an_instance_ID
```

El sistema devuelve información similar a la siguiente.

```
{
"InstanceAssociationStatusInfos": [
         {
            "Status": "Pending",
            "DetailedStatus": "Associated",
            "Name": "reInvent2016PolicyDocumentTest",
            "InstanceId": "i-1a2b3c4d5e6f7g",
            "AssociationId": "1a2b3c4d5e6f7g-1a2b3c-1a2b3c-1a2b3c-1a2b3c4d5e6f7g",
            "DocumentVersion": "1"
        }
]
}
```

Explicación: cómo utilizar la sincronización de datos de recursos para agregar datos de inventario

La siguiente explicación describe cómo crear una configuración de sincronización de datos de recursos para AWS Systems Manager Inventory con la AWS Command Line Interface (AWS CLI). Una sincronización de datos de recursos transfiere automáticamente los datos de inventario de todos los nodos administrados a un bucket de Amazon Simple Storage Service (Amazon S3) central. La sincronización actualiza automáticamente los datos del bucket de Amazon S3 central siempre que se detectan nuevos datos de inventario.

Esta explicación también describe cómo utilizar Amazon Athena y Amazon QuickSight para consultar y analizar los datos agregados. Para obtener información acerca de cómo crear una sincronización de datos de recursos mediante Systems Manager en la AWS Management Console, consulte Explicación: cómo utilizar la sincronización de datos de recursos para agregar datos de inventario.

Para obtener información sobre la consulta de inventario desde varias Regiones de AWS y cuentas mediante Systems Manager en la AWS Management Console, consulte Consulta de datos de Inventory de varias regiones y cuentas.

> **Note**
>
> Este tutorial incluye información sobre cómo cifrar la sincronización mediante AWS Key Management Service (AWS KMS). Inventory no recopila los datos privados, específicos del usuario ni información confidencial, por lo que el cifrado es opcional. Para obtener más información acerca de AWS KMS, consulte Guía para desarrolladores de AWS Key Management Service.

Antes de empezar

Revise o complete las siguientes tareas antes de comenzar la explicación en esta sección:

- Recopile datos de inventario de los nodos administrados. A efectos de las secciones de Amazon Athena y Amazon QuickSight en esta explicación, se recomienda que recopile los datos de Application. Para obtener más información acerca de cómo se recopilan los datos de inventario, consulte Configuración de la recopilación de inventario o Uso de la AWS CLI para configurar la recopilación de datos de inventario.
- (Opcional) Si los datos de inventario se almacenan en un bucket de Amazon Simple Storage Service (Amazon S3) que utiliza el cifrado de AWS Key Management Service (AWS KMS), también configure la cuenta de IAM y el rol de servicio de `Amazon-GlueServiceRoleForSSM` para el cifrado de AWS KMS. Si no configura su cuenta de IAM y este rol, Systems Manager muestra `Cannot load Glue tables` cuando elige la pestaña Vista detallada en la consola. Para obtener más información, consulte (Opcional) Configuración de permisos para ver datos cifrados de AWS KMS.
- (Opcional) Si desea cifrar la sincronización de datos de recursos mediante AWS KMS, cree una clave nueva que incluya la siguiente política o actualice una clave existente y agréguele esta política.

  JSON

```
{
    "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/*"
                }
            }
        }
    ]
}
```

Cómo crear una sincronización de datos de recursos para Inventory

1. Abra la consola de Amazon S3 en https://console.aws.amazon.com/s3.
2. Cree un bucket para almacenar los datos de inventario agregados. Para obtener más información, consulte Crear un bucket en la Guía del usuario de Amazon Simple Storage Service. Anote el nombre de bucket y la Región de AWS donde lo creó.
3. Después de crear el bucket, seleccione la pestaña Permisos y, a continuación, elija Política de bucket.
4. Copie y pegue la siguiente política de bucket en el editor de políticas. Reemplace amzn-s3-demo-bucket y *account-id* por el nombre del bucket de Amazon S3 que ha creado y un ID de Cuenta de AWS válido. Al agregar varias cuentas, agrega una cadena de condición adicional y un ARN para cada cuenta. Elimine los marcadores de posición adicionales del ejemplo al añadir una cuenta. Si lo desea, reemplace *bucket-prefix* por el nombre de un prefijo de Amazon S3 (subdirectorio). Si no ha creado un prefijo, quite *bucket-prefix/* del ARN de la política.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": " SSMBucketDelivery",
            "Effect": "Allow",
            "Principal": {
                "Service": "ssm.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-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:*:111122223333:resource-data-sync/*",
                        "arn:aws:ssm:*:111122223333:resource-data-sync/*",
                        "arn:aws:ssm:*:111122223333:resource-data-sync/*",
                        "arn:aws:ssm:*:111122223333:resource-data-sync/*"
                    ]
                }
            }
        }
    ]
}
```

5. (Opcional) Si desea cifrar la sincronización, debe agregar las siguientes condiciones a la política del paso anterior. Agréguelas en la sección `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"
```

A continuación se muestra un ejemplo:

```
"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. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

7. (Opcional) Si desea cifrar la sincronización, ejecute el siguiente comando para comprobar que la política del bucket esté implementando el requisito de clave de AWS KMS. Reemplace cada *example resource placeholder* con su propia información.

   Linux & macOS

   ```
   aws s3 cp ./A_file_in_the_bucket s3://amzn-s3-demo-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://amzn-s3-demo-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. Ejecute el siguiente comando para crear una configuración de sincronización de datos de recursos con el bucket de Amazon S3 que ha creado al inicio de este procedimiento. Este comando crea una sincronización de la Región de AWS en la que ha iniciado sesión.

> **Note**
>
> Si la sincronización y el bucket de Amazon S3 de destino se encuentran en regiones diferentes, es posible que esté sujeto a precios de transferencia de datos. Para obtener más información, consulte Precios de Amazon S3.

Linux & macOS

```
aws ssm create-resource-data-sync \
--sync-name a_name \
--s3-destination "BucketName=amzn-s3-demo-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=amzn-s3-demo-bucket,Prefix=prefix_name,
 if_specified,SyncFormat=JsonSerDe,Region=bucket_region"
```

Puede utilizar el parámetro `region` para especificar si debe crearse la configuración de la sincronización. En el siguiente ejemplo, los datos de inventario de la región us-west-1 se sincronizarán en el bucket de Amazon S3 en la región us-west-2.

Linux & macOS

```
aws ssm create-resource-data-sync \
    --sync-name InventoryDataWest \
    --s3-destination "BucketName=amzn-s3-demo-
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=amzn-s3-demo-
bucket,Prefix=HybridEnv,SyncFormat=JsonSerDe,Region=us-west-2" ^ --region us-
west-1
```

(Opcional) Si desea cifrar la sincronización mediante AWS KMS, ejecute el siguiente comando para crear la sincronización. Si cifra la sincronización, la clave de AWS KMS y el bucket de Amazon S3 deben estar en la misma región.

Linux & macOS

```
aws ssm create-resource-data-sync \
--sync-name sync_name \
--s3-destination "BucketName=amzn-s3-demo-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=amzn-s3-demo-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. Ejecute el siguiente comando para ver el estado de la configuración de la sincronización.

```
aws ssm list-resource-data-sync
```

Si ha creado la configuración de la sincronización en una región distinta, debe especificar el parámetro `region` tal y como se muestra en el siguiente ejemplo.

```
aws ssm list-resource-data-sync --region us-west-1
```

10. Después de haber creado correctamente la configuración de la sincronización, examine el bucket de destino en Amazon S3. Los datos de inventario deberían aparecer en unos minutos.

Cómo utilizar los datos en Amazon Athena

En la siguiente sección se describe cómo ver y consultar los datos en Amazon Athena. Antes de comenzar, le recomendamos que conozca Athena. Para obtener más información, consulte ¿Qué es Amazon Athena? y Uso de datos en la Guía del usuario de Amazon Athena.

Cómo ver y consultar los datos en Amazon Athena

1. Abra la consola de Athena en https://console.aws.amazon.com/athena/.
2. Copie y pegue la siguiente instrucción en el editor de consultas y, a continuación, elija Ejecutar consulta.

   ```
   CREATE DATABASE ssminventory
   ```

   El sistema crea una base de datos denominada ssminventory.
3. Copie y pegue la siguiente instrucción en el editor de consultas y, a continuación, elija Ejecutar consulta. Reemplace amzn-s3-demo-bucket y *bucket_prefix* por el nombre y el prefijo del destino de 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://amzn-s3-demo-bucket/bucket_prefix/AWS:Application/'
   ```
4. Copie y pegue la siguiente instrucción en el editor de consultas y, a continuación, elija Ejecutar consulta.

   ```
   MSCK REPAIR TABLE ssminventory.AWS_Application
   ```

El sistema particiona la tabla.

> **Note**
>
> Si crea sincronizaciones de datos de recursos de otras Regiones de AWS o Cuentas de AWS, deberá ejecutar este comando de nuevo para actualizar las particiones. Es posible que también deba actualizar la política del bucket de Amazon S3.

5. Para realizar una vista previa de los datos, elija el icono de visualización que aparece junto a la tabla `AWS_Application`.



6. Copie y pegue la siguiente instrucción en el editor de consultas y, a continuación, elija Ejecutar consulta.

```
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 consulta regresa un recuento de diferentes versiones de `aws-cfn-bootstrap`, que es una aplicación de AWS presente en instancias de Amazon Elastic Compute Cloud (Amazon EC2) para Linux, macOS y Windows Server.

7. Copie y pegue una por una las siguientes instrucciones en el editor de consultas, reemplace amzn-s3-demo-bucket y *`bucket-prefix`* por los datos de Amazon S3 y, a continuación, seleccione Ejecutar consulta. Estas instrucciones configuran otras tablas de inventario en 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://amzn-s3-demo-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://amzn-s3-demo-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://amzn-s3-demo-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://amzn-s3-demo-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://amzn-s3-demo-bucket/bucket-prefix/AWS:PatchSummary/'
```

```
MSCK REPAIR TABLE ssminventory.AWS_PatchSummary
```

Cómo utilizar los datos en Amazon QuickSight

En la siguiente sección se proporciona información general con enlaces para crear una visualización en Amazon QuickSight.

Cómo crear una visualización en Amazon QuickSight

1. Regístrese en Amazon QuickSight y, a continuación, inicie sesión en la consola de QuickSight.
2. Cree un conjunto de datos a partir de la tabla `AWS_Application` y de cualquier otra tabla que haya creado. Para obtener más información, consulte Creación de un conjunto de datos con los datos de Amazon Athena.
3. Combine las tablas. Por ejemplo, puede combinar la columna `instanceid` de `AWS_InstanceInformation` porque coincide con la columna `resourceid` en otras tablas de inventario. Para obtener más información acerca de la combinación de tablas, consulte Combinación de tablas.
4. Cree una visualización. Para obtener más información, consulte Trabajo con elementos visuales de Amazon QuickSight.

Solución de problemas con Systems Manager Inventory

Este tema contiene información acerca de cómo solucionar errores o problemas comunes de AWS Systems Manager Inventory. Si tiene problemas para ver los nodos en Systems Manager, consulte Solución de problemas de disponibilidad de nodos administrados.

Temas

- No se admiten varias aplicaciones de todas las asociaciones con el documento “AWS-GatherSoftwareInventory”
- El estado de ejecución del inventario nunca sale de pendiente
- El documento AWS-ListWindowsInventory no se ejecuta
- La consola no muestra Inventario con las pestañas Panel | Vista detallada | Configuración
- UnsupportedAgent

- Omitido
- Con error
- Error de cumplimiento del inventario de una instancia de Amazon EC2
- El objeto de bucket de S3 contiene datos antiguos

No se admiten varias aplicaciones de todas las asociaciones con el documento "**`AWS-GatherSoftwareInventory`**"

Un error `Multiple apply all associations with document 'AWS-GatherSoftwareInventory' are not supported` significa que una o varias Regiones de AWS donde está intentando configurar una asociación de Inventory para todos los nodos ya están configuradas con una asociación de inventario para todos los nodos. Si es necesario, puede eliminar la asociación de inventario existente para todos los nodos y, a continuación, crear una nueva. Para ver las asociaciones de inventario existentes, elija State Manager en la consola de Systems Manager y, a continuación, busque las asociaciones que utilizan el documento de SSM `AWS-GatherSoftwareInventory`. Si la asociación de inventario existente para todos los nodos se creó en varias regiones y desea crear una nueva, debe eliminar la asociación existente de cada región donde haya una.

El estado de ejecución del inventario nunca sale de pendiente

Hay dos razones por las que la recopilación de inventario nunca sale del estado `Pending`:

- No hay nodos en la Región de AWS seleccionada:

  Si crea una asociación de inventario global mediante Systems Manager Quick Setup, el estado de la asociación de inventario (documento `AWS-GatherSoftwareInventory`) muestra `Pending` si no hay nodos disponibles en la región seleccionada.

- Permisos insuficientes:

  Una asociación de inventario muestra `Pending` si uno o más nodos no tienen permiso para ejecutar Systems Manager Inventory. Compruebe que el perfil de instancias de AWS Identity and Access Management (IAM) incluye la política administrada AmazonSSMManagedInstanceCore. Para obtener información acerca de cómo agregar esta política a un perfil de instancias, consulte Configuración alternativa para permisos de instancia de EC2.

  Como mínimo, el perfil de instancias debe tener los siguientes permisos de IAM.

JSON

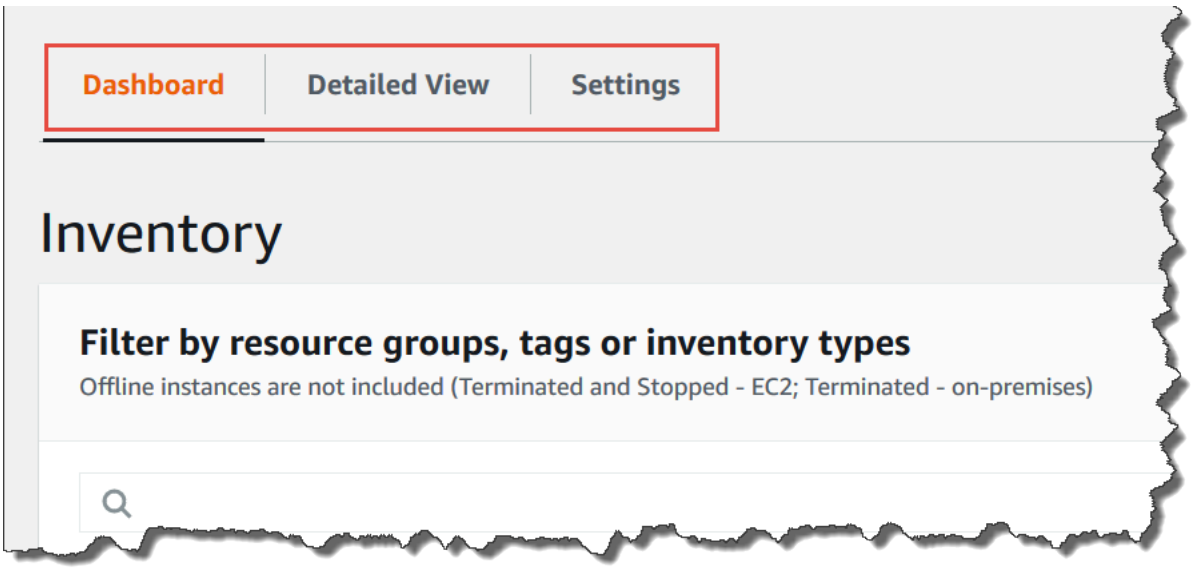
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeAssociation",
                "ssm:ListAssociations",
                "ssm:ListInstanceAssociations",
                "ssm:PutInventory",
                "ssm:PutComplianceItems",
                "ssm:UpdateAssociationStatus",
                "ssm:UpdateInstanceAssociationStatus",
                "ssm:UpdateInstanceInformation",
                "ssm:GetDocument",
                "ssm:DescribeDocument"
            ],
            "Resource": "*"
        }
    ]
}
```

El documento **`AWS-ListWindowsInventory`** no se ejecuta

El documento `AWS-ListWindowsInventory` está obsoleto. No utilice este documento para recopilar el inventario. En su lugar, utilice uno de los procesos que se describen en Configuración de la recopilación de inventario.

La consola no muestra Inventario con las pestañas Panel | Vista detallada | Configuración

La página Vista detallada de Inventory solo está disponible en la Regiones de AWS que ofrece Amazon Athena. Si las siguientes pestañas no se muestran en la página de Inventory, significa que Athena no está disponible en la región y que no puede utilizar la Vista detallada para consultar datos.



UnsupportedAgent

Si el estado detallado de una asociación de inventario es UnsupportedAgent y el valor de Estado de la asociación es Error, la versión de AWS Systems Manager SSM Agent del nodo administrado no es correcta. Para crear una asociación de inventario global (un inventario de todos los nodos de la Cuenta de AWS), debe utilizar, por ejemplo, la versión 2.0.790.0 de SSM Agent u otra versión posterior. En la página Instancias administradas, en la sección Versión del agente, puede ver la versión del agente que se ejecuta en cada nodo. Para obtener más información sobre cómo actualizar el SSM Agent en los nodos, consulte Actualización de SSM Agent mediante Run Command.

Omitido

Si el estado de la asociación de inventario de un nodo muestra Omitido, esto significa que una asociación de inventario de mayor prioridad ya se está ejecutando en ese nodo. Systems Manager sigue un orden de prioridad específico cuando se pueden aplicar varias asociaciones de inventario al mismo nodo administrado.

Orden de prioridad de asociación de inventario

Systems Manager aplica las asociaciones de inventario en el siguiente orden de prioridad:

1. Asociaciones de inventario de Quick Setup: asociaciones creadas con Quick Setup y la consola unificada. Estas asociaciones tienen nombres que comienzan por `AWS-QuickSetup-SSM-CollectInventory-` y se dirigen a todos los nodos administrados.

2. Asociaciones de inventario explícitas: asociaciones que se dirigen a nodos administrados específicos mediante:
   - ID de instancia
   - Pares clave-valor de etiqueta
   - Grupos de recursos de AWS
3. Asociaciones de inventario global: asociaciones que se dirigen a todos los nodos administrados (con `--targets "Key=InstanceIds,Values=*"`) pero que no se crearon a través de Quick Setup.

Escenarios habituales

Escenario 1: la asociación de Quick Setup anula la asociación explícita

- Tienes una asociación de inventario de Quick Setup dirigida a todas las instancias
- Puede crear una asociación manual dirigida a nodos administrados específicos por etiqueta
- Resultado: la asociación manual muestra `Skipped` con un estado detallado `OverriddenByExplicitInventoryAssociation`
- La asociación Quick Setup sigue recopilando el inventario de todas las instancias

Escenario 2: la asociación explícita anula la asociación global

- Tienes una asociación de inventario global dirigida a todas las instancias (no creada por Quick Setup)
- Crear una asociación para instancias de destino específicas
- Resultado: la asociación global muestra `Skipped` para las instancias de destino específicas
- La asociación explícita se ejecuta en las instancias de destino

Pasos de resolución

Si quiere usar su propia asociación de inventario en lugar de Quick Setup:

1. Identifique asociaciones de Quick Setup: en la consola de Systems Manager, vaya a State Manager y busque asociaciones cuyos nombres comiencen por `AWS-QuickSetup-SSM-CollectInventory-`.
2. Elimine la configuración de Quick Setup:

- Vaya a Quick Setup en la consola de Systems Manager.
- Busque su configuración de recopilación de inventario.
- Elimina la configuración de Quick Setup (esto elimina la asociación de inventario asociada).

> **Note**
>
> No es necesario eliminar manualmente la asociación creada por Quick Setup.

3. Compruebe que la asociación se ejecute: tras eliminar la configuración de Quick Setup, su asociación de inventario explícita debería empezar a ejecutarse correctamente.

Si quiere modificar el comportamiento existente:

- Para ver todas las asociaciones de inventario existentes, elija State Manager en la consola de Systems Manager y, a continuación, busque las asociaciones que utilizan el documento de SSM `AWS-GatherSoftwareInventory`.
- Recuerde que cada nodo administrado solo puede tener una asociación de inventario activa a la vez.

> **Important**
>
> - Los datos de inventario se siguen recopilando de los nodos omitidos cuando se ejecuta la asociación de inventario asignada (de mayor prioridad).
> - Las asociaciones de inventario de Quick Setup tienen prioridad sobre todos los demás tipos, incluso aquellas con destinos explícitos.
> - El mensaje de estado detallado `OverriddenByExplicitInventoryAssociation` aparece cuando una asociación es sustituida por una de mayor prioridad, independientemente del tipo de asociación.

Con error

Si el estado de la asociación de inventario de un nodo es Error, podría significar que el nodo tiene varias asociaciones de inventario asignadas. Un nodo solo puede tener una asociación de inventario asignada a la vez. Las asociaciones de inventario utilizan el documento `AWS-GatherSoftwareInventory` de AWS Systems Manager (documento de SSM). Puede ejecutar el

siguiente comando mediante la AWS Command Line Interface (AWS CLI) para ver una lista de las asociaciones de un nodo.

```
aws ssm describe-instance-associations-status --instance-id instance-ID
```

Error de cumplimiento del inventario de una instancia de Amazon EC2

Se puede producir un error de cumplimiento del inventario de una instancia de Amazon Elastic Compute Cloud (Amazon EC2) si se asignan varias asociaciones de inventario a la instancia.

Para resolver este problema, elimine una o más asociaciones de inventario asignadas a la instancia. Para obtener más información, consulte Eliminación de una asociación.

> **Note**
>
> Tenga en cuenta el siguiente comportamiento si crea varias asociaciones de inventario para un nodo administrado:
>
> - A cada nodo se le puede asignar una asociación de inventario que se dirija a todos los nodos (--targets "Key=InstanceIds,Values=*").
> - A cada nodo también se le puede asignar una asociación específica que utilice pares clave-valor de etiqueta o un grupo de recursos de AWS.
> - Si a un nodo se le asignan varias asociaciones de inventario, el estado muestra Omitido para la asociación que no se ha ejecutado. La asociación que se ha ejecutado más recientemente muestra el estado real de la asociación de inventario.
> - Si a un nodo se le asignan varias asociaciones de inventario y cada una utiliza un par clave-valor de etiqueta, esas asociaciones de inventario no se ejecutan en el nodo debido al conflicto de etiqueta. La asociación sigue ejecutándose en nodos que no tienen el conflicto clave-valor de etiqueta.

El objeto de bucket de S3 contiene datos antiguos

Los datos del objeto de bucket de Amazon S3 se actualizan cuando la asociación del inventario se realiza correctamente y se descubren datos nuevos. El objeto de bucket de Amazon S3 se actualiza para cada nodo cuando la asociación se ejecuta y se produce un error, pero los datos del objeto no se actualizan en este caso. Los datos del objeto de bucket de Amazon S3 se actualizarán

únicamente cuando la asociación se ejecute correctamente. Cuando se produzca un error en la asociación del inventario, verá datos antiguos en el objeto de bucket de Amazon S3.

AWS Systems Manager Patch Manager

Patch Manager, una herramienta de AWS Systems Manager, automatiza el proceso de aplicación de revisiones a los nodos administrados con actualizaciones relacionadas con la seguridad y otros tipos de actualizaciones.

> **Note**
>
> Systems Manager proporciona soporte para las políticas de revisiones en Quick Setup, una herramienta de AWS Systems Manager. El uso de políticas de revisiones es el método recomendado para configurar las operaciones de aplicación de revisiones. Con una configuración de política de revisiones única, puede definir la aplicación de revisiones para todas las cuentas de todas las regiones de su organización; solo para las cuentas y regiones que elija; o para un solo par de cuentas-regiones. Para obtener más información, consulte Configuraciones de políticas de revisiones en Quick Setup.

Puede utilizar Patch Manager para aplicar revisiones a los sistemas operativos y a las aplicaciones. (En Windows Server, la compatibilidad con las aplicaciones se limita a las actualizaciones de las aplicaciones publicadas por Microsoft). Puede utilizar Patch Manager para instalar Service Packs en los nodos de Windows y realizar actualizaciones de versiones secundarias en nodos de Linux. Puede aplicar revisiones a flotas de instancias de Amazon Elastic Compute Cloud (Amazon EC2), dispositivos de borde, servidores en las instalaciones y a máquinas virtuales (VM) por tipo de sistema operativo. Esto incluye las versiones compatibles de varios sistemas operativos, tal como se indica en Requisitos previos de Patch Manager. Puede analizar instancias solo para ver un informe de las revisiones que faltan, o bien puede analizar e instalar automáticamente todas las revisiones que faltan. Para comenzar a utilizar Patch Manager, abra la consola de Systems Manager. En el panel de navegación, elija Patch Manager.

AWS no prueba las revisiones antes de que estén disponibles en Patch Manager. Además, Patch Manager no admite la actualización de versiones principales de sistemas operativos, como Windows Server 2016 a Windows Server 2019 o SUSE Linux Enterprise Server (SLES) 12.0 a SLES 15.0.

Para los tipos de sistemas operativos basados en Linux que informan de un nivel de gravedad de las revisiones, Patch Manager utiliza el nivel de gravedad notificado por el editor de software para el

aviso de actualización o la revisión individual. Patch Manager no deriva los niveles de gravedad de orígenes de terceros, como el Sistema de puntuación de vulnerabilidades comunes (CVSS), ni de métricas publicadas por la Base de datos nacional de vulnerabilidades (NVD).

¿Cómo puede Patch Manager beneficiar a mi organización?

Patch Manager automatiza el proceso de aplicación de revisiones a los nodos administrados con actualizaciones relacionadas con la seguridad y otros tipos de actualizaciones. Proporciona varios beneficios clave:

- Control de revisiones centralizado: mediante políticas de revisiones, puede configurar operaciones de revisión periódicas para todas las cuentas en todas las regiones de su organización, cuentas y regiones específicas o un solo par de cuenta y región.
- Operaciones de revisión flexibles: puede analizar instancias para ver solo un informe de las revisiones que faltan, o bien puede analizar todas las revisiones que faltan e instalarlas automáticamente.
- Informes de conformidad exhaustivos: tras las operaciones de análisis, puede ver información detallada sobre qué nodos administrados no cumplen con las revisiones y qué revisiones faltan.
- Compatibilidad entre plataformas: Patch Manager admite varios sistemas operativos, incluidas varias distribuciones de Linux, macOS y Windows Server.
- Líneas de base de revisiones personalizadas: puede definir qué constituye el cumplimiento de las revisiones para su organización mediante líneas de base de revisiones personalizadas que especifican qué revisiones están aprobados para su instalación.
- Integración con otros servicios de AWS: Patch Manager se integra con AWS Organizations, AWS Security Hub, AWS CloudTrail y AWS Config para mejorar la administración y la seguridad.
- Actualizaciones deterministas: es compatible con actualizaciones deterministas mediante repositorios versionados para sistemas operativos como Amazon Linux 2023.

¿Quién debe utilizar Patch Manager?

Patch Manager está diseñado para lo siguiente:

- Administradores de TI que necesitan mantener el cumplimiento de las revisiones en toda su flota de nodos administrados.
- Gerentes de operaciones que necesitan visibilidad del estado de cumplimiento de las revisiones en toda su infraestructura.

- Arquitectos de nube que desean implementar soluciones de aplicación de revisiones automatizadas a escala.
- Ingenieros de DevOps que necesitan integrar la aplicación de revisiones en sus flujos de trabajo operativos.
- Organizaciones con implementaciones de varias cuentas o regiones que necesitan una administración de revisiones centralizada.
- Cualquier persona responsable de mantener la postura de seguridad y el estado operativo de los nodos administrados, los dispositivos de periferia, los servidores en las instalaciones y las máquinas virtuales de AWS

¿Cuáles son las características principales de Patch Manager?

Patch Manager ofrece varias características fundamentales:

- Políticas de revisión: configure las operaciones de revisión en varias Cuentas de AWS y regiones mediante una única política a través de la integración con AWS Organizations.
- Líneas de base de revisiones personalizadas: defina reglas para la aprobación automática de revisiones a los pocos días de su lanzamiento, junto con listas de revisiones aprobadas y rechazadas.
- Múltiples métodos de revisión: elija entre políticas de revisiones, ventanas de mantenimiento u operaciones bajo demanda del tipo “Aplicar revisión ahora” para satisfacer sus necesidades específicas.
- Informes de cumplimiento: genere informes detallados sobre el estado de cumplimiento de las revisiones que se pueden enviar a un bucket de Amazon S3 en formato CSV.
- Compatibilidad entre plataformas: revise tanto a sistemas operativos como a aplicaciones en Windows Server, en varias distribuciones de Linux y en macOS.
- Flexibilidad de planificación: establezca diferentes plazos para analizar e instalar revisiones mediante expresiones de frecuencia o CRON personalizadas.
- Enlaces de ciclo de vida: ejecute scripts personalizados antes y después de las operaciones de revisión mediante documentos de Systems Manager.
- Enfoque de seguridad: Patch Manager se centra de forma predeterminada en las actualizaciones relacionadas con la seguridad en lugar de instalar todas las revisiones disponibles.
- Control de velocidad: configure los umbrales de concurrencia y error para las operaciones de revisión a fin de minimizar el impacto operativo.

¿En qué consiste el cumplimiento en Patch Manager?

El punto de referencia para determinar qué constituye el cumplimiento de las revisiones para los nodos administrados de sus flotas de Systems Manager no lo definen , los proveedores de sistemas operativos (SO) ni terceros, como las empresas de consultoría de seguridad.

En su lugar, usted define lo que significa el cumplimiento de las revisiones para los nodos administrados de su organización o cuenta a partir de una línea de base de revisiones. Una línea de base de revisiones es una configuración que especifica las reglas sobre qué revisiones deben instalarse en un nodo administrado. Un nodo administrado es compatible con las revisiones cuando está actualizado con todas las revisiones que cumplen los criterios de aprobación que se especifican en la línea de base de revisiones.

Tenga en cuenta que el cumplimiento con la línea de base de revisiones no significa que un nodo administrado sea necesariamente seguro. El cumplimiento implica que las revisiones definidas en la línea de base de revisiones, que están disponibles y han sido aprobadas, se han instalado en el nodo. La seguridad general de un nodo administrado viene determinada por muchos factores ajenos al ámbito de Patch Manager. Para obtener más información, consulte Seguridad.

La línea de base de revisiones es una configuración para un tipo de sistema operativo (SO) compatible específico, como Red Hat Enterprise Linux (RHEL), macOS o Windows Server. La línea de base de revisiones puede definir las reglas de aplicación de revisiones para todas las versiones compatibles de un sistema operativo o limitarse únicamente a las que usted especifique, como RHEL 6.10, RHEL 7.8 y RHEL 9.3.

En una línea de base de revisiones, puede especificar que se aprueben para la instalación todas las revisiones con determinadas clasificaciones y niveles de gravedad. Por ejemplo, puede incluir todas las revisiones clasificadas como `Security`, pero excluir otras clasificaciones, como `Bugfix` o `Enhancement`. También puede incluir todas las revisiones con una gravedad `Critical` igual o excluir otras, como `Important` y `Moderate`.

También puede definir las revisiones de forma explícita en una línea de base de revisiones si agrega sus ID a listas de revisiones específicas para aprobarlas o rechazarlas, como `KB2736693` para Windows Server o `dbus.x86_64:1:1.12.28-1.amzn2023.0.1` para Amazon Linux 2023 (AL2023). Si lo desea, puede especificar un número determinado de días de espera para aplicar las revisiones una vez que una revisión esté disponible. En el caso de Linux y macOS, tiene la opción de especificar una lista externa de revisiones para garantizar el cumplimiento (una lista de anulación de la instalación) en lugar de las definidas en la reglas de la línea de base de revisiones.

Cuando se ejecuta una operación de aplicación de revisiones, Patch Manager compara las revisiones aplicadas actualmente a un nodo administrado con las que deberían aplicarse de acuerdo con las reglas establecidas en la línea de base de revisiones o en una lista de anulación de la instalación. Puede optar por que Patch Manager muestre solo un informe de las revisiones faltantes (una operación `Scan`) o puede optar por que Patch Manager instale automáticamente todas las revisiones que encuentre que faltan en un nodo administrado (un operación `Scan and install`).

Patch Manager proporciona líneas de base de revisiones predefinidas que puede utilizar para sus operaciones de aplicación de revisiones; sin embargo, estas configuraciones predefinidas se proporcionan como ejemplos y no como prácticas recomendadas. Le recomendamos que cree sus propias líneas de base de revisiones personalizadas para tener un mayor control sobre lo que constituye el cumplimiento de las revisiones en su flota.

Para obtener más información sobre la creación de líneas de base de revisiones personalizadas, consulte los temas siguientes:

- Líneas de base de revisiones personalizadas y predefinidas
- Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas
- Visualización de bases de referencia de parches predefinidas de AWS
- Uso de bases de referencia de parches personalizadas
- Trabajo con informes de conformidad de las revisiones

Componentes principales

Antes de empezar a trabajar con la herramienta Patch Manager, debe familiarizarse con algunos componentes y características principales de las operaciones de aplicación de revisiones de la herramienta.

líneas de base de revisiones

Patch Manager utiliza líneas de base de revisiones, las cuales incluyen reglas para la aprobación automática de revisiones a los pocos días de su lanzamiento, así como una lista de las revisiones aprobadas y rechazadas. Cuando se ejecuta una operación de aplicación de revisiones, Patch Manager compara las revisiones aplicadas actualmente a un nodo administrado con las que deberían aplicarse de acuerdo con las reglas establecidas en la línea de base de revisiones. Puede optar por que Patch Manager muestre solo un informe de las revisiones faltantes (una operación `Scan`) o puede optar por que Patch Manager instale automáticamente todas las revisiones que encuentre que faltan en un nodo administrado (un operación `Scan and install`).

Métodos de operación de aplicación de revisiones

Patch Manager ofrece actualmente cuatro métodos para ejecutar operaciones `Scan` y `Scan and install`:

- (Recomendado) Una política de revisiones configurada en Quick Setup: basada en la integración con AWS Organizations, una única política de revisiones puede definir programaciones de aplicación de revisiones y líneas de base de revisiones para toda una organización, incluidas varias Cuentas de AWS y en las Regiones de AWS que operan todas esas cuentas. Una política de revisiones también puede dirigirse únicamente a algunas unidades organizativas (OU) de una organización. Puede utilizar una única política de revisiones para analizar e instalar en diferentes programaciones. Para obtener más información, consulte Cómo configurar las revisiones para las instancias de una organización mediante una política de parches Quick Setup y Configuraciones de políticas de revisiones en Quick Setup.
- Una opción de administración de host configurada en Quick Setup: las configuraciones de administración de host también son compatibles con la integración con AWS Organizations, lo que permite ejecutar una operación de aplicación de revisiones para una organización completa. Sin embargo, esta opción se limita a analizar revisiones faltantes utilizando la línea de base de revisiones predeterminada actual y a proporcionar resultados en los informes de conformidad. Este método de operación no puede instalar revisiones. Para obtener más información, consulte Instalar la administración de host de Amazon EC2 mediante Quick Setup.
- Un periodo de mantenimiento para ejecutar una tarea **`Scan`** o **`Install`** de revisiones: se puede establecer un periodo de mantenimiento, el cual se configura en la herramienta de Systems Manager denominada Maintenance Windows, para ejecutar diferentes tipos de tareas según una programación que defina. Se puede utilizar una tarea de tipo Run Command para ejecutar tareas `Scan` o `Scan and install` en un conjunto de nodos administrados que usted elija. Cada tarea del periodo de mantenimiento puede dirigirse a los nodos administrados en un único par de Cuenta de AWS-Región de AWS. Para obtener más información, consulte Tutorial: creación de un periodo de mantenimiento para la aplicación de revisiones mediante la consola.
- Una operación de Aplicar revisión ahora bajo demanda en Patch Manager: la opción de Aplicar revisión ahora le permite omitir las configuraciones programadas cuando necesite aplicar revisiones a los nodos administrados lo antes posible. Con Patch now (Aplicar revisión ahora), se especifica si se va a ejecutar una operación de `Scan` o `Scan and install` y en qué nodos administrados se va a ejecutar la operación. También puede optar por ejecutar documentos de Systems Manager (documentos SSM) como enlaces de ciclo de vida durante la operación de revisión. Cada operación de Patch now (Aplicar revisión ahora) puede dirigirse a los nodos

administrados en un único par Cuenta de AWS-Región de AWS. Para obtener más información, consulte Aplicación de revisiones a nodos administrados bajo demanda.

Informes de conformidad

Tras una operación de `Scan`, puede utilizar la consola de Systems Manager para ver información sobre cuáles nodos administrados no cumplen con las revisiones y qué revisiones faltan en cada uno de esos nodos. También puede generar informes relativos a la conformidad de las revisiones en formato .csv que se envían a un bucket de Amazon Simple Storage Service (Amazon S3) de su elección. Puede generar informes por única vez o generarlos de manera periódica. Para un único nodo administrado, los informes incluyen detalles de todas las revisiones del nodo. Para un informe sobre todos los nodos administrados, solo se proporciona un resumen de cuántas revisiones faltan. Una vez generado un informe, puede utilizar una herramienta como Amazon QuickSight para importar y analizar los datos. Para obtener más información, consulte Trabajo con informes de conformidad de las revisiones.

> **Note**
>
> Un elemento de conformidad generado mediante el uso de una política de revisiones tiene un tipo de ejecución de `PatchPolicy`. Un elemento de conformidad no generado en una operación de política de revisiones tiene un tipo de ejecución de `Command`.

Integraciones

Patch Manager se integra con los siguientes otros Servicios de AWS:

- AWS Identity and Access Management (IAM): utilice IAM para controlar qué usuarios, grupos y roles tienen acceso a las operaciones de Patch Manager. Para obtener más información, consulte Cómo funciona AWS Systems Manager con IAM y Configuración de permisos de instancia requeridos para Systems Manager.
- AWS CloudTrail: utilice CloudTrail para registrar un historial auditable de los eventos de operación de aplicación de revisiones iniciados por usuarios, roles o grupos. Para obtener más información, consulte Registro de llamadas a la API de AWS Systems Manager con AWS CloudTrail.
- AWS Security Hub: se pueden enviar los datos de conformidad de revisiones de Patch Manager a AWS Security Hub. Security Hub ofrece una visión completa de las alertas de seguridad de alta prioridad y el estado de conformidad. También monitorea el estado de aplicación de revisiones de

la flota. Para obtener más información, consulte Integración de Patch Manager con AWS Security Hub.

- AWS Config: configure la grabación en AWS Config para ver los datos de administración de instancias de Amazon EC2 en el panel de control de Patch Manager. Para obtener más información, consulte Visualización de resúmenes del panel de revisiones.

Temas

- Configuraciones de políticas de revisiones en Quick Setup
- Requisitos previos de Patch Manager
- Cómo funcionan las operaciones de Patch Manager
- Documentos de comando de SSM para la aplicación de revisiones a nodos administrados
- líneas de base de revisiones
- Uso de Kernel Live Patching en nodos administrados de Amazon Linux 2
- Trabajo con los recursos Patch Manager y el cumplimiento mediante la consola
- Trabajar con recursos Patch Manager mediante la AWS CLI
- Tutoriales de AWS Systems Manager Patch Manager
- Solución de problemas de Patch Manager

Configuraciones de políticas de revisiones en Quick Setup

AWS recomienda el uso de políticas de revisiones para configurar las revisiones para su organización y sus Cuentas de AWS. Las políticas de revisiones se introdujeron en Patch Manager en diciembre de 2022.

Una política de revisiones es un ajuste que se configura mediante Quick Setup, una herramienta de AWS Systems Manager. Las políticas de revisiones brindan un control más amplio y centralizado sobre sus operaciones de aplicación de revisiones que el que estaba disponible con los métodos anteriores de configuración de aplicación de revisiones. Las políticas de revisiones se pueden usar con todos los sistemas operativos compatibles con Patch Manager, incluidas las versiones compatibles de Linux, macOS y Windows Server. Para obtener instrucciones sobre cómo crear una política de revisión, consulte Cómo configurar las revisiones para las instancias de una organización mediante una política de parches Quick Setup.

Características principales de las políticas de revisiones

En lugar de usar otros métodos para aplicar revisiones a sus nodos, use una política de revisiones para aprovechar estas características principales:

- Configuración única: la configuración de operaciones de aplicación de revisiones mediante un periodo de mantenimiento o una asociación State Manager puede requerir múltiples tareas en diferentes partes de la consola de Systems Manager. Mediante una política de revisiones, todas las operaciones de aplicación de revisiones se pueden configurar en un solo asistente.
- Compatibilidad con cuentas y regiones múltiples: al usar un periodo de mantenimiento, una asociación State Manager o la función Patch now (Aplicar revisión ahora) en Patch Manager, usted se ve limitado a apuntar a los nodos administrados en un único par Cuenta de AWS-Región de AWS. Si utiliza varias cuentas y regiones, sus tareas de configuración y mantenimiento pueden requerir una gran cantidad de tiempo, ya que debe realizar tareas de configuración en cada par de cuenta y región. Sin embargo, si utiliza AWS Organizations, puede configurar una política de revisiones que aplique a todos sus nodos administrados en todas las Regiones de AWS en todas sus Cuentas de AWS. O, si lo desea, la política de revisiones solo se puede aplicar a algunas unidades organizativas (OU) de las cuentas y regiones que elija. También se puede aplicar una política de revisiones a una sola cuenta local, si así lo desea.
- Soporte de instalación a nivel organizativo: la opción de configuración de administración de host existente en Quick Setup brinda soporte para un análisis diario de sus nodos administrados para la conformidad de revisiones. Sin embargo, este análisis se realiza en un momento predeterminado y solo proporciona información sobre la conformidad de las revisiones. No se realiza ninguna instalación de revisiones. Mediante una política de revisiones, puede especificar diferentes programaciones de análisis e instalación. También puede elegir la frecuencia y el tiempo de estas operaciones mediante expresiones CRON o Rate personalizadas. Por ejemplo, puede hacer un análisis para encontrar revisiones faltantes todos los días y así obtener información de conformidad que se actualiza periódicamente. Sin embargo, su programación de instalación puede ser solo una vez por semana para evitar tiempos de inactividad no deseados.
- Selección simplificada de la línea de base de revisiones: las políticas de revisiones aún incorporan líneas de base de revisiones y no hay cambios en la forma en que estas se configuran. Sin embargo, cuando crea o actualiza una política de revisiones, puede seleccionar la línea de base de AWS personalizada o administrada que desea utilizar para cada tipo de sistema operativo (SO) en una sola lista. No es necesario especificar la línea de base predeterminada para cada tipo de sistema operativo en tareas independientes.

> **Note**
>
> Cuando se ejecutan operaciones de aplicación de revisiones basadas en una política de revisiones, utilizan el documento de SSM `AWS-RunPatchBaseline`. Para obtener más información, consulte Documento de comandos SSM para aplicar revisiones: `AWS-RunPatchBaseline`.

Información relacionada

Implemente de forma centralizada las operaciones de aplicación de revisiones en toda la organización de AWS mediante Systems Manager Quick Setup (Blog sobre operaciones y migraciones en la nube de AWS)

Otras diferencias con las políticas de revisiones

Estas son algunas otras diferencias que se deben tener en cuenta al utilizar políticas de revisiones en lugar de los métodos anteriores para configurar la aplicación de revisiones:

- No se requieren grupos de revisiones: en operaciones de aplicación de revisiones anteriores, podía etiquetar varios nodos para que pertenecieran a un grupo de revisiones y luego especificar la línea de base de revisiones que se usaría para ese grupo de revisiones. Si no se definió ningún grupo de revisiones, Patch Manager aplicó revisiones en las instancias con la línea de base de revisiones predeterminada actual para el tipo de sistema operativo. Con las políticas de revisiones, ya no es necesario configurar y mantener grupos de revisiones.

  > **Note**
  >
  > La consola no admite la funcionalidad de grupos de parches para pares de cuentas y regiones que no usaban grupos de parches antes de que se publicara la compatibilidad con las políticas de parches el 22 de diciembre de 2022. La funcionalidad de grupos de parches sigue estando disponible en los pares de cuentas y regiones que empezaron a usar grupos de parches antes de esta fecha.

- Se eliminó la página "Configurar aplicación de revisiones": antes del lanzamiento de las políticas de aplicación de revisiones, podía especificar valores predeterminados para qué nodos aplicar revisiones, una programación de aplicación de revisiones y una operación de aplicación de revisiones en una página para Configurar aplicación de revisiones. Se ha eliminado esta página de Patch Manager. Estas opciones ahora se especifican en las políticas de revisiones.

- No hay compatibilidad con 'Patch now' (Aplicar revisión ahora): la capacidad de aplicar revisiones a nodos bajo demanda todavía está limitada a un solo par Cuenta de AWS-Región de AWS a la vez. Para obtener más información, consulta Aplicación de revisiones a nodos administrados bajo demanda.
- Políticas de revisiones e información de conformidad: cuando se analizan sus nodos administrados para verificar la conformidad de acuerdo con una configuración de política de aplicación de revisiones, los datos de conformidad se ponen a su disposición. Puede ver los datos y trabajar con ellos de la misma manera que con otros métodos de análisis de conformidad. Aunque puede configurar una política de revisiones para toda una organización o varias unidades organizativas, la información de conformidad se brinda individualmente para cada par Cuenta de AWS-Región de AWS. Para obtener más información, consulte Trabajo con informes de conformidad de las revisiones.
- Estado de conformidad de la asociación y políticas de revisiones: el estado de las revisiones para un nodo administrado bajo una política de revisiones de Quick Setup coincide con el estado de la ejecución de la asociación de State Manager de ese nodo. Si el estado de ejecución de la asociación es `Compliant`, el estado de las revisiones del nodo administrado también se marca como `Compliant`. Si el estado de ejecución de la asociación es `Non-Compliant`, el estado de las revisiones del nodo administrado también se marca como `Non-Compliant`.

Regiones de AWS compatibles con las políticas de revisiones

Las configuraciones de políticas de revisiones en Quick Setup se admiten actualmente en las siguientes regiones:

- Este de EE. UU. (Ohio) (us-east-2)
- Este de EE. UU. (Norte de Virginia) (us-east-1)
- EE. UU. Oeste (Norte de California) (us-west-1)
- Oeste de EE. UU. (Oregón) (us-west-2)
- Asia Pacífico (Bombay) (ap-south-1)
- Asia-Pacífico (Seúl) (ap-northeast-2)
- Asia-Pacífico (Singapur) (ap-southeast-1)
- Asia-Pacífico (Sídney) (ap-southeast-2)
- Asia-Pacífico (Tokio) (ap-northeast-1)
- Canadá (centro) (ca-central-1)
- Europa (Fráncfort) (eu-central-1)

- Europa (Irlanda) (eu-west-1)
- Europa (Londres) (eu-west-2)
- UE (París) (eu-west-3)
- Europa (Estocolmo) (eu-north-1)
- América del Sur (São Paulo) (sa-east-1)

Requisitos previos de Patch Manager

Asegúrese de que haya cumplido los requisitos previos necesarios antes de usar Patch Manager, una herramienta de AWS Systems Manager.

Temas

- Versión de SSM Agent
- Versión de Python
- Conectividad con la fuente de revisiones
- Acceso al punto de enlace de S3
- Permisos para instalar revisiones localmente
- Sistemas operativos admitidos por Patch Manager

Versión de SSM Agent

La versión 2.0.834.0 o una versión posterior de SSM Agent debe ejecutarse en los nodos administrados que desee administrar con Patch Manager.

> **Note**
>
> Cada vez que se agregan herramientas nuevas a Systems Manager o se actualizan las herramientas existentes, se lanza una versión actualizada de SSM Agent. No utilizar la versión más reciente del agente puede impedir que el nodo administrado utilice diversas herramientas y características de Systems Manager. Por este motivo, se recomienda automatizar el proceso de mantener SSM Agent actualizado en los equipos. Para obtener más información, consulta Automatización de las actualizaciones de SSM Agent. Suscríbase a la página SSM Agent Release Notes en GitHub para recibir notificaciones sobre las actualizaciones de SSM Agent.

Versión de Python

En la actualidad, para macOS y la mayoría de los sistemas operativos (SO) Linux, Patch Manager es compatible con las versiones 2.6 a 3.10 de Python. Los sistemas operativos de AlmaLinux, Debian Server, Raspberry Pi OS y Ubuntu Server requieren una versión compatible de Python 3 (3.0 a 3.10).

Conectividad con la fuente de revisiones

Si los nodos administrados no disponen de una conexión directa a Internet y utiliza una Amazon Virtual Private Cloud (Amazon VPC) con un punto de conexión de VPC, debe asegurarse de que los nodos tengan acceso a los repositorios (repos) de revisiones de origen. En los nodos de Linux, las actualizaciones de revisiones suelen descargarse de los repositorios remotos configurados en el nodo. Por lo tanto, el nodo debe poder conectarse a los repositorios para que puedan aplicarse las revisiones. Para obtener más información, consulte Cómo se seleccionan las revisiones de seguridad.

CentOS y CentOS Stream: habilitar la marca **`EnableNonSecurity`**

Los nodos administrados de CentOS 6 y 7 utilizan Yum como administrador de paquetes. Los nodos de CentOS 8 y CentOS Stream utilizan DNF como administrador de paquetes. Ambos administradores de paquetes utilizan el concepto de un aviso de actualización. Un aviso de actualización es simplemente una colección de paquetes que solucionan un problema determinado.

Sin embargo, los repositorios predeterminados de CentOS y CentOS Stream no se configuran con un aviso de actualización. Esto significa que Patch Manager no detecta paquetes en repositorios de CentOS y CentOS Stream predeterminados. Para permitir que Patch Manager procese paquetes no incluidos en avisos de actualización, debe activar la marca `EnableNonSecurity` en las reglas de base de referencia de los parches.

Windows Server: garantiza la conectividad al catálogo de Windows Update o a Windows Server Update Services (WSUS)

Los nodos administrados de Windows Server deben poder conectarse al catálogo de Windows Update o a Windows Server Update Services (WSUS). Confirme que los nodos cuentan con conectividad al catálogo de Microsoft Update a través de una puerta de enlace de Internet, una puerta de enlace NAT o una instancia NAT. Si utiliza WSUS, confirme que el nodo dispone de conectividad con el servidor WSUS de su entorno. Para obtener más información, consulte Asunto: el nodo administrado no tiene acceso al catálogo de Windows Update ni a WSUS.

Acceso al punto de enlace de S3

Tanto si los nodos administrados operan en una red privada como en una pública, sin tener acceso a los buckets requeridos de Amazon Simple Storage Service (Amazon S3) administrados por AWS, las operaciones de aplicación de revisiones pueden producir errores. Para obtener información acerca de los buckets de S3 a los que deben tener acceso los nodos administrados, consulte Comunicaciones de SSM Agent con buckets de S3 administrados de AWS y Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager.

Permisos para instalar revisiones localmente

En los sistemas operativos Windows Server y Linux, Patch Manager asume las cuentas de administrador y usuario raíz, respectivamente, para instalar las revisiones.

Sin embargo, en macOS, para Brew y Brew Cask, Homebrew no admite que sus comandos se ejecuten con la cuenta del usuario raíz. Como resultado, Patch Manager consulta y ejecuta los comandos de Homebrew como propietario del directorio de Homebrew o como usuario válido perteneciente al grupo de propietarios de ese directorio. Por lo tanto, para instalar las revisiones, el propietario del directorio `homebrew` también necesita permisos de propietario recursivos para el directorio `/usr/local`.

> **Tip**
>
> El siguiente comando proporciona este permiso al usuario especificado:
>
> ```
> sudo chown -R $USER:admin /usr/local
> ```

Sistemas operativos admitidos por Patch Manager

La herramienta Patch Manager no es compatible con las mismas versiones de sistemas operativos que sí lo son con las demás herramienta de Systems Manager. Por ejemplo, Patch Manager no es compatible con CentOS 6.3 o Raspberry Pi OS 8 (Jessie). (Para ver una lista completa de los sistemas operativos admitidos por Systems Manager, consulte Sistemas operativos compatibles con Systems Manager). Por lo tanto, asegúrese de que los nodos administrados que desea utilizar con Patch Manager ejecutan uno de los sistemas operativos que se muestran en la siguiente tabla.

Note

Patch Manager utiliza los repositorios de revisiones que estén configurados en un nodo administrado, como el Catálogo de Windows Update y Windows Server Update Services para Windows, para recuperar las revisiones disponibles que se deben instalar. Por lo tanto, para versiones del sistema operativo en el final de su vida útil (EOL), si no hay nuevas actualizaciones disponibles, es posible que Patch Manager no pueda informar sobre las nuevas actualizaciones. Esto puede deberse a que el responsable de la distribución de Linux, Microsoft o Apple no han publicado nuevas actualizaciones, o a que el nodo administrado no dispone de la licencia adecuada para acceder a las nuevas actualizaciones. Patch Manager informa del estado de conformidad respecto de las revisiones disponibles en el nodo administrado. Por lo tanto, si en una instancia se está ejecutando un sistema operativo EOL y no hay actualizaciones disponibles, es posible que Patch Manager indique que el nodo es conforme, dependiendo de las líneas de base de revisiones configuradas para la operación de revisión.

Sistema operativo	Detalles
Linux	• AlmaLinux 8.x, 9.x • Amazon Linux 2012.03 a 2018.03 • Amazon Linux 2 versión 2.0 y todas las versiones posteriores • Amazon Linux 2022 • Amazon Linux 2023 • CentOS 6.5–7.9, 8.x • CentOS Stream 8, 9 • Debian Server 8.x, 9.x, 10.x, 11.x y 12.x • Oracle Linux 7.5–8.x, 9.x • Raspberry Pi OS (antes: Raspbian) 9 (Stretch) • Red Hat Enterprise Linux (RHEL) 6.5–8.x, 9.x, 10.x • Rocky Linux 8.x, 9.x

Sistema operativo	Detalles
	• SUSE Linux Enterprise Server (SLES) 12.0 y versiones 12.x posteriores; 15.x • Ubuntu Server 14.04 LTS, 16.04 LTS, 18.04 LTS, 20.04 LTS, 20.10 STR, 22.04 LTS, 23.04, 23.10, 24.04, 24.10, y 25.04

Sistema operativo	Detalles
macOS	macOS solo es compatible con las instancias de Amazon EC2. 11.3.1; 11.4-11.7 (Big Sur) 12.0-12.7 (Monterey) 13.0-13.7 (Ventura) 14.x (Sonoma) 15.x (Sequoia) Actualizaciones del sistema operativo macOS Patch Manager no es compatible con actualizaciones ni mejoras del sistema operativo (SO) para macOS, tales como pasar de la versión 12.x a la 13.x o de la 13.1 a la 13.2. Para realizar actualizaciones de la versión del SO macOS, se recomienda utilizar los mecanismos de actualización del SO integrados de Apple. Para obtener más información, consulte Device Management en el sitio web Apple Developer Documentation. Compatibilidad con Homebrew Patch Manager requiere que Homebrew, el sistema de administración de paquetes de software de código abierto, esté instalado en cualquiera de las siguientes ubicaciones de instalación predeterminadas: • `/usr/local/bin` • `/opt/homebrew/bin`

Sistema operativo	Detalles
	Las operaciones de revisión que utilizan Patch Manager no funcionan correctamente cuando Homebrew no está instalado. Homebrew ha dejado de ofrecer soporte para macOS 10.14.x (Mojave) y 10.15.x (Catalina). Como resultado, no se admiten las operacion es de revisión con Patch Manager en estas versiones. Compatibilidad de regiones macOS no se admite en todas las Regiones de AWS. Para obtener más información sobre la compatibilidad de Amazon EC2 con macOS, consulte Instancias de Mac de Amazon EC2 en la Guía del usuario de Amazon EC2. Dispositivos periféricos de macOS No se admite SSM Agent para los dispositi vos de núcleo de AWS IoT Greengrass en macOS. No se puede utilizar Patch Manager para aplicar revisiones a dispositivos de borde de macOS.

Sistema operativo	Detalles
Windows	De Windows Server 2008 a Windows Server 2025, incluidas las versiones R2. **Note** No se admite SSM Agent para los dispositivos de núcleo AWS IoT Greengrass en Windows 10. No se puede utilizar Patch Manager para aplicar revisiones a dispositivos de borde de Windows 10. Soporte de Windows Server 2008 A partir del 14 de enero de 2020, Windows Server 2008 ya no será compatible para obtener actualizaciones de características o de seguridad de Microsoft. Amazon Machine Images heredadas (AMIs) para Windows Server 2008 y 2008 R2 aún incluyen la versión 2 de SSM Agent preinstalada, pero Systems Manager ya no admite oficialmente las versiones 2008 ni actualiza el agente para estas versiones de Windows Server. Además, es posible que la versión 3 de SSM Agent no sea compatible con todas las operacion es en Windows Server 2008 y 2008 R2. La versión final oficialmente admitida de SSM Agent para las versiones Windows Server 2008 es 2.3.1644.0. Windows Server 2012 y soporte de R2 de 2012 La compatibilidad con Windows Server 2012 y 2012 R2 finalizó el 10 de octubre de 2023. Para

Sistema operativo	Detalles
	utilizar Patch Manager con estas versiones, se recomienda utilizar también las actualizaciones de seguridad ampliada (ESU) de Microsoft . Para obtener más información, consulte Finaliza el soporte para Windows Server 2012 y 2012 R2 en el sitio web de Microsoft.

Cómo funcionan las operaciones de Patch Manager

Esta sección proporciona detalles técnicos que explican cómo Patch Manager, una herramienta de AWS Systems Manager, determina cuáles son las revisiones que deben instalarse y el modo en que lo hace en cada sistema operativo compatible. En el caso de los sistemas operativos Linux, también proporciona información sobre cómo especificar un repositorio de origen en una base de referencia de revisiones personalizada para revisiones distintas de las predeterminadas configuradas en un nodo administrado. En esta sección se proporciona también información detallada acerca de cómo funcionan las reglas de líneas de base de revisiones en diversas distribuciones del sistema operativo Linux.

> **Note**
>
> La información en los siguientes temas se aplica independientemente del método o el tipo de configuración que utilice para las operaciones de aplicación de revisiones:
>
> - Una política de revisiones configurada en Quick Setup
> - Una opción de administración de host configurada en Quick Setup
> - Una ventana de mantenimiento para ejecutar una revisión `Scan` o una tarea `Install`
> - Una operación Patch Now (Aplicar revisión ahora) bajo demanda

Temas

- Cómo se calculan las fechas de lanzamiento y actualización de los paquetes
- Cómo se seleccionan las revisiones de seguridad
- Cómo especificar un repositorio de origen de parches alternativo (Linux)
- Cómo se instalan las revisiones

- Funcionamiento de las reglas de bases de referencia de parches en los sistemas basados en Linux
- Diferencias en la operación de revisiones entre Linux y Windows Server

Cómo se calculan las fechas de lanzamiento y actualización de los paquetes

> **Important**
>
> La información de esta página es válida para los sistemas operativos (SO) Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 y Amazon Linux 2023 para las instancias de Amazon Elastic Compute Cloud (Amazon EC2). Amazon Web Services crea y mantiene los paquetes para estos tipos de sistemas operativos. La forma en que los fabricantes de otros sistemas operativos administran sus paquetes y repositorios afecta a la forma en que se calculan sus fechas de lanzamiento y actualización. Para sistemas operativos distintos de Amazon Linux, Amazon Linux 2, Amazon Linux 2022 y Amazon Linux 2023, como Red Hat Enterprise Linux (RHEL) y SUSE Linux Enterprise Server (SLES), consulte la documentación del fabricante para obtener información sobre cómo se actualizan y mantienen sus paquetes.

En la configuración de las líneas de base de revisiones personalizadas que cree, para la mayoría de los tipos de sistemas operativos, puede especificar que la instalación de las revisiones se apruebe automáticamente después de un número de días determinado. AWS proporciona varias líneas de base de revisiones predefinidas que incluyen fechas de aprobación automática de 7 días.

Un tiempo de espera hasta la aprobación automática es el número de días que se debe esperar después de que se haya lanzado la revisión y antes de que se apruebe automáticamente para aplicarla. Por ejemplo, se crea una regla mediante la clasificación de `CriticalUpdates` y se configura para establecer un tiempo de espera hasta la aprobación automática de 7 días. Como resultado, una nueva revisión crítica con una fecha de lanzamiento o una fecha de última actualización datada del 7 de julio se aprueba automáticamente el 14 de julio.

Para evitar resultados imprevistos en los retrasos de aprobación automática para Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 y Amazon Linux 2023, es importante saber cómo se calculan las fechas de lanzamiento y las de actualización.

En la mayoría de los casos, el tiempo de espera hasta la aprobación automática para instalar las revisiones se calcula a partir de un valor de `Updated Date` en `updateinfo.xml`, no un valor de `Release Date`. Los siguientes son detalles importantes sobre estos cálculos de fechas:

- La `Release Date` es la fecha en que se lanza un aviso. Esto no significa que el paquete esté necesariamente disponible en los repositorios asociados.
- La `Update Date` es la fecha más reciente en que se actualizó el aviso. La actualización de un aviso puede representar algo tan pequeño como la actualización de un texto o descripción. Esto no significa que el paquete se lanzó ese día o que esté necesariamente disponible en los repositorios asociados.

  Esto significa que un paquete puede tener un valor de `Update Date` del 7 de julio, pero no podrá instalarse hasta, por ejemplo, el 13 de julio. Supongamos que, en este caso, una línea de base de revisiones que especifica un tiempo de espera de 7 días hasta la aprobación automática se ejecuta en una operación `Install` el 14 de julio. Dado que el valor de `Update Date` es 7 días antes de la fecha de ejecución, las revisiones y actualizaciones del paquete se instalan el 14 de julio. La instalación se realiza a pesar de que solo ha pasado 1 día desde que el paquete se puede instalar.
- Un paquete que contenga revisiones del sistema operativo o aplicación se puede actualizar más de una vez después del lanzamiento inicial.
- Se puede lanzar un paquete en los repositorios administrados por AWS, y más tarde revertirse si se identifican problemas relacionados con él.

En algunas operaciones de aplicación de revisiones, es posible que estos factores no sean importantes. Por ejemplo, si una línea de base de revisiones está configurada para instalar una revisión con valores de gravedad de `Low` y `Medium`, además de una clasificación de `Recommended`, cualquier tiempo de espera hasta la aprobación automática podría tener poco impacto en las operaciones.

Sin embargo, en los casos de revisiones críticas o de gravedad alta cuya programación sea más importante, puede ser conveniente tener un mayor control sobre cuándo deben instalarse. El método recomendado para hacerlo es utilizar repositorios de origen de revisiones alternativos en lugar de los predeterminados para las operaciones de aplicación de revisiones en un nodo administrado.

Puede especificar repositorios de origen de parches alternativos al crear una base de referencia de parches personalizada. En cada base de referencia de parches personalizada, es posible especificar configuraciones de origen de parches para un máximo de 20 versiones de un sistema operativo Linux compatible. Para obtener más información, consulte Cómo especificar un repositorio de origen de parches alternativo (Linux).

Cómo se seleccionan las revisiones de seguridad

Patch Manager, una herramienta de AWS Systems Manager, se centra principalmente en instalar actualizaciones relacionadas con la seguridad de los sistemas operativos en los nodos administrados. De forma predeterminada, Patch Manager no instala todos los parches disponibles, sino un conjunto de parches más reducido centrado en la seguridad.

Patch Manager no reemplaza de forma predeterminada un paquete que se haya marcado como obsoleto en un repositorio de paquetes con ningún paquete de reemplazo con un nombre diferente, a menos que este reemplazo sea necesario para la instalación de otra actualización de paquete. En cambio, para los comandos que actualizan un paquete, Patch Manager solo informa las actualizaciones que faltan para el paquete que está instalado, pero obsoleto y las instala. Esto se debe a que la sustitución de un paquete obsoleto suele requerir la desinstalación de un paquete existente y la instalación de uno de reemplazo. Reemplazar el paquete obsoleto podría introducir cambios importantes o funcionalidades adicionales que no se habían previsto.

Este comportamiento es coherente con el comando `update-minimal` de YUM y DNF, que se centra en las actualizaciones de seguridad más que en las mejoras de características. Para obtener más información, consulte the section called “Cómo se instalan las revisiones”.

Para los tipos de sistemas operativos basados en Linux que informan de un nivel de gravedad de las revisiones, Patch Manager utiliza el nivel de gravedad notificado por el editor de software para el aviso de actualización o la revisión individual. Patch Manager no deriva los niveles de gravedad de orígenes de terceros, como el Sistema de puntuación de vulnerabilidades comunes (CVSS), ni de las métricas publicadas por la Base de datos nacional de vulnerabilidades (NVD).

> **Note**
>
> En todos los sistemas basados en Linux compatibles con Patch Manager, es posible elegir un repositorio de origen distinto configurado para el nodo administrado, normalmente para instalar actualizaciones no relacionadas con la seguridad. Para obtener más información, consulta Cómo especificar un repositorio de origen de parches alternativo (Linux).

En el resto de esta sección se explica cómo Patch Manager selecciona los parches de seguridad para los distintos sistemas operativos compatibles.

Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022, and Amazon Linux 2023

La administración de los repositorios preconfigurados en Amazon Linux 1 y Amazon Linux 2 difiere con la administración en Amazon Linux 2022 y Amazon Linux 2023.

En Amazon Linux 1 y Amazon Linux 2, el servicio de línea de base de revisiones de Systems Manager utiliza repositorios preconfigurados en los nodos administrados. Normalmente, hay dos repositorios preconfigurados (repos) en un nodo:

En Amazon Linux 1

- ID del repositorio: `amzn-main/latest`

  Nombre del repositorio: `amzn-main-Base`
- ID del repositorio: `amzn-updates/latest`

  Nombre del repositorio: `amzn-updates-Base`

En Amazon Linux 2

- ID del repositorio: `amzn2-core/2/`*`architecture`*

  Nombre del repositorio: `Amazon Linux 2 core repository`
- ID del repositorio: `amzn2extra-docker/2/`*`architecture`*

  Nombre del repositorio: `Amazon Extras repo for docker`

> **Note**
>
> La *`arquitectura`* puede ser x86_64 o aarch64.

Cuando crea una instancia de Amazon Linux 2023 (AL2023), esta contiene las actualizaciones que estaban disponibles en la versión de AL2023 y la AMI específica que seleccionó. Su instancia AL2023 no recibe de manera automática las actualizaciones de seguridad críticas e importantes adicionales en el momento del lanzamiento. En cambio, con la característica de actualizaciones deterministas a través de repositorios versionados compatibles con AL2023, que está activada de forma predeterminada, puede aplicar las actualizaciones según una programación que se

adapte a sus necesidades específicas. Para obtener información, consulte Deterministic upgrades through versioned repositories en la Guía del usuario de Amazon Linux 2023.

En Amazon Linux 2022, los repositorios preconfigurados están vinculados a las versiones bloqueadas de las actualizaciones de paquetes. Cuando se publiquen nuevas Amazon Machine Images (AMIs) para Amazon Linux 2022, se bloquearán a una versión específica. Para las actualizaciones de revisiones, Patch Manager recupera la última versión bloqueada del repositorio de actualizaciones de revisiones y, a continuación, actualiza los paquetes en el nodo administrado en función del contenido de esa versión bloqueada.

En AL2023, el repositorio preconfigurado es el siguiente:

- ID del repositorio: `amazonlinux`

  Nombre del repositorio: repositorio de Amazon Linux 2023

En Amazon Linux 2022 (versión preliminar), los repositorios preconfigurados están vinculados a las versiones bloqueadas de las actualizaciones de paquetes. Cuando se publiquen nuevas Amazon Machine Images (AMIs) para Amazon Linux 2022, se bloquearán a una versión específica. Para las actualizaciones de revisiones, Patch Manager recupera la última versión bloqueada del repositorio de actualizaciones de revisiones y, a continuación, actualiza los paquetes en el nodo administrado en función del contenido de esa versión bloqueada.

En Amazon Linux 2022, el repositorio preconfigurado es el siguiente:

- ID del repositorio: `amazonlinux`

  Nombre del repositorio: repositorio de Amazon Linux 2022

> **Note**
>
> Todas las actualizaciones se descargan desde los repositorios remotos configurados en el nodo administrado. Por lo tanto, el nodo debe tener acceso saliente a Internet para conectarse a los repositorios y que puedan implementarse las revisiones.

Los nodos administrados de Amazon Linux 1 y Amazon Linux 2 utilizan el administrador de paquetes Yum. Amazon Linux 2022 y Amazon Linux 2023 utilizan DNF como administrador de paquetes.

Ambos administradores de paquetes utilizan el concepto de un aviso de actualización como un archivo llamado `updateinfo.xml`. Un aviso de actualización es simplemente una colección de paquetes que solucionan un problema determinado. Patch Manager considera todos los paquetes que se encuentran en un aviso de actualización como paquetes de seguridad. A los paquetes individuales no se les asignan clasificaciones ni niveles de gravedad. Por este motivo, Patch Manager asigna los atributos de un aviso de actualización a los paquetes relacionados.

> **Note**
>
> Si selecciona la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad en la página Crear una línea de base de revisiones, los paquetes que no estén clasificados en un archivo `updateinfo.xml` (o un paquete que contenga un archivo sin los valores de Clasificación, Gravedad y Fecha con el formato correcto) se podrán incluir en la lista de revisiones filtrada con anterioridad. Sin embargo, para poder aplicar un parche, este debe seguir cumpliendo las reglas de base de referencia de parches especificadas por el usuario.
>
> Para obtener más información sobre la opción Incluir actualizaciones no relacionadas con la seguridad, consulte Cómo se instalan las revisiones y Funcionamiento de las reglas de bases de referencia de parches en los sistemas basados en Linux.

CentOS and CentOS Stream

En CentOS y CentOS Stream, el servicio de la línea de base de revisiones de Systems Manager utiliza repositorios (repos) preconfigurados en el nodo administrado. En la siguiente lista, se muestran ejemplos para una Amazon Machine Image (AMI) de CentOS 8.2 ficticia:

- ID del repositorio: `example-centos-8.2-base`

  Nombre del repositorio: `Example CentOS-8.2 - Base`
- ID del repositorio: `example-centos-8.2-extras`

  Nombre del repositorio: `Example CentOS-8.2 - Extras`
- ID del repositorio: `example-centos-8.2-updates`

  Nombre del repositorio: `Example CentOS-8.2 - Updates`
- ID del repositorio: `example-centos-8.x-examplerepo`

  Nombre del repositorio: `Example CentOS-8.x – Example Repo Packages`

> **Note**
>
> Todas las actualizaciones se descargan desde los repositorios remotos configurados en el nodo administrado. Por lo tanto, el nodo debe tener acceso saliente a Internet para conectarse a los repositorios y que puedan implementarse las revisiones.

Los nodos administrados de CentOS 6 y 7 utilizan Yum como administrador de paquetes. Los nodos de CentOS 8 y CentOS Stream utilizan DNF como administrador de paquetes. Ambos administradores de paquetes utilizan el concepto de un aviso de actualización. Un aviso de actualización es simplemente una colección de paquetes que solucionan un problema determinado.

Sin embargo, los repositorios predeterminados de CentOS y CentOS Stream no se configuran con un aviso de actualización. Esto significa que Patch Manager no detecta paquetes en repositorios de CentOS y CentOS Stream predeterminados. Para permitir que Patch Manager procese paquetes no incluidos en avisos de actualización, debe activar la marca `EnableNonSecurity` en las reglas de base de referencia de los parches.

> **Note**
>
> Los avisos de actualización de CentOS y CentOS Stream son compatibles. Los repositorios con avisos de actualización se pueden descargar tras el lanzamiento.

Servidor Debian and Raspberry Pi OS

En Debian Server y en Raspberry Pi OS (anteriormente Raspbian), el servicio de bases de referencia de revisiones de Systems Manager utiliza repositorios (repos) preconfigurados en la instancia. Estos repositorios preconfigurados se utilizan para obtener una lista actualizada de actualizaciones de paquetes disponibles. Por este motivo, Systems Manager realiza el equivalente a un comando `sudo apt-get update`.

A continuación, los paquetes se filtran desde los repositorios `debian-security` *`codename`*. Esto significa que en cada versión de Debian Server, Patch Manager solo identifica las actualizaciones que son parte del repositorio asociado para esta versión, como se muestra a continuación:

- Debian Server 8: `debian-security jessie`

- Debian Server 9: `debian-security stretch`
- Debian Server 10: `debian-security buster`
- Debian Server 11: `debian-security bullseye`
- Debian Server 12: `debian-security bookworm`

> **Note**
>
> Solo en Debian Server 8: debido a que algunos nodos administrados de Debian Server 8.* hacen referencia a un repositorio de paquetes obsoleto (`jessie-backports`), Patch Manager lleva a cabo pasos adicionales para garantizar que las operaciones de aplicación de revisiones se efectúen correctamente. Para obtener más información, consulte Cómo se instalan las revisiones.

Oracle Linux

En Oracle Linux, el servicio de base de referencia de revisiones de Systems Manager utiliza repositorios (repos) preconfigurados en el nodo administrado. Normalmente, hay dos repositorios preconfigurados en un nodo.

Oracle Linux 7:

- ID del repositorio: `ol7_UEKR5/x86_64`

  Nombre del repositorio: `Latest Unbreakable Enterprise Kernel Release 5 for Oracle Linux 7Server (x86_64)`
- ID del repositorio: `ol7_latest/x86_64`

  Nombre del repositorio: `Oracle Linux 7Server Latest (x86_64)`

Oracle Linux 8:

- ID del repositorio: `ol8_baseos_latest`

  Nombre del repositorio: `Oracle Linux 8 BaseOS Latest (x86_64)`
- ID del repositorio: `ol8_appstream`

  Nombre del repositorio: `Oracle Linux 8 Application Stream (x86_64)`

- ID del repositorio: `ol8_UEKR6`

  Nombre del repositorio: `Latest Unbreakable Enterprise Kernel Release 6 for Oracle Linux 8 (x86_64)`

Oracle Linux 9:

- ID del repositorio: `ol9_baseos_latest`

  Nombre del repositorio: `Oracle Linux 9 BaseOS Latest (x86_64)`
- ID del repositorio: `ol9_appstream`

  Nombre del repositorio: `Oracle Linux 9 Application Stream Packages(x86_64)`
- ID del repositorio: `ol9_UEKR7`

  Nombre del repositorio: `Oracle Linux UEK Release 7 (x86_64)`

> **Note**
>
> Todas las actualizaciones se descargan desde los repositorios remotos configurados en el nodo administrado. Por lo tanto, el nodo debe tener acceso saliente a Internet para conectarse a los repositorios y que puedan implementarse las revisiones.

Los nodos administrados de Oracle Linux utilizan Yum como administrador de paquetes y Yum utiliza el concepto de aviso de actualización como un archivo denominado `updateinfo.xml`. Un aviso de actualización es simplemente una colección de paquetes que solucionan un problema determinado. A los paquetes individuales no se les asignan clasificaciones ni niveles de gravedad. Por este motivo, Patch Manager asigna los atributos de un aviso de actualización a los paquetes relacionados e instala los paquetes en función de los filtros de clasificación especificados en la base de referencia de parches.

> **Note**
>
> Si selecciona la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad en la página Crear una línea de base de revisiones, los paquetes que no estén clasificados en un archivo `updateinfo.xml` (o un paquete que contenga un archivo sin los valores de Clasificación, Gravedad y Fecha con el formato correcto) se podrán

incluir en la lista de revisiones filtrada con anterioridad. Sin embargo, para poder aplicar un parche, este debe seguir cumpliendo las reglas de base de referencia de parches especificadas por el usuario.

AlmaLinux, RHEL, and Rocky Linux

En AlmaLinux, Red Hat Enterprise Linux y Rocky Linux, el servicio de línea de base de revisiones de Systems Manager utiliza repositorios (repos) preconfigurados en el nodo administrado. Normalmente, hay tres repositorios preconfigurados en un nodo.

Todas las actualizaciones se descargan desde los repositorios remotos configurados en el nodo administrado. Por lo tanto, el nodo debe tener acceso saliente a Internet para conectarse a los repositorios y que puedan implementarse las revisiones.

> **Note**
>
> Si selecciona la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad en la página Crear una línea de base de revisiones, los paquetes que no estén clasificados en un archivo `updateinfo.xml` (o un paquete que contenga un archivo sin los valores de Clasificación, Gravedad y Fecha con el formato correcto) se podrán incluir en la lista de revisiones filtrada con anterioridad. Sin embargo, para poder aplicar un parche, este debe seguir cumpliendo las reglas de base de referencia de parches especificadas por el usuario.

Los nodos administrados de Red Hat Enterprise Linux 7 utilizan YUM como administrador de paquetes. Los nodos administrados de AlmaLinux, Red Hat Enterprise Linux 8 y Rocky Linux utilizan DNF como gestor de paquetes. Ambos administradores de paquetes utilizan el concepto de un aviso de actualización como un archivo llamado `updateinfo.xml`. Un aviso de actualización es simplemente una colección de paquetes que solucionan un problema determinado. A los paquetes individuales no se les asignan clasificaciones ni niveles de gravedad. Por este motivo, Patch Manager asigna los atributos de un aviso de actualización a los paquetes relacionados e instala los paquetes en función de los filtros de clasificación especificados en la base de referencia de parches.

RHEL 7

> **Note**
>
> Los siguientes ID de repositorio están asociados con RHUI 2. RHUI 3 se lanzó en diciembre de 2019 y supuso la introducción de un esquema de nomenclatura diferente para los ID de repositorio de Yum. En función de la AMI RHEL-7 desde la que cree los nodos administrados, es posible que tenga que actualizar los comandos. Para obtener más información, consulte Repository IDs for RHEL 7 in AWS Have Changed en el Portal del cliente de Red Hat.

- ID del repositorio: `rhui-REGION-client-config-server-7/x86_64`

  Nombre del repositorio: `Red Hat Update Infrastructure 2.0 Client Configuration Server 7`
- ID del repositorio: `rhui-REGION-rhel-server-releases/7Server/x86_64`

  Nombre del repositorio: `Red Hat Enterprise Linux Server 7 (RPMs)`
- ID del repositorio: `rhui-REGION-rhel-server-rh-common/7Server/x86_64`

  Nombre del repositorio: `Red Hat Enterprise Linux Server 7 RH Common (RPMs)`

AlmaLinux 8, RHEL 8 y Rocky Linux 8

- ID del repositorio: `rhel-8-appstream-rhui-rpms`

  Nombre del repositorio: `Red Hat Enterprise Linux 8 for x86_64 - AppStream from RHUI (RPMs)`
- ID del repositorio: `rhel-8-baseos-rhui-rpms`

  Nombre del repositorio: `Red Hat Enterprise Linux 8 for x86_64 - BaseOS from RHUI (RPMs)`
- ID del repositorio: `rhui-client-config-server-8`

  Nombre del repositorio: `Red Hat Update Infrastructure 3 Client Configuration Server 8`

AlmaLinux 9, RHEL 9 y Rocky Linux 9

- ID del repositorio: `rhel-9-appstream-rhui-rpms`

Nombre del repositorio: `Red Hat Enterprise Linux 9 for x86_64 - AppStream from RHUI (RPMs)`

- ID del repositorio: `rhel-9-baseos-rhui-rpms`

  Nombre del repositorio: `Red Hat Enterprise Linux 9 for x86_64 - BaseOS from RHUI (RPMs)`

- ID del repositorio: `rhui-client-config-server-9`

  Nombre del repositorio: `Red Hat Enterprise Linux 9 Client Configuration`

SLES

En los nodos administrados de SUSE Linux Enterprise Server (SLES), la biblioteca ZYPP obtiene la lista de revisiones disponibles (una colección de paquetes) de las siguientes ubicaciones:

- Lista de repositorios: `etc/zypp/repos.d/*`
- Información de paquetes: `/var/cache/zypp/raw/*`

Los nodos administrados de SLES utilizan Zypper como administrador de paquetes, y Zypper utiliza el concepto de revisión. Un parche es una simple colección de paquetes que corrigen un problema concreto. Patch Manager se ocupa de todos los paquetes que se referencian en un parche como relacionados con la seguridad. Dado que a los paquetes individuales no se les asignan clasificaciones ni gravedad, Patch Manager les asigna los atributos del parche al que pertenecen.

Servidor Ubuntu

En Ubuntu Server, el servicio de base de referencia de revisiones de Systems Manager utiliza repositorios (repos) preconfigurados en el nodo administrado. Estos repositorios preconfigurados se utilizan para obtener una lista actualizada de actualizaciones de paquetes disponibles. Por este motivo, Systems Manager realiza el equivalente a un comando `sudo apt-get update`.

A continuación, los paquetes se filtran desde los repositorios *`codename`*`-security`, cuyo nombre de código es único para la versión de lanzamiento, como `trusty` para Ubuntu Server 14. Patch Manager identifica únicamente las actualizaciones que forman parte de estos repositorios:

- 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 Server 23.04 (`lunar-security`)
- Ubuntu Server 23.10 (`mantic-security`)
- Ubuntu Server 24.04 LTS (`noble-security`)
- Ubuntu Server 24.10 (`oracular-security`)
- Ubuntu Server 25.04 (`plucky-security`)

Windows Server

En los sistemas operativos Microsoft Windows, Patch Manager recupera una lista de las actualizaciones disponibles que Microsoft publica a través de los servicios de Microsoft Update y que están disponibles de manera automática para Windows Server Update Services (WSUS).

> **Note**
>
> Patch Manager solo pone a disposición parches para versiones del sistema operativo Windows Server que son compatibles con Patch Manager. Por ejemplo, Patch Manager no se puede utilizar para aplicar parches a Windows RT.

Patch Manager monitorea continuamente las nuevas actualizaciones en cada Región de AWS. La lista de las actualizaciones de Windows disponibles se actualiza en cada región al menos una vez al día. Cuando la información sobre parches de Microsoft se procesa, Patch Manager elimina las actualizaciones que se han sustituido por actualizaciones posteriores de su lista de parches. Por lo tanto, solo se muestra la última actualización y, en su caso, están disponibles para su instalación. Por ejemplo, si `KB4012214` sustituye a `KB3135456`, solo `KB4012214` estará disponible como una actualización en Patch Manager.

Del mismo modo, solo Patch Manager puede instalar las revisiones que estén disponibles en el nodo gestionado durante la operación de aplicación de revisiones. De forma predeterminada, en Windows Server 2019 y Windows Server 2022 se eliminan las actualizaciones que se sustituyen por actualizaciones posteriores. Como resultado, si utiliza el parámetro `ApproveUntilDate`

en la línea de base de revisiones Windows Server, pero la fecha seleccionada en el parámetro `ApproveUntilDate` es anterior a la fecha del último parche, se produce la siguiente situación:

- La revisión sustituida se elimina del nodo y, por lo tanto, no se puede instalar con Patch Manager.
- La última revisión de reemplazo está presente en el nodo, pero aún no se ha aprobado su instalación en la fecha especificada por el parámetro `ApproveUntilDate`.

Esto significa que el nodo gestionado es compatible con las operaciones de Systems Manager, aunque es posible que no se haya instalado una revisión crítica del mes anterior. Esta misma situación puede ocurrir cuando se utiliza el parámetro `ApproveAfterDays`. Debido al comportamiento de las revisiones sustituidas por Microsoft, es posible establecer un número (generalmente superior a 30 días) para que las revisiones para Windows Server no se instalen nunca si la última revisión disponible de Microsoft se publica antes de que hayan transcurrido los días en `ApproveAfterDays`. Tenga en cuenta que este comportamiento del sistema no se aplica si ha modificado la configuración con objetos de políticas globales (GPO) de Windows para que la versión sustituida esté disponible en los nodos administrados.

> **Note**
>
> En algunos casos, Microsoft lanza parches para las aplicaciones que no especifican una hora ni una fecha de actualización. En estos casos, se suministra una fecha y hora actualizadas de `01/01/1970` de forma predeterminada.

Cómo especificar un repositorio de origen de parches alternativo (Linux)

Cuando se utilizan los repositorios predeterminados configurados en un nodo administrado para operaciones de aplicación de revisiones, Patch Manager, una herramienta de AWS Systems Manager, busca o instala las revisiones relacionadas con la seguridad. Este es el comportamiento predeterminado de Patch Manager. Para obtener información completa acerca de cómo Patch Manager selecciona e instala los parches de seguridad, consulte Cómo se seleccionan las revisiones de seguridad.

Sin embargo, en los sistemas Linux, también puede utilizar Patch Manager para instalar revisiones que no estén relacionadas con la seguridad o que se encuentren en un repositorio de origen diferente del configurado de forma predeterminada en el nodo administrado. Puede especificar repositorios de origen de parches alternativos al crear una base de referencia de parches

personalizada. En cada base de referencia de parches personalizada, es posible especificar configuraciones de origen de parches para un máximo de 20 versiones de un sistema operativo Linux compatible.

Por ejemplo, supongamos que su flota de Ubuntu Server incluye nodos administrados de Ubuntu Server 14.04 y Ubuntu Server 16.04. En este caso, puede especificar repositorios alternativos para cada versión en la misma base de referencia de parches personalizada. Para cada versión, es necesario proporcionar un nombre y especificar el tipo de versión del sistema operativo (producto), así como proporcionar una configuración de repositorio. También es posible especificar un único repositorio de origen alternativo que se aplique a todas las versiones de un sistema operativo compatible.

> **Note**
>
> La ejecución de una base de referencia de revisiones personalizada que especifica repositorios de revisiones alternativos para un nodo administrado no los convierte en los nuevos repositorios predeterminados del sistema operativo. Una vez completada la operación de aplicación de revisiones, los repositorios previamente configurados como valores predeterminados del sistema operativo del nodo siguen siendo los predeterminados.

Para obtener una lista de escenarios de ejemplo del uso de esta opción, consulte Ejemplos de uso de repositorios de origen de parches alternativos más adelante en este tema.

Para obtener más información sobre las bases de referencia de parches predeterminadas y personalizadas, consulte Líneas de base de revisiones personalizadas y predefinidas.

Ejemplo: uso de la consola

Para especificar repositorios de origen de parches alternativos al trabajar en la consola de Systems Manager, utilice la sección Patch sources (Orígenes de parches) de la página Create patch baseline (Crear una base de referencia de parches). Para obtener información sobre el uso de las opciones de Orígenes de parches, consulte Creación de una línea de base de revisiones personalizada para Linux.

Ejemplos: uso de la AWS CLI

Para ver un ejemplo de cómo usar la opción `--sources` con la AWS Command Line Interface (AWS CLI), consulte Creación de una base de referencia de parches con repositorios personalizados para distintas versiones del sistema operativo.

Temas

- Consideraciones importantes para repositorios alternativos
- Ejemplos de uso de repositorios de origen de parches alternativos

Consideraciones importantes para repositorios alternativos

Tenga en cuenta los siguientes puntos a la hora de planificar su estrategia de aplicación de parches con repositorios de parches alternativos.

Solo se utilizan los repositorios especificados para la aplicación de parches

Especificar repositorios alternativos no significa especificar repositorios adicionales. Puede elegir especificar repositorios distintos de los configurados como valores predeterminados en un nodo administrado. Sin embargo, también debe especificar los repositorios predeterminados como parte de la configuración de origen de parches alternativos si desea que sus actualizaciones se apliquen.

Por ejemplo, en los nodos administrados de Amazon Linux 2, los repositorios predeterminados son `amzn2-core` y `amzn2extra-docker`. Si desea incluir el repositorio Extra Packages for Enterprise Linux (EPEL) en sus operaciones de aplicación de parches, debe especificar los tres repositorios como repositorios alternativos.

> **Note**
>
> La ejecución de una base de referencia de revisiones personalizada que especifica repositorios de revisiones alternativos para un nodo administrado no los convierte en los nuevos repositorios predeterminados del sistema operativo. Una vez completada la operación de aplicación de revisiones, los repositorios previamente configurados como valores predeterminados del sistema operativo del nodo siguen siendo los predeterminados.

El comportamiento de aplicación de parches para las distribuciones basadas en YUM depende del manifiesto updateinfo.xml.

Al especificar repositorios de revisión alternativos para las distribuciones basadas en YUM, como Amazon Linux 1 o Amazon Linux 2, Red Hat Enterprise Linux o CentOS, el comportamiento de las revisiones depende de si el repositorio incluye un manifiesto de actualización en forma de un archivo `updateinfo.xml` con formato completo y correcto. Este archivo especifica la fecha de lanzamiento, clasificaciones y gravedad de los distintos paquetes. Cualquiera de los siguientes afectarán al comportamiento de aplicación de parches:

- Si filtra por Classification (Clasificación) y Severity (Gravedad), pero no están especificados en `updateinfo.xml`, el paquete no se incluirá el filtro. Esto también significa que los paquetes sin un archivo `updateinfo.xml` no se incluyen en la aplicación de parches.
- Si filtra por ApprovalAfterDays, pero la fecha de lanzamiento del paquete no está en formato de fecha de inicio Unix (o no tiene fecha de lanzamiento especificada), el paquete no se incluirá en el filtro.
- Existe una excepción cuando selecciona la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad en la página Crear línea de base de revisiones. En este caso, los paquetes sin un archivo `updateinfo.xml` (o que contengan este archivo sin valores de Classification [Clasificación], Severity [Gravedad] y Date [Fecha] con el formato adecuado) se incluirán en la lista de revisiones previamente filtrada. (Deben seguir cumpliendo los otros requisitos de reglas de base de referencia de los parches para instalarse).

Ejemplos de uso de repositorios de origen de parches alternativos

Ejemplo 1: actualizaciones que no son de seguridad para Ubuntu Server

Ya está utilizando Patch Manager para instalar revisiones de seguridad en una flota de nodos administrados de Ubuntu Server mediante la base de referencia de revisiones predefinida proporcionada por AWS, `AWS-UbuntuDefaultPatchBaseline`. Puede crear una nueva base de referencia de parches basada en esta base predeterminada, pero especifica en las reglas de aprobación que también desea instalar las actualizaciones que no son de seguridad y que forman parte de la distribución predeterminada. Cuando esta base de referencia de revisiones se ejecuta en los nodos, se aplican tanto las revisiones de seguridad como las que no lo son. También puede elegir aprobar los parches que no son de seguridad en las excepciones de parches especificadas para una base de referencia.

Ejemplo 2: Personal Package Archives (PPA) para Ubuntu Server Personal Package Archives

Los nodos administrados de Ubuntu Server ejecutan software que se distribuye a través de Personal Package Archives (PPA) para Ubuntu. En este caso, debe crear una base de referencia de revisiones que especifica un repositorio de PPA que ha configurado en el nodo administrado como repositorio de origen para la operación de aplicación de revisiones. A continuación, utilice Run Command para ejecutar el documento de base de referencia de revisiones en los nodos.

Ejemplo 3: aplicaciones corporativas internas en Amazon Linux

Necesita ejecutar algunas aplicaciones necesarias para el cumplimiento normativo del sector en los nodos administrados de Amazon Linux. Puede configurar un repositorio para estas aplicaciones en

los nodos, utilizar YUM para instalar inicialmente las aplicaciones y, a continuación, actualizar o crear una nueva base de referencia de revisiones para incluir este nuevo repositorio corporativo. Una vez hecho esto, puede utilizar Run Command para ejecutar el documento `AWS-RunPatchBaseline` con la opción `Scan` para comprobar si el paquete corporativo aparece entre los paquetes instalados y está actualizado en el nodo administrado. Si no está actualizado, puede ejecutar el documento de nuevo con la opción `Install` para actualizar las aplicaciones.

Cómo se instalan las revisiones

Patch Manager, una herramienta de AWS Systems Manager, utiliza el mecanismo integrado adecuado para un tipo de sistema operativo con el fin de instalar actualizaciones en un nodo administrado. Por ejemplo, en Windows Server, se utiliza la API de Windows Update y en Amazon Linux 2 se utiliza el administrador de paquetes `yum`.

Patch Manager no instala un paquete nuevo que sustituya a un paquete obsoleto que esté instalado actualmente. (Excepciones: el paquete nuevo es una dependencia de otro paquete que se está actualizando o el paquete nuevo tiene el mismo nombre que el paquete obsoleto). En su lugar, Patch Manager informa sobre las actualizaciones disponibles para los paquetes instalados y las instala. Este enfoque ayuda a evitar cambios inesperados en la funcionalidad del sistema que pueden producirse cuando un paquete reemplaza a otro.

Si necesita desinstalar un paquete que ha quedado obsoleto e instalar un paquete que lo sustituya, puede que necesite utilizar un script personalizado o utilizar comandos del administrador de paquetes fuera de las operaciones estándar del Patch Manager.

En el resto de esta sección, se explica cómo Patch Manager instala parches en un sistema operativo.

Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022, and Amazon Linux 2023

En los nodos administrados de Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 y Amazon Linux 2023, el flujo de trabajo de la instalación de parches es el siguiente:

1. Si se especifica una lista de parches mediante una URL de https o una URL de tipo ruta de Amazon Simple Storage Service (Amazon S3) con el parámetro `InstallOverrideList` para los documentos `AWS-RunPatchBaseline` o `AWS-RunPatchBaselineAssociation`, se instalan los parches de la lista y se omiten los pasos comprendidos entre el 2 y el 7.
2. Aplique GlobalFilters tal y como se especifica en la base de referencia de parches, de modo que se mantendrán los paquetes cualificados para poder procesarse más adelante.

3. Aplique ApprovalRules tal y como se especifica en la base de referencia de parches. Cada regla de aprobación puede definir un paquete como aprobado.

   Sin embargo, las reglas de aprobación también están sujetas a si se seleccionó la casilla de verificación Include nonsecurity updates (Incluir actualizaciones no relacionadas con la seguridad) durante la creación o la última actualización de una base de referencia de parches.

   Además, si se excluyen las actualizaciones no relacionadas con la seguridad, se aplica una regla implícita para seleccionar únicamente los paquetes con actualizaciones en repositorios de seguridad. Para cada paquete, la versión candidata del paquete (que suele ser la última) debe formar parte de un repositorio de seguridad.

   Si se incluyen actualizaciones no relacionadas con la seguridad, también se tienen en cuenta los parches de los demás repositorios.

4. Aplique ApprovedPatches tal y como se especifica en la base de referencia de parches. Las revisiones aprobadas se aprueban para su actualización, incluso si se descartan por GlobalFilters o si ninguna regla de aprobación especificada en ApprovalRules les concede la aprobación.

5. Aplique RejectedPatches tal y como se especifica en la base de referencia de parches. Los parches rechazados se eliminan de la lista de parches aprobados y no se aplicarán.

6. Si hay varias versiones de un parche aprobadas, se aplica la última.

7. La API de actualización de YUM (Amazon Linux 1, Amazon Linux 2) o la API de actualización de DNF (Amazon Linux 2022, Amazon Linux 2023) se aplica a los parches aprobados de la siguiente manera:

   - Para las líneas de base de revisiones predeterminadas y proporcionadas por AWS, solo se aplican las revisiones especificadas en `updateinfo.xml` (solo actualizaciones de seguridad). Esto se debe a que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad no está seleccionada. Las líneas de base predefinidas equivalen a una línea de base personalizada con lo siguiente:
     - La casilla de verificación Incluir actualizaciones no relacionadas con la seguridad no está seleccionada
     - Una lista de GRAVEDAD de `[Critical, Important]`
     - Una lista de CLASIFICACIÓN de `[Security, Bugfix]`

     En Amazon Linux 1 y Amazon Linux 2, el comando YUM equivalente para este flujo de trabajo es el siguiente:

```
sudo yum update-minimal --sec-severity=Critical,Important --bugfix -y
```

En Amazon Linux 2022 y Amazon Linux 2023, el comando dnf equivalente para este flujo de trabajo es el siguiente:

```
sudo dnf upgrade-minimal --sec-severity=Critical --sec-severity=Important --
bugfix -y
```

Si la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad está seleccionada, se aplican las revisiones que están en `updateinfo.xml` y no las que están en `updateinfo.xml` (actualizaciones de seguridad y no relacionadas con la seguridad).

En Amazon Linux 1 y Amazon Linux 2, si se selecciona una línea de base con Incluir actualizaciones no relacionadas con la seguridad y tiene una lista de gravedad de `[Critical, Important]` y una lista de clasificación de `[Security, Bugfix]`, el comando YUM equivalente es el siguiente:

```
sudo yum update --security --sec-severity=Critical,Important --bugfix -y
```

En Amazon Linux 2022 y Amazon Linux 2023, el comando dnf equivalente es el siguiente:

```
sudo dnf upgrade --security --sec-severity=Critical --sec-severity=Important --
bugfix -y
```

> **Note**
>
> Los paquetes nuevos que sustituyen a los ya obsoletos con nombres diferentes se instalan si ejecuta estos comandos `yum` o `dnf` fuera del Patch Manager. Sin embargo, no se instalan mediante las operaciones equivalentes del Patch Manager.

Detalles adicionales sobre los parches para Amazon Linux 2022 y Amazon Linux 2023

Asistencia para el nivel de gravedad “None”

Amazon Linux 2022 y Amazon Linux 2023 también admiten el nivel de gravedad de la revisión `None`, que es reconocido por el administrador de paquetes DNF.

Asistencia para el nivel de gravedad “Medium”

Para Amazon Linux 2022 y Amazon Linux 2023, un nivel de gravedad de revisiones `Medium` equivale a un nivel de gravedad `Moderate` que podría definirse en algunos repositorios externos. Si incluye `Medium` revisiones de gravedad en la línea de base de revisiones, también se instalarán en las instancias `Moderate` revisiones de gravedad de revisiones externas.

Cuando consulta los datos de cumplimiento mediante la acción DescribeInstancePatches de la API, el filtrado para el nivel de gravedad `Medium` informa revisiones con niveles de gravedad tanto `Medium` como `Moderate`.

Manejo transitivo de dependencias para Amazon Linux 2022 y Amazon Linux 2023

En Amazon Linux 2022 y Amazon Linux 2023, es posible que Patch Manager instale versiones de dependencias transitivas diferentes a las que instalan los comandos de `dnf` equivalentes. Las dependencias transitivas son paquetes que se instalan automáticamente para satisfacer los requisitos de otros paquetes (dependencias de dependencias).

Por ejemplo, `dnf upgrade-minimal --security` instala las versiones mínimas de las dependencias transitivas necesarias para resolver los problemas de seguridad conocidos, mientras que Patch Manager instala las últimas versiones disponibles de las mismas dependencias transitivas.

8. El nodo administrado se reinicia si las actualizaciones se instalaron. (Excepción: si el parámetro `RebootOption` se configura en `NoReboot` en el documento `AWS-RunPatchBaseline`, el nodo administrado no se reinicia una vez que se ejecuta Patch Manager. Para obtener más información, consulte Nombre del parámetro: `RebootOption`).

CentOS and CentOS Stream

En los nodos administrados de CentOS y CentOS Stream, el flujo de trabajo de instalación de revisiones es el siguiente:

1. Si se especifica una lista de parches mediante una URL de https o una URL de tipo ruta de Amazon Simple Storage Service (Amazon S3) con el parámetro `InstallOverrideList` para los documentos `AWS-RunPatchBaseline` o `AWS-RunPatchBaselineAssociation`, se instalan los parches de la lista y se omiten los pasos comprendidos entre el 2 y el 7.

Aplique GlobalFilters tal y como se especifica en la base de referencia de parches, de modo que se mantendrán los paquetes cualificados para poder procesarse más adelante.

2. Aplique ApprovalRules tal y como se especifica en la base de referencia de parches. Cada regla de aprobación puede definir un paquete como aprobado.

   Sin embargo, las reglas de aprobación también están sujetas a si se seleccionó la casilla de verificación Include nonsecurity updates (Incluir actualizaciones no relacionadas con la seguridad) durante la creación o la última actualización de una base de referencia de parches.

   Además, si se excluyen las actualizaciones no relacionadas con la seguridad, se aplica una regla implícita para seleccionar únicamente los paquetes con actualizaciones en repositorios de seguridad. Para cada paquete, la versión candidata del paquete (que suele ser la última) debe formar parte de un repositorio de seguridad.

   Si se incluyen actualizaciones no relacionadas con la seguridad, también se tienen en cuenta los parches de los demás repositorios.

3. Aplique ApprovedPatches tal y como se especifica en la base de referencia de parches. Las revisiones aprobadas se aprueban para su actualización, incluso si se descartan por GlobalFilters o si ninguna regla de aprobación especificada en ApprovalRules les concede la aprobación.
4. Aplique RejectedPatches tal y como se especifica en la base de referencia de parches. Los parches rechazados se eliminan de la lista de parches aprobados y no se aplicarán.
5. Si hay varias versiones de un parche aprobadas, se aplica la última.
6. Se aplica la API de actualización YUM (en las versiones CentOS 6.x y 7.x) o la actualización DNF (en CentOS 8 y CentOS Stream) en las revisiones aprobadas.

> **Note**
>
> Para CentOS, Patch Manager puede instalar versiones de dependencias transitivas diferentes a las que instalan de los comandos equivalentes de `dnf`. Las dependencias transitivas son paquetes que se instalan automáticamente para satisfacer los requisitos de otros paquetes (dependencias de dependencias).
> Por ejemplo, `dnf upgrade-minimal --security` instala las versiones mínimas de las dependencias transitivas necesarias para resolver los problemas de seguridad

conocidos, mientras que Patch Manager instala las últimas versiones disponibles de las mismas dependencias transitivas.

7. El nodo administrado se reinicia si las actualizaciones se instalaron. (Excepción: si el parámetro `RebootOption` se configura en `NoReboot` en el documento `AWS-RunPatchBaseline`, el nodo administrado no se reinicia una vez que se ejecuta Patch Manager. Para obtener más información, consulte Nombre del parámetro: `RebootOption`).

Servidor Debian and Raspberry Pi OS

En las instancias de Debian Server y Raspberry Pi OS (anteriormente Raspbian), el flujo de trabajo de instalación de revisiones es el siguiente:

1. Si se especifica una lista de parches mediante una URL de https o una URL de tipo ruta de Amazon Simple Storage Service (Amazon S3) con el parámetro `InstallOverrideList` para los documentos `AWS-RunPatchBaseline` o `AWS-RunPatchBaselineAssociation`, se instalan los parches de la lista y se omiten los pasos comprendidos entre el 2 y el 7.
2. Si está disponible una actualización para `python3-apt` (una interfaz de biblioteca de Python para `libapt`), se actualiza a la versión más reciente. (Este paquete no relacionado con la seguridad se actualiza aunque no haya seleccionado la opción Include nonsecurity updates [Incluir actualizaciones no relacionadas con la seguridad]).

> **Important**
>
> Solo en Debian Server 8: debido a que algunos nodos administrados de Debian Server 8.* hacen referencia a un repositorio de paquetes obsoleto (`jessie-backports`), Patch Manager lleva a cabo los siguientes pasos adicionales para garantizar que las operaciones de aplicación de revisiones se efectúen correctamente:
>
> a. En el nodo administrado, la referencia al repositorio `jessie-backports` se indica en la lista de ubicaciones de origen (`/etc/apt/sources.list.d/jessie-backports`). Por consiguiente, no se intenta descargar los parches desde esa ubicación.
>
> b. Se importa una clave de firma de actualización de seguridad de Stretch. Esta clave proporciona los permisos necesarios para las operaciones de actualización e instalación en las distribuciones de Debian Server 8.*.

c. A esta altura, se ejecuta la operación `apt-get` para asegurarse de que la versión más reciente de `python3-apt` esté instalada antes de que comience el proceso de aplicación de parches.

d. Una vez finalizado el proceso de instalación, se restaura la referencia al repositorio `jessie-backports` y se elimina la clave de firma del conjunto de claves de las fuentes de apt. Esto se hace para dejar la configuración del sistema tal como estaba antes de la operación de aplicación de parches.

La próxima vez que Patch Manager actualice el sistema, se repetirá el mismo proceso.

3. Aplique GlobalFilters tal y como se especifica en la base de referencia de parches, de modo que se mantendrán los paquetes cualificados para poder procesarse más adelante.
4. Aplique ApprovalRules tal y como se especifica en la base de referencia de parches. Cada regla de aprobación puede definir un paquete como aprobado.

> **Note**
>
> Como no es posible determinar de forma fiable las fechas de lanzamiento de los paquetes de actualización para Debian Server, las opciones de aprobación automática no son compatibles con este sistema operativo.

Sin embargo, las reglas de aprobación también están sujetas a si se seleccionó la casilla de verificación Include nonsecurity updates (Incluir actualizaciones no relacionadas con la seguridad) durante la creación o la última actualización de una base de referencia de parches.

Además, si se excluyen las actualizaciones no relacionadas con la seguridad, se aplica una regla implícita para seleccionar únicamente los paquetes con actualizaciones en repositorios de seguridad. Para cada paquete, la versión candidata del paquete (que suele ser la última) debe formar parte de un repositorio de seguridad.

Si se incluyen actualizaciones no relacionadas con la seguridad, también se tienen en cuenta los parches de los demás repositorios.

> **Note**
>
> En Debian Server y Raspberry Pi OS, las versiones candidatas a revisiones se limitan a las revisiones incluidas en `debian-security`.

5. Aplique ApprovedPatches tal y como se especifica en la base de referencia de parches. Las revisiones aprobadas se aprueban para su actualización, incluso si se descartan por GlobalFilters o si ninguna regla de aprobación especificada en ApprovalRules les concede la aprobación.
6. Aplique RejectedPatches tal y como se especifica en la base de referencia de parches. Los parches rechazados se eliminan de la lista de parches aprobados y no se aplicarán.
7. La biblioteca de APT se utiliza para actualizar paquetes.

   > **Note**
   >
   > Patch Manager no admite el uso de la opción `Pin-Priority` de APT para asignar prioridades a los paquetes. Patch Manager agrega las actualizaciones disponibles de todos los repositorios habilitados y selecciona la actualización más reciente que coincide con la línea de base de cada paquete instalado.

8. El nodo administrado se reinicia si las actualizaciones se instalaron. (Excepción: si el parámetro `RebootOption` se configura en `NoReboot` en el documento `AWS-RunPatchBaseline`, el nodo administrado no se reinicia una vez que se ejecuta Patch Manager. Para obtener más información, consulte Nombre del parámetro: `RebootOption`).

macOS

En los nodos administrados macOS, el flujo de trabajo de instalación de revisiones es el siguiente:

1. La lista de propiedades `/Library/Receipts/InstallHistory.plist` es un registro de software que se ha instalado y actualizado mediante los administradores de paquetes `softwareupdate` y `installer`. Mediante la herramienta de línea de comandos `pkgutil` (para `installer`) y el administrador de paquetes `softwareupdate`, se ejecutan comandos de la CLI para analizar esta lista.

   Para `installer`, la respuesta a los comandos de la CLI incluye detalles de `package name`, `version`, `volume`, `location` y `install-time`, pero Patch Manager solo utiliza `package name` y `version`.

   Para `softwareupdate`, la respuesta a los comandos de la CLI incluye el nombre del paquete (`display name`), `version` y `date`, pero Patch Manager utiliza únicamente el nombre del paquete y la versión.

Para Brew y Brew Cask, Homebrew no admite que sus comandos se ejecuten con el usuario raíz. Como resultado, Patch Manager consulta y ejecuta los comandos de Homebrew como propietario del directorio de Homebrew o como usuario válido perteneciente al grupo de propietarios de ese directorio. Los comandos se asemejan a `softwareupdate` y `installer`, y se ejecutan a través de un subproceso de Python para recopilar los datos de los paquetes, y el resultado se analiza con el objetivo de identificar los nombres y las versiones de los paquetes.

2. Aplique GlobalFilters tal y como se especifica en la base de referencia de parches, de modo que se mantendrán los paquetes cualificados para poder procesarse más adelante.
3. Aplique ApprovalRules tal y como se especifica en la base de referencia de parches. Cada regla de aprobación puede definir un paquete como aprobado.
4. Aplique ApprovedPatches tal y como se especifica en la base de referencia de parches. Las revisiones aprobadas se aprueban para su actualización, incluso si se descartan por GlobalFilters o si ninguna regla de aprobación especificada en ApprovalRules les concede la aprobación.
5. Aplique RejectedPatches tal y como se especifica en la base de referencia de parches. Los parches rechazados se eliminan de la lista de parches aprobados y no se aplicarán.
6. Si hay varias versiones de un parche aprobadas, se aplica la última.
7. Invoque la CLI del paquete correspondiente en el nodo administrado para procesar las revisiones aprobadas de la siguiente manera:

> **Note**
>
> `installer` carece de la funcionalidad para buscar e instalar actualizaciones. Por lo tanto, para `installer`, Patch Manager solo notifica qué paquetes están instalados. Como resultado, los paquetes `installer` nunca se notifican como `Missing`.

- Para las líneas de base de revisiones predeterminadas proporcionadas por AWS y las líneas de base de revisiones personalizadas en las que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad no está seleccionada, solo se aplican las actualizaciones de seguridad.
- Para las líneas de base de revisiones personalizadas en las que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad sí está seleccionada, se aplican tanto las actualizaciones de seguridad como las que no están relacionadas con la seguridad.

8. El nodo administrado se reinicia si las actualizaciones se instalaron. (Excepción: si el parámetro `RebootOption` se configura en `NoReboot` en el documento `AWS-RunPatchBaseline`, el nodo administrado no se reinicia una vez que se ejecuta Patch Manager. Para obtener más información, consulte Nombre del parámetro: `RebootOption`).

Oracle Linux

En los nodos administrados Oracle Linux, el flujo de trabajo de instalación de revisiones es el siguiente:

1. Si se especifica una lista de parches mediante una URL de https o una URL de tipo ruta de Amazon Simple Storage Service (Amazon S3) con el parámetro `InstallOverrideList` para los documentos `AWS-RunPatchBaseline` o `AWS-RunPatchBaselineAssociation`, se instalan los parches de la lista y se omiten los pasos comprendidos entre el 2 y el 7.
2. Aplique GlobalFilters tal y como se especifica en la base de referencia de parches, de modo que se mantendrán los paquetes cualificados para poder procesarse más adelante.
3. Aplique ApprovalRules tal y como se especifica en la base de referencia de parches. Cada regla de aprobación puede definir un paquete como aprobado.

   Sin embargo, las reglas de aprobación también están sujetas a si se seleccionó la casilla de verificación Include nonsecurity updates (Incluir actualizaciones no relacionadas con la seguridad) durante la creación o la última actualización de una base de referencia de parches.

   Además, si se excluyen las actualizaciones no relacionadas con la seguridad, se aplica una regla implícita para seleccionar únicamente los paquetes con actualizaciones en repositorios de seguridad. Para cada paquete, la versión candidata del paquete (que suele ser la última) debe formar parte de un repositorio de seguridad.

   Si se incluyen actualizaciones no relacionadas con la seguridad, también se tienen en cuenta los parches de los demás repositorios.
4. Aplique ApprovedPatches tal y como se especifica en la base de referencia de parches. Las revisiones aprobadas se aprueban para su actualización, incluso si se descartan por GlobalFilters o si ninguna regla de aprobación especificada en ApprovalRules les concede la aprobación.
5. Aplique RejectedPatches tal y como se especifica en la base de referencia de parches. Los parches rechazados se eliminan de la lista de parches aprobados y no se aplicarán.
6. Si hay varias versiones de un parche aprobadas, se aplica la última.

7. En los nodos administrados de la versión 7, la API de actualización de YUM se aplica a las revisiones aprobadas como se indica a continuación:

   - Para las líneas de base de revisiones proporcionadas por AWS y para las líneas de base de revisiones personalizadas en las que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad no está seleccionada, solo se aplican las revisiones especificadas en `updateinfo.xml` (solo actualizaciones de seguridad).

     El comando yum equivalente para este flujo de trabajo es:

     ```
     sudo yum update-minimal --sec-severity=Important,Moderate --bugfix -y
     ```

   - Para las líneas de base de revisiones personalizadas en las que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad sí está seleccionada, se aplican las revisiones que están en `updateinfo.xml` y las que no están en `updateinfo.xml` (actualizaciones de seguridad y no relacionadas con la seguridad).

     El comando yum equivalente para este flujo de trabajo es:

     ```
     sudo yum update --security --bugfix -y
     ```

     En los nodos administrados de las versiones 8 y 9, la API de actualización de DNF se aplica a las revisiones aprobadas como se indica a continuación:

     - Para las líneas de base de revisiones proporcionadas por AWS y para las líneas de base de revisiones personalizadas en las que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad no está seleccionada, solo se aplican las revisiones especificadas en `updateinfo.xml` (solo actualizaciones de seguridad).

       El comando yum equivalente para este flujo de trabajo es:

       ```
       sudo dnf upgrade-minimal --security --sec-severity=Moderate --sec-
       severity=Important
       ```

       > **Note**
       >
       > En el caso de Oracle Linux, Patch Manager puede instalar versiones de dependencias transitivas distintas a las que instalan los comandos de `dnf` equivalentes. Las dependencias transitivas son paquetes que se instalan

automáticamente para satisfacer los requisitos de otros paquetes (dependencias de dependencias).
Por ejemplo, `dnf upgrade-minimal --security` instala las versiones mínimas de las dependencias transitivas necesarias para resolver los problemas de seguridad conocidos, mientras que Patch Manager instala las últimas versiones disponibles de las mismas dependencias transitivas.

- Para las líneas de base de revisiones personalizadas en las que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad sí está seleccionada, se aplican las revisiones que están en `updateinfo.xml` y las que no están en `updateinfo.xml` (actualizaciones de seguridad y no relacionadas con la seguridad).

  El comando yum equivalente para este flujo de trabajo es:

```
sudo dnf upgrade --security --bugfix
```

> **Note**
>
> Los paquetes nuevos que sustituyen a los ya obsoletos con nombres diferentes se instalan si ejecuta estos comandos `yum` o `dnf` fuera del Patch Manager. Sin embargo, no se instalan mediante las operaciones equivalentes del Patch Manager.

8. El nodo administrado se reinicia si las actualizaciones se instalaron. (Excepción: si el parámetro `RebootOption` se configura en `NoReboot` en el documento `AWS-RunPatchBaseline`, el nodo administrado no se reinicia una vez que se ejecuta Patch Manager. Para obtener más información, consulte Nombre del parámetro: `RebootOption`).

AlmaLinux, RHEL, and Rocky Linux

En los nodos administrados de AlmaLinux, Red Hat Enterprise Linux y Rocky Linux, el flujo de trabajo de instalación de revisión es el siguiente:

1. Si se especifica una lista de parches mediante una URL de https o una URL de tipo ruta de Amazon Simple Storage Service (Amazon S3) con el parámetro `InstallOverrideList` para los documentos `AWS-RunPatchBaseline` o `AWS-RunPatchBaselineAssociation`, se instalan los parches de la lista y se omiten los pasos comprendidos entre el 2 y el 7.

2. Aplique GlobalFilters tal y como se especifica en la base de referencia de parches, de modo que se mantendrán los paquetes cualificados para poder procesarse más adelante.
3. Aplique ApprovalRules tal y como se especifica en la base de referencia de parches. Cada regla de aprobación puede definir un paquete como aprobado.

   Sin embargo, las reglas de aprobación también están sujetas a si se seleccionó la casilla de verificación Include nonsecurity updates (Incluir actualizaciones no relacionadas con la seguridad) durante la creación o la última actualización de una base de referencia de parches.

   Además, si se excluyen las actualizaciones no relacionadas con la seguridad, se aplica una regla implícita para seleccionar únicamente los paquetes con actualizaciones en repositorios de seguridad. Para cada paquete, la versión candidata del paquete (que suele ser la última) debe formar parte de un repositorio de seguridad.

   Si se incluyen actualizaciones no relacionadas con la seguridad, también se tienen en cuenta los parches de los demás repositorios.
4. Aplique ApprovedPatches tal y como se especifica en la base de referencia de parches. Las revisiones aprobadas se aprueban para su actualización, incluso si se descartan por GlobalFilters o si ninguna regla de aprobación especificada en ApprovalRules les concede la aprobación.
5. Aplique RejectedPatches tal y como se especifica en la base de referencia de parches. Los parches rechazados se eliminan de la lista de parches aprobados y no se aplicarán.
6. Si hay varias versiones de un parche aprobadas, se aplica la última.
7. La API de actualización de YUM (en RHEL 7) o la API de actualización de DNF (en AlmaLinux 8 y 9, RHEL 8, 9 y 10, y Rocky Linux 8 y 9) se aplica a las revisiones aprobadas de acuerdo con las siguientes reglas:

   Situación 1: se excluyen las actualizaciones no relacionadas con la seguridad

   - Se aplica a: líneas de base de revisiones predeterminadas y predefinidas proporcionadas por AWS y líneas de base de revisiones personalizadas.
   - Casilla de verificación Incluir actualizaciones no relacionadas con la seguridad: no está seleccionada.

- Revisiones aplicadas: las revisiones especificadas en `updateinfo.xml` (solo actualizaciones de seguridad) se aplican únicamente si coinciden con la configuración de la línea de base de revisiones y se encuentran en los repositorios configurados.

  En algunos casos, es posible que una revisión especificada en `updateinfo.xml` ya no esté disponible en un repositorio configurado. Los repositorios configurados suelen tener solo la última versión de una revisión, que es una recopilación acumulativa de todas las actualizaciones anteriores, pero es posible que la última versión no cumpla con las reglas de la línea de base de revisiones y se omita en la operación de aplicación de revisiones.

- Comandos: en RHEL 7, el comando yum equivalente para este flujo de trabajo es:

  ```
  sudo yum update-minimal --sec-severity=Critical,Important --bugfix -y
  ```

  En AlmaLinux, RHEL 8 y Rocky Linux, los comandos dnf equivalentes para este flujo de trabajo son los siguientes:

  ```
  sudo dnf update-minimal --sec-severity=Critical --bugfix -y ; \
  sudo dnf update-minimal --sec-severity=Important --bugfix -y
  ```

  > **Note**
  >
  > En el caso de AlmaLinux, RHEL, y Rocky LinuxRocky Linux, es posible que Patch Manager instale versiones de dependencias transitivas distintas a las que instalan los comandos de `dnf` equivalentes. Las dependencias transitivas son paquetes que se instalan automáticamente para satisfacer los requisitos de otros paquetes (dependencias de dependencias).
  > Por ejemplo, `dnf upgrade-minimal --security` instala las versiones mínimas de las dependencias transitivas necesarias para resolver los problemas de seguridad conocidos, mientras que Patch Manager instala las últimas versiones disponibles de las mismas dependencias transitivas.

Situación 2: se incluyen actualizaciones no relacionadas con la seguridad

- Se aplica a: líneas de base de revisiones personalizadas.
- Casilla de verificación Incluir actualizaciones no relacionadas con la seguridad: está seleccionada.

- Revisiones aplicadas: se aplican las revisiones incluidas en `updateinfo.xml` y las que no están en `updateinfo.xml` (actualizaciones de seguridad y no relacionadas con la seguridad).
- Comandos: en RHEL 7, el comando yum equivalente para este flujo de trabajo es:

```
sudo yum update --security --bugfix -y
```

En AlmaLinux 8 y 9, RHEL 8 y 9 y Rocky Linux 8 y 9, el comando dnf equivalente para este flujo de trabajo es el siguiente:

```
sudo dnf update --security --bugfix -y
```

> **Note**
>
> Los paquetes nuevos que sustituyen a los ya obsoletos con nombres diferentes se instalan si ejecuta estos comandos `yum` o `dnf` fuera del Patch Manager. Sin embargo, no se instalan mediante las operaciones equivalentes del Patch Manager.

8. El nodo administrado se reinicia si las actualizaciones se instalaron. (Excepción: si el parámetro `RebootOption` se configura en `NoReboot` en el documento `AWS-RunPatchBaseline`, el nodo administrado no se reinicia una vez que se ejecuta Patch Manager. Para obtener más información, consulte Nombre del parámetro: `RebootOption`).

SLES

En los nodos administrados de SUSE Linux Enterprise Server (SLES), el flujo de trabajo de instalación de revisiones es el siguiente:

1. Si se especifica una lista de parches mediante una URL de https o una URL de tipo ruta de Amazon Simple Storage Service (Amazon S3) con el parámetro `InstallOverrideList` para los documentos `AWS-RunPatchBaseline` o `AWS-RunPatchBaselineAssociation`, se instalan los parches de la lista y se omiten los pasos comprendidos entre el 2 y el 7.
2. Aplique GlobalFilters tal y como se especifica en la base de referencia de parches, de modo que se mantendrán los paquetes cualificados para poder procesarse más adelante.
3. Aplique ApprovalRules tal y como se especifica en la base de referencia de parches. Cada regla de aprobación puede definir un paquete como aprobado.

Sin embargo, las reglas de aprobación también están sujetas a si se seleccionó la casilla de verificación Include nonsecurity updates (Incluir actualizaciones no relacionadas con la seguridad) durante la creación o la última actualización de una base de referencia de parches.

Además, si se excluyen las actualizaciones no relacionadas con la seguridad, se aplica una regla implícita para seleccionar únicamente los paquetes con actualizaciones en repositorios de seguridad. Para cada paquete, la versión candidata del paquete (que suele ser la última) debe formar parte de un repositorio de seguridad.

Si se incluyen actualizaciones no relacionadas con la seguridad, también se tienen en cuenta los parches de los demás repositorios.

4. Aplique ApprovedPatches tal y como se especifica en la base de referencia de parches. Las revisiones aprobadas se aprueban para su actualización, incluso si se descartan por GlobalFilters o si ninguna regla de aprobación especificada en ApprovalRules les concede la aprobación.
5. Aplique RejectedPatches tal y como se especifica en la base de referencia de parches. Los parches rechazados se eliminan de la lista de parches aprobados y no se aplicarán.
6. Si hay varias versiones de un parche aprobadas, se aplica la última.
7. La API de actualización de Zypper se aplica a los parches aprobados.
8. El nodo administrado se reinicia si las actualizaciones se instalaron. (Excepción: si el parámetro `RebootOption` se configura en `NoReboot` en el documento `AWS-RunPatchBaseline`, el nodo administrado no se reinicia una vez que se ejecuta Patch Manager. Para obtener más información, consulte Nombre del parámetro: `RebootOption`).

Servidor Ubuntu

En los nodos administrados Ubuntu Server, el flujo de trabajo de instalación de revisiones es el siguiente:

1. Si se especifica una lista de parches mediante una URL de https o una URL de tipo ruta de Amazon Simple Storage Service (Amazon S3) con el parámetro `InstallOverrideList` para los documentos `AWS-RunPatchBaseline` o `AWS-RunPatchBaselineAssociation`, se instalan los parches de la lista y se omiten los pasos comprendidos entre el 2 y el 7.
2. Si está disponible una actualización para `python3-apt` (una interfaz de biblioteca de Python para `libapt`), se actualiza a la versión más reciente. (Este paquete no relacionado con la

seguridad se actualiza aunque no haya seleccionado la opción Include nonsecurity updates [Incluir actualizaciones no relacionadas con la seguridad]).

3. Aplique GlobalFilters tal y como se especifica en la base de referencia de parches, de modo que se mantendrán los paquetes cualificados para poder procesarse más adelante.
4. Aplique ApprovalRules tal y como se especifica en la base de referencia de parches. Cada regla de aprobación puede definir un paquete como aprobado.

> **Note**
>
> Debido a que no es posible determinar de forma fiable las fechas de lanzamiento de los paquetes de actualización para Ubuntu Server, las opciones de aprobación automática no son compatibles con este sistema operativo.

Sin embargo, las reglas de aprobación también están sujetas a si se seleccionó la casilla de verificación Include nonsecurity updates (Incluir actualizaciones no relacionadas con la seguridad) durante la creación o la última actualización de una base de referencia de parches.

Además, si se excluyen las actualizaciones no relacionadas con la seguridad, se aplica una regla implícita para seleccionar únicamente los paquetes con actualizaciones en repositorios de seguridad. Para cada paquete, la versión candidata del paquete (que suele ser la última) debe formar parte de un repositorio de seguridad.

Si se incluyen actualizaciones no relacionadas con la seguridad, también se tienen en cuenta los parches de los demás repositorios.

Sin embargo, las reglas de aprobación también están sujetas a si se seleccionó la casilla de verificación Include nonsecurity updates (Incluir actualizaciones no relacionadas con la seguridad) durante la creación o la última actualización de una base de referencia de parches.

> **Note**
>
> Para cada versión de Ubuntu Server, las versiones candidatas a parches se limitan a los parches que forman parte del repositorio asociado a esa versión, como se muestra a continuación:
>
> - 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 Server 23.04 (`lunar-security`)
- Ubuntu Server 23.10 (`mantic-security`)
- Ubuntu Server 24.04 LTS (`noble-security`)
- Ubuntu Server 24.10 (`oracular-security`)
- Ubuntu Server 25.04 (`plucky-security`)

5. Aplique ApprovedPatches tal y como se especifica en la base de referencia de parches. Las revisiones aprobadas se aprueban para su actualización, incluso si se descartan por GlobalFilters o si ninguna regla de aprobación especificada en ApprovalRules les concede la aprobación.
6. Aplique RejectedPatches tal y como se especifica en la base de referencia de parches. Los parches rechazados se eliminan de la lista de parches aprobados y no se aplicarán.
7. La biblioteca de APT se utiliza para actualizar paquetes.

> **Note**
>
> Patch Manager no admite el uso de la opción `Pin-Priority` de APT para asignar prioridades a los paquetes. Patch Manager agrega las actualizaciones disponibles de todos los repositorios habilitados y selecciona la actualización más reciente que coincide con la línea de base de cada paquete instalado.

8. El nodo administrado se reinicia si las actualizaciones se instalaron. (Excepción: si el parámetro `RebootOption` se configura en `NoReboot` en el documento `AWS-RunPatchBaseline`, el nodo administrado no se reinicia una vez que se ejecuta Patch Manager. Para obtener más información, consulte Nombre del parámetro: `RebootOption`).

Windows Server

Cuando se realiza una operación de aplicación de revisiones en un nodo administrado de Windows Server, este solicita una instantánea de la base de referencia de revisiones adecuada a Systems Manager. Esta instantánea contiene la lista de todas las actualizaciones disponibles

en la base de referencia de parches que se aprobaron para su implementación. Esta lista de actualizaciones se envía a la API de Windows Update, que determina qué actualizaciones son aplicables al nodo administrado y se instala cuando sea necesario. Windows solo permite instalar la versión más reciente disponible de una KB. Patch Manager instala la última versión de una KB cuando esta, o cualquier versión anterior de la KB, coincide con la línea de base de revisiones aplicada. Si se instalan las actualizaciones, el nodo administrado se reinicia después, tantas veces como sea necesario para completar todas las revisiones necesarias. (Excepción: si el parámetro `RebootOption` se configura en `NoReboot` en el documento `AWS-RunPatchBaseline`, el nodo administrado no se reinicia una vez que se ejecuta Patch Manager. Para obtener más información, consulte Nombre del parámetro: `RebootOption`). El resumen de la operación de aplicación de parches se puede encontrar en la salida de la solicitud de Run Command. Puede encontrar más registros en el nodo administrado de la carpeta `%PROGRAMDATA%\Amazon\PatchBaselineOperations\Logs`.

Como la API de Windows Update se utiliza para descargar e instalar KB, se respetan todos los ajustes de la política de grupos de Windows Update. No se necesitan ajustes de la política de grupos para utilizar Patch Manager, pero se aplicará la configuración que haya definido, por ejemplo, dirigir nodos administrados a un servidor Windows Server Update Services (WSUS).

> **Note**
>
> De forma predeterminada, Windows descarga todos los KB del sitio de Windows Update de Microsoft porque Patch Manager utiliza la API de Windows Update para impulsar la descarga e instalar los parches. Como resultado, el nodo administrado debe poder acceder al sitio de Microsoft Windows Update; si no, la aplicación de revisiones no se realizará correctamente. De forma alternativa, puede configurar un servidor WSUS para que funcione como un repositorio de KB y configurar los nodos administrados para que se dirijan al servidor WSUS mediante las políticas de grupo.
> Patch Manager puede hacer referencia a los ID de KB al crear líneas de base de revisiones personalizadas para Windows Server, por ejemplo, cuando se incluye una lista de revisiones aprobadas o una lista de revisiones rechazadas en la configuración de línea de base. Patch Manager solo instala las actualizaciones a las que se les asigna un ID de KB en Microsoft Windows Update o en un servidor WSUS. Las actualizaciones que carecen de un ID de KB no se incluyen en las operaciones de aplicación de revisiones.
> Para obtener información sobre la creación de líneas de base de revisiones personalizadas, consulte los temas siguientes:
>
> - Creación de una línea de base de revisiones personalizada para Windows Server

- Creación de una línea de base de revisiones de revisiones (CLI)
- Formatos de nombre de paquete para sistemas operativos Windows

Funcionamiento de las reglas de bases de referencia de parches en los sistemas basados en Linux

Las reglas de una base de referencia de parches para distribuciones de Linux funcionan de forma diferente en función del tipo de distribución. A diferencia de actualizaciones de revisiones en los nodos administrados de Windows Server, las reglas se evalúan en cada nodo para tener en cuenta los repositorios configurados en la instancia. Patch Manager, una herramienta de AWS Systems Manager, utiliza el administrador de paquetes nativo para impulsar la instalación de revisiones aprobadas por la línea de base de revisiones.

Para los tipos de sistemas operativos basados en Linux que informan de un nivel de gravedad de las revisiones, Patch Manager utiliza el nivel de gravedad notificado por el editor de software para el aviso de actualización o la revisión individual. Patch Manager no deriva los niveles de gravedad de orígenes de terceros, como el Sistema de puntuación de vulnerabilidades comunes (CVSS), ni de métricas publicadas por la Base de datos nacional de vulnerabilidades (NVD).

Temas

- Funcionamiento las reglas de la línea de base de revisiones en Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 y Amazon Linux 2023
- Funcionamiento de las reglas de la línea de base de revisiones en CentOS y CentOS Stream
- Funcionamiento de las reglas de bases de referencia de parches en Debian Server y Raspberry Pi OS
- Funcionamiento de las reglas de bases de referencia de parches en macOS
- Funcionamiento de las reglas de bases de referencia de parches en Oracle Linux
- Funcionamiento de las reglas de líneas de base de revisiones en AlmaLinux, RHEL y Rocky Linux
- Funcionamiento de las reglas de bases de referencia de parches en SUSE Linux Enterprise Server
- Funcionamiento de las reglas de bases de referencia de parches en Ubuntu Server

Funcionamiento las reglas de la línea de base de revisiones en Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 y Amazon Linux 2023

> **Note**
>
> Amazon Linux 2023 (AL2023) usa repositorios versionados que se pueden bloquear en una versión específica mediante una o más configuraciones del sistema. Para todas las operaciones de aplicación de parches en las instancias de EC2 de AL2023, Patch Manager usa las versiones más recientes del repositorio, independientemente de la configuración del sistema. Para obtener información, consulte Deterministic upgrades through versioned repositories en la Guía del usuario de Amazon Linux 2023.

En Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 y Amazon Linux 2023, el proceso de selección de parches es el siguiente:

1. En el nodo administrado, la biblioteca YUM (Amazon Linux 1, Amazon Linux 2) o la biblioteca DNF (Amazon Linux 2022 y Amazon Linux 2023) accede al archivo `updateinfo.xml` de cada repositorio configurado.

   Si no encuentra el archivo `updateinfo.xml`, la instalación de las revisiones variará en función de la configuración de la opción Incluir actualizaciones no relacionadas con la seguridad y Aprobación automática. Por ejemplo, si se permiten las actualizaciones no relacionadas con la seguridad, se instalarán en el momento en que se realice la aprobación automática.
2. Cada aviso de actualización de `updateinfo.xml` incluye varios atributos que denota las propiedades de los paquetes del aviso, tal y como se describe en la siguiente tabla.

   Atributos de aviso de actualización

Atributo	Descripción
type	Corresponde al valor del atributo clave Classification en el tipo de datos PatchFilter de la base de referencia de parches. Denota el tipo de paquete incluido en el aviso de actualización. Puede consultar la lista de valores admitidos mediante el comando describe-patch-

Atributo	Descripción
	properties de la AWS CLI o la operación DescribePatchProperties de la API. También puede consultar la lista en el área Approval rules (Reglas de aprobación) de la página Create patch baseline (Crear una base de referencia de parches) o Edit patch baseline (Editar una base de referencia de parches) de la consola de Systems Manager.
severity	Corresponde al valor del atributo clave Severity en el tipo de datos PatchFilter de la base de referencia de parches. Denota la gravedad de los paquetes incluidos en el aviso de actualización. Solo se suele aplicar para los avisos de actualización Security. Puede consultar la lista de valores admitidos mediante el comando describe-patch-properties de la AWS CLI o la operación DescribePatchProperties de la API. También puede consultar la lista en el área Approval rules (Reglas de aprobación) de la página Create patch baseline (Crear una base de referencia de parches) o Edit patch baseline (Editar una base de referencia de parches) de la consola de Systems Manager.
update_id	Denota el ID de Advisory, como ALAS-2017-867. El ID de Advisory se puede utilizar en los atributos ApprovedPatches o RejectedPatches en la base de referencia de parches.

Atributo	Descripción
references	Contiene información adicional acerca del aviso de actualización, como un ID de CVE (formato: cVE-2017-1234567). El ID de CVE se puede utilizar en los atributos ApprovedPatches o RejectedPatches en la base de referencia de parches.
updated	Corresponde a ApproveAfterDays en la base de referencia de parches. Denota la fecha de publicación (fecha de actualización) de los paquetes incluidos en el aviso de actualización. Una comparación entre la marca de tiempo actual y el valor de este atributo. Además, `ApproveAfterDays` se utiliza para determinar si la revisión está aprobada para la implementación.

Para obtener información acerca de los formatos aceptados para las Listas de revisiones aprobados y rechazados, consulte Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas.

3. El producto del nodo administrado se determina en función del SSM Agent. Este atributo corresponde al valor del atributo clave Product (Producto) del tipo de datos PatchFilter de la base de referencia de parches.
4. Los paquetes se seleccionan para la actualización de acuerdo con las siguientes directrices:

Opción de seguridad	Selección de parches
Líneas de base de revisiones predeterminadas y predefinidas proporcionadas por AWS y líneas de base de revisiones personalizadas en las que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad no está seleccionada	Para cada aviso de actualización de `updateinfo.xml` ; la base de referencia de parches se usa como filtro, de modo que solo se permiten que los paquetes cualificados se incluyan en la actualización. Si varios paquetes son aplicables después

Opción de seguridad	Selección de parches

	de la aplicación de la definición de bases de referencia de parches, se usa la más reciente.
	
	En Amazon Linux 1 y Amazon Linux 2, el comando YUM equivalente para este flujo de trabajo es el siguiente:
	
	`sudo yum update-minimal --sec-severity=Critical,Important --bugfix -y`
	
	En Amazon Linux 2022 y Amazon Linux 2023, el comando dnf equivalente para este flujo de trabajo es el siguiente:
	
	`sudo dnf upgrade-minimal --sec-severity=Critical --sec-severity=Important --bugfix -y`

Opción de seguridad	Selección de parches

Líneas de base de revisiones personalizadas en las que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad sí está seleccionada con una lista de GRAVEDAD de `[Critical, Important]` y una lista de CLASIFICACIÓN de `[Security, Bugfix]`	Además de aplicar las actualizaciones de seguridad que se han seleccionado desde `updateinfo.xml` , Patch Manager aplicará actualizaciones no relacionadas con la seguridad y que, por otro lado, cumplen las reglas de filtrado de parches.
	
	En Amazon Linux y Amazon Linux 2, el comando yum equivalente para este flujo de trabajo es el siguiente:
	
	`sudo yum update --security --sec-severity=Critical,Important --bugfix -y`
	
	En Amazon Linux 2022 y Amazon Linux 2023, el comando dnf equivalente para este flujo de trabajo es el siguiente:
	
	`sudo dnf upgrade --security --sec-severity=Critical --sec-severity=Important --bugfix -y`
	
	Note
	Los paquetes nuevos que sustituyen a los ya obsoletos con nombres diferentes se instalan si ejecuta estos comandos `yum` o `dnf` fuera del Patch Manager. Sin embargo, no se instalan mediante las operaciones equivalentes del Patch Manager.

Para obtener información sobre los valores de estado de conformidad de parches, consulte Valores de estado de conformidad de las revisiones.

Funcionamiento de las reglas de la línea de base de revisiones en CentOS y CentOS Stream

Los repositorios predeterminados de CentOS y CentOS Stream no incluyen un archivo `updateinfo.xml`. Sin embargo, los repositorios personalizados que se cree o se utilicen pueden incluir este archivo. En este tema, las referencias a `updateinfo.xml` se aplicarán únicamente a estos repositorios personalizados.

En CentOS y CentOS Stream, el proceso de selección de revisiones es el siguiente:

1. En el nodo administrado, la biblioteca YUM (en las versiones CentOS 6.x y 7.x) o la biblioteca DNF (en CentOS 8.x y CentOS Stream) obtiene acceso al archivo `updateinfo.xml`, si existe en un repositorio personalizado, correspondiente a cada uno de los repositorios configurados.

   Si no encuentra el archivo `updateinfo.xml`, el cual incluye los repositorios predeterminados, la instalación de los parches variará según la configuración de la opción Incluir actualizaciones no relacionadas con la seguridad y Aprobación automática. Por ejemplo, si se permiten las actualizaciones no relacionadas con la seguridad, se instalarán en el momento en que se realice la aprobación automática.
2. Si `updateinfo.xml` está presente, cada aviso de actualización en el archivo incluye varios atributos que denota las propiedades de los paquetes del aviso, tal y como se describe en la siguiente tabla.

   Atributos de aviso de actualización

Atributo	Descripción
type	Corresponde al valor del atributo clave Classification en el tipo de datos PatchFilter de la base de referencia de parches. Denota el tipo de paquete incluido en el aviso de actualización. Puede consultar la lista de valores admitidos mediante el comando describe-patch-properties de la AWS CLI o la operación DescribePatchProperties de la API. También puede consultar la lista en el área Approval

Atributo	Descripción
	rules (Reglas de aprobación) de la página Create patch baseline (Crear una base de referencia de parches) o Edit patch baseline (Editar una base de referencia de parches) de la consola de Systems Manager.
severity	Corresponde al valor del atributo clave Severity en el tipo de datos PatchFilter de la base de referencia de parches. Denota la gravedad de los paquetes incluidos en el aviso de actualización. Solo se suele aplicar para los avisos de actualización Security. Puede consultar la lista de valores admitidos mediante el comando describe-patch-properties de la AWS CLI o la operación DescribePatchProperties de la API. También puede consultar la lista en el área Approval rules (Reglas de aprobación) de la página Create patch baseline (Crear una base de referencia de parches) o Edit patch baseline (Editar una base de referencia de parches) de la consola de Systems Manager.
update_id	Denota el ID de Advisory, como CVE-2019-17055. El ID de Advisory se puede utilizar en los atributos ApprovedPatches o RejectedPatches en la base de referencia de parches.
references	Contiene información adicional acerca del aviso de actualización, como un ID de CVE (formato: cVE-2019-17055) o un ID de Bugzilla (formato: 1463241). El ID de CVE y de Bugzilla se pueden utilizar en los atributos ApprovedPatches o RejectedPatches en la base de referencia de parches.

Atributo	Descripción
updated	Corresponde a ApproveAfterDays en la base de referencia de parches. Denota la fecha de publicación (fecha de actualización) de los paquetes incluidos en el aviso de actualización. Una comparación entre la marca de tiempo actual y el valor de este atributo. Además, `ApproveAfterDays` se utiliza para determinar si la revisión está aprobada para la implementación.

Para obtener información acerca de los formatos aceptados para las Listas de revisiones aprobados y rechazados, consulte Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas.

3. En todos los casos, el producto del nodo administrado se determina según el SSM Agent. Este atributo corresponde al valor del atributo clave Product (Producto) del tipo de datos PatchFilter de la base de referencia de parches.
4. Los paquetes se seleccionan para la actualización de acuerdo con las siguientes directrices:

Opción de seguridad	Selección de parches
Líneas de base de revisiones predeterminadas y predefinidas proporcionadas por AWS y líneas de base de revisiones personalizadas en las que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad no está seleccionada	Para cada aviso de actualización de `updateinfo.xml` , si existe un repositorio personalizado, la base de referencia de parches se utiliza como un filtro, de modo que solo se permiten que los paquetes cualificados se incluyan en la actualización. Si varios paquetes son aplicables después de la aplicación de la definición de bases de referencia de parches, se usa la más reciente. En CentOS 6 y 7 donde `updateinfo.xml` esté presente, el comando YUM equivalente para este flujo de trabajo es el siguiente:

Opción de seguridad	Selección de parches

	`sudo yum update-minimal --sec-severity=Critical,Important --bugfix -y`
	
	En CentOS 8 y CentOS Stream donde `updateinfo.xml` esté presente, el comando YUM equivalente para este flujo de trabajo es el siguiente:
	
	`sudo dnf upgrade-minimal --sec-severity=Critical --sec-severity=Important --bugfix -y`

Opción de seguridad	Selección de parches

Líneas de base de revisiones personalizadas en las que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad sí está seleccionada con una lista de GRAVEDAD de `[Critical, Important]` y una lista de CLASIFICACIÓN de `[Security, Bugfix]`	Además de aplicar las actualizaciones de seguridad que se han seleccionado desde `updateinfo.xml`, si existe un repositorio personalizado, Patch Manager aplicará actualizaciones no relacionadas con la seguridad y que, por otro lado, cumplen las reglas de filtrado de parches.
	
	En CentOS 6 y 7 donde `updateinfo.xml` esté presente, el comando YUM equivalente para este flujo de trabajo es el siguiente:
	
	`sudo yum update --sec-severity=Critical,Important --bugfix -y`
	
	En CentOS 8 y CentOS Stream donde `updateinfo.xml` esté presente, el comando YUM equivalente para este flujo de trabajo es el siguiente:
	
	`sudo dnf upgrade --security --sec-severity=Critical --sec-severity=Important --bugfix -y`
	
	Para los repositorios predeterminados y los repositorios personalizados sin `updateinfo.xml`, debe seleccionar la casilla de verificación Incluir actualizaciones que no sean de seguridad para actualizar los paquetes del sistema operativo (SO).
	
	Note
	Los paquetes nuevos que sustituyen a los ya obsoletos con nombres diferentes se instalan si ejecuta estos

Opción de seguridad	Selección de parches
	comandos `yum` o `dnf` fuera del Patch Manager. Sin embargo, no se instalan mediante las operaciones equivalentes del Patch Manager.

Para obtener información sobre los valores de estado de conformidad de parches, consulte Valores de estado de conformidad de las revisiones.

Funcionamiento de las reglas de bases de referencia de parches en Debian Server y Raspberry Pi OS

En Debian Server y Raspberry Pi OS (anteriormente Raspbian), el servicio de bases de referencia de revisiones permite filtrar en los campos Priority (Prioridad) y Section (Sección). Estos campos suelen estar presentes para todos los paquetes de Debian Server y Raspberry Pi OS. Para determinar si un parche se ha seleccionado mediante la base de referencia de parches, Patch Manager hace lo siguiente:

1. En sistemas Debian Server y Raspberry Pi OS, el equivalente de `sudo apt-get update` es ejecutar para actualizar la lista de paquetes disponibles. Los repositorios no están configurados y los datos se obtienen de los repositorios configurados en una lista de `sources`.
2. Si está disponible una actualización para `python3-apt` (una interfaz de biblioteca de Python para `libapt`), se actualiza a la versión más reciente. (Este paquete no relacionado con la seguridad se actualiza aunque no haya seleccionado la opción Include nonsecurity updates [Incluir actualizaciones no relacionadas con la seguridad]).

> **Important**
>
> Solo en Debian Server 8: debido a que los sistemas operativos Debian Server 8.* hacen referencia a un repositorio de paquetes obsoleto (`jessie-backports`), Patch Manager lleva a cabo los siguientes pasos adicionales para garantizar que las operaciones de aplicación de revisiones se efectúen correctamente:
>
> a. En el nodo administrado, la referencia al repositorio `jessie-backports` se indica en la lista de ubicaciones de origen (`/etc/apt/sources.list.d/jessie-backports`). Por consiguiente, no se intenta descargar los parches desde esa ubicación.

b. Se importa una clave de firma de actualización de seguridad de Stretch. Esta clave proporciona los permisos necesarios para las operaciones de actualización e instalación en las distribuciones de Debian Server 8.*.

c. A esta altura, se ejecuta la operación `apt-get` para asegurarse de que la versión más reciente de `python3-apt` esté instalada antes de que comience el proceso de aplicación de parches.

d. Una vez finalizado el proceso de instalación, se restaura la referencia al repositorio `jessie-backports` y se elimina la clave de firma del conjunto de claves de las fuentes de apt. Esto se hace para dejar la configuración del sistema tal como estaba antes de la operación de aplicación de parches.

3. Después, se aplican las listas de GlobalFilters, ApprovalRules, ApprovedPatches y RejectedPatches.

> **Note**
>
> Como no es posible determinar de forma fiable las fechas de lanzamiento de los paquetes de actualización para Debian Server, las opciones de aprobación automática no son compatibles con este sistema operativo.

Sin embargo, las reglas de aprobación también están sujetas a si se seleccionó la casilla de verificación Include nonsecurity updates (Incluir actualizaciones no relacionadas con la seguridad) durante la creación o la última actualización de una base de referencia de parches.

Además, si se excluyen las actualizaciones no relacionadas con la seguridad, se aplica una regla implícita para seleccionar únicamente los paquetes con actualizaciones en repositorios de seguridad. Para cada paquete, la versión candidata del paquete (que suele ser la última) debe formar parte de un repositorio de seguridad. En este caso, en Debian Server, las versiones candidatas a revisiones se limitan a las revisiones incluidas en los siguientes repositorios:

Estos repositorios se denominan de la siguiente manera:

- Debian Server 8: `debian-security jessie`
- Debian Server y Raspberry Pi OS 9: `debian-security stretch`
- Debian Server 10: `debian-security buster`
- Debian Server 11: `debian-security bullseye`

- Debian Server 12: `debian-security bookworm`

Si se incluyen actualizaciones no relacionadas con la seguridad, también se tienen en cuenta los parches de los demás repositorios.

Para obtener información acerca de los formatos aceptados para las listas de parches aprobados y rechazados, consulte Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas.

Para ver el contenido de los campos Priority y Section , ejecute el siguiente comando `aptitude`:

> **Note**
>
> Es posible que necesite instalar por primera vez Aptitude en sistemas Debian Server.

```
aptitude search -F '%p %P %s %t %V#' '~U'
```

En la respuesta a este comando, todos los paquetes actualizables se notifican en este formato:

```
name, priority, section, archive, candidate version
```

Para obtener información sobre los valores de estado de conformidad de parches, consulte Valores de estado de conformidad de las revisiones.

Funcionamiento de las reglas de bases de referencia de parches en macOS

En las instancias de macOS, el proceso de selección de parches es el siguiente:

1. En el nodo administrado, Patch Manager accede al contenido analizado del archivo `InstallHistory.plist` e identifica los nombres y las versiones de los paquetes.

   Para obtener más detalles acerca del proceso de análisis, consulte la sección macOS en Cómo se instalan las revisiones.
2. El producto del nodo administrado se determina en función del SSM Agent. Este atributo corresponde al valor del atributo clave Product (Producto) del tipo de datos PatchFilter de la base de referencia de parches.
3. Los paquetes se seleccionan para la actualización de acuerdo con las siguientes directrices:

Opción de seguridad	Selección de parches
Líneas de base de revisiones predeterminadas y predefinidas proporcionadas por AWS y líneas de base de revisiones personalizadas en las que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad no está seleccionada	En cada actualización de paquetes disponibles, la base de referencia de parches se usa como filtro, de modo que solo se incluyen en la actualización los paquetes aplicables. Si varios paquetes son aplicables después de la aplicación de la definición de bases de referencia de parches, se usa la más reciente.
Líneas de base de revisiones personalizados en las que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad está seleccionada	Además de aplicar las actualizaciones de seguridad que se han identificado mediante `InstallHistory.plist` , Patch Manager aplicará actualizaciones no relacionadas con la seguridad que además cumplen las reglas de filtrado de parches.

Para obtener información sobre los valores de estado de conformidad de parches, consulte Valores de estado de conformidad de las revisiones.

Funcionamiento de las reglas de bases de referencia de parches en Oracle Linux

En las instancias de Oracle Linux, el proceso de selección de parches es el siguiente:

1. En el nodo administrado, la biblioteca YUM accede al archivo `updateinfo.xml` de cada repositorio configurado.

> **Note**
>
> El archivo `updateinfo.xml` podría no estar disponible si Oracle no administra el repositorio. Si no encuentra el archivo `updateinfo.xml`, la instalación de las revisiones variará en función de la configuración de la opción Incluir actualizaciones no relacionadas con la seguridad y Aprobación automática. Por ejemplo, si se permiten las actualizaciones no relacionadas con la seguridad, se instalarán en el momento en que se realice la aprobación automática.

2. Cada aviso de actualización de `updateinfo.xml` incluye varios atributos que denota las propiedades de los paquetes del aviso, tal y como se describe en la siguiente tabla.

   Atributos de aviso de actualización

Atributo	Descripción
type	Corresponde al valor del atributo clave Classification en el tipo de datos PatchFilter de la base de referencia de parches. Denota el tipo de paquete incluido en el aviso de actualización. Puede consultar la lista de valores admitidos mediante el comando describe-patch-properties de la AWS CLI o la operación DescribePatchProperties de la API. También puede consultar la lista en el área Approval rules (Reglas de aprobación) de la página Create patch baseline (Crear una base de referencia de parches) o Edit patch baseline (Editar una base de referencia de parches) de la consola de Systems Manager.

Atributo	Descripción
severity	Corresponde al valor del atributo clave Severity en el tipo de datos PatchFilter de la base de referencia de parches. Denota la gravedad de los paquetes incluidos en el aviso de actualización. Solo se suele aplicar para los avisos de actualización Security. Puede consultar la lista de valores admitidos mediante el comando describe-patch-properties de la AWS CLI o la operación DescribePatchProperties de la API. También puede consultar la lista en el área Approval rules (Reglas de aprobación) de la página Create patch baseline (Crear una base de referencia de parches) o Edit patch baseline (Editar una base de referencia de parches) de la consola de Systems Manager.
update_id	Denota el ID de Advisory, como CVE-2019-17055. El ID de Advisory se puede utilizar en los atributos ApprovedPatches o RejectedPatches en la base de referencia de parches.
references	Contiene información adicional acerca del aviso de actualización, como un ID de CVE (formato: cVE-2019-17055) o un ID de Bugzilla (formato: 1463241). El ID de CVE y de Bugzilla se pueden utilizar en los atributos ApprovedPatches o RejectedPatches en la base de referencia de parches.

Atributo	Descripción
updated	Corresponde a ApproveAfterDays en la base de referencia de parches. Denota la fecha de publicación (fecha de actualización) de los paquetes incluidos en el aviso de actualización. Una comparación entre la marca de tiempo actual y el valor de este atributo. Además, `ApproveAfterDays` se utiliza para determinar si la revisión está aprobada para la implementación.

Para obtener información acerca de los formatos aceptados para las Listas de revisiones aprobados y rechazados, consulte Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas.

3. El producto del nodo administrado se determina en función del SSM Agent. Este atributo corresponde al valor del atributo clave Product (Producto) del tipo de datos PatchFilter de la base de referencia de parches.
4. Los paquetes se seleccionan para la actualización de acuerdo con las siguientes directrices:

Opción de seguridad	Selección de parches
Líneas de base de revisiones predeterminadas y predefinidas proporcionadas por AWS y líneas de base de revisiones personalizadas en las que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad no está seleccionada	Para cada aviso de actualización de `updateinfo.xml` ; la base de referencia de parches se usa como filtro, de modo que solo se permiten que los paquetes cualificados se incluyan en la actualización. Si varios paquetes son aplicables después de la aplicación de la definición de bases de referencia de parches, se usa la más reciente. En los nodos administrados de la versión 7, el comando yum equivalente para este flujo de trabajo es:

Opción de seguridad	Selección de parches

	`sudo yum update-minimal --sec-severity=Important,Moderate --bugfix -y`
	
	En los nodos administrados de las versiones 8 y 9, el comando dnf equivalente para este flujo de trabajo es el siguiente:
	
	`sudo dnf upgrade-minimal --security --sec-severity=Moderate --sec-severity=Important`

Opción de seguridad	Selección de parches

Líneas de base de revisiones personalizadas en las que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad sí está seleccionada con una lista de GRAVEDAD de `[Critical, Important]` y una lista de CLASIFICACIÓN de `[Security, Bugfix]`	Además de aplicar las actualizaciones de seguridad que se han seleccionado desde `updateinfo.xml` , Patch Manager aplicará actualizaciones no relacionadas con la seguridad y que, por otro lado, cumplen las reglas de filtrado de parches.
	
	En los nodos administrados de la versión 7, el comando yum equivalente para este flujo de trabajo es:
	
	`sudo yum update --security --sec-severity=Critical,Important --bugfix -y`
	
	En los nodos administrados de las versiones 8 y 9, el comando dnf equivalente para este flujo de trabajo es el siguiente:
	
	`sudo dnf upgrade --security --sec-severity=Critical, --sec-severity=Important --bugfix y`
	
	Note
	Los paquetes nuevos que sustituyen a los ya obsoletos con nombres diferentes se instalan si ejecuta estos comandos `yum` o `dnf` fuera del Patch Manager. Sin embargo, no se instalan mediante las operaciones equivalentes del Patch Manager.

Para obtener información sobre los valores de estado de conformidad de parches, consulte Valores de estado de conformidad de las revisiones.

Funcionamiento de las reglas de líneas de base de revisiones en AlmaLinux, RHEL y Rocky Linux

En AlmaLinux, Red Hat Enterprise Linux (RHEL) y Rocky Linux, el proceso de selección de revisiones es el siguiente:

1. En el nodo administrado, la biblioteca YUM (RHEL 7) o la biblioteca DNF (AlmaLinux 8 y 9, RHEL 8, 9 y 10, y Rocky Linux 8 y 9) accede al archivo `updateinfo.xml` de cada repositorio configurado.

   > **Note**
   >
   > El archivo `updateinfo.xml` podría no estar disponible si Red Hat no administra el repositorio. Si no se encuentra `updateinfo.xml`; no se aplica ningún parche.

2. Cada aviso de actualización de `updateinfo.xml` incluye varios atributos que denota las propiedades de los paquetes del aviso, tal y como se describe en la siguiente tabla.

   Atributos de aviso de actualización

Atributo	Descripción
type	Corresponde al valor del atributo clave Classification en el tipo de datos PatchFilter de la base de referencia de parches. Denota el tipo de paquete incluido en el aviso de actualización. Puede consultar la lista de valores admitidos mediante el comando describe-patch-properties de la AWS CLI o la operación DescribePatchProperties de la API. También puede consultar la lista en el área Approval rules (Reglas de aprobación) de la página Create patch baseline (Crear una base de referencia de parches) o Edit patch baseline

Atributo	Descripción
	(Editar una base de referencia de parches) de la consola de Systems Manager.
severity	Corresponde al valor del atributo clave Severity en el tipo de datos PatchFilter de la base de referencia de parches. Denota la gravedad de los paquetes incluidos en el aviso de actualización. Solo se suele aplicar para los avisos de actualización Security. Puede consultar la lista de valores admitidos mediante el comando describe-patch-properties de la AWS CLI o la operación DescribePatchProperties de la API. También puede consultar la lista en el área Approval rules (Reglas de aprobación) de la página Create patch baseline (Crear una base de referencia de parches) o Edit patch baseline (Editar una base de referencia de parches) de la consola de Systems Manager.
update_id	Denota el ID de Advisory, como RHSA-2017 :0864. El ID de Advisory se puede utilizar en los atributos ApprovedPatches o RejectedP atches en la base de referencia de parches.
references	Contiene información adicional acerca del aviso de actualización, como un ID de CVE (formato: cVE-2017-1000371) o un ID de Bugzilla (formato: 1463241). El ID de CVE y de Bugzilla se pueden utilizar en los atributos ApprovedPatches o RejectedPatches en la base de referencia de parches.

Atributo	Descripción
updated	Corresponde a ApproveAfterDays en la base de referencia de parches. Denota la fecha de publicación (fecha de actualización) de los paquetes incluidos en el aviso de actualización. Una comparación entre la marca de tiempo actual y el valor de este atributo. Además, `ApproveAfterDays` se utiliza para determinar si la revisión está aprobada para la implementación.

Para obtener información acerca de los formatos aceptados para las Listas de revisiones aprobados y rechazados, consulte Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas.

3. El producto del nodo administrado se determina en función del SSM Agent. Este atributo corresponde al valor del atributo clave Product (Producto) del tipo de datos PatchFilter de la base de referencia de parches.
4. Los paquetes se seleccionan para la actualización de acuerdo con las siguientes directrices:

Opción de seguridad	Selección de parches
Líneas de base de revisiones predeterminadas y predefinidas proporcionas AWS y líneas de base de revisiones en las que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad no está seleccionada para ninguna regla	Para cada aviso de actualización de `updateinfo.xml` ; la base de referencia de parches se usa como filtro, de modo que solo se permiten que los paquetes cualificados se incluyan en la actualización. Si varios paquetes son aplicables después de la aplicación de la definición de bases de referencia de parches, se usa la más reciente. En RHEL 7, el comando yum equivalente para este flujo de trabajo es:

Opción de seguridad	Selección de parches

	`sudo yum update-minimal --sec-severity=Critical,Important --bugfix -y`
	
	En AlmaLinux 8 y 9, RHEL 8, 9 y 10, y Rocky Linux 8 y 9, el comando dnf equivalente para este flujo de trabajo es el siguiente:
	
	`sudo dnf upgrade-minimal --sec-severity=Critical --sec-severity=Important --bugfix -y`

Opción de seguridad	Selección de parches

Líneas de base de revisiones personalizadas en las que la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad sí está seleccionada con una lista de GRAVEDAD de `[Critical, Important]` y una lista de CLASIFICACIÓN de `[Security, Bugfix]`	Además de aplicar las actualizaciones de seguridad que se han seleccionado desde `updateinfo.xml` , Patch Manager aplicará actualizaciones no relacionadas con la seguridad y que, por otro lado, cumplen las reglas de filtrado de parches.
	
	En RHEL 7, el comando yum equivalente para este flujo de trabajo es:
	
	`sudo yum update --security --sec-severity=Critical,Important --bugfix -y`
	
	En AlmaLinux 8 y 9, RHEL 8, 9 y 10 y Rocky Linux 8 y 9, el comando dnf equivalente para este flujo de trabajo es el siguiente:
	
	`sudo dnf upgrade --sec-severity=Critical --sec-severity=Important --bugfix -y`
	
	Note
	Los paquetes nuevos que sustituyen a los ya obsoletos con nombres diferentes se instalan si ejecuta estos comandos `yum` o `dnf` fuera del Patch Manager. Sin embargo, no se instalan mediante las operaciones equivalentes del Patch Manager.

Para obtener información sobre los valores de estado de conformidad de parches, consulte Valores de estado de conformidad de las revisiones.

Funcionamiento de las reglas de bases de referencia de parches en SUSE Linux Enterprise Server

En SLES, cada parche incluye los siguientes atributos que denotan las propiedades de los paquetes del parche:

- Category (Categoría): Corresponde al valor del atributo clave Classification (Clasificación) en el tipo de datos PatchFilter de la base de referencia de parches. Denota el tipo de parche incluido en el aviso de actualización.

  Puede consultar la lista de valores admitidos mediante el comando describe-patch-properties de la AWS CLI o la operación DescribePatchProperties de la API. También puede consultar la lista en el área Approval rules (Reglas de aprobación) de la página Create patch baseline (Crear una base de referencia de parches) o Edit patch baseline (Editar una base de referencia de parches) de la consola de Systems Manager.
- Gravedad: corresponde al valor del atributo clave Gravedad en el tipo de datos PatchFilter de la línea de base de revisiones. Denota la gravedad de los parches.

  Puede consultar la lista de valores admitidos mediante el comando describe-patch-properties de la AWS CLI o la operación DescribePatchProperties de la API. También puede consultar la lista en el área Approval rules (Reglas de aprobación) de la página Create patch baseline (Crear una base de referencia de parches) o Edit patch baseline (Editar una base de referencia de parches) de la consola de Systems Manager.

El producto del nodo administrado se determina en función del SSM Agent. Este atributo corresponde al valor del atributo clave Product (Producto) del tipo de datos PatchFilter de la base de referencia de parches.

En cada parche, la base de referencia de parches se usa como filtro, de modo que solo se incluyen en la actualización los paquetes cualificados. Si varios paquetes son aplicables después de la aplicación de la definición de bases de referencia de parches, se usa la más reciente.

Para obtener información acerca de los formatos aceptados para las listas de parches aprobados y rechazados, consulte Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas.

Funcionamiento de las reglas de bases de referencia de parches en Ubuntu Server

En Ubuntu Server, el servicio de bases de referencia de parches permite filtrar en los campos Priority (Prioridad) y Section (Sección). Estos campos suelen estar presentes para todos los paquetes de

Ubuntu Server. Para determinar si un parche se ha seleccionado mediante la base de referencia de parches, Patch Manager hace lo siguiente:

1. En sistemas Ubuntu Server, el equivalente de `sudo apt-get update` es ejecutar para actualizar la lista de paquetes disponibles. Los repositorios no están configurados y los datos se obtienen de los repositorios configurados en una lista de `sources`.
2. Si está disponible una actualización para `python3-apt` (una interfaz de biblioteca de Python para `libapt`), se actualiza a la versión más reciente. (Este paquete no relacionado con la seguridad se actualiza aunque no haya seleccionado la opción Include nonsecurity updates [Incluir actualizaciones no relacionadas con la seguridad]).
3. Después, se aplican las listas de GlobalFilters, ApprovalRules, ApprovedPatches y RejectedPatches.

   > **Note**
   >
   > Debido a que no es posible determinar de forma fiable las fechas de lanzamiento de los paquetes de actualización para Ubuntu Server, las opciones de aprobación automática no son compatibles con este sistema operativo.

   Sin embargo, las reglas de aprobación también están sujetas a si se seleccionó la casilla de verificación Include nonsecurity updates (Incluir actualizaciones no relacionadas con la seguridad) durante la creación o la última actualización de una base de referencia de parches.

   Además, si se excluyen las actualizaciones no relacionadas con la seguridad, se aplica una regla implícita para seleccionar únicamente los paquetes con actualizaciones en repositorios de seguridad. Para cada paquete, la versión candidata del paquete (que suele ser la última) debe formar parte de un repositorio de seguridad. En este caso, en Ubuntu Server, las versiones candidatas a revisiones se limitan a las revisiones incluidas en los siguientes repositorios:

   - 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 Server 23.04 (`lunar-security`)

- Ubuntu Server 23.10 (`mantic-security`)
- Ubuntu Server 24.04 LTS (`noble-security`)
- Ubuntu Server 24.10 (`oracular-security`)
- Ubuntu Server 25.04 (`plucky-security`)

Si se incluyen actualizaciones no relacionadas con la seguridad, también se tienen en cuenta los parches de los demás repositorios.

Para obtener información acerca de los formatos aceptados para las listas de parches aprobados y rechazados, consulte Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas.

Para ver el contenido de los campos Priority y Section , ejecute el siguiente comando `aptitude`:

> **Note**
>
> Es posible que necesite instalar por primera vez Aptitude en sistemas Ubuntu Server 16.

```
aptitude search -F '%p %P %s %t %V#' '~U'
```

En la respuesta a este comando, todos los paquetes actualizables se notifican en este formato:

```
name, priority, section, archive, candidate version
```

Para obtener información sobre los valores de estado de conformidad de parches, consulte Valores de estado de conformidad de las revisiones.

Diferencias en la operación de revisiones entre Linux y Windows Server

En este tema se describen las principales diferencias entre la aplicación de revisiones de Linux y Windows Server en Patch Manager, una herramienta de AWS Systems Manager.

> **Note**
>
> Para aplicar revisiones a los nodos administrados de Linux, los nodos deben ejecutar la versión de SSM Agent 2.0.834.0 o posterior.

Cada vez que se agregan herramientas nuevas a Systems Manager o se actualizan las herramientas existentes, se lanza una versión actualizada de SSM Agent. No utilizar la versión más reciente del agente puede impedir que el nodo administrado utilice diversas herramientas y características de Systems Manager. Por este motivo, se recomienda automatizar el proceso de mantener SSM Agent actualizado en los equipos. Para obtener más información, consulta Automatización de las actualizaciones de SSM Agent. Suscríbase a la página SSM Agent Release Notes en GitHub para recibir notificaciones sobre las actualizaciones de SSM Agent.

Diferencia 1: evaluación de parches

Patch Manager utiliza procesos diferentes en los nodos administrados por Windows y los nodos administrados por Linux con el fin de evaluar qué revisiones deben incluirse.

Linux

En la aplicación de revisiones de Linux, Systems Manager evalúa las reglas de bases de referencia de revisiones y la lista de las revisiones aprobadas y rechazadas en cada nodo administrado. Systems Manager debe evaluar la aplicación de revisiones en cada nodo debido a que el servicio recupera la lista de actualizaciones y revisiones conocidas a partir de los repositorios configurados en el nodo administrado.

Windows

En la aplicación de parches de Windows, Systems Manager evalúa las reglas de base de referencia de parches y la lista de los parches aprobados y rechazados directamente en el servicio. Puede hacerlo porque los parches de Windows se obtienen de un único repositorio (Windows Update).

Diferencia 2: parches **`Not Applicable`**

Debido a la gran cantidad de paquetes disponibles para los sistemas operativos Linux, Systems Manager no informa los detalles sobre parches en el estado Not Applicable (No aplicable). Un parche `Not Applicable` es, por ejemplo, un parche del software Apache cuando la instancia no tiene instalado dicho servicio. Systems Manager informa la cantidad de revisiones `Not Applicable` en el resumen, pero si llama a la API DescribeInstancePatches para un nodo administrado, los datos devueltos no incluyen revisiones cuyo estado sea `Not Applicable`. Este comportamiento es diferente en Windows.

Diferencia 3: compatibilidad con documentos SSM

El documento `AWS-ApplyPatchBaseline` de Systems Manager (documento de SSM) no admite nodos administrados de Linux. Para aplicar bases de referencia de revisiones a nodos administrados de Linux, macOS y Windows Server, el documento de SSM recomendado es `AWS-RunPatchBaseline`. Para obtener más información, consulte Documentos de comando de SSM para la aplicación de revisiones a nodos administrados y Documento de comandos SSM para aplicar revisiones: `AWS-RunPatchBaseline`.

Diferencia 4: parches de aplicación

Patch Manager se centra principalmente en aplicar parches a sistemas operativos. Sin embargo, también puede utilizar Patch Manager para aplicar revisiones a algunas aplicaciones de los nodos administrados.

Linux

En los sistemas operativos Linux, Patch Manager utiliza los repositorios configurados para actualizaciones y no distingue entre sistemas operativos y parches de aplicaciones. Puede utilizar Patch Manager para definir de qué repositorios obtendrá actualizaciones. Para obtener más información, consulte Cómo especificar un repositorio de origen de parches alternativo (Linux).

Windows

En nodos administrados de Windows Server, puede aplicar reglas de aprobación, así como excepciones de revisiones aprobadas o rechazadas, para las aplicaciones publicadas por Microsoft, como Microsoft Word 2016 o Microsoft Exchange Server 2016. Para obtener más información, consulte Uso de bases de referencia de parches personalizadas.

Diferencia 5: opciones de listas de revisiones rechazadas en líneas de base de revisiones personalizadas

Al crear una línea de base de revisiones personalizada, puede especificar una o más revisiones para una lista de revisiones rechazadas. En el caso de los nodos gestionados por Linux, también puede optar por permitir su instalación si son dependencias de otra revisión permitida por la línea de base.

Windows Server, sin embargo, no admite el concepto de dependencias de revisiones. Puede agregar una revisión a la lista de revisiones rechazadas en una línea de base personalizada para Windows Server, pero el resultado depende de (1) si la revisión rechazada ya está instalada o no en un nodo gestionado y (2) de la opción que elija para la acción de revisiones rechazadas.

Consulte la siguiente tabla para obtener más información sobre las opciones de revisiones rechazadas en Windows Server.

Estado de la instalación	Opción: “Permitir como dependencia”	Opción: “Bloque”
La revisión ya está instalada	Estado registrado: `INSTALLED_OTHER`	Estado registrado: `INSTALLED_REJECTED`
La revisión no está instalada	Revisión omitida	Revisión omitida

Cada revisión para Windows Server que Microsoft lanza normalmente contiene toda la información necesaria para que la instalación se realice correctamente. Sin embargo, en ocasiones, es posible que se requiera un paquete de requisitos previos, que debe instalarse manualmente. Patch Manager no proporciona información acerca de estos requisitos previos. Para obtener información relacionada, consulte Solución de problemas de actualización de Windows en el sitio web de Microsoft.

Documentos de comando de SSM para la aplicación de revisiones a nodos administrados

En este tema se describen los nueve documentos de Systems Manager (documentos de SSM) disponibles para que pueda mantener los nodos administrados actualizados con las últimas revisiones relacionadas con la seguridad.

Se recomienda utilizar solo cinco de estos documentos en las operaciones de aplicación de revisiones. En conjunto, estos cinco documentos de SSM le proporcionan una gama completa de opciones de aplicación de revisiones con AWS Systems Manager. Cuatro de estos documentos se publicaron después de los cuatro documentos de SSM heredados a los que sustituyen, y contienen ampliaciones o consolidaciones de funcionalidad.

Documentos de SSM recomendados para las revisiones

Recomendamos utilizar los cinco documentos de SSM a continuación en las operaciones de revisión.

- `AWS-ConfigureWindowsUpdate`
- `AWS-InstallWindowsUpdates`
- `AWS-RunPatchBaseline`
- `AWS-RunPatchBaselineAssociation`

- `AWS-RunPatchBaselineWithHooks`

Documentos de SSM heredados para las revisiones

Los cuatro documentos de SSM heredados a continuación aún están disponibles para ser utilizados en algunos Regiones de AWS, pero no están actualizados ni se puede garantizar que funcionen en todos los casos y puede que ya no cuenten con más soporte en el futuro. Recomendamos que no se utilicen en las operaciones de revisión.

- `AWS-ApplyPatchBaseline`
- `AWS-FindWindowsUpdates`
- `AWS-InstallMissingWindowsUpdates`
- `AWS-InstallSpecificWindowsUpdates`

Consulte las secciones siguientes para obtener más información sobre el uso de estos documentos de SSM en las operaciones de aplicación de revisiones.

Temas

- Documentos de SSM recomendados para la aplicación de revisiones a nodos administrados
- Documentos de SSM heredados para la aplicación de revisiones a nodos administrados
- Documento de comandos SSM para aplicar revisiones: AWS-RunPatchBaseline
- Documento de comandos SSM para aplicar revisiones: AWS-RunPatchBaselineAssociation
- Documento de comandos SSM para aplicar revisiones: AWS-RunPatchBaselineWithHooks
- Ejemplo de escenario para utilizar el parámetro InstallOverrideList en AWS-RunPatchBaseline o AWS-RunPatchBaselineAssociation
- Uso del parámetro BaselineOverride

Documentos de SSM recomendados para la aplicación de revisiones a nodos administrados

Se recomienda utilizar los siguientes cinco documentos de SSM para las operaciones de aplicación de revisiones en los nodos administrados.

Documentos de SSM recomendados

- AWS-ConfigureWindowsUpdate
- AWS-InstallWindowsUpdates

- AWS-RunPatchBaseline
- AWS-RunPatchBaselineAssociation
- AWS-RunPatchBaselineWithHooks

AWS-ConfigureWindowsUpdate

Admite configurar las funciones básicas de Windows Update y utilizarlas para instalar actualizaciones de forma automática (o para desactivar las actualizaciones automáticas). Disponible en todas las Regiones de AWS.

Este documento de SSM indica a Windows Update que descargue e instale las actualizaciones especificadas y que reinicie los nodos administrados según sea necesario. Utilice este documento con State Manager, una herramienta de AWS Systems Manager, para asegurarse de que Windows Update conserve su configuración. También puede ejecutarlo de forma manual con Run Command, una herramienta de AWS Systems Manager, para cambiar la configuración de Windows Update.

Los parámetros disponibles en este documento permiten especificar la categoría de actualizaciones que se deben instalar (o si se deben desactivar las actualizaciones automáticas), así como especificar el día de la semana y la hora del día en que se deben ejecutar las operaciones de aplicación de revisiones. Este documento de SSM es más útil si no se necesita un control estricto sobre las actualizaciones de Windows y no es necesario recopilar información de conformidad.

Documentos de SSM heredados a los que sustituye:

- Ninguna

AWS-InstallWindowsUpdates

Instala actualizaciones en un nodo administrado de Windows Server. Disponible en todas las Regiones de AWS.

Este documento de SSM proporciona la funcionalidad básica de aplicación de revisiones para los casos en los que se desea instalar una actualización específica (mediante el parámetro `Include Kbs`), o se desea instalar revisiones con clasificaciones o categorías específicas, pero no se necesita información de conformidad de revisiones.

Documentos de SSM heredados a los que sustituye:

- `AWS-FindWindowsUpdates`

- `AWS-InstallMissingWindowsUpdates`
- `AWS-InstallSpecificWindowsUpdates`

Los tres documentos heredados realizan diferentes funciones, pero se pueden alcanzar los mismos resultados mediante una configuración de parámetros distinta con el documento `AWS-InstallWindowsUpdates` de SSM más reciente. Esta configuración de parámetros se describe en Documentos de SSM heredados para la aplicación de revisiones a nodos administrados.

`AWS-RunPatchBaseline`

Instala revisiones en los nodos administrados o los examina para determinar si falta alguna revisión que sea aplicable. Disponible en todas las Regiones de AWS.

`AWS-RunPatchBaseline` le permite controlar las aprobaciones de revisiones mediante la línea de base de revisiones que se especifica como "predeterminada" para un tipo de sistema operativo. Genera informes de conformidad de revisiones que se pueden visualizar con las herramientas de conformidad de Systems Manager. Estas herramientas proporcionan información sobre el estado de conformidad de revisiones de los nodos administrados como, por ejemplo, en qué nodos faltan revisiones y cuáles son esas revisiones. Cuando utiliza `AWS-RunPatchBaseline`, la información relativa a la conformidad de revisiones se registra mediante el comando `PutInventory` de la API. En el caso de los sistemas operativos Linux, se proporciona información sobre la conformidad para las revisiones tanto del repositorio de origen predeterminado configurado en un nodo administrado como de los repositorios de origen alternativos que se especifiquen en una base de referencia de revisiones personalizada. Para obtener más información sobre los repositorios de origen alternativos, consulte Cómo especificar un repositorio de origen de parches alternativo (Linux). Para obtener más información acerca de las herramientas de conformidad de Systems Manager, consulte Conformidad de AWS Systems Manager.

Documentos heredados a los que sustituye:

- `AWS-ApplyPatchBaseline`

El documento heredado `AWS-ApplyPatchBaseline` se aplica únicamente en el caso de los nodos administrados de Windows Server y no ofrece soporte para la aplicación de revisiones. El nuevo documento `AWS-RunPatchBaseline` ofrece la misma compatibilidad para sistemas Windows y Linux. Es necesario disponer de la versión 2.0.834.0 del SSM Agent u otra posterior para poder utilizar el documento `AWS-RunPatchBaseline`.

Para obtener más información acerca del documento AWS-RunPatchBaseline de SSM, consulte Documento de comandos SSM para aplicar revisiones: AWS-RunPatchBaseline.

AWS-RunPatchBaselineAssociation

Instala revisiones en las instancias o las examina para determinar si falta alguna revisión que sea aplicable. Disponible en todas las Regiones de AWS comerciales.

AWS-RunPatchBaselineAssociation difiere de AWS-RunPatchBaseline en algunos aspectos relevantes:

- AWS-RunPatchBaselineAssociation está diseñado para que se utilice principalmente con asociaciones de State Manager creadas con Quick Setup, una herramienta de AWS Systems Manager. Especialmente cuando se utiliza el tipo de configuración Quick Setup de administración de host, si elige la opción escanear las instancias para detectar las revisiones que faltan cada día, el sistema utiliza AWS-RunPatchBaselineAssociation para efectuar la operación.

  Sin embargo, en la mayoría de los casos, a la hora de configurar sus propias operaciones de aplicación de revisiones, debe elegir AWS-RunPatchBaseline o AWS-RunPatchBaselineWithHooks en lugar de AWS-RunPatchBaselineAssociation.

  Para obtener más información, consulte los temas siguientes:

  - AWS Systems Manager Quick Setup
  - Documento de comandos SSM para aplicar revisiones: AWS-RunPatchBaselineAssociation
- AWS-RunPatchBaselineAssociation admite el uso de etiquetas que permiten identificar cuál es la línea de base de revisiones que se utilizará con un conjunto de destinos cuando se ejecute.
- Para las operaciones de aplicación de revisiones que utilizan AWS-RunPatchBaselineAssociation, los datos de conformidad de las revisiones se compilan en función de una asociación específica de State Manager. Los datos de conformidad de revisiones que se recopilan cuando se ejecuta AWS-RunPatchBaselineAssociation se registran mediante el comando PutComplianceItems de la API en lugar del comando PutInventory. Por consiguiente, se evita que se sobrescriban los datos de conformidad que no se encuentran relacionados con esta asociación en particular.

  En el caso de los sistemas operativos Linux, se proporciona información sobre la conformidad para las revisiones tanto del repositorio de origen predeterminado configurado en una instancia como de los repositorios de origen alternativos que se especifiquen en una línea de base de revisiones

personalizada. Para obtener más información sobre los repositorios de origen alternativos, consulte Cómo especificar un repositorio de origen de parches alternativo (Linux). Para obtener más información acerca de las herramientas de conformidad de Systems Manager, consulte Conformidad de AWS Systems Manager.

Documentos heredados a los que sustituye:

- Ninguna

Para obtener más información acerca del documento `AWS-RunPatchBaselineAssociation` de SSM, consulte Documento de comandos SSM para aplicar revisiones: `AWS-RunPatchBaselineAssociation`.

`AWS-RunPatchBaselineWithHooks`

Instala revisiones en los nodos administrados o analiza los nodos para determinar si falta alguna revisión aplicable, con enlaces opcionales que se pueden utilizar para ejecutar documentos de SSM en tres puntos durante el ciclo de aplicación de revisiones. Disponible en todas las Regiones de AWS comerciales. No se admite en macOS.

`AWS-RunPatchBaselineWithHooks` se diferencia de `AWS-RunPatchBaseline` en la operación `Install`.

`AWS-RunPatchBaselineWithHooks` admite enlaces de ciclo de vida que se ejecutan en puntos designados durante la aplicación de revisiones en los nodos administrados. Dado que las instalaciones de revisiones en ocasiones requieren que se reinicien los nodos administrados, la operación de aplicación de revisiones se divide en dos eventos, lo que supone un total de tres enlaces que permiten una funcionalidad personalizada. El primer enlace tiene lugar antes de la operación `Install with NoReboot`. El segundo enlace tiene lugar después de la operación `Install with NoReboot`. El tercer enlace está disponible después del reinicio del nodo.

Documentos heredados a los que sustituye:

- Ninguna

Para obtener más información acerca del documento `AWS-RunPatchBaselineWithHooks` de SSM, consulte Documento de comandos SSM para aplicar revisiones: `AWS-RunPatchBaselineWithHooks`.

Documentos de SSM heredados para la aplicación de revisiones a nodos administrados

Los cuatro documentos de SSM a continuación aún están disponibles en algunas Regiones de AWS. Sin embargo, ya no están actualizados y puede que ya no cuenten con más soporte en el futuro, por lo que recomendamos que no se utilicen. En su lugar, utilice los documentos se describe en Documentos de SSM recomendados para la aplicación de revisiones a nodos administrados.

Documentos de SSM heredados

- AWS-ApplyPatchBaseline
- AWS-FindWindowsUpdates
- AWS-InstallMissingWindowsUpdates
- AWS-InstallSpecificWindowsUpdates

`AWS-ApplyPatchBaseline`

Admite solo los nodos administrados de Windows Server, pero no incluye la compatibilidad con el uso de revisiones en aplicaciones que se encuentra en su sustitución, `AWS-RunPatchBaseline`. No está disponible en Regiones de AWS lanzadas después de agosto de 2017.

> **Note**
>
> El sustituto de este documento de SSM, `AWS-RunPatchBaseline`, requiere la versión 2.0.834.0 del SSM Agent u otra posterior. Puede utilizar el documento `AWS-UpdateSSMAgent` para actualizar los nodos administrados a la versión más reciente del agente.

`AWS-FindWindowsUpdates`

Ha sido sustituido por `AWS-InstallWindowsUpdates`, que puede realizar las mismas acciones. No está disponible en Regiones de AWS lanzadas después de abril de 2017.

Para lograr el mismo resultado que obtendría a partir de este documento de SSM heredado, utilice la siguiente configuración de parámetros con el documento de sustitución recomendado, `AWS-InstallWindowsUpdates`:

- `Action` = `Scan`
- `Allow Reboot` = `False`

AWS-InstallMissingWindowsUpdates

Ha sido sustituido por `AWS-InstallWindowsUpdates`, que puede realizar las mismas acciones. No está disponible en ninguna de las Regiones de AWS lanzadas después de abril de 2017.

Para lograr el mismo resultado que obtendría a partir de este documento de SSM heredado, utilice la siguiente configuración de parámetros con el documento de sustitución recomendado, `AWS-InstallWindowsUpdates`:

- `Action` = `Install`
- `Allow Reboot` = `True`

AWS-InstallSpecificWindowsUpdates

Ha sido sustituido por `AWS-InstallWindowsUpdates`, que puede realizar las mismas acciones. No está disponible en ninguna de las Regiones de AWS lanzadas después de abril de 2017.

Para lograr el mismo resultado que obtendría a partir de este documento de SSM heredado, utilice la siguiente configuración de parámetros con el documento de sustitución recomendado, `AWS-InstallWindowsUpdates`:

- `Action` = `Install`
- `Allow Reboot` = `True`
- `Include Kbs` = *`lista de artículos de KB separada por comas`*

Documento de comandos SSM para aplicar revisiones: **AWS-RunPatchBaseline**

AWS Systems Manager es compatible con `AWS-RunPatchBaseline`, un documento de Systems Manager (documento de SSM) para Patch Manager, una herramienta de AWS Systems Manager. Este documento de SSM realiza operaciones de aplicación de revisiones en los nodos administrados para actualizaciones relacionadas con la seguridad y de otros tipos. Cuando el documento se ejecuta, utiliza la línea de base de revisiones especificada como “predeterminada” para un tipo de sistema operativo en caso de que no se hubiera indicado ningún grupo de revisiones. En caso contrario, utiliza la línea de base de revisiones que se asocia con el grupo de revisiones. Para obtener información acerca de los grupos de revisiones, consulte Grupos de revisiones.

Puede utilizar el documento `AWS-RunPatchBaseline` para aplicar revisiones a los sistemas operativos y a las aplicaciones. (En Windows Server, la compatibilidad con las aplicaciones se limita a las actualizaciones de las aplicaciones publicadas por Microsoft).

Este documento es compatible con los nodos administrados de Linux, macOS y Windows Server. El documento se encargará de realizar las acciones adecuadas para cada plataforma.

> **Note**
>
> Además, Patch Manager es compatible con el documento de SSM heredado `AWS-ApplyPatchBaseline`. Sin embargo, este documento solo es compatible con la aplicación de revisiones en nodos administrados de Windows. Se recomienda que utilice `AWS-RunPatchBaseline` en su lugar porque es compatible con la aplicación de revisiones en los tipos de nodos administrados de Linux, macOS y Windows Server. Es necesario disponer de la versión 2.0.834.0 del SSM Agent u otra posterior para poder utilizar el documento `AWS-RunPatchBaseline`.

Windows Server

En nodos administrados de Windows Server, el documento `AWS-RunPatchBaseline` descarga e invoca un módulo de PowerShell, que a su vez descarga una instantánea de la línea de base de revisiones que se aplica al nodo administrado. Esta instantánea de la línea de base de revisiones contiene una lista de revisiones aprobadas que se compila al consultar dicha línea de base de revisiones en un servidor de Windows Server Update Services (WSUS). Esta lista se transfiere a la API de Windows Update, que controla la descarga y la instalación de las revisiones aprobadas, según proceda.

Linux

En nodos administrados de Linux, el documento `AWS-RunPatchBaseline` invoca un módulo de Python, que a su vez descarga una instantánea de la línea de base de revisiones que se aplica al nodo administrado. Esta instantánea de la línea de base de revisiones utiliza las reglas definidas y las listas de revisiones aprobadas y bloqueadas con el fin de impulsar el administrador de paquetes adecuado para cada tipo de nodo:

- Los nodos administrados de Amazon Linux 1, Amazon Linux 2, CentOS, Oracle Linux y RHEL 7 utilizan YUM. Para las operaciones de YUM, Patch Manager requiere `Python 2.6` o una versión posterior compatible (2.6 a 3.10).
- Los nodos administrados de RHEL 8 utilizan DNF. Para las operaciones de DNF, Patch Manager requiere una versión compatible de `Python 2` o `Python 3` (2.6 a 3.10). (Ninguna de las dos versiones viene instalada en RHEL 8 de forma predeterminada. Para ello, deberá instalar una u otra versión manualmente).

- Las instancias de Debian Server, Raspberry Pi OS y Ubuntu Server usan APT. Para las operaciones de APT, Patch Manager requiere una versión compatible de `Python 3` (3.0 a 3.10).
- Los nodos administrados de SUSE Linux Enterprise Server utilizan Zypper. Para las operaciones de Zypper, Patch Manager requiere `Python 2.6` o una versión posterior compatible (2.6 a 3.10).

macOS

En nodos administrados de macOS, el documento `AWS-RunPatchBaseline` invoca un módulo de Python, que a su vez descarga una instantánea de la línea de base de revisiones que se aplica al nodo administrado. A continuación, un subproceso de Python invoca la AWS Command Line Interface (AWS CLI) en el nodo a fin de recuperar la información de instalación y actualización de los administradores de paquetes especificados e impulsar el administrador de paquetes adecuado para cada paquete de actualización.

Cada instantánea se corresponde con una Cuenta de AWS, un grupo de revisiones, un sistema operativo y un ID de instantánea. La instantánea se entrega a través de una URL prefirmada de Amazon Simple Storage Service (Amazon S3), que se vence transcurridas las 24 horas desde la creación de la instantánea. Sin embargo, si desea aplicar el mismo contenido de la instantánea a otros nodos administrados una vez que la URL se haya vencido, puede generar una nueva dirección de Amazon S3 prefirmada hasta tres días después de la creación de la instantánea. Para ello, utilice el comando get-deployable-patch-snapshot-for-instance.

Cuando se han instalado todas las actualizaciones aprobadas y aplicables, y se han realizado los reinicios necesarios, se genera la información de conformidad de revisiones en un nodo administrado y se notifica a Patch Manager.

> **Note**
>
> Si el parámetro `RebootOption` se configura en `NoReboot` en el documento `AWS-RunPatchBaseline`, el nodo administrado no se reinicia después de que se ejecuta Patch Manager. Para obtener más información, consulte Nombre del parámetro: `RebootOption`.

Para obtener información sobre cómo ver los datos de conformidad de revisiones, consulte Acerca de la conformidad de parches.

Parámetros **AWS-RunPatchBaseline**

`AWS-RunPatchBaseline` admite seis parámetros. El parámetro `Operation` es obligatorio. Los parámetros `InstallOverrideList`, `BaselineOverride` y `RebootOption` son opcionales. `Snapshot-ID` es opcional desde el punto de vista técnico, pero se recomienda proporcionar un valor personalizado al ejecutar `AWS-RunPatchBaseline` fuera de un periodo de mantenimiento. Patch Manager puede suministrar el valor automáticamente cuando el documento se ejecuta como parte de una operación de un periodo de mantenimiento.

Parámetros

- Nombre del parámetro: Operation
- Nombre del parámetro: AssociationId
- Nombre del parámetro: Snapshot ID
- Nombre del parámetro: InstallOverrideList
- Nombre del parámetro: RebootOption
- Nombre del parámetro: BaselineOverride
- Nombre del parámetro: StepTimeoutSeconds

Nombre del parámetro: **Operation**

Usage: requerido.

Opciones: `Scan` | `Install`.

Examen

Cuando elige la opción `Scan`, `AWS-RunPatchBaseline` determina el estado de conformidad de las revisiones del nodo administrado y notifica esta información a Patch Manager. `Scan` no solicita que se instalen actualizaciones ni que se reinicien los nodos administrados. En lugar de ello, la operación identifica las actualizaciones aprobadas que faltan y que son aplicables al nodo.

Instalación

Al elegir la opción `Install`, `AWS-RunPatchBaseline` intenta instalar las actualizaciones aprobadas y aplicables que faltan en el nodo administrado. La información de conformidad de revisiones generada como parte de una operación `Install` no muestra las actualizaciones que faltan, pero podría notificar aquellas que presentan un estado de error si la instalación de la actualización no se ha podido realizar por cualquier motivo. Cuando una actualización se instala

en un nodo administrado, este se reinicia para garantizar que la actualización esté instalada y activa. (Excepción: si el parámetro `RebootOption` se configura en `NoReboot` en el documento `AWS-RunPatchBaseline`, el nodo administrado no se reinicia una vez que se ejecuta Patch Manager. Para obtener más información, consulte Nombre del parámetro: `RebootOption`).

> **Note**
>
> Si se instala una revisión especificada por las reglas de la línea de base antes de que Patch Manager actualice el nodo administrado, es posible que el sistema no se reinicie como es debido. Esto puede ocurrir cuando un usuario instala manualmente una revisión o cuando la instala automáticamente otro programa, como el `unattended-upgrades` paquete de Ubuntu Server.

Nombre del parámetro: **`AssociationId`**

Usage: opcional.

`AssociationId` es el ID de una asociación existente en State Manager, una herramienta de AWS Systems Manager. Patch Manager lo utiliza para agregar datos de conformidad a la asociación especificada. Esta asociación está relacionada con una operación de aplicación de revisiones que está configurada en una política de revisiones en Quick Setup.

> **Note**
>
> Con el `AWS-RunPatchBaseline`, si se proporciona un valor `AssociationId` junto con una anulación de la línea de base de la política de revisiones, la aplicación de revisiones se realiza como una operación `PatchPolicy` y el valor `ExecutionType` informado en `AWS:ComplianceItem` también es `PatchPolicy`. Si no se proporciona un valor `AssociationId`, la aplicación de revisiones se realiza como una operación `Command` y el informe del valor `ExecutionType` en el `AWS:ComplianceItem` enviado también es `Command`.

Si aún no dispone de una asociación que desee utilizar, puede crear una mediante la ejecución del comando create-association.

Nombre del parámetro: **`Snapshot ID`**

Usage: opcional.

`Snapshot ID` es un ID exclusivo (GUID) que utiliza Patch Manager para garantizar que un conjunto de nodos administrados en los que se aplican revisiones en una sola operación tenga el mismo conjunto de revisiones aprobadas. Aunque el parámetro se define como opcional, nuestra mejor práctica recomendada depende de si se ejecuta o no `AWS-RunPatchBaseline` en un periodo de mantenimiento, tal como se describe en la siguiente tabla.

Prácticas recomendadas de **`AWS-RunPatchBaseline`**

Mode	Práctica recomendada	Detalles
Ejecución de `AWS-RunPatchBaseline` dentro de un periodo de mantenimiento	No proporcione un ID de instantánea, sino que Patch Manager lo hará en su lugar.	Si utiliza un periodo de mantenimiento para ejecutar `AWS-RunPatchBaseline`, no debe proporcionar su propio ID de instantánea generado. En este caso, Systems Manager proporciona un valor GUID en función del ID de ejecución del periodo de mantenimiento. De este modo, se garantiza que se utilice un ID correcto para todas las invocaciones de `AWS-RunPatchBaseline` en dicho periodo de mantenimiento. Si especifica un valor en este caso, tenga en cuenta que la instantánea de la línea de base de revisiones no podría mantenerse durante más de tres días. Después de eso, se generará una nueva instantánea, aunque especifique el

Mode	Práctica recomendada	Detalles
		mismo ID después de que expire la instantánea.

Mode	Práctica recomendada	Detalles
Ejecución de `AWS-RunPatchBaseline` fuera de un periodo de mantenimiento	Genere y especifique un valor GUID personalizado para el ID de instantánea.[1]	Cuando no esté utilizando un periodo de mantenimiento para ejecutar `AWS-RunPatchBaseline`, se recomienda generar y especificar un ID de instantánea exclusivo por cada línea de base de revisiones, especialmente si ejecuta el documento `AWS-RunPatchBaseline` en varios nodos administrados durante la misma operación. Si no especifica un ID en este caso, Systems Manager genera otro ID de instantánea para cada nodo administrado al que se envía el comando. Esto podría generar diferentes conjuntos de revisiones que se especifican entre los nodos administrados. Por ejemplo, supongamos que se ejecuta el documento `AWS-RunPatchBaseline` directamente a través de Run Command, una herramienta de AWS Systems Manager, y selecciona como destino un grupo de 50 nodos administrados. Al especificar un ID de instantánea personalizado, se genera una sola instantánea de la línea de base que se utiliza para evaluar y aplicar

Mode	Práctica recomendada	Detalles
		revisiones en todos los nodos, lo que garantiza que tengan un estado coherente.

[1] Puede utilizar cualquier herramienta que genere un GUID con el fin de crear un valor para el parámetro de ID de la instantánea. Por ejemplo, en PowerShell, puede utilizar el cmdlet `New-Guid` para generar un GUID con el formato `12345699-9405-4f69-bc5e-9315aEXAMPLE` .

Nombre del parámetro: **`InstallOverrideList`**

Usage: opcional.

Mediante `InstallOverrideList`, se puede especificar una URL de https o una URL de tipo ruta de Amazon S3 a una lista de revisiones que deben instalarse. Esta lista de instalación de revisiones, que mantiene en formato YAML, invalida las revisiones especificadas por la línea de base de revisiones predeterminada actual. De este modo, se le proporcionará un control pormenorizado sobre qué revisiones se instalan en los nodos administrados.

> **Important**
>
> El nombre de archivo `InstallOverrideList` no puede contener los siguientes caracteres: comillas invertidas (`), comillas simples ('), comillas dobles (") ni signos de dólar ($).

El comportamiento de la operación de revisión cuando se utiliza el parámetro `InstallOverrideList` difiere entre Linux y los nodos administrados por macOS y los nodos administrados por Windows Server. En Linux y macOS, Patch Manager intenta aplicar los parches incluidos en la lista de parches de `InstallOverrideList` que estén presentes en cualquier repositorio habilitado en el nodo, independientemente de que los parches coincidan o no con las reglas de la línea de base de revisiones. Sin embargo, en los nodos Windows Server, los parches de la lista de parches `InstallOverrideList` se aplican solo si también cumplen con las reglas de la línea de base de revisiones.

Tenga en cuenta que los informes de conformidad reflejan los estados de revisiones de acuerdo con lo que se especifica en la línea de base de revisiones, no lo que especifique en una lista de

revisiones `InstallOverrideList`. En otras palabras, las operaciones de análisis omiten el parámetro `InstallOverrideList`. De este modo, se garantiza que los informes de conformidad reflejen de forma coherente los estados de revisiones de acuerdo con la política, en lugar de lo que se ha aprobado una operación específica para la aplicación de revisiones.

Para obtener una descripción de cómo puede utilizar el parámetro `InstallOverrideList` para aplicar diferentes tipos de revisiones a un grupo de destino en diferentes programaciones de ventanas de mantenimiento al tiempo que utiliza una única línea de base de revisiones, consulte Ejemplo de escenario para utilizar el parámetro InstallOverrideList en `AWS-RunPatchBaseline` o `AWS-RunPatchBaselineAssociation`.

Formatos de URL válidos

> **Note**
>
> Si su archivo se encuentra almacenado en un bucket de acceso público, puede especificar un formato de URL de https o una URL de tipo ruta de Amazon S3. Si su archivo se encuentra almacenado en un bucket privado, debe especificar una URL de tipo ruta de Amazon S3.

- Formato de URL https:

```
https://s3.aws-api-domain/amzn-s3-demo-bucket/my-windows-override-list.yaml
```

- URL de tipo ruta de Amazon S3:

```
s3://amzn-s3-demo-bucket/my-windows-override-list.yaml
```

Formatos de contenido YAML válidos

Los formatos que utiliza para especificar revisiones en su lista dependen del sistema operativo del nodo administrado. El formato general, sin embargo, es el siguiente:

```
patches:
    -
        id: '{patch-d}'
        title: '{patch-title}'
```

```
        {additional-fields}:{values}
```

Aunque puede proporcionar campos adicionales en su archivo YAML, estos se pasan por alto durante las operaciones de revisiones.

Además, le recomendamos que verifique que el formato de su archivo YAML es válido antes de añadir o actualizar la lista de su bucket de S3. Para obtener más información acerca del formato YAML, consulte yaml.org. Para consultar las opciones de herramientas de validación, realice una búsqueda web de "validadores de formato yaml".

Linux

id

El campo id es obligatorio. Utilícelo para especificar revisiones mediante el nombre del paquete y la arquitectura. Por ejemplo: `'dhclient.x86_64'`. Puede utilizar comodines en id para indicar varios paquetes. Por ejemplo: `'dhcp*'` y `'dhcp*1.*'`.

Título

El campo title (título) es opcional, pero en sistemas Linux ofrece capacidades de filtrado adicionales. Si utiliza title (título), debería contener información de la versión del paquete en uno de los siguientes formatos:

YUM/SUSE Linux Enterprise Server (SLES):

```
{name}.{architecture}:{epoch}:{version}-{release}
```

APT

```
{name}.{architecture}:{version}
```

Para los títulos de las revisiones de Linux, puede utilizar uno o varios comodines en cualquier posición para ampliar el número de paquetes coincidentes. Por ejemplo: `'*32:9.8.2-0.*.rc1.57.amzn1'`.

Por ejemplo:

- la versión del paquete apt 1.2.25 actualmente está instalada en el nodo administrado, pero la versión 1.2.27 ya está disponible.

- Puede añadir la versión apt.amd64 1.2.27 a la lista de revisiones. Depende de la versión apt utils.amd64 1.2.27, pero la versión apt-utils.amd64 1.2.25 se especifica en la lista.

En este caso, la versión apt 1.2.27 se bloqueará a partir de la instalación y se registrará como "Failed-NonCompliant".

Windows Server

id

El campo id es obligatorio. Úselo para especificar las revisiones con los ID de la Base de conocimientos de Microsoft (por ejemplo, KB2736693) y del boletín de seguridad de Microsoft (por ejemplo, MS17-023).

Cualquier otro campo que desee proporcionar en una lista de revisiones para Windows es opcional y solo para fines informativos propios. Puede utilizar campos adicionales como, por ejemplo, title (título), classification (clasificación), severity (gravedad) o cualquier otro para proporcionar información más detallada sobre las revisiones especificados.

macOS

id

El campo id es obligatorio. Se puede proporcionar el valor del campo id mediante un formato `{package-name}.{package-version}` o un formato {package_name}.

Listas de revisiones de ejemplo

- 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'
```

Nombre del parámetro: **`RebootOption`**

Usage: opcional.

Opciones: `RebootIfNeeded` | `NoReboot`

Valor predeterminado: `RebootIfNeeded`

> **Warning**
>
> La opción predeterminada es `RebootIfNeeded`. Asegúrese de seleccionar la opción correcta para su caso de uso. Por ejemplo, si los nodos administrados deben reiniciarse inmediatamente para completar un proceso de configuración, elija `RebootIfNeeded`. O bien, si necesita mantener la disponibilidad de los nodos administrados hasta una hora de reinicio programada, elija `NoReboot`.

> **Important**
>
> No recomendamos el uso de Patch Manager para revisar las instancias de clústeres en Amazon EMR (antes denominado Amazon Elastic MapReduce). En concreto, no seleccione la opción `RebootIfNeeded` para el parámetro `RebootOption`. (Esta opción está disponible en los documentos de SSM Command para implementar revisiones `AWS-RunPatchBaseline`, `AWS-RunPatchBaselineAssociation`, y `AWS-RunPatchBaselineWithHooks`).
> Los comandos subyacentes para implementar revisiones mediante el uso de Patch Manager y los comandos `yum` y `dnf`. Por lo tanto, las operaciones generan incompatibilidades debido

a la forma en que se instalan los paquetes. Para obtener información sobre los métodos preferidos para actualizar el software en los clústeres de Amazon EMR, consulte Uso de la AMI predeterminada para Amazon EMR en la Guía de administración de Amazon EMR.

RebootIfNeeded

Cuando se elige la opción `RebootIfNeeded`, el nodo administrado se reinicia en cualquiera de los siguientes casos:

- Patch Manager instaló una o más revisiones.

  Patch Manager no evalúa si la revisión requiere llevar a cabo un reinicio. El sistema se reinicia aunque la revisión no requiera el reinicio.

- Patch Manager detecta uno o más revisiones con un estado de `INSTALLED_PENDING_REBOOT` durante la operación `Install`.

  El estado `INSTALLED_PENDING_REBOOT` puede indicar que la opción `NoReboot` se seleccionó la última vez que se ejecutó la operación `Install` o que se instaló un parche por fuera de Patch Manager desde la última vez que se reinició el nodo administrado.

El reinicio de los nodos administrados en estos dos casos garantiza que los paquetes actualizados se eliminen de la memoria y mantiene un comportamiento de aplicación de revisiones y reinicio consistente en todos los sistemas operativos.

NoReboot

Cuando elige la opción `NoReboot`, Patch Manager no reinicia un nodo administrado aunque haya instalado revisiones durante la operación `Install`. Esta opción es útil si sabe que los nodos administrados no requieren reinicio después de aplicar las revisiones, o si dispone de aplicaciones o procesos que se ejecutan en un nodo que no deberían verse interrumpidos como consecuencia del reinicio de una operación de aplicación de revisiones. También es útil cuando se desea tener más control sobre el tiempo de los reinicios del nodo administrado, por ejemplo, mediante un periodo de mantenimiento.

Si elige la opción `NoReboot` y se ha instalado una revisión, se asigna a la revisión un estado de `InstalledPendingReboot`. Sin embargo, el nodo administrado en sí mismo está marcado como `Non-Compliant`. Una vez que se produce un reinicio y se ejecuta una operación `Scan`, el estado del nodo administrado se actualiza a `Compliant`.

> **Important**
>
> La opción `NoReboot` solo impide los reinicios a nivel del sistema operativo. Los reinicios a nivel de servicio aún pueden producirse como parte del proceso de aplicación de revisiones. Por ejemplo, cuando se actualiza Docker, los servicios dependientes, como Amazon Elastic Container Service, pueden reiniciarse automáticamente incluso con `NoReboot` habilitado. Si tiene servicios críticos que no deben interrumpirse, considere tomar medidas adicionales, como eliminar temporalmente las instancias del servicio o programar la aplicación de revisiones durante los periodos de mantenimiento.

Archivo de seguimiento de instalación de revisiones: para realizar un seguimiento de la instalación de revisiones, especialmente las instaladas desde el último reinicio del sistema, Systems Manager mantiene un archivo en el nodo administrado.

> **Important**
>
> No elimine ni modifique el archivo de seguimiento. Si este archivo se elimina o está dañado, el informe de conformidad de revisiones para el nodo administrado es inexacto. Si esto sucede, reinicie el nodo y ejecute una operación de análisis de revisiones para restaurar el archivo.

Este archivo de seguimiento se almacena en las siguientes ubicaciones de los nodos administrados:

- Sistemas operativos Linux:
  - `/var/log/amazon/ssm/patch-configuration/patch-states-configuration.json`
  - `/var/log/amazon/ssm/patch-configuration/patch-inventory-from-last-operation.json`
- Sistema operativo Windows Server:
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchStatesConfiguration.json`
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchInventoryFromLastOperation.json`

Nombre del parámetro: **`BaselineOverride`**

Usage: opcional.

Puede definir las preferencias de aplicación de revisiones en tiempo de ejecución mediante el parámetro `BaselineOverride`. Esta anulación de la base de referencia se mantiene como un objeto JSON en un bucket de S3. Garantiza que las operaciones de aplicación de revisiones utilicen las bases de referencia proporcionadas que concuerdan con el sistema operativo del host en lugar de aplicar las reglas de la línea de base de revisiones predeterminada

> **Important**
>
> El nombre de archivo `BaselineOverride` no puede contener los siguientes caracteres: comillas invertidas (`), comillas simples ('), comillas dobles (") ni signos de dólar ($).

Para obtener más información acerca de cómo utilizar el parámetro `BaselineOverride`, consulte Uso del parámetro BaselineOverride.

Nombre del parámetro: **`StepTimeoutSeconds`**

Usage: opcional.

El tiempo en segundos, entre 1 y 36 000 segundos (10 horas), para que un comando se complete antes de considerar que se ha producido un error.

Documento de comandos SSM para aplicar revisiones: **`AWS-RunPatchBaselineAssociation`**

Al igual que el documento `AWS-RunPatchBaseline`, `AWS-RunPatchBaselineAssociation` realiza operaciones de aplicación de revisiones en las instancias para actualizaciones relacionadas con la seguridad y de otros tipos. Además, puede utilizar el documento `AWS-RunPatchBaselineAssociation` para aplicar revisiones a los sistemas operativos y a las aplicaciones. (En Windows Server, la compatibilidad con las aplicaciones se limita a las actualizaciones de las aplicaciones publicadas por Microsoft).

Este documento admite instancias de Amazon Elastic Compute Cloud (Amazon EC2) para Linux, macOS y Windows Server. No admite nodos que no sean de EC2 en un entorno híbrido y multinube. El documento se encargará de realizar las acciones adecuadas para cada plataforma, para lo cual invocará un módulo de Python en las instancias de Linux y macOS, así como un módulo de PowerShell en las instancias de Windows.

Sin embargo, `AWS-RunPatchBaselineAssociation` se diferencia de `AWS-RunPatchBaseline` en los siguientes aspectos:

- `AWS-RunPatchBaselineAssociation` está diseñado para que se utilice principalmente con asociaciones de State Manager creadas con Quick Setup, una herramienta de AWS Systems Manager. Especialmente cuando se utiliza el tipo de configuración Quick Setup de administración de host, si elige la opción escanear las instancias para detectar las revisiones que faltan cada día, el sistema utiliza `AWS-RunPatchBaselineAssociation` para efectuar la operación.

  Sin embargo, en la mayoría de los casos, a la hora de configurar sus propias operaciones de aplicación de revisiones, debe elegir `AWS-RunPatchBaseline` o `AWS-RunPatchBaselineWithHooks` en lugar de `AWS-RunPatchBaselineAssociation`.
- Cuando se utiliza el documento `AWS-RunPatchBaselineAssociation`, se puede especificar un par de claves de etiqueta en el campo correspondiente al parámetro `BaselineTags` del documento. Si una línea de base de revisiones personalizada en su Cuenta de AWS comparte estas etiquetas, Patch Manager, una herramienta de AWS Systems Manager, utiliza esa línea de base etiquetada cuando se ejecuta en las instancias de destino en lugar de la línea de base de revisiones “predeterminada” que actualmente se encuentra especificada para el tipo de sistema operativo.

  > **Note**
  >
  > Si elige utilizar `AWS-RunPatchBaselineAssociation` en las operaciones de aplicación de revisiones distintas de las configuradas mediante Quick Setup, y desea utilizar el parámetro opcional `BaselineTags`, es necesario que proporcione algunos permisos adicionales al perfil de instancias para las instancias de Amazon Elastic Compute Cloud (Amazon EC2). Para obtener más información, consulte Nombre del parámetro: `BaselineTags`.

  Los dos siguientes formatos son válidos para su parámetro `BaselineTags`:

  ```
  Key=tag-key,Values=tag-value
  ```

  ```
  Key=tag-key,Values=tag-value1,tag-value2,tag-value3
  ```

> **Important**
>
> Los valores y las claves de las etiquetas no pueden contener los siguientes caracteres: comillas invertidas (`), comillas simples ('), comillas dobles (") ni signos de dólar ($).

- Los datos de conformidad de revisiones que se recopilan cuando se ejecuta `AWS-RunPatchBaselineAssociation` se registran mediante el comando `PutComplianceItems` de la API en lugar del comando `PutInventory`, que utiliza `AWS-RunPatchBaseline`. Esta diferencia implica que la información de conformidad de revisiones se almacena y notifica en función de una asociación específica. Los datos de conformidad de revisiones generados fuera de esta asociación no se sobrescriben.
- La información de conformidad con la revisión notificada con posterioridad a la ejecución de `AWS-RunPatchBaselineAssociation` indica si una instancia está en conformidad o no. No se incluyen los detalles a nivel de revisión, como se demuestra con la salida del siguiente comando de la AWS Command Line Interface (AWS CLI). El comando filtra en `Association` como el tipo de conformidad:

```
aws ssm list-compliance-items \
    --resource-ids "i-02573cafcfEXAMPLE" \
    --resource-types "ManagedInstance" \
    --filters "Key=ComplianceType,Values=Association,Type=EQUAL" \
    --region us-east-2
```

El sistema devuelve información similar a la siguiente.

```
{
    "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 se ha especificado un valor de par de claves de etiqueta como parámetro para el documento `AWS-RunPatchBaselineAssociation`, Patch Manager busca una línea de base de revisiones personalizada que concuerde con el tipo de sistema operativo y que se haya etiquetado con ese mismo par de claves de etiquetas. Esta búsqueda no se limita a la línea de base de revisiones predeterminada que se haya especificado ni a la base de referencia asignada a un grupo de revisiones. Si no se encuentra ninguna base de referencia con las etiquetas especificadas, Patch Manager busca a continuación un grupo de revisiones, siempre que se haya especificado uno en el comando que ejecuta `AWS-RunPatchBaselineAssociation`. Si no hay ningún grupo de revisiones que concuerde, Patch Manager vuelve a la línea de base de revisiones predeterminada actual para la cuenta del sistema operativo.

Si se encuentra más de una línea de base de revisiones con las etiquetas especificadas en el documento `AWS-RunPatchBaselineAssociation`, Patch Manager devuelve un mensaje de error donde se indica que solo es posible etiquetar una línea de base de revisiones con ese par de valor de clave para proceder con la operación.

> **Note**
>
> En las instancias de Linux, se utiliza el administrador de paquetes adecuado para cada tipo de instancia a fin de instalar los paquetes:
>
> - Las instancias de Amazon Linux 1, Amazon Linux 2, CentOS, Oracle Linux y RHEL utilizan YUM. Para las operaciones de YUM, Patch Manager requiere `Python 2.6` o una versión posterior compatible (2.6 a 3.10).
> - Las instancias de Debian Server, Raspberry Pi OS y Ubuntu Server usan APT. Para las operaciones de APT, Patch Manager requiere una versión compatible de `Python 3` (3.0 a 3.10).

- Las instancias de SUSE Linux Enterprise Server utilizan Zypper. Para las operaciones de Zypper, Patch Manager requiere `Python 2.6` o una versión posterior compatible (2.6 a 3.10).

Una vez que se haya completado un análisis, o bien después de que se hayan instalado todas las actualizaciones aprobadas y aplicables, y se hayan realizado los reinicios necesarios, se genera la información de conformidad de revisiones en una instancia y se notifica al servicio de conformidad de revisiones.

> **Note**
>
> Si el parámetro `RebootOption` se establece en `NoReboot` en el documento `AWS-RunPatchBaselineAssociation`, la instancia no se reinicia después de que se ejecuta Patch Manager. Para obtener más información, consulte Nombre del parámetro: `RebootOption`.

Para obtener información sobre cómo ver los datos de conformidad de revisiones, consulte Acerca de la conformidad de parches.

Parámetros **`AWS-RunPatchBaselineAssociation`**

`AWS-RunPatchBaselineAssociation` admite cinco parámetros. Los parámetros `Operation` y `AssociationId` son obligatorios. Los parámetros `InstallOverrideList`, `RebootOption` y `BaselineTags` son opcionales.

Parámetros

- Nombre del parámetro: Operation
- Nombre del parámetro: BaselineTags
- Nombre del parámetro: AssociationId
- Nombre del parámetro: InstallOverrideList
- Nombre del parámetro: RebootOption

Nombre del parámetro: **`Operation`**

Usage: requerido.

Opciones: `Scan` | `Install`.

Examen

Cuando elija la opción `Scan`, `AWS-RunPatchBaselineAssociation` determina el estado de conformidad de las revisiones de la instancia y notifica esta información a Patch Manager. `Scan` no solicita que se instalen actualizaciones ni que se reinicien las instancias. En lugar de ello, la operación identifica las actualizaciones aprobadas que faltan y que son aplicables a la instancia.

Instalación

Al elegir la opción `Install`, `AWS-RunPatchBaselineAssociation` intenta instalar las actualizaciones aprobadas y aplicables que faltan en la instancia. La información de conformidad de revisiones generada como parte de una operación `Install` no muestra las actualizaciones que faltan, pero podría notificar aquellas que presentan un estado de error si la instalación de la actualización no se ha podido realizar por cualquier motivo. Cuando una actualización se instala en una instancia, esta se reinicia para garantizar que la actualización esté instalada y activa. (Excepción: si el parámetro `RebootOption` se establece en `NoReboot` en el documento `AWS-RunPatchBaselineAssociation`, la instancia no se reinicia una vez que se ejecuta Patch Manager. Para obtener más información, consulte Nombre del parámetro: `RebootOption`.)

> **Note**
>
> Si se instala una revisión especificado por las reglas de la base de referencia antes de que Patch Manager actualice la instancia, es posible que el sistema no se reinicie como es debido. Esto puede ocurrir cuando un usuario instala manualmente una revisión o cuando lo instala automáticamente otro programa, como el `unattended-upgrades` paquete de Ubuntu Server.

Nombre del parámetro: **`BaselineTags`**

Usage: opcional.

`BaselineTags` es un par de valor de clave de etiqueta único que usted elige y asigna a una línea de base de revisiones personalizada particular. Puede especificar uno o más valores para este parámetro. Los dos formatos que se indican a continuación son válidos:

`Key=`*`tag-key`*`,Values=`*`tag-value`*

`Key=`*`tag-key`*`,Values=`*`tag-value1`*`,`*`tag-value2`*`,`*`tag-value3`*

> **Important**
>
> Los valores y las claves de las etiquetas no pueden contener los siguientes caracteres: comillas invertidas (`), comillas simples ('), comillas dobles (") ni signos de dólar ($).

Patch Manager utiliza el valor `BaselineTags` para garantizar que un conjunto de instancias en las que se aplican revisiones en una sola operación tenga el mismo conjunto de revisiones aprobados. Cuando se ejecuta la operación de aplicación de revisiones, Patch Manager comprueba si una línea de base de revisiones para el tipo de sistema operativo está etiquetada con el mismo par de valor de clave que se especifica para `BaselineTags`. Si hay una concordancia, se utiliza esta línea de base de revisiones personalizada. Si no hay ninguna concordancia, se identifica una línea de base de revisiones en función de cualquier grupo de revisiones especificado para la operación de aplicación de revisiones. En caso de que no haya ninguno, se utiliza la línea de base de revisiones predefinida administrada por AWS para ese sistema operativo.

Requisitos de permisos adicionales

Si elige utilizar `AWS-RunPatchBaselineAssociation` en las operaciones de aplicación de revisiones distintas de las configuradas mediante Quick Setup, y desea utilizar el parámetro opcional `BaselineTags`, es necesario que agregue los siguientes permisos al perfil de instancias para las instancias de Amazon Elastic Compute Cloud (Amazon EC2).

> **Note**
>
> Quick Setup y `AWS-RunPatchBaselineAssociation` no admiten servidores locales ni máquinas virtuales.

```
{
    "Effect": "Allow",
    "Action": [
        "ssm:DescribePatchBaselines",
        "tag:GetResources"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
```

```
    "Action": [
        "ssm:GetPatchBaseline",
        "ssm:DescribeEffectivePatchesForPatchBaseline"
    ],
    "Resource": "patch-baseline-arn"
}
```

Sustituya *patch-baseline-arn* por el nombre de recurso de Amazon (ARN) de la línea de base de revisiones al que desea proporcionar acceso, con el formato `arn:aws:ssm:us-east-2:123456789012:patchbaseline/pb-0c10e65780EXAMPLE`.

Nombre del parámetro: **`AssociationId`**

Usage: requerido.

`AssociationId` es el ID de una asociación existente en State Manager, una herramienta de AWS Systems Manager. Patch Manager lo utiliza para agregar datos de conformidad a la asociación especificada. Esta asociación está relacionada con una operación de revisión `Scan` habilitada en una configuración de Administración de host creada en Quick Setup . Con el envío de los resultados de la aplicación de revisiones como datos de conformidad de la asociación en lugar de datos de conformidad de inventario, la información de conformidad de inventario existente para sus instancias ni otros ID de asociación no se sobrescribe luego de una operación de aplicación de revisiones. Si aún no dispone de una asociación que desee utilizar, puede crear una mediante la ejecución del comando create-association. Por ejemplo:

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
```

Nombre del parámetro: **`InstallOverrideList`**

Usage: opcional.

Mediante `InstallOverrideList`, se puede especificar una URL de https o una URL de tipo ruta de Amazon Simple Storage Service (Amazon S3) para una lista de revisiones que deben instalarse. Esta lista de instalación de revisiones, que mantiene en formato YAML, invalida las revisiones especificados por la línea de base de revisiones predeterminada actual. De este modo, se le proporcionará un control más detallado sobre qué revisiones se instalan en las instancias.

> **Important**
>
> El nombre de archivo `InstallOverrideList` no puede contener los siguientes caracteres: comillas invertidas (`), comillas simples ('), comillas dobles (") ni signos de dólar ($).

El comportamiento de la operación de revisión cuando se utiliza el parámetro `InstallOverrideList` difiere entre Linux y los nodos administrados por macOS y los nodos administrados por Windows Server. En Linux y macOS, Patch Manager intenta aplicar los parches incluidos en la lista de parches de `InstallOverrideList` que estén presentes en cualquier repositorio habilitado en el nodo, independientemente de que los parches coincidan o no con las reglas de la línea de base de revisiones. Sin embargo, en los nodos Windows Server, los parches de la lista de parches `InstallOverrideList` se aplican solo si también cumplen con las reglas de la línea de base de revisiones.

Tenga en cuenta que los informes de conformidad reflejan los estados de revisiones de acuerdo con lo que se especifica en la línea de base de revisiones, no lo que especifique en una lista de revisiones `InstallOverrideList`. En otras palabras, las operaciones de análisis omiten el parámetro `InstallOverrideList`. De este modo, se garantiza que los informes de conformidad

reflejen de forma coherente los estados de revisiones de acuerdo con la política, en lugar de lo que se ha aprobado una operación específica para la aplicación de revisiones.

Formatos de URL válidos

> **Note**
>
> Si su archivo se encuentra almacenado en un bucket de acceso público, puede especificar un formato de URL de https o una URL de tipo ruta de Amazon S3. Si su archivo se encuentra almacenado en un bucket privado, debe especificar una URL de tipo ruta de Amazon S3.

- Ejemplo de formato de URL https:

```
https://s3.amazonaws.com/amzn-s3-demo-bucket/my-windows-override-list.yaml
```

- Ejemplo de URL de estilo ruta de Amazon S3:

```
s3://amzn-s3-demo-bucket/my-windows-override-list.yaml
```

Formatos de contenido YAML válidos

Los formatos que utiliza para especificar revisiones en su lista dependen del sistema operativo de la instancia. El formato general, sin embargo, es el siguiente:

```
patches:
    -
        id: '{patch-d}'
        title: '{patch-title}'
        {additional-fields}:{values}
```

Aunque puede proporcionar campos adicionales en su archivo YAML, estos se pasan por alto durante las operaciones de revisiones.

Además, le recomendamos que verifique que el formato de su archivo YAML es válido antes de añadir o actualizar la lista de su bucket de S3. Para obtener más información acerca del formato YAML, consulte yaml.org. Para consultar las opciones de herramientas de validación, realice una búsqueda web de "validadores de formato yaml".

- Microsoft Windows

  id

  El campo id es obligatorio. Úselo para especificar las revisiones con los ID de la Base de conocimientos de Microsoft (por ejemplo, KB2736693) y del boletín de seguridad de Microsoft (por ejemplo, MS17-023).

  Cualquier otro campo que desee proporcionar en una lista de revisiones para Windows es opcional y solo para fines informativos propios. Puede utilizar campos adicionales como, por ejemplo, title (título), classification (clasificación), severity (gravedad) o cualquier otro para proporcionar información más detallada sobre las revisiones especificados.

- Linux

  id

  El campo id es obligatorio. Utilícelo para especificar revisiones mediante el nombre del paquete y la arquitectura. Por ejemplo: `'dhclient.x86_64'`. Puede utilizar comodines en id para indicar varios paquetes. Por ejemplo: `'dhcp*'` y `'dhcp*1.*'`.

  título

  El campo title (título) es opcional, pero en sistemas Linux ofrece capacidades de filtrado adicionales. Si utiliza title (título), debería contener información de la versión del paquete en uno de los siguientes formatos:

  YUM/SUSE Linux Enterprise Server (SLES):

  ```
  {name}.{architecture}:{epoch}:{version}-{release}
  ```

  APT

  ```
  {name}.{architecture}:{version}
  ```

  Para los títulos de las revisiones de Linux, puede utilizar uno o varios comodines en cualquier posición para ampliar el número de paquetes coincidentes. Por ejemplo: `'*32:9.8.2-0.*.rc1.57.amzn1'`.

  Por ejemplo:

- la versión del paquete apt 1.2.25 actualmente está instalada en la instancia, pero la versión 1.2.27 ya está disponible.
- Puede añadir la versión apt.amd64 1.2.27 a la lista de revisiones. Depende de la versión apt utils.amd64 1.2.27, pero la versión apt-utils.amd64 1.2.25 se especifica en la lista.

En este caso, la versión apt 1.2.27 se bloqueará a partir de la instalación y se registrará como "Failed-NonCompliant".

Otros campos

Cualquier otro campo que desee proporcionar en una lista de revisiones para Linux es opcional y solo para fines informativos propios. Puede utilizar campos adicionales como, por ejemplo, classification (clasificación), severity (gravedad) o cualquier otro para proporcionar información más detallada sobre las revisiones especificados.

Listas de revisiones de ejemplo

- 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'
```

Nombre del parámetro: **`RebootOption`**

Usage: opcional.

Opciones: `RebootIfNeeded` | `NoReboot`

Valor predeterminado: `RebootIfNeeded`

> **Warning**
>
> La opción predeterminada es `RebootIfNeeded`. Asegúrese de seleccionar la opción correcta para su caso de uso. Por ejemplo, si las instancias deben reiniciarse inmediatamente para completar un proceso de configuración, elija `RebootIfNeeded`. O bien, si necesita mantener la disponibilidad de las instancias hasta una hora de reinicio programada, elija `NoReboot`.

> **Important**
>
> La opción `NoReboot` solo impide los reinicios a nivel del sistema operativo. Los reinicios a nivel de servicio aún pueden producirse como parte del proceso de aplicación de revisiones. Por ejemplo, cuando se actualiza Docker, los servicios dependientes, como Amazon Elastic

Container Service, pueden reiniciarse automáticamente incluso con `NoReboot` habilitado. Si tiene servicios críticos que no deben interrumpirse, considere tomar medidas adicionales, como eliminar temporalmente las instancias del servicio o programar la aplicación de revisiones durante los periodos de mantenimiento.

Important

No recomendamos el uso de Patch Manager para revisar las instancias de clústeres en Amazon EMR (antes denominado Amazon Elastic MapReduce). En concreto, no seleccione la opción `RebootIfNeeded` para el parámetro `RebootOption`. (Esta opción está disponible en los documentos de SSM Command para implementar revisiones `AWS-RunPatchBaseline`, `AWS-RunPatchBaselineAssociation`, y `AWS-RunPatchBaselineWithHooks`).
Los comandos subyacentes para implementar revisiones mediante el uso de Patch Manager y los comandos `yum` y `dnf`. Por lo tanto, las operaciones generan incompatibilidades debido a la forma en que se instalan los paquetes. Para obtener información sobre los métodos preferidos para actualizar el software en los clústeres de Amazon EMR, consulte Uso de la AMI predeterminada para Amazon EMR en la Guía de administración de Amazon EMR.

RebootIfNeeded

Cuando se elige la opción `RebootIfNeeded`, la instancia se reinicia en cualquiera de los siguientes casos:

- Patch Manager instaló uno o más revisiones.

  Patch Manager no evalúa si la revisión requiere llevar a cabo un reinicio. El sistema se reinicia aunque la revisión no requiera el reinicio.

- Patch Manager detecta uno o más revisiones con un estado de `INSTALLED_PENDING_REBOOT` durante la operación `Install`.

  El estado `INSTALLED_PENDING_REBOOT` puede indicar que la opción `NoReboot` se seleccionó la última vez que se ejecutó la operación `Install` o que se instaló un parche por fuera de Patch Manager desde la última vez que se reinició el nodo administrado.

El reinicio de las instancias en estos dos casos garantiza que los paquetes actualizados se eliminen de la memoria y mantiene un comportamiento de aplicación de revisiones y reinicio consistente en todos los sistemas operativos.

NoReboot

Cuando elige la opción `NoReboot`, Patch Manager no reinicia una instancia aunque haya instalado revisiones durante la operación `Install`. Esta opción es útil si sabe que las instancias no requieren reinicio después de aplicar las revisiones, o si dispone de aplicaciones o procesos que se ejecutan en una instancia que no deberían verse interrumpidos como consecuencia del reinicio de una operación de aplicación de revisiones. También es útil cuando se desea tener más control sobre el tiempo de los reinicios de la instancia, por ejemplo, mediante un periodo de mantenimiento.

Archivo de seguimiento de instalación de revisiones: para realizar un seguimiento de la instalación de revisiones, especialmente las revisiones que se hayan instalados desde el último reinicio del sistema, Systems Manager mantiene un archivo en la instancia administrada.

> **Important**
>
> No elimine ni modifique el archivo de seguimiento. Si este archivo se elimina o está dañado, el informe de conformidad de revisiones para la instancia es inexacto. Si esto sucede, reinicie la instancia y ejecute una operación de análisis de revisión para restaurar el archivo.

Este archivo de seguimiento se almacena en las siguientes ubicaciones de las instancias administradas:

- Sistemas operativos Linux:
  - `/var/log/amazon/ssm/patch-configuration/patch-states-configuration.json`
  - `/var/log/amazon/ssm/patch-configuration/patch-inventory-from-last-operation.json`
- Sistema operativo Windows Server:
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchStatesConfiguration.json`
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchInventoryFromLastOperation.json`

Documento de comandos SSM para aplicar revisiones: **AWS-RunPatchBaselineWithHooks**

AWS Systems Manager es compatible con `AWS-RunPatchBaselineWithHooks`, un documento de Systems Manager (documento de SSM) para Patch Manager, una herramienta de AWS Systems Manager. Este documento de SSM realiza operaciones de aplicación de revisiones en los nodos administrados para actualizaciones relacionadas con la seguridad y de otros tipos.

`AWS-RunPatchBaselineWithHooks` se diferencia de `AWS-RunPatchBaseline` en los siguientes aspectos:

- Un documento contenedor: `AWS-RunPatchBaselineWithHooks` es un contenedor de `AWS-RunPatchBaseline` y se basa en `AWS-RunPatchBaseline` para realizar algunas de sus operaciones.
- La operación **`Install`**: `AWS-RunPatchBaselineWithHooks` admite enlaces de ciclo de vida que se ejecutan en puntos designados durante la aplicación de revisiones en los nodos administrados. Dado que las instalaciones de revisiones en ocasiones requieren que se reinicien los nodos administrados, la operación de aplicación de revisiones se divide en dos eventos, lo que supone un total de tres enlaces que permiten una funcionalidad personalizada. El primer enlace tiene lugar antes de la operación `Install with NoReboot`. El segundo enlace tiene lugar después de la operación `Install with NoReboot`. El tercer enlace está disponible después del reinicio del nodo administrado.
- Sin compatibilidad con la lista de parches personalizados: `AWS-RunPatchBaselineWithHooks` no admite el parámetro `InstallOverrideList`.
- Compatibilidad con SSM Agent: `AWS-RunPatchBaselineWithHooks` requiere que se instale la versión 3.0.502 o una posterior de SSM Agent en el nodo administrado en el que se aplicará la revisión.

Cuando el documento se ejecuta, utiliza la base de referencia de parches que actualmente se encuentra especificada como “predeterminada” para un tipo de sistema operativo en caso de que no se hubiera indicado ningún grupo de parches. En caso contrario, utiliza las bases de referencia de parches que se asocian con el grupo de parches. Para obtener información acerca de los grupos de parches, consulte Grupos de revisiones.

Puede utilizar el documento `AWS-RunPatchBaselineWithHooks` para aplicar revisiones a los sistemas operativos y a las aplicaciones. (En Windows Server, la compatibilidad con las aplicaciones se limita a las actualizaciones de las aplicaciones publicadas por Microsoft).

Este documento es compatible con los nodos administrados de Linux y Windows Server. El documento se encargará de realizar las acciones adecuadas para cada plataforma.

> **Note**
>
> `AWS-RunPatchBaselineWithHooks` no es compatible conmacOS.

Linux

En nodos administrados de Linux, el documento `AWS-RunPatchBaselineWithHooks` invoca un módulo de Python, que a su vez descarga una instantánea de la línea de base de revisiones que se aplica al nodo administrado. Esta instantánea de la línea de base de revisiones utiliza las reglas definidas y las listas de revisiones aprobadas y bloqueadas con el fin de impulsar el administrador de paquetes adecuado para cada tipo de nodo:

- Los nodos administrados de Amazon Linux 1, Amazon Linux 2, CentOS, Oracle Linux y RHEL 7 utilizan YUM. Para las operaciones de YUM, Patch Manager requiere `Python 2.6` o una versión posterior compatible (2.6 a 3.10).
- Los nodos administrados de RHEL 8 utilizan DNF. Para las operaciones de DNF, Patch Manager requiere una versión compatible de `Python 2` o `Python 3` (2.6 a 3.10). (Ninguna de las dos versiones viene instalada en RHEL 8 de forma predeterminada. Para ello, deberá instalar una u otra versión manualmente).
- Las instancias de Debian Server, Raspberry Pi OS y Ubuntu Server usan APT. Para las operaciones de APT, Patch Manager requiere una versión compatible de `Python 3` (3.0 a 3.10).
- Los nodos administrados de SUSE Linux Enterprise Server utilizan Zypper. Para las operaciones de Zypper, Patch Manager requiere `Python 2.6` o una versión posterior compatible (2.6 a 3.10).

Windows Server

En nodos administrados de Windows Server, el documento `AWS-RunPatchBaselineWithHooks` descarga e invoca un módulo de PowerShell, que a su vez descarga una instantánea de la línea de base de revisiones que se aplica al nodo administrado. Esta instantánea de la línea de base de revisiones contiene una lista de revisiones aprobadas que se compila al consultar dicha línea de base de revisiones en un servidor de Windows Server

Update Services (WSUS). Esta lista se transfiere a la API de Windows Update, que controla la descarga y la instalación de los parches aprobados, según proceda.

Cada instantánea se corresponde con una Cuenta de AWS, un grupo de parches, un sistema operativo y un ID de instantánea. La instantánea se entrega a través de una URL prefirmada de Amazon Simple Storage Service (Amazon S3), que se vence transcurridas las 24 horas desde la creación de la instantánea. Sin embargo, si desea aplicar el mismo contenido de la instantánea a otros nodos administrados una vez que la URL se haya vencido, puede generar una nueva dirección de Amazon S3 prefirmada hasta tres días después de la creación de la instantánea. Para ello, utilice el comando get-deployable-patch-snapshot-for-instance.

Cuando se han instalado todas las actualizaciones aprobadas y aplicables, y se han realizado los reinicios necesarios, se genera la información de conformidad de revisiones en un nodo administrado y se notifica a Patch Manager.

Si el parámetro `RebootOption` se configura en `NoReboot` en el documento `AWS-RunPatchBaselineWithHooks`, el nodo administrado no se reinicia después de que se ejecuta Patch Manager. Para obtener más información, consulte Nombre del parámetro: `RebootOption`.

> **Important**
>
> Si bien la opción `NoReboot` impide el reinicio del sistema operativo, no impide los reinicios a nivel de servicio que pueden producirse cuando se actualizan determinados paquetes. Por ejemplo, la actualización de paquetes como Docker puede provocar el reinicio automático de los servicios dependientes (como los servicios de orquestación de contenedores) incluso cuando `NoReboot` está especificado.

Para obtener información sobre cómo ver los datos de conformidad de revisiones, consulte Acerca de la conformidad de parches.

Pasos operativos de **`AWS-RunPatchBaselineWithHooks`**

Cuando se ejecuta `AWS-RunPatchBaselineWithHooks`, se llevan a cabo los siguientes pasos:

1. Análisis: se ejecuta una operación `Scan` con `AWS-RunPatchBaseline` en el nodo administrado, y, de este modo, se genera y carga un informe de conformidad.
2. Verificación de los estados del parche local: se ejecuta un script para determinar los pasos que se llevarán a cabo en función de la operación seleccionada y el resultado de `Scan` del paso 1.

a. Si la operación seleccionada es `Scan`, esta se marcará como completada. La operación finaliza.

b. Si la operación seleccionada es `Install`, Patch Manager evalúa el resultado de `Scan` del paso 1 para determinar qué ejecutar a continuación:

   i. Si no se detectan parches faltantes ni se requieren reinicios pendientes, la operación continúa directamente con el último paso (paso 8), que incluye un enlace que usted ha proporcionado. Se omiten los pasos intermedios.

   ii. Si no se detectan parches faltantes, pero hay reinicios pendientes, y la opción de reinicio seleccionada es `NoReboot`, la operación continúa directamente con el último paso (paso 8), que incluye un enlace que usted ha proporcionado. Se omiten los pasos intermedios.

   iii. De lo contrario, la operación continúa con el siguiente paso.

3. Operación de enlace previa a la aplicación de revisiones: el documento de SSM que ha proporcionado para el primer enlace de ciclo de vida, `PreInstallHookDocName`, se ejecuta en el nodo administrado.
4. Instalación con NoReboot: se ejecuta una operación `Install` con la opción de reinicio de `NoReboot` mediante `AWS-RunPatchBaseline` en el nodo administrado, y, de este modo, se genera y carga un informe de conformidad.
5. Operación de enlace posterior a la instalación: el documento de SSM que ha proporcionado para el segundo enlace de ciclo de vida, `PostInstallHookDocName`, se ejecuta en el nodo administrado.
6. Verificación del reinicio: se ejecuta un script para determinar si es necesario reiniciar el nodo administrado y cuáles son los pasos a ejecutar:

   a. Si la opción de reinicio seleccionada es `NoReboot`, la operación continúa directamente con el último paso (paso 8), que incluye un enlace que usted ha proporcionado. Se omiten los pasos intermedios.

   b. Si la opción de reinicio seleccionada es `RebootIfNeeded`, Patch Manager verifica que no haya reinicios pendientes necesarios a partir del inventario recopilado en el paso 4. Esto significa que la operación continúa con el paso 7 y el nodo administrado se reinicia en cualquiera de los siguientes casos:

      i. Patch Manager instaló una o más revisiones. (Patch Manager no evalúa si la revisión requiere llevar a cabo un reinicio. El sistema se reinicia aunque la revisión no requiera el reinicio.)

      ii. Patch Manager detecta una o más revisiones con un estado `INSTALLED_PENDING_REBOOT` durante la operación de instalación. El estado `INSTALLED_PENDING_REBOOT` puede

indicar que la opción `NoReboot` se seleccionó la última vez que se ejecutó la operación de instalación o que se instaló un parche por fuera de Patch Manager desde la última vez que se reinició el nodo administrado.

Si no se encuentran revisiones que requieran un reinicio, la operación de aplicación de revisiones en el nodo administrado se completa, de modo que la operación continúa directamente con el último paso (paso 8), que incluye un enlace que usted ha proporcionado. Se omiten los pasos intermedios.

7. Instalación e informe: se ejecuta una operación de instalación con la opción de reinicio de `RebootIfNeeded` en el nodo administrado mediante `AWS-RunPatchBaseline`, y, de este modo, se genera y carga un informe de conformidad.
8. Operación de enlace posterior al reinicio: el documento de SSM que ha proporcionado para el tercer enlace de ciclo de vida, `OnExitHookDocName`, se ejecuta en el nodo administrado.

Para una operación `Scan`, si se produce un error en el paso 1, el proceso de ejecución del documento se detiene y ese paso se notifica como un error, aunque los posteriores se hayan realizado correctamente.

Para una operación `Install`, si se produce un error en alguno de los pasos de `aws:runDocument` durante la operación, esos se notifican como error, de modo que la operación continúa directamente con el último paso (paso 8), que incluye un enlace que usted ha proporcionado. Se omiten los pasos intermedios. Este paso se notifica como error, mientras que el último paso notifica el estado del resultado de su operación, y todos los pasos intermedios se notifican como realizados correctamente.

Parámetros **`AWS-RunPatchBaselineWithHooks`**

`AWS-RunPatchBaselineWithHooks` admite seis parámetros.

El parámetro `Operation` es obligatorio.

Los parámetros `RebootOption`, `PreInstallHookDocName`, `PostInstallHookDocName` e `OnExitHookDocName` son opcionales.

`Snapshot-ID` es opcional desde el punto de vista técnico, pero se recomienda que proporcione un valor personalizado cuando ejecute `AWS-RunPatchBaselineWithHooks` fuera de un periodo de mantenimiento. Permita que Patch Manager proporcione el valor automáticamente cuando se ejecute el documento como parte de una operación del periodo de mantenimiento.

Parámetros

- Nombre del parámetro: Operation
- Nombre del parámetro: Snapshot ID
- Nombre del parámetro: RebootOption
- Nombre del parámetro: PreInstallHookDocName
- Nombre del parámetro: PostInstallHookDocName
- Nombre del parámetro: OnExitHookDocName

Nombre del parámetro: **`Operation`**

Usage: requerido.

Opciones: `Scan` | `Install`.

Examen

Cuando elige la opción `Scan`, el sistema utiliza el documento `AWS-RunPatchBaseline` para determinar el estado de conformidad de las revisiones del nodo administrado y notifica esta información a Patch Manager. `Scan` no solicita que se instalen actualizaciones ni que se reinicien los nodos administrados. En lugar de ello, la operación identifica las actualizaciones aprobadas que faltan y que son aplicables al nodo.

Instalación

Al elegir la opción `Install`, `AWS-RunPatchBaselineWithHooks` intenta instalar las actualizaciones aprobadas y aplicables que faltan en el nodo administrado. La información de conformidad de revisiones generada como parte de una operación `Install` no muestra las actualizaciones que faltan, pero podría notificar aquellas que presentan un estado de error si la instalación de la actualización no se ha podido realizar por cualquier motivo. Cuando una actualización se instala en un nodo administrado, este se reinicia para garantizar que la actualización esté instalada y activa. (Excepción: si el parámetro `RebootOption` se configura en `NoReboot` en el documento `AWS-RunPatchBaselineWithHooks`, el nodo administrado no se reinicia una vez que se ejecuta Patch Manager. Para obtener más información, consulte Nombre del parámetro: `RebootOption`).

> **Note**
>
> Si se instala una revisión especificada por las reglas de la línea de base antes de que Patch Manager actualice el nodo administrado, es posible que el sistema no se reinicie como es debido. Esto puede ocurrir cuando un usuario instala manualmente una revisión o cuando la instala automáticamente otro programa, como el `unattended-upgrades` paquete de Ubuntu Server.

Nombre del parámetro: **`Snapshot ID`**

Usage: opcional.

`Snapshot ID` es un ID exclusivo (GUID) que utiliza Patch Manager para garantizar que un conjunto de nodos administrados en los que se aplican revisiones en una sola operación tenga el mismo conjunto de revisiones aprobadas. Aunque el parámetro se define como opcional, nuestra mejor práctica recomendada depende de si se ejecuta o no `AWS-RunPatchBaselineWithHooks` en un periodo de mantenimiento, tal como se describe en la siguiente tabla.

Prácticas recomendadas de **`AWS-RunPatchBaselineWithHooks`**

Mode	Práctica recomendada	Detalles
Ejecución de `AWS-RunPatchBaselineWithHooks` dentro de un periodo de mantenimiento	No proporcione un ID de instantánea, sino que Patch Manager lo hará en su lugar.	Si utiliza un periodo de mantenimiento para ejecutar `AWS-RunPatchBaselineWithHooks`, no debe proporcionar su propio ID de instantánea generado. En este caso, Systems Manager proporciona un valor GUID en función del ID de ejecución del periodo de mantenimiento. De este modo, se garantiza que se utilice un ID correcto para todas las invocaciones de `AWS-RunPatchBaseli`

Mode	Práctica recomendada	Detalles
		`neWithHooks` en dicho periodo de mantenimiento. Si especifica un valor en este caso, tenga en cuenta que la instantánea de la línea de base de revisiones no podría mantenerse durante más de tres días. Después de eso, se generará una nueva instantán ea, aunque especifique el mismo ID después de que expire la instantánea.

Mode	Práctica recomendada	Detalles
Ejecución de `AWS-RunPatchBaselineWithHooks` fuera de un periodo de mantenimiento	Genere y especifique un valor GUID personalizado para el ID de instantánea.[1]	Cuando no esté utilizando un periodo de mantenimiento para ejecutar `AWS-RunPatchBaselineWithHooks`, se recomienda generar y especificar un ID de instantánea exclusivo por cada línea de base de revisiones, especialmente si ejecuta el documento `AWS-RunPatchBaselineWithHooks` en varios nodos administrados durante la misma operación. Si no especifica un ID en este caso, Systems Manager genera otro ID de instantánea para cada nodo administrado al que se envía el comando. Esto podría generar diferentes conjuntos de revisiones que se especifican entre los nodos. Por ejemplo, supongamos que se ejecuta el documento `AWS-RunPatchBaselineWithHooks` directamente a través de Run Command, una herramienta de AWS Systems Manager, y selecciona como destino un grupo de 50 nodos administrados. Al especificar un ID de instantánea personalizado, se genera una sola instantánea de la base de referencia

Mode	Práctica recomendada	Detalles
		que se utiliza para evaluar y aplicar revisiones en todos los nodos administrados, lo que garantiza que tengan un estado coherente.

[1] Puede utilizar cualquier herramienta que genere un GUID con el fin de crear un valor para el parámetro de ID de la instantánea. Por ejemplo, en PowerShell, puede utilizar el cmdlet `New-Guid` para generar un GUID con el formato `12345699-9405-4f69-bc5e-9315aEXAMPLE` .

Nombre del parámetro: **`RebootOption`**

Usage: opcional.

Opciones: `RebootIfNeeded` | `NoReboot`

Valor predeterminado: `RebootIfNeeded`

> **Warning**
>
> La opción predeterminada es `RebootIfNeeded`. Asegúrese de seleccionar la opción correcta para su caso de uso. Por ejemplo, si los nodos administrados deben reiniciarse inmediatamente para completar un proceso de configuración, elija `RebootIfNeeded`. O bien, si necesita mantener la disponibilidad de los nodos administrados hasta una hora de reinicio programada, elija `NoReboot`.

> **Important**
>
> No recomendamos el uso de Patch Manager para revisar las instancias de clústeres en Amazon EMR (antes denominado Amazon Elastic MapReduce). En concreto, no seleccione la opción `RebootIfNeeded` para el parámetro `RebootOption`. (Esta opción está disponible en los documentos de SSM Command para implementar revisiones `AWS-RunPatchBaseline`, `AWS-RunPatchBaselineAssociation`, y `AWS-RunPatchBaselineWithHooks`).

Los comandos subyacentes para implementar revisiones mediante el uso de Patch Manager y los comandos `yum` y `dnf`. Por lo tanto, las operaciones generan incompatibilidades debido a la forma en que se instalan los paquetes. Para obtener información sobre los métodos preferidos para actualizar el software en los clústeres de Amazon EMR, consulte Uso de la AMI predeterminada para Amazon EMR en la Guía de administración de Amazon EMR.

RebootIfNeeded

Cuando se elige la opción `RebootIfNeeded`, el nodo administrado se reinicia en cualquiera de los siguientes casos:

- Patch Manager instaló una o más revisiones.

  Patch Manager no evalúa si la revisión requiere llevar a cabo un reinicio. El sistema se reinicia aunque la revisión no requiera el reinicio.

- Patch Manager detecta uno o más revisiones con un estado de `INSTALLED_PENDING_REBOOT` durante la operación `Install`.

  El estado `INSTALLED_PENDING_REBOOT` puede indicar que la opción `NoReboot` se seleccionó la última vez que se ejecutó la operación `Install` o que se instaló un parche por fuera de Patch Manager desde la última vez que se reinició el nodo administrado.

El reinicio de los nodos administrados en estos dos casos garantiza que los paquetes actualizados se eliminen de la memoria y mantiene un comportamiento de aplicación de revisiones y reinicio consistente en todos los sistemas operativos.

NoReboot

Cuando elige la opción `NoReboot`, Patch Manager no reinicia un nodo administrado aunque haya instalado revisiones durante la operación `Install`. Esta opción es útil si sabe que los nodos administrados no requieren reinicio después de aplicar las revisiones, o si dispone de aplicaciones o procesos que se ejecutan en un nodo que no deberían verse interrumpidos como consecuencia del reinicio de una operación de aplicación de revisiones. También es útil cuando se desea tener más control sobre el tiempo de los reinicios del nodo administrado, por ejemplo, mediante un periodo de mantenimiento.

> **Note**
>
> Si elige la opción `NoReboot` y se ha instalado una revisión, se asigna a la revisión un estado de `InstalledPendingReboot`. Sin embargo, el nodo administrado en sí mismo está marcado como `Non-Compliant`. Una vez que se produce un reinicio y se ejecuta una operación `Scan`, el estado del nodo se actualiza a `Compliant`.

Archivo de seguimiento de instalación de revisiones: para realizar un seguimiento de la instalación de revisiones, especialmente las instaladas desde el último reinicio del sistema, Systems Manager mantiene un archivo en el nodo administrado.

> **Important**
>
> No elimine ni modifique el archivo de seguimiento. Si este archivo se elimina o está dañado, el informe de conformidad de revisiones para el nodo administrado es inexacto. Si esto sucede, reinicie el nodo y ejecute una operación de análisis de revisiones para restaurar el archivo.

Este archivo de seguimiento se almacena en las siguientes ubicaciones de los nodos administrados:

- Sistemas operativos Linux:
  - `/var/log/amazon/ssm/patch-configuration/patch-states-configuration.json`
  - `/var/log/amazon/ssm/patch-configuration/patch-inventory-from-last-operation.json`
- Sistema operativo Windows Server:
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchStatesConfiguration.json`
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchInventoryFromLastOperation.json`

Nombre del parámetro: **`PreInstallHookDocName`**

Usage: opcional.

Valor predeterminado: `AWS-Noop`.

El valor que se debe proporcionar para el parámetro `PreInstallHookDocName` es el nombre o el nombre de recurso de Amazon (ARN) de un documento de SSM de su elección. Puede proporcionar el nombre de un documento administrado por AWS o el nombre o ARN de un documento de SSM personalizado que haya creado o que le hayan compartido. (En el caso de un documento de SSM que le hayan compartido desde una Cuenta de AWS diferente, deberá especificar el ARN completo del recurso, como `arn:aws:ssm:us-east-2:123456789012:document/MySharedDocument`).

El documento de SSM que se especifica se ejecuta antes de la operación `Install` y lleva a cabo cualquier acción admitida por SSM Agent, como un script de shell que verifique la comprobación de estado de la aplicación antes de implementar revisiones en el nodo administrado. (Para ver la lista de acciones, consulte Referencia de complementos del documento de comandos). El nombre del documento de SSM predeterminado es `AWS-Noop`, el cual no realiza ninguna operación en el nodo administrado.

Para obtener información acerca de cómo crear un documento de SSM personalizado, consulte Crear contenido en el documento de SSM.

Nombre del parámetro: **`PostInstallHookDocName`**

Usage: opcional.

Valor predeterminado: `AWS-Noop`.

El valor que se debe proporcionar para el parámetro `PostInstallHookDocName` es el nombre o el nombre de recurso de Amazon (ARN) de un documento de SSM de su elección. Puede proporcionar el nombre de un documento administrado por AWS o el nombre o ARN de un documento de SSM personalizado que haya creado o que le hayan compartido. (En el caso de un documento de SSM que le hayan compartido desde una Cuenta de AWS diferente, deberá especificar el ARN completo del recurso, como `arn:aws:ssm:us-east-2:123456789012:document/MySharedDocument`).

El documento de SSM que se especifica se ejecuta después de la operación `Install with NoReboot` y realiza cualquier acción admitida por SSM Agent, como un script de shell que permite instalar actualizaciones de terceros antes de proceder al reinicio. (Para ver la lista de acciones, consulte Referencia de complementos del documento de comandos). El nombre del documento de SSM predeterminado es `AWS-Noop`, el cual no realiza ninguna operación en el nodo administrado.

Para obtener información acerca de cómo crear un documento de SSM personalizado, consulte Crear contenido en el documento de SSM.

Nombre del parámetro: **`OnExitHookDocName`**

Usage: opcional.

Valor predeterminado: `AWS-Noop`.

El valor que se debe proporcionar para el parámetro `OnExitHookDocName` es el nombre o el nombre de recurso de Amazon (ARN) de un documento de SSM de su elección. Puede proporcionar el nombre de un documento administrado por AWS o el nombre o ARN de un documento de SSM personalizado que haya creado o que le hayan compartido. (En el caso de un documento de SSM que le hayan compartido desde una Cuenta de AWS diferente, deberá especificar el ARN completo del recurso, como `arn:aws:ssm:us-east-2:123456789012:document/MySharedDocument`).

El documento de SSM que se especifica se ejecuta después de la operación de reinicio del nodo administrado y lleva a cabo cualquier acción admitida por SSM Agent, como un script de shell que compruebe el estado del nodo una vez completada la operación de aplicación de revisiones. (Para ver la lista de acciones, consulte Referencia de complementos del documento de comandos). El nombre del documento de SSM predeterminado es `AWS-Noop`, el cual no realiza ninguna operación en el nodo administrado.

Para obtener información acerca de cómo crear un documento de SSM personalizado, consulte Crear contenido en el documento de SSM.

Ejemplo de escenario para utilizar el parámetro InstallOverrideList en **`AWS-RunPatchBaseline`** o **`AWS-RunPatchBaselineAssociation`**

Puede utilizar el parámetro `InstallOverrideList` cuando desee anular las revisiones especificadas por la línea de base de revisiones predeterminada actual en Patch Manager, una herramienta de AWS Systems Manager. Este tema proporciona una serie de ejemplos que muestran cómo utilizar este parámetro para obtener los siguientes resultados:

- Aplique diferentes conjuntos de revisiones a un grupo de nodos administrados de destino.
- Aplique estos conjuntos de revisiones en diferentes frecuencias.
- Utilice la misma línea de base de revisiones para ambas operaciones.

Imagine que desea instalar dos categorías de revisiones distintas en los nodos administrados de Amazon Linux 2. Quiere instalar estas revisiones en diferentes programaciones mediante períodos de mantenimiento. Quiere que se ejecute un período de mantenimiento todas las semanas y que instale todas las revisiones de `Security`. Quiere que se ejecute otro período de mantenimiento

una vez al mes y que instale todas las revisiones disponibles, o revisiones de categorías distintas a `Security`.

Sin embargo, solo se puede definir una línea de base de revisiones a la vez como la predeterminada de un sistema operativo. Este requisito ayuda a evitar situaciones en las que una línea de base de revisiones aprueba una revisión mientras que otra la bloquea, lo que puede provocar problemas entre versiones en conflicto.

La siguiente estrategia le permite utilizar el parámetro `InstallOverrideList` para aplicar diferentes tipos de revisiones a un grupo de destino, en diferentes programaciones, sin dejar de utilizar la misma línea de base de revisiones.

1. En la línea de base de revisiones predeterminada, asegúrese de que solo se especifican las actualizaciones `Security`.
2. Cree un periodo de mantenimiento que ejecute `AWS-RunPatchBaseline` o `AWS-RunPatchBaselineAssociation` cada semana. No especifique una lista de anulación.
3. Cree una lista de anulación de las revisiones de todos los tipos que desee aplicar mensualmente y almacénela en un bucket de Amazon Simple Storage Service (Amazon S3).
4. Cree un segundo período de mantenimiento que se ejecute una vez al mes. Sin embargo, para la tarea Run Command que registre en este periodo de mantenimiento, especifique la ubicación de su lista de anulación.

El resultado: solo se instalan las revisiones de `Security` semanalmente, tal como se define en su línea de base de revisiones. Todas las revisiones disponibles, o cualquier subconjunto de revisiones que defina, se instalan cada mes.

Para obtener más información y muestras, consulte Nombre del parámetro: `InstallOverrideList`.

Uso del parámetro BaselineOverride

Puede definir las preferencias de parches en tiempo de ejecución mediante la característica de anulación de base de referencia de parches en Patch Manager, una herramienta de AWS Systems Manager. Para ello, especifique un bucket de Amazon Simple Storage Service (Amazon S3) que contenga un objeto JSON con una lista de bases de referencia de parches. La operación de aplicación de parches utiliza las bases de referencia proporcionadas en el objeto JSON que concuerda con el sistema operativo del host en lugar de aplicar las reglas de la base de referencia de parches predeterminada

> **Important**
>
> El nombre de archivo `BaselineOverride` no puede contener los siguientes caracteres: comillas invertidas (`), comillas simples ('), comillas dobles (") ni signos de dólar ($).

Salvo cuando una operación de revisión utiliza una política de revisión, el uso del parámetro `BaselineOverride` no sobrescribe la conformidad con la línea de base proporcionada en el parámetro. Los resultados de salida se registran en los registros Stdout de Run Command, una herramienta de AWS Systems Manager. Los resultados solo imprimen los paquetes marcados como `NON_COMPLIANT`. Esto significa que el paquete está marcado como `Missing`, `Failed`, `InstalledRejected`, o `InstalledPendingReboot`.

No obstante, cuando una operación de revisión utiliza una política de revisión, el sistema pasa el parámetro de anulación desde el bucket de S3 asociado y se actualiza el valor de conformidad del nodo administrado. Para obtener más información sobre los comportamientos de las políticas de revisiones, consulte Configuraciones de políticas de revisiones en Quick Setup.

Uso de la anulación de la base de referencia de parches con los parámetros de ID de instantánea o de lista de anulación de instalación

Se dan dos casos en los que la anulación de la base de referencia de parches presenta un comportamiento destacable.

Uso de la anulación de la base de referencia y el ID de la instantánea al mismo tiempo

Los ID de las instantáneas garantizan que todos los nodos administrados de un determinado comando de aplicación de revisiones apliquen lo mismo. Por ejemplo, si se aplican revisiones a 1000 nodos a la vez, estas serán las mismas.

Cuando se utiliza un ID de instantánea como una anulación de la base de referencia de parches, el ID de instantánea tiene prioridad sobre la anulación de la base de referencia de parches. Las reglas de anulación de la base de referencia se seguirán utilizando, pero solo se evaluarán una vez. En el ejemplo anterior, las revisiones en los 1000 nodos administrados continuarán siendo siempre las mismas. Si a mitad de la operación de aplicación de parches cambió el archivo JSON en el bucket de S3 referenciado para que sea diferente, los parches aplicados seguirán siendo los mismos. Esto se debe a que se proporcionó el ID de la instantánea.

Uso de la anulación de la base de referencia y de la lista de anulación de la instalación al mismo tiempo

No se pueden utilizar estos dos parámetros a la vez. El documento de aplicación de revisiones presenta un error si se proporcionan ambos parámetros, y no realiza ningún análisis ni instalación en el nodo administrado.

Ejemplos de código

En el siguiente ejemplo de código para Python, se muestra cómo generar la anulación de la base de referencia de parches.

```
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)
```

De este modo, se obtiene una anulación de la base de referencia de parches como la que se muestra a continuación.

```
[
    {
        "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": []
  }
]
```

líneas de base de revisiones

Los temas de esta sección proporcionan información acerca de cómo funcionan las líneas de base de revisiones en Patch Manager, una herramienta de AWS Systems Manager, cuando ejecuta una operación de `Scan` o `Install` en los nodos administrados.

Temas

- Líneas de base de revisiones personalizadas y predefinidas
- Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas
- Grupos de revisiones
- Revisiones de aplicaciones publicadas por Microsoft en Windows Server

Líneas de base de revisiones personalizadas y predefinidas

Patch Manager, una herramienta de AWS Systems Manager, proporciona líneas de base de revisiones predefinidas para todos los sistemas operativos compatibles con Patch Manager. Puede utilizar estas bases de referencia tal como están configuradas actualmente (no puede personalizarlas) o puede crear sus propias líneas de base de revisiones personalizadas. Las líneas de base de revisiones personalizadas le permiten tener un mayor control sobre qué revisiones se aprueban o rechazan en su entorno. Además, las bases de referencia predefinidas asignan un nivel de conformidad de `Unspecified` a todas las revisiones instalados con esas bases de referencia. Para asignar valores de conformidad, puede crear una copia de una base de referencia predefinida y especificar los valores de conformidad que desea asignar a las revisiones. Para obtener más información, consulte Líneas de base personalizadas y Uso de bases de referencia de parches personalizadas.

> **Note**
>
> La información de este tema aplica independientemente del método o tipo de configuración que utilice para sus operaciones de aplicación de revisiones:
>
> - Una política de revisiones configurada en Quick Setup
> - Una opción de administración de host configurada en Quick Setup
> - Una ventana de mantenimiento para ejecutar una revisión `Scan` o una tarea `Install`
> - Una operación Patch Now (Aplicar revisión ahora) bajo demanda

Temas

- Líneas de base predefinidas
- Líneas de base personalizadas

Líneas de base predefinidas

En la tabla siguiente se describe cada una de las líneas de base de revisiones predefinidas con Patch Manager.

Para obtener información acerca de qué versiones de cada sistema operativo admite Patch Manager, consulte Requisitos previos de Patch Manager.

Nombre	Sistemas operativos compatible	Detalles
`AWS-AlmaLinuxDefaultPatchBaseline`	AlmaLinux	Aprueba todas las revisiones del sistema operativo que están clasificadas como "Security" y con una gravedad de "Crítica" o "Importante". También aprueba todas las revisiones que están clasificadas como “Bugfix”. Las revisiones se aprueban automáticamente 7 días después de su lanzamiento o actualización.[1]
`AWS-AmazonLinuxDefaultPatchBaseline`	Amazon Linux 1	Aprueba todas las revisiones del sistema operativo que están clasificadas como "Security" y con una gravedad de "Crítica" o "Importante". También aprueba automáticamente todas las revisiones con una clasificación “Bugfix”. Las revisiones se aprueban automáticamente 7 días después de su lanzamiento o actualización.[1]

Nombre	Sistemas operativos compatible	Detalles
`AWS-AmazonLinux2DefaultPatchBaseline`	Amazon Linux 2	Aprueba todas las revisiones del sistema operativo que están clasificadas como "Security" y con una gravedad de "Crítica" o "Importante". También aprueba todas las revisiones con una clasificación “Bugfix”. Las revisiones se aprueban automáticamente siete días después de su lanzamiento.[1]
`AWS-AmazonLinux2022DefaultPatchBaseline`	Amazon Linux 2022	Aprueba todas las revisiones del sistema operativo que están clasificadas como "Security" y con una gravedad de "Crítica" o "Importante". Las revisiones se aprueban automáticamente siete días después de su publicación. También aprueba todas las revisiones con una clasificación de "Bugfix" siete días después de su publicación.

Nombre	Sistemas operativos compatible	Detalles
`AWS-AmazonLinux2023DefaultPatchBaseline`	Amazon Linux 2023	Aprueba todas las revisiones del sistema operativo que están clasificadas como "Security" y con una gravedad de "Crítica" o "Importante". Las revisiones se aprueban automáticamente siete días después de su publicación. También aprueba todas las revisiones con una clasificación de "Bugfix" siete días después de su publicación.
`AWS-CentOSDefaultPatchBaseline`	CentOS y CentOS Stream	Aprueba todas las actualizaciones siete días después de que estén disponibles (incluidas las actualizaciones que no son de seguridad).
`AWS-DebianDefaultPatchBaseline`	Debian Server	Aprueba inmediatamente todas las revisiones relacionadas con la seguridad del sistema operativo que tienen una prioridad de "Obligatorio", "Importante", "Estándar", "Opcional" o "Extra" No hay ninguna espera antes de la aprobación porque en los repositorios no hay fechas de lanzamiento fiables.

Nombre	Sistemas operativos compatible	Detalles
`AWS-MacOSDefaultPatchBaseline`	macOS	Aprueba todas las revisiones del sistema operativo que están clasificadas como “Security” (Seguridad). También aprueba todos los paquetes con una actualización vigente.
`AWS-OracleLinuxDefaultPatchBaseline`	Oracle Linux	Aprueba todas las revisiones del sistema operativo que están clasificadas como "Security" y con una gravedad de "Importante" o "Moderada". También aprueba todas las revisiones clasificadas como “Bugfix” siete días después de su lanzamiento. Las revisiones se aprueban automáticamente 7 días después de su lanzamiento o actualización.[1]
`AWS-DefaultRaspbianPatchBaseline`	Raspberry Pi OS	Aprueba inmediatamente todas las revisiones relacionadas con la seguridad del sistema operativo que tienen una prioridad de "Obligatorio", "Importante", "Estándar", "Opcional" o "Extra" No hay ninguna espera antes de la aprobación porque en los repositorios no hay fechas de lanzamiento fiables.

Nombre	Sistemas operativos compatible	Detalles
`AWS-RedHatDefaultPatchBaseline`	Red Hat Enterprise Linux (RHEL)	Aprueba todas las revisiones del sistema operativo que están clasificadas como "Security" y con una gravedad de "Crítica" o "Importante". También aprueba todas las revisiones que están clasificadas como “Bugfix”. Las revisiones se aprueban automáticamente 7 días después de su lanzamiento o actualización.[1]
`AWS-RockyLinuxDefaultPatchBaseline`	Rocky Linux	Aprueba todas las revisiones del sistema operativo que están clasificadas como "Security" y con una gravedad de "Crítica" o "Importante". También aprueba todas las revisiones que están clasificadas como “Bugfix”. Las revisiones se aprueban automáticamente 7 días después de su lanzamiento o actualización.[1]

Nombre	Sistemas operativos compatible	Detalles
`AWS-SuseDefaultPatchBaseline`	SUSE Linux Enterprise Server (SLES)	Aprueba todas las revisiones del sistema operativo que están clasificadas como "Security" y con una gravedad "Crítica" o "Importante". Las revisiones se aprueban automáticamente 7 días después de su lanzamiento o actualización.[1]
`AWS-UbuntuDefaultPatchBaseline`	Ubuntu Server	Aprueba inmediatamente todas las revisiones relacionadas con la seguridad del sistema operativo que tienen una prioridad de "Obligatorio", "Importante", "Estándar", "Opcional" o "Extra" No hay ninguna espera antes de la aprobación porque en los repositorios no hay fechas de lanzamiento fiables.
`AWS-DefaultPatchBaseline`	Windows Server	Aprueba todas las revisiones del sistema operativo Windows Server que están clasificados como "Critical Updates" o "SecurityUpdates" y que tienen una gravedad de MSRC de "Crítica" o "Importante". Las revisiones se aprueban automáticamente 7 días después de su lanzamiento o actualización.[2]

Nombre	Sistemas operativos compatible	Detalles
`AWS-WindowsPredefinedPatchBaseline-OS`	Windows Server	Aprueba todas las revisiones del sistema operativo Windows Server que están clasificados como "Critical Updates" o "SecurityUpdates" y que tienen una gravedad de MSRC de "Crítica" o "Importante". Las revisiones se aprueban automáticamente 7 días después de su lanzamiento o actualización.[2]
`AWS-WindowsPredefinedPatchBaseline-OS-Applications`	Windows Server	Para el sistema operativo Windows Server, aprueba todas las revisiones que están clasificadas como "Critical Updates" o "SecurityUpdates" y que tienen una gravedad de MSRC de "Crítica" o "Importante". Para aplicaciones publicadas por Microsoft, aprueba todas las revisiones. Las revisiones para sistemas operativos y aplicaciones se aprueban automáticamente 7 días después de su lanzamiento o actualización.[2]

[1] Para Amazon Linux 1 y Amazon Linux 2, la espera de siete días antes de la aprobación automática de los parches se calcula a partir de un valor `Updated Date` en `updateinfo.xml`, no a partir de un valor `Release Date`. Hay varios factores que pueden afectar al valor de `Updated Date`. Otros sistemas operativos administran las fechas de lanzamiento y actualización de forma diferente. Para obtener información que le ayude a evitar resultados inesperados en el tiempo de espera hasta la

aprobación automática, consulte Cómo se calculan las fechas de lanzamiento y actualización de los paquetes.

² Para Windows Server, las líneas de base predeterminadas incluyen un retraso de 7 días en la aprobación automática. Para instalar una revisión dentro de los 7 días posteriores al lanzamiento, debe crear una línea de base personalizada.

Líneas de base personalizadas

Utilice la siguiente información para ayudarle a crear líneas de base de revisiones personalizadas para cumplir sus objetivos de aplicación de revisiones.

Temas

- Cómo utilizar aprobaciones automáticas en líneas de base personalizadas
- Información adicional para crear líneas de base de revisiones

Cómo utilizar aprobaciones automáticas en líneas de base personalizadas

Si crea su propia línea de base de revisiones, puede elegir qué revisiones se aprueban automáticamente mediante las categorías siguientes.

- Sistema operativo: Windows Server, Amazon Linux, Ubuntu Server, etcétera.
- Nombre de producto (para sistemas operativos): por ejemplo, RHEL 6.5, Amazon Linux 2014.09, Windows Server 2012, Windows Server 2012 R2, etcétera.
- Nombre del producto (solo para las aplicaciones publicadas por Microsoft que se ejecutan en Windows Server): por ejemplo, Word 2016, BizTalk Server, etc.
- Clasificación: por ejemplo: actualizaciones críticas, actualizaciones de seguridad, etc.
- Gravedad: por ejemplo: crítica, importante, etc.

Para cada regla de aprobación que cree, puede elegir especificar un retraso de aprobación automática o una fecha límite de aprobación de revisiones.

> **Note**
>
> Debido a que no es posible determinar de forma fiable las fechas de lanzamiento de los paquetes de actualización para Ubuntu Server, las opciones de aprobación automática no son compatibles con este sistema operativo.

Un tiempo de espera hasta la aprobación automática es el número de días que se debe esperar después de que se haya lanzado o actualizado la revisión y antes de que se apruebe automáticamente su aplicación Por ejemplo, si crea una regla que use la clasificación `CriticalUpdates` y la configura con un tiempo de espera hasta la aprobación automática de siete días, una nueva revisión crítica lanzada el 7 de julio se aprobará automáticamente el 14 de julio.

Si un repositorio de Linux no proporciona información sobre la fecha de publicación de los paquetes, Patch Manager utiliza el tiempo de compilación como la fecha de las especificaciones de aprobación automática para Amazon Linux 1, Amazon Linux 2, SUSE Linux Enterprise Server (SLES), Red Hat Enterprise Linux (RHEL) y CentOS. Si no se puede determinar el tiempo de compilación del paquete, Patch Manager utiliza una fecha predeterminada del 1 de enero de 1970. Esto resulta en que Patch Manager omita cualquier especificación de fecha de aprobación automática en las líneas base de revisiones que estén configuradas para aprobar revisiones en cualquier fecha posterior al 1 de enero de 1970.

Al especificar una fecha límite de aprobación automática, Patch Manager aplica automáticamente todas las revisiones lanzadas o actualizadas en esa fecha o antes de ella. Por ejemplo, si especifica el 7 de julio de 2023 como fecha límite, no se instalarán automáticamente las revisiones lanzadas o actualizadas por última vez a partir del 8 de julio de 2023.

Cuando crea una línea de base de revisiones personalizada, puede especificar un nivel de gravedad de conformidad para las revisiones aprobadas por esa línea de base de revisiones, como `Critical` o `High`. Si se informa del estado de cualquier revisión aprobada como `Missing`, la gravedad de la conformidad general notificada por la línea de base de revisiones será el nivel de gravedad que haya especificado.

Información adicional para crear líneas de base de revisiones

Tenga en cuenta lo siguiente cuando cree una línea de base de revisiones:

- Patch Manager proporciona una línea de base de revisiones predefinida para cada sistema operativo admitido. Estas líneas de base de revisiones predefinidas se utilizan como las líneas de base de revisiones predeterminadas para cada tipo de sistema operativo a menos que cree su propia línea de base de revisiones y la designe como la opción predeterminada para el tipo de sistema operativo correspondiente.

> **Note**
>
> Para Windows Server, se proporcionan tres líneas de base de revisiones predefinidas. Las líneas de base de revisiones `AWS-DefaultPatchBaseline` y `AWS-WindowsPredefinedPatchBaseline-OS` solo admiten actualizaciones del sistema operativo en el propio sistema operativo Windows. `AWS-DefaultPatchBaseline` se utiliza como base de referencia de revisiones predeterminada para los nodos administrados de Windows Server, a menos que se especifique una base de referencia distinta. Los ajustes de configuración en estas dos líneas de base de revisiones son los mismos. El más nuevo de los dos, `AWS-WindowsPredefinedPatchBaseline-OS`, se creó para que se diferenciara de la tercera línea de base de revisiones predefinida para Windows Server. Esa línea de base de revisiones, `AWS-WindowsPredefinedPatchBaseline-OS-Applications`, puede utilizarse para aplicar revisiones al sistema operativo Windows Server y a las aplicaciones compatibles publicadas por Microsoft.

- De forma predeterminada, en Windows Server 2019 y Windows Server 2022 se eliminan las actualizaciones que se sustituyen por actualizaciones posteriores. Como resultado, si utiliza el parámetro `ApproveUntilDate` en la línea de base de revisiones de Windows Server, pero la fecha seleccionada en el parámetro `ApproveUntilDate` es anterior a la fecha de la última revisión, la nueva revisión no se instalará cuando se ejecute la operación de revisiones. Para obtener más información sobre la creación de reglas de revisiones en Windows Server, consulte la pestaña Windows Server en Cómo se seleccionan las revisiones de seguridad.

  Esto significa que el nodo gestionado es compatible con las operaciones de Systems Manager, aunque es posible que no se haya instalado una revisión crítica del mes anterior. Esta misma situación puede ocurrir cuando se utiliza el parámetro `ApproveAfterDays`. Debido al comportamiento de las revisiones sustituidas por Microsoft, es posible establecer un número (generalmente superior a 30 días) para que las revisiones para Windows Server no se instalen nunca si la última revisión disponible de Microsoft se publica antes de que hayan transcurrido los días en `ApproveAfterDays`.
- Solo para Windows Server, la revisión de actualización de seguridad disponible que no esté aprobada por la línea de base de revisiones puede tener un valor de cumplimiento (`Compliant` o `Non-Compliant`) tal como se define en una línea de base de revisiones personalizada.

Al crear o actualizar una línea de base de revisiones, usted elige el estado que desee asignar a las revisiones de seguridad que están disponibles pero no aprobadas porque no cumplen con los criterios de instalación especificados en la línea de base de revisiones. Por ejemplo, las revisiones de seguridad que desee instalar se pueden omitir si ha especificado un periodo prolongado de espera después del lanzamiento de la revisión antes de la instalación. Si se publica una actualización de la revisión durante el periodo de espera especificado, el periodo de espera para instalar la revisión vuelve a comenzar. Si el periodo de espera es demasiado extenso, es posible que se lancen varias versiones de la revisión, pero nunca se instalen.

Si utiliza la consola para crear o actualizar una línea de base de revisiones, especifique esta opción en el campo Estado de cumplimiento de las actualizaciones de seguridad disponibles. Si utiliza la AWS CLI para ejecutar el comando create-patch-baseline o update-patch-baseline, especifique esta opción en el parámetro `available-security-updates-compliance-status`.

- Para los servidores locales y las máquinas virtuales, Patch Manager intenta utilizar su línea de base de revisiones predeterminada personalizada. Si no existe una línea de base de revisiones predeterminada personalizada, el sistema usa la línea de base de revisiones predefinida para el sistema operativo correspondiente.
- Si una revisión aparece como aprobado y rechazado en la misma línea de base de revisiones, se rechaza.
- Un nodo administrado solo puede tener una base de referencia de revisiones definida para él.
- Los formatos de nombres de paquetes que se pueden agregar a las Listas de revisiones aprobados y rechazados en una línea de base de revisiones dependen del tipo de sistema operativo al que se le apliquen las revisiones.

  Para obtener información acerca de los formatos aceptados para las Listas de revisiones aprobados y rechazados, consulte Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas.
- Si utiliza una configuración de política de revisiones en Quick Setup, las actualizaciones que realice en las líneas de base de revisiones personalizadas se sincronizan con Quick Setup cada hora.

  Si se elimina una línea de base de revisiones personalizada a la que se hacía referencia en una política de revisiones, aparece un banner en la página Configuration details (Detalles de configuración) de Quick Setup correspondiente a la política de revisiones. El banner le informa que la política de revisiones hace referencia a una línea de base de revisiones que ya no existe y que

las operaciones de aplicación de revisiones posteriores fallarán. En este caso, vuelva a la página Configurations (Configuraciones) de Quick Setup, seleccione la configuración de Patch Manager y elija Actions (Acciones), Edit configuration (Editar configuración). El nombre de la línea de base de revisiones eliminado aparece resaltado y debe seleccionar una nueva línea de base de revisiones para el sistema operativo afectado.

Para obtener más información sobre cómo crear una línea de base de revisiones, consulte Uso de bases de referencia de parches personalizadas y Tutorial: implementación de revisiones en un entorno de servidores mediante la AWS CLI.

Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas

Los formatos de nombres de paquetes que se pueden agregar a las listas de parches aprobados y rechazados dependen del tipo de sistema operativo al que se le apliquen los parches.

Formatos de nombre de paquete para sistemas operativos Linux

Los formatos que puede especificar para parches aprobados y rechazados en su base de referencia de parches varían en función del tipo de Linux. En concreto, los formatos que se admiten dependen del administrador de paquetes utilizados por el tipo de sistema operativo Linux.

Temas

- Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022, Amazon Linux 2023, CentOS, Oracle Linux y Red Hat Enterprise Linux (RHEL)
- Debian Server, Raspberry Pi OS (anteriormente Raspbian) y Ubuntu Server
- SUSE Linux Enterprise Server (SLES)

Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022, Amazon Linux 2023, CentOS, Oracle Linux y Red Hat Enterprise Linux (RHEL)

Administrador de paquetes: YUM, excepto para Amazon Linux 2022, Amazon Linux 2023, RHEL 8 y CentOS 8, que utilizan DNF como administrador de paquetes

Parches aprobados: para este tipo de parches, puede especificar cualquiera de los siguientes elementos:

- ID de Bugzilla, en el formato `1234567` (el sistema procesa solo números de cadenas como los ID de Bugzilla).

- ID de CVE con el formato `CVE-2018-1234567`
- ID de Advisory con formatos como `RHSA-2017:0864` y `ALAS-2018-123`
- Nombres de paquetes que se crean con uno o varios de los componentes disponibles para la nomenclatura de paquetes. A modo ilustrativo, para el paquete denominado `dbus.x86_64:1:1.12.28-1.amzn2023.0.1`, los componentes son los siguientes:
  - `name`: `dbus`
  - `architecture`: `x86_64`
  - `epoch`: `1`
  - `version`: `1.12.28`
  - `release`: `1.amzn2023.0.1`

  Se admiten nombres de paquetes con las siguientes construcciones:
  - `name`
  - `name.arch`
  - `name-version`
  - `name-version-release`
  - `name-version-release.arch`
  - `version`
  - `version-release`
  - `epoch:version-release`
  - `name-epoch:version-release`
  - `name-epoch:version-release.arch`
  - `epoch:name-version-release.arch`
  - `name.arch:epoch:version-release`

  Presentamos algunos ejemplos:
  - `dbus.x86_64`
  - `dbus-1.12.28`
  - `dbus-1.12.28-1.amzn2023.0.1`
  - `dbus-1:1.12.28-1.amzn2023.0.1.x86_64`
  - También se admiten componentes de nombres de paquetes con un único comodín en los formatos anteriores, como los siguientes:

- dbus*
- dbus-1.12.2*
- dbus-*:1.12.28-1.amzn2023.0.1.x86_64

Parches rechazados: para este tipo de parches, puede especificar cualquiera de los siguientes elementos:

- Nombres de paquetes que se crean con uno o varios de los componentes disponibles para la nomenclatura de paquetes. A modo ilustrativo, para el paquete denominado dbus.x86_64:1:1.12.28-1.amzn2023.0.1, los componentes son los siguientes:
  - name: dbus
  - architecture; x86_64
  - epoch: 1
  - version: 1.12.28
  - release: 1.amzn2023.0.1

  Se admiten nombres de paquetes con las siguientes construcciones:
  - name
  - name.arch
  - name-version
  - name-version-release
  - name-version-release.arch
  - version
  - version-release
  - epoch:version-release
  - name-epoch:version-release
  - name-epoch:version-release.arch
  - epoch:name-version-release.arch
  - name.arch:epoch:version-release

  Presentamos algunos ejemplos:
  - dbus.x86_64
  - dbus-1.12.28

  - `dbus-1.12.28-1.amzn2023.0.1`
  - `dbus-1:1.12.28-1.amzn2023.0.1.x86_64`
- También se admiten componentes de nombres de paquetes con un único comodín en los formatos anteriores, como los siguientes:
  - `dbus*`
  - `dbus-1.12.2*`
  - `dbus-*:1.12.28-1.amzn2023.0.1.x86_64`

Debian Server, Raspberry Pi OS (anteriormente Raspbian) y Ubuntu Server

Administrador de paquetes: APT

Parches aprobados y rechazados: para estos tipos de parches, especifique lo siguiente:

- Nombres de paquete en el formato `ExamplePkg33`

> **Note**
>
> Para las listas de Debian Server, Raspberry Pi OS y Ubuntu Server, no incluya elementos como la arquitectura o las versiones. Por ejemplo, especifique el nombre del paquete `ExamplePkg33` para incluir todo lo siguiente en la lista de parches:
>
> - `ExamplePkg33.x86.1`
> - `ExamplePkg33.x86.2`
> - `ExamplePkg33.x64.1`
> - `ExamplePkg33.3.2.5-364.noarch`

SUSE Linux Enterprise Server (SLES)

Administrador de paquetes: Zypper

Parches aprobados y rechazados: para estos tipos de parches, puede especificar cualquiera de los elementos siguientes:

- Nombres de paquetes completos, con formatos como:
  - `SUSE-SLE-Example-Package-12-2018-123`

  - `example-pkg-2018.11.4-46.17.1.x86_64.rpm`
- Nombres de formatos con un solo comodín, como:
  - `SUSE-SLE-Example-Package-12-2018-*`
  - `example-pkg-2018.11.4-46.17.1.*.rpm`

Formatos de nombres de paquetes para macOS

Administradores de paquetes admitidos: softwareupdate, installer, Brew, Brew Cask

Parches aprobados y parches rechazados: en las listas de estos tipos de parches, debe especificar los nombres completos de los paquetes en formatos tales como los siguientes:

- `XProtectPlistConfigData`
- `MRTConfigData`

No se admiten comodines en las listas de parches aprobados y rechazados para macOS.

Formatos de nombre de paquete para sistemas operativos Windows

Para sistemas operativos de Windows, especifique los parches con los ID de la Base de conocimientos de Microsoft y del boletín de seguridad de Microsoft; por ejemplo:

```
KB2032276,KB2124261,MS10-048
```

Grupos de revisiones

> **Note**
>
> Los grupos de revisiones no se usan en operaciones de aplicación de revisiones basadas en políticas de revisiones. Para obtener información sobre el uso de las políticas de revisiones, consulte Configuraciones de políticas de revisiones en Quick Setup.
> La consola no admite la funcionalidad de grupos de parches para pares de cuentas y regiones que no usaban grupos de parches antes de que se publicara la compatibilidad con las políticas de parches el 22 de diciembre de 2022. La funcionalidad de grupos de parches sigue estando disponible en los pares de cuentas y regiones que empezaron a usar grupos de parches antes de esta fecha.

Puede utilizar un grupo de revisiones para asociar nodos administrados a una línea de base de revisiones específica en Patch Manager, una herramienta de AWS Systems Manager. Los grupos de revisiones ayudan a garantizar que implementará las revisiones adecuadas al conjunto correcto de nodos en función de las reglas de base de referencia de revisiones asociadas. Los grupos de revisiones también pueden ayudarle a evitar la implementación de revisiones antes de que estos se hayan probado suficientemente. Por ejemplo, puede crear grupos de revisiones para diferentes entornos (como desarrollo, prueba y producción) y registrar cada grupo de revisiones en una línea de base de revisiones adecuada.

Cuando ejecuta `AWS-RunPatchBaseline` o cualquier otro documento de comandos SSM para revisiones, puede tomar como objetivo nodos administrados mediante sus ID o etiquetas. Luego, SSM Agent y Patch Manager evalúan qué línea de base de revisiones se utilizará en función del valor del grupo de revisiones que agregó al nodo administrado.

Uso de etiquetas para definir grupos de revisiones

Puede crear un grupo de revisiones con etiquetas aplicadas a sus instancias de Amazon Elastic Compute Cloud (Amazon EC2) y a nodos que no sean de EC2 en un entorno híbrido y multinube. Tenga en cuenta los siguientes detalles sobre el uso de etiquetas para los grupos de revisiones:

- Un grupo de revisiones debe definirse mediante la clave de etiqueta `Patch Group` o `PatchGroup` aplicada a sus nodos administrados. Al registrar un grupo de revisiones para una línea de base de revisiones, cualquier valor idéntico especificado para estas dos claves se interpreta como parte del mismo grupo. Por ejemplo, supongamos que etiquetó cinco nodos con el primero de los siguientes pares clave-valor y cinco con el segundo:
  - `key=PatchGroup,value=DEV`
  - `key=Patch Group,value=DEV`

  El comando Patch Manager para crear una referencia combina estos 10 nodos administrados en un único grupo en función del valor `DEV`. El equivalente AWS CLI del comando en la creación de una línea de base de revisiones para grupos de parches es el siguiente:

  ```
  aws ssm register-patch-baseline-for-patch-group \
      --baseline-id pb-0c10e65780EXAMPLE \
      --patch-group DEV
  ```

  La combinación de valores de diferentes claves en un solo objetivo es exclusiva de este comando Patch Manager para crear un nuevo grupo de revisiones y no es compatible con otras acciones

de la API. Por ejemplo, si ejecuta acciones send-command con las claves `PatchGroup` y `Patch Group` con los mismos valores, se dirige a dos conjuntos de nodos completamente diferentes:

```
aws ssm send-command \
    --document-name AWS-RunPatchBaseline \
    --targets "Key=tag:PatchGroup,Values=DEV"
```

```
aws ssm send-command \
    --document-name AWS-RunPatchBaseline \
    --targets "Key=tag:Patch Group,Values=DEV"
```

- Existen límites en la segmentación basada en etiquetas. Cada matriz de destinos para `SendCommand` puede contener un máximo de cinco pares clave-valor.
- Le recomendamos que elija solo una de estas convenciones de claves de etiquetas, ya sea `PatchGroup` (sin espacio) o `Patch Group` (con espacio). Si ha permitido las etiquetas en los metadatos de las instancias de EC2, debe usar `PatchGroup`.
- La clave distingue entre mayúsculas y minúsculas. Puede especificar cualquier valor que le ayude a identificar y destinar los recursos de ese grupo, por ejemplo, "servidores web" o "US-EAST-PROD", pero la clave debe ser `Patch Group` o `PatchGroup`.

Después de crear un grupo de revisiones y etiquetar nodos administrados, puede registrar el grupo de revisiones con una base de referencia de revisiones. Registrar el grupo de revisiones en una base de referencia de revisiones garantiza que los nodos del grupo de revisiones utilicen las reglas definidas en la base de referencia de revisiones asociada.

Para obtener más información acerca de cómo crear un grupo de revisiones y asociarlo a una línea de base de revisiones, consulte Creación y administración de grupos de revisiones y Agregar un grupo de revisiones a una línea de base de revisiones.

Para ver un ejemplo de cómo crear una línea de base de revisiones y grupos de revisiones mediante la AWS Command Line Interface (AWS CLI), consulte Tutorial: implementación de revisiones en un entorno de servidores mediante la AWS CLI. Para obtener más información sobre las etiquetas de Amazon EC2, consulte Etiquetado de los recursos de Amazon EC2 en la Guía del usuario de Amazon EC2.

Funcionamiento

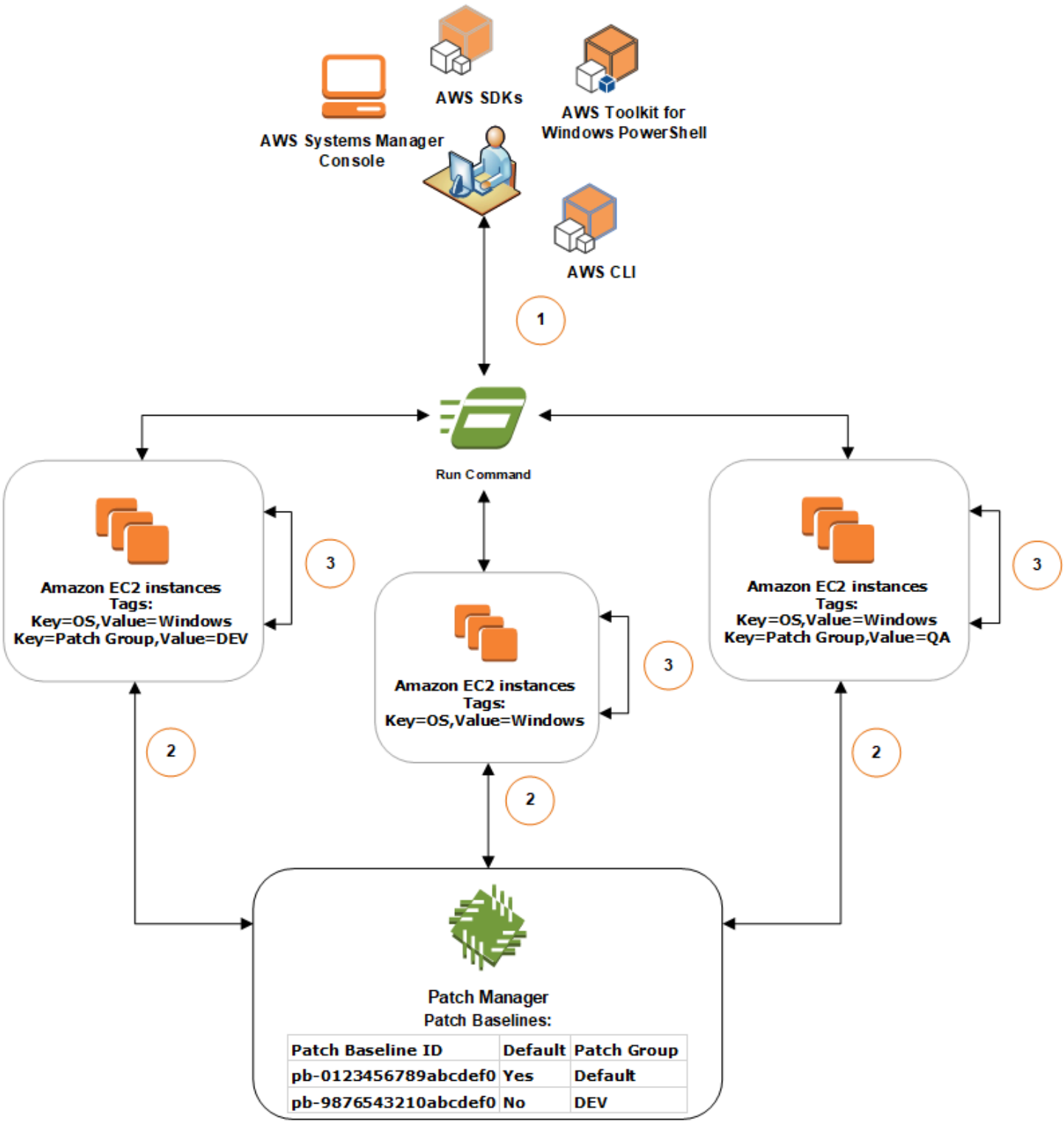
Cuando el sistema ejecuta la tarea para aplicar una base de referencia de revisiones a un nodo administrado, SSM Agent verifica si se ha definido un valor de grupo de revisiones para el nodo. Si se asigna el nodo a un grupo de revisiones, Patch Manager comprueba qué base de referencia de revisiones está registrada en ese grupo. Si se encuentra una línea de base de revisiones para ese grupo, Patch Manager indica a SSM Agent que utilice la línea de base de revisiones asociada. Si un nodo no está configurado para un grupo de revisiones, Patch Manager indica automáticamente a SSM Agent que debe utilizar la base de referencia de revisiones predeterminada que se encuentra configurada en la actualidad.

> **Important**
>
> Un nodo administrado solo puede estar en un grupo de revisiones.
> Un grupo de revisiones puede registrarse con solo una línea de base de revisiones para cada tipo de sistema operativo.
> Puede aplicar la etiqueta `Patch Group` (con un espacio) a una instancia de Amazon EC2 si la opción Allow tags in instance metadata (Permitir etiquetas en los metadatos de la instancia) no puede estar habilitada en la instancia. Al permitir etiquetas en los metadatos de la instancia, se impide que los nombres de las claves de las etiquetas contengan espacios. Si tiene etiquetas permitidas en metadatos de instancias de EC2, debe usar la clave de etiqueta `PatchGroup` (sin espacio).

Diagrama 1: ejemplo general de flujo de proceso de operaciones de aplicación de revisiones

En el siguiente diagrama, se muestra un ejemplo general de los procesos que Systems Manager lleva a cabo al enviar una tarea de Run Command a la flota de servidores a la que se aplicarán revisiones mediante Patch Manager. Estos procesos determinan qué líneas de base de revisiones que se deben usar en las operaciones de aplicación de revisiones. (Se emplea un proceso similar cuando se ha configurado un periodo de mantenimiento para enviar un comando que aplique revisiones mediante Patch Manager.

El proceso completo se explica debajo de la ilustración.



En este ejemplo, tenemos tres grupos de instancias EC2 de Windows Server con las siguientes etiquetas aplicadas:

Grupo de instancias EC2	Etiquetas
Grupo 1	`key=OS,value=Windows` `key=PatchGroup,value=DEV`
Grupo 2	`key=OS,value=Windows`
Grupo 3	`key=OS,value=Windows` `key=PatchGroup,value=QA`

En este ejemplo, también tenemos estas dos líneas de base de revisiones de Windows Server:

ID de línea de base de revisiones	Predeterminado/a	Grupo de revisiones asociado
`pb-0123456789abcdef0`	Sí	`Default`
`pb-9876543210abcdef0`	No	`DEV`

El proceso general de análisis o instalación de revisiones mediante Run Command, una herramienta de AWS Systems Manager, y Patch Manager es como se indica a continuación:

1. Envío de un comando para aplicar revisiones: utilice la consola de Systems Manager, el SDK, la AWS Command Line Interface (AWS CLI) o AWS Tools for Windows PowerShell para enviar una tarea de Run Command mediante el documento `AWS-RunPatchBaseline`. En el diagrama se muestra una tarea de Run Command para aplicar revisiones a instancias administradas al tomar como objetivo la etiqueta `key=OS,value=Windows`.
2. Determinación de la línea de base de revisiones: SSM Agent verifica las etiquetas de grupos de revisiones que se aplican a la instancia de EC2 y consulta a Patch Manager la línea de base de revisiones correspondiente.
   - Coincidencia del valor de un grupo de revisiones asociado con una línea de base de revisiones:
     1. SSM Agent, que está instalado en instancias EC2 en el grupo uno, recibe el comando ejecutado en el paso 1 para empezar una operación de aplicación de revisiones. SSM

Agent valida que las instancias EC2 tengan el valor de etiqueta `DEV` del grupo de revisiones aplicado y consulta a Patch Manager una línea de base de revisiones asociada.

2. Patch Manager verifica que la línea de base de revisiones `pb-9876543210abcdef0` tenga el grupo de revisiones `DEV` asociado y se lo notifica a SSM Agent.
3. SSM Agent recupera una instantánea de la línea de base de revisiones desde Patch Manager en función de las reglas de aprobación y las excepciones configuradas en `pb-9876543210abcdef0` y avanza al siguiente paso.

- La instancia no tiene una etiqueta de grupo de revisiones añadida:

  1. SSM Agent, que está instalado en instancias de EC2 en el grupo dos, recibe el comando ejecutado en el paso 1 para empezar una operación de aplicación de revisiones. SSM Agent valida que las instancias EC2 no tengan una etiqueta `Patch Group` o `PatchGroup` aplicada y, en consecuencia, SSM Agent consulta a Patch Manager la línea de base de revisiones de Windows predeterminada.
  2. Patch Manager verifica que la línea de base de revisiones de Windows Server predeterminada sea `pb-0123456789abcdef0` y se lo notifica a SSM Agent.
  3. SSM Agent recupera una instantánea de la línea de base de revisiones desde Patch Manager en función de las reglas de aprobación y las excepciones configuradas en la línea de base de revisiones predeterminada `pb-0123456789abcdef0` y avanza al siguiente paso.

- La línea de base de revisiones no tiene un valor de grupo de revisiones coincidente asociado:

  1. SSM Agent, que está instalado en instancias EC2 en el grupo tres, recibe el comando ejecutado en el paso 1 para empezar una operación de aplicación de revisiones. SSM Agent valida que las instancias EC2 tengan el valor de etiqueta `QA` del grupo de revisiones aplicado y consulta a Patch Manager una línea de base de revisiones asociada.
  2. Patch Manager no encuentra una línea de base de revisiones que tenga el grupo de revisiones `QA` asociado.
  3. Patch Manager indica a SSM Agent que debe usar la línea de base de revisiones de Windows predeterminada `pb-0123456789abcdef0`.
  4. SSM Agent recupera una instantánea de la línea de base de revisiones desde Patch Manager en función de las reglas de aprobación y las excepciones configuradas en la línea de base de revisiones predeterminada `pb-0123456789abcdef0` y avanza al siguiente paso.

3. Análisis de detección de revisiones o instalación de revisiones: después de determinar la línea de base de revisiones adecuada que se va a utilizar, SSM Agent comienza el análisis de detección de revisiones o la instalación de revisiones en función del valor de operación especificado en el paso 1. las revisiones que se analizan o se instalan se determinan a partir de las reglas de aprobación y las excepciones de revisiones que están definidas en la instantánea de la línea de base de revisiones que Patch Manager proporciona.

Más información

- Valores de estado de conformidad de las revisiones

Revisiones de aplicaciones publicadas por Microsoft en Windows Server

Utilice la información de este tema para prepararse para aplicar revisiones a las aplicaciones en Windows Server mediante Patch Manager, una herramienta de AWS Systems Manager.

Uso de parches en aplicaciones de Microsoft

La compatibilidad con el uso de revisiones para las aplicaciones en los nodos administrados por Windows Server se limita a las aplicaciones publicadas por Microsoft.

> **Note**
>
> En algunos casos, Microsoft lanza parches para las aplicaciones que no especifican una hora ni una fecha de actualización. En estos casos, se suministra una fecha y hora actualizadas de `01/01/1970` de forma predeterminada.

Bases de referencia de parches para usar parches en las aplicaciones publicadas por Microsoft

Para Windows Server, se proporcionan tres líneas de base de revisiones predefinidas. Las líneas de base de revisiones `AWS-DefaultPatchBaseline` y `AWS-WindowsPredefinedPatchBaseline-OS` solo admiten actualizaciones del sistema operativo en el propio sistema operativo Windows. `AWS-DefaultPatchBaseline` se utiliza como base de referencia de revisiones predeterminada para los nodos administrados de Windows Server, a menos que se especifique una base de referencia distinta. Los ajustes de configuración en estas dos líneas de base de revisiones son los mismos. El más nuevo de los dos, `AWS-WindowsPredefinedPatchBaseline-OS`, se creó para que se diferenciara de la tercera línea de base de revisiones predefinida para Windows Server. Esa base de referencia de parches, `AWS-`

`WindowsPredefinedPatchBaseline-OS-Applications`, puede utilizarse para aplicar parches al sistema operativo Windows Server y a las aplicaciones compatibles publicadas por Microsoft.

También puede crear una base de referencia de parches personalizada para actualizar aplicaciones publicadas por Microsoft en máquinas de Windows Server.

Compatibilidad con la aplicación de revisiones en aplicaciones lanzadas por Microsoft en servidores en las instalaciones, dispositivos periféricos, máquinas virtuales y otros nodos que no sean de EC2

Para revisar las aplicaciones lanzadas por Microsoft en máquinas virtuales (VM) y nodos administrados que no son de EC2, es necesario que active el nivel de instancias avanzadas. El uso del nivel de instancias avanzadas conlleva un cargo. Sin embargo, no hay ningún cargo adicional por usar revisiones en las aplicaciones lanzadas por Microsoft en instancias de Amazon Elastic Compute Cloud (Amazon EC2). Para obtener más información, consulte Configuración de los niveles de instancias.

Opción de actualización de Windows para “otros productos de Microsoft”

Para que Patch Manager pueda usar revisiones en las aplicaciones publicadas por Microsoft en los nodos administrados por Windows Server, se debe activar la opción de Windows Update Give me updates for other Microsoft products when I update Windows (Ofrecerme actualizaciones para otros productos de Microsoft cuando actualice Windows) en el nodo administrado.

Para obtener información acerca de cómo permitir esta opción en un único nodo administrado, consulte Actualización de Office con Microsoft Update en el sitio web de soporte técnico de Microsoft.

En el caso de una flota de nodos administrados que ejecuten la versión de Windows Server 2016 y posteriores, puede utilizar un objeto de política de grupo (GPO) para activar la configuración. En el editor de administración de políticas de grupo, vaya a Computer Configuration (Configuración del equipo), Administrative Templates (Plantillas administrativas), Windows Components (Componentes de Windows), Windows Updates (Actualizaciones de Windows) y, a continuación, elija Install updates for other Microsoft products (Instalar actualizaciones para otros productos de Microsoft). También se recomienda configurar el GPO con parámetros adicionales que impidan las actualizaciones automáticas no planificadas y los reinicios fuera de Patch Manager. Para obtener más información, consulte Configuring Automatic Updates in a Non-Active Directory Environment (Configuración de actualizaciones automáticas en un entorno que no es de Active Directory) en el sitio web de documentación técnica de Microsoft.

En el caso de una flota de nodos administrados que ejecuten la versión de Windows Server 2012 o 2012 R2, puede activar la opción mediante un script, según se describe en Habilitar y desactivar

Microsoft Update en Windows 7 mediante Script en el sitio web del blog de Microsoft Docs. Por ejemplo, podría hacer lo siguiente:

1. Guarde el script de la publicación de blog en un archivo.
2. Cargue el archivo en un bucket de Amazon Simple Storage Service (Amazon S3) o en otra ubicación accesible.
3. Utilice Run Command, una herramienta de AWS Systems Manager, para ejecutar el script en los nodos administrados mediante el documento de Systems Manager (documento de SSM) `AWS-RunPowerShellScript` con un comando similar al siguiente.

```
Invoke-WebRequest `
    -Uri "https://s3.aws-api-domain/amzn-s3-demo-bucket/script.vbs" `
    -Outfile "C:\script.vbs" cscript c:\script.vbs
```

Requisitos mínimos de los parámetros

Para incluir aplicaciones publicadas por Microsoft en su base de referencia de parches personalizada, debe, como mínimo, especificar el producto al que desea aplicar parches. El siguiente comando de la AWS Command Line Interface (AWS CLI) muestra los requisitos mínimos para aplicar parches a un producto como, 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 especifica la familia de productos de aplicaciones de Microsoft, cada producto que especifique debe ser un miembro compatible de la familia de productos seleccionados. Por ejemplo, para aplicar parches al producto "Active Directory Rights Management Services Client 2.0", debe especificar su familia de productos como "Active Directory" y no, por ejemplo, "Office" ni "SQL Server". En el siguiente comando de la AWS CLI, se muestra un correcto emparejamiento de familia de productos y producto.

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 recibe un mensaje de error sobre un emparejamiento de producto y familia incorrecto, consulte Asunto: pares de familia de productos y productos no coincidentes para obtener asistencia sobre cómo solucionar el problema.

Uso de Kernel Live Patching en nodos administrados de Amazon Linux 2

Kernel Live Patching para Amazon Linux 2 le permite aplicar revisiones de vulnerabilidad de seguridad y de errores críticos a un kernel de Linux en ejecución, sin reinicios ni interrupciones en las aplicaciones en ejecución. Esto le permite beneficiarse de una mejor disponibilidad de servicios y aplicaciones, a la vez que mantiene su infraestructura segura y actualizada. Kernel Live Patching

se admite en instancias de Amazon EC2, dispositivos de núcleo de AWS IoT Greengrass y máquinas virtuales locales que ejecutan Amazon Linux 2.

Para obtener información general sobre Kernel Live Patching, consulte Kernel Live Patching on AL2 en la Guía del usuario de Amazon Linux 2.

Después de activar Kernel Live Patching en un nodo administrado de Amazon Linux 2, puede utilizar Patch Manager, una herramienta de AWS Systems Manager, para aplicar revisiones en caliente del kernel al nodo administrado. El uso de Patch Manager es una alternativa al uso de los flujos de trabajo yum existentes en el nodo para aplicar las actualizaciones.

Antes de empezar

Para utilizar Patch Manager para aplicar revisiones en caliente del kernel a los nodos administrados de Amazon Linux 2, asegúrese de que los nodos se basan en la arquitectura y la versión del kernel correctas. Para obtener información, consulte Configuraciones admitidas y requisitos previos en la Guía del usuario de Amazon EC2.

Temas

- Kernel Live Patching mediante Patch Manager
- Cómo funciona Kernel Live Patching mediante Patch Manager
- Activación de Kernel Live Patching con Run Command
- Aplicación de actualizaciones en caliente del kernel mediante Run Command
- Desactivación de Kernel Live Patching con Run Command

Kernel Live Patching mediante Patch Manager

Actualización de la versión del kernel

> No es necesario reiniciar un nodo administrado después de aplicar una revisión en caliente del kernel. Sin embargo, AWS proporciona revisiones en caliente del kernel para una versión del kernel de Amazon Linux 2 durante un máximo de tres meses después de su lanzamiento. Después del período de tres meses, debe actualizar a una versión posterior del kernel para continuar recibiendo actualizaciones en caliente de kernel. Recomendamos utilizar un periodo de mantenimiento para programar un reinicio del nodo al menos una vez cada tres meses para solicitar la actualización de la versión del kernel.

Desinstalación de actualizaciones en caliente del kernel

las revisiones en caliente del kernel no se pueden desinstalar mediante Patch Manager. En su lugar, puede deshabilitar Kernel Live Patching, lo que elimina los paquetes RPM de las revisiones en caliente del kernel aplicados. Para obtener más información, consulte Desactivación de Kernel Live Patching con Run Command.

Conformidad del kernel

En algunos casos, la instalación de todas las correcciones de CVE desde actualizaciones en caliente para la versión actual del kernel puede llevar ese kernel al mismo estado de conformidad que tendría una versión más reciente del kernel. Cuando eso sucede, la versión más reciente se notifica como `Installed`, y el nodo administrado se informa como `Compliant`. Sin embargo, no se informa de tiempo de instalación para la versión más reciente del kernel.

Una actualización en caliente del kernel, varias CVE

Si una actualización en caliente del kernel se dirige a varias CVE, y esas CVE tienen varios valores de clasificación y gravedad, solo se informará de la clasificación y gravedad más altas de entre las CVE para la revisión.

En el resto de esta sección, se describe cómo utilizar Patch Manager para aplicar revisiones en caliente del kernel a los nodos administrados que cumplan estos requisitos.

Cómo funciona Kernel Live Patching mediante Patch Manager

AWS lanza dos tipos de revisiones en caliente del kernel para Amazon Linux 2: actualizaciones de seguridad y correcciones de errores. Para aplicar estos tipos de revisiones, se utiliza un documento de línea de base de revisiones que se centra únicamente en las clasificaciones y gravedades enumeradas en la siguiente tabla.

Clasificación	Gravedad
`Security`	`Critical`, `Important`
`Bugfix`	`All`

Puede crear una línea de base de revisiones personalizada que se orienta únicamente a estos revisiones, o utilizar la línea de base de revisiones predefinida de `AWS-`

`AmazonLinux2DefaultPatchBaseline`. En otras palabras, puede utilizar `AWS-AmazonLinux2DefaultPatchBaseline` con nodos administrados de Amazon Linux 2 en los que Kernel Live Patching está habilitado, y las revisiones en caliente del kernel se aplicarán.

> **Note**
>
> La configuración de `AWS-AmazonLinux2DefaultPatchBaseline` especifica un periodo de espera de siete días después del lanzamiento o última actualización de una revisión antes de que se instale automáticamente. Si no desea esperar ese plazo para que se aprueben automáticamente las revisiones en caliente del kernel, puede crear y utilizar una línea de base de revisiones personalizada. En la línea de base de revisiones, puede especificar que no haya ningún periodo de espera para la aprobación automática, o bien puede especificar uno más corto o más largo. Para obtener más información, consulte Uso de bases de referencia de parches personalizadas.

Recomendamos la siguiente estrategia para aplicar revisiones a los nodos administrados con actualizaciones en vivo del kernel:

1. Active Kernel Live Patching en los nodos administrados de Amazon Linux 2.
2. Utilice Run Command, una herramienta de AWS Systems Manager, para ejecutar una operación `Scan` en los nodos administrados mediante la `AWS-AmazonLinux2DefaultPatchBaseline` predefinida o una línea de base de revisiones personalizada que también tiene como objetivo solo las actualizaciones de `Security` con gravedad clasificada como `Critical` o `Important` y la gravedad `Bugfix` de `All`.
3. Utilice Cumplimiento, una herramienta de AWS Systems Manager, con el fin de revisar si se notifica la falta de conformidad para la aplicación de revisiones en cualquiera de los nodos administrados que se analizaron. Si es así, consulte los detalles de conformidad del nodo para determinar si faltan actualizaciones en caliente del kernel en el nodo administrado.
4. Para instalar las actualizaciones en caliente del kernel que faltan, use Run Command con la misma línea de base de revisiones que especificó antes, pero esta vez ejecute una operación `Install` en lugar de una operación `Scan`.

   Debido a que las revisiones en vivo del kernel se instalan sin necesidad de reiniciar, puede elegir la opción de reinicio `NoReboot` para esta operación.

> **Note**
>
> Todavía puede reiniciar el nodo administrado si es necesario para otros tipos de revisiones instaladas en el nodo, o si desea actualizar a un kernel más reciente. En estos casos, elija la opción de reinicio `RebootIfNeeded` en su lugar.

5. Vuelva a Compliance para verificar que se instalaron las actualizaciones en caliente del kernel.

Activación de Kernel Live Patching con Run Command

Para activar Kernel Live Patching, puede ejecutar comandos `yum` en los nodos administrados o utilizar Run Command y un documento de Systems Manager personalizado (documento de SSM) que cree.

Para obtener información sobre cómo activar Kernel Live Patching mediante la ejecución de comandos `yum` directamente en el nodo administrado, consulte Habilitación de Kernel Live Patching en la Guía del usuario de Amazon EC2.

> **Note**
>
> Cuando se activa Kernel Live Patching, el proceso instala la versión más reciente del kernel disponible y reinicia el nodo administrado si el kernel que se esté ejecutando en el nodo administrado es una versión anterior a la versión `kernel-4.14.165-131.185.amzn2.x86_64` (la versión mínima admitida). Si el nodo ya está ejecutando la versión `kernel-4.14.165-131.185.amzn2.x86_64` o posterior, el proceso no instala una versión más reciente ni reinicia el nodo.

Para activar Kernel Live Patching con Run Command (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command.
3. Elija Run command (Ejecutar comando).
4. En la lista Command document (Documento de Command), elija el documento de SSM `AWS-ConfigureKernelLivePatching` personalizado.

5. En la sección Command parameters (Parámetros de comandos) especifique si desea que los nodos administrados se reinicien como parte de esta operación.
6. Para obtener información sobre cómo trabajar con los controles restantes de esta página, consulte Ejecución de comandos desde la consola.
7. Seleccione Ejecutar.

Para activar Kernel Live Patching (AWS CLI)

- Ejecute el siguiente comando en el equipo local.

  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"
  ```

  Sustituya *instance-id* por el ID del nodo administrado de Amazon Linux 2 en el que desea activar la característica, como i-02573cafcfEXAMPLE. Para activar la característica en varios nodos administrados, puede utilizar cualquiera de los siguientes formatos.

  - `--targets "Key=instanceids,Values=instance-id1,instance-id2"`
  - `--targets "Key=tag:tag-key,Values=tag-value"`

  Para obtener información acerca de otras opciones que puede utilizar con el comando, consulte send-command en la Referencia de comandos de la AWS CLI.

Aplicación de actualizaciones en caliente del kernel mediante Run Command

Para aplicar revisiones en caliente del kernel, puede ejecutar comandos `yum` en los nodos administrados o utilizar Run Command y el documento de SSM `AWS-RunPatchBaseline`.

Para obtener información sobre cómo aplicar revisiones activas del kernel mediante la ejecución de comandos `yum` directamente en el nodo administrado, consulte Aplicación de revisiones activas del kernel en la Guía del usuario de Amazon EC2.

Para aplicar actualizaciones en caliente del kernel mediante Run Command (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command.
3. Elija Run command (Ejecutar comando).
4. En la lista Command document (Documento de Command), elija el documento de SSM `AWS-RunPatchBaseline`.
5. En la sección Command parameters (Parámetros de comandos) realice una de las acciones siguientes:
   - Si está verificando si hay nuevos revisiones en caliente del kernel disponibles, en Operation (Operación), elija `Scan`. En Reboot Option (Opción de reinicio), elija `NoReboot` si no desea que los nodos administrados se reinicien después de esta operación. Una vez completada la operación, puede verificar si hay nuevos revisiones y el estado de conformidad en Compliance.
   - Si ya ha comprobado la conformidad de las revisiones y está listo para aplicar las actualizaciones en caliente del kernel disponibles, en Operation (Operación), elija `Install`. En Reboot Option (Opción de reinicio), elija `NoReboot` si no desea que los nodos administrados se reinicien después de esta operación.
6. Para obtener información sobre cómo trabajar con los controles restantes de esta página, consulte Ejecución de comandos desde la consola.
7. Seleccione Ejecutar.

Para aplicar actualizaciones en caliente del kernel mediante Run Command (AWS CLI)

1. Para realizar una operación `Scan` antes de verificar sus resultados en Compliance, ejecute el siguiente comando desde su equipo local.

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
\"]}
```

Para obtener información acerca de otras opciones que puede utilizar con el comando, consulte send-command en la Referencia de comandos de la AWS CLI.

2. Para realizar una operación `Install` después de verificar los resultados en Compliance, ejecute el siguiente comando desde el equipo local.

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\"]}
```

En los dos comandos anteriores, reemplace *instance-id* por el ID del nodo administrado de Amazon Linux 2 en el que desea aplicar revisiones en caliente del kernel, como

i-02573cafcfEXAMPLE. Para activar la característica en varios nodos administrados, puede utilizar cualquiera de los siguientes formatos.

- `--targets "Key=instanceids,Values=instance-id1,instance-id2"`
- `--targets "Key=tag:tag-key,Values=tag-value"`

Para obtener información sobre otras opciones que puede utilizar con estos comandos, consulte send-command en la Referencia de comandos de la AWS CLI.

Desactivación de Kernel Live Patching con Run Command

Para desactivar Kernel Live Patching, puede ejecutar comandos `yum` en los nodos administrados o utilizar Run Command y un documento de SSM `AWS-ConfigureKernelLivePatching` personalizado.

> **Note**
>
> Si ya no necesita utilizar Kernel Live Patching, puede desactivarlo en cualquier momento. En la mayoría de los casos, no es necesario desactivar la característica.

Para obtener información sobre cómo desactivar Kernel Live Patching mediante la ejecución de comandos `yum` directamente en el nodo administrado, consulte Habilitación de Kernel Live Patching en la Guía del usuario de Amazon EC2.

> **Note**
>
> Cuando desactiva Kernel Live Patching, el proceso desinstala el complemento de Kernel Live Patching y, a continuación, reinicia el nodo administrado.

Para desactivar Kernel Live Patching con Run Command (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command.
3. Elija Run command (Ejecutar comando).

4. En la lista Command document (Documento de Command), elija el documento de SSM `AWS-ConfigureKernelLivePatching`.
5. En la sección Command Parameters, especifique los valores de los parámetros obligatorios.
6. Para obtener información sobre cómo trabajar con los controles restantes de esta página, consulte Ejecución de comandos desde la consola.
7. Seleccione Ejecutar.

Para desactivar Kernel Live Patching(AWS CLI)

- Ejecute un comando similar al siguiente:

  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"
  ```

  Sustituya *instance-id* por el ID del nodo administrado de Amazon Linux 2 en el que desea desactivar la característica, como i-02573cafcfEXAMPLE. Para desactivar la característica en varios nodos administrados, puede utilizar cualquiera de los siguientes formatos.

  - `--targets "Key=instanceids,Values=instance-id1,instance-id2"`
  - `--targets "Key=tag:tag-key,Values=tag-value"`

  Para obtener información acerca de otras opciones que puede utilizar con el comando, consulte send-command en la Referencia de comandos de la AWS CLI.

Trabajo con los recursos Patch Manager y el cumplimiento mediante la consola

Para usar Patch Manager, una herramienta de AWS Systems Manager, complete las siguientes tareas. Estas tareas se describen con más detalle en esta sección.

1. Compruebe que la línea de base de revisiones predefinida de AWS para cada tipo de sistema operativo que utiliza seas adecuada para sus necesidades. Si no es así, cree una base de referencia de revisiones que defina un conjunto estándar de revisiones para dicho tipo de nodo administrado y establézcalo como la opción predeterminada.
2. Organice los nodos administrados en grupos de revisiones mediante etiquetas de Amazon Elastic Compute Cloud (Amazon EC2) (opcional, pero recomendado).
3. Realice una de las siguientes acciones:
   - (Recomendado) Configure una política de revisiones en Quick Setup, una herramienta de Systems Manager, que le permita instalar revisiones faltantes en una programación para una organización completa, un subconjunto de unidades organizacionales o una sola Cuenta de AWS. Para obtener más información, consulte Cómo configurar las revisiones para las instancias de una organización mediante una política de parches Quick Setup.
   - Cree un periodo de mantenimiento que utilice el documento de Systems Manager (documento de SSM) `AWS-RunPatchBaseline` en un tipo de tarea de Run Command. Para obtener más información, consulte Tutorial: creación de un periodo de mantenimiento para la aplicación de revisiones mediante la consola.
   - Ejecute manualmente `AWS-RunPatchBaseline` en una operación Run Command. Para obtener más información, consulte Ejecución de comandos desde la consola.
   - Aplique revisiones manualmente a los nodos bajo demanda con la función Patch now (Aplicar revisión ahora). Para obtener más información, consulte Aplicación de revisiones a nodos administrados bajo demanda.
4. Monitorice la aplicación de revisiones para verificar la conformidad e investigar los errores.

Temas

- Creación de una política de revisiones
- Visualización de resúmenes del panel de revisiones
- Trabajo con informes de conformidad de las revisiones
- Aplicación de revisiones a nodos administrados bajo demanda
- Trabajo con línea de base de revisiones

- Visualización de revisiones disponibles
- Creación y administración de grupos de revisiones
- Integración de Patch Manager con AWS Security Hub

Creación de una política de revisiones

Una política de revisiones es un ajuste que se configura mediante Quick Setup, una herramienta de AWS Systems Manager. Las políticas de revisiones brindan un control más amplio y centralizado sobre sus operaciones de aplicación de revisiones que el que estaba disponible con otros métodos de configuración de aplicación de revisiones. Una política de revisiones define la programación y la línea de base que se usarán cuando se apliquen revisiones automáticamente a sus nodos y aplicaciones.

Para obtener más información, consulte los temas siguientes:

- Configuraciones de políticas de revisiones en Quick Setup
- Cómo configurar las revisiones para las instancias de una organización mediante una política de parches Quick Setup

Visualización de resúmenes del panel de revisiones

La pestaña Panel en Patch Manager proporciona una vista de resumen en la consola que puede utilizar para supervisar las operaciones de aplicación de revisiones en una vista consolidada. Patch Manager es una herramienta de AWS Systems Manager. En la pestaña Dashboard (Panel), puede ver los siguientes gráficos:

- Una instantánea que muestra cuántos nodos administrados están en conformidad o no con las reglas de aplicación de revisiones.
- Una instantánea que muestra la antigüedad de los resultados de conformidad de las revisiones para los nodos administrados.
- Recuento vinculado de cuántos nodos administrados no conformes existen para cada uno de los motivos más comunes de no conformidad.
- Lista vinculada de las operaciones de revisión más recientes.
- Lista vinculada de las tareas de revisión periódicas que se han configurado.

Para ver los resúmenes del panel de revisiones

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Patch Manager.
3. Elija la pestaña Dashboard (Panel).
4. Desplácese hasta la sección que contiene los datos resumidos que desea ver:
   - Amazon EC2 instance management (Administración de instancias de Amazon EC2)
   - Compliance summary (Resumen de conformidad)
   - Noncompliance counts (Recuentos de no conformidad)
   - Compliance reports (Informes de conformidad)
   - Non-patch policy-based operations (Operaciones basadas en políticas no relacionadas con revisiones)
   - Non-patch policy-based recurring tasks (Tareas recurrentes basadas en políticas no relacionadas con revisiones)

Trabajo con informes de conformidad de las revisiones

Utilice la información en los siguientes temas como ayuda para generar y trabajar con informes de conformidad de las revisiones en Patch Manager, una herramienta de AWS Systems Manager.

La información de los siguientes temas se aplica independientemente del método o el tipo de configuración que utilice para las operaciones de aplicación de revisiones:

- Una política de revisiones configurada en Quick Setup
- Una opción de administración de host configurada en Quick Setup
- Una ventana de mantenimiento para ejecutar una revisión `Scan` o una tarea `Install`
- Una operación Patch now bajo demanda

> **Important**
>
> Si tiene varios tipos de operaciones implementadas para analizar las instancias en busca de conformidad de revisiones, recuerde que cada análisis sobreescribe los datos de conformidad de revisiones de los análisis anteriores. Como consecuencia, es posible que obtenga resultados inesperados en los datos de cumplimiento de sus revisiones.

Para obtener más información, consulte Evitar sobrescrituras involuntarias de datos de conformidad de revisiones.
Para comprobar qué línea de base de revisiones se utilizó para generar la información de cumplimiento más reciente, vaya a la pestaña Informes de cumplimiento en Patch Manager, busque la fila del nodo administrado del que desea obtener información y elija el ID de referencia en la columna ID de línea de base utilizado.

Temas

- Visualización de resultados de conformidad de revisiones
- Generación de informes de conformidad de parches en formato .csv
- Corrección de los nodos gestionados no conformes con Patch Manager
- Evitar sobrescrituras involuntarias de datos de conformidad de revisiones

Visualización de resultados de conformidad de revisiones

Utilice estos procedimientos para ver la información de conformidad de revisiones sobre los nodos administrados.

Este procedimiento se aplica a las operaciones de parches que utilizan el documento `AWS-RunPatchBaseline`. Para obtener información acerca de cómo ver la información de conformidad de parches para las operaciones de parches que utilizan el documento `AWS-RunPatchBaselineAssociation`, consulte Identificación de nodos administrados no conformes.

Note

Las operaciones de escaneo de revisiones para Quick Setup y Explorer utilizan el documento `AWS-RunPatchBaselineAssociation`. Tanto Quick Setup como Explorer son herramientas de AWS Systems Manager.

Identificar la solución de parches para un problema CVE específico (Linux)

Para muchos sistemas operativos basados en Linux, los resultados de conformidad de parches indican qué problemas del boletín de vulnerabilidades y exposiciones comunes (CVE) se solucionan con cada uno de los parches. Esta información puede ayudarlo a determinar con qué urgencia necesita instalar un parche faltante o fallido.

Se incluyen detalles de la CVE para las versiones compatibles de los siguientes tipos de sistemas operativos:

- 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**
>
> De forma predeterminada, CentOS y CentOS Stream no proporcionan información de CVE relativa a las actualizaciones. Sin embargo, puede permitir este soporte mediante el uso de repositorios de terceros como es el caso del repositorio Extra Packages for Enterprise Linux (EPEL) publicado por Fedora. Para obtener información, consulte EPEL en el wiki de Fedora. En la actualidad, los valores de ID de CVE solo se reportan para los parches con el estado `Missing` o `Failed`.

También puede agregar ID de CVE a sus listas de parches aprobados o rechazados en sus bases de referencia de parches, según lo justifique la situación y sus objetivos de aplicación de parches.

Para obtener información acerca de cómo trabajar con las listas de parches aprobados y rechazados, consulte los siguientes temas:

- Uso de bases de referencia de parches personalizadas
- Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas
- Funcionamiento de las reglas de bases de referencia de parches en los sistemas basados en Linux
- Cómo se instalan las revisiones

> **Note**
>
> En algunos casos, Microsoft lanza parches para las aplicaciones que no especifican una hora ni una fecha de actualización. En estos casos, se suministra una fecha y hora actualizadas de `01/01/1970` de forma predeterminada.

Visualización de los resultados de conformidad de revisiones

Utilice los siguientes procedimientos para ver los resultados de conformidad de parches en la consola de AWS Systems Manager.

> **Note**
>
> Para obtener información acerca de cómo generar informes de conformidad de parches que se descargan en un bucket de Amazon Simple Storage Service (Amazon S3), consulte Generación de informes de conformidad de parches en formato .csv.

Para ver los resultados de conformidad de parches

1. Aplique alguna de las siguientes acciones.

   Opción 1 (recomendada). Navegue desde Patch Manager, una herramienta de AWS Systems Manager:

   - En el panel de navegación, elija Patch Manager.
   - Elija la pestaña Compliance reporting (Informes de conformidad).
   - En el área de Detalles de revisión de nodos, seleccione el ID del nodo administrado para el cual quiera revisar los resultados de conformidad de los parches.
   - En el área Detalles, en la lista Propiedades, seleccione Parches.

   Opción 2. Navegue desde Cumplimiento, una herramienta de AWS Systems Manager:

   - En el panel de navegación, elija Compliance.
   - En la sección Compliance resources summary (Resumen de recursos de conformidad), elija un número en la columna correspondiente a los tipos de recursos de parches que desee revisar, como Non-Compliant resources (Recursos no conformes).

- Debajo, en la lista Recursos, seleccione el ID del nodo administrado para el cual quiera revisar los resultados de conformidad de los parches.
- En el área Detalles, en la lista Propiedades, seleccione Parches.

Opción 3. Navegue desde Fleet Manager, una herramienta de AWS Systems Manager.

- En el panel de navegación, elija Fleet Manager.
- En el área Instancias administradas, seleccione el ID del nodo administrado para el cual quiera revisar los resultados de conformidad de los parches.
- En el área Detalles, en la lista Propiedades, seleccione Parches.

2. (Opcional) En el cuadro de búsqueda (), elija entre los filtros disponibles.

   Por ejemplo, para Red Hat Enterprise Linux (RHEL), elija entre las siguientes opciones:

   - Nombre
   - Clasificación
   - Estado
   - Gravedad

   Para Windows Server, elija entre las siguientes opciones:

   - KB
   - Clasificación
   - Estado
   - Gravedad

3. Elija uno de los valores disponibles para el tipo de filtro que seleccionó. Por ejemplo, si eligió State (Estado), ahora elija un estado de conformidad como InstalledPendingReboot (Instalado pendiente de reinicio), Failed (Con error) o Missing (Ausente).

> **Note**
>
> En la actualidad, los valores de ID de CVE solo se reportan para los parches con el estado `Missing` o `Failed`.

4. En función del estado de conformidad del nodo administrado, puede elegir qué acción llevar a cabo para corregir los nodos no conformes.

   Por ejemplo, puede elegir aplicar una revisión a los nodos administrados no conformes de forma inmediata. Para obtener información acerca de la aplicación de revisiones a los nodos administrados bajo demanda, consulte Aplicación de revisiones a nodos administrados bajo demanda.

   Para obtener información acerca de los estados de conformidad de parches, consulte Valores de estado de conformidad de las revisiones.

Generación de informes de conformidad de parches en formato .csv

Puede utilizar la consola de AWS Systems Manager para generar informes de conformidad de parches que se guardan como un archivo .csv en un bucket de Amazon Simple Storage Service (Amazon S3) de su elección. Puede generar un informe único bajo demanda o especificar una programación para generar los informes de forma automática.

Los informes se pueden generar para un único nodo administrado o para todos los nodos administrados de la selección de Cuenta de AWS y Región de AWS. En el caso de un único nodo, un informe presenta detalles completos, incluidos los ID de las revisiones relacionadas con un nodo que no es conforme. En el caso de un informe sobre todos los nodos administrados, solo se proporciona información de resumen y recuentos de las revisiones de los nodos no conformes.

Una vez generado un informe, puede utilizar una herramienta como Amazon QuickSight para importar y analizar los datos. QuickSight es un servicio de inteligencia empresarial (BI) que se puede utilizar para explorar e interpretar la información en un entorno visual interactivo. Para obtener más información, consulte la Guía del usuario de Amazon QuickSight.

> **Note**
>
> Cuando crea una línea de base de revisiones personalizada, puede especificar un nivel de gravedad de conformidad para las revisiones aprobadas por esa línea de base de

revisiones, como `Critical` o `High`. Si se informa del estado de cualquier revisión aprobada como `Missing`, la gravedad de la conformidad general notificada por la línea de base de revisiones será el nivel de gravedad que haya especificado.

También puede especificar un tema de Amazon Simple Notification Service (Amazon SNS) que se utilizará para enviar notificaciones cuando se genera un informe.

Roles de servicio para la generación de informes de conformidad de parches

La primera vez que se genera un informe, Systems Manager crea un rol de asunción de Automation denominado `AWS-SystemsManager-PatchSummaryExportRole` para utilizarlo en el proceso de exportación a S3.

Note

Si va a exportar datos de conformidad a un bucket de S3 cifrado, debe actualizar su política de claves de AWS KMS asociada para proporcionar los permisos necesarios para `AWS-SystemsManager-PatchSummaryExportRole`. Por ejemplo, agregue un permiso similar a este a la política AWS KMS del bucket de S3:

```
{
    "Effect": "Allow",
    "Action": [
        "kms:GenerateDataKey"
    ],
    "Resource": "role-arn"
}
```

Reemplace *`role-arn`* por el nombre de recurso de Amazon (ARN) del creado en su cuenta, en el formato `arn:aws:iam::`*`111222333444`*`:role/service-role/AWS-SystemsManager-PatchSummaryExportRole`.
Para obtener más información, consulte Políticas de claves en AWS KMS en la Guía para desarrolladores de AWS Key Management Service.

La primera vez que se genera un informe en una programación, Systems Manager crea otro rol de servicio denominado `AWS-EventBridge-Start-SSMAutomationRole`, así como el rol de servicio `AWS-SystemsManager-PatchSummaryExportRole` (en caso de no haberse creado ya) para

utilizarlo en el proceso de exportación. `AWS-EventBridge-Start-SSMAutomationRole` permite que Amazon EventBridge inicie una automatización mediante el manual de procedimientos AWS-ExportPatchReportToS3.

Se recomienda no intentar modificar estas políticas y roles. En caso de hacerlo, podría producirse un error al generar el informe de conformidad de parches. Para obtener más información, consulte Solución de problemas relacionados con la generación de informes de conformidad de parches.

Temas

- ¿Qué incluye un informe de conformidad de parches generado?
- Generación de informes de conformidad de revisiones para un único nodo administrado
- Generación de informes de conformidad de revisiones para todos los nodos administrados
- Visualización del historial de informes de conformidad de parches
- Visualización de la programación de generación de informes de conformidad de parches
- Solución de problemas relacionados con la generación de informes de conformidad de parches

¿Qué incluye un informe de conformidad de parches generado?

Este tema proporciona información acerca de los tipos de contenido incluidos en los informes de conformidad de parches que se generan y descargan en un bucket de S3 especificado.

Formato de informe para un único nodo administrado

Un informe generado para un único nodo administrado proporciona información de resumen y detallada.

Descargar un ejemplo de informe (único nodo)

La información de resumen de un único nodo administrado incluye lo siguiente:

- Índice
- ID de instancia
- Nombre de instancia
- IP de la instancia
- nombre de la plataforma
- Versión de la plataforma

- Versión de SSM Agent
- Línea de base de revisiones
- grupo de parches
- Compliance status (Estado de conformidad)
- severidad de la conformidad
- recuento de parches de severidad crítica no conformes
- recuento de parches de severidad alta no conformes
- recuento de parches de severidad media no conformes
- recuento de parches de severidad baja no conformes
- recuento de parches de severidad informativa no conformes
- recuento de parches de severidad sin especificar no conformes

La información detallada de un único nodo administrado incluye lo siguiente:

- Índice
- ID de instancia
- Nombre de instancia
- nombre del parche
- ID de KB o ID de parche
- estado del parche
- hora del último informe
- nivel de conformidad
- gravedad del parche
- clasificación del parche
- ID de CVE
- Línea de base de revisiones
- URL de registros
- IP de la instancia
- nombre de la plataforma
- Versión de la plataforma
- Versión de SSM Agent

> **Note**
>
> Cuando crea una línea de base de revisiones personalizada, puede especificar un nivel de gravedad de conformidad para las revisiones aprobadas por esa línea de base de revisiones, como `Critical` o `High`. Si se informa del estado de cualquier revisión aprobada como `Missing`, la gravedad de la conformidad general notificada por la línea de base de revisiones será el nivel de gravedad que haya especificado.

Formato de informe para todos los nodos administrados

Un informe generado para todos los nodos administrados proporciona únicamente información de resumen.

Descargar un informe de ejemplo (todos los nodos administrados)

La información de resumen de todos los nodos administrados incluye lo siguiente:

- Índice
- ID de instancia
- Nombre de instancia
- IP de la instancia
- nombre de la plataforma
- Versión de la plataforma
- Versión de SSM Agent
- Línea de base de revisiones
- grupo de parches
- Compliance status (Estado de conformidad)
- severidad de la conformidad
- recuento de parches de severidad crítica no conformes
- recuento de parches de severidad alta no conformes
- recuento de parches de severidad media no conformes
- recuento de parches de severidad baja no conformes
- recuento de parches de severidad informativa no conformes
- recuento de parches de severidad sin especificar no conformes

Generación de informes de conformidad de revisiones para un único nodo administrado

Utilice el siguiente procedimiento para generar un informe de resumen de revisiones para un único nodo administrado en la Cuenta de AWS. El informe para un único nodo administrado proporciona detalles sobre cada revisión que no está en conformidad, incluidos los nombres e ID de las revisiones.

Para generar informes de conformidad de revisiones para un único nodo administrado

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Patch Manager.
3. Elija la pestaña Compliance reporting (Informes de conformidad).
4. Elija el botón de la fila del nodo administrado para el que desea generar un informe y, a continuación, elija View detail (Ver detalle).
5. En la sección Patch summary (Resumen de revisiones), elija Export to S3 (Exportar a S3).
6. En Report name (Nombre del informe), ingrese un nombre que le permita identificar el informe más adelante.
7. En Reporting frequency (Frecuencia de la generación de informes), elija una de las siguientes opciones:
   - On demand (Bajo demanda): cree un informe único Vaya al paso 9.
   - On a schedule (Programación): especifique una programación periódica para la generación automática de informes. Continúe con el paso 8.
8. En Schedule type (Tipo de programación), especifique una expresión rate, por ejemplo, cada 3 días, o proporcione una expresión cron que configure la frecuencia del informe.

   Para obtener más información acerca de las expresiones cron, consulte Referencia: expresiones cron y rate para Systems Manager.
9. En Bucket name (Nombre del bucket), seleccione el nombre de un bucket de S3 en el que desee almacenar los archivos de informe .csv.

   > **Important**
   >
   > Si trabaja en una Región de AWS que se lanzó después del 20 de marzo de 2019, deberá seleccionar un bucket de S3 en la misma región. Las regiones lanzadas después de esa fecha se desactivaron de forma predeterminada. Para obtener más información

sobre estas regiones y una lista de ellas, consulte Enabling a Region en la Referencia general de Amazon Web Services.

10. (Opcional) Para enviar notificaciones cuando se genere el informe, expanda la sección SNS topic (Tema de SNS) y, a continuación, elija un tema de Amazon SNS existente desde SNS topic Amazon Resource Name (ARN) (Nombre de recurso de Amazon (ARN) del tema de SNS).
11. Elija Enviar.

Para obtener información acerca de cómo ver un historial de informes generados, consulte Visualización del historial de informes de conformidad de parches.

Para obtener información acerca de cómo ver los detalles de las programaciones de generación de informes que ha creado, consulte Visualización de la programación de generación de informes de conformidad de parches.

Generación de informes de conformidad de revisiones para todos los nodos administrados

Utilice el siguiente procedimiento para generar un informe de resumen de revisiones para todos los nodos administrados en la Cuenta de AWS. El informe de todos los nodos administrados muestra cuáles son los nodos y los números de las revisiones que no están en conformidad. No proporciona los nombres ni otros identificadores de los parches. Para obtener estos detalles adicionales, puede generar un informe de conformidad de revisiones para un único nodo administrado. Para obtener información, consulte la sección Generación de informes de conformidad de revisiones para un único nodo administrado que se ha expuesto anteriormente en este tema.

Para generar informes de conformidad de revisiones para todos los nodos administrados

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Patch Manager.
3. Elija la pestaña Compliance reporting (Informes de conformidad).
4. Elija Export to S3 (Exportar a S3). (No seleccione primero un ID de nodo).
5. En Report name (Nombre del informe), ingrese un nombre que le permita identificar el informe más adelante.
6. En Reporting frequency (Frecuencia de la generación de informes), elija una de las siguientes opciones:

- On demand (Bajo demanda): cree un informe único Vaya al paso 8.
- On a schedule (Programación): especifique una programación periódica para la generación automática de informes. Continúe en el paso 7.

7. En Schedule type (Tipo de programación), especifique una expresión rate, por ejemplo, cada 3 días, o proporcione una expresión cron que configure la frecuencia del informe.

   Para obtener más información acerca de las expresiones cron, consulte Referencia: expresiones cron y rate para Systems Manager.

8. En Bucket name (Nombre del bucket), seleccione el nombre de un bucket de S3 en el que desee almacenar los archivos de informe .csv.

   > **Important**
   >
   > Si trabaja en una Región de AWS que se lanzó después del 20 de marzo de 2019, deberá seleccionar un bucket de S3 en la misma región. Las regiones lanzadas después de esa fecha se desactivaron de forma predeterminada. Para obtener más información sobre estas regiones y una lista de ellas, consulte Enabling a Region en la Referencia general de Amazon Web Services.

9. (Opcional) Para enviar notificaciones cuando se genere el informe, expanda la sección SNS topic (Tema de SNS) y, a continuación, elija un tema de Amazon SNS existente desde SNS topic Amazon Resource Name (ARN) (Nombre de recurso de Amazon (ARN) del tema de SNS).
10. Elija Enviar.

Para obtener información acerca de cómo ver un historial de informes generados, consulte Visualización del historial de informes de conformidad de parches.

Para obtener información acerca de cómo ver los detalles de las programaciones de generación de informes que ha creado, consulte Visualización de la programación de generación de informes de conformidad de parches.

Visualización del historial de informes de conformidad de parches

Utilice la información de este tema para que pueda ver los detalles acerca de los informes de conformidad de parches generados en su Cuenta de AWS.

Para ver el historial de informes de conformidad de parches

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Patch Manager.
3. Elija la pestaña Compliance reporting (Informes de conformidad).
4. Elija View all S3 exports (Ver todas las exportaciones de S3) y, a continuación, elija la pestaña Export history (Historial de exportación).

Visualización de la programación de generación de informes de conformidad de parches

Utilice la información de este tema para que pueda ver los detalles acerca de las programaciones de generación de informes de conformidad de parches que ha creado en su Cuenta de AWS.

Para ver el historial de informes de conformidad de parches

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Patch Manager.
3. Elija la pestaña Compliance reporting (Informes de conformidad).
4. Elija View all S3 exports (Ver todas las exportaciones de S3) y, a continuación, elija la pestaña Report scheduled rules (Informar reglas programadas).

Solución de problemas relacionados con la generación de informes de conformidad de parches

Utilice la siguiente información que lo ayudará a solucionar problemas con la generación de informes de conformidad de parches en Patch Manager, una herramienta de AWS Systems Manager.

Temas

- Un mensaje notifica que la política AWS-SystemsManager-PatchManagerExportRolePolicy está dañada.
- Después de eliminar roles o políticas de conformidad de parches, los informes programados no se generan correctamente

Un mensaje notifica que la política **AWS-SystemsManager-PatchManagerExportRolePolicy** está dañada.

Problema: recibe un mensaje de error similar al siguiente, en el que se indica que AWS-SystemsManager-PatchManagerExportRolePolicy está dañado:

```
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.
```

- Solución: utilice la consola Patch Manager o la AWS CLI para eliminar los roles y las políticas afectadas antes de generar un nuevo informe de cumplimiento de las revisiones.

  Para eliminar la política dañada con la consola

  1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
  2. Realice una de las siguientes acciones:

     Informes bajo demanda: si el problema se produjo cuando se generaba un informe bajo demanda único, en el panel de navegación izquierdo, elija Policies (Políticas), busque AWS-SystemsManager-PatchManagerExportRolePolicy, y, a continuación, elimine la política. Luego, elija Roles (Roles), busque AWS-SystemsManager-PatchSummaryExportRole y, a continuación, elimine el rol.

     Informes programados: si el problema se produjo mientras generaba un informe en una programación, en el panel de navegación izquierdo, elija Políticas, busque una a la vez en AWS-EventBridge-Start-SSMAutomationRolePolicy y AWS-SystemsManager-PatchManagerExportRolePolicy y elimine cada política. Luego, elija Roles (Roles), busque uno a la vez en AWS-EventBridge-Start-SSMAutomationRole y AWS-SystemsManager-PatchSummaryExportRole y, a continuación, elimine cada rol.

  Para eliminar una política dañada con la AWS CLI

  Sustituya los *valores de marcador* por su ID de la cuenta.

  - Si el problema se produjo al generar un informe único bajo demanda, ejecute los siguientes comandos:

```
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 el problema se produjo al generar un informe programado, ejecute los siguientes comandos:

```
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
```

Después de completar cualquiera de los procedimientos, siga los pasos para generar o programar un nuevo informe de conformidad de revisiones.

Después de eliminar roles o políticas de conformidad de parches, los informes programados no se generan correctamente

Problema: la primera vez que se genera un informe, Systems Manager crea un rol de servicio y una política para utilizarlos en el proceso de exportación (`AWS-SystemsManager-PatchSummaryExportRole` y `AWS-SystemsManager-PatchManagerExportRolePolicy`). La primera vez que se genera un informe en una programación, Systems Manager crea otro rol de servicio y una política (`AWS-EventBridge-Start-SSMAutomationRole` y `AWS-EventBridge-Start-SSMAutomationRolePolicy`). Estas permiten que Amazon EventBridge inicie una automatización mediante el manual de procedimientos AWS-ExportPatchReportToS3 .

Si elimina alguna de estas políticas o roles, es posible que se pierdan las conexiones entre su programación y el bucket de S3 y el tema de Amazon SNS especificados.

- Solución: para resolver este problema, se recomienda eliminar la programación anterior y crear una nueva que reemplace a la que presentaba problemas.

Corrección de los nodos gestionados no conformes con Patch Manager

Los temas incluidos en esta sección ofrecen información general sobre cómo identificar los nodos administrados que no están conformes con la aplicación de revisiones y cómo lograr esa conformidad.

Temas

- Identificación de nodos administrados no conformes
- Valores de estado de conformidad de las revisiones
- Revisiones en nodos administrados que no están en conformidad

Identificación de nodos administrados no conformes

Los nodos administrados que no están en conformidad se identifican en el momento en que se ejecuta cualquiera de los dos documentos de AWS Systems Manager (documentos de SSM). Estos documentos de SSM hacen referencia a la línea de base de revisiones adecuada para cada nodo administrado en Patch Manager, una herramienta de AWS Systems Manager. A continuación, se evalúa el estado de la revisión del nodo administrado y se ponen a disposición los resultados de conformidad.

Hay dos documentos de SSM que se utilizan para identificar o actualizar los nodos administrados no conformes: `AWS-RunPatchBaseline` y `AWS-RunPatchBaselineAssociation`. Cada una de ellas se utiliza en diferentes procesos, y sus resultados de conformidad se encuentran disponibles en distintos canales. En la siguiente tabla se exponen las diferencias entre estos documentos.

> **Note**
>
> Se pueden enviar los datos de conformidad de revisiones de Patch Manager a AWS Security Hub. Security Hub ofrece una visión completa de las alertas de seguridad de alta prioridad y el estado de conformidad. También monitorea el estado de aplicación de revisiones de la flota. Para obtener más información, consulte Integración de Patch Manager con AWS Security Hub.

	AWS-RunPatchBaseline	AWS-RunPatchBaselineAssociation
Procesos que utilizan el documento	Revisión bajo demanda: puede analizar o aplicar revisiones a nodos administrados bajo demanda mediante la opción Patch now (Aplicar revisión ahora). Para obtener más información, consulta Aplicación de revisiones a nodos administrados bajo demanda. Políticas de revisiones de Quick Setup de Systems Manager: puede crear una configuración de revisiones en Quick Setup, una herramienta de AWS Systems Manager, que pueda buscar o instalar las revisiones que faltan según programaciones independientes para toda una organización, un subconjunto de unidades organizativas o una sola Cuenta de AWS. Para obtener más información, consulta Cómo configurar las revisiones para las instancias de una organización mediante una política de parches Quick Setup. Ejecución de un comando: puede ejecutar `AWS-RunPatchBaseline` manualmen	Administración de host de Quick Setup de Systems Manager: puede habilitar una opción de configuración de administración de host en Quick Setup para analizar diariamente sus instancias administradas con el fin de comprobar la conformidad de las revisiones. Para obtener más información, consulta Instalar la administración de host de Amazon EC2 mediante Quick Setup. Explorer de Systems Manager: cuando permite Explorer, una herramienta de AWS Systems Manager, esta analiza periódicamente sus instancias administradas con el fin de comprobar la conformidad con las revisiones e informa los resultados en el panel de Explorer.

	AWS-RunPatchBaseline	AWS-RunPatchBaselineAssociation
	te en una operación en Run Command, una herramienta de AWS Systems Manager. Para obtener más información, consulta Ejecución de comandos desde la consola. Periodo de mantenimiento: puede crear un periodo de mantenimiento que utilice el documento de SSM `AWS-RunPatchBaseline` en un tipo de tarea de Run Command. Para obtener más información, consulta Tutorial: creación de un periodo de mantenimiento para la aplicación de revisiones mediante la consola.	
Formato de los datos de los resultados obtenidos en el análisis de revisiones	Una vez que se ejecuta `AWS-RunPatchBaseline`, Patch Manager envía un objeto `AWS:PatchSummary` a Inventario, una herramienta de AWS Systems Manager.	Una vez que se ejecuta `AWS-RunPatchBaselineAssociation`, Patch Manager envía un objeto `AWS:ComplianceItem` a Systems Manager Inventory.

	AWS-RunPatchBaseline	**AWS-RunPatchBaselineAssociation**
Visualización de informes de conformidad de revisiones en la consola	Puede visualizar la información de conformidad de las revisiones para los procesos que utilizan `AWS-RunPatchBaseline` en Conformidad de configuración de Systems Manager y Trabajo con nodos administrados. Para obtener más información, consulte Visualización de resultados de conformidad de revisiones.	Si utiliza Quick Setup para analizar sus instancias administradas para comprobar la conformidad con las revisiones, podrá consultar el informe de conformidad en Systems Manager Fleet Manager. En la consola Fleet Manager, elija el ID de nodo del nodo administrado. En el menú General, seleccione Cumplimiento de la configuración. Si utiliza Explorer para analizar sus instancias administradas para comprobar la conformidad con las revisiones, podrá consultar el informe de conformidad tanto en Explorer como en Systems Manager OpsCenter.

	AWS-RunPatchBaseline	AWS-RunPatchBaselineAssociation
Comandos de la AWS CLI para visualizar los resultados de conformidad de revisiones	Para los procesos que utilizan AWS-RunPatchBaseline, puede usar los siguientes comandos de la AWS CLI para obtener información de resumen acerca de las revisiones en un nodo administrado. • describe-instance-patch-states • describe-instance-patch-states-for-patch-group • describe-patch-group-state	Para los procesos que utilizan AWS-RunPatchBaselineAssociation, puede usar el siguiente comando de la AWS CLI para obtener información de resumen acerca de las revisiones en una instancia. • list-compliance-items
Operaciones de aplicación de revisiones	Para los procesos que utilizan AWS-RunPatchBaseline, se especifica si se desea que la operación ejecute únicamente una operación Scan o una operación Scan and install. Si el objetivo es identificar los nodos administrados no conformes en lugar de corregirlos, ejecute únicamente una operación Scan.	Los procesos Quick Setup y Explorer, que utilizan AWS-RunPatchBaselineAssociation, ejecutan únicamente una operación Scan.
Más información	Documento de comandos SSM para aplicar revisiones: AWS-RunPatchBaseline	Documento de comandos SSM para aplicar revisiones: AWS-RunPatchBaselineAssociation

Para obtener información acerca de los distintos estados de conformidad de las revisiones que se podrían notificar, consulte Valores de estado de conformidad de las revisiones

Para obtener información acerca de cómo corregir los nodos administrados que no están en conformidad con las revisiones, consulte Revisiones en nodos administrados que no están en conformidad.

Valores de estado de conformidad de las revisiones

La información sobre las revisiones de un nodo administrado incluye un informe sobre el estado de cada revisión individual.

> **Tip**
>
> Si desea asignar un estado de conformidad de revisiones específico a un nodo administrado, puede utilizar el comando de la AWS Command Line Interface (AWS CLI) put-compliance-items o la operación de la API PutComplianceItems. La asignación del estado de conformidad no se admite en la consola.

Utilice la información que aparece en las siguientes tablas para que pueda identificar el motivo por el cual un nodo administrado podría no estar en conformidad con las revisiones.

Valores de conformidad de revisiones para Debian Server, Raspberry Pi OS y Ubuntu Server

En el caso de Debian Server, Raspberry Pi OS y Ubuntu Server, las reglas para la clasificación de los paquetes en los diferentes estados de conformidad se describen en la siguiente tabla.

> **Note**
>
> Tenga en cuenta lo siguiente al evaluar los valores de estado `INSTALLED`, `INSTALLED_OTHER`, y `MISSING`: Si no selecciona la casilla Incluir actualizaciones no relacionadas con la seguridad al crear o actualizar una línea de base de revisiones, las versiones candidatas a parches se limitan a los parches en los siguientes repositorios:
>
> - 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 Server 23.04 (`lunar-security`)
- Ubuntu Server 23.10 (`mantic-security`)
- Ubuntu Server 22.04 LTS (`noble-security`)
- Ubuntu Server 24.10 (`oracular-security`)
- Ubuntu Server 25.04 (`plucky-security`)
- `debian-security` (Debian Server y Raspberry Pi OS)

Si selecciona la casilla Include nonsecurity updates (Incluir actualizaciones no relacionadas con la seguridad), también se tendrán en cuenta los parches de otros repositorios.

Estado del parche	Descripción	Compliance status (Estado de conformidad)
`INSTALLED`	La revisión aparece en la base de referencia de revisiones y se instala en el nodo administrado. Podría haberse instalado de forma manual por un usuario o de forma automática por Patch Manager cuando se ejecutó el documento `AWS-RunPatchBaseline` en el nodo administrado.	Conforme
`INSTALLED_OTHER`	La revisión no se incluye en la base de referencia ni se encuentra aprobada por ella, pero se instala en el nodo administrado. Es posible que el parche se haya instalado manualmente, que el paquete sea una dependenc	Conforme

Estado del parche	Descripción	Compliance status (Estado de conformidad)
	ia necesaria de otro parche aprobado o que se haya incluido en una operación InstallOverrideList. Si no especifica `Block` como acción de parches rechazados, los parches `INSTALLED_OTHER` también incluyen los parches instalados pero que han sido rechazados.	

Estado del parche	Descripción	Compliance status (Estado de conformidad)
INSTALLED_PENDING_REBOOT	INSTALLED_PENDING_REBOOT puede significar cualquiera de las siguientes posibilidades: • La operación Install de Patch Manager aplicó la revisión al nodo administrado, pero el nodo no se ha reiniciado desde que se aplicó la revisión. Normalmente esto significa que se seleccionó la opción NoReboot para el parámetro RebootOption cuando el documento AWS-RunPatchBaseline se ejecutó por última vez en el nodo administrado. Para obtener más información, consulte Nombre del parámetro: RebootOption. • Se instaló un parche por fuera de Patch Manager desde última vez que se reinició el nodo administrado. En ninguno de los dos casos significa que una revisión con este estado requiera un	No conforme

Estado del parche	Descripción	Compliance status (Estado de conformidad)
	reinicio, solo que el nodo no se ha reiniciado desde que se instaló el parche.	
INSTALLED_REJECTED	La revisión está instalada en el nodo administrado, pero se especifica en una lista de revisiones rechazadas. Normalmente esto significa que el parche se instaló antes de que se añadiera a una lista de parches rechazados.	No conforme
MISSING	Paquetes que se filtran a través de la base de referenci a y aún no están instalados.	No conforme
FAILED	Los paquetes que no se pudieron instalar durante la operación de aplicación de parches.	No conforme

Valores de conformidad de parches para otros sistemas operativos

Para todos los sistemas operativos además de Debian Server, Raspberry Pi OS y Ubuntu Server, las reglas para la clasificación de los paquetes en los diferentes estados de conformidad se describen en la siguiente tabla.

Estado del parche	Descripción	Valor de conformidad
INSTALLED	La revisión aparece en la base de referencia de revisiones y se instala en el nodo administr ado. Podría haberse instalado de forma manual por un	Conforme

Estado del parche	Descripción	Valor de conformidad
	usuario o de forma automática por Patch Manager cuando se ejecutó el documento `AWS-RunPatchBaseline` en el nodo.	

Estado del parche	Descripción	Valor de conformidad
`INSTALLED_OTHER` [1]	La revisión no está en la base de referencia, pero se ha instalado en el nodo administrado. Existen dos razones posibles para ello: 1. Es posible que la revisión se haya instalado manualmente. 2. Solo para Linux: es posible que el paquete se haya instalado como una dependencia obligatoria de una revisión diferente aprobada. Si especifica `Allow as dependency` como la acción Revisiones rechazadas, las revisiones que se instalan como dependencias reciben el estado de notificación `INSTALLED_OTHER` . **Note** Windows Server no admite el concepto de dependencias de revisiones. Para obtener información sobre cómo Patch Manager gestiona las revisiones en la lista de revisiones rechazada	Conforme

Estado del parche	Descripción	Valor de conformidad
	sWindows Server, consulte las opciones de la lista de revisiones rechazadas en las líneas de base de revisiones personalizadas.	
INSTALLED_REJECTED	La revisión está instalada en el nodo administrado, pero se especifica en una lista de revisiones rechazadas. Normalmente esto significa que el parche se instaló antes de que se añadiera a una lista de parches rechazados.	No conforme

Estado del parche	Descripción	Valor de conformidad
INSTALLED_PENDING_REBOOT	INSTALLED_PENDING_REBOOT puede significar cualquiera de las siguientes posibilidades: • La operación Install de Patch Manager aplicó la revisión al nodo administrado, pero el nodo no se ha reiniciado desde que se aplicó la revisión. Normalmente esto significa que se seleccionó la opción NoReboot para el parámetro RebootOption cuando el documento AWS-RunPatchBaseline se ejecutó por última vez en el nodo administrado. Para obtener más información, consulte Nombre del parámetro: RebootOption. • Se instaló un parche por fuera de Patch Manager desde última vez que se reinició el nodo administrado. En ninguno de los dos casos significa que una revisión con este estado requiera un reinicio, solo que el nodo no	No conforme

Estado del parche	Descripción	Valor de conformidad
	se ha reiniciado desde que se instaló el parche.	
MISSING	La revisión está aprobada en la base de referencia, pero no se ha instalado en el nodo administrado. Si configura la tarea de documento `AWS-RunPatchBaseline` para analizar (en vez de instalar), el sistema informa este estado para los parches que se encontraron durante el análisis, pero que no se han instalado.	No conforme
FAILED	El parche está aprobado en la base de referencia, pero no se ha podido instalar. Para resolver esta situación, revise la salida del comando para buscar información que pueda ayudarle a comprender el problema.	No conforme

Estado del parche	Descripción	Valor de conformidad
NOT_APPLICABLE [1]	Este estado de cumplimiento solo se notifica en los sistemas operativos de Windows Server. La revisión está aprobada en la base de referencia, pero el servicio o la característica que usa la revisión no se ha instalado en el nodo administrado. Por ejemplo, una revisión de un servicio de servidor web como Internet Information Services (IIS) mostrará `NOT_APPLICABLE` si se ha aprobado en la base de referencia, pero el servicio web no se ha instalado en el nodo administrado. También se puede marcar un parche `NOT_APPLICABLE` si se ha reemplazado por una actualización posterior. Esto significa que la actualización posterior está instalada y la actualización `NOT_APPLICABLE` ya no es necesaria.	No aplicable

Estado del parche	Descripción	Valor de conformidad
AVAILABLE_SECURITY_UPDATES	Este estado de cumplimiento solo se notifica en los sistemas operativos de Windows Server. La revisión de actualización de seguridad disponible que no esté aprobada por la línea de base de revisiones puede tener un valor de cumplimiento `Compliant` o `Non-Compliant`, tal como se define en una línea de base de revisiones personalizada. Al crear o actualizar una línea de base de revisiones, usted elige el estado que desee asignar a las revisiones de seguridad que están disponibles pero no aprobadas porque no cumplen con los criterios de instalación especificados en la línea de base de revisiones. Por ejemplo, las revisiones de seguridad que desee instalar se pueden omitir si ha especificado un periodo prolongado de espera después del lanzamiento de la revisión antes de la instalación. Si se publica una actualización de la revisión durante el periodo de espera especificado, el periodo de espera para	Conforme o No conforme, según la opción seleccionada para las actualizaciones de seguridad disponibles. **Note** Si utiliza la consola para crear o actualizar una línea de base de revisiones, especifique esta opción en el campo Estado de cumplimiento de las actualizaciones de seguridad disponibles. Si utiliza la AWS CLI para ejecutar el comando create-patch-baseline o update-patch-baseline, especifique esta opción en el parámetro `available-security-updates-compliance-status`.

Estado del parche	Descripción	Valor de conformidad
	instalar la revisión vuelve a comenzar. Si el periodo de espera es demasiado extenso, es posible que se lancen varias versiones de la revisión, pero nunca se instalen. En cuanto a los recuentos resumidos de revisione s, cuando una revisión se notifica como `Available SecurityUpdate` , siempre se incluirá en `AvailableSecurityU pdateCount` . Si la línea de base está configurada para indicar estas revisiones como `NonCompliant` , también se incluye en `SecurityN onCompliantCount` . Si la línea de base está configura da para indicar estas revisione s como `Compliant` , no se incluyen en `SecurityN onCompliantCount` . Estas revisiones siempre se notifican con una gravedad no especificada y nunca se incluyen en el `CriticalN onCompliantCount` .	

[1] Para revisiones con el estado `INSTALLED_OTHER` y `NOT_APPLICABLE`, Patch Manager omite algunos datos de los resultados de la consulta en función del comando describe-instance-patches, como los valores de `Classification` y `Severity`. Esto se hace para ayudar a evitar que se

supere el límite de datos en los nodos individuales de Inventario, una herramienta de AWS Systems Manager. Para ver todos los detalles de la revisión, puede utilizar el comando describe-available-patches.

Revisiones en nodos administrados que no están en conformidad

Muchas de las mismas herramientas y procesos de AWS Systems Manager que puede utilizar para verificar la conformidad de los nodos administrados con las revisiones sirven para lograr que los nodos cumplan las reglas de las revisiones que se les aplican actualmente. Para que los nodos administrados estén en conformidad con la revisión, Patch Manager, una herramienta de AWS Systems Manager, debe ejecutar una operación `Scan and install`. (Si el objetivo es solo identificar los nodos administrados que no están en conformidad en vez de corregirlos, ejecute una operación `Scan` en su lugar. Para obtener más información, consulte Identificación de nodos administrados no conformes).

Instalar revisiones con Systems Manager

Puede elegir entre varias herramientas para realizar una operación `Scan and install`:

- (Recomendado) Configure una política de revisiones en Quick Setup, una herramienta de Systems Manager, que le permita instalar revisiones faltantes en una programación para una organización completa, un subconjunto de unidades organizacionales o una sola Cuenta de AWS. Para obtener más información, consulte Cómo configurar las revisiones para las instancias de una organización mediante una política de parches Quick Setup.
- Cree un periodo de mantenimiento que utilice el documento de Systems Manager (documento de SSM) `AWS-RunPatchBaseline` en un tipo de tarea de Run Command. Para obtener más información, consulta Tutorial: creación de un periodo de mantenimiento para la aplicación de revisiones mediante la consola.
- Ejecute manualmente `AWS-RunPatchBaseline` en una operación Run Command. Para obtener más información, consulta Ejecución de comandos desde la consola.

- Instale las revisiones bajo demanda con la opción Patch now (Aplicar revisión ahora). Para obtener más información, consulta Aplicación de revisiones a nodos administrados bajo demanda.

Evitar sobrescrituras involuntarias de datos de conformidad de revisiones

Si tiene varios tipos de operaciones implementadas para analizar las instancias para comprobar la conformidad de revisiones, cada análisis sobrescribe los datos de conformidad de revisiones de los

análisis anteriores. Como consecuencia, es posible que obtenga resultados inesperados en los datos de cumplimiento de sus revisiones.

Por ejemplo, supongamos que crea una política de revisiones que analiza la conformidad de las revisiones todos los días a las 2:00 h, hora local. Esa política utiliza una línea de base de revisiones que se centra en las revisiones con una gravedad marcada como `Critical`, `Important`, y `Moderate`. Esta línea de base de revisiones también especifica algunas revisiones que se rechazaron de forma específica.

Supongamos también que ya configuró un periodo de mantenimiento para analizar el mismo conjunto de nodos administrados todos los días a las 4 h, hora local, que no elimina ni desactiva. La tarea de ese periodo de mantenimiento utiliza una línea de base de revisiones diferente, una que se enfoca solo en las revisiones con una gravedad `Critical` y no excluye ninguna revisión específica.

Cuando el periodo de mantenimiento realiza este segundo análisis, los datos de conformidad de las revisiones del primer análisis se eliminan y se reemplazan por los del segundo análisis.

Por lo tanto, recomendamos encarecidamente usar solo un método automatizado para analizar e instalar en sus operaciones de aplicación de revisiones. Si está configurando políticas de revisiones, debe eliminar o desactivar otros métodos de análisis para comprobar el cumplimiento de revisiones. Para obtener más información, consulte los temas siguientes:

- Para eliminar una tarea de operación de aplicación de revisiones de un periodo de mantenimiento: Actualizar o eliminar tareas de un periodo de mantenimiento mediante la consola
- Para eliminar una asociación de State Manager: Eliminación de una asociación.

Para desactivar los análisis diarios de cumplimiento de revisiones en una configuración de administración de host, haga lo siguiente en Quick Setup:

1. En el panel de navegación, elija Quick Setup.
2. Seleccione la configuración de administración de host para actualizar.
3. Elija Actions (Acciones) y Edit configuration (Editar la configuración).
4. Desactive la casilla Scan instances for missing patches daily (Analizar diariamente las instancias en busca de revisiones que falten).
5. Elija Actualizar.

> **Note**
>
> El uso de la opción Patch now (Aplicar revisión ahora) para analizar un nodo administrado para comprobar la conformidad también da como resultado una sobrescritura de los datos de conformidad de las revisiones.

Aplicación de revisiones a nodos administrados bajo demanda

Puede ejecutar las operaciones de aplicación de revisiones bajo demanda desde la consola de Systems Manager mediante la opción Aplicar el parche ahora en Patch Manager, una herramienta de AWS Systems Manager. De este modo, no tendrá que crear una programación para actualizar el estado de conformidad de los nodos administrados o para instalar revisiones en los nodos no conformes. Tampoco es necesario cambiar la consola de Systems Manager entre Patch Manager y Maintenance Windows, una herramienta de AWS Systems Manager, para configurar o modificar un periodo de aplicación de revisiones programado.

Patch now (Aplicar revisión ahora) resulta de gran utilidad cuando se deben aplicar actualizaciones de día cero o instalar otras revisiones críticas en los nodos administrados lo antes posible.

> **Note**
>
> Se admite la aplicación de revisiones bajo demanda para un solo par de Cuenta de AWS-Región de AWS a la vez. No se puede usar con operaciones de revisiones basadas en políticas de revisiones. Recomendamos utilizar políticas de revisiones para mantener todos los nodos administrados en conformidad. Para obtener más información sobre el uso de las políticas de revisiones, consulte Configuraciones de políticas de revisiones en Quick Setup.

Temas

- Funcionamiento de “Patch now” (Aplicar revisión ahora)
- Ejecución de “Patch now” (Aplicar revisión ahora)

Funcionamiento de “Patch now” (Aplicar revisión ahora)

Para ejecutar Patch now (Aplicar revisión ahora), solo tiene que especificar dos ajustes necesarios:

- Si se analizan solo las revisiones faltantes o si se analizan e instalan las revisiones en los nodos administrados
- En qué nodos administrados se ejecuta la operación

Cuando la operación Patch now (Aplicar revisión ahora) se ejecuta, determina qué línea de base de revisiones se va a utilizar de la misma manera que se selecciona una para otras operaciones de revisiones. Si un nodo administrado está asociado a un grupo de revisiones, se utiliza la base de referencia de revisiones especificada para ese grupo. Si el nodo administrado no está asociado a un grupo de revisiones, la operación utiliza la base de referencia de revisiones que está configurada actualmente como predeterminada para el tipo de sistema operativo del nodo administrado. Puede tratarse de una base de referencia predefinida o de la base de referencia personalizada que haya definido como predeterminada. Para obtener más información acerca de la selección de línea de base de revisiones, consulte Grupos de revisiones.

Las opciones que se pueden especificar para Patch now (Aplicar revisión ahora) incluyen elegir cuándo reiniciar los nodos administrados después de la aplicación de revisiones, o si corresponde hacerlo, especificar un bucket de Amazon Simple Storage Service (Amazon S3) para almacenar los datos de registro de esa operación de aplicación de revisiones y ejecutar documentos de Systems Manager (documentos de SSM) como enlaces de ciclo de vida durante la aplicación de revisiones.

Umbrales de concurrencia y error para ‘Patch now’

Para las operaciones de Patch now (Aplicar revisión ahora), las opciones de umbral de concurrencia y de error se gestionan mediante Patch Manager. No es necesario especificar en cuántos nodos administrados se aplicará la revisión a la vez ni cuántos errores se permiten antes de que la operación no funcione. Patch Manager aplica las configuraciones de simultaneidad y límites de errores descritas en las siguientes tablas cuando se aplica una revisión bajo demanda.

> **Important**
>
> Los siguientes umbrales se aplican únicamente a las operaciones `Scan and install`. En el caso de las operaciones `Scan`, Patch Manager intenta analizar hasta 1000 nodos simultáneamente y continúa el análisis hasta que haya detectado un máximo de 1000 errores.

Simultaneidad: operaciones de instalación

Número total de nodos administrados en la operación Patch now (Aplicar revisión ahora)	Número de nodos administrados analizados o con revisiones a la vez
Menos de 25	1
25-100	5%
101 a 1000	8 %
Más de 1000	10%

Umbral de error: operaciones de instalación

Número total de nodos administrados en la operación Patch now (Aplicar revisión ahora)	Número de errores permitidos antes de que la operación no funcione
Menos de 25	1
25-100	5
101 a 1000	10
Más de 1000	10

Uso de los enlaces de ciclo de vida ‘Patch now’

Patch now (Aplicar revisión ahora) le proporciona la capacidad de ejecutar documentos de comando de SSM como enlaces de ciclo de vida durante una operación de revisióno de `Install`. Puede utilizar estos enlaces para tareas tales como apagar aplicaciones antes de aplicar revisiones o ejecutar comprobaciones de estado en las aplicaciones después de aplicar revisiones o después de un reinicio.

Para obtener más información acerca del uso de enlaces de ciclo de vida, consulte Documento de comandos SSM para aplicar revisiones: `AWS-RunPatchBaselineWithHooks`.

En la siguiente tabla se indican los enlaces de ciclo de vida disponibles para cada uno de las tres opciones de reinicio Patch now (Aplicar revisión ahora), además de los usos de muestra para cada enlace.

Enlaces de ciclo de vida y usos de muestra

Opción de reinicio	Enlace: antes de la instalación	Enlace: después de la instalación	Enlace: en la salida	Enlace: después del reinicio programado
Reboot if needed (Reiniciar si es necesario)	Ejecute un documento SSM antes de que comience la revisióno. Ejemplo de uso: Cierre las aplicaciones de forma segura antes de que comience el proceso de revisiones.	Ejecute un documento SSM al final de la operación de aplicación de revisione s y antes del reinicio del nodo administrado. Ejemplo de uso: Ejecute operaciones como la instalaci ón de aplicacio nes de terceros antes de un posible reinicio.	Ejecute un documento de SSM después de que finalice la operación de revisión y se hayan reiniciado las instancias. Ejemplo de uso: Asegúrese de que las aplicaciones se ejecuten según lo esperado tras la aplicación de revisiones.	No disponible
Do not reboot my instances (No reiniciar mis instancias)	Igual que lo mencionado anteriormente.	Ejecute un documento de SSM al final de la operación de revisióno. Ejemplo de uso: Asegúrese de que las aplicaciones se ejecuten según lo esperado tras	No disponible	No disponible

Opción de reinicio	Enlace: antes de la instalación	Enlace: después de la instalación	Enlace: en la salida	Enlace: después del reinicio programado
		la aplicación de revisiones.		
Schedule a reboot time (Programar una hora de reinicio)	Igual que lo mencionado anteriormente.	Igual que para Do not reboot my instances (No reiniciar mis instancias)	No disponible	Ejecute un documento de SSM inmediatamente después de que finalice el reinicio programado. Ejemplo de uso: Asegúrese de que las aplicaciones se ejecuten según lo esperado después del reinicio.

Ejecución de “Patch now” (Aplicar revisión ahora)

Utilice el siguiente procedimiento para aplicar revisiones a los nodos administrados bajo demanda.

Para ejecutar “Patch now” (Aplicar revisión ahora)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Patch Manager.
3. Elija Patch now (Aplicar revisión ahora).
4. En Patching operation (Operación de aplicación de revisiones), elija una de las siguientes opciones:

- Scan (Analizar): Patch Manager busca las revisiones que faltan en los nodos administrados, pero no las instala. Puede ver los resultados en el panel Compliance o en otras herramientas que utilice para la visualización de la conformidad de las revisiones.
- Scan and install (Analizar e instalar): Patch Manager busca las revisiones faltantes en los nodos administrados y las instala.

5. Utilice este paso solo si eligió la opción Scan and install (Analizar e instalar) en el paso anterior. En Reboot (Reinicio), elija una de las siguientes opciones:

   - Reboot if needed (Reiniciar si es necesario): luego de la instalación, Patch Manager reinicia los nodos administrados solo si es necesario para completar la instalación de una revisión.
   - Don't reboot my instances (No reiniciar las instancias): luego de la instalación, Patch Manager no reinicia los nodos administrados. Puede reiniciar los nodos administrados de forma manual en el momento que desee o administrar los reinicios fuera de Patch Manager.
   - Schedule a reboot time (Programar una hora de reinicio): especifique la fecha, la hora y la zona horaria UTC para que Patch Manager reinicie los nodos administrados. Después de ejecutar la operación Patch now (Aplicar revisión ahora), el reinicio programado aparece como asociación en State Manager con el nombre `AWS-PatchRebootAssociation`.

6. En Instances to patch (Instancias a las que se aplicarán revisiones), elija una de las siguientes opciones:

   - Patch all instances (Aplicar revisiones a todas las instancias): Patch Manager ejecuta la operación especificada en todos los nodos administrados en su Cuenta de AWS en la Región de AWS actual.
   - Patch only the target instances I specify (Aplicar revisiones solo a las instancias de destino especificadas): debe especificar los nodos administrados a los que se aplican revisiones en el siguiente paso.

7. Utilice este paso solo si elige Patch only the target instances I specify (Aplicar revisiones solo a las instancias de destino especificadas) en el paso anterior. En la sección Target selection (Selección de destinos), identifique los nodos en los que desea ejecutar esta operación, especifique las etiquetas, seleccione los nodos manualmente o especifique un grupo de recursos.

> **Note**
>
> Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.
> Si elige como destino un grupo de recursos, tenga en cuenta que estos grupos que se basan en una pila de AWS CloudFormation aún deben etiquetarse con la etiqueta `aws:cloudformation:stack-id` predeterminada. Si se ha eliminado, es posible que Patch Manager no pueda determinar cuáles son los nodos administrados que pertenecen al grupo de recursos.

8. (Opcional) En Patching log storage (Almacenamiento de registros de aplicación de revisiones), si desea crear y guardar registros de esta operación de aplicación de revisiones, seleccione el bucket de S3 para almacenar los registros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación del rol de servicio de IAM requerido para Systems Manager en entornos híbridos y multinube. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

9. Si desea ejecutar documentos de SSM como enlaces de ciclo de vida durante puntos específicos de la operación de aplicación de revisiones, haga lo siguiente:

   - Elija Use lifecycle hooks (Usar enlaces de ciclo de vida).
   - En cada enlace disponible, seleccione el documento de SSM a ejecutar en el punto especificado de la operación:
     - Antes de la instalación
     - Después de la instalación
     - A la salida

- Después del reinicio programado

> **Note**
>
> El documento predeterminado, `AWS-Noop`, no ejecuta operaciones.

10. Elija Patch now (Aplicar revisión ahora).

    Se abre la página Association execution summary (Resumen de la ejecución de la asociación). (La opción “Aplicar el parche ahora” utiliza asociaciones en State Manager, una herramienta de AWS Systems Manager, para sus operaciones). En el área Operation summary (Resumen de la operación), puede monitorear el estado del análisis o la aplicación de revisiones en los nodos administrados que especificó.

Trabajo con línea de base de revisiones

Una línea de base de revisiones en Patch Manager, una herramienta de AWS Systems Manager, define qué revisiones se aprueban para la instalación en los nodos administrados. Puede especificar revisiones aprobadas o rechazadas de forma de uno en uno. También puede crear reglas de aprobación automática para especificar que determinados tipos de actualizaciones (por ejemplo, las actualizaciones críticas) se deben aprobar automáticamente. La lista de rechazados anula las reglas y la lista de aprobados. Para utilizar una lista de revisiones aprobadas para instalar paquetes específicos, primero elimine las reglas de aprobación automática. Si identifica explícitamente una revisión como rechazada, no se aprobará ni instalará, aunque concuerde con todos los criterios de una regla de aprobación automática. Además, una revisión solo se instala en un nodo administrado si se aplica al software de este, aunque haya sido aprobada para el nodo administrado.

Temas

- Visualización de bases de referencia de parches predefinidas de AWS
- Uso de bases de referencia de parches personalizadas
- Configuración de una línea de base de revisiones existente como valor predeterminado

Más información

- líneas de base de revisiones

Visualización de bases de referencia de parches predefinidas de AWS

Patch Manager, una herramienta de AWS Systems Manager, cuenta con una línea de base de revisiones predefinida para todos los sistemas operativos compatibles con Patch Manager. Puede utilizar estas bases de referencia de parches (no puede personalizarlas) o puede crear una. El siguiente procedimiento describe cómo ver una base de referencia de parches predefinida para comprobar si cumple sus necesidades. Para obtener más información sobre las bases de referencia de parches, consulte Líneas de base de revisiones personalizadas y predefinidas.

Para ver bases de referencia de parches predefinidas de AWS

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Patch Manager.
3. En la lista de bases de referencia de parches, seleccione el ID de base de referencia de una de las bases de referencia de parches predefinida.

   -o bien-

   Si va a acceder a Patch Manager por primera vez en la Región de AWS actual, elija Comenzar con una descripción general, elija la pestaña Líneas de base de revisiones, y luego, elija el ID de la línea de base de una de las líneas de base de revisiones predefinidas.

   > **Note**
   >
   > Para Windows Server, se proporcionan tres líneas de base de revisiones predefinidas. Las líneas de base de revisiones `AWS-DefaultPatchBaseline` y `AWS-WindowsPredefinedPatchBaseline-OS` solo admiten actualizaciones del sistema operativo en el propio sistema operativo Windows. `AWS-DefaultPatchBaseline` se utiliza como base de referencia de revisiones predeterminada para los nodos administrados de Windows Server, a menos que se especifique una base de referencia distinta. Los ajustes de configuración en estas dos líneas de base de revisiones son los mismos. El más nuevo de los dos, `AWS-WindowsPredefinedPatchBaseline-OS`, se creó para que se diferenciara de la tercera línea de base de revisiones predefinida para Windows Server. Esa base de referencia de parches, `AWS-WindowsPredefinedPatchBaseline-OS-Applications`, puede utilizarse para aplicar parches al sistema operativo Windows Server y a las aplicaciones compatibles publicadas por Microsoft.

Para obtener más información, consulte Configuración de una línea de base de revisiones existente como valor predeterminado.

4. Elija la sección Reglas de aprobación y revise la configuración de la línea de base de revisiones.
5. Si la configuración es aceptable para los nodos administrados, puede pasar directamente al procedimiento Creación y administración de grupos de revisiones.

   -o bien-

   Para crear su propia base de referencia de parches predeterminada, continúe con el tema Uso de bases de referencia de parches personalizadas.

Uso de bases de referencia de parches personalizadas

Patch Manager, una herramienta de AWS Systems Manager, cuenta con una línea de base de revisiones predefinida para todos los sistemas operativos compatibles con Patch Manager. Puede utilizar estas bases de referencia de parches (no puede personalizarlas) o puede crear una.

En los siguientes procedimientos se describe cómo crear, actualizar y eliminar su propia base de referencia de parches personalizada. Para obtener más información sobre las bases de referencia de parches, consulte Líneas de base de revisiones personalizadas y predefinidas.

Temas

- Creación de una línea de base de revisiones personalizada para Linux
- Cómo crear una línea de base de revisiones personalizada para macOS
- Creación de una línea de base de revisiones personalizada para Windows Server
- Actualización o eliminación de una línea de base de revisiones personalizada

Creación de una línea de base de revisiones personalizada para Linux

Utilice el siguiente procedimiento para crear una línea de base de revisiones personalizada para nodos administrados de Linux en Patch Manager, una herramienta de AWS Systems Manager.

Para obtener información sobre la creación de una base de referencia de revisiones para nodos administrados macOS, consulte Cómo crear una línea de base de revisiones personalizada para macOS. Para obtener información sobre la creación de una base de referencia de revisiones para nodos administrados de Windows, consulte Creación de una línea de base de revisiones personalizada para Windows Server.

Para crear una base de referencia de revisiones personalizada para nodos administrados de Linux

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Patch Manager.
3. Seleccione la pestaña Bases de referencia de parches, y luego Crear una base de referencia de parches.

   -o bien-

   Si va a acceder a Patch Manager por primera vez en la Región de AWS actual, seleccione Comience por la información general, luego la pestaña Bases de referencia de parches y, por último, Crear una base de referencia de parches.
4. En Nombre, escriba un nombre para la nueva línea de base de revisiones; por ejemplo, `MyRHELPatchBaseline`.
5. (Opcional) En Description (Descripción), escriba una descripción para esta línea de base de revisiones.
6. En Operating system (Sistema operativo) elija un sistema operativo; por ejemplo, `Red Hat Enterprise Linux`.
7. Si desea empezar a utilizar esta línea de base de revisiones de forma predeterminada para el sistema operativo seleccionado tan pronto como la haya creado, active la casilla de verificación situada junto a Set this patch baseline as the default patch baseline for ***`operating system name`*** instances (Establecer esta línea de base de revisiones como la línea de base de revisiones para las instancias de [nombre del sistema operativo]).

   > **Note**
   >
   > Esta opción solo está disponible si accedió a Patch Manager por primera vez antes de las políticas de parches publicadas el 22 de diciembre de 2022.
   > Para obtener información sobre la configuración de una línea de base de revisiones existente como la opción predeterminada, consulte Configuración de una línea de base de revisiones existente como valor predeterminado.
8. En la sección Approval Rules for operating-systems (Reglas de aprobación para sistemas operativos), use los campos para crear una o varias reglas de aprobación automática.

- Productos: versión de los sistemas operativos a la que se aplica la regla de aprobación; por ejemplo, `RedhatEnterpriseLinux7.4`. La selección predeterminada es `All`.
- Clasificación: el tipo de revisiones a los que se aplica la regla de aprobación; como `Security` o `Enhancement`. La selección predeterminada es `All`.

> **Tip**
>
> Puede configurar una línea de base de revisiones para controlar si se instalan actualizaciones de versiones secundarias para Linux, como RHEL 7.8. Patch Manager puede instalar automáticamente actualizaciones de versiones secundarias siempre que la actualización esté disponible en el repositorio adecuado.
> Para los sistemas operativos Linux, las actualizaciones de versiones secundarias no se clasifican de forma consistente. Pueden clasificarse como correcciones de errores o actualizaciones de seguridad, o no clasificarse, incluso dentro de la misma versión del kernel. A continuación se muestran algunas opciones para controlar si una línea de base de revisiones las instala.
>
> - Opción 1: la regla de aprobación más amplia para garantizar que se instalen actualizaciones de versiones secundarias cuando estén disponibles es especificar Clasificación como `All` (*) y elegir la opción Incluir las actualizaciones que no sean de seguridad.
> - Opción 2: para asegurarse de que se instalan revisiones para una versión del sistema operativo, puede utilizar un comodín (*) para especificar el formato del kernel en la sección Excepciones de revisiones de la base de referencia. Por ejemplo, el formato del kernel para RHEL 7.* es `kernel-3.10.0-*.el7.x86_64`.
>
>   Ingrese `kernel-3.10.0-*.el7.x86_64` en la lista Approved patches (Revisiones aprobadas) de la base de referencia de revisiones para asegurarse de que todas las revisiones, incluidas las actualizaciones de versiones secundarias, se aplican a los nodos administrados de RHEL 7.*. (Si conoce el nombre exacto del paquete de una revisión de versión secundaria, puede escribirlo en su lugar).
> - Opción 3: puede tener el máximo control sobre qué revisiones se aplican a los nodos administrados, incluidas las actualizaciones de versiones secundarias, mediante el parámetro InstallOverrideList del documento `AWS-RunPatchBaseline`. Para obtener más información, consulte Documento de comandos SSM para aplicar revisiones: `AWS-RunPatchBaseline`.

- Severity (Gravedad): el valor de gravedad de las revisiones a los que se aplica la regla; como `Critical`. La selección predeterminada es `All`.
- Auto-approval (Aprobación automática): método para seleccionar revisiones para su aprobación automática.

> **Note**
>
> Debido a que no es posible determinar de forma fiable las fechas de lanzamiento de los paquetes de actualización para Ubuntu Server, las opciones de aprobación automática no son compatibles con este sistema operativo.

  - Approve patches after a specified number of days (Aprobar las revisiones después de una cantidad determinada de días): la cantidad de días que Patch Manager debe esperar después de lanzar o actualizar por última vez una revisión y antes de que se apruebe automáticamente. Puede ingresar cualquier número entero entre cero (0) y 360. En la mayoría de los casos, se recomienda no esperar más de 100 días.
  - Approve patches released up to a specific date (Aprobar las revisiones publicadas hasta una fecha específica): la fecha de lanzamiento de la revisión para la que Patch Manager aplica automáticamente todas las revisiones publicadas o actualizadas en esa fecha o con anterioridad a ella. Por ejemplo, si especifica el 7 de julio de 2023, no se instalarán automáticamente las revisiones publicadas o actualizadas a partir del 8 de julio de 2023.
- (Opcional). Informes de conformidad: el nivel de gravedad que desea asignar a las revisiones aprobadas por la línea de base, como `Critical` o `High`.

> **Note**
>
> Si especifica un nivel de notificación de conformidad y se informa el estado de cualquier revisión aprobada como `Missing`, la gravedad de la conformidad general notificada por la línea de base de revisiones será el nivel de gravedad que especificó.

- Include non-security updates (Incluir actualizaciones que no son de seguridad): seleccione esta casilla de verificación para instalar las revisiones del sistema operativo Linux que no son de seguridad y que están disponibles en el repositorio de origen, además de las revisiones relacionados con la seguridad.

> **Note**
>
> En SUSE Linux Enterprise Server (SLES), no es necesario seleccionar la casilla de verificación, ya que tanto las revisiones de seguridad como las que no lo son se instalan de forma predeterminada en los nodos administrados de SLES. Para obtener más información, consulte el contenido sobre SLES en Cómo se seleccionan las revisiones de seguridad.

Para obtener más información sobre cómo trabajar con reglas de aprobación en una línea de base de revisiones personalizada, consulte Líneas de base personalizadas.

9. Si desea aprobar alguna revisión de forma explícita además de los que cumplan las reglas de aprobación, haga lo siguiente en la sección Patch exceptions (Excepciones de revisiones):

   - En Approved patches (revisiones aprobados) escriba una lista separada por comas de las revisiones que desea aprobar.

     Para obtener información acerca de los formatos aceptados para las Listas de revisiones aprobados y rechazados, consulte Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas.
   - (Opcional). En Approved patches compliance level (Nivel de conformidad de revisiones aprobados), asigne un nivel de conformidad a las revisiones de la lista.
   - Si alguna de las revisiones aprobadas que ha especificado no está relacionada con la seguridad, seleccione la casilla de verificación Incluir actualizaciones no relacionadas con la seguridad para que se instalen también estas revisiones en su sistema operativo Linux.

10. Si desea rechazar alguna revisión de forma explícita que cumpla las reglas de aprobación, haga lo siguiente en la sección Patch exceptions (Excepciones de revisiones):

    - En Rejected patches (revisiones rechazados) escriba una lista separada por comas de las revisiones que desea rechazar.

      Para obtener información acerca de los formatos aceptados para las Listas de revisiones aprobados y rechazados, consulte Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas.
    - En Rejected patches action (Acción de revisiones rechazados), seleccione la acción que Patch Manager realizará en las revisiones de la lista Rejected patches (revisiones rechazados).

- Allow as dependency (Permitir como dependencia): un paquete en la lista Rejected patches (revisiones rechazados) solo se instala si es una dependencia de otro paquete. Se considera conforme con la línea de base de revisiones y su estado se registra como InstalledOther. Esta es la opción predeterminada si no se especifica ninguna opción.
- Bloquear: Patch Manager no instala, bajo ninguna circunstancia, los paquetes de la lista Parches rechazados y los paquetes que los incluyen como dependencias. Si un paquete se instaló antes de agregarlo a la lista Parches rechazados, o si luego se instala por fuera de Patch Manager, se considera no conforme con la línea de base de revisiones y su estado se reporta como InstalledRejected.

> **Note**
>
> Patch Manager busca las dependencias de las revisiones de forma recursiva.

11. (Opcional) Si desea especificar repositorios de revisiones alternativos para diferentes versiones de un sistema operativo, como AmazonLinux2016.03 y AmazonLinux2017.09, haga lo siguiente para cada producto en la sección Patch sources (Orígenes de revisiones):

    - En Name (Nombre), escriba un nombre que le ayude a identificar la configuración de origen.
    - En Product (Producto), seleccione la versión de los sistemas operativos a los que va dirigido el repositorio de origen de revisiones, por ejemplo `RedhatEnterpriseLinux7.4`.
    - En Configuración, ingrese el valor de la configuración del repositorio que desea utilizar en el formato apropiado:

      Example for yum repositories

      ```
      [main]
      name=MyCustomRepository
      baseurl=https://my-custom-repository
      enabled=1
      ```

      > **Tip**
      >
      > Para obtener información sobre otras opciones disponibles para la configuración del repositorio yum, consulte dnf.conf(5).

Examples for Servidor Ubuntu and Servidor Debian

```
deb http://security.ubuntu.com/ubuntu jammy main
```

```
deb https://site.example.com/debian distribution component1
component2 component3
```

La información del repositorio para los repositorios de Ubuntu Server debe especificarse en una sola línea. Para obtener más ejemplos e información, consulte jammy (5) sources.list.5.gz en el sitio web Manuales del servidor Ubuntu y sources.list format en la wiki de Debian.

Elija Add another source (Añadir otro origen) para especificar un repositorio de origen para cada versión adicional del sistema operativo, hasta un máximo de 20.

Para obtener más información sobre los repositorios de origen de revisiones alternativos, consulte Cómo especificar un repositorio de origen de parches alternativo (Linux).

12. (Opcional) En Manage tags (Administrar etiquetas), aplique uno o varios pares de claves nombre/valor al a la línea de base de revisiones.

    Las etiquetas son metadatos opcionales que usted asigna a un recurso. Las etiquetas permiten clasificar los recursos de diversas maneras, por ejemplo, según la finalidad, el propietario o el entorno. Por ejemplo, es posible que desee etiquetar una línea de base de revisiones para identificar el nivel de seguridad de revisiones que especifica, la familia de sistemas operativos a la que se aplica y el tipo de entorno. En este caso, puede especificar etiquetas similares a los siguientes pares de claves nombre-valor:

    - `Key=PatchSeverity,Value=Critical`
    - `Key=OS,Value=RHEL`
    - `Key=Environment,Value=Production`

13. Elija Create patch baseline (Crear base de referencia de revisiones).

Cómo crear una línea de base de revisiones personalizada para macOS

Utilice el siguiente procedimiento para crear una línea de base de revisiones personalizada para nodos administrados de macOS en Patch Manager, una herramienta de AWS Systems Manager.

Para obtener información sobre la creación de una base de referencia de revisiones para nodos administrados Windows Server, consulte Creación de una línea de base de revisiones personalizada para Windows Server. Para obtener información sobre la creación de una base de referencia de revisiones para nodos administrados de Linux, consulte Creación de una línea de base de revisiones personalizada para Linux.

> **Note**
>
> macOS no se admite en todas las Regiones de AWS. Para obtener más información sobre la compatibilidad de Amazon EC2 con macOS, consulte Instancias de Mac de Amazon EC2 en la Guía del usuario de Amazon EC2.

Cómo crear una base de referencia de revisiones personalizada para nodos administrados de macOS

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Patch Manager.
3. Seleccione la pestaña Bases de referencia de parches, y luego Crear una base de referencia de parches.

   -o bien-

   Si va a acceder a Patch Manager por primera vez en la Región de AWS actual, seleccione Comience por la información general, luego la pestaña Bases de referencia de parches y, por último, Crear una base de referencia de parches.
4. En Nombre, escriba un nombre para la nueva línea de base de revisiones; por ejemplo, `MymacOSPatchBaseline`.
5. (Opcional) En Description (Descripción), escriba una descripción para esta línea de base de revisiones.
6. En Operating system (Sistema operativo), elija macOS.
7. Si desea empezar a utilizar esta base de referencia de parches de forma predeterminada para macOS tan pronto como la haya creado, tilde la casilla de verificación situada junto a Set this patch baseline as the default patch baseline for macOS instances (Establezca esta base de referencia de parches como la predeterminada para las instancias de macOS).

> **Note**
>
> Esta opción solo está disponible si accedió a Patch Manager por primera vez antes de las políticas de parches publicadas el 22 de diciembre de 2022.
> Para obtener información sobre la configuración de una línea de base de revisiones existente como la opción predeterminada, consulte Configuración de una línea de base de revisiones existente como valor predeterminado.

8. En la sección Approval Rules for operating-systems (Reglas de aprobación para sistemas operativos), use los campos para crear una o varias reglas de aprobación automática.

   - Productos: versión de los sistemas operativos a la que se aplica la regla de aprobación; por ejemplo, `Mojave10.14.1` o `Catalina10.15.1`. La selección predeterminada es `All`.

     > **Note**
     >
     > El sistema de administración de paquetes de software de código abierto Homebrew ya no ofrece soporte para macOS 10.14.x (Mojave) y 10.15.x (Catalina). Como resultado, no se admiten las operaciones de revisión con Patch Manager en estas versiones.

   - Classification (Clasificación): el administrador o los administradores de paquetes a los que desea aplicar los paquetes durante el proceso de aplicación de parches. Puede elegir entre las siguientes opciones:
     - softwareupdate
     - installer (instalador)
     - brew
     - brew cask

     La selección predeterminada es `All`.
   - (Opcional). Informes de conformidad: el nivel de gravedad que desea asignar a las revisiones aprobadas por la línea de base, como `Critical` o `High`.

> **Note**
>
> Si especifica un nivel de notificación de conformidad y se informa el estado de cualquier revisión aprobada como `Missing`, la gravedad de la conformidad general notificada por la línea de base de revisiones será el nivel de gravedad que especificó.

Para obtener más información sobre cómo trabajar con reglas de aprobación en una línea de base de revisiones personalizada, consulte Líneas de base personalizadas.

9. Si desea aprobar alguna revisión de forma explícita además de los que cumplan las reglas de aprobación, haga lo siguiente en la sección Patch exceptions (Excepciones de revisiones):

   - En Approved patches (revisiones aprobados) escriba una lista separada por comas de las revisiones que desea aprobar.

     Para obtener información acerca de los formatos aceptados para las Listas de revisiones aprobados y rechazados, consulte Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas.

   - (Opcional). En Approved patches compliance level (Nivel de conformidad de revisiones aprobados), asigne un nivel de conformidad a las revisiones de la lista.

10. Si desea rechazar alguna revisión de forma explícita que cumpla las reglas de aprobación, haga lo siguiente en la sección Patch exceptions (Excepciones de revisiones):

    - En Rejected patches (revisiones rechazados) escriba una lista separada por comas de las revisiones que desea rechazar.

      Para obtener información acerca de los formatos aceptados para las Listas de revisiones aprobados y rechazados, consulte Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas.

    - En Rejected patches action (Acción de revisiones rechazados), seleccione la acción que Patch Manager realizará en las revisiones de la lista Rejected patches (revisiones rechazados).
      - Allow as dependency (Permitir como dependencia): un paquete en la lista Rejected patches (revisiones rechazados) solo se instala si es una dependencia de otro paquete. Se considera conforme con la línea de base de revisiones y su estado se registra como InstalledOther. Esta es la opción predeterminada si no se especifica ninguna opción.

- Bloquear: Patch Manager no instala, bajo ninguna circunstancia, los paquetes de la lista Parches rechazados y los paquetes que los incluyen como dependencias. Si un paquete se instaló antes de agregarlo a la lista Parches rechazados, o si luego se instala por fuera de Patch Manager, se considera no conforme con la línea de base de revisiones y su estado se reporta como InstalledRejected.

11. (Opcional) En Manage tags (Administrar etiquetas), aplique uno o varios pares de claves nombre/valor al a la línea de base de revisiones.

    Las etiquetas son metadatos opcionales que usted asigna a un recurso. Las etiquetas permiten clasificar los recursos de diversas maneras, por ejemplo, según la finalidad, el propietario o el entorno. Por ejemplo, es posible que desee etiquetar una base de referencia de parches para identificar el nivel de gravedad de parches que especifica, el administrador de paquetes al que se aplica y el tipo de entorno. En este caso, puede especificar etiquetas similares a los siguientes pares de claves nombre-valor:

    - `Key=PatchSeverity,Value=Critical`
    - `Key=PackageManager,Value=softwareupdate`
    - `Key=Environment,Value=Production`

12. Elija Create patch baseline (Crear base de referencia de revisiones).

Creación de una línea de base de revisiones personalizada para Windows Server

Utilice el siguiente procedimiento para crear una línea de base de revisiones personalizada para nodos administrados de Windows en Patch Manager, una herramienta de AWS Systems Manager.

Para obtener información sobre la creación de una base de referencia de revisiones para nodos administrados de Linux, consulte Creación de una línea de base de revisiones personalizada para Linux. Para obtener información sobre la creación de una línea de base de revisiones para nodos administrados macOS, consulte Cómo crear una línea de base de revisiones personalizada para macOS.

Para obtener un ejemplo de creación de una línea de base de revisiones limitada para instalar únicamente Service Packs de Windows, consulte Tutorial: creación de una línea de base de revisiones para instalar Service Packs de Windows mediante la consola.

Para crear una línea de base de revisiones personalizada (Windows)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Patch Manager.
3. Seleccione la pestaña Bases de referencia de parches, y luego Crear una base de referencia de parches.

   -o bien-

   Si va a acceder a Patch Manager por primera vez en la Región de AWS actual, seleccione Comience por la información general, luego la pestaña Bases de referencia de parches y, por último, Crear una base de referencia de parches.
4. En Nombre, escriba un nombre para la nueva línea de base de revisiones; por ejemplo, `MyWindowsPatchBaseline`.
5. (Opcional) En Description (Descripción), escriba una descripción para esta línea de base de revisiones.
6. En Operating system (Sistema operativo), elija `Windows`.
7. En Estado de cumplimiento de las actualizaciones de seguridad disponibles, elija el estado que desee asignar a las revisiones de seguridad que están disponibles pero no aprobadas, porque no cumplen con los criterios de instalación especificados en la línea de base de revisiones: No conforme o Conforme.

   Situación de ejemplo: las revisiones de seguridad que desee instalar se pueden omitir si ha especificado un periodo prolongado de espera después del lanzamiento de la revisión antes de la instalación. Si se publica una actualización de la revisión durante el periodo de espera especificado, el periodo de espera para instalar la revisión vuelve a comenzar. Si el periodo de espera es demasiado extenso, es posible que se lancen varias versiones de la revisión, pero nunca se instalen.
8. Si desea comenzar a utilizar esta línea de base de revisiones de forma predeterminada para Windows tan pronto como la cree, seleccione Set this patch baseline as the default patch baseline for Windows Server instances (Establecer esta línea de base de revisiones como la línea de base de revisiones predeterminada para las instancias de Windows Server).

> **Note**
>
> Esta opción solo está disponible si accedió a Patch Manager por primera vez antes de las políticas de parches publicadas el 22 de diciembre de 2022.
> Para obtener información sobre la configuración de una línea de base de revisiones existente como la opción predeterminada, consulte Configuración de una línea de base de revisiones existente como valor predeterminado.

9. En la sección Approval Rules for operating systems (Reglas de aprobación para sistemas operativos), use los campos para crear una o varias reglas de aprobación automática.

   - Productos: versión de los sistemas operativos a la que se aplica la regla de aprobación; por ejemplo, `WindowsServer2012`. La selección predeterminada es `All`.
   - Clasificación: el tipo de revisiones a los que se aplica la regla de aprobación; como `CriticalUpdates`, `Drivers` y `Tools`. La selección predeterminada es `All`.

     > **Tip**
     >
     > Puede incluir instalaciones de Service Packs de Windows en las reglas de aprobación incluyendo `ServicePacks` o eligiendo `All` en la lista Clasificación. Para ver un ejemplo, consulta Tutorial: creación de una línea de base de revisiones para instalar Service Packs de Windows mediante la consola.

   - Severity (Gravedad): el valor de gravedad de las revisiones a los que se aplica la regla; como `Critical`. La selección predeterminada es `All`.
   - Auto-approval (Aprobación automática): método para seleccionar revisiones para su aprobación automática.
     - Approve patches after a specified number of days (Aprobar las revisiones después de una cantidad determinada de días): la cantidad de días que Patch Manager debe esperar después de que se lance o actualice una revisión y antes de que se apruebe automáticamente. Puede ingresar cualquier número entero entre cero (0) y 360. En la mayoría de los casos, se recomienda no esperar más de 100 días.
     - Approve patches released up to a specific date (Aprobar las revisiones publicadas hasta una fecha específica): la fecha de lanzamiento de la revisión para la que Patch Manager aplica automáticamente todas las revisiones publicadas o actualizadas en esa fecha o

con anterioridad a ella. Por ejemplo, si especifica el 7 de julio de 2023, no se instalarán automáticamente las revisiones publicadas o actualizadas a partir del 8 de julio de 2023.

- (Opcional). Compliance reporting (Informes de conformidad): el nivel de gravedad que desea asignar a las revisiones aprobadas por la base de referencia, como `High`.

> **Note**
>
> Si especifica un nivel de notificación de conformidad y se informa el estado de cualquier revisión aprobada como `Missing`, la gravedad de la conformidad general notificada por la línea de base de revisiones será el nivel de gravedad que especificó.

10. (Opcional) En la sección Approval rules for applications (Reglas de aprobación para aplicaciones) use los campos para crear una o más reglas de aprobación automática.

> **Note**
>
> En lugar de especificar reglas de aprobación, puede especificar Listas de revisiones aprobados y rechazados como excepciones de revisiones. Consulte los pasos 10 y 11.

- Product family (Familia de productos): la familia de productos de Microsoft generales para la que desea especificar una regla, como, por ejemplo, `Office` o `Exchange Server`.
- Productos: versión de la aplicación a la que se aplica la regla de aprobación; por ejemplo, `Office 2016` o `Active Directory Rights Management Services Client 2.0 2016`. La selección predeterminada es `All`.
- Classification (Clasificación): el tipo de revisiones a los que se aplica la regla de aprobación; como `CriticalUpdates`. La selección predeterminada es `All`.
- Severity (Gravedad): el valor de gravedad de las revisiones a los que se aplica la regla, como `Critical`. La selección predeterminada es `All`.
- Auto-approval (Aprobación automática): método para seleccionar revisiones para su aprobación automática.
  - Approve patches after a specified number of days (Aprobar las revisiones después de una cantidad determinada de días): la cantidad de días que Patch Manager debe esperar después de que se lance o actualice una revisión y antes de que se apruebe automáticamente. Puede ingresar cualquier número entero entre cero (0) y 360. En la mayoría de los casos, se recomienda no esperar más de 100 días.

- Approve patches released up to a specific date (Aprobar las revisiones publicadas hasta una fecha específica): la fecha de lanzamiento de la revisión para la que Patch Manager aplica automáticamente todas las revisiones publicadas o actualizadas en esa fecha o con anterioridad a ella. Por ejemplo, si especifica el 7 de julio de 2023, no se instalarán automáticamente las revisiones publicadas o actualizadas a partir del 8 de julio de 2023.

- (Opcional). Informes de conformidad: el nivel de gravedad que desea asignar a las revisiones aprobadas por la línea de base, como `Critical` o `High`.

> **Note**
>
> Si especifica un nivel de notificación de conformidad y se informa el estado de cualquier revisión aprobada como `Missing`, la gravedad de la conformidad general notificada por la línea de base de revisiones será el nivel de gravedad que especificó.

11. (Opcional) Si desea aprobar explícitamente las revisiones en lugar de dejar que estos se seleccionen conforme a las reglas de aprobación, haga lo siguiente en la sección Patch exceptions (Excepciones de revisiones):

    - En Approved patches (revisiones aprobados) escriba una lista separada por comas de las revisiones que desea aprobar.

      Para obtener información acerca de los formatos aceptados para las Listas de revisiones aprobados y rechazados, consulte Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas.

    - (Opcional). En Approved patches compliance level (Nivel de conformidad de revisiones aprobados), asigne un nivel de conformidad a las revisiones de la lista.

12. Si desea rechazar alguna revisión de forma explícita que cumpla las reglas de aprobación, haga lo siguiente en la sección Patch exceptions (Excepciones de revisiones):

    - En Rejected patches (revisiones rechazados) escriba una lista separada por comas de las revisiones que desea rechazar.

      Para obtener información acerca de los formatos aceptados para las Listas de revisiones aprobados y rechazados, consulte Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas.

    - En Rejected patches action (Acción de revisiones rechazados), seleccione la acción que Patch Manager realizará en las revisiones de la lista Rejected patches (revisiones rechazados).

- Permitir como dependencia: Windows Server no admite el concepto de dependencias de paquetes. Si un paquete está en la lista de revisiones rechazadas y ya está instalada en el nodo, su estado se indica como `INSTALLED_OTHER`. Se omite cualquier paquete que aún no esté instalado en el nodo.
- Bloquear: Patch Manager no instala los paquetes de la lista de revisiones rechazadas bajo ninguna circunstancia. Si un paquete se instaló antes de agregarlo a la lista Versiones rechazadas, o si luego se instala por fuera de Patch Manager, se considera no conforme con la línea de base de revisiones y su estado se reporta como `INSTALLED_REJECTED`.

Para obtener más información sobre las acciones de los paquetes rechazados, consulte las opciones de la lista de revisiones rechazadas en las líneas de base de revisiones personalizadas.

13. (Opcional) En Manage tags (Administrar etiquetas), aplique uno o varios pares de claves nombre/valor al a la línea de base de revisiones.

    Las etiquetas son metadatos opcionales que usted asigna a un recurso. Las etiquetas permiten clasificar los recursos de diversas maneras, por ejemplo, según la finalidad, el propietario o el entorno. Por ejemplo, es posible que desee etiquetar una línea de base de revisiones para identificar el nivel de seguridad de revisiones que especifica, la familia de sistemas operativos a la que se aplica y el tipo de entorno. En este caso, puede especificar etiquetas similares a los siguientes pares de claves nombre-valor:

    - `Key=PatchSeverity,Value=Critical`
    - `Key=OS,Value=RHEL`
    - `Key=Environment,Value=Production`
14. Elija Create patch baseline (Crear base de referencia de revisiones).

Actualización o eliminación de una línea de base de revisiones personalizada

Puede actualizar o eliminar una línea de base de revisiones personalizada que haya creado en Patch Manager, una herramienta de AWS Systems Manager. Al actualizar una línea de base de revisiones, puede cambiar su nombre o descripción, sus reglas de aprobación y sus excepciones para revisiones aprobadas y rechazadas. También puede actualizar las etiquetas que se aplican a la línea de base de revisiones. No se puede cambiar el tipo de sistema operativo para el que se ha creado una línea de base de revisiones y no puede realizar cambios en una línea de base de revisiones predefinida que AWS proporciona.

Actualización o eliminación de una línea de base de revisiones

Siga estos pasos para actualizar o eliminar una línea de base de revisiones.

> **Important**
>
> Tenga cuidado al eliminar una línea de base de revisiones personalizada que pueda utilizar una configuración de política de revisiones en Quick Setup.
> Si utiliza una configuración de política de revisiones en Quick Setup, las actualizaciones que realice en las líneas de base de revisiones personalizadas se sincronizan con Quick Setup cada hora.
> Si se elimina una línea de base de revisiones personalizada a la que se hacía referencia en una política de revisiones, aparece un banner en la página Configuration details (Detalles de configuración) de Quick Setup correspondiente a la política de revisiones. El banner le informa que la política de revisiones hace referencia a una línea de base de revisiones que ya no existe y que las operaciones de aplicación de revisiones posteriores fallarán. En este caso, vuelva a la página Configurations (Configuraciones) de Quick Setup, seleccione la configuración de Patch Manager y elija Actions (Acciones), Edit configuration (Editar configuración). El nombre de la línea de base de revisiones eliminado aparece resaltado y debe seleccionar una nueva línea de base de revisiones para el sistema operativo afectado.

Para actualizar o eliminar una línea de base de revisiones

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Patch Manager.
3. Elija la línea de base de revisiones que desea actualizar o eliminar y, a continuación, lleve a cabo alguna de las siguientes operaciones:

   - Para eliminar la línea de base de revisiones de su Cuenta de AWS, elija Delete (Eliminar). El sistema le pedirá que confirme sus acciones.
   - Para cambiar el nombre o la descripción de la línea de base de revisiones, las reglas de aprobación o las excepciones de revisiones, elija Edit (Editar). En la página Edit patch baseline (Editar línea de base de revisiones), cambie los valores y las opciones que desee y, a continuación, elija Save changes (Guardar cambios).

- Para añadir, cambiar o eliminar etiquetas aplicadas a la línea de base de revisiones, elija la pestaña Tags (Etiquetas) y, a continuación, elija Edit tags (Editar etiquetas). En la página Edit patch baseline tags (Editar etiquetas de línea de base de revisiones), lleve a cabo actualizaciones de las etiquetas de la línea de base de revisiones y, a continuación, elija Save changes (Guardar cambios).

Para obtener más información acerca de las opciones de configuración que puede realizar, consulte Uso de bases de referencia de parches personalizadas.

Configuración de una línea de base de revisiones existente como valor predeterminado

> **Important**
>
> Las selecciones de línea de base de revisiones predeterminadas que realice aquí no se aplicarán a las operaciones de aplicación de revisiones basadas en una política de revisiones. Las políticas de revisiones utilizan sus propias especificaciones de línea de base de revisiones. Para obtener más información sobre las políticas de revisiones, consulte Configuraciones de políticas de revisiones en Quick Setup.

Cuando se crea una línea de base de revisiones personalizada en Patch Manager, una herramienta de AWS Systems Manager, puede establecer la base de referencia como valor predeterminado para el tipo de sistema operativo asociado tan pronto como la crea. Para obtener más información, consulta Uso de bases de referencia de parches personalizadas.

También puede definir una línea de base de revisiones ya existente de como valor predeterminado para un tipo de sistema operativo.

> **Note**
>
> Los pasos que siga dependerán de si accedió por primera vez a Patch Manager antes o después del lanzamiento de las políticas de revisiones el 22 de diciembre de 2022. Si utilizó Patch Manager antes de esa fecha, puede utilizar el procedimiento de la consola. De lo contrario, utilice el procedimiento de AWS CLI. El menú Acciones al que se hace referencia en el procedimiento de la consola no se muestra en las regiones en las que Patch Manager no se utilizaba antes de la publicación de las políticas de revisiones.

Para definir una línea de base de revisiones como la opción predeterminada

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Patch Manager.
3. Elija la pestaña Patch baselines (Bases de referencia de parches).
4. En la lista de bases de referencia de parches, elija el botón de una base de referencia de parches que no esté definida actualmente como la predeterminada para un tipo de sistema operativo.

   La columna Default baseline (Base de referencia de revisiones predeterminada) indica qué líneas de base de revisiones están actualmente definidas como los valores predeterminados.
5. En el menú Actions (Acciones), elija Set default patch baseline (Definir línea de base de revisiones predeterminada).

   > **Important**
   >
   > El menú Acciones no está disponible si no trabajó con Patch Manager en la Cuenta de AWS y la región actuales antes del 22 de diciembre de 2022. Para obtener más información, consulte la Nota que aparece anteriormente en este tema.
6. En el cuadro de diálogo de confirmación, elija Set default (Definir valor predeterminado).

Para definir una línea de base de revisiones como la opción predeterminada (AWS CLI)

1. Ejecute el comando describe-patch-baselines para ver una lista de las líneas de base de revisiones disponibles y sus ID y nombres de recursos de Amazon (ARN).

   ```
   aws ssm describe-patch-baselines
   ```
2. Ejecute el comando register-default-patch-baseline para establecer una línea de base como predeterminada para el sistema operativo al que está asociada. Reemplace *baseline-id-or-ARN* con el ID de la línea de base de revisiones personalizada o la línea de base de preferencia que se utilizará.

   Linux & macOS

   ```
   aws ssm register-default-patch-baseline \
   ```

```
    --baseline-id baseline-id-or-ARN
```

A continuación, se muestra un ejemplo de configuración de una línea de base personalizada como predeterminada.

```
aws ssm register-default-patch-baseline \
    --baseline-id pb-abc123cf9bEXAMPLE
```

A continuación, se muestra un ejemplo de configuración de una línea de base de preferencia administrada por AWS como predeterminada.

```
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
```

A continuación, se muestra un ejemplo de configuración de una línea de base personalizada como predeterminada.

```
aws ssm register-default-patch-baseline ^
    --baseline-id pb-abc123cf9bEXAMPLE
```

A continuación, se muestra un ejemplo de configuración de una línea de base de preferencia administrada por AWS como predeterminada.

```
aws ssm register-default-patch-baseline ^
    --baseline-id arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-071da192df1226b63
```

Visualización de revisiones disponibles

Patch Manager, una herramienta de AWS Systems Manager, permite que vea todos los parches disponibles para un sistema operativo especificado y, de manera opcional, una versión específica de este.

> **Tip**
>
> Para generar una lista de revisiones disponibles y guardarlos en un archivo, puede utilizar el comando describe-available-patches y especificar su salida preferida.

Para ver los parches disponibles

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Patch Manager.
3. Elija la pestaña Patches (Parches).

   -o bien-

   Si va a acceder a Patch Manager por primera vez en la Región de AWS actual, seleccione Comience por la información general, y luego la pestaña Parches.

   > **Note**
   >
   > Para Windows Server, la pestaña Revisiones muestra las actualizaciones que están disponibles en el Servicio de actualizaciones de Windows Server (WSUS).

4. En Operating system (Sistema operativo), elija el sistema operativo para el que desea ver los parches disponibles, como `Windows` o `Amazon Linux`.
5. (Opcional) En Product (Producto), elija una versión del sistema operativo, como `WindowsServer2019` o `AmazonLinux2018.03`.
6. (Opcional) Para agregar o eliminar columnas de información para sus resultados, elija el botón de configuración
   ()
   en la parte superior derecha de la lista Patches (Parches). (De forma predeterminada, la pestaña Patches (Parches) muestra columnas solo para algunos de los metadatos de parches disponibles).

   Para obtener información acerca de los tipos de metadatos que puede agregar a la vista, consulte Parche en la Referencia de la API de AWS Systems Manager.

Creación y administración de grupos de revisiones

Si no utiliza políticas de revisiones en las operaciones, puede utilizar las acciones de aplicación de revisiones mediante el uso de etiquetas para agregar nodos administrados a grupos de revisiones.

> **Note**
>
> Los grupos de revisiones no se usan en operaciones de aplicación de revisiones basadas en políticas de revisiones. Para obtener información sobre el uso de las políticas de revisiones, consulte Configuraciones de políticas de revisiones en Quick Setup.
> La consola no admite la funcionalidad de grupos de parches para pares de cuentas y regiones que no usaban grupos de parches antes de que se publicara la compatibilidad con las políticas de parches el 22 de diciembre de 2022. La funcionalidad de grupos de parches sigue estando disponible en los pares de cuentas y regiones que empezaron a usar grupos de parches antes de esta fecha.

Para utilizar etiquetas en las operaciones de aplicación de revisiones, debe aplicar la clave de etiqueta `Patch Group` o `PatchGroup` a los nodos administrados. También debe especificar el nombre que quiere dar al grupo de revisiones como el valor de la etiqueta. Puede especificar cualquier valor de etiqueta, pero la clave de etiqueta debe ser `Patch Group` o `PatchGroup`.

Tiene que usar `PatchGroup` (sin espacio), si ha permitido las etiquetas en los metadatos de las instancias de EC2.

Después de agrupar los nodos administrados mediante etiquetas, tiene que agregar el valor de grupo de revisiones a una base de referencia de revisiones. Al registrar el grupo de revisiones en una línea de base de revisiones, se asegura de que se instalen las revisiones correctas durante la operación de aplicación de revisiones. Para obtener más información acerca de los grupos de revisiones, consulte Grupos de revisiones.

Complete las tareas de este tema para preparar los nodos administrados para la aplicación de revisiones mediante etiquetas con los nodos y la línea de base de revisiones. La tarea 1 solo es necesaria si está implementando revisiones a las instancias de Amazon EC2. La tarea 2 solo es necesaria si va a aplicar revisiones a instancias que no son de EC2 en un entorno híbrido y multinube. La tarea 3 es necesaria para todos los nodos administrados.

> **Tip**
>
> Además, puede agregar etiquetas a nodos administrados mediante el AWS CLI comando `add-tags-to-resource` de la o la operación de API de Systems Manager ssm-agent-minimum-s3-permissions-required`AddTagsToResource`.

Tareas

- Tarea 1: agregar instancias de EC2 a un grupo de revisiones mediante etiquetas
- Tarea 2: agregar nodos administrados a un grupo de revisiones mediante etiquetas
- Tarea 3: añadir un grupo de revisiones a una línea de base de revisiones

Tarea 1: agregar instancias de EC2 a un grupo de revisiones mediante etiquetas

Puede agregar etiquetas a instancias de EC2 administradas mediante la consola de Amazon EC2 o la línea de comandos de Systems Manager. Esta tarea solo es necesaria si está aplicando revisiones a las instancias de Amazon EC2.

> **Important**
>
> Puede aplicar la etiqueta `Patch Group` (con un espacio) a una instancia de Amazon EC2 si la opción Allow tags in instance metadata (Permitir etiquetas en los metadatos de la instancia) no puede estar habilitada en la instancia. Al permitir etiquetas en los metadatos de la instancia, se impide que los nombres de las claves de las etiquetas contengan espacios. Si tiene etiquetas permitidas en metadatos de instancias de EC2, debe usar la clave de etiqueta `PatchGroup` (sin espacio).

Opción 1: agregar instancias de EC2 administradas a un grupo de revisiones (consola de Systems Manager)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. En la lista Nodos administrados, elija el ID de una instancia de EC2 administrada que desee configurar para la aplicación de revisiones. Los ID de nodo de las instancias de EC2 comienzan con `i-`.

> **Note**
>
> Cuando se utiliza la consola de Amazon EC2 y la AWS CLI, es posible aplicar etiquetas `Key = Patch Group` o `Key = PatchGroup` a instancias que aún no están configuradas para utilizarlas con Systems Manager.
> Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

4. Seleccione la pestaña Etiquetas y, luego, elija Editar.
5. En la columna izquierda, ingrese **`Patch Group`** o **`PatchGroup`**. Si ha permitido las etiquetas en los metadatos de las instancias de EC2, debe usar `PatchGroup` (sin espacio).
6. En la columna de la derecha, ingrese un valor de etiqueta que lo identifique como nombre para este grupo de revisiones.
7. Seleccione Save.
8. Repita este procedimiento para agregar otras instancias de EC2 en el mismo grupo de revisiones.

Opción 2: agregar instancias de EC2 a un grupo de revisiones (consola de Amazon EC2)

1. Abra la consola de Amazon EC2 y, a continuación, elija Instances (Instancias) en el panel de navegación.
2. En la lista de instancias, elija la que desea configurar para la aplicación de revisiones.
3. En el menú Acciones, elija Configuración de instancia, Administrar etiquetas.
4. Elija Añadir nueva etiqueta.
5. En Key (Clave), ingrese **`Patch Group`** o **`PatchGroup`**. Si ha permitido las etiquetas en los metadatos de las instancias de EC2, debe usar `PatchGroup` (sin espacio).
6. Para Valor, escriba un valor que sirva como nombre para este grupo de revisiones.
7. Seleccione Save.
8. Repita este procedimiento para añadir otras instancias en el mismo grupo de revisiones.

Tarea 2: agregar nodos administrados a un grupo de revisiones mediante etiquetas

Siga los pasos de este tema para añadir etiquetas a los dispositivos principales AWS IoT Greengrass y a los nodos administrados activados de manera híbrida (mi-*) que no son de EC2. Esta tarea solo es necesaria si va a aplicar revisiones a instancias que no son de EC2 en un entorno híbrido y multinube.

> **Note**
>
> No puede agregar etiquetas para los nodos administrados que no son de EC2 mediante la consola de Amazon EC2.

Para agregar nodos administrados que no son de EC2 a un grupo de revisiones (consola de Systems Manager)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. En la lista de Nodos administrados, elija el nombre del nodo administrado que desea configurar para aplicar revisiones.

   > **Note**
   >
   > Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

4. Seleccione la pestaña Etiquetas y, luego, elija Editar.
5. En la columna izquierda, ingrese **`Patch Group`** o **`PatchGroup`**. Si ha permitido las etiquetas en los metadatos de las instancias de EC2, debe usar `PatchGroup` (sin espacio).
6. En la columna de la derecha, ingrese un valor de etiqueta que lo identifique como nombre para este grupo de revisiones.
7. Seleccione Save.
8. Repita este procedimiento para agregar otros nodos administrados en el mismo grupo de revisiones.

Tarea 3: añadir un grupo de revisiones a una línea de base de revisiones

Para asociar una base de referencia de revisiones específica a los nodos administrados, tiene que agregar el valor del grupo de revisiones a la base de referencia de revisiones. Al registrar el grupo de revisiones con una línea de base de revisiones, se asegura de que se instalen las revisiones correctos durante la operación de aplicación de revisiones. Esta tarea es necesaria si va a aplicar revisiones a instancias de EC2, a nodos administrados que no son de EC2 o a ambos.

Para obtener más información acerca de los grupos de revisiones, consulte Grupos de revisiones.

> **Note**
>
> Los pasos que siga dependerán de si accedió por primera vez a Patch Manager antes o después del lanzamiento de las políticas de revisiones el 22 de diciembre de 2022.

Para añadir un grupo de revisiones a una línea de base de revisiones (consola de Systems Manager)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Patch Manager.
3. Si accede a Patch Manager por primera vez en la Región de AWS actual y se abre la página de inicio de Patch Manager, elija Comenzar con una descripción general.
4. Elija la pestaña Líneas de base de revisiones y, luego, en la lista Líneas de base de revisiones, elija la línea de base de revisiones que desea configurar para su grupo de revisiones.

   Si no accedió por primera vez a Patch Manager hasta después de la publicación de las políticas de revisiones, debe elegir una línea de base personalizada que haya creado.
5. Si la página de detalles del ID de línea base incluye un menú Acciones, haga lo siguiente:

   - Elija Actions (Acciones) y luego Modify patch groups (Modificar grupos de revisiones).
   - Ingrese el valor de la etiqueta que ha agregado a los nodos administrados en Tarea 2: agregar nodos administrados a un grupo de revisiones mediante etiquetas y, a continuación, elija Agregar.

   Si la página de detalles del ID de línea de base no incluye un menú Acciones, los grupos de revisiones no se pueden configurar en la consola. En cambio, puede seguir uno de los procedimientos a continuación:

- (Recomendado) Configure una política de revisiones en Quick Setup, una herramienta de AWS Systems Manager, para asignar una línea de base de revisiones a una o más instancias de EC2.

  Para obtener más información, consulte Uso de políticas de revisiones de Quick Setup y Automatizar la implementación de revisiones en toda la organización mediante una política de revisiones de Quick Setup.
- Utilice el comando register-patch-baseline-for-patch-group en AWS Command Line Interface (AWS CLI) para configurar un grupo de revisiones.

Integración de Patch Manager con AWS Security Hub

AWS Security Hub proporciona una visión completa de su estado de seguridad en AWS. Security Hub recopila datos de seguridad de todas las Cuentas de AWS, los Servicios de AWS y los productos de socios terceros compatibles. Security Hub le permite comprobar su entorno con los estándares y las prácticas recomendadas del sector de la seguridad. Security Hub lo ayuda a analizar sus tendencias de seguridad y a identificar los problemas de seguridad de mayor prioridad.

Gracias a la integración entre Patch Manager, una herramienta de AWS Systems Manager, y Security Hub, puede enviar los resultados sobre los nodos no conformes de Patch Manager a Security Hub. Un resultado consiste en el registro observable de una comprobación de seguridad o de una detección relacionada con la seguridad. Después, Security Hub puede incluir esos hallazgos relacionados con revisiones en su análisis sobre la posición de seguridad.

La información de los siguientes temas se aplica independientemente del método o el tipo de configuración que utilice para las operaciones de aplicación de revisiones:

- Una política de revisiones configurada en Quick Setup
- Una opción de administración de host configurada en Quick Setup
- Una ventana de mantenimiento para ejecutar una revisión `Scan` o una tarea `Install`
- Una operación Patch Now (Aplicar revisión ahora) bajo demanda

Contenido

- Cómo Patch Manager envía los resultados a Security Hub
  - Tipos de resultados que envía Patch Manager
  - Latencia para el envío de resultados

- Reintento cuando Security Hub no está disponible
- Visualización de resultados de en Security Hub
- Resultado típico de Patch Manager
- Activación y configuración de la integración
- Cómo dejar de enviar resultados

Cómo Patch Manager envía los resultados a Security Hub

En Security Hub, los problemas de seguridad se rastrean como resultados. Algunos resultados provienen de problemas detectados por otros Servicios de AWS o por socios terceros. Security Hub también cuenta con un conjunto de reglas que utiliza para detectar problemas de seguridad y generar resultados.

Patch Manager es una de las herramientas de Systems Manager que envía los resultados a Security Hub. Después de llevar a cabo una operación de aplicación de revisiones mediante la ejecución de un documento de SSM (`AWS-RunPatchBaseline`, `AWS-RunPatchBaselineAssociation` o `AWS-RunPatchBaselineWithHooks`), la información de revisión se envía a Inventario o Cumplimiento, herramientas de AWS Systems Manager, o ambas. Después de que Inventory, Compliance o ambos hayan recibido los datos, Patch Manager recibe una notificación. A continuación, Patch Manager evalúa los datos para comprobar la precisión, el formato y la conformidad. Si se cumplen todas las condiciones, Patch Manager reenvía los datos a Security Hub.

Security Hub proporciona herramientas para administrar los resultados de todas estas fuentes. Puede ver y filtrar listas de resultados y ver los detalles de una búsqueda. Para obtener más información, consulte Visualización de resultados en la Guía del usuario de AWS Security Hub. También puede realizar un seguimiento del estado de una investigación de un resultado. Para obtener más información, consulte Adopción de medidas en función de los resultados en la Guía del usuario de AWS Security Hub.

Todos los resultados en Security Hub usan un formato JSON estándar denominado AWS Security Finding Format (ASFF). El ASFF incluye detalles sobre el origen del problema, los recursos afectados y el estado actual del resultado. Para obtener más información, consulte AWS Security Finding Format (ASFF) en la Guía del usuario de AWS Security Hub.

Tipos de resultados que envía Patch Manager

Patch Manager envía los resultados a Security Hub mediante AWS Security Finding Format (ASFF). En ASFF, el campo `Types` proporciona el tipo de resultado. Los resultados de Patch Manager tienen el siguiente valor para `Types`:

- Verificaciones de software y configuración/Administración de revisiones

Patch Manager envía una búsqueda por nodo administrado no conforme. El resultado se notifica con el tipo de recurso de `AwsEc2Instance` para que los resultados puedan relacionarse con otras integraciones de Security Hub que notifican tipos de recursos de `AwsEc2Instance`. Patch Manager solo envía un resultado a Security Hub si la operación detectó que el nodo administrado no era conforme. El resultado incluye los datos del resumen de revisiones.

> **Note**
>
> Tras informar de un nodo no conforme a Security Hub, Patch Manager no envía una actualización a Security Hub cuando el nodo cumple con las normas. Puede resolver manualmente los resultados en Security Hub una vez que se hayan aplicado las revisiones necesarias al nodo administrado.

Para obtener más información acerca de las definiciones de conformidad, consulte Valores de estado de conformidad de las revisiones. Para obtener más información acerca de `PatchSummary`, consulte PatchSummary en la Referencia de la API de AWS Security Hub.

Latencia para el envío de resultados

Cuando Patch Manager crea un nuevo resultado, por lo general, se envía a Security Hub en un plazo de entre unos pocos segundos y 2 horas. La velocidad varía en función del tráfico de la Región de AWS que se esté procesando en ese momento.

Reintento cuando Security Hub no está disponible

Si hay una interrupción del servicio, se ejecuta una función de AWS Lambda para devolver los mensajes a la cola principal una vez que el servicio vuelve a funcionar. Una vez que los mensajes se encuentran en la cola principal, la acción de reintentar es automática.

Si Security Hub no está disponible, Patch Manager reintenta enviar los resultados hasta que se reciban.

Visualización de resultados de en Security Hub

Este procedimiento describe cómo ver en Security Hub los resultados sobre los nodos administrados de su flota no conformes con las revisiones.

Revision de los resultados de Security Hub con el objetivo de comprobar el cumplimiento de las revisiones

1. Inicie sesión en la AWS Management Console y abra la consola de AWS Security Hub en https://console.aws.amazon.com/securityhub/.
2. En el panel de navegación, seleccione Resultados.
3. Seleccione la casilla Agregar filtros ().
4. En el menú, en Filtros, seleccione Nombre del producto.
5. En el cuadro de diálogo que se abre, elija es en el primer campo y, a continuación, introduzca **`Systems Manager Patch Manager`** en el segundo campo.
6. Seleccione Aplicar.
7. Agregue los filtros adicionales que desee para reducir los resultados.
8. En la lista de resultados, elija el título de un resultado sobre el que desee obtener más información.

   Se abre un panel en la parte derecha de la pantalla con más detalles sobre el recurso, el problema descubierto y una solución recomendada.

   > **Important**
   >
   > En este momento, Security Hub informa que el tipo de recurso de todos los nodos administrados es `EC2 Instance`. Esto incluye servidores en las instalaciones y las máquinas virtuales (VM) que haya registrado para utilizarlas con Systems Manager.

Clasificaciones de gravedad

La lista de resultados para **`Systems Manager Patch Manager`**, incluye un informe sobre la gravedad del resultado. Los niveles de gravedad incluyen los siguientes, de el más bajo al más alto:

- INFORMATIVO: no se encontró ningún problema.

- BAJO: el problema no requiere solución.
- MEDIO: el problema debe abordarse, pero no es urgente.
- ALTO: el problema debe abordarse con prioridad.
- CRÍTICO: el problema debe solucionarse de inmediato para evitar una escalada.

La gravedad se determina según el paquete de incumplimiento más severo de una instancia. Como puede tener varios líneas de base de revisiones con varios niveles de gravedad, se indica el nivel de severidad más alto de todos los paquetes no conformes. Por ejemplo, supongamos que tiene dos paquetes no conformes en los que la gravedad del paquete A es “crítica” y la del paquete B es “baja”. La gravedad se indicará como “crítica”.

Tenga en cuenta que el campo de gravedad se correlaciona directamente con el campo Patch Manager `Compliance`. Se trata de un campo que puede configurar para asignar a las revisiones individuales que coincidan con la regla. Como este campo `Compliance` está asignado a revisiones individuales, no se refleja en el nivel de resumen de revisiones.

Contenido relacionado

- Resultados en la Guía del usuario de AWS Security Hub
- Cumplimiento de las revisiones multicuenta con Patch Manager y Security Hub en el blog de administración y gobernanza de AWS

Resultado típico de Patch Manager

Patch Manager envía los resultados a Security Hub mediante AWS Security Finding Format (ASFF).

Aquí hay un ejemplo de un hallazgo típico 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/fleet-manager/
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"
  }
}
```

Activación y configuración de la integración

Para utilizar la integración de Patch Manager a Security Hub, debe activar Security Hub. Para obtener información acerca de cómo activar Security Hub, consulte la Configuración de Security Hub en la Guía del usuario de AWS Security Hub.

En el siguiente procedimiento, se describe cómo integrar Patch Manager y Security Hub cuando Security Hub ya está activo pero la integración de Patch Manager se encuentra desactivada. Solo es necesario completar este procedimiento si la integración se desactivó de forma manual.

Para agregar Patch Manager a la integración de Security Hub

1. En el panel de navegación, elija Patch Manager.
2. Elija la pestaña Settings.

   -o bien-

   Si va a acceder a Patch Manager por primera vez en la Región de AWS actual, seleccione Comience por la información general, y luego la pestaña Configuración.
3. En la sección Export to Security Hub (Exportar a Security Hub), situada a la derecha de Patch compliance findings aren't being exported to Security Hub (Los resultados de conformidad de revisiones no se exportan a Security Hub), elija Enable (Habilitar).

Cómo dejar de enviar resultados

Para dejar de enviar resultados a Security Hub, puede usar la consola de Security Hub o la API.

Para obtener más información, consulte los siguientes temas en la Guía del usuario de AWS Security Hub:

- Disabling and enabling the flow of findings from an integration (console) (Desactivación y habilitación del flujo de resultados desde una integración [consola])
- Desactivación del flujo de hallazgos desde una integración (API de Security Hub, AWS CLI)

Trabajar con recursos Patch Manager mediante la AWS CLI

En esta sección se incluyen ejemplos de comandos de la AWS Command Line Interface (AWS CLI) que puede utilizar para llevar a cabo tareas de configuración de Patch Manager, una herramienta de AWS Systems Manager.

Para obtener un ejemplo de cómo utilizar la AWS CLI para aplicar parches a un entorno de servidor mediante una base de referencia de parches personalizada, consulte Tutorial: implementación de revisiones en un entorno de servidores mediante la AWS CLI.

Para obtener más información acerca del uso de las tareas de la AWS CLI para AWS Systems Manager, consulte la sección de AWS Systems Manager de la Referencia de comandos de la AWS CLI.

Temas

- Comandos de la AWS CLI para las bases de referencia de parches
- Comandos de la AWS CLI para grupos de parches
- Comandos de la AWS CLI para ver resúmenes y detalles de parches
- Comandos de la AWS CLI para escanear y aplicar revisiones a nodos administrados

Comandos de la AWS CLI para las bases de referencia de parches

Ejemplos de comandos para las bases de referencia de parches

- Creación de una base de referencia de parches
- Creación de una base de referencia de parches con repositorios personalizados para distintas versiones del sistema operativo
- Actualización de una base de referencia de parches
- Cambio del nombre de una base de referencia de parches
- Eliminación de una base de referencia de parches
- Enumeración de todas las bases de referencia de parches
- Enumeración de todas las bases de referencia de parches proporcionadas por AWS

- Enumeración de mis bases de referencia de parches
- Visualización de una base de referencia de parches
- Obtención de la base de referencia de parches predeterminada
- Establecer una base de referencia de parches personalizada como predeterminada
- Restablecimiento de una base de referencia de parches de AWS como la predeterminada
- Etiquetado de una base de referencia de parches
- Enumeración de las etiquetas de una base de referencia de parches
- Eliminación de una etiqueta de una base de referencia de parches

Creación de una base de referencia de parches

El siguiente comando crea una línea de base de revisiones que aprueba todas las actualizaciones de seguridad críticas e importantes para Windows Server 2012 R2 5 días después de su lanzamiento. También se han especificado parches para las listas de parches aprobados y rechazados. Además, la base de referencia de parches se ha etiquetado para indicar que es para un entorno de producción.

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}]"
```

El sistema devuelve información similar a la siguiente.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Creación de una base de referencia de parches con repositorios personalizados para distintas versiones del sistema operativo

Solo se aplica a nodos administrados de Linux. El siguiente comando muestra cómo especificar el repositorio de parches que se debe utilizar para una versión determinada del sistema operativo de Amazon Linux. En este ejemplo se utiliza un repositorio de origen habilitado de forma predeterminada en Amazon Linux 2017.09, pero puede adaptarlo para otro diferente que haya configurado para un nodo administrado.

> **Note**
>
> Para ilustrar mejor este comando, que tiene mayor complejidad, utilizamos la opción `--cli-input-json` junto con otras opciones almacenadas en un archivo JSON externo.

1. Cree un archivo JSON con un nombre similar a `my-patch-repository.json` y agregue el siguiente contenido:

```
{
    "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. En el directorio en el que almacenó el archivo, ejecute el siguiente comando:

```
aws ssm create-patch-baseline --cli-input-json file://my-patch-repository.json
```

El sistema devuelve información similar a la siguiente.

```
{
    "BaselineId": "pb-0c10e65780EXAMPLE"
}
```

Actualización de una base de referencia de parches

El siguiente comando agrega dos parches como rechazados y uno como aprobado a una base de referencia de parches existente.

Para obtener información acerca de los formatos aceptados para las listas de parches aprobados y rechazados, consulte Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas.

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"
```

El sistema devuelve información similar a la siguiente.

```
{
   "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"
```

```
}
```

Cambio del nombre de una base de referencia de parches

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"
```

El sistema devuelve información similar a la siguiente.

```
{
   "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"
}
```

Eliminación de una base de referencia de parches

```
aws ssm delete-patch-baseline --baseline-id "pb-0c10e65780EXAMPLE"
```

El sistema devuelve información similar a la siguiente.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Enumeración de todas las bases de referencia de parches

```
aws ssm describe-patch-baselines
```

El sistema devuelve información similar a la siguiente.

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

A continuación, se muestra otro comando que enumera todas las bases de referencia de parches de una Región de 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]"
```

El sistema devuelve información similar a la siguiente.

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

Enumeración de todas las bases de referencia de parches proporcionadas por 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]"
```

El sistema devuelve información similar a la siguiente.

```
{
   "BaselineIdentities":[
      {
         "BaselineName":"AWS-DefaultPatchBaseline",
         "DefaultBaseline":true,
         "BaselineDescription":"Default Patch Baseline Provided by AWS.",
         "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/
pb-0c10e65780EXAMPLE"
      }
   ]
```

```
}
```

Enumeración de mis bases de referencia de parches

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]"
```

El sistema devuelve información similar a la siguiente.

```
{
   "BaselineIdentities":[
      {
         "BaselineName":"Windows-Server-2012R2",
         "DefaultBaseline":false,
         "BaselineDescription":"Windows Server 2012 R2, Important and Critical security
 updates",
         "BaselineId":"pb-0c10e65780EXAMPLE"
      }
   ]
}
```

Visualización de una base de referencia de parches

```
aws ssm get-patch-baseline --baseline-id pb-0c10e65780EXAMPLE
```

> **Note**
>
> Para bases de referencia de parches personalizadas, puede especificar el ID de la base de referencia de parches o el nombre de recurso de Amazon (ARN) completo. Para bases de referencia de parches proporcionadas por AWS, debe especificar el ARN

completo. Por ejemplo, `arn:aws:ssm:us-east-2:075727635805:patchbaseline/pb-0c10e65780EXAMPLE`.

El sistema devuelve información similar a la siguiente.

```
{
   "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"
}
```

Obtención de la base de referencia de parches predeterminada

```
aws ssm get-default-patch-baseline --region us-east-2
```

El sistema devuelve información similar a la siguiente.

```
{
   "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/pb-0c10e65780EXAMPLE"
}
```

Establecer una base de referencia de parches personalizada como predeterminada

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"
```

El sistema devuelve información similar a la siguiente.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Restablecimiento de una base de referencia de parches de AWS como la predeterminada

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"
```

El sistema devuelve información similar a la siguiente.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Etiquetado de una base de referencia de parches

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"
```

Enumeración de las etiquetas de una base de referencia de parches

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"
```

Eliminación de una etiqueta de una base de referencia de parches

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"
```

Comandos de la AWS CLI para grupos de parches

Ejemplos de comandos para grupos de parches

- Crear un grupo de parches
- Registro del grupo de parches "Web Servers" con una base de referencia de parches

- Registro del grupo de parches “Backend” con la base de referencia de parches proporcionada por AWS
- Visualización de los registros de grupos de parches
- Anulación del registro de un grupo de parches en una base de referencia de parches

Crear un grupo de parches

> **Note**
>
> Los grupos de revisiones no se usan en operaciones de aplicación de revisiones basadas en políticas de revisiones. Para obtener información sobre el uso de las políticas de revisiones, consulte Configuraciones de políticas de revisiones en Quick Setup.

Para ayudarlo a organizar las acciones de aplicación de revisiones, le recomendamos que utilice las etiquetas para agregar nodos administrados a grupos de revisiones. Los grupos de revisiones requieren el uso de la clave de etiqueta `Patch Group` o `PatchGroup`. Si ha permitido las etiquetas en los metadatos de las instancias de EC2, debe usar `PatchGroup` (sin espacio). Puede especificar cualquier valor de etiqueta, pero la clave de etiqueta debe ser `Patch Group` o `PatchGroup`. Para obtener más información acerca de los grupos de revisiones, consulte Grupos de revisiones.

Después de agrupar los nodos administrados mediante etiquetas, tiene que agregar el valor de grupo de revisiones a una base de referencia de revisiones. Al registrar el grupo de revisiones en una línea de base de revisiones, se asegura de que se instalen las revisiones correctos durante la operación de aplicación de revisiones.

Tarea 1: agregar instancias de EC2 a un grupo de revisiones mediante etiquetas

> **Note**
>
> Cuando se utiliza la consola de Amazon Elastic Compute Cloud (Amazon EC2) y la AWS CLI, es posible aplicar etiquetas `Key = Patch Group` o `Key = PatchGroup` a instancias que aún no están configuradas para utilizarlas con Systems Manager. Si una instancia de EC2 que espera ver en Patch Manager no aparece en la lista luego de aplicar la etiqueta `Patch Group` o `Key = PatchGroup`, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

Ejecute el siguiente comando para añadir la etiqueta `PatchGroup` a una instancia de EC2.

```
aws ec2 create-tags --resources "i-1234567890abcdef0" --tags
 "Key=PatchGroup,Value=GroupValue"
```

Tarea 2: agregar nodos administrados a un grupo de revisiones mediante etiquetas

Ejecute el siguiente comando para agregar la etiqueta `PatchGroup` a un nodo administrado.

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"
```

Tarea 3: añadir un grupo de revisiones a una línea de base de revisiones

Ejecute el siguiente comando para asociar un valor de etiqueta `PatchGroup` a la base de referencia de parches especificada.

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"
```

El sistema devuelve información similar a la siguiente.

```
{
  "PatchGroup": "Development",
  "BaselineId": "pb-0c10e65780EXAMPLE"
}
```

Registro del grupo de parches "Web Servers" con una base de referencia de parches

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"
```

El sistema devuelve información similar a la siguiente.

```
{
   "PatchGroup":"Web Servers",
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Registro del grupo de parches “Backend” con la base de referencia de parches proporcionada por 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"
```

El sistema devuelve información similar a la siguiente.

```
{
   "PatchGroup":"Backend",
   "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/pb-0c10e65780EXAMPLE"
}
```

Visualización de los registros de grupos de parches

```
aws ssm describe-patch-groups --region us-east-2
```

El sistema devuelve información similar a la siguiente.

```
{
   "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"
```

```
        }
      }
   ]
}
```

Anulación del registro de un grupo de parches en una base de referencia de parches

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"
```

El sistema devuelve información similar a la siguiente.

```
{
   "PatchGroup":"Production",
   "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/pb-0c10e65780EXAMPLE"
}
```

Comandos de la AWS CLI para ver resúmenes y detalles de parches

Ejemplos de comandos para ver resúmenes y detalles de parches

- Obtención de todos los parches definidos por una base de referencia de parches
- Obtención de todos los parches para AmazonLinux2018.03 que tienen una clasificación SECURITY y gravedad Critical
- Obtención de todos los parches para Windows Server 2012 que tienen una gravedad MSRC Critical

- Obtención de todos los parches disponibles
- Obtención de estados del resumen de revisiones por nodo administrado
- Obtención de detalles de conformidad de revisiones de un nodo administrado
- Vista de los resultados de conformidad de parches (AWS CLI)

Obtención de todos los parches definidos por una base de referencia de parches

> **Note**
>
> Este comando solo se admite para bases de referencia de parches 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"
```

El sistema devuelve información similar a la siguiente.


```
{
   "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---
```

Obtención de todos los parches para AmazonLinux2018.03 que tienen una clasificación **`SECURITY`** y gravedad **`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
```

El sistema devuelve información similar a la siguiente.

```
{
    "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---
```

Obtención de todos los parches para Windows Server 2012 que tienen una gravedad MSRC **`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
```

El sistema devuelve información similar a la siguiente.

```
{
   "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---
```

Obtención de todos los parches disponibles

```
aws ssm describe-available-patches --region us-east-2
```

El sistema devuelve información similar a la siguiente.

```
{
   "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---
```

Obtención de estados del resumen de revisiones por nodo administrado

El resumen por nodo administrado aporta el número de revisiones en los siguientes estados por nodo: “NotApplicable”, “Missing”, “Failed”, “InstalledOther” e “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
```

El sistema devuelve información similar a la siguiente.

```
{
   "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---
```

Obtención de detalles de conformidad de revisiones de un nodo administrado

```
aws ssm describe-instance-patches --instance-id i-08ee91c0b17045407
```

El sistema devuelve información similar a la siguiente.

```
{
   "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---
```

Vista de los resultados de conformidad de parches (AWS CLI)

Para ver los resultados de conformidad de revisiones de un único nodo administrado

Ejecute el siguiente comando en la AWS Command Line Interface (AWS CLI) para ver los resultados de conformidad de revisiones de un único nodo administrado.

```
aws ssm describe-instance-patch-states --instance-id instance-id
```

Sustituya *instance-id* por el ID del nodo administrado del que desea ver los resultados, con el formato `i-02573cafcfEXAMPLE` o `mi-0282f7c436EXAMPLE`.

El sistema devuelve información similar a la siguiente.

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

Para ver un resumen del recuento de revisiones para todas las instancias de EC2 de una región

El comando `describe-instance-patch-states` permite recuperar los resultados de una única instancia administrada a la vez. Sin embargo, si se utiliza un script personalizado con el comando `describe-instance-patch-states`, se puede generar un informe más pormenorizado.

Por ejemplo, si la herramienta de filtro jq está instalada en su equipo local, podría ejecutar el siguiente comando para identificar cuáles de sus instancias EC2 en una Región de AWS concreta presentan el estado de `InstalledPendingReboot`.

```
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 representa el identificador de una Región de AWS compatible con AWS Systems Manager, como `us-east-2` para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

Por ejemplo:

```
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}'
```

El sistema devuelve información similar a la siguiente.

```
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
```

Además de `InstalledPendingRebootCount`, la lista de tipos de recuento que puede buscar incluye lo siguiente:

- `CriticalNonCompliantCount`
- `SecurityNonCompliantCount`

- `OtherNonCompliantCount`
- `UnreportedNotApplicableCount`
- `InstalledPendingRebootCount`
- `FailedCount`
- `NotApplicableCount`
- `InstalledRejectedCount`
- `InstalledOtherCount`
- `MissingCount`
- `InstalledCount`

Comandos de la AWS CLI para escanear y aplicar revisiones a nodos administrados

Una vez ejecutados los siguientes comandos para analizar la conformidad de parches o instalar parches, puede utilizar los comandos de la sección Comandos de la AWS CLI para ver resúmenes y detalles de parches para ver la información sobre el estado y la conformidad de los parches.

Ejemplos de comandos

- Analizar nodos administrados para comprobar la conformidad de revisiones (AWS CLI)
- Instalación de revisiones en nodos administrados (AWS CLI)

Analizar nodos administrados para comprobar la conformidad de revisiones (AWS CLI)

Para analizar nodos administrados específicos para comprobar la conformidad de revisiones

Ejecute el siguiente comando.

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
```

El sistema devuelve información similar a la siguiente.

```
{
    "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---

    }
}
```

Para analizar los nodos administrados y comprobar la conformidad de revisiones por etiqueta de grupo de revisiones

Ejecute el siguiente comando.

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
```

El sistema devuelve información similar a la siguiente.

```
{
    "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---

    }
}
```

Instalación de revisiones en nodos administrados (AWS CLI)

Para instalar revisiones en nodos administrados específicos

Ejecute el siguiente comando.

> **Note**
>
> Los nodos administrados de destino se reinician según sea necesario para completar la instalación de la revisión. Para obtener más información, consulte Documento de comandos SSM para aplicar revisiones: `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
```

El sistema devuelve información similar a la siguiente.

```
{
```

```
    "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---

    }
}
```

Para instalar revisiones en nodos administrados en un grupo de revisiones específico

Ejecute el siguiente comando.

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
```

El sistema devuelve información similar a la siguiente.

```
{
    "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---

    }
}
```

Tutoriales de AWS Systems Manager Patch Manager

Los tutoriales de esta sección muestran cómo utilizar Patch Manager, una herramienta de AWS Systems Manager, en diversos casos de aplicación de revisiones.

Temas

- Tutorial: creación de una línea de base de revisiones para instalar Service Packs de Windows mediante la consola
- Tutorial: actualización de las dependencias de la aplicación, implementación de una revisión a un nodo administrado y realización de una comprobación de estado específica de la aplicación mediante la consola
- Tutorial: implementación de revisiones en un entorno de servidores mediante la AWS CLI

Tutorial: creación de una línea de base de revisiones para instalar Service Packs de Windows mediante la consola

Al crear una línea de base de revisiones personalizada, puede especificar que se instalen todos, parte o solo un tipo de parche compatible.

En las líneas de base de revisiones para Windows, puede seleccionar `ServicePacks` como única opción de clasificación para limitar las actualizaciones de revisiones solo a Service Packs. Patch Manager, una herramienta de AWS Systems Manager, puede instalar automáticamente Service Packs siempre que la actualización esté disponible en Windows Update o Windows Server Update Services (WSUS).

Puede configurar una línea de base de revisiones para controlar si se instalan los Service Packs para todas las versiones de Windows o solo los de versiones específicas, como Windows 7 o Windows Server 2016.

Utilice el procedimiento siguiente para crear una base de referencia de revisiones personalizada que se utilizará exclusivamente para instalar todos los Service Packs en los nodos administrados de Windows.

Para crear una línea de base de revisiones para instalar Service Packs de Windows (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Patch Manager.

3. Seleccione la pestaña Bases de referencia de parches, y luego Crear una base de referencia de parches.
4. En Nombre, escriba un nombre para la nueva línea de base de revisiones; por ejemplo, `MyWindowsServicePackPatchBaseline`.
5. (Opcional) En Description (Descripción), escriba una descripción para esta línea de base de revisiones.
6. En Operating system (Sistema operativo), elija `Windows`.
7. Si desea comenzar a utilizar esta línea de base de revisiones de forma predeterminada para Windows tan pronto como la cree, seleccione Set this patch baseline as the default patch baseline for Windows Server instances (Establecer esta línea de base de revisiones como la línea de base de revisiones predeterminada para las instancias de Windows Server).

   > **Note**
   >
   > Esta opción solo está disponible si accedió a Patch Manager por primera vez antes de las políticas de parches publicadas el 22 de diciembre de 2022.
   > Para obtener información sobre la configuración de una línea de base de revisiones existente como la opción predeterminada, consulte Configuración de una línea de base de revisiones existente como valor predeterminado.

8. En la sección Approval Rules for operating systems (Reglas de aprobación para sistemas operativos), use los campos para crear una o varias reglas de aprobación automática.

   - Productos: versiones de los sistemas operativos a las que se aplica la regla de aprobación; por ejemplo, `WindowsServer2012`. Puede elegir una, más de una o todas las versiones compatibles de Windows. La selección predeterminada es `All`.
   - Clasificación: elija `ServicePacks`.
   - Gravedad: el valor de gravedad de las revisiones a los que se aplica la regla. Para asegurarse de que todos los Service Packs están incluidos en la regla, elija `All`.
   - Auto-approval (Aprobación automática): método para seleccionar revisiones para su aprobación automática.
     - Approve patches after a specified number of days (Aprobar las revisiones después de una cantidad determinada de días): la cantidad de días que Patch Manager debe esperar después de que se lance o actualice una revisión y antes de que se apruebe automáticamente. Puede ingresar cualquier número entero entre cero (0) y 360. En la mayoría de los casos, se recomienda no esperar más de 100 días.

- Approve patches released up to a specific date (Aprobar las revisiones publicadas hasta una fecha específica): la fecha de lanzamiento de la revisión para la que Patch Manager aplica automáticamente todas las revisiones publicadas o actualizadas en esa fecha o con anterioridad a ella. Por ejemplo, si especifica el 7 de julio de 2023, no se instalarán automáticamente las revisiones publicadas o actualizadas a partir del 8 de julio de 2023.

- (Opcional). Informes de conformidad: el nivel de gravedad que desea asignar a los Service Packs aprobados por la base de referencia; como `High`.

> **Note**
>
> Si especifica un nivel de notificación de conformidad y se informa el estado de cualquier revisión aprobada como `Missing`, la gravedad de la conformidad general notificada por la línea de base de revisiones será el nivel de gravedad que especificó.

9. (Opcional) En Manage tags (Administrar etiquetas), aplique uno o varios pares de claves nombre/valor al a la línea de base de revisiones.

   Las etiquetas son metadatos opcionales que usted asigna a un recurso. Las etiquetas permiten clasificar los recursos de diversas maneras, por ejemplo, según la finalidad, el propietario o el entorno. Para esta línea de base de revisiones dedicada a la actualización de Service Packs, puede especificar pares clave-valor como los siguientes:

   - `Key=OS,Value=Windows`
   - `Key=Classification,Value=ServicePacks`

10. Elija Create patch baseline (Crear base de referencia de revisiones).

Tutorial: actualización de las dependencias de la aplicación, implementación de una revisión a un nodo administrado y realización de una comprobación de estado específica de la aplicación mediante la consola

En muchos casos, un nodo administrado debe reiniciarse una vez que se haya aplicado la revisión correspondiente a la última actualización de software. No obstante, el reinicio de un nodo en producción sin contar con las debidas medidas de protección puede ocasionar varios problemas, como la invocación de alarmas, el registro de datos de métrica incorrectos y la interrupción de las sincronizaciones de datos.

En este tutorial se demuestra cómo evitar este tipo de problemas a través del uso del documento de AWS Systems Manager (documento de SSM) `AWS-RunPatchBaselineWithHooks` para realizar una operación de implementación de una revisión compleja y de varios pasos que permita lograr lo siguiente:

1. impedir nuevas conexiones a la aplicación
2. instalar las actualizaciones del sistema operativo.
3. actualizar las dependencias del paquete de la aplicación
4. reiniciar el sistema
5. realizar una comprobación de estado específica de la aplicación

Para este ejemplo, se ha configurado la infraestructura de la siguiente manera:

- Las máquinas virtuales de destino se registran como nodos administrados con Systems Manager.
- `Iptables` se utiliza como un firewall local.
- La aplicación alojada en los nodos administrados se ejecuta en el puerto 443.
- La aplicación alojada en los nodos administrados es una aplicación de `nodeJS`.
- El administrador de procesos pm2 es el encargado de administrar la aplicación alojada en los nodos administrados.
- La aplicación ya cuenta con un punto de enlace de comprobación de estado especificado.
- El punto de enlace de comprobación de estado de la aplicación no requiere autenticación del usuario final. El punto de enlace permite una comprobación de estado conforme a los requisitos de la organización a la hora de establecer la disponibilidad. (En sus entornos, es posible que sea suficiente con solo comprobar que la aplicación de `nodeJS` se está ejecutando y es capaz de atender las solicitudes. No obstante, en otros casos, es posible que se desee verificar también que se ha establecido una conexión con el nivel de almacenamiento en caché o el nivel de base de datos).

Los ejemplos expuestos en este tutorial son únicamente para fines ilustrativos y no se pretende su implementación como tales en entornos de producción. Asimismo, tenga en cuenta que la característica de los enlaces de ciclo de vida de Patch Manager, una herramienta de Systems Manager, con el documento `AWS-RunPatchBaselineWithHooks` puede admitir muchos otros casos. A continuación, se presentan varios ejemplos.

- Detenga un agente de informes de métricas antes de aplicar una revisión y reiniciarlo después de que el nodo administrado se reinicie.
- Desconecte el nodo administrado de un clúster de CRM o PCS antes de aplicar las revisiones y vuelva a adjuntarlo después de reiniciar el nodo.
- Actualice el software de terceros (por ejemplo, Java, Tomcat, aplicaciones de Adobe, etc.) en las máquinas de Windows Server luego de aplicar las actualizaciones del sistema operativo (SO), pero antes de que el nodo administrado se reinicie.

Para actualizar las dependencias de la aplicación, aplicar una revisión a un nodo administrado y realizar una comprobación de estado específica de la aplicación

1. Cree un documento de SSM para su script de preinstalación con el siguiente contenido y asígnele el nombre `NodeJSAppPrePatch`. Sustituya *`your_application`* por el nombre de la aplicación.

   Este script bloquea inmediatamente las nuevas solicitudes entrantes y proporciona cinco segundos para que las ya activas se completen antes de comenzar la operación de aplicación de parches. Para la opción `sleep`, especifique una cantidad de segundos mayor que la que suele tardar en completarse las solicitudes entrantes.

   ```
   # 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
   ```

   Para obtener más información acerca de la creación de documentos de SSM, consulte Crear contenido en el documento de SSM.

2. Cree otro documento SSM con el siguiente contenido para su script de posinstalación para actualizar las dependencias de su aplicación y asígnele el nombre `NodeJSAppPostPatch`. Sustituya */your/application/path* por la ruta de acceso a su aplicación.

   ```
   cd /your/application/path
   npm update
   ```

```
# you can use npm-check-updates if you want to upgrade major versions
```

3. Cree otro documento de SSM con el siguiente contenido para que su script de `onExit` recupere su aplicación y realice una comprobación de estado. Asigne un nombre a este documento de SSM `NodeJSAppOnExitPatch`. Sustituya *`your_application`* por el nombre de la aplicación.

```
# 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. Cree una asociación en State Manager, una herramienta de AWS Systems Manager para ejecutar la operación mediante los siguientes pasos:

   1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
   2. En el panel de navegación, elija State Manager y, a continuación, elija Create association (Crear asociación).
   3. En Name (Nombre) proporcione un nombre que ayude a identificar la finalidad de la asociación.
   4. En la lista Document (Documento), elija `AWS-RunPatchBaselineWithHooks`.
   5. En Operation (Operación), elija Install (Instalar).
   6. (Opcional) En Snapshot Id, (ID de instantánea), proporcione un valor GUID que genere de modo que le permita agilizar la operación y garantizar la consistencia. El valor GUID puede ser tan sencillo como `00000000-0000-0000-0000-111122223333`.
   7. En Pre Install Hook Doc Name (Nombre del documento del enlace de preinstalación), ingrese `NodeJSAppPrePatch`.
   8. En Post Install Hook Doc Name (Nombre del documento del enlace de posinstalación), ingrese `NodeJSAppPostPatch`.

9. En On ExitHook Doc Name (Nombre del documento del enlace de salida), ingrese `NodeJSApp0nExitPatch`.

5. En el caso de Targets (Destinos), especifique etiquetas, elija nodos manualmente, elija un grupo de recursos o elija todos los nodos administrados para identificar sus nodos administrados.
6. En Specify schedule (Especificar programación), especifique con qué frecuencia se ejecutará la asociación. En el caso de la aplicación de revisiones en el nodo administrado, la cadencia habitual suele ser una vez a la semana.
7. En la sección Rate control (Control de velocidad), elija las opciones para controlar cómo se ejecuta la asociación en varios nodos administrados. Asegúrese de que solo se actualiza una parte de los nodos administrados a la vez. De lo contrario, toda su flota o la mayor parte de ella podría quedar sin conexión a la vez. Para obtener más información sobre el uso de controles de velocidad, consulte Cómo comprender los controles de frecuencia y destinos en las asociaciones de State Manager.
8. (Opcional) En Output options (Opciones de salida), para guardar la salida del comando en un archivo, seleccione el cuadro Enable writing output to S3 (Permitir la escritura de salida en S3). Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancias asignado al nodo administrado, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, verifique que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

9. Elija Crear asociación.

Tutorial: implementación de revisiones en un entorno de servidores mediante la AWS CLI

En el siguiente tutorial se describe cómo aplicar parches a un entorno de servidor mediante una base de referencia de parches predeterminada, grupos de parches y un periodo de mantenimiento.

Antes de empezar

- Instalar o actualizar SSM Agent en los nodos administrados. Para aplicar revisiones a los nodos administrados de Linux, los nodos deben ejecutar la versión de SSM Agent 2.0.834.0 o posterior. Para obtener más información, consulte Actualización de SSM Agent mediante Run Command.
- Configure roles y permisos para Maintenance Windows, una herramienta de AWS Systems Manager. Para obtener más información, consulte Configuración de Maintenance Windows.
- Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

  Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

Para configurar Patch Manager y aplicar revisiones a los nodos administrados (línea de comandos)

1. Ejecute el siguiente comando para crear una base de referencia de revisiones para Windows denominada `Production-Baseline`. Esta línea de base de revisiones aprueba las revisiones para un entorno de producción siete días después de su lanzamiento. Es decir, se ha etiquetado la base de referencia de parches para indicar que es para un entorno de producción.

   > **Note**
   >
   > El parámetro `OperatingSystem` y `PatchFilters` varían en función del sistema operativo de los nodos administrados de destino a los que se aplica la base de referencia de revisiones. Para obtener más información, consulte OperatingSystem y 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"
```

El sistema devuelve información similar a la siguiente.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

2. Ejecute los siguientes comandos para registrar la base de referencia de parches "Production-Baseline" para dos grupos de parches. Los grupos se denominan "Database Servers" (Servidores de base de datos) y "Front-End Servers" (Servidores 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"
   ```

   El sistema devuelve información similar a la siguiente.

```
{
   "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"
```

El sistema devuelve información similar a la siguiente.

```
{
   "PatchGroup":"Front-End Servers",
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

3. Ejecute los siguientes comandos para crear dos períodos de mantenimiento para los servidores de producción. El primer periodo se ejecuta cada martes a las 22:00 h. El segundo período se ejecuta cada sábado a las 22:00. Además, el periodo de mantenimiento está etiquetado para indicar que es para un entorno de producción.

   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
```

El sistema devuelve información similar a la siguiente.

```
{
   "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
```

El sistema devuelve información similar a la siguiente.

```
{
   "WindowId":"mw-9a8b7c6d5eEXAMPLE"
}
```

4. Ejecute los siguientes comandos para registrar los grupos de parches de servidores `Database` y `Front-End` con sus respectivos periodos de mantenimiento.

   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"
   ```

   El sistema devuelve información similar a la siguiente.

   ```
   {
      "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"
```

El sistema devuelve información similar a la siguiente.

```
{
   "WindowTargetId":"faa01c41-1d57-496c-ba77-ff9caEXAMPLE"
}
```

5. Ejecute los siguientes comandos para registrar una tarea de aplicación de parches que instale las actualizaciones que faltan en los servidores `Database` y `Front-End` durante sus respectivos periodos de mantenimiento.

   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}}"
```

El sistema devuelve información similar a la siguiente.

```
{
   "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}}"
```

El sistema devuelve información similar a la siguiente.

```
{
   "WindowTaskId":"8a5c4629-31b0-4edd-8aea-33698EXAMPLE"
}
```

6. Ejecute el siguiente comando para obtener el resumen de conformidad de parches de alto nivel de un grupo de parches. El resumen de conformidad de revisiones de alto nivel incluye el número de nodos administrados con revisiones en los respectivos estados de revisiones.

> **Note**
>
> Lo normal es que vea ceros para el número de nodos administrados en el resumen hasta que se ejecute la tarea de revisiones durante el primer periodo de mantenimiento.

Linux & macOS

```
aws ssm describe-patch-group-state \
    --patch-group "Database Servers"
```

Windows Server

```
aws ssm describe-patch-group-state ^
    --patch-group "Database Servers"
```

El sistema devuelve información similar a la siguiente.

```
{
   "Instances": number,
   "InstancesWithFailedPatches": number,
   "InstancesWithInstalledOtherPatches": number,
   "InstancesWithInstalledPatches": number,
   "InstancesWithInstalledPendingRebootPatches": number,
   "InstancesWithInstalledRejectedPatches": number,
   "InstancesWithMissingPatches": number,
   "InstancesWithNotApplicablePatches": number,
   "InstancesWithUnreportedNotApplicablePatches": number
```

```
}
```

7. Ejecute el siguiente comando para obtener los estados del resumen de revisiones por nodo administrado para un grupo de revisiones. El resumen por nodo administrado incluye un número de revisiones en los respectivos estados de revisiones por nodo administrado para un grupo de revisiones.

   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"
   ```

   El sistema devuelve información similar a la siguiente.

   ```
   {
      "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
   ```

```
        }
    ]
}
```

Para ver un ejemplo de otros comandos de la AWS CLI que podría utilizar para sus tareas de configuración de Patch Manager, consulte Trabajar con recursos Patch Manager mediante la AWS CLI.

Solución de problemas de Patch Manager

Utilice la siguiente información como ayuda para solucionar problemas con Patch Manager, una herramienta de AWS Systems Manager.

Temas

- Problema: error “Invoke-PatchBaselineOperation : Access Denied” o error “Unable to download file from S3” para baseline_overrides.json
- Problema: la implementación de revisiones falla sin una causa aparente y sin arrojar un mensaje de error
- Problema: resultados de conformidad de revisiones inesperados
- Errores al ejecutar AWS-RunPatchBaseline en Linux
- Errores al ejecutar AWS-RunPatchBaseline en Windows Server
- Cómo utilizar manuales de procedimientos de Automatización de AWS Support
- Cómo ponerse en contacto con AWS Support

Problema: error “Invoke-PatchBaselineOperation : Access Denied” o error “Unable to download file from S3” para **`baseline_overrides.json`**

Problema: cuando se ejecutan las operaciones de implementación de revisiones especificadas en su política de revisiones, recibe un error similar al siguiente ejemplo.

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
```

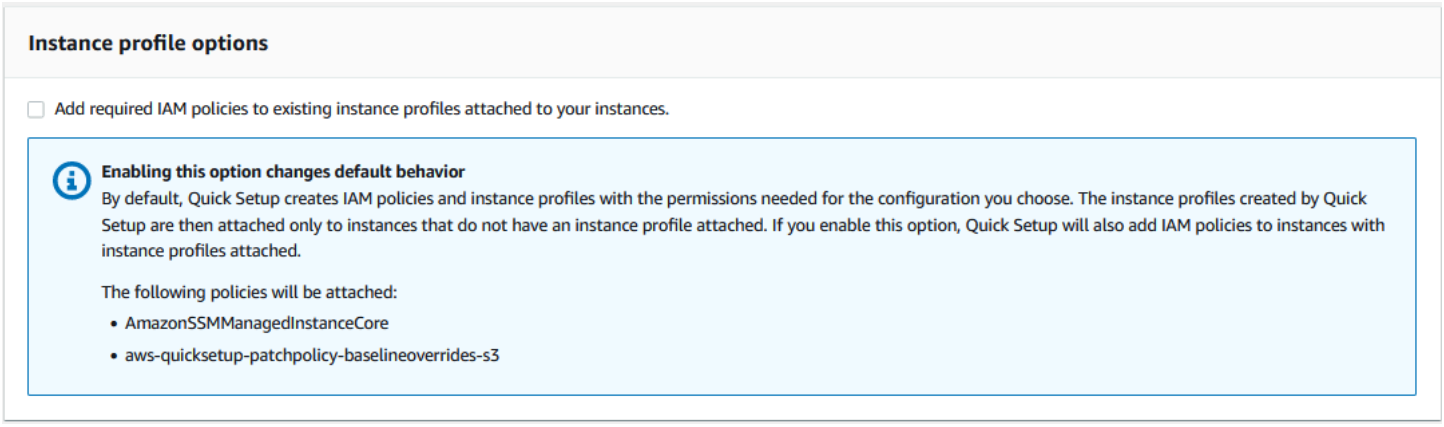
```
+ $response = Invoke-PatchBaselineOperation -Operation Install -Snapsho ...
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
+ CategoryInfo : OperationStopped: (Amazon.Patch.Ba...UpdateOpera
tion:InstallWindowsUpdateOperation) [Invoke-PatchBaselineOperation], Amazo
nS3Exception
+ 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.
```

Causa: creó una política de revisiones en Quick Setup y algunos de sus nodos gestionados ya tenían un perfil de instancia asociado (para instancias de EC2) o un rol de servicio asociado (para máquinas que no sean de EC2).

Sin embargo, como se muestra en la siguiente imagen, no seleccionó la casilla Agregar las políticas de IAM requeridas a los perfiles de instancia existentes asociados a sus instancias.



Cuando crea una política de revisiones, se crea un bucket de Amazon S3 para almacenar el archivo `baseline_overrides.json` de configuración de la política. Si no selecciona la casilla Agregar las políticas de IAM requeridas a los perfiles de instancia existentes asociados a sus instancias cuando crea la política, las políticas de IAM y las etiquetas de recursos que se necesitan para acceder a `baseline_overrides.json` en el bucket de S3 no se agregan automáticamente a los perfiles de instancia de IAM y roles de servicio existentes.

Solución 1: elimine la configuración de políticas de revisiones existente y, a continuación, cree una que la sustituya. Asegúrese de seleccionar la casilla Agregar las políticas de IAM requeridas a los perfiles de instancia existentes asociados a sus instancias. Esta selección aplica las políticas de IAM creadas por Quick Setup a los nodos que ya tienen un perfil de instancia o un rol de servicio asociado. (De forma predeterminada, Quick Setup agrega las políticas necesarias a las instancias y los nodos que aún no tienen perfiles de instancia o roles de servicio). Para obtener más información, consulte Automatizar la aplicación de revisiones en toda la organización mediante una política de revisiones de Quick Setup.

Solución 2: agregue manualmente los permisos y las etiquetas necesarios a cada perfil de instancia de IAM y rol de servicio de IAM que utilice con Quick Setup. Para obtener instrucciones, consulte Permisos para el bucket de S3 de la política de revisiones.

Problema: la implementación de revisiones falla sin una causa aparente y sin arrojar un mensaje de error

Problema: una operación de implementación de revisiones falla sin arrojar un error.

Causa posible: si `AWS-RunPatchBaseline` se invoca más de una vez, estas pueden entrar en conflicto entre sí y provocar un error en las tareas de implementación de revisiones. Es posible que esto no se indique en los registros de implementación de revisiones.

Para comprobar si las operaciones simultáneas de aplicación de revisiones pueden haberse interrumpido entre sí, revise el historial de comandos en Run Command, una herramienta de AWS Systems Manager. En el caso de un nodo administrado con un error durante la implementación de revisiones, compruebe si se intentó implementar revisiones a la máquina mediante varias operaciones con una diferencia de 2 minutos entre sí. En ocasiones, este escenario puede provocar un error.

También puede usar la AWS Command Line Interface (AWS CLI) para comprobar si hay intentos simultáneos de implementación de revisiones mediante el siguiente comando. Sustituya el valor del *`node-id`* por el ID del nodo administrado.

```
aws ssm list-commands \
    --filter "key=DocumentName,value=AWS-RunPatchBaseline" \
    --query 'Commands[*].
{CommandId:CommandId,RequestedDateTime:RequestedDateTime,Status:Status}' \
    --instance-id node-id \
    --output table
```

Solución: si determina que la implementación de revisiones falló debido a operaciones de implementación de revisiones realizadas en el mismo nodo administrado, ajuste las configuraciones de implementación de revisiones para evitar que esto vuelva a ocurrir. Por ejemplo, si dos periodos de mantenimiento especifican tiempos de implementación de revisiones superpuestas, elimine o revise uno de ellos. Si un periodo de mantenimiento especifica una operación de implementación de revisiones, pero una política de revisiones especifica una operación diferente para el mismo momento, considere eliminar la tarea del periodo de mantenimiento.

Si determina que las operaciones de implementación de revisiones conflictivas no fueron la causa del error en este escenario, recomendamos que se ponga en contacto con AWS Support.

Problema: resultados de conformidad de revisiones inesperados

Problema: al revisar los detalles de conformidad de revisiones generados después de una operación de `Scan`, los resultados incluyen información que no refleja las reglas establecidas en la línea de base de revisiones. Por ejemplo, una excepción que haya agregado a la lista Rejected patches (Revisiones rechazadas) en una línea de base de revisiones aparece como `Missing`. O bien, las revisiones clasificadas como `Important` aparecen como faltantes, aunque la línea de base de revisiones especifique únicamente las revisiones `Critical`.

Causa: Patch Manager admite actualmente varios métodos de ejecución de operaciones `Scan`:

- Una política de revisiones configurada en Quick Setup
- Una opción de administración de host configurada en Quick Setup
- Una ventana de mantenimiento para ejecutar una revisión `Scan` o una tarea `Install`
- Una operación Patch Now (Aplicar revisión ahora) bajo demanda

Cuando se ejecuta una operación de `Scan`, sobrescribe los detalles de conformidad del análisis más reciente. Si tiene más de un método configurado para ejecutar una operación de `Scan` y estos utilizan diferentes líneas de base de revisiones con diferentes reglas, los resultados de conformidad de las revisiones variarán.

Solución: para evitar resultados inesperados en cuanto a la conformidad de las revisiones, se recomienda utilizar solo un método a la vez para ejecutar la operación de Patch Manager `Scan`. Para obtener más información, consulte Evitar sobrescrituras involuntarias de datos de conformidad de revisiones.

Errores al ejecutar **AWS-RunPatchBaseline** en Linux

Temas

- Asunto: error “No such file or directory” (No existe tal archivo o directorio)
- Asunto: error “another process has acquired yum lock” (otro proceso tiene el bloqueo de yum)
- Asunto: error “Permission denied / failed to run commands” (Permiso denegado/Error al ejecutar comandos)
- Asunto: error “Unable to download payload” (No se puede descargar la carga)
- Asunto: error “unsupported package manager and python version combination” (Combinación de administrador de paquetes y versión de python no compatible)
- Asunto: Patch Manager no aplica las reglas especificadas para excluir determinados paquetes
- Asunto: se produce un error en la aplicación de revisiones y Patch Manager notifica que la extensión de la indicación de nombre de servidor a TLS no está disponible
- Asunto: Patch Manager notifica “No more mirrors to try” (No más espejos para probar)
- Problema: la implementación de revisiones falla y arroja el mensaje “Error code returned from curl is 23”.
- Problema: la implementación de revisiones falla y arroja el mensaje “Error unpacking rpm package…”
- Problema: la instalación de revisiones falla y arroja el mensaje “Encounter service side error when uploading the inventory”.
- Problema: la implementación de revisiones falla y arroja el mensaje “Se encontraron errores cuando se descargaron los paquetes”.
- Problema: la implementación de revisiones falla y arroja el mensaje “The following signatures couldn't be verified because the public key is not available”
- Problema: la implementación de revisiones falla y arroja el mensaje “NoMoreMirrorsRepoError”
- Problema: la implementación de revisiones falla y arroja el mensaje “No se puede descargar la carga”.
- Problema: la implementación de revisiones falla y arroja el mensaje “install errors: dpkg: error: dpkg frontend is locked by another process”
- Problema: cuando implementa las revisiones, se produce un error del Ubuntu Server que indica “dpkg was interrupted”
- Problema: la utilidad del administrador de paquetes no puede resolver una dependencia del paquete

- Problema: errores de dependencia de bloqueo de paquetes de Zypper en los nodos administrados de SLES

Asunto: error “No such file or directory” (No existe tal archivo o directorio)

Problema: cuando se ejecuta `AWS-RunPatchBaseline`, la aplicación de revisiones presenta uno de los siguientes errores.

```
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
```

Causa 1: se ejecutaban dos comandos `AWS-RunPatchBaseline` al mismo tiempo en el mismo nodo administrado. De este modo, se crea una condición de velocidad que da lugar a que no se cree `file patch-baseline-operations*` temporal ni se acceda correctamente a él.

Causa 2: no hay suficiente espacio de almacenamiento en el directorio `/var`.

Solución 1: asegúrese de que no haya ningún periodo de mantenimiento con dos o más tareas de Run Command que ejecuten `AWS-RunPatchBaseline` con el mismo nivel de prioridad y que se ejecuten en los mismos ID de destino. Si este es el caso, se debe reordenar la prioridad. Run Command es una herramienta de AWS Systems Manager.

Solución 2: asegúrese de que solo un periodo de mantenimiento a la vez esté ejecutando tareas de Run Command que utilicen `AWS-RunPatchBaseline` en los mismos destinos y dentro de la misma programación. En este caso, cambie la programación.

Solución 3: asegúrese de que solo una asociación de State Manager esté ejecutando `AWS-RunPatchBaseline` en la misma programación y se dirija a los mismos nodos administrados. State Manager es una herramienta de AWS Systems Manager.

Solución 4: libere espacio de almacenamiento suficiente en el directorio `/var` para los paquetes de actualización.

Asunto: error “another process has acquired yum lock” (otro proceso tiene el bloqueo de yum)

Problema: cuando se ejecuta `AWS-RunPatchBaseline`, la aplicación de revisiones presenta el siguiente error.

```
12/20/2019 21:41:48 root [INFO]: another process has acquired yum lock, waiting 2 s and
 retry.
```

Causa: el documento `AWS-RunPatchBaseline` ha comenzado a ejecutarse en un nodo administrado que ya se ejecuta en otra operación y ha adquirido el proceso `yum` del administrador de paquetes.

Solución: asegúrese de que ninguna asociación de State Manager, tarea de periodo de mantenimiento u otra configuración que ejecute `AWS-RunPatchBaseline` de forma programada tenga como destino el mismo nodo administrado aproximadamente al mismo tiempo.

Asunto: error “Permission denied / failed to run commands” (Permiso denegado/Error al ejecutar comandos)

Problema: cuando se ejecuta `AWS-RunPatchBaseline`, la aplicación de revisiones presenta el siguiente error.

```
sh:
/var/lib/amazon/ssm/instanceid/document/orchestration/commandid/PatchLinux/_script.sh:
 Permission denied
failed to run commands: exit status 126
```

Causa: `/var/lib/amazon/` podría montarse con permisos de `noexec`. Esto supone un problema, ya que SSM Agent descarga los scripts de carga en `/var/lib/amazon/ssm` y los ejecuta desde esa ubicación.

Solución: asegúrese de haber configurado particiones exclusivas para `/var/log/amazon` y `/var/lib/amazon`, y que están montados con permisos de `exec`.

Asunto: error “Unable to download payload” (No se puede descargar la carga)

Problema: cuando se ejecuta `AWS-RunPatchBaseline`, la aplicación de revisiones presenta el siguiente error.

```
Unable to download payload: https://s3.amzn-s3-demo-bucket.region.amazonaws.com/
aws-ssm-region/patchbaselineoperations/linux/payloads/patch-baseline-operations-
X.XX.tar.gz.failed to run commands: exit status 156
```

Causa: el nodo administrado no cuenta con los permisos necesarios para obtener acceso al bucket de Amazon Simple Storage Service (Amazon S3) especificado.

Solución: actualice la configuración de la red para que se pueda acceder a los puntos de enlace de S3. Para obtener más información, consulte la información acerca del acceso necesario a los buckets de S3 para Patch Manager en Comunicaciones de SSM Agent con buckets de S3 administrados de AWS.

Asunto: error “unsupported package manager and python version combination” (Combinación de administrador de paquetes y versión de python no compatible)

Problema: cuando se ejecuta `AWS-RunPatchBaseline`, la aplicación de revisiones presenta el siguiente error.

```
An unsupported package manager and python version combination was found. Apt requires
 Python3 to be installed.
failed to run commands: exit status 1
```

Causa: no se ha instalado una versión de python3 compatible en la instancia de Debian Server, Raspberry Pi OS o Ubuntu Server.

Solución: instale una versión de python3 compatible (3.0 o 3.10) en el servidor, lo que es necesario para los nodos administrados de Debian Server, Raspberry Pi OS y Ubuntu Server.

Asunto: Patch Manager no aplica las reglas especificadas para excluir determinados paquetes

Problema: ha intentado excluir determinados paquetes al especificarlos en el archivo `/etc/yum.conf`, con el formato `exclude=`*`package-name`*, pero no se excluyen durante la operación `Install` de Patch Manager.

Causa Patch Manager: no incluye las exclusiones especificadas en el archivo `/etc/yum.conf`.

Solución: para excluir paquetes específicos, cree una línea de base de revisiones personalizada y cree una regla que excluya aquellos paquetes que no desea instalar.

Asunto: se produce un error en la aplicación de revisiones y Patch Manager notifica que la extensión de la indicación de nombre de servidor a TLS no está disponible

Problema: la operación de aplicación de revisiones muestra el siguiente mensaje.

```
/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
```

Causa: este mensaje no indica un error. En su lugar, aparece una advertencia de que la versión más antigua de Python distribuida con el sistema operativo no es compatible con la indicación de nombre de servidor de TLS. El script de carga de revisiones de Systems Manager muestra esta advertencia cuando se conecta a las API de AWS que son compatibles con SNI.

Solución: para solucionar cualquier error en la aplicación de revisiones cuando se notifica este mensaje, revise el contenido de los archivos `stdout` y `stderr`. Si no ha configurado la línea de base de revisiones para almacenar estos archivos en un bucket de S3 o en Registros de Amazon CloudWatch, puede ubicar los archivos en la siguiente ubicación en su nodo administrado de Linux.

`/var/lib/amazon/ssm/`*`instance-id`*`/document/orchestration/`*`Run-Command-execution-id`*`/awsrunShellScript/PatchLinux`

Asunto: Patch Manager notifica “No more mirrors to try” (No más espejos para probar)

Problema: la operación de aplicación de revisiones muestra el siguiente mensaje.

```
[Errno 256] No more mirrors to try.
```

Causa: los repositorios configurados en el nodo administrado no funcionan correctamente. Entre las causas posibles se incluyen las siguientes:

- La memoria caché de yum está dañada.
- No se puede acceder a la URL de un repositorio debido a problemas con la red.

Solución: Patch Manager utiliza el administrador de paquetes predeterminado del nodo administrado para realizar la operación de aplicación de revisiones. Verifique que los repositorios se han configurado y funcionan correctamente.

Problema: la implementación de revisiones falla y arroja el mensaje “Error code returned from curl is 23”.

Problema: una operación de implementación de revisiones que usa `AWS-RunPatchBaseline` falla y arroja un error similar al siguiente:

```
05/01/2025 17:04:30 root [ERROR]: Error code returned from curl is 23
```

Causa: la herramienta curl que se utiliza en sus sistemas carece de los permisos necesarios para escribir en el sistema de archivos. Esto puede ocurrir si la herramienta curl predeterminada del administrador de paquetes se reemplazó por una versión diferente, como una instalada con snap.

Solución: si la versión curl que proporciona el administrador de paquetes se desinstaló tras la instalación de una versión diferente, vuelva a instalarla.

Si necesita mantener instaladas varias versiones de curl, asegúrese de que la versión asociada al administrador de paquetes esté en el primer directorio de la variable `PATH`. Para comprobarlo, ejecute el comando `echo $PATH` para ver el orden actual de los directorios en los que se comprueban los archivos ejecutables del sistema.

Problema: la implementación de revisiones falla y arroja el mensaje “Error unpacking rpm package…”

Problema: una operación de implementación de revisiones falla y arroja un error similar al siguiente:

```
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
```

Causa 1: cuando un paquete concreto está presente en varios instaladores de paquetes como `pip`, `yum` o `dnf`, pueden producirse conflictos cuando se utiliza el administrador de paquetes predeterminado.

Un ejemplo habitual es el del paquete `urllib3`, que se encuentra en `pip`, `yum` y `dnf`.

Causa 2: el paquete de `python-urllib3` está dañado. Esto puede suceder si los archivos del paquete se instalaron o actualizaron usando `pip` después de que el paquete `rpm` fuera instalado previamente mediante `yum` o `dnf`.

Solución: elimine el paquete `python-urllib3` de pip ejecutando el comando `sudo pip uninstall urllib3` y manteniendo el paquete solo en el administrador de paquetes predeterminado (`yum` o `dnf`).

Problema: la instalación de revisiones falla y arroja el mensaje “Encounter service side error when uploading the inventory”.

Problema: al ejecutar el documento `AWS-RunPatchBaseline`, aparece el siguiente mensaje de error:

```
Encounter service side error when uploading the inventory
```

Causa: se ejecutaban dos comandos `AWS-RunPatchBaseline` al mismo tiempo en el mismo nodo administrado. Esto crea una condición de carrera al inicializar el cliente boto3 durante las operaciones de aplicación de revisiones.

Solución: asegúrese de que ninguna asociación de State Manager, tarea de periodo de mantenimiento u otra configuración que ejecute `AWS-RunPatchBaseline` de forma programada tenga como destino el mismo nodo administrado aproximadamente al mismo tiempo.

Problema: la implementación de revisiones falla y arroja el mensaje “Se encontraron errores cuando se descargaron los paquetes”.

Problema: durante la implementación de revisiones, recibe un error similar al siguiente:

```
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',
```

Causa: este error puede producirse cuando no hay suficiente memoria disponible en un nodo administrado.

Solución: configure la memoria de intercambio o actualice la instancia a un tipo diferente para aumentar la compatibilidad con la memoria. A continuación, inicie una nueva operación de implementación de revisiones.

Problema: la implementación de revisiones falla y arroja el mensaje “The following signatures couldn't be verified because the public key is not available”

Problema: la implementación de revisiones falla en Ubuntu Server y arroja un error similar al siguiente:

```
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
```

Causa: la clave GNU Privacy Guard (GPG) caducó o falta.

Solución: actualice la clave GPG o vuelva a agregarla.

Por ejemplo, si tomamos como ejemplo el error anterior, vemos que falta la clave `467B942D3A79BD29` y es necesario agregarla. Para ello, ejecute cualquiera de los comandos siguientes:

```
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
```

O bien, ejecute el siguiente comando para actualizar todas las claves:

```
sudo apt-key adv --keyserver hkps://keyserver.ubuntu.com --refresh-keys
```

Si el error se repite después de esto, recomendamos informar el problema a la organización que mantiene el repositorio. Hasta que haya una solución disponible, puede editar el archivo `/etc/apt/sources.list` para omitir el repositorio durante el proceso de implementación de revisiones.

Para ello, abra el archivo `sources.list` para editarlo, localice la línea del repositorio e inserte un carácter # al principio de la línea para comentarla. Guarde el archivo y, a continuación, ciérrelo.

Problema: la implementación de revisiones falla y arroja el mensaje “NoMoreMirrorsRepoError”

Problema: recibe un error similar al siguiente:

```
NoMoreMirrorsRepoError: failure: repodata/repomd.xml from pgdg94: [Errno 256] No more
 mirrors to try.
```

Causa: hay un error en el repositorio de origen.

Solución: recomendamos informar el problema a la organización que mantiene el repositorio. Hasta que se corrija el error, puede deshabilitar el repositorio a nivel del sistema operativo. Para ello, ejecute el siguiente comando y reemplace el valor *repo-name* por el nombre de su repositorio:

```
yum-config-manager --disable repo-name
```

A continuación se muestra un ejemplo.

```
yum-config-manager --disable pgdg94
```

Tras ejecutar este comando, ejecute otra operación de implementación de revisiones.

Problema: la implementación de revisiones falla y arroja el mensaje “No se puede descargar la carga”.

Problema: recibe un error similar al siguiente:

```
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
```

Causa: la configuración del nodo administrado contiene errores o está incompleta.

Solución: asegúrese de que el nodo administrado esté configurado con lo siguiente:

- Regla TCP 443 de salida en el grupo de seguridad.
- Regla TCP 443 de salida en NACL.
- Regla TCP 1024-65535 de ingreso en NACL.
- NAT/IGW en la tabla de enrutamiento para proporcionar conectividad a un punto de conexión de S3. Si la instancia no tiene acceso a Internet, proporcione la conectividad con el punto de conexión de S3. Para ello, agregue un punto de conexión de puerta de enlace de S3 en la VPC e intégrelo con la tabla de enrutamiento del nodo administrado.

Problema: la implementación de revisiones falla y arroja el mensaje “install errors: dpkg: error: dpkg frontend is locked by another process”

Problema: la implementación de revisiones falla y arroja un error similar al siguiente:

```
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
```

Causa: el administrador de paquetes ya está ejecutando otro proceso en un nodo administrado a nivel del sistema operativo. Si ese otro proceso tarda mucho en completarse, es posible que se agote el tiempo de espera de la operación de implementación de revisiones de Patch Manager y se produzca un error.

Solución: una vez finalizado el otro proceso que utiliza el administrador de paquetes, ejecute una nueva operación de implementación de revisiones.

Problema: cuando implementa las revisiones, se produce un error del Ubuntu Server que indica “dpkg was interrupted”

Problema: la implementación de revisiones falla en el Ubuntu Server y arroja un error similar al siguiente:

```
E: dpkg was interrupted, you must manually run
'dpkg --configure -a' to correct the problem.
```

Causa: uno o más paquetes están mal configurados.

Solución: siga los pasos que se indican a continuación:

1. Compruebe qué paquetes están afectados y cuáles son los problemas de cada paquete ejecutando los siguientes comandos, uno por uno:

   ```
   sudo apt-get check
   ```

   ```
   sudo dpkg -C
   ```

   ```
   dpkg-query -W -f='${db:Status-Abbrev} ${binary:Package}\n' | grep -E ^.[^nci]
   ```

2. Corrija los paquetes que tengan problemas ejecutando el siguiente comando:

   ```
   sudo dpkg --configure -a
   ```

3. Si el comando anterior no resolvió por completo el problema, ejecute el comando siguiente:

```
sudo apt --fix-broken install
```

Problema: la utilidad del administrador de paquetes no puede resolver una dependencia del paquete

Problema: el administrador de paquetes nativo del nodo administrado no puede resolver una dependencia del paquete y se produce un error cuando se implementan las revisiones. El siguiente ejemplo de mensaje de error indica este tipo de error en un sistema operativo que utiliza `yum` como administrador de paquetes.

```
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']
```

Causa: en los sistemas operativos Linux, Patch Manager utiliza el administrador de paquetes nativo de la máquina para ejecutar las operaciones de implementación de revisiones, como `yum`, `dnf`, `apt` y `zypper`. Las aplicaciones detectan, instalan, actualizan o eliminan automáticamente los paquetes dependientes según sea necesario. Sin embargo, algunas condiciones pueden provocar que el administrador de paquetes no pueda completar una operación de dependencia, como las siguientes:

- Hay varios repositorios conflictivos configurados en el sistema operativo.
- No se puede acceder a la URL de un repositorio remoto debido a problemas relacionados con la red.
- En el repositorio se encuentra un paquete para una arquitectura incorrecta.

Solución: la implementación de revisiones puede fallar debido a un problema de dependencia por una amplia variedad de motivos. Por lo tanto, recomendamos que se ponga en contacto con AWS Support para obtener ayuda con la solución de problemas.

Problema: errores de dependencia de bloqueo de paquetes de Zypper en los nodos administrados de SLES

Problema: cuando se ejecuta `AWS-RunPatchBaseline` con la operación `Install` en las instancias de SUSE Linux Enterprise Server, se produce un error al aplicar los parches y se producen errores de comprobación de dependencias relacionados con el bloqueo de paquetes. Podría obtener mensajes de error similares al siguiente:

```
Problem: mock-pkg-has-dependencies-0.2.0-21.adistro.noarch requires mock-pkg-standalone
 = 0.2.0, but this requirement cannot be provided
  uninstallable providers: mock-pkg-standalone-0.2.0-21.adistro.noarch[local-repo]
 Solution 1: remove lock to allow installation of mock-pkg-
standalone-0.2.0-21.adistro.noarch[local-repo]
 Solution 2: do not install mock-pkg-has-dependencies-0.2.0-21.adistro.noarch
 Solution 3: break mock-pkg-has-dependencies-0.2.0-21.adistro.noarch by ignoring some
 of its dependencies

Choose from above solutions by number or cancel [1/2/3/c] (c): c
```

En este ejemplo, el paquete `mock-pkg-standalone` está bloqueado; puede comprobarlo si ejecuta `sudo zypper locks` y busca el nombre de este paquete en el resultado.

O puede ver entradas de registro que indican errores en la comprobación de dependencias:

```
Encountered a known exception in the CLI Invoker:
 CLIInvokerError(error_message='Dependency check failure during commit process',
 error_code='4')
```

> **Note**
>
> Este problema solo se produce durante las operaciones `Install`. Las operaciones `Scan` no aplican bloqueos a los paquetes y no se ven afectadas por los bloqueos existentes.

Causa: este error se produce cuando los bloqueos de paquetes de Zypper impiden la instalación o actualización de los paquetes debido a conflictos de dependencia. Los bloqueos de paquetes pueden estar presentes por varias razones:

- Bloqueos aplicados por el cliente: usted o el administrador del sistema bloquearon de forma manual los paquetes mediante comandos de Zypper como `zypper addlock`.
- Parches rechazados por el Administrador de parches: Patch Manager bloquea automáticamente los paquetes cuando se especifican paquetes en la lista de parches rechazados de la línea de base de revisiones para impedir su instalación.
- Bloqueos residuales por operaciones interrumpidas: en raras ocasiones, si una operación de parche se interrumpió (por ejemplo, al reiniciar el sistema) antes de que Patch Manager pueda eliminar los bloqueos temporales, es posible que los bloqueos de paquetes residuales permanezcan en el nodo administrado.

Solución: para resolver los problemas de bloqueo de paquetes de Zypper, siga estos pasos en función de la causa:

Paso 1: identifique los paquetes bloqueados

Conéctese al nodo de SLES administrado y ejecute el siguiente comando para ver todos los paquetes bloqueados actualmente:

```
sudo zypper locks
```

Paso 2: determine el origen de los bloqueos

- Si los paquetes bloqueados son aquellos que usted bloqueó intencionalmente para garantizar la estabilidad del sistema, considere si es necesario que permanezcan bloqueados o si se pueden desbloquear temporalmente para parchearlos.
- Si los paquetes bloqueados coinciden con las entradas de la lista de parches rechazados de la línea de base de revisiones, es probable que se trate de bloqueos residuales debidos a una interrupción de la operación del parche. Durante el funcionamiento normal, Patch Manager aplica estos bloqueos de manera temporal y los elimina automáticamente cuando finaliza la operación. Puede eliminar los paquetes de la lista de rechazados o modificar las reglas de la línea de base de revisiones.
- Si no reconoce los paquetes bloqueados y no se bloquearon intencionalmente, es posible que se trate de bloqueos residuales de una operación de parche interrumpida anteriormente.

Paso 3: elimine los bloqueos según convenga

Utilice el siguiente comando para eliminar bloqueos de paquetes específicos:

```
sudo zypper removelock package-name
```

Para eliminar todos los bloqueos de paquetes (utilícelo con precaución), ejecute:

```
sudo zypper cleanlocks
```

Paso 4: actualice la línea de base de revisiones (si corresponde)

Si los bloqueos se debieron a parches rechazados en la línea de base de revisiones:

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Patch Manager.
3. Seleccione la pestaña Línea de base de revisiones, y luego elija su línea de base de revisiones personalizada.
4. Seleccione Acciones y Modificar la línea de base de revisiones.
5. En la sección Parches rechazados, revise los paquetes de la lista y elimine los que deberían poder instalarse.
6. Seleccione Save changes (Guardar cambios).

Paso 5: vuelva a intentar la operación del parche

Tras eliminar los bloqueos problemáticos y actualizar la línea de base de revisiones, si fuera necesario, vuelva a ejecutar el documento `AWS-RunPatchBaseline`.

> **Note**
>
> Cuando Patch Manager aplica bloqueos a los parches rechazados durante las operaciones `Install`, está diseñado para eliminar estos bloqueos automáticamente una vez finalizada la operación del parche. Si ve estos bloqueos cuando se ejecuta `sudo zypper locks`, significa que se interrumpió una operación de parche anterior antes de que se pudiera proceder a la limpieza. Sin embargo, si se interrumpe la operación del parche, es posible que sea necesaria una limpieza manual, tal como se describe en este procedimiento.

Prevención: para evitar futuros conflictos de bloqueo de Zypper:

- Revise detenidamente la lista de parches rechazados de la línea de base de revisiones para asegurarse de que solo incluye los paquetes que realmente desea excluir.
- Evite bloquear de forma manual los paquetes que puedan ser necesarios como dependencias para las actualizaciones de seguridad.
- Si debe bloquear los paquetes manualmente, documente los motivos y revise los bloqueos con regularidad.
- Asegúrese de que las operaciones del parche se completen correctamente y no se vean interrumpidas por el reinicio del sistema u otros factores.

- Supervise las operaciones del parche hasta que se completen y evite interrumpirlas al reiniciar el sistema o realizar otras acciones que puedan impedir una limpieza adecuada de los bloqueos temporales.

Errores al ejecutar **`AWS-RunPatchBaseline`** en Windows Server

Temas

- Asunto: pares de familia de productos y productos no coincidentes
- Asunto: el resultado de AWS-RunPatchBaseline devuelve un HRESULT (Windows Server)
- Asunto: el nodo administrado no tiene acceso al catálogo de Windows Update ni a WSUS
- Asunto: el módulo PatchBaselineOperations de PowerShell no se puede descargar
- Asunto: revisiones faltantes

Asunto: pares de familia de productos y productos no coincidentes

Problema: cuando crea una línea de base de revisiones en la consola de Systems Manager, debe especificar una familia de productos y un producto. Por ejemplo, puede elegir:

- Product family (Familia de productos): `Office`

  Product (Producto): `Office 2016`

Causa: si intenta crear una línea de base de revisiones con una par de producto y familia de productos que no coincide, se mostrará un mensaje de error. A continuación se indican los motivos por los que esto puede ocurrir:

- Ha seleccionado un par de producto y familia de productos válidos, pero luego ha eliminado la selección de familia.
- Ha seleccionado un producto de la lista secundaria Obsolete or mismatched options (Opciones obsoletas o no coincidentes) en lugar de seleccionar la lista secundaria Available and matching options (Opciones disponibles y coincidentes).

  Los elementos de la lista secundaria Obsolete or mismatched options (Opciones obsoletas o no coincidentes) de productos podrían haberse ingresado por error mediante un SDK o un comando `create-patch-baseline` de la AWS Command Line Interface (AWS CLI). Esto podría significar que se introdujo un error tipográfico o se asignó un producto a la familia del producto equivocado.

Un producto también aparece en la lista secundaria Obsolete or mismatched options (Opciones obsoletas o no coincidentes) si se ha especificado para una línea de base de revisiones anterior, pero no tiene revisiones disponibles de Microsoft.

Solución: para evitar este problema en la consola, elija siempre opciones de las listas secundarias Currently available options (Opciones disponibles actualmente).

También puede ver los productos que tienen revisiones disponibles mediante el comando `describe-patch-properties` de la AWS CLI o el comando `DescribePatchProperties` de la API.

Asunto: el resultado de **`AWS-RunPatchBaseline`** devuelve un **`HRESULT`** (Windows Server)

Problema: ha recibido un error como el siguiente.

```
----------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
```

Causa: este resultado indica que las API nativas de Windows Update no han podido ejecutar las operaciones de aplicación de revisiones.

Solución: verifique el código `HResult` en los siguientes temas de microsoft.com para identificar los pasos de solución de problemas que permitan resolver el error:

- Códigos de error de Windows Update por componente
- Errores comunes y mitigación de Windows Update

Asunto: el nodo administrado no tiene acceso al catálogo de Windows Update ni a WSUS

Problema: ha recibido un error como el siguiente.

```
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----
```

Causa: este error podría estar relacionado con los componentes de Windows Update, o bien con la falta de conectividad con el catálogo de Windows Update o con Windows Server Update Services (WSUS).

Solución: confirme que el nodo administrado tiene conectividad con el catálogo de Microsoft Update a través de una puerta de enlace de Internet, una puerta de enlace NAT o una instancia NAT. Si utiliza WSUS, confirme que el nodo administrado dispone de conectividad con el servidor WSUS de su entorno. Si la conectividad está disponible para el destino previsto, verifique otras causas posibles de `HResult 0x80072EE2` en la documentación de Microsoft. Esto podría significar un problema a nivel del sistema operativo.

Asunto: el módulo PatchBaselineOperations de PowerShell no se puede descargar

Problema: ha recibido un error como el siguiente.

```
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
```

Solución: verifique la conectividad del nodo administrado y los permisos a Amazon Simple Storage Service (Amazon S3). El rol de AWS Identity and Access Management (IAM) del nodo administrado

debe utilizar los permisos mínimos citados en Comunicaciones de SSM Agent con buckets de S3 administrados de AWS. El nodo debe comunicarse con el punto de conexión de Amazon S3 a través del punto de conexión de la puerta de enlace de Amazon S3, la puerta de enlace NAT o la puerta de enlace de Internet. Para obtener más información acerca de los requisitos del punto de enlace de la VPC para AWS Systems Manager SSM Agent (SSM Agent), consulte Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager.

Asunto: revisiones faltantes

Problema: `AWS-RunPatchbaseline` se ha completado correctamente, pero faltan algunos revisiones.

A continuación, se presentan algunas causas comunes y sus respectivas soluciones.

Causa 1: la base de referencia no es efectiva.

Solución 1: para comprobar si esta es la causa, utilice el siguiente procedimiento.

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command.
3. Seleccione la pestaña Command history (Historial de comandos) y, a continuación, seleccione el comando cuya base de referencia desea verificar.
4. Seleccione el nodo administrado al que faltan revisiones.
5. Seleccione Step 1 - Output (Paso 1: Resultado) y busque el valor `BaselineId`.
6. Verifique la configuración de la línea de base de revisiones asignada, es decir, el sistema operativo, el nombre del producto, la clasificación y la gravedad de la línea de base de revisiones.
7. Vaya a Microsoft Update Catalog (Catálogo de Microsoft Update).
8. Busque los ID de los artículos de Microsoft Knowledge Base (KB) (por ejemplo, KB3216916).
9. Compruebe que el valor en Product (Producto) concuerde con el del nodo administrado y seleccione el Title (Título) correspondiente. Se abrirá una nueva ventana Update Details (Actualizar detalles).
10. En la pestaña Overview (Información general), la classification (clasificación) y la MSRC severity (gravedad de MSRC) deben concordar con la configuración de la línea de base de revisiones que encontró con anterioridad.

Causa 2: se reemplazó la revisión.

Solución 2: para verificar si esta es así, utilice el siguiente procedimiento.

1. Vaya a Microsoft Update Catalog (Catálogo de Microsoft Update).
2. Busque los ID de los artículos de Microsoft Knowledge Base (KB) (por ejemplo, KB3216916).
3. Compruebe que el valor en Product (Producto) concuerde con el del nodo administrado y seleccione el Title (Título) correspondiente. Se abrirá una nueva ventana Update Details (Actualizar detalles).
4. Vaya a la pestaña Package Details (Detalles del paquete). Busque una entrada en el encabezado This update has been replaced by the following updates: (Esta actualización se ha sustituido por las siguientes actualizaciones:).

Causa 3: la misma revisión puede tener diferentes números de KB debido a que Microsoft gestiona las actualizaciones online de WSUS y Window como canales de publicación independientes.

Solución 3: compruebe la elegibilidad de las revisiones. Si el paquete no está disponible en WSUS, instale la compilación 14393.3115 del sistema operativo. Si el paquete está disponible para todas las compilaciones del sistema operativo, instale las compilaciones 18362.1256 y 18363.1256 del sistema operativo.

Cómo utilizar manuales de procedimientos de Automatización de AWS Support

AWS Support proporciona dos manuales de procedimientos de Automatización que puede usar para solucionar determinados problemas relacionados con las revisiones.

- `AWSSupport-TroubleshootWindowsUpdate`: el manual de procedimientos `AWSSupport-TroubleshootWindowsUpdate` se utiliza para identificar problemas que podrían causar un error en las actualizaciones de Windows Server para instancias de Windows Server de Amazon Elastic Compute Cloud (Amazon EC2).
- `AWSSupport-TroubleshootPatchManagerLinux`: el manual de procedimientos `AWSSupport-TroubleshootPatchManagerLinux` soluciona problemas comunes que pueden causar un error en las revisiones en nodos administrados basados en Linux con Patch Manager. El objetivo principal de este manual de procedimientos es identificar la causa raíz del error del comando de revisión y sugerir un plan de corrección.

> **Note**
>
> Se cobra un cargo por ejecutar manuales de procedimientos de Automatización. Para obtener más información, consulte los precios de AWS Systems Manager para Automatización.

Cómo ponerse en contacto con AWS Support

Si no encuentra soluciones a los problemas en esta sección o en los problemas de Systems Manager en AWS re:Post, y cuenta con un plan Developer Business o Enterprise de Soporte, puede crear un caso de soporte técnico en AWS Support.

Antes de contactar con Soporte, recopile los siguientes elementos:

- Registros de SSM Agent
- ID de comando de Run Command, ID de periodo de mantenimiento o ID de ejecución de Automation
- Para nodos administrados de Windows Server, recopile también lo siguiente:
  - `%PROGRAMDATA%\Amazon\PatchBaselineOperations\Logs` como se describe en la pestaña Windows de Cómo se instalan las revisiones.
  - Registros de actualización de Windows: para Windows Server 2012 R2 y versiones anteriores, utilice `%windir%/WindowsUpdate.log`. Para Windows Server 2016 y más recientes, ejecute primero el comando `Get-WindowsUpdateLog` de PowerShell antes de utilizar `%windir%/WindowsUpdate.log`.
- Para nodos administrados de Linux, recopile también lo siguiente:
  - El contenido del directorio `/var/lib/amazon/ssm/`*`instance-id`*`/document/orchestration/`*`Run-Command-execution-id`*`/awsrunShellScript/PatchLinux`

AWS Systems Manager Run Command

Con Run Command, una herramienta de AWS Systems Manager, puede administrar de forma remota y segura la configuración de los nodos administrados. Un nodo administrado es cualquier instancia de Amazon Elastic Compute Cloud (Amazon EC2) o cualquier máquina que no sea de EC2, en su entorno híbrido y multinube, que se haya configurado para Systems Manager. Run Command permite automatizar las tareas administrativas comunes y realizar cambios de configuración de

una sola vez a escala. Puede utilizar Run Command desde la AWS Management Console, la AWS Command Line Interface (AWS CLI), AWS Tools for Windows PowerShell o los AWS SDK. Run Command se ofrece sin costo adicional. Para comenzar a utilizar Run Command, abra la consola de Systems Manager. En el panel de navegación, elija Run Command.

Los administradores utilizan Run Command instalar o arrancar aplicaciones, crear una canalización de implementación, capturar archivos de registros cuando se elimina una instancia de un grupo de escalado automático, unir instancias a un dominio de Windows, y más.

La API de Run Command sigue un modelo de consistencia final debido a la naturaleza distribuida del sistema que admite la API. Esto significa que el resultado de ejecutar un comando de API que afecte a los recursos que tiene podría no estar inmediatamente visible para todos los comandos posteriores que ejecute. Debe tenérselo en cuenta cuando se ejecute un comando de API que siga inmediatamente a un comando de API anterior.

Introducción

En la siguiente tabla, se incluye información para ayudarlo a comenzar a utilizar Run Command.

Tema	Detalles
Configuración de nodos administrados para AWS Systems Manager	Verifique que haya completado los requisitos de configuración para las instancias de Amazon Elastic Compute Cloud (Amazon EC2) y las máquinas que no sean de EC2 en un entorno híbrido y multinube.
Administrador de nodos en entornos híbridos y multinube con Systems Manager	(Opcional) Registre los servidores locales y las máquinas virtuales en AWS para poder administrarlos con Run Command.
the section called “Administración de dispositivos periféricos con Systems Manager”	(Opcional) Configure dispositivos de borde para que pueda administrarlos mediante Run Command.
Ejecución de comandos en nodos administrados	Obtenga información sobre cómo ejecutar un comando que se dirige a uno o varios nodos administrados mediante la AWS Management Console.

Tema	Detalles
Tutoriales de Run Command	Aprenda cómo ejecutar comandos mediante Tools for Windows PowerShell o la AWS CLI.

Compatibilidad con EventBridge

Esta herramienta de Systems Manager se admite como un tipo de evento y un tipo de destino en las reglas de Amazon EventBridge. Para obtener más información, consulte Monitoreo de eventos de Systems Manager con Amazon EventBridge y Referencia: patrones y tipos de eventos de Amazon EventBridge para Systems Manager.

Más información

- Remotely Run Command on an EC2 Instance (10 minute tutorial) (Uso remoto de Run Command en una instancia de EC2 [tutorial de 10 minutos])
- Service Quotas de Systems Manager en la Referencia general de Amazon Web Services
- Referencia de la API de AWS Systems Manager

Temas

- Cómo configurar Run Command
- Ejecución de comandos en nodos administrados
- Uso de códigos de salida en los comandos
- Descripción de los estados del comando
- Tutoriales de Run Command
- Solución de problemas Systems Manager Run Command

Cómo configurar Run Command

Antes de poder administrar nodos con Run Command, una herramienta de AWS Systems Manager, debe configurar una política de AWS Identity and Access Management (IAM) para todos los usuarios que vayan a ejecutar comandos. Si utiliza alguna clave de condición global para la acción `SendCommand` en sus políticas de IAM, debe incluir la clave de condición `aws:ViaAWSService` y establecer el valor booleano en `true`. A continuación se muestra un ejemplo.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:SendCommand"
            ],
            "Resource": [
                "arn:aws:ssm:us-east-1:111122223333:document/YourDocument"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:SourceVpce": [
                        "vpce-1234567890abcdef0"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:SendCommand"
            ],
            "Resource": [
                "arn:aws:ssm:us-east-1:111122223333:document/YourDocument"
            ],
            "Condition": {
                "Bool": {
                    "aws:ViaAWSService": "true"
                }
            }
        }
    ]
}
```

También debe configurar los nodos para Systems Manager. Para obtener más información, consulte Configuración de nodos administrados para AWS Systems Manager.

Le recomendamos completar las siguientes tareas de configuración opcionales para ayudar a minimizar la posición de seguridad y la gestión diaria de los nodos administrados.

Monitoreo de la ejecución de comandos con Amazon EventBridge

Puede utilizar EventBridge para registrar los cambios de estado de la ejecución de comandos. Tiene la opción de crear una regla que se ejecute siempre que haya una transición de estado o cuando haya una transición a uno o varios estados de interés. También puede especificar Run Command como una acción de destino cuando se produce un evento de EventBridge. Para obtener más información, consulte Configuración de EventBridge para eventos de Systems Manager.

Monitoreo de la ejecución de comandos con los Registros de Amazon CloudWatch

Puede configurar Run Command para que envíe periódicamente todos los resultados y los registros de errores de los comandos a un grupo de registros de Amazon CloudWatch. Puede monitorizar estos registros de salida prácticamente en tiempo real, buscar frases, valores o patrones específicos y crear alarmas en función de la búsqueda. Para obtener más información, consulte Configuración de Registros de Amazon CloudWatch para Run Command.

Restrinja el acceso de Run Command a nodos administrados específicos

Puede restringir la capacidad de un usuario para ejecutar comandos en nodos administrados mediante AWS Identity and Access Management (IAM). En concreto, puede crear una política de IAM con la condición de que el usuario solo pueda ejecutar comandos en nodos administrados que estén etiquetados con etiquetas específicas. Para obtener más información, consulte Restricción de acceso de Run Command basado en etiquetas.

Restricción de acceso de Run Command basado en etiquetas

En esta sección se describe cómo restringir la capacidad de un usuario para ejecutar comandos en nodos administrados especificando una condición de etiqueta en una política de IAM. Los nodos administrados incluyen instancias de Amazon EC2 y nodos que no son de EC2 en un entorno híbrido y multinube configurado para Systems Manager. Aunque la información no se presenta explícitamente, también puede restringir el acceso a dispositivos de núcleo administrados de AWS IoT Greengrass. Para comenzar, debe etiquetar los dispositivos de AWS IoT Greengrass. Para obtener más información, consulte Etiquetar los recursos de AWS IoT Greengrass Version 2 en la Guía para desarrolladores de AWS IoT Greengrass Version 2.

Puede restringir la ejecución de comandos a determinados nodos administrados mediante la creación de una política de IAM que incluya la condición de que el usuario solo pueda ejecutar comandos en los nodos que tengan determinadas etiquetas. En el ejemplo siguiente, el usuario tiene permitido utilizar Run Command (`Effect: Allow, Action: ssm:SendCommand`) mediante cualquier documento de SSM (`Resource: arn:aws:ssm:*:*:document/*`) en cualquier nodo (`Resource: arn:aws:ec2:*:*:instance/*`) con la condición de que el nodo sea un servidor web de finanzas (`ssm:resourceTag/Finance: WebServer`). Si el usuario envía un comando a un nodo que no está etiquetado o que tiene una etiqueta que no es `Finance: WebServer`, los resultados de la ejecución mostrarán `AccessDenied`.

JSON

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

```
}
```

Puede crear políticas de IAM que permitan al usuario ejecutar comandos en los nodos administrados etiquetados con varias etiquetas. La siguiente política permite al usuario ejecutar comandos en nodos administrados que tienen dos etiquetas. Si un usuario envía un comando a un nodo que no está etiquetado con estas dos etiquetas, los resultados de la ejecución mostrarán `AccessDenied`.

JSON

```
{
   "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":"*"
      }
   ]
}
```

También puede crear políticas de IAM que permitan al usuario ejecutar comandos en varios grupos de nodos administrados etiquetados. La siguiente política de ejemplo permite al usuario ejecutar comandos en uno de los grupos de nodos etiquetados o en ambos grupos.

JSON

```
{
   "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":"*"
      }
   ]
}
```

Para obtener más información acerca de la creación de políticas de IAM, consulte Políticas administradas y políticas insertadas en la Guía del usuario de IAM. Para obtener más información acerca del etiquetado de nodos administrados, consulte Editor de etiquetas en la Guía del usuario de AWS Resource Groups.

Ejecución de comandos en nodos administrados

Esta sección contiene información sobre cómo enviar comandos desde la consola de AWS Systems Manager a nodos administrados. La sección también incluye información acerca de cómo cancelar un comando.

Tenga en cuenta que siisu nodo está configurado con la opción de montaje `noexec` para el directorio var, Run Command no podrá ejecutar correctamente los comandos.

> **Important**
>
> Cuando ejecute un comando con Run Command, no incluya información confidencial como texto sin formato, por ejemplo, contraseñas, datos de configuración u otros secretos. Toda la actividad de la API de Systems Manager de la cuenta se registra en un bucket de S3, para registros de AWS CloudTrail. Esto significa que cualquier usuario con acceso al bucket de S3 puede ver los valores en texto sin formato de esos secretos. Por este motivo, le recomendamos crear y utilizar parámetros `SecureString` para cifrar la información confidencial que utiliza en las operaciones de Systems Manager.
> Para obtener más información, consulte Restringir el acceso a los parámetros de Parameter Store mediante políticas de IAM.

Retención del historial de ejecución

El historial de cada comando está disponible durante un máximo de 30 días. Además, puede almacenar una copia de todos los archivos de registro en Amazon Simple Storage Service o disponer de un registro de auditoría de todas las llamadas de API en AWS CloudTrail.

Información relacionada

Para obtener más información acerca de cómo enviar comandos con otras herramientas, consulte los siguientes temas:

- Tutorial: uso de la AWS Tools for Windows PowerShell con Run Command o consulte los ejemplos de la sección de AWS Systems Manager de Referencia de Cmdlet de Herramientas de AWS para PowerShell.
- Tutorial: uso de la AWS CLI con Run Command o los ejemplos de la Referencia de la CLI de SSM

Contenido

- Ejecución de comandos desde la consola
- Ejecución de comandos mediante una versión de documento específica
- Ejecución de comandos a escala
- Cancelación de un comando

Ejecución de comandos desde la consola

Puede utilizar Run Command, una herramienta de AWS Systems Manager, desde la AWS Management Console para configurar nodos administrados sin tener que iniciar sesión en ellos. En este tema, se incluye un ejemplo en el que se muestra cómo actualizar SSM Agent en un nodo administrado utilizando Run Command.

Antes de empezar

Antes de enviar un comando con Run Command, verifique que los nodos administrados cumplan los requisitos de configuración de Systems Manager.

Para enviar un comando con Run Command

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command.
3. Elija Run command (Ejecutar comando).
4. En la lista Command document (Documento de Command), elija un documento de Systems Manager.
5. En la sección Command Parameters, especifique los valores de los parámetros obligatorios.
6. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

   > **Tip**
   >
   > Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

7. En Otros parámetros:
   - En Comentario, ingrese la información acerca de este comando.
   - En Tiempo de espera (segundos), especifique el número de segundos que tiene que esperar el sistema antes de indicar que se ha producido un error en la ejecución del comando general.
8. En Rate control (Control de velocidad):

- En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

  > **Note**
  >
  > Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

- En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

9. (Opcional) Elija una alarma de CloudWatch que desee aplicar al comando para fines de monitoreo. Para adjuntar una alarma de CloudWatch a su comando, la entidad principal de IAM que lo ejecute debe tener permiso para la acción `iam:createServiceLinkedRole`. Para obtener más información sobre las alarmas de CloudWatch, consulte Uso de alarmas de Amazon CloudWatch. Tenga en cuenta que si la alarma se activa, no se ejecutará ninguna invocación de comando pendiente.
10. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

    > **Note**
    >
    > Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

11. En la sección Notificaciones de SNS, seleccione la casilla de verificación Habilitar notificaciones de SNS si desea recibir notificaciones sobre el estado de ejecución de los comandos.

    Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.
12. Seleccione Ejecutar.

Para obtener más información acerca de la cancelación de un comando, consulte the section called “Cancelación de un comando”.

Volver a ejecutar comandos

Systems Manager incluye dos opciones que lo ayudarán a volver a ejecutar un comando desde la página de Run Command en la consola de Systems Manager.

- Rerun (Volver a ejecutar): este botón le permite ejecutar el mismo comando sin realizarle cambios.
- Copy to new (Copiar en nuevo): este botón copia la configuración de un comando en un comando nuevo y le ofrece la opción de editar esa configuración antes de ejecutarlo.

Para volver a ejecutar un comando

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command.
3. Elija el comando que desea volver a ejecutar. Puede volver a ejecutar un comando inmediatamente después de ejecutarlo desde la página de detalles del comando. O bien, puede elegir un comando que haya ejecutado anteriormente en la pestaña Command history (Historial de comandos).
4. Elija Rerun (Volver a ejecutar) para ejecutar el mismo comando sin realizar ningún cambio o elija Copy to new (Copiar en nuevo) para editar la configuración del comando antes de ejecutarlo.

Ejecución de comandos mediante una versión de documento específica

Puede utilizar el parámetro document-version para especificar la versión de un documento de AWS Systems Manager que se va a usar cuando se ejecute el comando. Puede especificar una de las opciones siguientes para este parámetro:

- $DEFAULT
- $LATEST
- Número de versión

Lleve a cabo el siguiente procedimiento para ejecutar un comando utilizando el parámetro de versión del documento.

Linux

Para ejecutar comandos mediante la AWS CLI en equipos locales con Linux

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Enumerar todos los documentos disponibles

   Este comando enumera todos los documentos disponibles para su cuenta en función de los permisos de AWS Identity and Access Management (IAM).

   ```
   aws ssm list-documents
   ```

3. Ejecute el siguiente comando para ver las diferentes versiones de un documento. Reemplace el *nombre del documento* con su propia información.

   ```
   aws ssm list-document-versions \
       --name "document name"
   ```

4. Utilice el siguiente comando para ejecutar un comando que utilice una versión del documento de SSM. Reemplace cada *example resource placeholder* con su propia información.

   ```
   aws ssm send-command \
       --document-name "AWS-RunShellScript" \
       --parameters commands="echo Hello" \
       --instance-ids instance-ID \
       --document-version '$LATEST'
   ```

Windows

Para ejecutar comandos mediante la AWS CLI en equipos locales con Windows

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.
2. Enumerar todos los documentos disponibles

   Este comando enumera todos los documentos disponibles para su cuenta en función de los permisos de AWS Identity and Access Management (IAM).

   ```
   aws ssm list-documents
   ```
3. Ejecute el siguiente comando para ver las diferentes versiones de un documento. Reemplace el *nombre del documento* con su propia información.

   ```
   aws ssm list-document-versions ^
       --name "document name"
   ```
4. Utilice el siguiente comando para ejecutar un comando que utilice una versión del documento de SSM. Reemplace cada *example resource placeholder* con su propia información.

   ```
   aws ssm send-command ^
       --document-name "AWS-RunShellScript" ^
       --parameters commands="echo Hello" ^
       --instance-ids instance-ID ^
       --document-version "$LATEST"
   ```

PowerShell

Para ejecutar comandos con Tools for PowerShell

1. Instale y configure Herramientas de AWS para PowerShell (Herramientas para Windows PowerShell), si aún no lo ha hecho.

   Para obtener más información, consulte Instalación de Herramientas de AWS para PowerShell.
2. Enumerar todos los documentos disponibles

Este comando enumera todos los documentos disponibles para su cuenta en función de los permisos de AWS Identity and Access Management (IAM).

```
Get-SSMDocumentList
```

3. Ejecute el siguiente comando para ver las diferentes versiones de un documento. Reemplace el *nombre del documento* con su propia información.

```
Get-SSMDocumentVersionList `
    -Name "document name"
```

4. Utilice el siguiente comando para ejecutar un comando que utilice una versión del documento de SSM. Reemplace cada *example resource placeholder* con su propia información.

```
Send-SSMCommand `
    -DocumentName "AWS-RunShellScript" `
    -Parameter @{commands = "echo helloWorld"} `
    -InstanceIds "instance-ID" `
    -DocumentVersion $LATEST
```

Ejecución de comandos a escala

Puede utilizar Run Command, una herramienta de AWS Systems Manager, para ejecutar comandos en una flota de nodos administrados mediante `targets`. El parámetro `targets` acepta una combinación de `Key,Value` basada en las etiquetas que haya especificado en los nodos administrados. Al ejecutar el comando, el sistema localiza e intenta ejecutar dicho comando en todos los nodos administrados que coinciden con las etiquetas especificadas. Para obtener más información sobre el etiquetado de instancias administradas consulte, Etiquetado de recursos de AWS en la Guía del usuario de Etiquetado de recursos de AWS. Para obtener más información acerca del etiquetado de dispositivos IoT administrados, consulte Etiquetado de recursos de AWS IoT Greengrass Version 2 en la Guía para desarrolladores de AWS IoT Greengrass Version 2.

También puede utilizar el parámetro `targets` para dirigirse a una lista de ID de nodos administrados específicos, tal y como se describe en la sección siguiente.

Para controlar la forma en que los comandos ejecutan cientos o miles de nodos administrados, Run Command también incluye parámetros con el fin de restringir cuántos nodos puede procesar una solicitud simultáneamente y qué cantidad de errores puede generar un comando antes de finalizar.

Contenido

- Indicar destino de varios nodos administrados
- Uso de controles de velocidad

Indicar destino de varios nodos administrados

Puede ejecutar un comando y dirigirse a nodos administrados especificando etiquetas, nombres de grupos de recursos de AWS o ID de nodos administrados.

En los siguientes ejemplos, se muestra el formato de comandos cuando se utiliza Run Command desde la AWS Command Line Interface (AWS CLI). Reemplace cada *`example resource placeholder`* con su propia información. Los ejemplos de comandos de esta sección se truncan con `[...]`.

Ejemplo 1: dirigirse a etiquetas

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 ^
    [...]
```

Ejemplo 2: dirigirse a un grupo de recursos de AWS por nombre

Puede especificar un máximo de un nombre de grupo de recursos por comando. Al crear un grupo de recursos, le recomendamos que incluya `AWS::SSM:ManagedInstance` y `AWS::EC2::Instance` como tipos de recursos en sus criterios de creación de grupo.

> **Note**
>
> Con el fin de enviar comandos que tengan como destino un grupo de recursos, debe haber recibido los permisos de AWS Identity and Access Management (IAM) para mostrar o ver los recursos que pertenecen a ese grupo. Para obtener más información, consulte la sección Configuración de permisos en la Guía del usuario de 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 ^
    [...]
```

Ejemplo 3: dirigirse a un grupo de recursos de AWS por tipo de recurso

Puede especificar un máximo de cinco tipos de grupos de recursos por comando. Al crear un grupo de recursos, le recomendamos que incluya `AWS::SSM:ManagedInstance` y `AWS::EC2::Instance` como tipos de recursos en sus criterios de creación de grupo.

> **Note**
>
> Con el fin de enviar comandos que tienen como destino un grupo de recursos, se le deben haber concedido los permisos de IAM para mostrar o ver los recursos que pertenecen a ese grupo. Para obtener más información, consulte la sección Configuración de permisos en la Guía del usuario de 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 ^
    [...]
```

Ejemplo 4: dirigirse a ID de instancias

En los siguientes ejemplos, se muestra cómo dirigirse a nodos administrados mediante la clave de `instanceids` con el parámetro `targets`. Puede utilizar esta clave para dirigirse a dispositivos de núcleo de AWS IoT Greengrass administrados porque a cada dispositivo se le asigna un mi-*`Id_number`*. Puede ver los ID de los dispositivos en Fleet Manager, una herramienta 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 ha etiquetado nodos administrados para diferentes entornos utilizando `Environment` para `Key`, y `Development`, `Test`, `Pre-production` y `Production` para `Values`, entonces podría enviar un comando a todos los nodos administrados que se encuentren en uno de estos entornos utilizando el parámetro `targets` con la siguiente sintaxis.

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 ^
    [...]
```

Podría dirigirse a nodos administrados adicionales en otros entornos agregando elementos a la lista `Values`. Separe los elementos mediante comas.

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 ^
    [...]
```

Variación: refinado de los destinos con varios criterios `Key`

Puede refinar el número de destinos para el comando incluyendo varios criterios `Key`. Si incluye más de un criterio de `Key`, el sistema se dirige a los nodos administrados que cumplen todos los criterios. El siguiente comando se dirige a todos los nodos administrados etiquetados para el Departamento de Finanzas y etiquetados para el rol de servidor de base de datos.

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 ^
    [...]
```

Variación : uso de varios criterios `Key` y `Value`

Profundizando en el ejemplo anterior, puede dirigirse a varios departamentos y diversas roles de servidor mediante la inclusión de elementos adicionales en el criterio `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 ^
    [...]
```

Variación: dirigirse a nodos administrados etiquetados mediante varios criterios de `Values`

Si etiquetó nodos administrados para diferentes entornos utilizando una `Key` denominada `Department` y `Values` de `Sales` y `Finance`, podría enviar un comando a todos los nodos administrados de estos entornos mediante el uso del parámetro `targets` con la siguiente sintaxis.

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 ^
    [...]
```

Puede especificar un máximo de cinco claves y cinco valores para cada clave.

Si una clave de etiqueta (el nombre de la etiqueta) o un valor de etiqueta incluye espacios, deberá encerrar la clave de etiqueta o el valor entre comillas, como se muestra en los siguientes ejemplos.

Ejemplo: espacios en la etiqueta `Value`

Linux & macOS

```
aws ssm send-command \
    --document-name document-name \
    --targets Key=tag:OS,Values="Windows Server 2016" \
    [...]
```

Windows

```
aws ssm send-command ^
    --document-name document-name ^
    --targets Key=tag:OS,Values="Windows Server 2016" ^
    [...]
```

Ejemplo: espacios en la clave y `Value` de una `tag`

Linux & macOS

```
aws ssm send-command \
```

```
    --document-name document-name \
    --targets Key="tag:Operating System",Values="Windows Server 2016" \
    [...]
```

Windows

```
aws ssm send-command ^
    --document-name document-name ^
    --targets Key="tag:Operating System",Values="Windows Server 2016" ^
    [...]
```

Ejemplo: espacios en un elemento de una lista 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" ^
    [...]
```

Uso de controles de velocidad

Puede controlar la velocidad a la que se envían los comandos a los nodos administrados de un grupo mediante controles de simultaneidad y controles de error.

Temas

- Uso de controles de simultaneidad
- Uso de controles de error

Uso de controles de simultaneidad

Puede controlar el número de nodos administrados que ejecutan el comando al mismo tiempo mediante el parámetro `max-concurrency` (las opciones de Concurrency [Simultaneidad] de la página Run a command [Ejecutar un comando]). Puede especificar un número absoluto de nodos administrados, por ejemplo, **`10`**, o un porcentaje del destino definido, por ejemplo, **`10%`**. El sistema de colas entrega el comando a un único nodo y espera hasta que el sistema confirme la invocación inicial antes de enviar el comando a dos nodos más. El sistema envía comandos de forma exponencial a más nodos hasta que alcanza el valor de `max-concurrency`. El valor predeterminado del valor de `max-concurrency` es 50. Los siguientes ejemplos le muestran cómo especificar valores para el parámetro `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 ^
```

```
    [...]
```

Uso de controles de error

También puede controlar la ejecución de un comando para cientos o miles de nodos administrado al configurar un límite de error mediante los parámetros `max-errors` (el campo Error threshold [Umbral de error] de la página Run a command [Ejecutar un comando]). El parámetro especifica la cantidad de errores que están permitidos antes de que el sistema detenga el envío del comando a nodos administrados adicionales. Puede especificar un número absoluto de errores, por ejemplo, **`10`**, o un porcentaje del destino definido, por ejemplo, **`10%`**. Si especifica **`3`**, por ejemplo, el sistema dejará de enviar el comando cuando se reciba el cuarto error. Si especifica **`0`**, el sistema dejará de enviar el comando a otros nodos administrados tras el primer resultado de error que se devuelva. Si envía un comando a 50 nodos administrados y configura `max-errors` en un **`10%`**, el sistema dejará de enviar el comando a otros nodos cuando se reciba el sexto error.

Las invocaciones que ya están ejecutando un comando cuando se alcanza el `max-errors` tienen permiso para completarse, pero algunas de estas invocaciones también pueden producir errores. Si necesita asegurarse de que no habrá más de `max-errors` invocaciones erróneas, establezca `max-concurrency` en **`1`**, de modo que las invocaciones continuarán de una en una. El valor predeterminado para max-errors es 0. Los siguientes ejemplos le muestran cómo especificar valores para el parámetro `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 ^
    [...]
```

Cancelación de un comando

Puede intentar cancelar un comando siempre y cuando el servicio muestre que está en estado pendiente o en ejecución. Sin embargo, incluso si un comando todavía tiene alguno de estos estados, no se puede garantizar que el comando se cancelará y el proceso subyacente se detendrá.

Para cancelar un comando con la consola

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command.
3. Seleccione la invocación del comando que desea cancelar.
4. Elija Cancel Command.

Para cancelar un comando con la AWS CLI

Ejecute el siguiente comando de la . Reemplace cada *example resource placeholder* con su propia información.

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"
```

Para obtener más información acerca del estado de un comando cancelado, consulte Descripción de los estados del comando.

Uso de códigos de salida en los comandos

En algunos casos, es posible que necesite administrar cómo se gestionan los comandos mediante el uso de códigos de salida.

Especificación de códigos de salida en los comandos

Con Run Command, una herramienta de AWS Systems Manager, puede especificar códigos de salida para determinar cómo se gestionan los comandos. De forma predeterminada, el código de salida del último comando ejecutado en un script se registra como el código de salida de todo el script. Suponga, por ejemplo, que tiene un scripts que contiene tres comandos. El primero da un error, pero los demás se ejecutan correctamente. Como el comando final se ejecutó correctamente, el estado de la ejecución se registra como `succeeded`.

Scripts de shell

Para que todo el script produzca un error en el primer error del comando, puede incluir una declaración condicional de intérprete para salir del script si algún comando anterior al último produce un error. Utilice el siguiente enfoque.

```
<command 1>
```

```
    if [ $? != 0 ]
    then
        exit <N>
    fi
    <command 2>
    <command 3>
```

En el ejemplo siguiente, se produce un error en todo el script si se produce un error en el primer comando.

```
cd /test
    if [ $? != 0 ]
    then
        echo "Failed"
        exit 1
    fi
    date
```

Scripts de PowerShell

PowerShell requiere que llame explícitamente a `exit` en sus scripts para que Run Command capture correctamente el código de salida.

```
<command 1>
    if ($?) {<do something>}
    else {exit <N>}
    <command 2>
    <command 3>
    exit <N>
```

A continuación se muestra un ejemplo:

```
cd C:\
    if ($?) {echo "Success"}
    else {exit 1}
    date
```

Gestión de reinicios al ejecutar comandos

Si utiliza Run Command, una herramienta de AWS Systems Manager, para ejecutar scripts que reinician nodos administrados, le recomendamos que especifique un código de salida en el script. Si intenta utilizar algún otro mecanismo para reiniciar un nodo desde un script, el estado de ejecución

de ese script podría no actualizarse correctamente, aunque el reinicio sea el último paso del script. Para los nodos administrados de Windows, especifique `exit 3010` en el script. Para los nodos administrados de Linux y macOS, especifique `exit 194`. El código de salida indica al agente AWS Systems Manager (SSM Agent) que reinicie el nodo administrado y, a continuación, cuando esa operación haya finalizado, reinicie el script. Antes de comenzar el reinicio, el SSM Agent informará al servicio Systems Manager en la nube que la comunicación se va a interrumpir mientras se reinicia el servidor.

> **Note**
>
> El script de reinicio no puede formar parte de un complemento `aws:runDocument`. Si un documento contiene el script de reinicio y otro documento intenta ejecutarlo a través del complemento `aws:runDocument`, SSM Agent devuelve un error.

Creación de scripts idempotentes

Al desarrollar scripts que se utilizan para reiniciar nodos administrados, es importante que se asegure de que estos sean idempotentes para que, una vez finalizado el reinicio, la ejecución de los scripts continúe en el punto en que se encontraban anteriormente. Los scripts idempotentes administran el estado y validan si la acción se ha realizado o no. Esto impide que un paso se ejecute varias veces cuando solo está diseñado para ejecutarse una vez.

A continuación, se muestra un ejemplo resumido de un script idempotente que reinicia el nodo administrado varias veces.

```
$name = Get current computer name
If ($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
    }
```

Ejemplos

En los ejemplos de scripts siguientes, se utilizan códigos de salida para reiniciar nodos administrados. En el ejemplo de Linux, se instalan actualizaciones de paquetes en Amazon Linux y, a continuación, se reinicia el nodo. En el ejemplo de Windows Server, se instala Telnet-Client en el nodo y a continuación se reinicia.

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
    }
```

Descripción de los estados del comando

Run Command, una herramienta de AWS Systems Manager, ofrece información detallada sobre los diferentes estados que experimenta un comando durante el procesamiento y para cada nodo administrado que procesó el comando. Puede monitorear los estados del comando con los siguientes métodos:

- Seleccione el icono Refresh (actualizar) en la pestaña Commands (comandos) de la interfaz de la consola de Run Command.
- Llame a list-commands o list-command-invocations con la AWS Command Line Interface (AWS CLI). O llame a Get-SSMCommand o Get-SSMCommandInvocation con AWS Tools for Windows PowerShell.
- Configure Amazon EventBridge para que responda a los cambios de estado.
- Configure Amazon Simple Notification Service (Amazon SNS) para enviar notificaciones de todos los cambios de estado o de estados específicos, como `Failed` o `TimedOut`.

Estado de Run Command

Run Command notifica los detalles de estado de tres áreas: complementos, invocaciones y un estado general del comando. Un complemento es un bloque de ejecución de códigos definido en el documento de SSM del comando. Para obtener más información acerca de los complementos, consulte Referencia de complementos del documento de comandos.

Al enviar un comando para varios nodos administrados al mismo tiempo, cada copia del comando dirigida a cada nodo es una invocación de comandos. Por ejemplo, si utiliza el documento `AWS-RunShellScript` y envía un comando `ifconfig` a 20 instancias de Linux, dicho comando tendrá 20 invocaciones. Cada invocación de comandos notifica el estado individualmente. Los complementos para una invocación de comandos determinada también notifican el estado de forma individual.

Por último, Run Command contiene un estado agregado del comando para todos los complementos y las invocaciones. El estado agregado del comando puede ser diferente del estado notificado por los complementos o las invocaciones, como se describe en las siguientes tablas.

> **Note**
>
> Si ejecuta comandos para un gran número de nodos administrados utilizando los parámetros `max-concurrency` o `max-errors`, el estado del comando refleja las limitaciones impuestas por esos parámetros, tal y como se describe en las siguientes tablas. Para obtener más información sobre estos parámetros, consulte Ejecución de comandos a escala.

Estado detallado de los complementos y las invocaciones de comandos

Estado	Detalles
Pendiente	El comando aún no se ha enviado al nodo administrado o no ha sido recibido por el SSM Agent. Si el agente no recibe el comando antes de que pase el tiempo que es igual a la suma del parámetro Timeout (seconds) (Tiempo de espera [en segundos]) y del parámetro Execution timeout (Tiempo de espera de ejecución), el estado cambia a `Delivery Timed Out`.
InProgress	Systems Manager está intentando enviar el comando al nodo administrado, o el comando fue recibido por SSM Agent y ha comenzado a ejecutarse en la instancia. En función del resultado de todos los complemen tos del comando, el estado cambiará a `Success`, `Failed`, `Delivery Timed Out` o `Execution Timed Out`. Excepción: si el agente no está en ejecución o no está disponible en el nodo, el estado del comando permanece en `In Progress` hasta que el agente está disponible de nuevo o hasta que se alcanza el límite de tiempo de espera de ejecución. A continuación, el estado cambiará a un estado terminal.
Delayed	El sistema intentó enviar el comando al nodo administrado, pero no se envió correctamente. El sistema volverá a intentarlo.
Success	Este estado se devuelve en diversas condicion es. Este estado no significa que el comando se procesó en el nodo. Por ejemplo, el comando puede recibirlo SSM Agent en el nodo administr

Estado	Detalles
	ado y devolver un código de salida igual a cero como resultado de que la política `ExecutionPolicy` de PowerShell impide la ejecución del comando. Se trata de un estado terminal. Las condiciones que provocan que un comando devuelva el estado `Success` son las siguientes: • Al dirigirse a una única instancia, el comando lo recibió SSM Agent en el nodo administr ado y devolvió un código de salida igual a cero. • Al dirigirse a varias instancias, el número de invocaciones fallidas no ha superado el umbral de error especificado en el comando. • Al dirigirse a varias instancias, al menos una invocación ha funcionado correctam ente, mientras que las demás han agotado el tiempo de espera. El umbral de error especificado sigue siendo de aplicación. • Al dirigirse a una etiqueta, no se encuentra ninguna instancia asociada a la etiqueta. • Al dirigirse a una etiqueta, el número de invocaciones fallidas no ha superado el umbral de error especificado en el comando. • Al dirigirse a una etiqueta, al menos una invocación ha funcionado correctamente, mientras que las demás han agotado el tiempo de espera. El umbral de error especificado sigue siendo de aplicación. • Tiene aplicaciones o políticas aplicadas a nivel del SO que impiden o anulan la ejecución de un comando, lo que provoca

Estado	Detalles
	que se devuelva un código de salida igual a cero. **Note** Son de aplicación las mismas condiciones al dirigirse a grupos de recursos. Para solucionar los problemas con los errores u obtener más información acerca de la ejecución del comando, envíe un comando que administre los errores o las excepciones devolviendo códigos de salida adecuados (códigos de salida que no sean cero para los errores del comando).
DeliveryTimedOut	El comando no se entregó al nodo administrado antes de que se agotara el tiempo de espera total. Los tiempos de espera totales no cuentan para el límite de `max-errors` del comando principal, pero sí contribuyen a que el estado del comando principal sea `Success`, `Incomplete` o `Delivery Timed Out`. Se trata de un estado terminal.
ExecutionTimedOut	La automatización de comandos se inició en el nodo administrado, pero el comando no se completó antes de que se agotara el tiempo de espera de la ejecución. Agotar los tiempos de espera de las ejecuciones cuenta como un error, que enviará una respuesta distinta de cero, y Systems Manager dejará de intentar ejecutar la automatización de comandos e informará de un estado de error.

Estado	Detalles
Con error	El comando no se ejecutó correctamente en el nodo administrado. Para un complemento, esto indica que el código de resultado no era cero. Para una invocación de comando, esto indica que el código de resultado de uno o más complementos no era cero. Los errores de invocación cuentan para el `max-errors` límite del comando principal. Se trata de un estado terminal.
Cancelado	El comando se canceló antes de completarse. Se trata de un estado terminal.
Undeliverable	El comando no se puede entregar al nodo administrado. Puede que no exista el nodo o que no responda. Las invocaciones no disponibles para entrega no cuentan para el límite de `max-errors` del comando principal , pero sí contribuyen a que el estado del comando principal sea `Success` o `Incomplete` . Por ejemplo, si todas las invocaciones de un comando tienen el estado `Undeliverable` , el estado del comando que se devuelve es `Failed`. Sin embargo, si un comando tiene cinco invocaciones, cuatro de las cuales devuelven el estado `Undeliverable` y una devuelve el estado `Success`, el estado del comando principal será `Success`. Se trata de un estado terminal.
Rescindido	El comando principal supera su `max-errors` límite y el sistema ha cancelado las posteriores invocaciones de comandos. Se trata de un estado terminal.

Estado	Detalles
InvalidPlatform	El comando se envió a un nodo administrado que no coincidía con las plataformas necesarias especificadas por el documento seleccionado. `Invalid Platform` no cuenta para el límite de máximo de errores del comando principal, pero sí contribuye a que el estado del comando principal sea Success (Correcto) o Failed (Fallido). Por ejemplo, si todas las invocaciones de un comando tienen el estado `Invalid Platform`, el estado del comando que se devuelve es `Failed`. Sin embargo, si un comando tiene cinco invocaciones, cuatro de las cuales devuelven el estado `Invalid Platform` y una devuelve el estado `Success`, el estado del comando principal será `Success`. Se trata de un estado terminal.
AccessDenied	El usuario o el rol de AWS Identity and Access Management (IAM) que inicia el comando no tiene acceso al nodo administrado de destino. `Access Denied` no cuenta para el límite de `max-errors` del comando principal, pero sí contribuye a que el estado del comando principal sea `Success` o `Failed`. Por ejemplo, si todas las invocaciones de un comando tienen el estado `Access Denied`, el estado del comando que se devuelve es `Failed`. Sin embargo, si un comando tiene cinco invocaciones, cuatro de las cuales devuelven el estado `Access Denied` y una devuelve el estado `Success`, el estado del comando principal será `Success`. Se trata de un estado terminal.

Estado detallado de un comando

Estado	Detalles
Pendiente	Los agentes de los nodos administrados todavía no reciben el comando.
InProgress	El comando se ha enviado al menos a un nodo administrado, pero no ha alcanzado un estado definitivo en ninguno de los nodos.
Delayed	El sistema intentó enviar el comando al nodo, pero no se envió correctamente. El sistema volverá a intentarlo.
Success	El comando llegó al SSM Agent de todos los nodos administrados especificados o de destino y devolvió el código de salida cero. Todas las invocaciones de comandos han alcanzado un estado terminal y no se alcanzó el valor de `max-errors` . Este estado no significa que el comando se haya procesado correctamente en todos los nodos administr ados especificados o de destino. Se trata de un estado terminal. **Note** Para solucionar los problemas con los errores u obtener más información acerca de la ejecución del comando, envíe un comando que administre los errores o las excepciones devolviendo códigos de salida adecuados (códigos de salida que no sean cero para los errores del comando).

Estado	Detalles
DeliveryTimedOut	El comando no se entregó al nodo administrado antes de que se agotara el tiempo de espera total. El valor de `max-errors` o más invocaciones de comandos muestra el estado `Delivery Timed Out`. Se trata de un estado terminal.
Con error	El comando no se ejecutó correctamente en el nodo administrado. El valor de `max-errors` o más invocaciones de comandos muestra el estado `Failed`. Se trata de un estado terminal.
Incomplete	El comando se intentó en todos los nodos administrados y una o más de las invocaciones no tiene el valor `Success`. Sin embargo, no se ha producido error en suficientes invocaciones para que el estado sea `Failed`. Se trata de un estado terminal.
Cancelado	El comando se canceló antes de completarse. Se trata de un estado terminal.
RateExceeded	El número de nodos administrados de destino del comando ha superado la cuota de cuenta para las invocaciones pendientes. El sistema ha cancelado el comando antes de ejecutarlo en ningún nodo. Se trata de un estado terminal.

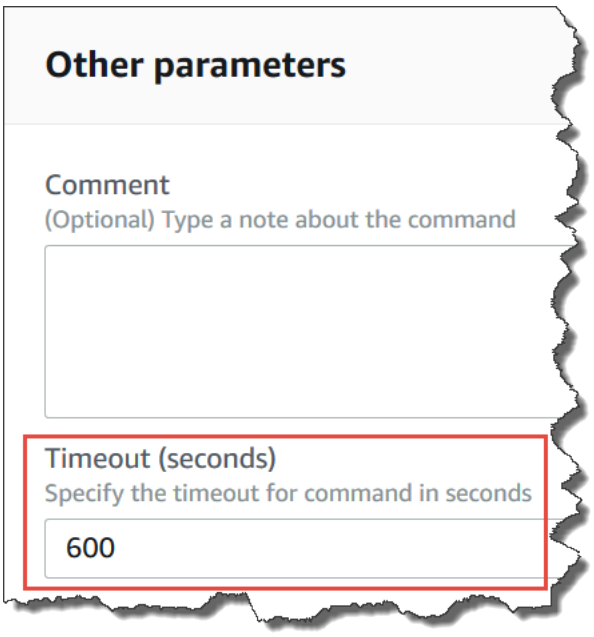
Estado	Detalles
AccessDenied	El usuario o el rol que inicia el comando no tiene acceso al grupo de recursos de destino. `AccessDenied` no cuenta para el límite de `max-errors` del comando principal, pero sí contribuye a que el estado del comando principal sea `Success` o `Failed`. (Por ejemplo, si todas las invocaciones de un comando tienen el estado `AccessDenied`, entonces el estado del comando que se devuelve es `Failed`. Sin embargo, si un comando tiene 5 invocaciones, 4 de las cuales devuelven el estado `AccessDenied` y 1 devuelve el estado `Success`, entonces el estado del comando principal será `Success`). Se trata de un estado terminal.
No hay instancias en la etiqueta	El grupo de recursos o el valor del par de claves de etiqueta seleccionado por el comando no coincide con ningún nodo administrado. Se trata de un estado terminal.

Descripción de los valores de tiempo de espera de los comandos

Systems Manager aplica los siguientes valores de tiempo de espera cuando ejecuta comandos.

Tiempo de espera total

En la consola de Systems Manager, especifique el valor del tiempo de espera en el campo Timeout (seconds) (Tiempo de espera [segundos]). Después de enviar un comando, Run Command verifica si el comando ha vencido o no. Si un comando alcanza el límite de vencimiento del comando (tiempo de espera total), cambia su estado a `DeliveryTimedOut` para todas las invocaciones que tienen el estado `InProgress`, `Pending` o `Delayed`.



En un nivel más técnico, el tiempo de espera total —Timeout (seconds) (Tiempo de espera [segundos])— es una combinación de dos valores de tiempo de espera, como se muestra aquí:

`Total timeout = "Timeout(seconds)" from the console + "timeoutSeconds": "{{ executionTimeout }}" from your SSM document`

Por ejemplo, el valor predeterminado de Timeout (seconds) (Tiempo de espera [en segundos]) en la consola de Systems Manager es de 600 segundos. Si ejecuta un comando mediante el documento `AWS-RunShellScript` de SSM, el valor predeterminado de "timeoutSeconds": "{{ executionTimeout }}" es de 3600 segundos, como se muestra en el siguiente ejemplo de documento:

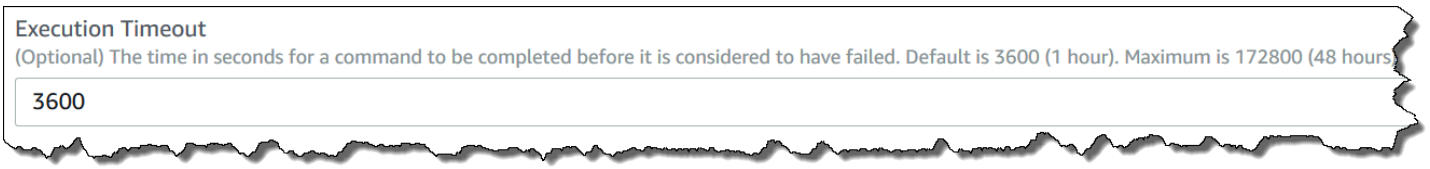
```
"executionTimeout": {
    "type": "String",
    "default": "3600",

"runtimeConfig": {
  "aws:runShellScript": {
    "properties": [
      {
        "timeoutSeconds": "{{ executionTimeout }}"
```

Esto significa que el comando se ejecuta durante 4200 segundos (70 minutos) antes de que el sistema establezca el estado del comando como `DeliveryTimedOut`.

Tiempo de espera de la ejecución

En la consola de Systems Manager, especifique el valor del tiempo de espera de la ejecución en el campo Execution Timeout (Tiempo de espera de ejecución), si está disponible. No todos los documentos de SSM requieren que especifique un tiempo de espera de ejecución. El campo Execution Timeout (Tiempo de espera hasta ejecución) solo se muestra cuando se ha definido un parámetro de entrada correspondiente en el documento de SSM. Si se especifica, el comando debe completarse dentro de este período de tiempo.

> **Note**
>
> Run Command se basa en la respuesta terminal del documento del SSM Agent para determinar si el comando se entregó al agente o no. SSM Agent debe enviar una señal de `ExecutionTimedOut` para que una invocación o un comando se marquen como `ExecutionTimedOut`.



Tiempo de espera de ejecución predeterminado

Si un documento de SSM no requiere que especifique explícitamente un valor de tiempo de espera de ejecución, entonces Systems Manager aplica el tiempo de espera de ejecución de codificación rígida predeterminado.

Cómo informa Systems Manager los tiempos de espera

Si Systems Manager recibe una respuesta de `execution timeout` del SSM Agent de un destino, Systems Manager marca la invocación del comando como `executionTimeout`.

Si Run Command no recibe una respuesta de terminal del documento de SSM Agent, la invocación del comando se marca como `deliveryTimeout`.

Para determinar el estado del tiempo de espera en un destino, SSM Agent combina todos los parámetros y el contenido del documento de SSM para calcular el `executionTimeout`. Cuando SSM Agent determina que se ha agotado el tiempo de espera para un comando, se envía `executionTimeout` al servicio.

El valor predeterminado de Timeout (seconds) (Tiempo de espera [en segundos]) es de 3600 segundos. El valor predeterminado de Execution Timeout (Tiempo de espera de ejecución) también es de 3600 segundos. Por lo tanto, el tiempo de espera predeterminado total de un comando es de 7200 segundos.

> **Note**
>
> SSM Agent procesa el `executionTimeout` de manera diferente según el tipo de documento de SSM y la versión de este último.

Tutoriales de Run Command

En los tutoriales de esta sección, se muestra cómo ejecutar comandos con Run Command, una herramienta de AWS Systems Manager, mediante la AWS Command Line Interface (AWS CLI) o AWS Tools for Windows PowerShell.

Contenido

- Actualización del software mediante Run Command
- Tutorial: uso de la AWS CLI con Run Command
- Tutorial: uso de la AWS Tools for Windows PowerShell con Run Command

También puede ver comandos de muestra en las siguientes referencias.

- Referencia de la AWS CLI de Systems Manager
- AWS Tools for Windows PowerShell - AWS Systems Manager

Actualización del software mediante Run Command

En los siguientes procedimientos se describe cómo actualizar el software en los nodos administrados.

Actualización de SSM Agent mediante Run Command

En el siguiente procedimiento, se describe cómo actualizar el SSM Agent que se ejecuta en los nodos administrados. Puede actualizar a la versión más reciente de SSM Agent o volver a una versión anterior. Cuando se ejecuta el comando, el sistema descarga la versión de AWS, la instala y,

a continuación, desinstala la versión que existía antes de ejecutar el comando. Si se produce un error durante este proceso, el sistema vuelve a la versión del servidor anterior a la ejecución del comando y el estado del comando mostrará que el comando ha tenido un error.

> **Note**
>
> Si una instancia ejecuta la versión 11.0 (Big Sur) o posterior de macOS, la instancia debe tener la versión 3.1.941.0 de SSM Agent o superior para ejecutar el documento de AWS-UpdateSSMAgent. Si la instancia ejecuta una versión de SSM Agent anterior a la 3.1.941.0, puede actualizar SSM Agent para ejecutar el documento de AWS-UpdateSSMAgent si ejecuta los comandos `brew update` y `brew upgrade amazon-ssm-agent`.

Si desea recibir notificaciones sobre actualizaciones de SSM Agent, suscríbase a la página de SSM Agent Release Notes en GitHub.

Para actualizar el SSM Agent con Run Command

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command.
3. Elija Run command (Ejecutar comando).
4. En la lista Command document (Documento de Command), elija **`AWS-UpdateSSMAgent`**.
5. En la sección Command Parameters, especifique los valores de los parámetros siguientes, si así lo desea:
   a. (Opcional) En Version (Versión), escriba la versión del SSM Agent que se va a instalar. Puede instalar versiones anteriores del agente. Si no especifica ninguna versión, el servicio instalará la más reciente.
   b. (Opcional). En Allow Downgrade (Permitir versiones anteriores), elija true para instalar una versión anterior del SSM Agent. Si elige esta opción, debe especificar el número de la versión anterior. Elija false para instalar solo la versión más reciente del servicio.
6. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

> **Tip**
>
> Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

7. En Otros parámetros:

   - En Comentario, ingrese la información acerca de este comando.
   - En Tiempo de espera (segundos), especifique el número de segundos que tiene que esperar el sistema antes de indicar que se ha producido un error en la ejecución del comando general.

8. En Rate control (Control de velocidad):

   - En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

     > **Note**
     >
     > Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

   - En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

9. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario

de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

10. En la sección Notificaciones de SNS, seleccione la casilla de verificación Habilitar notificaciones de SNS si desea recibir notificaciones sobre el estado de ejecución de los comandos.

    Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.

11. Seleccione Ejecutar.

Actualización de PowerShell con Run Command

En el siguiente procedimiento, se describe cómo actualizar PowerShell a la versión 5.1 en los nodos administrados de Windows Server 2012 y 2012 R2. El script proporcionado en este procedimiento descarga la actualización de Windows Management Framework (WMF) versión 5.1 e inicia la instalación de la actualización. El nodo se reinicia durante este proceso debido a que es necesario cuando se instala WMF 5.1. La descarga y la instalación de la actualización tardan aproximadamente cinco minutos en completarse.

Para actualizar PowerShell con Run Command

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command.
3. Elija Run command (Ejecutar comando).
4. En la lista Command document (Documento de Command), elija **`AWS-RunPowerShellScript`**.
5. En la sección Commands (Comandos), pegue los siguientes comandos para el sistema operativo.

   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. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

   > **Tip**
   >
   > Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

7. En Otros parámetros:

   - En Comentario, ingrese la información acerca de este comando.
   - En Tiempo de espera (segundos), especifique el número de segundos que tiene que esperar el sistema antes de indicar que se ha producido un error en la ejecución del comando general.

8. En Rate control (Control de velocidad):

   - En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

> **Note**
>
> Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

- En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

9. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

10. En la sección Notificaciones de SNS, seleccione la casilla de verificación Habilitar notificaciones de SNS si desea recibir notificaciones sobre el estado de ejecución de los comandos.

    Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.

11. Seleccione Ejecutar.

Una vez reiniciado el nodo administrado y finalizada la instalación de la actualización, conéctese al nodo para confirmar que PowerShell se actualizó de forma correcta a la versión 5.1. Para verificar la versión de PowerShell en el nodo, abra PowerShell e ingrese `$PSVersionTable`. El valor de `PSVersion` en la tabla de salida mostrará 5.1 si la actualización se realizó de manera correcta.

Si el valor de `PSVersion` no es 5.1, por ejemplo, 3.0 o 4.0, revise los registros de Setup (Configuración) en el lector de eventos de Windows Logs (Registros de Windows). Estos registros indicarán por qué falló la instalación de la actualización.

Tutorial: uso de la AWS CLI con Run Command

El siguiente ejemplo de explicación muestra cómo utilizar la AWS Command Line Interface (AWS CLI) para ver información acerca de los comandos y los parámetros de los comandos, de cómo ejecutar comandos y de cómo consultar el estado de dichos comandos.

> **Important**
>
> Solo los administradores de confianza deben utilizar los documentos de AWS Systems Manager preconfigurados que se muestran en este tema. Los comandos o los scripts especificados en los documentos de Systems Manager se ejecutan con permisos administrativos en los nodos administrados. Si un usuario tiene permiso para ejecutar cualquiera de los documentos de Systems Manager predefinidos (cualquier documento que empiece por `AWS-`), dicho usuario también tendrá acceso de administrador al nodo. Para todos los demás usuarios, debe crear documentos restrictivos y compartirlos con los usuarios específicos.

Temas

- Paso 1: introducción
- Paso 2: ejecutar scripts de shell para ver los detalles de los recursos
- Paso 3: enviar comandos simples utilizando el documento AWS-RunShellScript
- Paso 4: ejecutar una secuencia de comandos simple de Python mediante Run Command
- Paso 5: ejecutar un script de Bash utilizando Run Command

Paso 1: introducción

Debe tener permisos de administrador en el nodo administrado que desea configurar o se le deben haber otorgado los permisos adecuados en AWS Identity and Access Management (IAM). También

debe tener en cuenta que en este ejemplo se utiliza la región EE. UU. Este (Ohio) (us-east-2). Run Command está disponible en las Regiones de AWS que se indican en Puntos de enlace de servicio de Systems Manager en la Referencia general de Amazon Web Services. Para obtener más información, consulte Configuración de nodos administrados para AWS Systems Manager.

Para ejecutar comandos utilizando la AWS CLI

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.
2. Enumere todos los documentos disponibles.

   Este comando enumera todos los documentos disponibles para su cuenta en función de los permisos de IAM.

   ```
   aws ssm list-documents
   ```
3. Verifique si un nodo administrado está listo para recibir comandos.

   La salida del siguiente comando muestra si los nodos administrados están en línea.

   Linux & macOS

   ```
   aws ssm describe-instance-information \
       --output text --query "InstanceInformationList[*]"
   ```

   Windows

   ```
   aws ssm describe-instance-information ^
       --output text --query "InstanceInformationList[*]"
   ```
4. Ejecute el siguiente comando para ver los detalles sobre un nodo administrado en particular.

   > **Note**
   >
   > Para ejecutar los comandos de esta explicación, debe sustituir los ID de la instancia y del comando. Para dispositivos de núcleo de AWS IoT Greengrass administrados, utilice el mi-*ID_number* para el ID de instancia. El ID de comando se devuelve como

respuesta a send-command. Los ID de instancia están disponibles en Fleet Manager, una herramienta 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
```

Paso 2: ejecutar scripts de shell para ver los detalles de los recursos

Si utiliza Run Command y el documento `AWS-RunShellScript`, puede ejecutar cualquier comando o script en un nodo administrado como si hubiera iniciado sesión de manera local.

Ver la descripción y los parámetros disponibles

Ejecute el siguiente comando para ver una descripción del documento JSON de 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]"
```

Ejecute el siguiente comando para ver los parámetros disponibles y los detalles sobre esos parámetros.

Linux & macOS

```
aws ssm describe-document \
    --name "AWS-RunShellScript" \
    --query "Document.Parameters[*]"
```

Windows

```
aws ssm describe-document ^
    --name "AWS-RunShellScript" ^
    --query "Document.Parameters[*]"
```

Paso 3: enviar comandos simples utilizando el documento **AWS-RunShellScript**

Ejecute el siguiente comando para obtener la información de la dirección IP de un nodo administrado de Linux.

Si se dirige a un nodo administrado de Windows Server, cambie el `document-name` a `AWS-RunPowerShellScript` y cambie el `command` de `ifconfig` a `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
```

Obtener información de comandos con datos de respuesta

El comando siguiente utiliza el ID de comando que ha devuelto el comando anterior para obtener los detalles y los datos de respuesta de la ejecución de comandos. El sistema devuelve los datos de respuesta si el comando se completó. Si la ejecución del comando muestra `"Pending"` o `"InProgress"`, ejecute este comando de nuevo para ver los datos de respuesta.

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
```

Identificar usuario

El siguiente comando muestra el usuario predeterminado que ejecuta los comandos.

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")
```

Obtener el estado del comando

El comando siguiente utiliza el ID de comando para obtener el estado de la ejecución del comando en el nodo administrado. Este ejemplo utiliza el ID de comando devuelto en el comando anterior.

Linux & macOS

```
aws ssm list-commands \
```

```
    --command-id "command-ID"
```

Windows

```
aws ssm list-commands ^
    --command-id "command-ID"
```

Obtener los detalles del comando

El comando siguiente utiliza el ID de comando del comando anterior para obtener el estado de la ejecución del comando nodo por nodo.

Linux & macOS

```
aws ssm list-command-invocations \
    --command-id "command-ID" \
    --details
```

Windows

```
aws ssm list-command-invocations ^
    --command-id "command-ID" ^
    --details
```

Obtención de información de comando con los datos de respuesta de un nodo administrado concreto

El siguiente comando devuelve la salida de la solicitud original `aws ssm send-command` para un nodo administrado concreto.

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
```

Mostrar versión de Python

El siguiente comando devuelve la versión de Python que se ejecuta en un nodo.

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}"'
```

Paso 4: ejecutar una secuencia de comandos simple de Python mediante Run Command

El siguiente comando ejecuta un simple script de Python “Hello World” mediante 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}"'
```

Paso 5: ejecutar un script de Bash utilizando Run Command

En los ejemplos de esta sección, se muestra cómo ejecutar el siguiente script de bash utilizando Run Command.

Para ver ejemplos de cómo utilizar Run Command para ejecutar scripts almacenados en ubicaciones remotas, consulte Ejecución de scripts desde Amazon S3 y Ejecución de scripts desde 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
```

Este script instala el agente de AWS CodeDeploy en Amazon Linux y en las instancias de Red Hat Enterprise Linux (RHEL), tal como se describe en Crear una instancia de Amazon EC2 para CodeDeploy en la Guía del usuario de AWS CodeDeploy.

El script instala el agente de CodeDeploy desde un bucket de S3 administrado por AWS en la región Este de EE. UU. (Ohio) (us-east-2), `aws-codedeploy-us-east-2`.

Ejecución de un script de bash en un comando de la AWS CLI

En el ejemplo siguiente, se muestra cómo incluir el script de bash en un comando de la CLI mediante la opción `--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"]}'
```

Ejecutar un script de bash en un archivo JSON

En el ejemplo siguiente, el contenido del script de bash se almacena en un archivo JSON y el archivo se incluye en el comando mediante la opción `--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
```

El contenido del archivo `installCodeDeployAgent.json` al que se hace referencia se muestra en el ejemplo siguiente.

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

Tutorial: uso de la AWS Tools for Windows PowerShell con Run Command

Los siguientes ejemplos muestran cómo utilizar AWS Tools for Windows PowerShell para ver información sobre los comandos y los parámetros de comando, cómo ejecutar comandos y cómo consultar el estado de dichos comandos. Este tutorial incluye un ejemplo para cada uno de los documentos de AWS Systems Manager predefinidos.

> **Important**
>
> Solo los administradores de confianza deben tener permiso para utilizar los documentos preconfigurados de Systems Manager que se muestran en este tema. Los comandos o los scripts especificados en los documentos de Systems Manager se ejecutan con permisos administrativos en los nodos administrados. Si un usuario tiene permiso para ejecutar cualquiera de los documentos de Systems Manager predefinidos (cualquier documento que empiece con AWS), dicho usuario también tendrá acceso de administrador al nodo. Para todos los demás usuarios, debe crear documentos restrictivos y compartirlos con los usuarios específicos.

Temas

- Configurar las opciones de la sesión de AWS Tools for Windows PowerShell
- Enumerar todos los documentos disponibles
- Ejecutar comandos o scripts de PowerShell
- Instalar una aplicación utilizando el documento AWS-InstallApplication
- Instalar un módulo de PowerShell utilizando el documento JSON AWS-InstallPowerShellModule
- Unión de un nodo administrado a un dominio utilizando el documento JSON AWS-JoinDirectoryServiceDomain
- Enviar métricas de Windows a los Registros de Amazon CloudWatch mediante el documento AWS-ConfigureCloudWatch
- Actualizar EC2Config utilizando el documento AWS-UpdateEC2Config
- Activar o desactivar la actualización automática de Windows con el documento AWS-ConfigureWindowsUpdate
- Administrar las actualizaciones de Windows mediante Run Command

Configurar las opciones de la sesión de AWS Tools for Windows PowerShell

Especificar sus credenciales

Abra Tools for Windows PowerShell en su equipo local y ejecute el siguiente comando para especificar sus credenciales. Debe tener permisos de administrador en los nodos administrados que desea configurar o se le deben haber otorgado los permisos adecuados en AWS Identity and Access

Management (IAM). Para obtener más información, consulte Configuración de nodos administrados para AWS Systems Manager.

```
Set-AWSCredentials -AccessKey key-name -SecretKey key-name
```

Establezca una Región de AWS predeterminada.

Ejecute el siguiente comando para establecer la región de la sesión de PowerShell. En el ejemplo, se utiliza la región Este de EE. UU. (Ohio) (us-east-2). Run Command está disponible en las Regiones de AWS que se indican en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

```
Set-DefaultAWSRegion `
    -Region us-east-2
```

Enumerar todos los documentos disponibles

Este comando enumera todos los documentos disponibles para su cuenta.

```
Get-SSMDocumentList
```

Ejecutar comandos o scripts de PowerShell

Si utiliza Run Command y el documento `AWS-RunPowerShell`, puede ejecutar cualquier comando o script en un nodo administrado como si hubiera iniciado sesión de manera local. Puede emitir comandos o ingresar una ruta a un script local para ejecutar el comando.

> **Note**
>
> Para obtener información acerca de cómo reiniciar los nodos administrados cuando se utiliza Run Command para llamar a scripts, consulte Gestión de reinicios al ejecutar comandos.

Ver la descripción y los parámetros disponibles

```
Get-SSMDocumentDescription `
    -Name "AWS-RunPowerShellScript"
```

Ver más información sobre los parámetros

```
Get-SSMDocumentDescription `
    -Name "AWS-RunPowerShellScript" | Select -ExpandProperty Parameters
```

Enviar comandos utilizando el documento **`AWS-RunPowerShellScript`**

El siguiente comando muestra el contenido del directorio `"C:\Users"` y el contenido del directorio `"C:\"` en dos nodos administrados.

```
$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:\')}
```

Obtener los detalles de la solicitud del comando

El comando siguiente utiliza el `CommandId` para obtener el estado de la ejecución del comando en ambos nodos administrados. Este ejemplo utiliza el `CommandId` devuelto en el comando anterior.

```
Get-SSMCommand `
    -CommandId $runPSCommand.CommandId
```

El estado del comando en este ejemplo puede ser Success, Pending o InProgress.

Obtención de información de comandos por nodo administrado

El comando siguiente utiliza el `CommandId` del comando anterior para obtener el estado de la ejecución del comando nodo por nodo.

```
Get-SSMCommandInvocation `
    -CommandId $runPSCommand.CommandId
```

Obtención de información de comando con los datos de respuesta de un nodo administrado concreto

El siguiente comando devuelve la salida del `Send-SSMCommand` original para un nodo administrado concreto.

```
Get-SSMCommandInvocation `
```

```
    -CommandId $runPSCommand.CommandId `
    -Details $true `
    -InstanceId instance-ID | Select -ExpandProperty CommandPlugins
```

Cancelar un comando

El siguiente comando cancela el comando `Send-SSMCommand` para el documento `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
```

Comprobar el estado del comando

El comando siguiente comprueba el estado del comando `Cancel`.

```
Get-SSMCommand `
    -CommandId $cancelCommand.CommandId
```

Instalar una aplicación utilizando el documento **`AWS-InstallApplication`**

Si utiliza Run Command y el documento `AWS-InstallApplication`, puede instalar, reparar o desinstalar aplicaciones en nodos administrados. El comando requiere la ruta o dirección de un MSI.

> **Note**
>
> Para obtener información acerca de cómo reiniciar los nodos administrados cuando se utiliza Run Command para llamar a scripts, consulte Gestión de reinicios al ejecutar comandos.

Ver la descripción y los parámetros disponibles

```
Get-SSMDocumentDescription `
    -Name "AWS-InstallApplication"
```

Ver más información sobre los parámetros

```
Get-SSMDocumentDescription `
    -Name "AWS-InstallApplication" | Select -ExpandProperty Parameters
```

Enviar comandos utilizando el documento **AWS-InstallApplication**

El siguiente comando instala una versión de Python en el nodo administrado en modo desatendido y registra la salida en un archivo de texto local de la unidad `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'}
```

Obtención de información de comandos por nodo administrado

El comando siguiente utiliza el `CommandId` para obtener el estado de la ejecución del comando.

```
Get-SSMCommandInvocation `
    -CommandId $installAppCommand.CommandId `
    -Details $true
```

Obtención de información de comando con los datos de respuesta de un nodo administrado concreto

El siguiente comando devuelve el resultado de la instalación de Python.

```
Get-SSMCommandInvocation `
    -CommandId $installAppCommand.CommandId `
    -Details $true `
    -InstanceId instance-ID | Select -ExpandProperty CommandPlugins
```

Instalar un módulo de PowerShell utilizando el documento JSON **AWS-InstallPowerShellModule**

Puede utilizar Run Command para instalar módulos de PowerShell en nodos administrados. Para obtener más información acerca de los módulos de PowerShell, consulte Módulos de Windows PowerShell.

Ver la descripción y los parámetros disponibles

```
Get-SSMDocumentDescription `
    -Name "AWS-InstallPowerShellModule"
```

Ver más información sobre los parámetros

```
Get-SSMDocumentDescription `
    -Name "AWS-InstallPowerShellModule" | Select -ExpandProperty Parameters
```

Instalar un módulo de PowerShell

El siguiente comando descarga el archivo EZOut.zip, lo instala y, a continuación, ejecuta un comando adicional para instalar el visor de XPS. Por último, la salida de este comando se carga en un bucket de S3 denominado “amzn-s3-demo-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 amzn-s3-demo-bucket
```

Obtención de información de comandos por nodo administrado

El comando siguiente utiliza el `CommandId` para obtener el estado de la ejecución del comando.

```
Get-SSMCommandInvocation `
    -CommandId $installPSCommand.CommandId `
    -Details $true
```

Obtención de información de comandos con los datos de respuesta del nodo administrado

El siguiente comando devuelve el resultado del comando `Send-SSMCommand` original para el `CommandId` especificado.

```
Get-SSMCommandInvocation `
    -CommandId $installPSCommand.CommandId `
```

```
    -Details $true | Select -ExpandProperty CommandPlugins
```

Unión de un nodo administrado a un dominio utilizando el documento JSON **AWS-JoinDirectoryServiceDomain**

Si utiliza Run Command, puede unir rápidamente un nodo administrado a un dominio de AWS Directory Service. Antes de ejecutar este comando, cree un directorio. También le recomendamos obtener más información acerca de AWS Directory Service. Para obtener más información, consulte la Guía de administración de AWS Directory Service.

Solo puede unir un nodo administrado a un dominio. No se puede eliminar un nodo de un dominio.

> **Note**
>
> Para obtener información acerca de los nodos administrados cuando se utiliza Run Command para llamar a scripts, consulte Gestión de reinicios al ejecutar comandos.

Ver la descripción y los parámetros disponibles

```
Get-SSMDocumentDescription `
    -Name "AWS-JoinDirectoryServiceDomain"
```

Ver más información sobre los parámetros

```
Get-SSMDocumentDescription `
    -Name "AWS-JoinDirectoryServiceDomain" | Select -ExpandProperty Parameters
```

Unión de un nodo administrado a un dominio

El siguiente comando une un nodo administrado con el dominio AWS Directory Service dado y carga cualquier salida generada en el bucket de ejemplo de 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 amzn-s3-demo-bucket
```

Obtención de información de comandos por nodo administrado

El comando siguiente utiliza el `CommandId` para obtener el estado de la ejecución del comando.

```
Get-SSMCommandInvocation `
    -CommandId $domainJoinCommand.CommandId `
    -Details $true
```

Obtención de información de comandos con los datos de respuesta del nodo administrado

Este comando devuelve la salida del comando `Send-SSMCommand` original para el `CommandId` especificado.

```
Get-SSMCommandInvocation `
    -CommandId $domainJoinCommand.CommandId `
    -Details $true | Select -ExpandProperty CommandPlugins
```

Enviar métricas de Windows a los Registros de Amazon CloudWatch mediante el documento **`AWS-ConfigureCloudWatch`**

Puede enviar mensajes de Windows Server en los registros de aplicación, sistema, seguridad y Seguimiento de eventos para Windows (ETW) a los Registros de Amazon CloudWatch. Cuando se permite el registro por primera vez, Systems Manager envía todos los registros generados en un plazo de un (1) minuto a partir del momento en que empieza a cargar registros para los registros de aplicación, sistema, seguridad y ETW. No se incluyen los registros que se produjeron antes de este tiempo. Si desactiva el registro y después vuelve a activarlo, Systems Manager envía registros desde el momento en que dejó de hacerlo. En el caso de cualquier archivo de registros personalizado y registro de Internet Information Services (IIS), Systems Manager lee los archivos de registros desde el principio. Además, Systems Manager también puede enviar datos del contador de rendimiento a los Registros de CloudWatch.

Si con anterioridad activó la integración de CloudWatch en EC2Config, la configuración de Systems Manager anulará cualquier configuración almacenada de forma local en el nodo administrado en el archivo `C:\Program Files\Amazon\EC2ConfigService\Settings\AWS.EC2.Windows.CloudWatch.json`. Para obtener información sobre cómo se utiliza EC2Config para administrar los contadores de rendimiento y los registros en un solo nodo administrado, consulte Recopilación de métricas y registros de instancias de Amazon EC2 y servidores locales con el agente de CloudWatch en la Guía del usuario de Amazon CloudWatch.

Ver la descripción y los parámetros disponibles

```
Get-SSMDocumentDescription `
    -Name "AWS-ConfigureCloudWatch"
```

Ver más información sobre los parámetros

```
Get-SSMDocumentDescription `
    -Name "AWS-ConfigureCloudWatch" | Select -ExpandProperty Parameters
```

Enviar registros de aplicación a CloudWatch

El siguiente comando configura el nodo administrado y mueve los registros de aplicaciones de Windows a 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"]}}}'}
```

Obtención de información de comandos por nodo administrado

El comando siguiente utiliza el `CommandId` para obtener el estado de la ejecución del comando.

```
Get-SSMCommandInvocation `
    -CommandId $cloudWatchCommand.CommandId `
    -Details $true
```

Obtención de información de comando con los datos de respuesta de un nodo administrado concreto

El siguiente comando devuelve los resultados de la configuración de Amazon CloudWatch.

```
Get-SSMCommandInvocation `
    -CommandId $cloudWatchCommand.CommandId `
    -Details $true `
    -InstanceId instance-ID | Select -ExpandProperty CommandPlugins
```

Enviar contadores de rendimiento a CloudWatch utilizando el documento **AWS-ConfigureCloudWatch**

El siguiente comando de demostración carga contadores de desempeño en CloudWatch. Para obtener más información, consulte la Guía del usuario de Amazon CloudWatch.

```
$cloudWatchMetricsCommand = Send-SSMCommand `
    -InstanceID instance-ID `
    -DocumentName "AWS-ConfigureCloudWatch" `
    -Parameter @{'properties'='{"engineConfiguration": {"PollInterval":"00:00:15",
 "Components":[{"Id":"PerformanceCounter",
 "FullName":"AWS.EC2.Windows.CloudWatch.PerformanceCounterComponent.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"]}}}'}
```

Actualizar EC2Config utilizando el documento **AWS-UpdateEC2Config**

Si utiliza Run Command y el documento `AWS-EC2ConfigUpdate`, puede actualizar el servicio EC2Config que se ejecuta en los nodos administrados de Windows Server. Este comando puede actualizar el servicio EC2Config a la versión más reciente o a la versión que especifique.

Ver la descripción y los parámetros disponibles

```
Get-SSMDocumentDescription `
    -Name "AWS-UpdateEC2Config"
```

Ver más información sobre los parámetros

```
Get-SSMDocumentDescription `
    -Name "AWS-UpdateEC2Config" | Select -ExpandProperty Parameters
```

Actualizar EC2Config a la versión más reciente

```
$ec2ConfigCommand = Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-UpdateEC2Config"
```

Obtención de información de comandos con los datos de respuesta del nodo administrado

Este comando devuelve la salida del comando especificado del comando anterior `Send-SSMCommand`.

```
Get-SSMCommandInvocation `
    -CommandId $ec2ConfigCommand.CommandId `
    -Details $true `
    -InstanceId instance-ID | Select -ExpandProperty CommandPlugins
```

Actualizar EC2Config a una versión específica

El siguiente comando degradará EC2Config a una versión anterior.

```
Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-UpdateEC2Config" `
    -Parameter @{'version'='4.9.3519'; 'allowDowngrade'='true'}
```

Activar o desactivar la actualización automática de Windows con el documento **`AWS-ConfigureWindowsUpdate`**

Si utiliza Run Command y el documento `AWS-ConfigureWindowsUpdate`, puede activar o desactivar las actualizaciones automáticas de Windows en los nodos administrados de Windows Server. Este comando configura el agente de Windows Update para que descargue e instale las actualizaciones de Windows el día y a la hora que usted especifique. Si una actualización requiere un reinicio, el nodo administrado se reiniciará automáticamente 15 minutos después de haber instalado las actualizaciones. Con este comando también puede configurar Windows Update para que verifique si hay actualizaciones, pero sin instalarlas. El documento `AWS-ConfigureWindowsUpdate` es compatible con Windows Server 2008, 2008 R2, 2012, 2012 R2 y 2016.

Ver la descripción y los parámetros disponibles

```
Get-SSMDocumentDescription `
    -Name "AWS-ConfigureWindowsUpdate"
```

Ver más información sobre los parámetros

```
Get-SSMDocumentDescription `
```

```
    -Name "AWS-ConfigureWindowsUpdate" | Select -ExpandProperty Parameters
```

Activar la actualización automática de Windows

El siguiente comando configura Windows Update para que descargue e instale las actualizaciones automáticamente todos los días a las 10:00 h.

```
$configureWindowsUpdateCommand = Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-ConfigureWindowsUpdate" `
    -Parameters @{'updateLevel'='InstallUpdatesAutomatically';
 'scheduledInstallDay'='Daily'; 'scheduledInstallTime'='22:00'}
```

Ver el estado del comando para permitir la actualización automática de Windows

El comando siguiente utiliza el `CommandId` para obtener el estado de la ejecución del comando para permitir la actualización automática de Windows.

```
Get-SSMCommandInvocation `
    -Details $true `
    -CommandId $configureWindowsUpdateCommand.CommandId | Select -ExpandProperty
 CommandPlugins
```

Desactivar la actualización automática de Windows

El comando siguiente reduce el nivel de notificación de Windows Update para que el sistema verifique si hay actualizaciones, pero no actualice automáticamente el nodo administrado.

```
$configureWindowsUpdateCommand = Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-ConfigureWindowsUpdate" `
    -Parameters @{'updateLevel'='NeverCheckForUpdates'}
```

Ver el estado del comando para desactivar la actualización automática de Windows

El comando siguiente utiliza el `CommandId` para obtener el estado de la ejecución del comando para desactivar la actualización automática de Windows.

```
Get-SSMCommandInvocation `
    -Details $true `
```

```
    -CommandId $configureWindowsUpdateCommand.CommandId | Select -ExpandProperty
CommandPlugins
```

Administrar las actualizaciones de Windows mediante Run Command

Mediante el uso de Run Command y el documento `AWS-InstallWindowsUpdates`, puede administrar actualizaciones para nodos administrados de Windows Server. Este comando busca o instala las actualizaciones que faltan en los nodos administrados y, de forma opcional, se reinicia después de la instalación. También puede especificar las clasificaciones y los niveles de gravedad adecuados para las actualizaciones que se van a instalar en su entorno.

> **Note**
>
> Para obtener información acerca de cómo reiniciar los nodos administrados cuando se utiliza Run Command para llamar a scripts, consulte Gestión de reinicios al ejecutar comandos.

Los siguientes ejemplos muestran cómo realizar las tareas de administración de Windows Update especificadas.

Buscar todas las actualizaciones de Windows que faltan

```
Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-InstallWindowsUpdates" `
    -Parameters @{'Action'='Scan'}
```

Instalar actualizaciones de Windows específicas

```
Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-InstallWindowsUpdates" `
    -Parameters @{'Action'='Install';'IncludeKbs'='kb-ID-1,kb-ID-2,kb-
ID-3';'AllowReboot'='True'}
```

Instalar las actualizaciones importantes de Windows que faltan

```
Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-InstallWindowsUpdates" `
```

```
    -Parameters @{'Action'='Install';'SeverityLevels'='Important';'AllowReboot'='True'}
```

Instalar las actualizaciones de Windows que faltan con exclusiones específicas

```
Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-InstallWindowsUpdates" `
    -Parameters @{'Action'='Install';'ExcludeKbs'='kb-ID-1,kb-
ID-2';'AllowReboot'='True'}
```

Solución de problemas Systems Manager Run Command

Run Command, una herramienta de AWS Systems Manager, proporciona detalles de estado con la ejecución de cada comando. Para obtener más información acerca de los detalles de los estados del comando, consulte Descripción de los estados del comando. También puede utilizar la información de este tema como ayuda para solucionar problemas con Run Command.

Temas

- Faltan algunos de los nodos administrados
- Un paso de mi script produjo un error, pero el estado general es “correcto”
- SSM Agent no se ejecuta correctamente.

Faltan algunos de los nodos administrados

En la página Run a command (Ejecutar un comando), después de elegir el documento de SSM que se va a ejecutar y de seleccionar Manually selecting instances (Seleccionar instancias manualmente) en la sección Targets (Destinos), se muestra una lista de los nodos administrados que puede elegir para ejecutar el comando.

Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

Después de crear, activar o reiniciar un nodo administrado, instale Run Command en un nodo o adjunte un perfil de instancias de AWS Identity and Access Management (IAM) a un nodo. Pueden pasar algunos minutos hasta que el nodo administrado aparezca en la lista.

Un paso de mi script produjo un error, pero el estado general es “correcto”

Mediante el uso de Run Command, puede definir cómo los scripts manejan los códigos de salida. De forma predeterminada, el código de salida del último comando ejecutado en un script se registra

como el código de salida de todo el script. Sin embargo, puede incluir una instrucción condicional para salir del script si algún comando anterior al final produce un error. Para obtener más información y ejemplos, consulte Especificación de códigos de salida en los comandos.

SSM Agent no se ejecuta correctamente.

Si tiene dificultades cuando ejecute comandos con Run Command, es posible que haya algún problema con SSM Agent. Para obtener más información acerca de cómo investigar problemas con SSM Agent, consulte Solución de problemas de SSM Agent.

AWS Systems Manager Session Manager

Session Manager es una herramienta de AWS Systems Manager completamente administrada. Con Session Manager, puede administrar las instancias de Amazon Elastic Compute Cloud (Amazon EC2), los dispositivos periféricos, los servidores en las instalaciones y las máquinas virtuales (VM). Puede utilizar un intérprete de comandos interactivo basado en navegador con un solo clic o la AWS Command Line Interface (AWS CLI). Session Manager proporciona una administración de nodos segura sin la necesidad de abrir los puertos de entrada, mantener hosts bastión ni administrar claves SSH. Session Manager también permite el cumplimiento con las políticas corporativas que requieren acceso controlado a nodos administrados, prácticas de seguridad estrictas y registros con detalles del acceso a los nodos, a la vez que ofrecen a los usuarios finales un acceso multiplataforma sencillo con un solo clic a los nodos administrados. Para comenzar a utilizar Session Manager, abra la consola de Systems Manager. En el panel de navegación, elija Session Manager.

¿Cómo puede Session Manager beneficiar a mi organización?

Session Manager ofrece las ventajas siguientes:

- Control centralizado del acceso a los nodos administrados mediante políticas de IAM

  Los administradores disponen de un solo lugar para conceder y denegar el acceso a los nodos administrados. Si utiliza solo políticas de AWS Identity and Access Management (IAM), puede controlar qué usuarios individuales o grupos de la organización pueden utilizar Session Manager y a qué nodos administrados pueden acceder.
- Sin puertos de entrada abiertos y sin necesidad de administrar hosts bastión o claves SSH

  Dejar abiertos los puertos SSH de entrada y los puertos PowerShell remotos en los nodos administrados aumenta enormemente el riesgo de que las entidades ejecuten comandos no autorizados o maliciosos en los nodos administrados. Session Manager ayuda a mejorar la

posición de seguridad y permite cerrar estos puertos entrantes, por lo que no tendrá que realizar tareas de administración de claves y certificados SSH, hosts bastión y cajas de conexiones.

- Acceso con un solo clic a los nodos administrados desde la consola y la CLI

  Si usa la consola de AWS Systems Manager o de Amazon EC2, podrá iniciar una sesión con un solo clic. Con la AWS CLI, también puede iniciar una sesión que ejecute un único comando o una secuencia de comandos. Dado que los permisos a nodos administrados se proporcionan a través de las políticas de IAM en lugar de con claves SSH u otros mecanismos, el tiempo de conexión se reduce significativamente.

- Conexión a instancias de Amazon EC2 y a nodos administrados que no sean de EC2 en entornos híbridos y multinube

  Puede conectarse a instancias de Amazon Elastic Compute Cloud (Amazon EC2) y a nodos que no sean de EC2 en su entorno híbrido y multinube.

  Para conectarse a nodos que no sean de EC2 con Session Manager, primero debe activar el nivel de instancias avanzadas. El uso del nivel de instancias avanzadas conlleva un cargo. No obstante, no se generan cargos adicionales por la conexión a instancias de EC2 mediante Session Manager. Para obtener más información, consulta Configuración de los niveles de instancias.

- Enrutamiento de puertos

  Redirija cualquier puerto dentro del nodo administrado a un puerto local de un cliente. Después, conéctese al puerto local y obtenga acceso a la aplicación de servidor que se está ejecutando dentro del nodo.

- Compatibilidad entre plataformas para Windows, Linux y macOS

  Session Manager proporciona compatibilidad con Windows, Linux y macOS desde una sola herramienta. Por ejemplo, no es necesario utilizar un cliente de SSH para los nodos administrados de Linux ni de macOS, ni tampoco una conexión de RDP para los nodos administrados de Windows Server.

- Registro de la actividad de la sesión

  Para cumplir los requisitos de seguridad y operativos de la organización, es posible que tenga que proporcionar un registro de las conexiones realizadas en los nodos administrados y los comandos que se ejecutaron en ellos. También puede recibir notificaciones cuando un usuario de su organización comienza o finaliza la actividad de sesiones.

Las capacidades de registro se proporcionan a través de la integración a los siguientes Servicios de AWS:

- AWS CloudTrail: AWS CloudTrail recopila información acerca de las llamadas a la API de Session Manager realizadas en su Cuenta de AWS y la escribe en archivos de registros que se almacenan en un bucket de Amazon Simple Storage Service (Amazon S3) que usted especifique. Se utiliza un bucket para todos los registros de CloudTrail de su cuenta. Para obtener más información, consulte Registro de llamadas a la API de AWS Systems Manager con AWS CloudTrail.
- Amazon Simple Storage Service: puede elegir almacenar los datos de registro de las sesiones en un bucket de Amazon S3 que prefiera con fines de depuración y solución de problemas. Los datos de registro se pueden enviar al bucket de Amazon S3 con o sin cifrado mediante su AWS KMS key. Para obtener más información, consulte Registro de los datos de la sesión con Amazon S3 (consola).
- Registros de Amazon CloudWatch: Registros de CloudWatch le permite acceder a los archivos de registros de diversos Servicios de AWS, supervisarlos y almacenarlos. Puede enviar los datos de registro de las sesiones a un grupo de registros de los Registros de CloudWatch con fines de depuración y solución de problemas. Los datos de registro se pueden enviar al grupo de registros con o sin cifrado de AWS KMS usando su clave de KMS. Para obtener más información, consulte Registro de los datos de la sesión con los Registros de Amazon CloudWatch (consola).
- Amazon EventBridge y Amazon Simple Notification Service: EventBridge le permite configurar reglas para detectar cuándo se producen cambios en los recursos de AWS que especifique. Puede crear una regla para detectar cuándo un usuario de su organización inicia o detiene una sesión y, luego, recibir una notificación a través de Amazon SNS (por ejemplo, un mensaje de texto o de email) sobre el evento. También puede configurar un evento de CloudWatch para iniciar otras respuestas. Para obtener más información, consulte Supervisión de la actividad de la sesión con Amazon EventBridge (consola).

> **Note**
>
> El registro no está disponible para las sesiones de Session Manager que se conectan a través del reenvío de puertos o de SSH. Esto se debe a que SSH cifra todos los datos de la sesión y Session Manager solo sirve como túnel para las conexiones de SSH.

¿Quién debe utilizar Session Manager?

- Cualquier cliente de AWS que quiera mejorar su posición de seguridad, reducir los costos operativos mediante la centralización del control de accesos de los nodos administrados y reducir el acceso de entrada de los nodos.
- Expertos en seguridad de la información que deseen monitorear y realizar un seguimiento de la actividad de los nodos administrados y del acceso a ellos, cerrar los puertos de entrada de los nodos administrados o permitir las conexiones a nodos administrados que no tengan una dirección IP pública.
- Administradores que deseen conceder y revocar acceso desde un solo lugar y que deseen proporcionar una solución a los usuarios para los nodos administrados de Linux, macOS y Windows Server.
- Usuarios que deseen conectarse a un nodo administrado con un solo clic desde el navegador o la AWS CLI sin tener que proporcionar claves de SSH.

¿Cuáles son las características principales de Session Manager?

- Compatibilidad con nodos administrados de Windows Server, Linux y macOS

  Session Manager le permite establecer conexiones seguras con las instancias de Amazon Elastic Compute Cloud (EC2), los dispositivos periféricos, los servidores en las instalaciones y las máquinas virtuales (VM). Para ver una lista de los tipos de sistemas operativos admitidos, consulte Configuración de Session Manager.

  > **Note**
  >
  > La compatibilidad de Session Manager con máquinas locales se proporciona solo para el nivel de instancias avanzadas. Para obtener más información, consulta Activación del nivel de instancias avanzadas.

- Acceso de la consola, la CLI y el SDK a las funcionalidades de Session Manager

  Puede trabajar con Session Manager de las siguientes formas:

  La consola de AWS Systems Manager incluye el acceso a todas las capacidades de Session Manager, tanto para los administradores como para los usuarios finales. Puede llevar a cabo cualquier tarea relacionada con las sesiones por medio de la consola de Systems Manager.

La consola de Amazon EC2 proporciona a los usuarios finales la posibilidad de conectarse a instancias de EC2 para las cuales se les han concedido permisos de sesión.

La AWS CLI incluye acceso a las funcionalidades de Session Manager para usuarios finales. Puede comenzar una sesión, ver una lista de sesiones y terminar una sesión de forma permanente a través de la AWS CLI.

> **Note**
>
> Si desea utilizar la AWS CLI para ejecutar comandos de sesión, debe utilizar la versión 1.16.12 (o una posterior) de la CLI y tener instalado el complemento de Session Manager en su equipo local. Para obtener más información, consulta Instalación del complemento de Session Manager para la AWS CLI. Para ver el complemento en GitHub, consulte session-manager-plugin.

- Control de acceso de IAM

  Mediante el uso de políticas de IAM, puede controlar qué miembros de la organización pueden iniciar sesiones en nodos administrados y a qué nodos pueden acceder. También puede proporcionar acceso temporal a los nodos administrados. Por ejemplo, es posible que desee ofrecer a un ingeniero de guardia (o un grupo de ingenieros de guardia) el acceso a servidores de producción solo para la duración de su turno.

- Compatibilidad de registro

  Session Manager proporciona opciones para los historiales de las sesiones de registro en su Cuenta de AWS a través de la integración a una serie de otros Servicios de AWS. Para obtener más información, consulte Registro de la actividad de la sesión y Habilitar y deshabilitar el registro de la sesión.

- Perfiles configurables de shell

  Session Manager le ofrece opciones para configurar preferencias dentro de las sesiones. Estos perfiles personalizables le permiten definir preferencias, como las preferencias de shell, las variables de entorno, los directorios de trabajo y la ejecución de varios comandos cuando se inicia una sesión.

- Compatibilidad con el cifrado de datos clave del cliente

Puede configurar Session Manager para que cifre los registros de datos de las sesiones que se envían a un bucket de Amazon Simple Storage Service (Amazon S3) o se transmiten a un grupo de Registros de CloudWatch. También puede configurar Session Manager para cifrar aún más los datos transmitidos entre equipos del cliente y los nodos administrados durante las sesiones. Para obtener información, consulte Habilitar y deshabilitar el registro de la sesión y Configurar preferencias de sesión.

- Compatibilidad de AWS PrivateLink para nodos administrados sin direcciones IP públicas

  También puede configurar los puntos de conexión de VPC para Systems Manager mediante AWS PrivateLink para asegurar aún más las sesiones. AWS PrivateLink limita todo el tráfico de red entre los nodos administrados, Systems Manager y Amazon EC2 a la red de Amazon. Para obtener más información, consulte Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager.

- Tunelización

  En una sesión, utilice un documento de AWS Systems Manager (SSM) de tipo sesión para tunelizar el tráfico, como http o un protocolo personalizado, entre un puerto local en un equipo cliente y un puerto remoto en un nodo administrado.

- Comandos interactivos

  Cree un documento SSM de tipo sesión que utilice una sesión para ejecutar de forma interactiva un único comando, lo que le ofrece una forma de administrar lo que los usuarios pueden hacer en un nodo administrado.

¿Qué es una sesión?

Una sesión es una conexión a un nodo administrado efectuada con Session Manager. Las sesiones se basan en un canal de comunicación bidireccional seguro entre el cliente (usted) y el nodo administrado remoto que transmite elementos de entrada y salida para los comandos. El tráfico entre un cliente y un nodo administrado se cifra con TLS 1.2, y las solicitudes para crear la conexión se firman con Sigv4. Esta comunicación bidireccional permite el acceso interactivo de bash y PowerShell a los nodos administrados. También puede utilizar una clave de AWS Key Management Service (AWS KMS) para cifrar aún más los datos, más allá del cifrado TLS predeterminado.

Por ejemplo, digamos que John es un ingeniero de guardia en su departamento de TI. Este recibe la notificación de un problema que le exige conectarse remotamente a un nodo administrado, como, por ejemplo, un error que requiere la solución de problemas o una directiva para cambiar una opción

de configuración sencilla en un nodo. Mediante la consola de AWS Systems Manager, la consola de Amazon EC2 o la AWS CLI, John inicia una sesión que lo conecta al nodo administrado, ejecuta comandos en el nodo necesarios para completar la tarea y, a continuación, termina la sesión.

Cuando John envía ese primer comando para iniciar la sesión, el servicio Session Manager autentica su ID, verifica los permisos concedidos por una política de IAM, comprueba las opciones de configuración (como, por ejemplo, comprobando los límites permitidos para las sesiones) y envía un mensaje a SSM Agent para abrir la conexión bidireccional. Una vez establecida la conexión y después de que John escribe el siguiente comando, la salida del comando de SSM Agent se carga en este canal de comunicación y se envía a su máquina local.

Temas

- Configuración de Session Manager
- Uso de Session Manager
- Registro de la actividad de la sesión
- Habilitar y deshabilitar el registro de la sesión
- Esquema del documento de Session
- Solución de problemas de Session Manager

Configuración de Session Manager

Antes de usar AWS Systems Manager Session Manager para conectarse a los nodos administrados de la cuenta, realice los pasos en los siguientes temas.

Temas

- Paso 1: completar los requisitos previos de Session Manager
- Paso 2: verificación o agregación de permisos de instancia para Session Manager
- Paso 3: controlar el acceso de la sesión a los nodos administrados
- Paso 4: configurar las preferencias de sesión
- Paso 5: (Opcional) Restringir el acceso a los comandos de una sesión
- Paso 6: (Opcional) Utilizar AWS PrivateLink para configurar un punto de enlace de la VPC para Session Manager
- Paso 7: (Opcional) Activar o desactivar los permisos administrativos de la cuenta ssm-user
- Paso 8: (Opcional) Permitir y controlar permisos para conexiones de SSH mediante Session Manager

Paso 1: completar los requisitos previos de Session Manager

Antes de utilizar Session Manager, asegúrese de que el entorno cumple los siguientes requisitos.

Requisitos previos de Session Manager

Requisito	Descripción
Sistemas operativos compatibles	Session Manager admite la conexión a instancias de Amazon Elastic Compute Cloud (Amazon EC2) y a equipos que no son de EC2 en su entorno híbrido y multinube que utilizan el nivel de instancias avanzadas. Session Manager es compatible con las siguientes versiones de los sistemas operativos: **Note** Session Manager admite instancias de EC2, dispositivos periféricos, servidores en las instalaciones y máquinas virtuales (VM) en su entorno híbrido y multinube que utilizan el nivel de instancias avanzadas. Para obtener más información acerca de las instancias avanzadas, consulte Configuración de los niveles de instancias. Linux y macOS Session Manager es compatible con todas las versiones de Linux y macOS que admite AWS Systems Manager. Para obtener más información, consulta Sistemas operativos y tipos de equipos compatibles.

Requisito	Descripción
	Windows Session Manager es compatible con las versiones de Windows Server de 2012 hasta las versiones de Windows Server de 2022. **Note** Microsoft Windows Server 2016 Nano no es compatible.

Requisito	Descripción
SSM Agent	Como mínimo, debe estar instalada la versión 2.3.68.0 de AWS Systems Manager SSM Agent o una posterior en los nodos administrados a los que desee conectarse a través de sesiones. Para utilizar la opción de cifrar los datos de la sesión con una clave creada en AWS Key Management Service (AWS KMS), el nodo administrado debe tener instalada la versión 2.3.539.0 de SSM Agent o una posterior. Para utilizar perfiles de shell en una sesión, el nodo administrado debe tener instalada la versión 3.0.161.0 de SSM Agent o una posterior. Para iniciar una sesión de SSH o de reenvío de puertos de Session Manager, el nodo administrado debe tener instalada la versión 3.0.222.0 de SSM Agent o una posterior. Para transmitir datos de la sesión mediante los Registros de Amazon CloudWatch, el nodo administrado debe tener instalada la versión 3.0.284.0 de SSM Agent o una posterior. Para obtener información acerca de cómo determinar el número de versión que se ejecuta en una instancia, consulte Verificación del número de versión de SSM Agent. Para obtener más información acerca de la instalación manual o la actualización automática de SSM Agent, consulte Uso de SSM Agent.

Requisito	Descripción
	Acerca de la cuenta ssm-user A partir de la versión 2.3.50.0 de SSM Agent, el agente crea una cuenta de usuario en el nodo administrado, con permisos raíz o de administrador, denominada `ssm-user`. (En las versiones anteriores a la 2.3.612.0, la cuenta se crea cuando SSM Agent se inicia o se reinicia. En la versión 2.3.612.0 y posterior es, la cuenta `ssm-user` se crea la primera vez que se inicia una sesión en el nodo administr ado). Las sesiones se lanzan con las credencia les administrativas de esta cuenta de usuario. Para obtener más información acerca de cómo restringir el control administrativo para esta cuenta, consulte Activación o desactivación de los permisos administrativos de la cuenta ssm-user. ssm-user en controladores de dominio de Windows Server A partir de la versión 2.3.612.0 de SSM Agent, la cuenta `ssm-user` no se crea automátic amente en los nodos administrados que se utilizan como controladores de dominio de Windows Server. Para utilizar Session Manager en un equipo Windows Server que se utiliza como un controlador de dominio, debe crear la cuenta `ssm-user` de forma manual si ella aún no está presente y asignar permisos de administrador de dominio al usuario. En Windows Server, SSM Agent establece una nueva contraseña para la cuenta `ssm-user` cada vez que se inicia una sesión, por lo que

Requisito	Descripción
	no es necesario especificar ninguna contraseña cuando se crea la cuenta.
Conectividad a puntos de enlace	Los nodos administrados a los que se conecta también deben permitir el tráfico saliente HTTPS (puerto 443) a los siguientes puntos de conexión: • ec2messages.*region*.amazonaws.com • ssm.*region*.amazonaws.com • ssmmessages.*region*.amazonaws.com Para obtener más información, consulte los temas siguientes: • Referencia: ec2messages, ssmmessages y otras operaciones de la API • ¿Cómo creo puntos de conexión de VPC para utilizar Systems Manager para administrar instancias de EC2 privadas sin acceso a Internet? en el Centro de conocimientos de AWS re:Post. Alternativamente, puede conectarse a los puntos de enlace necesarios mediante los puntos de enlace de la interfaz. Para obtener más información, consulte Paso 6: (Opcional) Utilizar AWS PrivateLink para configurar un punto de enlace de la VPC para Session Manager.

Requisito	Descripción
AWS CLI	(Opcional) Si utiliza la AWS Command Line Interface (AWS CLI) para iniciar sus sesiones (en lugar de utilizar la consola de AWS Systems Manager o la consola de Amazon EC2), su equipo local debe tener instalada la versión 1.16.12 de la CLI o una posterior. Puede llamar a `aws --version` para comprobar la versión. Si necesita instalar o actualizar la CLI, consulte Instalación de la AWS Command Line Interface en la Guía del usuario de AWS Command Line Interface. **Important** Cada vez que se agregan herramientas nuevas a Systems Manager o se actualizan las herramientas existentes, se lanza una versión actualizada de SSM Agent. No utilizar la versión más reciente del agente puede impedir que el nodo administrado utilice diversas herramientas y características de Systems Manager. Por este motivo, se recomienda automatizar el proceso de mantener SSM Agent actualizado en los equipos. Para obtener más información, consulta Automatización de las actualizaciones de SSM Agent. Suscríbase a la página SSM Agent Release Notes en GitHub para recibir

Requisito	Descripción
	notificaciones sobre las actualizaciones de SSM Agent. Además, si desea utilizar la CLI para administrar los nodos con Session Manager, debe instalar primero el complemento de Session Manager en su equipo local. Para obtener más información, consulta Instalación del complemento de Session Manager para la AWS CLI.
Activación del nivel de instancias avanzadas (entornos híbridos y multinube)	Para conectarse a equipos que no son de EC2 mediante Session Manager, debe activar el nivel de instancias avanzadas en la Cuenta de AWS y la Región de AWS donde crea activaciones híbridas para registrar equipos que no son de EC2 como nodos administrados. El uso del nivel de instancias avanzadas conlleva un cargo. Para obtener más información acerca del nivel de instancias avanzadas, consulte Configuración de los niveles de instancias.

Requisito	Descripción
Verificación de los permisos de rol de servicio de IAM (entornos híbridos y multinube)	Los nodos activados de manera híbrida utilizan el rol de servicio de AWS Identity and Access Management (IAM) especificado en la activación híbrida para comunicarse con las operaciones de la API de Systems Manager. Este rol de servicio debe contener los permisos necesarios para conectarse a los equipos híbridos y multinube mediante Session Manager. Si el rol de servicio contiene la política administrada por AWS `AmazonSSMManagedInstanceCore`, ya se habrán proporcionado los permisos necesarios para Session Manager. Si descubre que el rol de servicio no contiene los permisos necesarios, debe anular el registro de la instancia administrada y registrarla con una nueva activación híbrida que utilice un rol de servicio de IAM con los permisos necesarios. Para obtener más información acerca de cómo se anula el registro de las instancias administradas, consulte Anulación del registro de nodos administrados en un entorno híbrido y multinube. Para obtener más información sobre la creación de políticas de IAM con permisos de Session Manager, consulte Paso 2: verificar o agregar permisos de instancia para Session Manager.

Paso 2: verificación o agregación de permisos de instancia para Session Manager

De forma predeterminada, AWS Systems Manager no tiene permiso para realizar acciones en sus instancias. Puede proporcionar permisos de instancia a nivel de cuenta mediante un rol de AWS Identity and Access Management (IAM) o a nivel de instancia mediante un perfil de instancia. Si su caso de uso lo permite, le recomendamos que conceda el acceso a nivel de cuenta mediante la configuración de administración de host predeterminada. Si ya completó

la configuración de administración de host predeterminada para su cuenta mediante la política `AmazonSSMManagedEC2InstanceDefaultPolicy`, puede continuar con el siguiente paso. Para obtener más información sobre la configuración de administración de host predeterminada, consulte Administración automática de instancias EC2 con la configuración de administración de hosts predeterminada.

Como alternativa, puede usar perfiles de instancia para proporcionar los permisos necesarios a sus instancias. Un perfil de instancias pasa un rol de IAM a una instancia de Amazon EC2. Puede adjuntar un perfil de instancias de IAM a una instancia de Amazon EC2 en el momento de lanzarla, o bien, adjuntarlo a una instancia ya lanzada anteriormente. Para obtener más información, consulte Uso de perfiles de instancia.

Para servidores en las instalaciones o máquinas virtuales, los permisos los proporciona el rol de servicio de IAM asociado a la activación híbrida utilizada para registrar los servidores en las instalaciones y las máquinas virtuales con Systems Manager. Los servidores locales y las máquinas virtuales no utilizan perfiles de instancia.

Si ya utiliza otras herramientas de Systems Manager, como Run Command o Parameter Store, es posible que ya se haya adjuntado a las instancias de Amazon EC2 un perfil de instancias con los permisos básicos necesarios para Session Manager. Si un perfil de instancias que contiene la política administrada por AWS `AmazonSSMManagedInstanceCore` ya se ha adjuntado a las instancias, los permisos necesarios para Session Manager ya se habrán proporcionado. Esto también aplica si el rol de servicio de IAM utilizado en la activación híbrida contiene la política administrada `AmazonSSMManagedInstanceCore`.

Sin embargo, en algunos casos, es posible que tenga que modificar los permisos asociados al perfil de instancia. Por ejemplo, si desea proporcionar un conjunto más limitado de permisos de instancia, si ha creado una política personalizada para su perfil de instancias o si desea utilizar el cifrado de Amazon Simple Storage Service (Amazon S3) o las opciones de cifrado de AWS Key Management Service (AWS KMS) para proteger los datos de la sesión. En estos casos, realice una de las siguientes operaciones para permitir que se realicen acciones de Session Manager en las instancias:

- Inserción de permisos de acciones de Session Manager en un rol de IAM personalizado

  Con el fin de agregar permisos para las acciones de Session Manager a un rol de IAM existente que no se base en la política predeterminada proporcionada por AWS `AmazonSSMManagedInstanceCore`, siga los pasos que se indican en Adición de permisos de Session Manager a un rol de IAM existente.

- Creación de un rol de IAM personalizado solo con permisos de Session Manager

  Para crear un rol de IAM que contenga permisos solo para las acciones de Session Manager, siga los pasos que se indican en Creación de un rol de IAM personalizado para Session Manager.

- Creación y uso de un nuevo rol de IAM con permisos para todas las acciones de Systems Manager

  Para crear un rol de IAM para instancias administradas de Systems Manager que utilice una política predeterminada suministrada por AWS que conceda todos los permisos de Systems Manager, siga los pasos que se indican en Configuración de permisos de instancia requeridos para Systems Manager.

Temas

- Adición de permisos de Session Manager a un rol de IAM existente
- Creación de un rol de IAM personalizado para Session Manager

Adición de permisos de Session Manager a un rol de IAM existente

Utilice el siguiente procedimiento para agregar permisos de Session Manager a un rol de AWS Identity and Access Management (IAM) existente. Si agrega permisos a un rol existente, puede mejorar la seguridad de su entorno de computación sin tener que utilizar la política `AmazonSSMManagedInstanceCore` de AWS para los permisos de instancia.

> **Note**
>
> Tenga en cuenta la siguiente información:
>
> - Este procedimiento supone que el rol existente ya incluye otros permisos `ssm` de Systems Manager para las acciones a las que desea permitir el acceso. Esta política sola no es suficiente para utilizar Session Manager.
> - El siguiente ejemplo de política incluye una acción `s3:GetEncryptionConfiguration`. Esta acción es necesaria si elige la opción Ejecutar cifrado de registros de S3 en las preferencias de registro de Session Manager.

Para agregar permisos de Session Manager a un rol existente (consola)

1. Inicie sesión en AWS Management Console y abra la consola IAM en https://console.aws.amazon.com/iam/.
2. Seleccione Roles en el panel de navegación.
3. Elija el nombre del rol al que le va a agregar los permisos.
4. Elija la pestaña Permisos.
5. Elija Agregar permisos y, a continuación, seleccione Crear política insertada.
6. Seleccione la pestaña JSON.
7. Reemplace la política predeterminada por el siguiente contenido. Reemplace *key-name* por el nombre de recurso de Amazon (ARN) de la clave de AWS Key Management Service (AWS KMS key) que desee utilizar.

   JSON

```
{
    "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": "arn:aws:kms:us-east-1:111122223333:key/key-name"
        }
    ]
}
```

Para obtener más información acerca de cómo utilizar una clave de KMS para cifrar los datos de la sesión, consulte Activación del cifrado de datos de sesión con claves de KMS (consola).

Si no va a usar el cifrado de AWS KMS para los datos de la sesión, puede eliminar el siguiente contenido de la política.

```
,
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "key-name"
        }
```

8. Elija Siguiente: etiquetas.
9. (Opcional) Para agregar etiquetas, elija Add tag (Agregar etiqueta) e ingrese las etiquetas preferidas para la política.
10. Elija Siguiente: Revisar.
11. En la página Review Policy (Revisar política), en Name (Nombre), escriba un nombre para la política insertada, como **`SessionManagerPermissions`**.
12. (Opcional) En Description (Descripción), escriba una descripción para la política.

    Seleccione Crear política.

Para obtener información acerca de las acciones `ssmmessages`, consulte Referencia: ec2messages, ssmmessages y otras operaciones de la API.

Creación de un rol de IAM personalizado para Session Manager

Puede crear un rol de AWS Identity and Access Management (IAM) que conceda a Session Manager el permiso para realizar acciones en las instancias administradas de Amazon EC2. Además, puede

incluir una política para conceder los permisos necesarios para los registros de la sesión que se enviarán a Amazon Simple Storage Service (Amazon S3) y a los Registros de Amazon CloudWatch.

Después de crear el rol de IAM, para obtener información sobre cómo adjuntar el rol a una instancia, consulte Adjuntar o reemplazar un perfil de instancia en el sitio web de AWS re:Post. Para obtener más información sobre los roles y los perfiles de instancia de IAM, consulte Uso de perfiles de instancia en la Guía del usuario de IAM y Roles de IAM para Amazon EC2 en la Guía del usuario de instancias de Linux de Amazon Elastic Compute Cloud. Para obtener más información sobre cómo crear un rol de servicio de IAM para equipos locales, consulte Creación del rol de servicio de IAM requerido para Systems Manager en entornos híbridos y multinube.

Temas

- Creación de un rol de IAM con permisos mínimos de Session Manager (consola)
- Creación de un rol de IAM con permisos para Session Manager, Amazon S3 y los Registros de CloudWatch (consola)

Creación de un rol de IAM con permisos mínimos de Session Manager (consola)

Utilice el siguiente procedimiento para crear un rol de IAM personalizado con una política que proporcione permisos solo para acciones de Session Manager en las instancias.

Para crear un perfil de instancia con permisos mínimos de Session Manager (consola)

1. Inicie sesión en AWS Management Console y abra la consola IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, seleccione Políticas y, a continuación, Crear política. (Si aparece el botón Get Started [Comenzar], elíjalo y, a continuación, elija Create Policy [Crear política]).
3. Seleccione la pestaña JSON.
4. Reemplace el contenido predeterminado por la siguiente política. Para cifrar los datos de la sesión mediante AWS Key Management Service (AWS KMS), reemplace *key-name* por el nombre de recurso de Amazon (ARN) de AWS KMS key que desea utilizar.

   JSON

```
{
    "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": "arn:aws:kms:us-east-1:111122223333:key/key-name"
        }
    ]
}
```

Para obtener más información acerca de cómo utilizar una clave de KMS para cifrar los datos de la sesión, consulte Activación del cifrado de datos de sesión con claves de KMS (consola).

Si no va a usar el cifrado de AWS KMS para los datos de la sesión, puede eliminar el siguiente contenido de la política.

```
,
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "key-name"
        }
```

5. Elija Siguiente: etiquetas.
6. (Opcional) Para agregar etiquetas, elija Add tag (Agregar etiqueta) e ingrese las etiquetas preferidas para la política.
7. Elija Siguiente: Revisar.

8. En la página Review Policy (Revisar política), en Name (Nombre), escriba un nombre para la política insertada, como **`SessionManagerPermissions`**.
9. (Opcional) En Description (Descripción), escriba una descripción para la política.
10. Elija Crear política.
11. En el panel de navegación, seleccione Roles y luego seleccione Crear rol.
12. En la página Create role (Crear un rol), elija AWS service (Servicio de), y en Use case (Caso de uso) elija EC2.
13. Elija Siguiente.
14. En la página Add permissions (Agregar permisos), seleccione la casilla de verificación situada a la izquierda del nombre de la política que acaba de crear, como **`SessionManagerPermissions`**.
15. Elija Siguiente.
16. En la página Name, review, and create (Asignar nombre, revisar y crear), en Role name (Nombre de rol), ingrese un nombre para el rol de IAM, como **`MySessionManagerRole`**.
17. (Opcional) En Role description (Descripción del rol), escriba una descripción para el perfil de instancia.
18. (Opcional) Para agregar etiquetas, elija Add tag (Agregar etiqueta) e ingrese las etiquetas preferidas para el rol.

    Seleccione Crear rol.

Para obtener información acerca de las acciones `ssmmessages`, consulte Referencia: ec2messages, ssmmessages y otras operaciones de la API.

Creación de un rol de IAM con permisos para Session Manager, Amazon S3 y los Registros de CloudWatch (consola)

Utilice el siguiente procedimiento para crear un rol de IAM personalizado con una política que proporcione permisos para acciones de Session Manager en las instancias. La política también proporciona los permisos necesarios para que los registros de la sesión se almacenen en buckets de Amazon Simple Storage Service (Amazon S3) y en grupos de registros de los Registros de Amazon CloudWatch.

> **Important**
>
> Para enviar registros de sesión a un bucket de Amazon S3 que pertenezca a otra Cuenta de AWS, debe agregar el permiso `s3:PutObjectAcl` a la política de rol de IAM. Además, debe asegurarse de que la política de bucket conceda acceso entre cuentas al rol de IAM utilizado por la cuenta propietaria para conceder permisos de Systems Manager para instancias administradas. Si el bucket utiliza el cifrado de Key Management Service (KMS), la política de KMS del bucket también debe conceder este acceso entre cuentas. Para obtener más información sobre cómo configurar permisos de bucket entre cuentas en Amazon S3, consulte Concesión de permisos de bucket entre cuentas en la Guía del usuario de Amazon Simple Storage Service. Si no se agregan estos permisos entre cuentas, la cuenta que posee el bucket de Amazon S3 no podrá acceder a los registros de salida de la sesión.

Para obtener información acerca de cómo especificar preferencias para almacenar los registros de sesiones, consulte Habilitar y deshabilitar el registro de la sesión.

Para crear un rol de IAM con permisos para Session Manager, Amazon S3 y los Registros de CloudWatch (consola)

1. Inicie sesión en AWS Management Console y abra la consola IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, seleccione Políticas y, a continuación, Crear política. (Si aparece el botón Get Started [Comenzar], elíjalo y, a continuación, elija Create Policy [Crear política]).
3. Seleccione la pestaña JSON.
4. Reemplace el contenido predeterminado por la siguiente política. Reemplace cada *example resource placeholder* por su propia información.

   JSON

```
{
    "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:::amzn-s3-demo-bucket/s3-prefix/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetEncryptionConfiguration"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "arn:aws:kms:us-east-1:111122223333:key/key-name"
        },
        {
            "Effect": "Allow",
            "Action": "kms:GenerateDataKey",
            "Resource": "*"
        }
```

```
    ]
}
```

5. Elija Siguiente: etiquetas.
6. (Opcional) Para agregar etiquetas, elija Add tag (Agregar etiqueta) e ingrese las etiquetas preferidas para la política.
7. Elija Siguiente: Revisar.
8. En la página Review Policy (Revisar política), en Name (Nombre), escriba un nombre para la política insertada, como **`SessionManagerPermissions`**.
9. (Opcional) En Description (Descripción), escriba una descripción para la política.
10. Elija Crear política.
11. En el panel de navegación, seleccione Roles y luego seleccione Crear rol.
12. En la página Create role (Crear un rol), elija AWS service (Servicio de), y en Use case (Caso de uso) elija EC2.
13. Elija Siguiente.
14. En la página Add permissions (Agregar permisos), seleccione la casilla de verificación situada a la izquierda del nombre de la política que acaba de crear, como **`SessionManagerPermissions`**.
15. Elija Siguiente.
16. En la página Name, review, and create (Asignar nombre, revisar y crear), en Role name (Nombre de rol), ingrese un nombre para el rol de IAM, como **`MySessionManagerRole`**.
17. (Opcional) En Role description (Descripción del rol), ingrese una descripción para el rol.
18. (Opcional) Para agregar etiquetas, elija Add tag (Agregar etiqueta) e ingrese las etiquetas preferidas para el rol.
19. Seleccione Crear rol.

Paso 3: controlar el acceso de la sesión a los nodos administrados

Puede conceder o revocar el acceso a Session Manager de los nodos administrados mediante políticas de AWS Identity and Access Management (IAM). Puede crear una política y adjuntarla a un grupo o usuario de IAM que especifique a qué nodos administrados se puede conectar el grupo o el usuario. También puede especificar las operaciones de la API de Session Manager que el usuario o los grupos pueden realizar en esos nodos administrados.

Para ayudarlo a empezar a utilizar las políticas de permisos de IAM para Session Manager, creamos ejemplos de políticas para un usuario final y un usuario administrador. Puede utilizar estas políticas solo con cambios pequeños. O utilícelas como guía para crear políticas de IAM personalizadas. Para obtener más información, consulte Ejemplos de políticas de IAM para Session Manager. Para obtener información acerca de cómo crear políticas de IAM y adjuntarlas a usuarios o grupos, consulte Creación de políticas de IAM y Adición y eliminación de políticas de IAM en la Guía del usuario de IAM.

Acerca de los formatos de ARN de ID de sesión

Cuando cree una política de IAM para el acceso a Session Manager, especifique un ID de sesión como parte del nombre de recurso de Amazon (ARN). El ID de la sesión incluye el nombre de usuario como variable. Para ayudar a ilustrarlo, este es el formato de un ARN de Session Manager y un ejemplo:

```
arn:aws:ssm:region-id:account-id:session/session-id
```

Por ejemplo:

```
arn:aws:ssm:us-east-2:123456789012:session/JohnDoe-1a2b3c4d5eEXAMPLE
```

Para obtener más información acerca del uso de variables en políticas de IAM, consulte Elementos de la política de IAM: variables.

Temas

- Inicio de una sesión de intérprete de comandos predeterminada mediante la especificación del documento de sesión predeterminado en las políticas de IAM
- Inicio de una sesión con un documento mediante la especificación de los documentos de sesión en las políticas de IAM
- Ejemplos de políticas de IAM para Session Manager
- Políticas de IAM de ejemplo adicionales para Session Manager

Inicio de una sesión de intérprete de comandos predeterminada mediante la especificación del documento de sesión predeterminado en las políticas de IAM

Cuando configura Session Manager para su Cuenta de AWS o cambia las preferencias de sesión en la consola de Systems Manager, el sistema crea un documento de sesión SSM llamado `SSM-`

`SessionManagerRunShell`. Es el documento de sesión predeterminado. Session Manager utiliza este documento para almacenar las preferencias de sesión, que incluyen información como la siguiente:

- Una ubicación en la que desee guardar los datos de la sesión, como un bucket de Amazon Simple Storage Service (Amazon S3) o un grupo de registro de Registros de Amazon CloudWatch.
- Un ID de clave de AWS Key Management Service (AWS KMS) para cifrar los datos de la sesión.
- Si se permite el soporte Ejecutar como para las sesiones.

A continuación, se incluye un ejemplo de la información incluida en el documento de preferencias de sesión denominado `SSM-SessionManagerRunShell`.

```
{
  "schemaVersion": "1.0",
  "description": "Document to hold regional settings for Session Manager",
  "sessionType": "Standard_Stream",
  "inputs": {
    "s3BucketName": "amzn-s3-demo-bucket",
    "s3KeyPrefix": "MyS3Prefix",
    "s3EncryptionEnabled": true,
    "cloudWatchLogGroupName": "MyCWLogGroup",
    "cloudWatchEncryptionEnabled": false,
    "kmsKeyId": "1a2b3c4d",
    "runAsEnabled": true,
    "runAsDefaultUser": "RunAsUser"
  }
}
```

De forma predeterminada, Session Manager utiliza el documento de sesión predeterminado cuando un usuario inicia una sesión desde la AWS Management Console. Esto se aplica a Fleet Manager o Session Manager en la consola de Systems Manager o a EC2 Connect en la consola de Amazon EC2. Además, Session Manager utiliza el documento de sesión predeterminado cuando un usuario inicia una sesión mediante un comando de la AWS CLI, como el siguiente:

```
aws ssm start-session \
    --target i-02573cafcfEXAMPLE
```

Para iniciar una sesión de intérprete de comandos predeterminada, debe especificar el documento de la sesión predeterminada en la política de IAM, tal como se muestra en el siguiente ejemplo.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnableSSMSession",
      "Effect": "Allow",
      "Action": [
        "ssm:StartSession"
      ],
      "Resource": [
        "arn:aws:ec2:us-east-1:111122223333:instance/instance-id",
        "arn:aws:ssm:us-east-1:111122223333:document/SSM-SessionManagerRunShell"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "ssmmessages:OpenDataChannel"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

Inicio de una sesión con un documento mediante la especificación de los documentos de sesión en las políticas de IAM

Si utiliza el comando de la AWS CLI start-session con el documento de sesión predeterminado, puede omitir el nombre del documento. El sistema llama de manera automática al documento de sesión `SSM-SessionManagerRunShell`.

En los demás casos, debe especificar un valor para el parámetro `document-name`. Cuando un usuario especifica el nombre de un documento de sesión en un comando, el sistema comprueba su política de IAM para verificar que tiene permiso para acceder al documento. Si no tiene permiso, se produce un error en la solicitud de conexión. Los siguientes ejemplos incluyen el parámetro `document-name` con el documento de sesión `AWS-StartPortForwardingSession`.

```
aws ssm start-session \
    --target i-02573cafcfEXAMPLE \
    --document-name AWS-StartPortForwardingSession \
    --parameters '{"portNumber":["80"], "localPortNumber":["56789"]}'
```

Para ver un ejemplo acerca de cómo se especifica un documento de sesión de Session Manager en una política de IAM, consulte Políticas de usuarios finales de inicio rápido de Session Manager.

> **Note**
>
> Para iniciar una sesión con SSH, los pasos de configuración deben completarse en el nodo administrado de destino y en la máquina local del usuario. Para obtener información, consulte (Opcional) Permitir y controlar permisos para conexiones de SSH mediante Session Manager.

Ejemplos de políticas de IAM para Session Manager

Utilice las muestras de esta sección para que lo ayuden a crear políticas de AWS Identity and Access Management (IAM) que proporcionen los permisos que se necesitan más a menudo para acceder a Session Manager.

> **Note**
>
> También puede utilizar una política de AWS KMS key para controlar qué entidades de IAM (usuarios o roles) y Cuentas de AWS tienen acceso a su clave de KMS. Para obtener información, consulte Información general sobre la administración del acceso a sus recursos de AWS KMS y Uso de políticas de claves en AWS KMS en la Guía para desarrolladores de AWS Key Management Service.

Temas

- Políticas de usuarios finales de inicio rápido de Session Manager
- Política de administradores de inicio rápido para Session Manager

Políticas de usuarios finales de inicio rápido de Session Manager

Utilice los siguientes ejemplos para crear políticas de usuario final de IAM para Session Manager.

Puede crear una política que permita a los usuarios iniciar sesiones únicamente desde la consola de Session Manager, la AWS Command Line Interface (AWS CLI), la consola de Amazon Elastic Compute Cloud (Amazon EC2) o desde las tres.

Estas políticas proporcionan a los usuarios finales la capacidad de iniciar una sesión en un nodo administrado particular y de terminar solo sus propias sesiones. Consulte Políticas de IAM de ejemplo adicionales para Session Manager para ver ejemplos de personalizaciones recomendadas para aplicar en la política.

En las siguientes políticas de ejemplo, reemplace cada *example resource placeholder* por su propia información.

Consulte las secciones siguientes para ver ejemplos de políticas correspondientes al rango de acceso a la sesión que desea proporcionar.

Session Manager and Fleet Manager

Utilice este ejemplo de política para brindar a los usuarios la capacidad de iniciar y continuar sesiones solo desde las consolas Session Manager y Fleet Manager.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:us-
east-1:111122223333:instance/i-02573cafcfEXAMPLE",
                "arn:aws:ssm:us-east-1:111122223333:document/SSM-
SessionManagerRunShell"
            ]
        },
        {
         "Effect": "Allow",
         "Action": ["ssmmessages:OpenDataChannel"],
         "Resource": ["arn:aws:ssm:*:*:session/${aws:userid}-*"]
       },
```

```
        {
            "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"
            ],
            "Resource": "arn:aws:kms:us-east-1:111122223333:key/key-name"
        }
    ]
}
```

Amazon EC2

Utilice este ejemplo de política para brindar a los usuarios la capacidad de iniciar y continuar sesiones solo desde la consola de Amazon EC2. Esta política no proporciona todos los permisos necesarios para iniciar sesiones desde la consola de Session Manager y la AWS CLI.

JSON

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

```
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession",
                "ssm:SendCommand"
            ],
            "Resource": [
                "arn:aws:ec2:us-
east-1:111122223333:instance/i-02573cafcfEXAMPLE",
                "arn:aws:ssm:us-east-1:111122223333:document/SSM-
SessionManagerRunShell"
            ]
        },
        {
         "Effect": "Allow",
         "Action": ["ssmmessages:OpenDataChannel"],
         "Resource": ["arn:aws:ssm:*:*:session/${aws:userid}-*"]
       },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceInformation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:username}-*"
            ]
        }
    ]
}
```

AWS CLI

Utilice este ejemplo de política para brindar a los usuarios la capacidad de iniciar y continuar sesiones solo desde AWS CLI.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession",
                "ssm:SendCommand"
            ],
            "Resource": [
                "arn:aws:ec2:us-
east-1:111122223333:instance/i-02573cafcfEXAMPLE",
                "arn:aws:ssm:us-east-1:111122223333:document/SSM-
SessionManagerRunShell"
            ]
        },
        {
         "Effect": "Allow",
         "Action": ["ssmmessages:OpenDataChannel"],
         "Resource": ["arn:aws:ssm:*:*:session/${aws:userid}-*"]
       },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:GenerateDataKey"
            ],
```

```
            "Resource": "arn:aws:kms:us-east-1:111122223333:key/key-name"
        }
    ]
}
```

[1] `SSM-SessionManagerRunShell` es el nombre predeterminado del documento de SSM que Session Manager crea para almacenar sus preferencias de configuración de la sesión. En cambio, puede crear un documento de Session personalizado y especificarlo en esta política. También puede especificar el documento proporcionado por AWS `AWS-StartSSHSession` para los usuarios que inician sesiones mediante SSH. Para obtener más información acerca de los pasos de configuración necesarios para admitir sesiones mediante SSH, consulte (Opcional) Permitir y controlar permisos para conexiones de SSH a través de Session Manager.

[2] El permiso `kms:GenerateDataKey` le permite crear una clave de cifrado de datos que se utilizará para cifrar los datos de la sesión. Si va a usar el cifrado de AWS Key Management Service (AWS KMS) para los datos de la sesión, sustituya *key-name* por el nombre de recurso de Amazon (ARN) de la clave de KMS que desee utilizar, con el formato `arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-12345EXAMPLE`. Si no va a usar el cifrado de claves de KMS para sus datos de sesión, elimine el siguiente contenido de la política.

```
{
            "Effect": "Allow",
            "Action": [
                "kms:GenerateDataKey"
            ],
            "Resource": "key-name"
        }
```

Para obtener más información acerca del uso de AWS KMS para cifrar los datos de la sesión, consulte Activación del cifrado de datos de sesión con claves de KMS (consola).

[3] El permiso para SendCommand es necesario para los casos en que un usuario intente iniciar una sesión desde la consola Amazon EC2, pero primero debe actualizar SSM Agent a la versión mínima requerida para Session Manager. Run Commandse utiliza para enviar un comando a la instancia para actualizar el agente.

Política de administradores de inicio rápido para Session Manager

Utilice los siguientes ejemplos para crear políticas de administrador de IAM para Session Manager.

Estas políticas proporcionan a los administradores la capacidad de iniciar una sesión en los nodos administrados que están etiquetados con `Key=Finance,Value=WebServers`, el permiso para crear, actualizar y eliminar las preferencias de los usuarios y el permiso para terminar únicamente sus propias sesiones. Consulte Políticas de IAM de ejemplo adicionales para Session Manager para ver ejemplos de personalizaciones recomendadas para aplicar en la política.

Puede crear una política que permita a los administradores realizar estas tareas únicamente desde la consola de Session Manager, la AWS CLI, la consola de Amazon EC2 o desde las tres.

En las siguientes políticas de ejemplo, reemplace cada *example resource placeholder* por su propia información.

Consulte las secciones siguientes para ver ejemplos de políticas para los tres escenarios de permisos.

Session Manager and CLI

Utilice este ejemplo de política para brindar a los administradores la capacidad de llevar a cabo las tareas relacionadas con la sesión solo desde la consola de Session Manager y la AWS CLI. Esta política no proporciona todos los permisos necesarios para realizar las tareas relacionadas con la sesión desde la consola de Amazon EC2.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:*:111122223333:instance/*"
            ],
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/Finance": [
                        "WebServers"
                    ]
                }
```

```
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssmmessages:OpenDataChannel"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeSessions",
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceProperties",
                "ec2:DescribeInstances"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:CreateDocument",
                "ssm:UpdateDocument",
                "ssm:GetDocument",
                "ssm:StartSession"
            ],
            "Resource": "arn:aws:ssm:us-east-1:111122223333:document/SSM-
SessionManagerRunShell"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssmmessages:OpenDataChannel"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
```

```
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        }
    ]
}
```

Amazon EC2

Utilice este ejemplo de política para brindar a los administradores la capacidad de llevar a cabo tareas relacionadas con la sesión solo desde la consola de Amazon EC2. Esta política no proporciona todos los permisos necesarios para realizar tareas relacionadas con la sesión desde la consola de Session Manager y la AWS CLI.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession",
                "ssm:SendCommand"
            ],
            "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": ["ssmmessages:OpenDataChannel"],
         "Resource": ["arn:aws:ssm:*:*:session/${aws:userid}-*"]
       },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceInformation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        }
    ]
}
```

Session Manager, CLI, and Amazon EC2

Utilice este ejemplo de política para brindar a los administradores la capacidad de llevar a cabo tareas relacionadas con la sesión desde la consola de Session Manager, la AWS CLI y la consola de Amazon EC2.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession",
                "ssm:SendCommand"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1:111122223333:instance/*"
            ],
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/tag-key": [
                        "tag-value"
                    ]
                }
            }
        },
        {
         "Effect": "Allow",
         "Action": ["ssmmessages:OpenDataChannel"],
         "Resource": ["arn:aws:ssm:*:*:session/${aws:userid}-*"]
       },
        {
            "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:us-east-1:111122223333:document/SSM-
SessionManagerRunShell"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        }
    ]
}
```

[1] El permiso para SendCommand es necesario en aquellos casos en los que un usuario intenta iniciar una sesión desde la consola de Amazon EC2, pero antes se debe enviar un comando para actualizar SSM Agent.

Políticas de IAM de ejemplo adicionales para Session Manager

Consulte los siguientes ejemplos de políticas para crear una política de AWS Identity and Access Management (IAM) personalizada para todos los escenarios de acceso de usuarios a Session Manager que desee admitir.

Temas

- Ejemplo 1: conceder acceso a documentos en la consola
- Ejemplo 2: restringir el acceso a nodos administrados específicos
- Ejemplo 3: restringir el acceso en función de etiquetas
- Ejemplo 4: permitir a un usuario finalizar solo las sesiones que inició
- Ejemplo 5: permitir acceso completo (administrativo) a todas las sesiones

Ejemplo 1: conceder acceso a documentos en la consola

Puede permitir que los usuarios especifiquen un documento personalizado cuando inician una sesión mediante la consola del administrador de sesiones. El siguiente ejemplo de política de IAM concede permiso para acceder a documentos con nombres que comiencen con **`SessionDocument-`** en la Región de AWS y la Cuenta de AWS especificadas.

Para utilizar esta política, reemplace cada *`example-resource-placeholder`* con su propia información.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetDocument",
                "ssm:ListDocuments"
            ],
            "Resource": [
                "arn:aws:ssm:us-east-1:111122223333:document/SessionDocument-*"
            ]
        }
    ]
}
```

> **Note**
>
> La consola del administrador de sesiones solo admite documentos de sesión que tienen un `sessionType` de `Standard_Stream`, los cuales se utilizan para definir las preferencias de sesión. Para obtener más información, consulte Esquema del documento de Session.

Ejemplo 2: restringir el acceso a nodos administrados específicos

Puede crear una política de IAM que defina a qué nodos administrados puede conectarse un usuario mediante Administrador de sesiones. Por ejemplo, la siguiente política otorga al usuario el permiso

para iniciar, finalizar y reanudar sus sesiones en tres nodos específicos. La política impide que el usuario se conecte a nodos distintos de los especificados.

> **Note**
>
> Para los usuarios federados, consulte Ejemplo 4: permitir a un usuario finalizar solo las sesiones que inició.

JSON

```
{
    "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": ["ssmmessages:OpenDataChannel"],
         "Resource": ["arn:aws:ssm:*:*:session/${aws:userid}-*"]
       },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
```

```
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        },
        {
         "Effect": "Allow",
         "Action": ["ssmmessages:OpenDataChannel"],
         "Resource": ["arn:aws:ssm:*:*:session/${aws:userid}-*"]
       }
    ]
}
```

Ejemplo 3: restringir el acceso en función de etiquetas

Puede restringir el acceso a nodos administrados en función de etiquetas específicas. En el siguiente ejemplo, el usuario tiene permitido iniciar y reanudar sesiones (`Effect: Allow, Action: ssm:StartSession, ssm:ResumeSession`) en cualquier nodo administrado (`Resource: arn:aws:ec2:`*region*`:`*987654321098*`:instance/*`) con la condición de que el nodo sea un servidor web de finanzas (`ssm:resourceTag/Finance: WebServer`). Si el usuario envía un comando a un nodo administrado que no está etiquetado o que tiene alguna etiqueta que no sea `Finance: WebServer`, el resultado del comando incluirá `AccessDenied`.

JSON

```
{
    "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": ["ssmmessages:OpenDataChannel"],
         "Resource": ["arn:aws:ssm:*:*:session/${aws:userid}-*"]
       },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ssm:us-east-2:123456789012:document/SSM-
SessionManagerRunShell"
            ]
        }
    ]
}
```

Puede crear políticas de IAM que permitan al usuario iniciar sesiones en los nodos administrados que tengan varias etiquetas. La siguiente política permite al usuario iniciar sesiones en nodos administrados que tengan las dos etiquetas especificadas aplicadas. Si un usuario envía un comando a un nodo administrado que no está etiquetado con estas dos etiquetas, el resultado del comando incluirá `AccessDenied`.

JSON

```
{
   "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": ["ssmmessages:OpenDataChannel"],
         "Resource": ["arn:aws:ssm:*:*:session/${aws:userid}-*"]
       },
      {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ssm:us-east-2:123456789012:document/SSM-
SessionManagerRunShell"
            ]
      }
   ]
}
```

Para obtener más información acerca de la creación de políticas de IAM, consulte Políticas administradas y políticas insertadas en la Guía del usuario de IAM. Para obtener más información sobre el etiquetado de nodos administrados, consulte Etiquetado de los recursos de Amazon EC2 en la Guía del usuario de Amazon EC2 (el contenido se aplica a los nodos administrados de Windows y Linux). Para obtener más información acerca de cómo aumentar su posición de seguridad frente a comandos de nivel raíz no autorizados en los nodos administrados, consulte Restricción del acceso a los comandos de nivel raíz con SSM Agent

Ejemplo 4: permitir a un usuario finalizar solo las sesiones que inició

Session Manager proporciona dos métodos para controlar qué sesiones tiene permitido finalizar un usuario federado en su Cuenta de AWS.

- Utilice la variable `{aws:userid}` en una política de permisos de AWS Identity and Access Management (IAM). Los usuarios federados solo pueden finalizar las sesiones que iniciaron. Para los usuarios no federados, utilice el Método 1. Para los usuarios federados, utilice el Método 2.
- Utilice las etiquetas proporcionadas por AWS en una política de permisos de IAM. En la política, incluya una condición que permita a los usuarios terminar solo las sesiones marcadas con etiquetas específicas proporcionadas por AWS. Este método funciona para todas las cuentas, incluidas las que utilizan ID federados para otorgar acceso a AWS.

Método 1: conceder privilegios TerminateSession usando la variable **`{aws:username}`**

La siguiente política de IAM permite al usuario ver los ID de todas las sesiones de su cuenta. Sin embargo, los usuarios solo pueden interactuar con los nodos administrados a través de las sesiones que iniciaron. Un usuario con la siguiente política asignada no puede conectarse a las sesiones de otros usuarios ni terminarlas. La política utiliza la variable `{aws:username}` para lograrlo.

> **Note**
>
> Este método no funciona para las cuentas que otorgan acceso a AWS con ID federados.

JSON

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

```
        {
            "Action": [
                "ssm:TerminateSession"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:username}-*"
            ]
        }
    ]
}
```

Método 2: otorgar privilegios de TerminateSession mediante etiquetas proporcionadas por AWS

Puede controlar qué sesiones puede finalizar un usuario si incluye variables de claves de etiqueta condicionales en una política de IAM. La condición especifica que el usuario solo puede finalizar las sesiones etiquetadas con una o ambas de estas variables de clave de etiqueta específicas y un valor especificado.

Cuando un usuario de su Cuenta de AWS inicia una sesión, Session Manager aplica dos etiquetas de recurso a la sesión. La primera etiqueta de recurso es `aws:ssmmessages:target-id`, con la que se especifica el ID del destino que el usuario puede finalizar. La otra etiqueta de recurso es `aws:ssmmessages:session-id`, con un valor en el formato *`role-id`*`:`*`caller-specified-role-name`*.

> **Note**
>
> Session Manager no admite etiquetas personalizadas para esta política de control de acceso de IAM. Debe utilizar las etiquetas de recursos proporcionadas por AWS que se describen a continuación.

`aws:ssmmessages:target-id`

Con esta clave de etiqueta, el ID del nodo administrado se incluye como valor en la política. En el siguiente bloque de política, la instrucción de condición permite al usuario terminar solo el nodo i-02573cafcfEXAMPLE.

JSON

```
{
     "Version": "2012-10-17",
     "Statement": [
         {
             "Effect": "Allow",
             "Action": [
                "ssm:TerminateSession"
             ],
             "Resource": "*",
             "Condition": {
                 "StringLike": {
                     "ssm:resourceTag/aws:ssmmessages:target-id": [
                        "i-02573cafcfEXAMPLE"
                     ]
                 }
             }
         }
     ]
}
```

Si el usuario intenta finalizar una sesión para la que no se le ha concedido este permiso `TerminateSession`, recibirá un error `AccessDeniedException`.

`aws:ssmmessages:session-id`

Esta clave de etiqueta incluye una variable para el ID de sesión como valor en la solicitud para iniciar una sesión.

En el ejemplo siguiente se muestra una política para los casos en los que el tipo de intermediario es `User`. El valor que proporciona para `aws:ssmmessages:session-id` es el ID del usuario. En este ejemplo, `AIDIODR4TAW7CSEXAMPLE` representa el ID de un usuario de su Cuenta de AWS. Para recuperar el ID de un usuario de su Cuenta de AWS, utilice el comando `get-user` de IAM. Para obtener información, consulte get-user en la sección AWS Identity and Access Management de la Guía del usuario de IAM.

JSON

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
               "ssm:TerminateSession"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/aws:ssmmessages:session-id": [
                       "AIDIODR4TAW7CSEXAMPLE"
                    ]
                }
            }
        }
    ]
}
```

En el ejemplo siguiente se muestra una política para los casos en los que el tipo de intermediario es `AssumedRole`. Puede utilizar la variable `{aws:userid}` en el valor de `aws:ssmmessages:session-id`. También puede codificar de forma rígida un ID de rol para el valor de `aws:ssmmessages:session-id`. Si codifica un ID de rol, debe proporcionar el valor en el formato *`role-id`*`:`*`caller-specified-role-name`*. Por ejemplo, `AIDIODR4TAW7CSEXAMPLE:MyRole`.

> **Important**
>
> Para que se apliquen las etiquetas del sistema, el ID de rol que proporcione solo puede contener los siguientes caracteres: letras Unicode, 0-9, espacio `_`, `.`, `:`, `/`, `=`, `+`, `-`, `@` y `\`.

Para recuperar el ID de un rol de su Cuenta de AWS, utilice el comando `get-caller-identity`. Para obtener información, consulte get-caller-identity en la Referencia de comandos de la AWS CLI.

JSON

```
{
    "Version": "2012-10-17",
```

```
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
               "ssm:TerminateSession"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/aws:ssmmessages:session-id": [
                       "${aws:userid}*"
                    ]
                }
            }
        }
    ]
}
```

Si un usuario intenta finalizar una sesión para la que no se le ha concedido este permiso `TerminateSession`, recibirá un error `AccessDeniedException`.

`aws:ssmmessages:target-id` y `aws:ssmmessages:session-id`

También puede crear políticas de IAM que permitan al usuario terminar sesiones marcadas con ambas etiquetas del sistema, como se muestra en este ejemplo.

JSON

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

Ejemplo 5: permitir acceso completo (administrativo) a todas las sesiones

La siguiente política de IAM permite a un usuario interactuar totalmente con todos los nodos administrados y sesiones creadas por todos los usuarios de todos los nodos. Debe ser concedida únicamente a un administrador que necesita un control completo sobre las actividades de Session Manager de la organización.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "ssm:StartSession",
                "ssm:TerminateSession",
                "ssm:ResumeSession",
                "ssm:DescribeSessions",
                "ssm:GetConnectionStatus"
            ],
            "Effect": "Allow",
            "Resource": [
                "*"
            ]
        },
        {
         "Effect": "Allow",
         "Action": ["ssmmessages:OpenDataChannel"],
         "Resource": ["arn:aws:ssm:*:*:session/${aws:userid}-*"]
       }
    ]
}
```

Paso 4: configurar las preferencias de sesión

Los usuarios a los que se les hayan concedido permisos administrativos en su política de AWS Identity and Access Management (IAM) pueden configurar las preferencias de sesión, incluidas las siguientes:

- Active Ejecutar como soporte para nodos administrados de Linux. Esto permite iniciar sesiones con las credenciales de un usuario especificado del sistema operativo en lugar de con las credenciales de una cuenta `ssm-user` generada por el sistema que AWS Systems Manager Session Manager puede crear en un nodo administrado.
- Configure Session Manager para que utilice el cifrado de AWS KMS key para proporcionar protección adicional a los datos transmitidos entre los equipos del clientes y los nodos administrados.
- Configure Session Manager para crear y enviar los registros del historial de sesiones a un bucket de Amazon Simple Storage Service (Amazon S3) o a un grupo de registros de los Registros de Amazon CloudWatch. Los datos de registro almacenados pueden utilizarse para informar sobre conexiones de sesiones realizadas en los nodos administrados y los comandos ejecutados durante las sesiones.
- Configure los tiempos de espera de sesión. Puede utilizar esta configuración para especificar cuándo terminar una sesión después de un periodo de inactividad.
- Configure Session Manager para que utilice perfiles de shell configurables. Estos perfiles personalizables le permiten definir preferencias dentro de las sesiones, como las preferencias de shell, las variables de entorno, los directorios de trabajo y la ejecución de varios comandos cuando se inicia una sesión.

Para obtener más información acerca de los permisos necesarios para configurar Session Manager, consulte the section called “Concesión o denegación de permisos de usuario para actualizar preferencias de Session Manager”.

Temas

- Concesión o denegación de permisos de usuario para actualizar preferencias de Session Manager
- Especificación de un valor de tiempo de espera de sesión inactiva
- Especificar la duración máxima de la sesión
- Permiso de perfiles de shell configurable
- Activación del soporte Ejecutar como para nodos administrados de Linux y macOS

- Activación del cifrado de datos de sesión con claves de KMS (consola)
- Creación de un documento de preferencias de Session Manager (línea de comandos)
- Actualizar preferencias de Session Manager (línea de comandos)

Para obtener más información acerca de cómo utilizar la consola de Systems Manager para configurar opciones para el registro de los datos de las sesiones, consulte los siguientes temas:

- Registro de los datos de la sesión con Amazon S3 (consola)
- Streaming de los datos de la sesión con los Registros de Amazon CloudWatch (consola)
- Registro de los datos de la sesión con los Registros de Amazon CloudWatch (consola)

Concesión o denegación de permisos de usuario para actualizar preferencias de Session Manager

Las preferencias de la cuenta se almacenan como documentos de AWS Systems Manager (SSM) para cada Región de AWS. Antes de que un usuario pueda actualizar las preferencias de cuenta para las sesiones de su cuenta, es preciso que se le concedan los permisos necesarios para obtener acceso al tipo de documento de SSM donde se almacenan estas preferencias. Estos permisos se conceden por medio de políticas de AWS Identity and Access Management (IAM).

Política de administradores para permitir la creación y actualización de preferencias

Un administrador puede tener la siguiente política para crear y actualizar las preferencias en cualquier momento. La siguiente política otorga permisos para acceder al documento `SSM-SessionManagerRunShell` de la cuenta 123456789012 en la región us-east-2 y actualizarlo.

JSON

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

Política de usuarios para evitar que se actualicen las preferencias

Utilice la siguiente política para impedir que los usuarios finales de su cuenta actualicen o anulen las preferencias de Session Manager.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "ssm:CreateDocument",
                "ssm:GetDocument",
                "ssm:UpdateDocument",
                "ssm:DeleteDocument"
            ],
            "Effect": "Deny",
            "Resource": [
                "arn:aws:ssm:us-east-1:111122223333:document/SSM-
SessionManagerRunShell"
            ]
        }
    ]
}
```

Especificación de un valor de tiempo de espera de sesión inactiva

Session Manager, una herramienta de AWS Systems Manager, le permite especificar la cantidad de tiempo que permitirá al usuario estar inactivo antes de que una sesión termine. De forma predeterminada, el tiempo de espera de las sesiones finaliza después de 20 minutos de inactividad.

Puede modificar esta configuración para especificar que la sesión termine después de entre 1 y 60 minutos de inactividad. Algunas agencias profesionales de seguridad informática recomiendan configurar los tiempos de espera de sesión inactiva en un máximo de 15 minutos.

Para permitir el tiempo de espera de sesión inactiva (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Session Manager.
3. Elija la pestaña Preferences (Preferencias) y, a continuación, Edit (Editar).
4. Especifique la cantidad de tiempo que permitirá que el usuario esté inactivo antes de que finalice la sesión en el campo minutes (minutos) dentro de Idle session timeout (Tiempo de espera de sesión inactiva).
5. Seleccione Guardar.

Especificar la duración máxima de la sesión

Session Manager, una herramienta de AWS Systems Manager, le permite determinar la duración máxima de una sesión antes de que finalice. De forma predeterminada, las sesiones no tienen una duración máxima. El valor que defina para la duración máxima de la sesión debe estar entre 1 y 1440 minutos.

Para especificar la duración máxima de la sesión (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Session Manager.
3. Elija la pestaña Preferences (Preferencias) y, a continuación, Edit (Editar).
4. Seleccione la casilla de verificación situada junto a Enable maximum session duration (Habilitación de la duración máxima de la sesión).
5. Especifique la duración máxima de la sesión antes de que finalice en el campo minutes (minutos) en Maximum session duration (Duración máxima de la sesión).
6. Seleccione Guardar.

Permiso de perfiles de shell configurable

De forma predeterminada, las sesiones en las instancias de EC2 para Linux se inician mediante el shell Bourne (sh). Sin embargo, es posible que prefiera utilizar otro shell, como bash. Cuando permite perfiles de shell configurables, puede personalizar las preferencias dentro de las sesiones, como preferencias de shell, variables de entorno, directorios de trabajo y ejecutar varios comandos cuando se inicia una sesión.

Important

Systems Manager no verifica los comandos ni los scripts del perfil de shell antes de ejecutarlos para ver qué cambios realizarían en una instancia. Para restringir la capacidad de un usuario de modificar los comandos o los scripts ingresados en su perfil de shell, se recomienda lo siguiente:

- Cree un documento personalizado de tipo sesión para sus usuarios y roles de AWS Identity and Access Management (IAM). A continuación, modifique la política de IAM para estos usuarios y roles, de modo que la operación `StartSession` de la API solo pueda utilizar el documento de tipo sesión que creó para ellos. Para obtener más información, consulte Creación de un documento de preferencias de Session Manager (línea de comandos) y Políticas de usuarios finales de inicio rápido de Session Manager.
- Modifique la política de IAM de los usuarios y los roles de IAM para denegar el acceso a la operación `UpdateDocument` de la API para el recurso del documento de tipo sesión que cree. Esto permite a los usuarios y los roles usar el documento que creó para sus preferencias de sesión, sin permitirles modificar ninguna de las configuraciones.

Para activar perfiles de shell configurables

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Session Manager.
3. Elija la pestaña Preferences (Preferencias) y, a continuación, Edit (Editar).
4. Especifique las variables de entorno, las preferencias de shell o los comandos que desee ejecutar cuando se inicie la sesión en los campos de los sistemas operativos aplicables.
5. Seleccione Guardar.

A continuación, se muestran algunos comandos de ejemplo que se pueden agregar a su perfil de shell.

Cambie al shell bash y al directorio /usr en las instancias de Linux.

```
exec /bin/bash
cd /usr
```

Muestre una marca de tiempo y un mensaje de bienvenida al inicio de una sesión.

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."
```

Vea la actividad dinámica del sistema al inicio de una sesión.

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 "-------  ------    -----      ----- -----   ------     -- -----------"}
```

Activación del soporte Ejecutar como para nodos administrados de Linux y macOS

De forma predeterminada, Session Manager autentica las conexiones con las credenciales de la cuenta `ssm-user` generada por el sistema que se crea en un nodo administrado. (En los equipos Linux y macOS, la cuenta se agrega a `/etc/sudoers/`). Si lo desea, puede autenticar las sesiones con las credenciales de una cuenta de usuario del sistema operativo (SO) o un usuario de dominio para instancias unidas a un Active Directory. En este caso, Session Manager comprueba que la cuenta del sistema operativo que especificó existe en el nodo o en el dominio antes de iniciar la sesión. Si intenta iniciar una sesión con una cuenta del sistema operativo que no existe en el nodo ni en el dominio, se produce un error en la conexión.

> **Note**
>
> Administrador de sesiones no admite el uso de una cuenta de usuario `root` de un sistema operativo para autenticar las conexiones. En el caso de las sesiones que se autentican mediante una cuenta de usuario del sistema operativo, es posible que no se apliquen las políticas de directorio y nivel de sistema del nodo, como las restricciones de inicio de sesión o las restricciones de uso de los recursos del sistema.

Funcionamiento

Si activa Ejecutar como soporte para las sesiones, el sistema verifica los permisos de acceso tal como se indica a continuación:

1. Para el usuario que está iniciando la sesión, ¿se ha etiquetado su entidad de IAM (usuario o rol) con `SSMSessionRunAs = os user account name`?

   En caso afirmativo, ¿existe el nombre de usuario del sistema operativo en el nodo administrado? En caso afirmativo, inicie la sesión. Si no es así, no permita que se inicie una sesión.

   Si la entidad de IAM no se ha etiquetado con `SSMSessionRunAs = os user account name`, continúe con el paso 2.
2. Si la entidad de IAM no se ha etiquetado con `SSMSessionRunAs = os user account name`, ¿se ha especificado un nombre de usuario del sistema operativo en las preferencias de la Cuenta de AWS de Session Manager?

En caso afirmativo, ¿existe el nombre de usuario del sistema operativo en el nodo administrado? En caso afirmativo, inicie la sesión. Si no es así, no permita que se inicie una sesión.

> **Note**
>
> Al activar el soporte Ejecutar como, se impide que Administrador de sesiones inicie sesiones con la cuenta `ssm-user` en un nodo administrado. Esto significa que si Session Manager no se puede conectar mediante la cuenta de usuario del sistema operativo especificada, no se recurre a la conexión mediante el método predeterminado.
> Si activa Ejecutar como sin especificar una cuenta de sistema operativo ni etiquetar una entidad de IAM y no ha especificado ninguna cuenta de sistema operativo en las preferencias de Administrador de sesiones, se produce un error en los intentos de conexión a la sesión.

Para activar el soporte Ejecutar como para nodos administrados de Linux y macOS

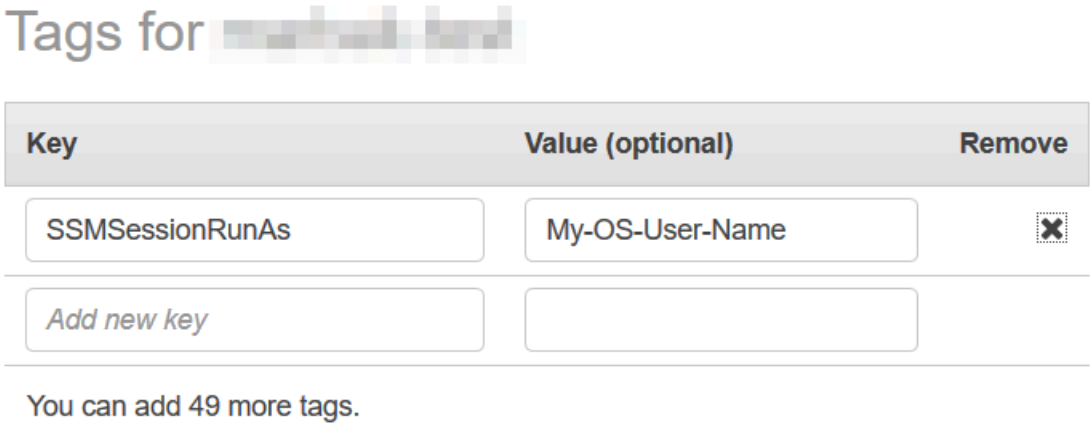
1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Session Manager.
3. Elija la pestaña Preferences (Preferencias) y, a continuación, Edit (Editar).
4. Seleccione la casilla de verificación situada junto a Activar soporte Ejecutar como para instancias de Linux.
5. Realice una de las siguientes acciones:

   - Opción 1: en el campo Nombre de usuario del sistema operativo, ingrese el nombre de la cuenta de usuario del sistema operativo que desea usar para iniciar sesiones. Con esta opción, el mismo usuario del sistema operativo ejecuta todas las sesiones para todos los usuarios de su Cuenta de AWS que se conectan mediante Session Manager.
   - Opción 2 (recomendada): elija el enlace Abrir la consola de IAM. En el panel de navegación, seleccione Usuarios o Roles. Elija la entidad (usuario o rol) a la que añadir etiquetas y, a continuación, elija la pestaña Tags. Escriba `SSMSessionRunAs` para el nombre de la clave. Ingrese el nombre de una cuenta de usuario del sistema operativo para el valor clave. Seleccione Save changes (Guardar cambios).

     Con esta opción, puede especificar usuarios de sistema operativo únicos para diferentes entidades de IAM, si así lo desea. Para obtener más información acerca del etiquetado de

entidades (usuarios o roles) de IAM, consulte Etiquetado de recursos de IAM en la Guía del usuario de IAM.

A continuación se muestra un ejemplo.



6. Seleccione Save.

Activación del cifrado de datos de sesión con claves de KMS (consola)

Utilice AWS Key Management Service (AWS KMS) para crear y administrar las claves de cifrado. Con AWS KMS, puede controlar el uso del cifrado en una amplia gama de Servicios de AWS y en sus aplicaciones. Puede especificar que los datos de la sesión transmitidos entre los nodos administrados y los equipos locales de los usuarios en la Cuenta de AWS se cifren con cifrado de claves de KMS. (Esto se suma al cifrado TLS 1.2/1.3 que AWS ya proporciona de forma predeterminada). Para cifrar los datos de la sesión Session Manager, cree una clave KMS simétrica mediante AWS KMS.

El cifrado de AWS KMS está disponible para `Standard_Stream`, `InteractiveCommands` y los tipos de sesión de `NonInteractiveCommands`. Para utilizar la opción de cifrar los datos de la sesión con una clave creada en AWS KMS, el nodo administrado debe tener instalada la versión 2.3.539.0 de AWS Systems Manager SSM Agent o una posterior.

> **Note**
>
> Debe permitir el cifrado de AWS KMS con el fin de restablecer las contraseñas en los nodos administrados desde la consola de AWS Systems Manager. Para obtener más información, consulte Restablecer una contraseña en un nodo administrado.

Puede utilizar una clave que haya creado en su Cuenta de AWS. También puede utilizar una clave que se haya creado en una Cuenta de AWS diferente. El creador de la clave de una Cuenta de AWS diferente debe proporcionarle los permisos necesarios para usar la clave.

Después de activar el cifrado con clave de KMS para los datos de la sesión, tanto los usuarios que inicien sesiones como los nodos administrados a los que se conecten deberán tener permiso para utilizar la clave. Usted proporciona el permiso para utilizar la clave de KMS con Session Manager a través de las políticas de AWS Identity and Access Management (IAM). Para obtener información, consulte los siguientes temas:

- Agregar permisos de AWS KMS para los usuarios de su cuenta: Ejemplos de políticas de IAM para Session Manager.
- Agregar permisos de AWS KMS para los nodos administrados de la cuenta: Paso 2: verificación o agregación de permisos de instancia para Session Manager.

Para obtener más información acerca de cómo crear y administrar las claves de KMS, consulte la Guía para desarrolladores de AWS Key Management Service.

Para obtener más información acerca de cómo utilizar la AWS CLI para activar el cifrado de datos de sesión en su cuenta con claves de KMS, consulte Creación de un documento de preferencias de Session Manager (línea de comandos) o Actualizar preferencias de Session Manager (línea de comandos).

> **Note**
>
> Se aplica un cargo por utilizar claves de KMS. Para obtener información, consulte Precios de AWS Key Management Service.

Para activar el cifrado de datos de sesión con claves de KMS (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Session Manager.
3. Elija la pestaña Preferences (Preferencias) y, a continuación, Edit (Editar).
4. Seleccione la casilla de verificación situada junto a Enable KMS encryption (Habilitar cifrado de KMS).

5. Realice una de las siguientes acciones siguientes:

   - Elija el botón que se encuentra junto a Select a KMS key in my current account (Seleccione una clave de KMS en la cuenta actual) y, a continuación, seleccione una de las claves de la lista.

     -o bien-

     Elija el botón junto a Enter a KMS key alias or KMS key ARN (Introducir un alias de clave de KMS o un ARN de clave de KMS). Ingrese de forma manual un alias de clave de KMS para una clave creada en su cuenta actual o ingrese el nombre de recurso de Amazon (ARN) de la clave perteneciente a otra cuenta. A continuación se muestran algunos ejemplos:

     - Alias de clave: `alias/my-kms-key-alias`
     - ARN de clave: `arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-12345EXAMPLE`

     -o bien-

     Elija Create new key (Crear nueva clave) para crear una nueva clave de KMS en su cuenta. Después de crear la clave nueva, vuelva a la pestaña Preferences (Preferencias) y seleccione la clave para el cifrado de los datos de la sesión en su cuenta.

   Para obtener más información acerca de cómo compartir claves, consulte Permiso para que las Cuentas de AWS externas accedan a una clave en la Guía para desarrolladores de AWS Key Management Service.

6. Seleccione Guardar.

Creación de un documento de preferencias de Session Manager (línea de comandos)

Utilice el siguiente procedimiento para crear documentos SSM que definan las preferencias para las sesiones de AWS Systems Manager Session Manager. Puede utilizar el documento para configurar las opciones de sesión, incluidos el cifrado de datos, la duración de la sesión y el registro. Por ejemplo, puede especificar si desea almacenar datos de registro de sesión en un bucket de Amazon Simple Storage Service (Amazon S3) o en un grupo de registro de Registros de Amazon CloudWatch. Puede crear documentos que definan las preferencias generales para todas las sesiones de una Cuenta de AWS y una Región de AWS, o que definan las preferencias para las sesiones individuales.

> **Note**
>
> Además, puede configurar las preferencias generales de la sesión mediante la consola del administrador de sesiones.

Los documentos utilizados para configurar las preferencias del administrador de sesiones deben tener un `sessionType` de `Standard_Stream`. Para obtener más información sobre los documentos de sesiones, consulte the section called “Esquema del documento de Session”.

Para obtener información sobre el uso de la línea de comandos para actualizar las preferencias de Session Manager existentes, consulte Actualizar preferencias de Session Manager (línea de comandos).

Para ver un ejemplo de cómo se crean las preferencias de sesión mediante AWS CloudFormation, consulte Creación de un documento de Systems Manager para preferencias de Session Manager en la Guía del usuario de AWS CloudFormation.

> **Note**
>
> Este procedimiento describe cómo crear documentos para establecer las preferencias de Session Manager a nivel de Cuenta de AWS. Para crear documentos que se utilizarán para establecer las preferencias a nivel de sesión, especifique un valor distinto de `SSM-SessionManagerRunShell` para las entradas de comandos relacionadas con el nombre del archivo.
>
> Para utilizar el documento para establecer las preferencias de las sesiones iniciadas desde AWS Command Line Interface (AWS CLI), proporcione el nombre del documento como el valor del parámetro `--document-name`. Para establecer las preferencias de las sesiones iniciadas desde la consola del administrador de sesiones, puede escribir el nombre del documento o seleccionarlo de una lista.

Para crear preferencias de Session Manager (línea de comandos)

1. Cree un archivo JSON en su equipo local con un nombre similar a `SessionManagerRunShell.json` y, a continuación, péguele el siguiente contenido.

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

También puede pasar valores a sus preferencias de sesión usando parámetros en lugar de codificar de forma rígida los valores, como se muestra en el siguiente ejemplo.

```
{
   "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. Especifique a dónde quiere enviar los datos de la sesión. Puede especificar el nombre de un bucket de S3 (con un prefijo opcional) o el nombre de un grupo de registros de los Registros de CloudWatch. Si desea cifrar aún más los datos entre el cliente local y los nodos administrados, proporcione la clave de KMS para utilizarla para el cifrado. A continuación, se muestra un ejemplo.

```
{
  "schemaVersion": "1.0",
  "description": "Document to hold regional settings for Session Manager",
  "sessionType": "Standard_Stream",
  "inputs": {
    "s3BucketName": "amzn-s3-demo-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 no desea cifrar los datos de registro de la sesión, cambie `true` por `false` en `s3EncryptionEnabled`.
> Si no va a enviar registros a un bucket de Amazon S3 ni a un grupo de registros de los Registros de CloudWatch, si no desea cifrar los datos de sesión activa ni desea activar Ejecutar como soporte para las sesiones de su cuenta, puede eliminar las líneas de esas opciones. Asegúrese de que la última línea de la sección `inputs` no termine con una coma.
> Si agrega un ID de clave de KMS para cifrar los datos de la sesión, tanto los usuarios que inician sesiones como los nodos administrados a los que se conectan deben tener permiso para utilizar la clave. Usted proporciona el permiso para utilizar la clave de KMS con Session Manager a través de las políticas de IAM. Para obtener información, consulte los siguientes temas:
>
> - Agregar permisos de AWS KMS para los usuarios de su cuenta: Ejemplos de políticas de IAM para Session Manager
> - Agregar permisos de AWS KMS para los nodos administrados de la cuenta: Paso 2: verificación o agregación de permisos de instancia para Session Manager

3. Guarde el archivo.
4. En el directorio en el que creó el archivo JSON, ejecute el siguiente comando.

   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 se ejecuta correctamente, el comando devolverá información similar a la siguiente.

```
{
    "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"
    }
}
```

Actualizar preferencias de Session Manager (línea de comandos)

El siguiente procedimiento describe cómo utilizar su herramienta de línea de comandos preferida para realizar cambios en las preferencias de AWS Systems Manager Session Manager para su

Cuenta de AWS en la Región de AWS seleccionada. Use las preferencias de Session Manager para especificar opciones para registrar los datos de la sesión en un bucket de Amazon Simple Storage Service (Amazon S3) o en un grupo de registros de Amazon CloudWatch Logs. También puede utilizar las preferencias de Session Manager para cifrar los datos de la sesión.

Para actualizar las preferencias de Session Manager (línea de comandos)

1. Cree un archivo JSON en su equipo local con un nombre similar a `SessionManagerRunShell.json` y, a continuación, péguele el siguiente contenido.

```
{
    "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. Especifique a dónde quiere enviar los datos de la sesión. Puede especificar el nombre de un bucket de S3 (con un prefijo opcional) o el nombre de un grupo de registros de los Registros de CloudWatch. Si desea cifrar aún más los datos entre el cliente local y los nodos administrados, proporcione la AWS KMS key para utilizarla para el cifrado. A continuación, se muestra un ejemplo.

```
{
  "schemaVersion": "1.0",
  "description": "Document to hold regional settings for Session Manager",
```

```
  "sessionType": "Standard_Stream",
  "inputs": {
    "s3BucketName": "amzn-s3-demo-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 no desea cifrar los datos de registro de la sesión, cambie `true` por `false` en `s3EncryptionEnabled`.

Si no va a enviar registros a un bucket de Amazon S3 ni a un grupo de registros de los Registros de CloudWatch, si no desea cifrar los datos de sesión activa ni desea activar Ejecutar como soporte para las sesiones de su cuenta, puede eliminar las líneas de esas opciones. Asegúrese de que la última línea de la sección `inputs` no termine con una coma.

Si agrega un ID de clave de KMS para cifrar los datos de la sesión, tanto los usuarios que inician sesiones como los nodos administrados a los que se conectan deben tener permiso para utilizar la clave. Usted proporciona el permiso para utilizar la clave de KMS con Session Manager a través de las políticas de AWS Identity and Access Management (IAM). Para obtener información, consulte los siguientes temas:

- Agregar permisos de AWS KMS para los usuarios de su cuenta: Ejemplos de políticas de IAM para Session Manager.
- Agregar permisos de AWS KMS para los nodos administrados de la cuenta: Paso 2: verificación o agregación de permisos de instancia para Session Manager.

3. Guarde el archivo.
4. En el directorio en el que creó el archivo JSON, ejecute el siguiente comando.

   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 se ejecuta correctamente, el comando devolverá información similar a la siguiente.

   ```
   {
       "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"
    }
}
```

Paso 5: (Opcional) Restringir el acceso a los comandos de una sesión

Puede restringir los comandos que un usuario puede ejecutar en una sesión de AWS Systems Manager Session Manager mediante un documento personalizado de tipo `Session` de AWS Systems Manager (SSM). En el documento, debe definir el comando que se ejecuta cuando el usuario inicia una sesión y los parámetros que el usuario puede proporcionar al comando. El valor del documento de `Session` `schemaVersion` debe ser 1.0 y su `sessionType` debe ser `InteractiveCommands`. Luego, puede crear políticas de AWS Identity and Access Management (IAM) que permitan a los usuarios acceder solo a los documentos de `Session` que usted defina. Para obtener más información acerca del uso de las políticas de IAM para restringir el acceso a los comandos de una sesión, consulte Ejemplos de políticas de IAM para comandos interactivos.

Los documentos con el `sessionType` de `InteractiveCommands` solo se admiten para sesiones iniciadas desde AWS Command Line Interface (AWS CLI). El usuario brinda el nombre del documento personalizado como valor del parámetro `--document-name` y proporciona cualquier valor de parámetro de comando mediante la opción `--parameters`. Para obtener más información sobre la ejecución de comandos interactivos, consulte Inicio de una sesión (comandos interactivos y no interactivos).

Utilice el siguiente procedimiento para crear un documento personalizado de SSM de tipo `Session` que defina el comando que el usuario tiene permitido ejecutar.

Restringir el acceso a los comandos de una sesión (consola)

Para restringir los comandos que un usuario puede ejecutar en una sesión de Session Manager (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. Elija Create command or session (Crear comando o sesión).

4. En Name (Nombre), ingrese un nombre descriptivo para el documento.
5. En Document type (Tipo de documento), elija Session document (Documento Session).
6. Escriba el contenido del documento para definir el comando que un usuario puede ejecutar en una sesión de Session Manager utilizando JSON o YAML, tal y como se muestra en el siguiente ejemplo.

   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. Elija Create document (Crear documento).

Restringir el acceso a los comandos de una sesión (línea de comandos)

Antes de empezar

Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI) o las Herramientas de AWS para PowerShell. Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.

Para restringir los comandos que un usuario puede ejecutar en una sesión de Session Manager (línea de comandos)

1. Cree un archivo JSON o YAML con el contenido del documento que defina el comando que un usuario puede ejecutar en una sesión de Session Manager, tal y como se muestra en el siguiente ejemplo.

   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. Ejecute los siguientes comandos para crear un documento de SSM en el que se defina el comando que el usuario puede ejecutar en una sesión de 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"
```

Parámetros de comandos interactivos y la AWS CLI

Hay una variedad de formas en las que puede proporcionar parámetros de comandos interactivos cuando utiliza la AWS CLI. Según el sistema operativo (SO) del equipo cliente que utilice para conectarse a nodos administrados con la AWS CLI, la sintaxis que proporcione para los comandos que contengan caracteres especiales o de escape puede diferir. En los siguientes ejemplos, se muestran algunas de las diferentes formas de proporcionar parámetros de comando cuando se utiliza la AWS CLI y cómo manejar los caracteres especiales o de escape.

Los parámetros almacenados en Parameter Store se pueden referenciar en la AWS CLI para obtener los parámetros de comandos, como se muestra en el siguiente ejemplo.

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}}"]}'
```

En el siguiente ejemplo, se muestra cómo puede utilizar una sintaxis abreviada con la AWS CLI para pasar parámetros.

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"
```

También puede proporcionar parámetros en JSON, tal como se muestra en el siguiente ejemplo.

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"]}'
```

Los parámetros también se pueden almacenar en un archivo JSON y proporcionarse a la AWS CLI, como se muestra en el siguiente ejemplo. Para obtener más información acerca del uso de los parámetros de la AWS CLI desde un archivo, consulte Carga de los parámetros de la AWS CLI desde un archivo en la Guía del usuario de la 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
```

También puede generar un esqueleto de la AWS CLI desde un archivo JSON de entrada, tal como se muestra en el siguiente ejemplo. Para obtener más información acerca de la generación de esqueletos de la AWS CLI desde archivos JSON de entrada, consulte Generación del esqueleto de la AWS CLI y de los parámetros de entrada desde un archivo de entrada JSON o YAML en la Guía del usuario de la AWS Command Line Interface.

```
{
    "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
```

Para los caracteres de escape que se encuentran entre comillas, debe agregar barras inversas adicionales a los caracteres de escape, como se muestra en el siguiente ejemplo.

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\""]}'
```

Para obtener más información acerca de cómo utilizar comillas con parámetros de comando en la AWS CLI, consulte Uso de comillas con cadenas en la AWS CLI en la Guía del usuario de la AWS Command Line Interface.

Ejemplos de políticas de IAM para comandos interactivos

Puede crear políticas de IAM que permitan a los usuarios acceder solo a los documentos de `Session` que usted defina. De este modo, restringirá los comandos que puede ejecutar un usuario en una sesión de Session Manager a solo aquellos comandos que estén especificados en los documentos personalizados de tipo `Session` de SSM.

Permitir que un usuario ejecute un comando interactivo en un solo nodo administrado

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"ssm:StartSession",
         "Resource":[
            "arn:aws:ec2:us-east-1:444455556666:instance/i-02573cafcfEXAMPLE",
            "arn:aws:ssm:us-east-1:444455556666:document/allowed-session-
document"
         ]
```

```
      },
      {
         "Effect": "Allow",
         "Action": ["ssmmessages:OpenDataChannel"],
         "Resource": ["arn:aws:ssm:*:*:session/${aws:userid}-*"]
      }
   ]
}
```

Permitir que un usuario ejecute un comando interactivo en todos los nodos administrados

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"ssm:StartSession",
         "Resource":[
            "arn:aws:ec2:us-east-1:444455556666:instance/*",
            "arn:aws:ssm:us-east-1:444455556666:document/allowed-session-
document"
         ]
      },
      {
         "Effect": "Allow",
         "Action": ["ssmmessages:OpenDataChannel"],
         "Resource": ["arn:aws:ssm:*:*:session/${aws:userid}-*"]
      }
   ]
}
```

Permitir que un usuario ejecute varios comandos interactivos en todos los nodos administrados

JSON

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

```
            "Resource":[
               "arn:aws:ec2:us-east-1:444455556666:instance/*",
               "arn:aws:ssm:us-east-1:444455556666:document/allowed-session-
document",
               "arn:aws:ssm:us-east-1:444455556666:document/allowed-session-
document-2"
            ]
         },
         {
            "Effect": "Allow",
            "Action": ["ssmmessages:OpenDataChannel"],
            "Resource": ["arn:aws:ssm:*:*:session/${aws:userid}-*"]
         }
      ]
}
```

Paso 6: (Opcional) Utilizar AWS PrivateLink para configurar un punto de enlace de la VPC para Session Manager

Puede mejorar aún más la posición de seguridad de los nodos administrados configurando AWS Systems Manager para que use un punto de conexión de la nube virtual privada (VPC) de interfaz. Los puntos de enlace de interfaz tienen la tecnología de AWS PrivateLink que le permite acceder de forma privada a las API de Amazon Elastic Compute Cloud (Amazon EC2) y Systems Manager mediante el uso de direcciones IP privadas.

AWS PrivateLink restringe todo el tráfico de red entre los nodos administrados, Systems Manager y Amazon EC2 a la red de Amazon. (Los nodos administrados no tienen acceso a Internet). Asimismo, no necesita una gateway de Internet ni un dispositivo NAT ni una gateway privada virtual.

Para obtener información sobre cómo crear un punto de conexión de VPC, consulte Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager.

La alternativa a usar un punto de conexión de VPC es permitir el acceso a Internet saliente en los nodos administrados. En este caso, los nodos administrados también deben permitir el tráfico saliente HTTPS (puerto 443) a los siguientes puntos de conexión:

- `ec2messages.region.amazonaws.com`
- `ssm.region.amazonaws.com`
- `ssmmessages.region.amazonaws.com`

Systems Manager utiliza el último de estos puntos de enlace, `ssmmessages.region.amazonaws.com`, para realizar llamadas desde SSM Agent hacia el servicio Session Manager en la nube.

Para utilizar características opcionales como cifrado de AWS Key Management Service (AWS KMS), streaming de registros a Amazon CloudWatch Logs (CloudWatch Logs) y envío de registros a Amazon Simple Storage Service (Amazon S3), debe permitir el tráfico saliente HTTPS (puerto 443) a los siguientes puntos de conexión:

- `kms.region.amazonaws.com`
- `logs.region.amazonaws.com`
- `s3.region.amazonaws.com`

Para obtener más información acerca de los puntos de enlace requeridos para Systems Manager, consulte Referencia: ec2messages, ssmmessages y otras operaciones de la API.

Paso 7: (Opcional) Activar o desactivar los permisos administrativos de la cuenta ssm-user

A partir de la versión 2.3.50.0 de AWS Systems Manager SSM Agent, el agente crea una cuenta de usuario local llamada `ssm-user` y la agrega a `/etc/sudoers` (Linux y macOS) o al grupo de administradores (Windows). En las versiones anteriores a la 2.3.612.0 del agente, la cuenta se crea la primera vez que SSM Agent se inicia o reinicia después de la instalación. En la versión 2.3.612.0 y posteriores, la cuenta `ssm-user` se crea la primera vez que se inicia una sesión en un nodo. Este `ssm-user` es el usuario predeterminado del sistema operativo (SO) cuando se inicia una sesión de AWS Systems Manager Session Manager. La versión 2.3.612.0 de SSM Agent se lanzó el 8 de mayo de 2019.

Si desea impedir que los usuarios de Session Manager ejecuten comandos administrativos en un nodo, puede actualizar los permisos de las cuentas `ssm-user`. También puede restaurar estos permisos después de que se hayan eliminado.

Temas

- Administración de permisos de cuentas ssm-user sudo en Linux y macOS
- Administración de los permisos de las cuentas de administrador ssm-user en Windows Server

Administración de permisos de cuentas ssm-user sudo en Linux y macOS

Utilice uno de los siguientes procedimientos para activar o desactivar los permisos sudo de las cuentas ssm-user en nodos administrados de Linux y macOS.

Use Run Command para modificar permisos sudo de ssm-user (consola)

- Utilice el procedimiento en Ejecución de comandos desde la consola con los siguientes valores:
  - En Command document (Documento Command), elija `AWS-RunShellScript`.
  - Para eliminar el acceso sudo, en el área Command parameters (Parámetros de comandos), pegue lo siguiente en el cuadro Commands (Comandos).

    ```
    cd /etc/sudoers.d
    echo "#User rules for ssm-user" > ssm-agent-users
    ```

    -o bien-

    Para restaurar el acceso sudo, en el área Command parameters (Parámetros de comandos), pegue lo siguiente en el cuadro Commands (Comandos).

    ```
    cd /etc/sudoers.d
    echo "ssm-user ALL=(ALL) NOPASSWD:ALL" > ssm-agent-users
    ```

Uso de la línea de comandos para modificar permisos sudo de ssm-user (AWS CLI)

1. Conéctese al nodo administrado y ejecute el siguiente comando.

   ```
   sudo -s
   ```

2. Cambie el directorio de trabajo con el siguiente comando.

   ```
   cd /etc/sudoers.d
   ```

3. Abra el archivo con el nombre `ssm-agent-users` para editarlo.
4. Para eliminar el acceso sudo, elimine la siguiente línea.

   ```
   ssm-user ALL=(ALL) NOPASSWD:ALL
   ```

-o bien-

Para restaurar el acceso sudo, agregue la siguiente línea.

```
ssm-user ALL=(ALL) NOPASSWD:ALL
```

5. Guarde el archivo.

Administración de los permisos de las cuentas de administrador ssm-user en Windows Server

Utilice uno de los siguientes procedimientos para activar o desactivar los permisos de administrador de las cuentas ssm-user en los nodos administrados de Windows Server.

Uso de Run Command para modificar los permisos de administrador (consola)

- Utilice el procedimiento en Ejecución de comandos desde la consola con los siguientes valores:

  En Command document (Documento Command), elija `AWS-RunPowerShellScript`.

  Para eliminar el acceso administrativo, en el área Command parameters (Parámetros de comandos), pegue lo siguiente en el cuadro Commands (Comandos).

  ```
  net localgroup "Administrators" "ssm-user" /delete
  ```

  -o bien-

  Para restaurar el acceso administrativo, en el área Command parameters (Parámetros de comandos), pegue lo siguiente en el cuadro Commands (Comandos).

  ```
  net localgroup "Administrators" "ssm-user" /add
  ```

Uso de la ventana de PowerShell o del símbolo del sistema para modificar los permisos de administrador

1. Conéctese al nodo administrado y abra la ventana de PowerShell o del símbolo del sistema.
2. Para eliminar el acceso administrativo, ejecute el siguiente comando.

   ```
   net localgroup "Administrators" "ssm-user" /delete
   ```

-o bien-

Para restaurar el acceso administrativo, ejecute el siguiente comando.

```
net localgroup "Administrators" "ssm-user" /add
```

Uso de la consola de Windows para modificar los permisos de administrador

1. Conéctese al nodo administrado y abra la ventana de PowerShell o del símbolo del sistema.
2. En la línea de comandos, ejecute `lusrmgr.msc` para abrir la consola Local Users and Groups (Grupos y usuarios locales).
3. Abra el directorio Usuarios y, a continuación, ssm-user.
4. En la pestaña Member Of (Miembro de), realice una de las siguientes acciones:
   - Para eliminar el acceso administrativo, seleccione Administradores y, a continuación, seleccione Quitar.

     -o bien-

     Para restaurar el acceso administrativo, ingrese **`Administrators`** en el cuadro de texto y, a continuación, elija Add (Agregar).
5. Seleccione Aceptar.

Paso 8: (Opcional) Permitir y controlar permisos para conexiones de SSH mediante Session Manager

Puede permitir que los usuarios de la Cuenta de AWS utilicen la AWS Command Line Interface (AWS CLI) para establecer conexiones de Secure Shell (SSH) a los nodos administrados usando AWS Systems Manager Session Manager. Los usuarios que se conectan usando SSH también pueden copiar archivos entre sus máquinas locales y nodos administrados usando el protocolo de copia segura (SCP). Puede usar esta funcionalidad para conectarse a nodos administrados sin abrir puertos de entrada o mantener hosts de bastión.

Después de permitir las conexiones de SSH, puede utilizar las políticas de AWS Identity and Access Management (IAM) a fin de permitir o denegar explícitamente las conexiones de SSH mediante Session Manager a usuarios, grupos o roles.

> **Note**
>
> El registro no está disponible para las sesiones de Session Manager que se conectan a través del reenvío de puertos o de SSH. Esto se debe a que SSH cifra todos los datos de la sesión y Session Manager solo sirve como túnel para las conexiones de SSH.

Temas

- Permitir las conexiones de SSH para Session Manager
- Control de los permisos de usuario para las conexiones de SSH mediante Session Manager

Permitir las conexiones de SSH para Session Manager

Siga los siguientes pasos para permitir las conexiones de SSH mediante Session Manager en un nodo administrado.

Para permitir las conexiones de SSH para Session Manager

1. En el nodo administrado en el que desea permitir las conexiones de SSH, realice el siguiente procedimiento:

   - Asegúrese de que se está ejecutando SSH en el nodo administrado. (Puede cerrar los puertos de entrada del nodo).
   - Asegúrese de la versión 2.3.672.0 de SSM Agent o una versión posterior esté instalada en el nodo administrado.

     Para obtener información sobre cómo instalar o actualizar el SSM Agent en un nodo administrado, consulte los siguientes temas:

     - Cómo instalar y desinstalar de forma manual SSM Agent en instancias de EC2 para Windows Server.
     - Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux
     - Instalación y desinstalación manual de SSM Agent en instancias de EC2 para macOS
     - Cómo instalar SSM Agent en nodos de Windows híbridos
     - Cómo instalar SSM Agent en nodos de Linux híbridos

> **Note**
>
> Para utilizar Session Manager con servidores en las instalaciones, dispositivos periféricos y máquinas virtuales (VM) que activó como nodos administrados, debe utilizar el nivel de instancias avanzadas. Para obtener más información acerca de las instancias avanzadas, consulte Configuración de los niveles de instancias.

2. En la máquina local desde la que desea conectarse a un nodo administrado utilizando SSH, haga lo siguiente:

   - Asegúrese de que la versión 1.1.23.0 o posterior del complemento Session Manager está instalada.

     Para obtener más información sobre la Session Manager instalación del plugin, consulte Instalación del complemento de Session Manager para la AWS CLI.

   - Actualice el archivo de configuración de SSH para permitir la ejecución de un comando de proxy que inicie una sesión de Session Manager y transfiera todos los datos a través de la conexión.

     Linux y macOS

     > **Tip**
     >
     > El archivo de configuración de SSH suele estar en `~/.ssh/config`.

     Agregue lo siguiente al archivo de configuración en el equipo local.

     ```
     # SSH over Session Manager
     Host i-* mi-*
         ProxyCommand sh -c "aws ssm start-session --target %h --document-name AWS-StartSSHSession --parameters 'portNumber=%p'"
         User ec2-user
     ```

     Windows

> **Tip**
>
> El archivo de configuración de SSH suele estar en `C:\Users\<username>\.ssh\config`.

Agregue lo siguiente al archivo de configuración en el equipo 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"
```

- Cree o verifique que tiene un certificado Privacy Enhanced Mail (un archivo PEM) o, al menos, una clave pública, que se utilizará a la hora de establecer conexiones a los nodos administrados. Debe ser una clave que ya esté asociada al nodo administrado. Se deben configurar los permisos del archivo de clave privada para que solo usted pueda leerlo. Puede utilizar el siguiente comando para configurar los permisos del archivo de clave privada para que solo usted pueda leerlo.

  ```
  chmod 400 <my-key-pair>.pem
  ```

  Por ejemplo, para una instancia de Amazon Elastic Compute Cloud (Amazon EC2), el archivo de par de claves que creó o seleccionó cuando creó la instancia. (Debe especificar la ruta al certificado o a la clave como parte del comando para iniciar una sesión. Para obtener más información acerca de cómo iniciar una sesión mediante SSH, consulte Inicio de una sesión (SSH)).

Control de los permisos de usuario para las conexiones de SSH mediante Session Manager

Después de habilitar las conexiones de SSH mediante Session Manager en un nodo administrado, puede utilizar las políticas de IAM para permitir o denegar la capacidad de establecer conexiones de SSH a usuarios, grupos o roles a través de Session Manager.

Uso de políticas de IAM para permitir las conexiones de SSH a través de Session Manager

- Utilice una de las siguientes opciones:

- Opción 1: abra la consola de IAM en https://console.aws.amazon.com/iam/.

  En el panel de navegación, elija Policies (Políticas) y actualice la política de permisos del usuario o rol al que desea permitir que inicie conexiones SSH a través de Session Manager.

  Por ejemplo, agregue el siguiente elemento a la política de inicio rápido que creó en Políticas de usuarios finales de inicio rápido de Session Manager. Reemplace cada *example resource placeholder* por su propia información.

  JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ssm:StartSession",
            "Resource": [
                "arn:aws:ec2:us-east-1:111122223333:instance/instance-id",
                "arn:aws:ssm:*:*:document/AWS-StartSSHSession"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "ssmmessages:OpenDataChannel",
            "Resource": "arn:aws:ssm:*:*:session/${aws:userid}-*"
        }
    ]
}
```

- Opción 2: adjunte una política insertada a una política de usuario mediante la AWS Management Console, la AWS CLI o la API de AWS.

  Uso del método que elija para adjuntar la declaración de política en la Opción 1 a la política para un usuario, grupo o rol de AWS.

  Para obtener información, consulte Agregagado y eliminación de permisos de identidad de IAM en la Guía del usuario de IAM.

Uso de una política de IAM para denegar las conexiones de SSH mediante Session Manager

- Utilice una de las siguientes opciones:

  - Opción 1: abra la consola de IAM en https://console.aws.amazon.com/iam/. En el panel de navegación, elija Policies (Políticas), y, a continuación, actualice la política de permisos de forma que el usuario o el rol no puedan iniciar sesiones de Session Manager.

    Por ejemplo, agregue el siguiente elemento a la política de inicio rápido que creó en Políticas de usuarios finales de inicio rápido de Session Manager.

    JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "ssm:StartSession",
            "Resource": "arn:aws:ssm:*:*:document/AWS-StartSSHSession"
        },
        {
            "Effect": "Allow",
            "Action": "ssmmessages:OpenDataChannel",
            "Resource": "arn:aws:ssm:*:*:session/${aws:userid}-*"
        }
    ]
}
```

  - Opción 2: adjunte una política insertada a una política de usuario mediante la AWS Management Console, la AWS CLI o la API de AWS.

    Uso del método que elija para adjuntar la declaración de política en la Opción 1 a la política para un usuario, grupo o rol de AWS.

    Para obtener información, consulte Agregagado y eliminación de permisos de identidad de IAM en la Guía del usuario de IAM.

Uso de Session Manager

Puede utilizar la consola de AWS Systems Manager, la consola de Amazon Elastic Compute Cloud (Amazon EC2) o la AWS Command Line Interface (AWS CLI) para iniciar sesiones que lo conecten

a los nodos administrados a los que el administrador del sistema le ha otorgado acceso mediante políticas de AWS Identity and Access Management (IAM). En función de sus permisos, también puede ver información sobre las sesiones, reanudar las sesiones inactivas cuyo tiempo de espera no haya finalizado y terminar las sesiones. Una vez que se establece una sesión, no se ve afectada por la duración de la sesión del rol de IAM. Para obtener información acerca de cómo limitar la duración de la sesión con Session Manager, consulte Especificación de un valor de tiempo de espera de sesión inactiva y Especificar la duración máxima de la sesión.

Para obtener más información acerca de las sesiones, consulte ¿Qué es una sesión?.

Temas

- Instalación del complemento de Session Manager para la AWS CLI
- Inicio de una sesión
- Finalización de una sesión
- Visualización del historial de sesiones

Instalación del complemento de Session Manager para la AWS CLI

Para iniciar sesiones de Session Manager con los nodos administrados mediante AWS Command Line Interface (AWS CLI), debe instalar el complemento de Session Manager en su equipo local. Puede instalar el complemento en versiones compatibles de Microsoft Windows Server, macOS, Linux y Ubuntu Server.

> **Note**
>
> La versión 1.16.12 o una posterior de la AWS CLI debe estar instalada en su equipo local para poder utilizar el complemento de Session Manager. Para obtener más información, consulte Instalación o actualización de la versión de AWS Command Line Interface más reciente.

Temas

- Última versión del complemento de Session Manager e historial de versiones
- Instalación del complemento de Session Manager en Windows
- Instalación del complemento de Session Manager en macOS
- Cómo instalar el complemento de Session Manager en Linux

- Verificación de la instalación del complemento de Session Manager
- El complemento Session Manager en GitHub
- (Opcional) Activación del registro del complemento de Session Manager

Última versión del complemento de Session Manager e historial de versiones

El equipo local debe ejecutar una versión compatible del complemento de Session Manager. En la actualidad, la versión más antigua compatible es la 1.1.17.0. Si ejecuta una versión anterior, es posible que las operaciones de Session Manager no se efectúen de manera correcta.

Para ver si tiene la versión más reciente, ejecute el siguiente comando en la AWS CLI.

> **Note**
>
> El comando devuelve los resultados únicamente si el complemento se encuentra en el directorio de instalación predeterminado de su sistema operativo. También puede comprobar la versión en el contenido del archivo `VERSION` en el directorio en el que ha instalado el complemento.

```
session-manager-plugin --version
```

En la siguiente tabla se muestran todas las versiones del complemento de Session Manager, así como las características y las mejoras incluidas en cada versión.

> **Important**
>
> Le recomendamos que utilice siempre la versión más reciente. La versión más reciente incluye mejoras que mejoran la experiencia de uso del complemento.

Versión	Fecha de lanzamiento	Detalles
1.2.707.0	6 de febrero de 2025	Mejora: se actualizó la versión Go a la 1.23 en el Dockerfile. Se actualizó el paso de configuración de la versión en el archivo README.
1.2.694.0	20 de noviembre de 2024	Corrección de error: se anuló el cambio que agregaba credenciales a las solicitudes de OpenDataChannel.
1.2.688.0	6 de noviembre de 2024	Esta versión quedó obsoleta el 20 de noviembre de 2024. Mejoras: • Se agregaron credenciales a las solicitudes de OpenDataChannel. • Se actualizaron los paquetes dependientes `testify` y `objx`.
1.2.677.0	10 de octubre de 2024	Mejora: se ha agregado compatibilidad para pasar la versión del complemento con las solicitudes de OpenDataChannel.
1.2.650.0	2 de julio de 2024	Mejora: se actualizó aws-sdk-go a la versión 1.54.10. Corrección de errores: se han reformulado los comentarios para la verificación de gofmt.
1.2.633.0	30 de mayo de 2024	Mejora: Se ha actualizado el Dockerfile para que utilice una imagen de Amazon Elastic Container Registry (Amazon ECR).
1.2.553.0	10 de enero de 2024	Mejora: se actualizaron aws-sdk-go y los paquetes dependientes de Golang.
1.2.536.0	4 de diciembre de 2023	Mejora: se agregó soporte para pasar una respuesta de la API StartSession como variable de entorno al complemento Session Manager.

Versión	Fecha de lanzamiento	Detalles
1.2.497.0	1 de agosto de 2023	Mejora: se actualizó el Go SDK a la versión 1.44.302.
1.2.463.0	15 de marzo de 2023	Mejora: se agregó la compatibilidad con Mac with Apple silicon para Apple Mac (M1) en el instalador agregado y firmado de macOS.
1.2.398.0	14 de octubre de 2022	Mejora: es compatible con la versión 1.17 de Golang. Actualice el ejecutor de complementos del administrador de sesiones predeterminado para macOS si desea usar python3. Actualice la ruta de importación de SSMCLI al complemento del administrador de sesiones.
1.2.339.0	16 de junio de 2022	Corrección de error: corrección del tiempo de espera de sesión inactiva para las sesiones de puertos.
1.2.331.0	27 de mayo de 2022	Corrección de error: corrección del cierre prematuro de las sesiones de puertos cuando el servidor local no se conecta antes de que se agote el tiempo de espera.
1.2.323.0	19 de mayo de 2022	Corrección de error: desactivación de mantenimiento de conexión de smux para utilizar la función de tiempo de espera de sesión inactiva.
1.2.312.0	31 de marzo de 2022	Mejora: admite más tipos de carga de mensajes de salida.
1.2.295.0	12 de enero de 2022	Solución de errores: sesiones que dejan de responder porque el cliente reenvía datos de flujo cuando el agente queda inactivo y registros incorrectos para mensajes de `start_publication` y `pause_publication` .
1.2.279.0	27 de octubre de 2021	Mejora: empaquetado zip para la plataforma Windows.

Versión	Fecha de lanzamiento	Detalles
1.2.245.0	19 de agosto de 2021	Mejora: Actualice `aws-sdk-go` a la versión más reciente (v1.40.17) para admitir el AWS IAM Identity Center.
1.2.234.0	26 de julio de 2021	Corrección de errores: gestione el escenario de sesión abruptamente terminada en el tipo de sesión interactiva.
1.2.205.0	10 de junio de 2021	Mejora: se agregó compatibilidad con el instalador firmado de macOS.
1.2.54.0	29 de enero de 2021	Mejora: se agregó compatibilidad con la ejecución de sesiones en modo de ejecución NonInteractiveCommands.
1.2.30.0	24 de noviembre de 2020	Mejora: se mejoró el rendimiento general (solo en las sesiones de reenvío de puertos).
1.2.7.0	15 de octubre de 2020	Mejora: se redujo la latencia y se mejoró el rendimiento general (solo en las sesiones de reenvío de puertos).
1.1.61.0	17 de abril de 2020	Mejora: se agregó compatibilidad de ARM para Linux y Ubuntu.
1.1.54.0	6 de enero de 2020	Corrección de errores: gestione el escenario de condición de carrera de paquetes que se descartan cuando el complemento de Session Manager no está listo.
1.1.50.0	19 de noviembre de 2019	Mejora: se ha añadido soporte para reenviar un puerto a un socket Unix local.
1.1.35.0	7 de noviembre de 2019	Mejora: (Solo sesiones de reenvío de puertos) Enviar un comando TerminateSession a SSM Agent cuando el usuario local presione `Ctrl+C`.
1.1.33.0	26 de septiembre de 2019	Mejora: (solo sesiones de reenvío de puertos) Envíe una señal de desconexión al servidor cuando el cliente interrumpa la conexión TCP.

Versión	Fecha de lanzamiento	Detalles
1.1.31.0	6 de septiembre de 2019	Mejora: actualización para mantener abierta la sesión de enrutamiento de puertos hasta que el servidor remoto cierre la conexión.
1.1.26.0	30 de julio de 2019	Mejora: actualización para limitar la velocidad de transferencia de datos durante una sesión.
1.1.23.0	9 de julio de 2019	Mejora: se agregó compatibilidad con la ejecución de sesiones de SSH mediante Session Manager.
1.1.17.0	4 de abril de 2019	Mejora: se ha añadido soporte para cifrar aún más los datos de la sesión utilizando AWS Key Management Service (AWS KMS).
1.0.37.0	20 de septiembre de 2018	Mejora: corrección de errores para la versión de Windows.
1.0.0.0	11 de septiembre de 2018	Versión inicial del complemento de Session Manager.

Instalación del complemento de Session Manager en Windows

Puede instalar el complemento de Session Manager en Windows Vista o una versión posterior con el instalador independiente.

Cuando se hayan publicado las actualizaciones, deberá repetir el proceso de instalación para obtener la versión más reciente del complemento de Session Manager.

> **Note**
>
> Observe la siguiente información.
>
> - El instalador del complemento de Session Manager necesita derechos de administrador para instalar el complemento.
> - Para obtener mejores resultados, le recomendamos iniciar las sesiones en los clientes de Windows utilizando Windows PowerShell , versión 5 o una posterior. También puede

utilizar el intérprete de comandos en Windows 10. El complemento de Session Manager solo es compatible con PowerShell y el intérprete de comandos. Es posible que las herramientas de línea de comandos de terceros no sean compatibles con el complemento.

Para instalar el complemento de Session Manager con el instalador EXE

1. Descargue el instalador mediante la siguiente dirección URL.

   ```
   https://s3.amazonaws.com/session-manager-downloads/plugin/latest/windows/
   SessionManagerPluginSetup.exe
   ```

   También puede descargar una versión comprimida del instalador utilizando la siguiente URL.

   ```
   https://s3.amazonaws.com/session-manager-downloads/plugin/latest/windows/
   SessionManagerPlugin.zip
   ```

2. Ejecute el instalador descargado y siga las instrucciones que aparecen en la pantalla. Si descargó la versión comprimida del instalador, primero debe descomprimir el instalador.

   Deje el cuadro de ubicación de la instalación en blanco para instalar el complemento en el directorio predeterminado.

   - `%PROGRAMFILES%\Amazon\SessionManagerPlugin\bin\`

3. Compruebe que la instalación se ha realizado correctamente. Para obtener más información, consulta Verificación de la instalación del complemento de Session Manager.

   > **Note**
   >
   > Si Windows no puede encontrar el ejecutable, es posible que tenga que volver a abrir el símbolo del sistema o añadir el directorio de instalación a su variable de entorno PATH manualmente. Para obtener información, consulte el tema sobre solución de problemas Complemento de Session Manager no agregado de manera automática a la ruta de la línea de comandos (Windows).

Instalación del complemento de Session Manager en macOS

Elija uno de los siguientes temas para instalar el complemento de Session Manager en macOS.

> **Note**
>
> El instalador firmado es un archivo `.pkg` firmado. El instalador agrupado utiliza un archivo `.zip`. Después de descomprimir el archivo, puede utilizar el binario para instalar el complemento.

Instalación del complemento de Session Manager en macOS con el instalador firmado

En esta sección se describe cómo instalar el complemento de Session Manager en macOS con el instalador firmado.

Para instalar el complemento de Session Manager con el instalador firmado (macOS)

1. Descargue el instalador firmado.

   x86_64

   ```
   curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/mac/
   session-manager-plugin.pkg" -o "session-manager-plugin.pkg"
   ```

   Mac con silicona de Apple

   ```
   curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/
   mac_arm64/session-manager-plugin.pkg" -o "session-manager-plugin.pkg"
   ```

2. Ejecute los comandos de instalación. Si el comando falla, verifique que la carpeta `/usr/local/bin` exista. Si no es así, créela y vuelva a ejecutar el comando.

   ```
   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. Compruebe que la instalación se ha realizado correctamente. Para obtener más información, consulta Verificación de la instalación del complemento de Session Manager.

Instalación del complemento de Session Manager en macOS

En esta sección se describe cómo instalar el complemento de Session Manager en macOS con el instalador incluido.

Important

Tenga en cuenta la siguiente información importante.

- De forma predeterminada, el instalador requiere acceso sudo para ejecutarse, ya que el script instala el complemento en el directorio del sistema `/usr/local/sessionmanagerplugin`. Si no quiere instalar el complemento mediante sudo, actualice manualmente el script del instalador para instalar el complemento en un directorio que no requiera acceso sudo.
- El instalador empaquetado no admite la instalación en rutas que contienen espacios.

Para instalar el complemento de Session Manager con el instalador empaquetado (macOS)

1. Descargue el instalador empaquetado.

   x86_64

   ```
   curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/mac/sessionmanager-bundle.zip" -o "sessionmanager-bundle.zip"
   ```

   Mac con silicona de Apple

   ```
   curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/mac_arm64/sessionmanager-bundle.zip" -o "sessionmanager-bundle.zip"
   ```

2. Descomprima el paquete.

   ```
   unzip sessionmanager-bundle.zip
   ```

3. Ejecute el comando de instalación.

   ```
   sudo ./sessionmanager-bundle/install -i /usr/local/sessionmanagerplugin -b /usr/local/bin/session-manager-plugin
   ```

   Note

   El complemento requiere Python 2.6.5 o uno posterior o Python 3.3 o uno posterior. El script de instalación se ejecuta en la versión de Python predeterminada del sistema.

Si tiene instalada una versión alternativa de Python y desea utilizarla para instalar el complemento de Session Manager, ejecute el script de instalación de esa versión mediante una ruta absoluta al ejecutable de Python. A continuación se muestra un ejemplo.

```
sudo /usr/local/bin/python3.8 sessionmanager-bundle/install -i /usr/local/
sessionmanagerplugin -b /usr/local/bin/session-manager-plugin
```

El instalador instala el complemento de Session Manager en `/usr/local/sessionmanagerplugin` y crea el symlink `session-manager-plugin` en el directorio `/usr/local/bin`. De este modo, no es necesario especificar el directorio de instalación en la variable $PATH del usuario.

Para ver una explicación de las opciones `-i` y `-b`, use la opción `-h`.

```
./sessionmanager-bundle/install -h
```

4. Compruebe que la instalación se ha realizado correctamente. Para obtener más información, consulta Verificación de la instalación del complemento de Session Manager.

> **Note**
>
> Para desinstalar el complemento, ejecute los dos siguientes comandos en el orden que aparecen.
>
> ```
> sudo rm -rf /usr/local/sessionmanagerplugin
> ```
>
> ```
> sudo rm /usr/local/bin/session-manager-plugin
> ```

Cómo instalar el complemento de Session Manager en Linux

En esta sección, se incluye información sobre la verificación de la firma del paquete del instalador del complemento de Session Manager y la instalación del complemento en las siguientes distribuciones de Linux:

- Amazon Linux 2
- AL2023
- RHEL
- Debian
- Ubuntu

Temas

- Verificación de la firma del complemento de Session Manager
- Instalar el complemento Session Manager en Amazon Linux 2, Amazon Linux 2023 y las distribuciones de Red Hat Enterprise Linux
- Instale el complemento Session Manager en Debian Server y Ubuntu Server

Verificación de la firma del complemento de Session Manager

Los paquetes de instaladores RPM y Debian del complemento de Session Manager para instancias de Linux están firmados criptográficamente. Puede usar la clave pública para verificar que el binario y el paquete del complemento sean originales y que no se hayan modificado. Si hay algún tipo de daño o alteración en el archivo, se produce un error en la verificación. Puede verificar la firma del paquete del instalador mediante la herramienta GNU Privacy Guard (GPG). La siguiente información es para las versiones del complemento de Session Manager 1.2.707.0 o posteriores.

Complete los siguientes pasos para verificar la firma del paquete del instalador del complemento de Session Manager.

Temas

- Paso 1: descargue el paquete del instalador del complemento de Session Manager
- Paso 2: descargue el archivo de firma asociado
- Paso 3: instale la herramienta GPG
- Paso 4: verifique el paquete del instalador del complemento de Session Manager en un servidor Linux

Paso 1: descargue el paquete del instalador del complemento de Session Manager

Descargue el paquete del instalador del complemento de Session Manager que quiere verificar.

Amazon Linux 2, AL2023 y los paquetes RPM de RHEL

x86_64

```
curl -o "session-manager-plugin.rpm" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/linux_64bit/session-manager-plugin.rpm"
```

x86

```
curl -o "session-manager-plugin.rpm" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/linux_32bit/session-manager-plugin.rpm"
```

ARM64

```
curl -o "session-manager-plugin.rpm" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/linux_arm64/session-manager-plugin.rpm"
```

Paquetes de Deb para Debian y Ubuntu

x86_64

```
curl -o "session-manager-plugin.deb" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/ubuntu_64bit/session-manager-plugin.deb"
```

x86

```
curl -o "session-manager-plugin.deb" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/ubuntu_32bit/session-manager-plugin.deb"
```

ARM64

```
curl -o "session-manager-plugin.deb" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/ubuntu_arm64/session-manager-plugin.deb"
```

Paso 2: descargue el archivo de firma asociado

Después de descargar el paquete del instalador, descargue el archivo de firma asociado para la verificación del paquete. Para proporcionar un nivel adicional de protección contra la copia o el uso no autorizados del archivo binario del complemento de Session Manager incluido en el paquete,

también ofrecemos firmas binarias, que puede usar para validar archivos binarios individuales. Puede optar por usar estas firmas binarias en función de sus necesidades de seguridad.

Amazon Linux 2, AL2023 y paquetes de firma de RHEL

x86_64

Paquete:

```
curl -o "session-manager-plugin.rpm.sig" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/linux_64bit/session-manager-plugin.rpm.sig"
```

Binario:

```
curl -o "session-manager-plugin.sig" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/linux_64bit/session-manager-plugin.sig"
```

x86

Paquete:

```
curl -o "session-manager-plugin.rpm.sig" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/linux_32bit/session-manager-plugin.rpm.sig"
```

Binario:

```
curl -o "session-manager-plugin.sig" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/linux_32bit/session-manager-plugin.sig"
```

ARM64

Paquete:

```
curl -o "session-manager-plugin.rpm.sig" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/linux_arm64/session-manager-plugin.rpm.sig"
```

Binario:

```
curl -o "session-manager-plugin.sig" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/linux_arm64/session-manager-plugin.sig"
```

Paquetes de firmas Deb para Debian y Ubuntu

x86_64

Paquete:

```
curl -o "session-manager-plugin.deb.sig" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/ubuntu_64bit/session-manager-plugin.deb.sig"
```

Binario:

```
curl -o "session-manager-plugin.sig" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/ubuntu_64bit/session-manager-plugin.sig"
```

x86

Paquete:

```
curl -o "session-manager-plugin.deb.sig" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/ubuntu_32bit/session-manager-plugin.deb.sig"
```

Binario:

```
curl -o "session-manager-plugin.sig" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/ubuntu_32bit/session-manager-plugin.sig"
```

ARM64

Paquete:

```
curl -o "session-manager-plugin.deb.sig" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/ubuntu_arm64/session-manager-plugin.deb.sig"
```

Binario:

```
curl -o "session-manager-plugin.sig" "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/ubuntu_arm64/session-manager-plugin.sig"
```

Paso 3: instale la herramienta GPG

Para verificar la firma del complemento de Session Manager, debe tener la herramienta GNU Privacy Guard (GPG) instalada en el sistema. El proceso de verificación requiere el uso de la versión 2.1 o posterior de GPG. Para comprobar su versión de GPG, ejecute el siguiente comando:

```
gpg --version
```

Si su versión de GPG es anterior a la 2.1, actualícela antes de continuar con el proceso de verificación. En la mayoría de los sistemas, puede actualizar la herramienta GPG con su administrador de paquetes. Por ejemplo, en sistemas de Amazon Linux y RHEL, puede usar el siguiente comando:

```
sudo yum update
sudo yum install gnupg2
```

En los sistemas Ubuntu o Debian, puede usar los siguientes comandos:

```
sudo apt-get update
sudo apt-get install gnupg2
```

Asegúrese de tener la versión de GPG requerida antes de continuar con el proceso de verificación.

Paso 4: verifique el paquete del instalador del complemento de Session Manager en un servidor Linux

Use el siguiente procedimiento para verificar el paquete del instalador del complemento de Session Manager en un servidor Linux.

> **Note**
>
> Amazon Linux 2 no es compatible con la versión 2.1 o superior de la herramienta GPG. Si el siguiente procedimiento no funciona en las instancias de Amazon Linux 2, verifique la firma en una plataforma diferente antes de instalarla en las instancias de Amazon Linux 2.

1. Copie la siguiente clave pública y guárdela en un archivo denominado “session-manager-plugin.gpg”.

   ```
   -----BEGIN PGP PUBLIC KEY BLOCK-----
   ```

```
mFIEZ5ERQxMIKoZIzj0DAQcCAwQjuZy+IjFoYg57sLTGhF3aZLBaGpzB+gY6j7Ix
P7NqbpXyjVj8a+dy79gSd64OEaMxUb7vw/jug+CfRXwVGRMNtIBBV1MgU1NNIFNl
c3Npb24gTWFuYWdlciA8c2Vzc2lvbi1tYW5hZ2VyLXBsdWdpbi1zaWduZXJAYW1h
em9uLmNvbT4gKEFXUyBTeXN0ZW1zIE1hbmFnZXIgU2Vzc2lvbiBNYW5hZ2VyIFBs
dWdpbiBMaW51eCBTaWduZXIgS2V5KYkBAAQQEwgAqAUCZ5ERQ4EcQVdTIFNTTSBT
ZXNzaW9uIE1hbmFnZXIgPHNlc3Npb24tbWFuYWdlci1wbHVnaW4tc2lnbmVyQGFt
YXpvbi5jb20+IChBV1MgU3lzdGVtcyBNYW5hZ2VyIFNlc3Npb24gTWFuYWdlciBQ
bHVnaW4gTGludXggU2lnbmVyIEtleSkWIQR5WWNxJM4JOtUB1HosTUr/b2dX7gIe
AwIbAwIVCAAKCRAsTUr/b2dX7rO1AQCa1kig3lQ78W/QHGU76uHx3XAyv0tfpE9U
oQBCIwFLSgEA3PDHt3lZ+s6m9JLGJsy+Cp5ZFzpiF6RgluR/2gA861M=
=2DQm
-----END PGP PUBLIC KEY BLOCK-----
```

2. Importe la clave pública en su llavero. El valor de clave devuelto debe ser 2C4D4AFF6F6757EE.

```
$ gpg --import session-manager-plugin.gpg
gpg: key 2C4D4AFF6F6757EE: public key "AWS SSM Session Manager <session-manager-
plugin-signer@amazon.com> (AWS Systems Manager Session Manager Plugin Linux Signer
 Key)" imported
gpg: Total number processed: 1
gpg:               imported: 1
```

3. Ejecute el siguiente comando para verificar la huella digital.

```
gpg --fingerprint 2C4D4AFF6F6757EE
```

La huella digital del resultado del comando debe coincidir con lo siguiente.

```
7959 6371 24CE 093A D501 D47A 2C4D 4AFF 6F67 57EE
```

```
pub   nistp256 2025-01-22 [SC]
      7959 6371 24CE 093A D501  D47A 2C4D 4AFF 6F67 57EE
uid           [ unknown] AWS SSM Session Manager <session-manager-plugin-
signer@amazon.com> (AWS Systems Manager Session Manager Plugin Linux Signer Key)
```

Si la huella digital no coincide, no instale el complemento. Ponte en contacto con AWS Support.

4. Verifique la firma del paquete del instalador. Reemplace *signature-filename* y *downloaded-plugin-filename* por los valores que especificó cuando descargó el archivo

de firma y session-manager-plugin, como se muestra en la tabla que figura más arriba en este tema.

```
gpg --verify signature-filename downloaded-plugin-filename
```

Por ejemplo, para la arquitectura x86_64 en Amazon Linux 2, el comando es el siguiente:

```
gpg --verify session-manager-plugin.rpm.sig session-manager-plugin.rpm
```

Este comando devuelve un resultado similar al siguiente.

```
gpg: Signature made Mon Feb 3 20:08:32 2025 UTC gpg: using ECDSA key
 2C4D4AFF6F6757EE
gpg: Good signature from "AWS Systems Manager Session Manager <session-manager-
plugin-signer@amazon.com> (AWS Systems Manager Session Manager Plugin Linux Signer
 Key)" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg: There is no indication that the signature belongs to the owner.
Primary key fingerprint: 7959 6371 24CE 093A D501 D47A 2C4D 4AFF 6F67 57EE
```

Si el resultado incluye la expresión `BAD signature`, compruebe si ha realizado el procedimiento correctamente. Si sigue recibiendo esta respuesta, comuníquese con AWS Support y no instale el paquete. El mensaje de advertencia sobre la confianza no significa que la firma no sea válida, sino que no se ha verificado la clave pública. Una clave solo es de confianza si la ha firmado usted o alguien en quien confíe. Si el resultado incluye la frase `Can't check signature: No public key`, verifique que haya descargado la versión 1.2.707.0 o posterior del complemento de Session Manager.

Instalar el complemento Session Manager en Amazon Linux 2, Amazon Linux 2023 y las distribuciones de Red Hat Enterprise Linux

Siga el siguiente procedimiento para instalar el complemento Session Manager en Amazon Linux 2, Amazon Linux 2023 (AL2023) y las distribuciones de RHEL.

> **Note**
>
> El complemento Session Manager no es compatible en Amazon Linux 1. Es compatible en Amazon Linux 2 o posterior.

1. Descargue el paquete RPM del complemento de Session Manager.

   x86_64

   En Amazon Linux 2 y RHEL 7, ejecute el siguiente comando:

   ```
   sudo yum install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
   latest/linux_64bit/session-manager-plugin.rpm
   ```

   En AL2023 y RHEL 8 y 9, ejecute el siguiente comando:

   ```
   sudo dnf install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
   latest/linux_64bit/session-manager-plugin.rpm
   ```

   x86

   En RHEL 7, ejecute el siguiente comando:

   ```
   sudo yum install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
   latest/linux_32bit/session-manager-plugin.rpm
   ```

   En RHEL, 8 y 9, ejecute el siguiente comando:

   ```
   sudo dnf install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
   latest/linux_32bit/session-manager-plugin.rpm
   ```

   ARM64

   En Amazon Linux 2 y RHEL 7, ejecute el siguiente comando:

   ```
   sudo yum install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
   latest/linux_arm64/session-manager-plugin.rpm
   ```

   En AL2023 y RHEL 8 y 9, ejecute el siguiente comando:

   ```
   sudo dnf install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
   latest/linux_arm64/session-manager-plugin.rpm
   ```

2. Compruebe que la instalación se ha realizado correctamente. Para obtener más información, consulta Verificación de la instalación del complemento de Session Manager.

> **Note**
>
> Si quiere desinstalar el complemento, ejecute `sudo yum erase session-manager-plugin -y`

Instale el complemento Session Manager en Debian Server y Ubuntu Server

1. Descargue el paquete deb del complemento de 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. Ejecute el comando de instalación.

   ```
   sudo dpkg -i session-manager-plugin.deb
   ```

3. Compruebe que la instalación se ha realizado correctamente. Para obtener más información, consulta Verificación de la instalación del complemento de Session Manager.

> **Note**
>
> Si alguna vez desea desinstalar el complemento, ejecute `sudo dpkg -r session-manager-plugin`

Verificación de la instalación del complemento de Session Manager

Ejecute los siguientes comandos para verificar que el complemento de Session Manager se instaló correctamente.

```
session-manager-plugin
```

Si la instalación se realizó de forma correcta, se devuelve el siguiente mensaje.

```
The Session Manager plugin is installed successfully. Use the AWS CLI to start a
 session.
```

También puede probar la instalación ejecutando el comando start-session en la AWS Command Line Interface (AWS CLI). En el siguiente comando, reemplace *instance-id* por su propia información.

```
aws ssm start-session --target instance-id
```

Este comando solo funcionará si ha instalado y configurado la AWS CLI, y si el administrador de Session Manager le ha concedido los permisos de IAM necesarios para acceder al nodo administrado de destino con Session Manager.

El complemento Session Manager en GitHub

El código fuente del complemento Session Manager está disponible en GitHub para que pueda adaptar el complemento a sus necesidades. Le recomendamos enviar solicitudes de inserción para los cambios que le gustaría que incluyamos. No obstante, Amazon Web Services no admite la ejecución de copias modificadas de este software.

(Opcional) Activación del registro del complemento de Session Manager

El complemento de Session Manager incluye una opción para permitir el registro de las sesiones que ejecute. De forma predeterminada, el registro está desactivado.

Si permite el registro, el complemento de Session Manager creará archivos de registros para la actividad de la aplicación (`session-manager-plugin.log`) y los errores (`errors.log`) en su equipo local.

Temas

- Activación del registro del complemento de Session Manager (Windows)

- Habilitación del registro del complemento de Session Manager (Linux y macOS)

Activación del registro del complemento de Session Manager (Windows)

1. Localice el archivo `seelog.xml.template` del complemento.

   La ubicación predeterminada es `C:\Program Files\Amazon\SessionManagerPlugin\seelog.xml.template`.

2. Cambie el nombre del archivo a `seelog.xml`.
3. Abra el archivo y cambie `minlevel="off"` a `minlevel="info"` o `minlevel="debug"`.

   > **Note**
   >
   > De forma predeterminada, las entradas de registro sobre la apertura de un canal de datos y la reconexión de sesiones se registran en el nivel DE INFORMACIÓN. Las entradas de flujos de datos (paquetes y confirmación) se registran en el nivel DE DEPURACIÓN.

4. Cambie otras opciones de configuración que desea modificar. Entre las opciones que se pueden cambiar se encuentran las siguientes:

   - Nivel de depuración: puede cambiar el nivel de depuración de `formatid="fmtinfo"` a `formatid="fmtdebug"`.
   - Opciones de archivos de registro: puede realizar cambios en las opciones de archivos de registro, como la ubicación en la que se almacenan los registros, con la excepción de los nombres de los archivos de registro.

     > **Important**
     >
     > No cambie los nombres de los archivos, si no, el registro no funcionará correctamente.

     ```
     <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. Guarde el archivo.

Habilitación del registro del complemento de Session Manager (Linux y macOS)

1. Localice el archivo `seelog.xml.template` del complemento.

   La ubicación predeterminada es `/usr/local/sessionmanagerplugin/seelog.xml.template`.

2. Cambie el nombre del archivo a `seelog.xml`.
3. Abra el archivo y cambie `minlevel="off"` a `minlevel="info"` o `minlevel="debug"`.

> **Note**
>
> De forma predeterminada, las entradas de registro sobre la apertura de canales de datos y la reconexión de sesiones se registran en el nivel DE INFORMACIÓN. Las entradas de flujos de datos (paquetes y confirmación) se registran en el nivel DE DEPURACIÓN.

4. Cambie otras opciones de configuración que desea modificar. Entre las opciones que se pueden cambiar se encuentran las siguientes:

   - Nivel de depuración: puede cambiar el nivel de depuración de `formatid="fmtinfo"` a `outputs formatid="fmtdebug"`.
   - Opciones de archivos de registro: puede realizar cambios en las opciones de archivos de registro, como la ubicación en la que se almacenan los registros, con la excepción de los nombres de los archivos de registro.

> **Important**
>
> No cambie los nombres de los archivos, si no, el registro no funcionará correctamente.

```
<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 utiliza el directorio predeterminado especificado para almacenar registros, debe ejecutar comandos de sesión usando sudo o proporcionar el directorio donde el complemento instaló todos los permisos de lectura y escritura. Para eludir estas restricciones, cambie la ubicación donde se almacenan los registros.

5. Guarde el archivo.

Inicio de una sesión

Puede utilizar la consola de AWS Systems Manager, la consola de Amazon Elastic Compute Cloud (Amazon EC2), la AWS Command Line Interface (AWS CLI) o SSH para iniciar una sesión.

Temas

- Inicio de una sesión (consola de Systems Manager)
- Inicio de una sesión (consola de Amazon EC2)
- Inicio de una sesión (AWS CLI)
- Inicio de una sesión (SSH)
- Inicio de una sesión (enrutamiento de puertos)
- Inicio de una sesión (reenvío de puertos a host remoto)
- Inicio de una sesión (comandos interactivos y no interactivos)

Inicio de una sesión (consola de Systems Manager)

Puede utilizar la consola de AWS Systems Manager para iniciar una sesión con un nodo administrado en la cuenta.

> **Note**
>
> Antes de iniciar una sesión, asegúrese de haber completado los pasos de configuración de Session Manager. Para obtener más información, consulta Configuración de Session Manager.

Para iniciar una sesión (consola de Systems Manager)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Session Manager.
3. Haga clic en Start session (Iniciar sesión).
4. (Opcional) Ingrese una descripción de la sesión en el campo Motivo de la sesión.
5. En Instancias de destino, elija el botón de opción situado a la izquierda del nodo administrado al que desea conectarse.

   Si el nodo que desea no está en la lista o si selecciona un nodo y recibe un error de configuración, consulte Nodo administrado no disponible o no configurado para Session Manager para seguir los pasos de solución de problemas.
6. Seleccione Iniciar sesión para iniciar la sesión de forma inmediata.

   -o bien-

   Elija Siguiente para ver las opciones de sesión.
7. (Opcional) En Documento de sesión, seleccione el documento que desea ejecutar cuando se inicie la sesión. Si el documento admite parámetros de tiempo de ejecución, puede introducir uno o más valores separados por comas en cada campo de parámetro.
8. Elija Siguiente.
9. Haga clic en Start session (Iniciar sesión).

Una vez establecida la conexión, puede ejecutar comandos bash (Linux y macOS) o comandos PowerShell (Windows), tal como lo haría con cualquier otro tipo de conexión.

> **Important**
>
> Si desea permitir a los usuarios especificar un documento al iniciar las sesiones en la consola de Session Manager, tenga en cuenta lo siguiente:
>
> - Debe conceder a los usuarios los permisos `ssm:GetDocument` y `ssm:ListDocuments` en su política de IAM. Para obtener más información, consulte Conceder acceso a documentos de sesión personalizados en la consola.

- La consola solo admite los documentos de Session que tengan el `sessionType` definido como `Standard_Stream`. Para obtener más información, consulte Esquema del documento de Session.

Inicio de una sesión (consola de Amazon EC2)

Puede utilizar la consola de Amazon Elastic Compute Cloud (Amazon EC2) para iniciar una sesión con una instancia de su cuenta.

> **Note**
>
> Si recibe un mensaje de error que indica que no está autorizado para llevar a cabo una o más acciones de Systems Manager (`ssm:command-name`), debe contactar con su administrador para recibir ayuda. El gestionador es la persona que le proporcionó las credenciales de inicio de sesión. Pídale a esa persona que actualice las políticas de modo que le permitan iniciar sesiones desde la consola de Amazon EC2. Si es administrador, consulte Ejemplos de políticas de IAM para Session Manager para obtener más información.

Para iniciar una sesión (consola de Amazon EC2)

1. Abra la consola de Amazon EC2 en https://console.aws.amazon.com/ec2/.
2. En el panel de navegación, seleccione Instances (Instancia[s]).
3. Seleccione la instancia y elija Connect (Conectar).
4. Para Connection Method (Método de conexión), seleccione Session Manager.
5. Elija Conectar.

Una vez establecida la conexión, puede ejecutar comandos bash (Linux y macOS) o comandos PowerShell (Windows), tal como lo haría con cualquier otro tipo de conexión.

Inicio de una sesión (AWS CLI)

Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

Antes de iniciar una sesión, asegúrese de haber completado los pasos de configuración de Session Manager. Para obtener más información, consulta Configuración de Session Manager.

Para utilizar la AWS CLI para ejecutar comandos de sesión, el complemento de Session Manager también debe estar instalado en su equipo local. Para obtener más información, consulta Instalación del complemento de Session Manager para la AWS CLI.

Para iniciar una sesión mediante la AWS CLI, ejecute el siguiente comando y reemplace *instance-id* por su propia información.

```
aws ssm start-session \
    --target instance-id
```

Para obtener más información sobre otras opciones que puede utilizar con el comando start-session, consulte start-session en la sección de AWS Systems Manager en la Referencia de comandos de la AWS CLI.

Inicio de una sesión (SSH)

Para iniciar una sesión de SSH de Session Manager, la versión 2.3.672.0 o una versión posterior de SSM Agent debe estar instalada en el nodo administrado.

Requisitos de conexión de SSH

Tome nota de los siguientes requisitos y limitaciones de las conexiones de sesión mediante SSH:

- El nodo administrado de destino debe estar configurado para admitir conexiones SSH. Para obtener más información, consulte (Opcional) Habilitación y control de permisos para conexiones de SSH mediante Session Manager.
- Debe conectarse utilizando la cuenta del nodo administrado asociada al certificado Privacy Enhanced Mail (PEM), no la cuenta `ssm-user` que se utiliza para otros tipos de conexiones de sesión. Por ejemplo, en las instancias de EC2 de Linux y macOS, el usuario predeterminado es `ec2-user`. Para obtener información sobre cómo identificar al usuario predeterminado de cada tipo de instancia, consulte Obtención de información sobre una instancia en la Guía del usuario de Amazon EC2.
- El registro no está disponible para las sesiones de Session Manager que se conectan a través del reenvío de puertos o de SSH. Esto se debe a que SSH cifra todos los datos de la sesión y Session Manager solo sirve como túnel para las conexiones de SSH.

> **Note**
>
> Antes de iniciar una sesión, asegúrese de haber completado los pasos de configuración de Session Manager. Para obtener más información, consulta Configuración de Session Manager.

Para iniciar una sesión con SSH, ejecute el siguiente comando. Reemplace cada *example resource placeholder* por su propia información.

```
ssh -i /path/my-key-pair.pem username@instance-id
```

> **Tip**
>
> Cuando inicia una sesión con SSH, puede copiar archivos locales al nodo administrado de destino con el siguiente formato de comando.
>
> ```
> scp -i /path/my-key-pair.pem /path/ExampleFile.txt username@instance-id:~
> ```

Para obtener más información sobre otras opciones que puede utilizar con el comando start-session, consulte start-session en la sección de AWS Systems Manager en la Referencia de comandos de la AWS CLI.

Inicio de una sesión (enrutamiento de puertos)

Para iniciar una sesión de reenvío de puertos de Session Manager, el nodo administrado debe tener instalada la versión 2.3.672.0 de SSM Agent o una posterior.

> **Note**
>
> Antes de iniciar una sesión, asegúrese de haber completado los pasos de configuración de Session Manager. Para obtener más información, consulta Configuración de Session Manager.
> Para utilizar la AWS CLI para ejecutar comandos de sesión, debe instalar el complemento de Session Manager en su equipo local. Para obtener más información, consulta Instalación del complemento de Session Manager para la AWS CLI.

Según el sistema operativo y la herramienta de línea de comandos, la colocación de comillas puede variar, y es posible que se requieran caracteres de escape.

Para iniciar una sesión de reenvío de puertos, ejecute el siguiente comando desde la CLI. Reemplace cada *example resource placeholder* por su propia información.

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"
```

El valor `portNumber` representa el puerto remoto del nodo administrado al que desea que se redirija el tráfico de la sesión. Por ejemplo, puede especificar el puerto `3389` para conectarse a un nodo de Windows mediante el protocolo de escritorio remoto (RDP). Si no especifica el parámetro `portNumber`, Session Manager utiliza `80` como el valor predeterminado.

`localPortNumber` es el puerto del equipo local donde comienza el tráfico, por ejemplo `56789`. Este valor es lo que se ingresa cuando se conecta a un nodo administrado mediante un cliente. Por ejemplo, **`localhost:56789`**.

Para obtener más información sobre otras opciones que puede utilizar con el comando start-session, consulte start-session en la sección de AWS Systems Manager en la Referencia de comandos de la AWS CLI.

Para obtener más información acerca de las sesiones de reenvío de puertos, consulte Port Forwarding Using AWS Systems ManagerSession Manager en el Blog de noticias de AWS.

Inicio de una sesión (reenvío de puertos a host remoto)

Para iniciar una sesión de reenvío de puertos de Session Manager a un host remoto, el nodo administrado debe tener instalada la versión 3.1.1374.0 o posterior de SSM Agent. No se requiere que Systems Manager administre el host remoto.

> **Note**
>
> Antes de iniciar una sesión, asegúrese de haber completado los pasos de configuración de Session Manager. Para obtener más información, consulta Configuración de Session Manager.
>
> Para utilizar la AWS CLI para ejecutar comandos de sesión, debe instalar el complemento de Session Manager en su equipo local. Para obtener más información, consulta Instalación del complemento de Session Manager para la AWS CLI.
>
> Según el sistema operativo y la herramienta de línea de comandos, la colocación de comillas puede variar, y es posible que se requieran caracteres de escape.

Para iniciar una sesión de reenvío de puertos, ejecute el siguiente comando desde la AWS CLI. Reemplace cada *example resource placeholder* con su propia información.

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"
```

El valor `host` representa el nombre de host o la dirección IP del host remoto al que desea conectarse. Se siguen aplicando los requisitos generales de conectividad y resolución de nombres entre el nodo administrado y el host remoto.

El valor `portNumber` representa el puerto remoto del nodo administrado al que desea que se redirija el tráfico de la sesión. Por ejemplo, puede especificar el puerto `3389` para conectarse a un nodo de Windows mediante el protocolo de escritorio remoto (RDP). Si no especifica el parámetro `portNumber`, Session Manager utiliza `80` como el valor predeterminado.

`localPortNumber` es el puerto del equipo local donde comienza el tráfico, por ejemplo `56789`. Este valor es lo que se ingresa cuando se conecta a un nodo administrado mediante un cliente. Por ejemplo, **`localhost:56789`**.

Para obtener más información sobre otras opciones que puede utilizar con el comando start-session, consulte start-session en la sección de AWS Systems Manager en la Referencia de comandos de la AWS CLI.

Iniciar una sesión con una tarea de Amazon ECS

Session Manager permite iniciar una sesión de reenvío de puertos con una tarea dentro de un clúster de Amazon Elastic Container Service (Amazon ECS). Para ello, debe actualizar el rol de tarea en IAM para incluir los siguientes permisos:

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
       {
       "Effect": "Allow",
       "Action": [
            "ssmmessages:CreateControlChannel",
            "ssmmessages:CreateDataChannel",
            "ssmmessages:OpenControlChannel",
            "ssmmessages:OpenDataChannel"
       ],
      "Resource": "*"
      }
   ]
}
```

Para iniciar una sesión de reenvío de puertos con una tarea de Amazon ECS, ejecute el siguiente comando desde la AWS CLI. Reemplace cada *example resource placeholder* con su propia información.

> **Note**
>
> Elimine los símbolos < y > del parámetro `target`. Estos símbolos se proporcionan únicamente para que el lector los aclare.

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"
```

Inicio de una sesión (comandos interactivos y no interactivos)

Antes de iniciar una sesión, asegúrese de haber completado los pasos de configuración de Session Manager. Para obtener más información, consulta Configuración de Session Manager.

Para utilizar la AWS CLI para ejecutar comandos de sesión, el complemento de Session Manager también debe estar instalado en su equipo local. Para obtener más información, consulta Instalación del complemento de Session Manager para la AWS CLI.

Para iniciar una sesión de comandos interactivos, ejecute el siguiente comando. Reemplace cada *example resource placeholder* por su propia información.

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"
```

Para obtener más información sobre otras opciones que puede utilizar con el comando start-session, consulte start-session en la sección de AWS Systems Manager en la Referencia de comandos de la AWS CLI.

Más información

- Uso del reenvío de puertos en AWS Systems Manager Session Manager para conectarse a hosts remotos
- Reenvío de puertos de instancias Amazon EC2 con AWS Systems Manager
- Administre los recursos de AWS administrados de Microsoft AD con el reenvío de puertos Session Manager
- Port Forwarding Using AWS Systems ManagerSession Manager en el Blog de noticias de AWS

Finalización de una sesión

Puede utilizar la consola de AWS Systems Manager o la AWS Command Line Interface (AWS CLI) para terminar una sesión que haya iniciado en la cuenta. Si toca en el botón Terminar de una sesión en la consola o llama a la acción de la API TerminateSession por medio de la AWS CLI, Session Manager finaliza permanentemente la sesión y cierra la conexión de datos entre el cliente Session Manager y SSM Agent en el nodo administrado. No puede reanudar una sesión terminada.

Si no hay actividad del usuario en una sesión abierta durante 20 minutos, el estado de inactividad acciona un tiempo de espera. Session Manager no llama a `TerminateSession`,pero sí cierra el canal subyacente. No puede reanudar una sesión cerrada debido a un tiempo de espera de inactividad.

Recomendamos siempre cerrar una sesión de forma explícita con el comando `terminate-session`, cuando se utilice el AWS CLI, o el botón Finalizar cuando se utilice la consola. (Los

botones Finalizar se encuentran tanto en la ventana de la sesión como en la página principal de la consola Session Manager). Si solo cierra un navegador o una ventana de comandos, la sesión permanecerá como Activa en la consola durante 30 días. Cuando no cierra una sesión de forma explícita o cuando la sesión termina por inactividad, todos los procesos que estaban en ejecución en el nodo administrado en ese momento seguirán ejecutándose.

Temas

- Finalización de una sesión (consola)
- Finalización de una sesión (AWS CLI)

Finalización de una sesión (consola)

Puede utilizar la consola de AWS Systems Manager para terminar una sesión en la cuenta.

Para terminar una sesión (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Session Manager.
3. En Sessions (Sesiones), elija la opción que aparece a la izquierda de la sesión que desea terminar.
4. Elija Finalizar.

Finalización de una sesión (AWS CLI)

Para terminar una sesión con la AWS CLI, ejecute el siguiente comando. Reemplace *session-id* por su propia información.

```
aws ssm terminate-session \
    --session-id session-id
```

Para obtener más información sobre el comando terminate-session, consulte terminate-session en la sección AWS Systems Manager de la Referencia de comando de la AWS CLI.

Visualización del historial de sesiones

Puede usar la consola de AWS Systems Manager o la AWS Command Line Interface (AWS CLI) para ver información acerca de las sesiones de su cuenta. En la consola, puede ver los detalles de la sesión, como los siguientes:

- El ID de la sesión
- Qué usuario se ha conectado a un nodo administrado a través de una sesión
- El ID del nodo administrado
- Cuándo la sesión comenzó y terminó
- El estado de la sesión
- la ubicación especificada para almacenar los registros de la sesión (si está habilitado)

Al usar la AWS CLI, puede ver una lista de sesiones de su cuenta, pero no los detalles adicionales que están disponibles en la consola.

Para obtener más información acerca del historial de sesiones de registro, consulte Habilitar y deshabilitar el registro de la sesión.

Temas

- Visualización del historial de sesiones (consola)
- Visualización del historial de sesiones (AWS CLI)

Visualización del historial de sesiones (consola)

Puede usar la consola de AWS Systems Manager para ver los detalles de las sesiones de su cuenta.

Para ver el historial de sesiones (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Session Manager.
3. Elija la pestaña Session history (Historial de sesiones).

   -o bien-

Si la página principal de Session Manager se abre primero, elija Configurar preferencias y, a continuación, elija la pestaña Historial de sesiones.

Visualización del historial de sesiones (AWS CLI)

Para ver una lista de las sesiones de su cuenta con la AWS CLI, ejecute el siguiente comando.

```
aws ssm describe-sessions \
    --state History
```

> **Note**
>
> Este comando solo devuelve los resultados de las conexiones a destinos iniciados mediante Session Manager. No muestra las conexiones realizadas a través de otros medios, como el Protocolo de escritorio remoto (RDP) o el Protocolo de shell seguro (SSH).

Para obtener más información sobre otras opciones que puede utilizar con el comando describe-sessions, consulte describe-sessions en la sección sobre AWS Systems Manager de la Referencia de comandos de la AWS CLI.

Registro de la actividad de la sesión

Además de proporcionar información acerca de las sesiones actuales y completadas en la consola de Systems Manager, el Session Manager proporciona la posibilidad de registrar la actividad de las sesiones en la Cuenta de AWS con AWS CloudTrail.

CloudTrail registra las llamadas a la API de la sesión realizadas a través de la consola de Systems Manager, la AWS Command Line Interface (AWS CLI) y el SDK de Systems Manager. Puede ver la información en la consola de CloudTrail o almacenarla en un bucket de Amazon Simple Storage Service (Amazon S3) especificado. Se utiliza un bucket de Amazon S3 para todos los registros de CloudTrail de su cuenta. Para obtener más información, consulte Registro de llamadas a la API de AWS Systems Manager con AWS CloudTrail.

> **Note**
>
> Para llevar a cabo un análisis periódico e histórico de sus archivos de registro, considere la posibilidad de consultar los registros de CloudTrail mediante CloudTrail Lake o una tabla

que usted mantenga. Para obtener más información, consulte Consulta de registros de AWS CloudTrail en la Guía del usuario de AWS CloudTrail.

Supervisión de la actividad de la sesión con Amazon EventBridge (consola)

Con EventBridge, puede configurar reglas para detectar cuándo se producen cambios en los recursos de AWS. Puede crear una regla para detectar cuándo un usuario de su organización inicia o termina una sesión y, a continuación, por ejemplo, recibir una notificación a través de Amazon SNS sobre el evento.

La compatibilidad de EventBridge con Session Manager se basa en los registros de las operaciones de la API que CloudTrail registró. (Puede usar la integración de CloudTrail a EventBridge para responder a la mayoría de los eventos de AWS Systems Manager). Las acciones que se llevan a cabo dentro de una sesión, como un comando `exit`, que no hacen una llamada a la API no son detectadas por EventBridge.

En los siguientes pasos se describe cómo iniciar las notificaciones a través de Amazon Simple Notification Service (Amazon SNS) cuando se produce un evento de la API de Session Manager, como StartSession.

Para supervisar la actividad de la sesión con Amazon EventBridge (consola)

1. Cree un tema de Amazon SNS que se utilice para enviar notificaciones cuando se produzca un evento de Session Manager del que desea realizar un seguimiento.

   Para obtener más información, consulte Creación de un tema en la Guía para desarrolladores de Amazon Simple Notification Service.
2. Cree una regla de EventBridge para invocar el destino de Amazon SNS para el tipo de evento de Session Manager del que desea realizar un seguimiento.

   Para obtener más información sobre cómo crear la regla, consulte Creating Amazon EventBridge rules that react to events en la Guía del usuario de Amazon EventBridge.

   Cuando siga los pasos para crear la regla, realice las siguientes selecciones:

   - En AWS service (Servicio de), elija Systems Manager.
   - En Tipo de evento, elija Llamada a la API de AWS con CloudTrail.

- Elija Specific operation(s) (Operaciones específicas) y, a continuación, introduzca los comandos de Session Manager (uno a uno) para recibir notificaciones. También puede elegir StartSession, ResumeSession y TerminateSession. (EventBridge no admite los comandos `Get*`, `List*` ni `Describe*`).
- Para Seleccione un destino, elija Tema de SNS. En Topic (Tema), seleccione el nombre del tema de Amazon SNS que creó en el paso 1.

Para obtener más información, consulte la Guía del usuario de Amazon EventBridge y la Guía de introducción a Amazon Simple Notification Service.

Habilitar y deshabilitar el registro de la sesión

El registro de sesiones registra la información sobre las sesiones actuales y finalizadas en la consola de Systems Manager. También se pueden registrar los detalles sobre los comandos que se ejecutan durante las sesiones en su Cuenta de AWS. Registrarse en la sesión le permite hacer lo siguiente:

- Crear y almacenar los registros de sesión para fines de archivado.
- Generar un informe que muestre los detalles de cada conexión realizada en los nodos administrados mediante Session Manager en los últimos 30 días.
- Genere notificaciones del registro de la sesión en su Cuenta de AWS, como las notificaciones de Amazon Simple Notification Service (Amazon SNS).
- Iniciar otra acción automáticamente en un recurso de AWS como resultado de las acciones realizadas durante una sesión, como la ejecución de una función de AWS Lambda, el inicio de una canalización de AWS CodePipeline o la ejecución de un documento de AWS Systems Manager Run Command.

> **Important**
>
> Tenga en cuenta los siguientes requisitos y limitaciones de Session Manager:
>
> - Session Manager registra los comandos que usted ingresa y sus resultados durante una sesión, en función de sus preferencias de sesión. Para evitar que la información confidencial, como las contraseñas, se vean en los registros de sesión, recomendamos utilizar los siguientes comandos al ingresar información confidencial durante una sesión.

Linux & macOS

```
stty -echo; read passwd; stty echo;
```

Windows

```
$Passwd = Read-Host -AsSecureString
```

- Si utiliza Windows Server 2012 o versiones anteriores, los datos de sus registros podrían no tener el formato óptimo. Recomendamos que utilice Windows Server 2012 R2 y posterior para contar con formatos de registro óptimo.
- Si utiliza nodos administrados de Linux o macOS, asegúrese de que la utilidad de pantalla esté instalada. Si no lo está, los datos de registro podrían truncarse. En Amazon Linux 1, Amazon Linux 2, AL2023 y Ubuntu Server, la utilidad de pantalla se instala de forma predeterminada. Para instalar la pantalla de forma manual, en función de la versión de Linux que utilice, ejecute `sudo yum install screen` o `sudo apt-get install screen`.
- El registro no está disponible para las sesiones de Session Manager que se conectan a través del reenvío de puertos o de SSH. Esto se debe a que SSH cifra todos los datos de la sesión y Session Manager solo sirve como túnel para las conexiones de SSH.

Para obtener más información acerca de los permisos necesarios para utilizar Amazon S3 o los Registros de Amazon CloudWatch para registrar los datos de la sesión, consulte Creación de un rol de IAM con permisos para Session Manager, Amazon S3 y los Registros de CloudWatch (consola).

Consulte los siguientes temas para obtener más información acerca de las opciones de registro de Session Manager.

Temas

- Streaming de los datos de la sesión con los Registros de Amazon CloudWatch (consola)
- Registro de los datos de la sesión con Amazon S3 (consola)
- Registro de los datos de la sesión con los Registros de Amazon CloudWatch (consola)
- Configuración del registro de sesiones en disco
- Ajuste del tiempo de almacenamiento del archivo de registro temporal de Session Manager en el disco

- Inhabilitación del registro de Session Manager en Registros de CloudWatch y Amazon S3

Streaming de los datos de la sesión con los Registros de Amazon CloudWatch (consola)

Puede enviar un flujo continuo de registros de datos de sesión a los Registros de Amazon CloudWatch. Los detalles esenciales, como los comandos que un usuario ha ejecutado en una sesión, el ID del usuario que ejecutó los comandos y las marcas de tiempo para el momento en que los datos de sesión se transmiten a los Registros de CloudWatch, se incluyen cuando se efectúa el streaming de los datos de la sesión. Con el streaming de los datos de la sesión, los registros tienen formato JSON para ayudarlo a integrarse a sus soluciones de registro existentes. El streaming de los datos de la sesión no es compatible con los comandos interactivos.

> **Note**
>
> Para transmitir los datos de la sesión desde los nodos administrados de Windows Server, debe tener instalado PowerShell 5.1 o una versión posterior. De forma predeterminada, Windows Server 2016 y las versiones posteriores tienen instalada la versión requerida de PowerShell. Sin embargo, Windows Server 2012 y 2012 R2 no tienen instalada de forma predeterminada la versión requerida de PowerShell. Si aún no ha actualizado PowerShell en los nodos administrados de Windows Server 2012 o 2012 R2, puede hacerlo mediante Run Command. Para obtener información sobre cómo actualizar PowerShell con Run Command, consulte Actualización de PowerShell con Run Command.

> **Important**
>
> Si tiene la configuración de política PowerShell Transcription (Transcripción de PowerShell) establecida en los nodos administrados de Windows Server, no podrá transmitir los datos de la sesión.

Para transmitir los datos de la sesión con los Registros de Amazon CloudWatch (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Session Manager.
3. Elija la pestaña Preferences (Preferencias) y, a continuación, Edit (Editar).

4. En CloudWatch logging (Registro de CloudWatch), seleccione la casilla de verificación ubicada junto a Enable (Habilitar).
5. Elija la opción de Stream session logs (Transmitir registros de sesiones).
6. (Recomendado) Seleccione la casilla de verificación situada junto a Allow only encrypted CloudWatch log groups (Permitir solo los grupos de registros cifrados de CloudWatch). Con esta opción activada, los datos de registro se cifran con la clave de cifrado del lado del servidor especificado para el grupo de registros. Si no desea cifrar los datos de registro que se envían a los Registros de CloudWatch, desactive la casilla de verificación. También debe desactivar la casilla de verificación si no se permite el cifrado en el grupo de registros.
7. En CloudWatch logs (Registros de CloudWatch), para especificar el grupo de registros de los Registros de CloudWatch existente en su Cuenta de AWS en el que se cargarán los registros de la sesión, seleccione una de las siguientes opciones:

   - Ingrese el nombre de un grupo de registros en el cuadro de texto que ya se haya creado en su cuenta para almacenar los datos de registro de las sesiones.
   - Browse log groups (Buscar grupos de registros): seleccione un grupo de registros que ya se haya creado en su cuenta para almacenar los datos de registro de las sesiones.
8. Seleccione Save.

Registro de los datos de la sesión con Amazon S3 (consola)

Puede elegir almacenar los datos de registro de las sesiones en un bucket de Amazon Simple Storage Service (Amazon S3) especificado con fines de depuración y solución de problemas. La opción predeterminada es para los registros que se van a enviar a un bucket cifrado de Amazon S3. El cifrado se realiza con la clave especificada para el bucket, ya sea una AWS KMS key o una clave de cifrado del lado del servidor (SSE) de Amazon S3 (AES-256).

> **Important**
>
> Si utiliza buckets de estilo de alojamiento virtual con Capa de conexión segura (SSL), el certificado comodín de SSL solo se asociará con los buckets que no contengan puntos. Para solucionar esto, use HTTP o escriba su propia lógica de verificación de certificado. Cuando use buckets de tipo de alojamiento virtual, le recomendamos no utilizar puntos (“.”) en los nombres de los buckets.

Cifrado de buckets de Amazon S3

Para poder enviar los registros a su bucket de Amazon S3 con cifrado, el cifrado debe estar permitido en el bucket. Para obtener más información acerca del cifrado de buckets de Amazon S3, consulte Cifrado predeterminado de Amazon S3 para los buckets de S3.

Clave administrada por el cliente

Si utiliza una clave de KMS que administra usted mismo para cifrar el bucket, el perfil de instancias de IAM adjunto a sus instancias debe tener permisos explícitos para leer la clave. Si utiliza una Clave administrada de AWS, la instancia no requiere este permiso explícito. Para obtener más información acerca de cómo proporcionar al perfil de instancias acceso para utilizar la clave, consulte Permiso del uso de la clave a los usuarios en la Guía para desarrolladores de AWS Key Management Service.

Siga estos pasos para configurar Session Manager para que almacene los registros de las sesiones en un bucket de Amazon S3.

> **Note**
>
> También puede utilizar la AWS CLI para especificar o cambiar el bucket de Amazon S3 al que se enviarán los datos de la sesión. Para obtener más información, consulta Actualizar preferencias de Session Manager (línea de comandos).

Para registrar los datos de la sesión con Amazon S3 (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Session Manager.
3. Elija la pestaña Preferences (Preferencias) y, a continuación, Edit (Editar).
4. Marque la casilla de verificación situada junto a Enable (Habilitar) en S3 logging (Registro de S3).
5. (Recomendado) Seleccione la casilla de verificación situada junto a Allow only encrypted S3 buckets (Permitir solo buckets cifrados de S3). Con esta opción activada, los datos de registro se cifran con la clave de cifrado del lado del servidor especificada para el bucket. Si no desea cifrar los datos de registro que se envían a Amazon S3, desactive la casilla de verificación. También debe desactivar la casilla de verificación si no se permite el cifrado en el bucket de S3.
6. En S3 bucket name (Nombre del bucket de S3), realice alguna de las siguientes operaciones:

> **Note**
>
> Cuando use buckets de tipo de alojamiento virtual, le recomendamos no utilizar puntos (“.”) en los nombres de los buckets. Para obtener más información acerca de las convenciones de nomenclatura de buckets de Amazon S3, consulte Restricciones y limitaciones de los buckets en la Guía del usuario de Amazon Simple Storage Service.

- Choose a bucket name from the list (Elegir un nombre de bucket de la lista): seleccione un bucket de Amazon S3 que ya se haya creado en su cuenta para almacenar los datos de registro de las sesiones.
- Enter a bucket name in the text box (Ingresar el nombre de un bucket en el cuadro de texto): escriba el nombre de un bucket de Amazon S3 que ya se haya creado en su cuenta para almacenar los datos de registro de las sesiones.

7. (Opcional) En S3 key prefix (Prefijo de clave de S3), escriba el nombre de una carpeta nueva o existente para almacenar registros en el bucket seleccionado.
8. Seleccione Save.

Para obtener más información acerca de cómo se trabaja con Amazon S3 y buckets de Amazon S3, consulte la Guía del usuario de Amazon Simple Storage Service y la Guía del usuario de Amazon Simple Storage Service.

Registro de los datos de la sesión con los Registros de Amazon CloudWatch (consola)

Con Registros de Amazon CloudWatch, puede acceder a los archivos de registros de diversos Servicios de AWS, supervisarlos y almacenarlos. Puede enviar los datos de registro de las sesiones a un grupo de registros de los Registros de CloudWatch con fines de depuración y solución de problemas. La opción predeterminada es enviar los datos de registro con cifrado mediante su clave de KMS, pero puede enviar los datos a su grupo de registros con o sin cifrado.

Siga estos pasos para configurar AWS Systems Manager Session Manager para que envíe los datos de registro de las sesiones a un grupo de registros de los Registros de CloudWatch al final de sus sesiones.

> **Note**
>
> También puede utilizar la AWS CLI para especificar o cambiar el grupo de registros de los Registros de CloudWatch al que se enviarán los datos de la sesión. Para obtener más información, consulta Actualizar preferencias de Session Manager (línea de comandos).

Para registrar los datos de la sesión con los Registros de Amazon CloudWatch (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Session Manager.
3. Elija la pestaña Preferences (Preferencias) y, a continuación, Edit (Editar).
4. En CloudWatch logging (Registro de CloudWatch), seleccione la casilla de verificación ubicada junto a Enable (Habilitar).
5. Elija la opción Upload session logs (Cargar registros de sesiones).
6. (Recomendado) Seleccione la casilla de verificación situada junto a Allow only encrypted CloudWatch log groups (Permitir solo los grupos de registros cifrados de CloudWatch). Con esta opción activada, los datos de registro se cifran con la clave de cifrado del lado del servidor especificado para el grupo de registros. Si no desea cifrar los datos de registro que se envían a los Registros de CloudWatch, desactive la casilla de verificación. También debe desactivar la casilla de verificación si no se permite el cifrado en el grupo de registros.
7. En CloudWatch logs (Registros de CloudWatch), para especificar el grupo de registros de CloudWatch Logs existente en su Cuenta de AWS en el que se cargarán los registros de la sesión, seleccione una de las siguientes opciones:
   - Choose a log group from the list (Elegir un grupo de recursos de la lista): seleccione un grupo de registros que ya se ha creado en su cuenta para almacenar datos de registros de sesiones.
   - Enter a log group name in the text box (Escribir un nombre del grupo de registros en el cuadro de texto): escriba el nombre de un grupo de registros que ya se haya creado en su cuenta para almacenar los datos de registro de la sesión.
8. Seleccione Save.

Para obtener más información acerca del uso de los Registros de CloudWatch, consulte la Guía del usuario de los Registros de Amazon CloudWatch.

Configuración del registro de sesiones en disco

Tras habilitar el registro de Session Manager en CloudWatch o Amazon S3, todos los comandos ejecutados durante una sesión (y los resultado de estos) se registran en un archivo temporal del disco de la instancia de destino. El archivo temporal se llama `ipcTempFile.log`.

`ipcTempFile.log` se controla mediante el parámetro `SessionLogsDestination` del archivo de configuración de SSM Agent. Este parámetro acepta los siguientes valores:

- disk: si especifica este parámetro y el registro de sesiones en CloudWatch o Amazon S3 está habilitado, SSM Agent crea el archivo de registro temporal `ipcTempFile.log` y registra los comandos de sesión y la salida en el disco. Session Manager carga este registro en CloudWatch o S3 durante o después de la sesión, según la configuración del registro. A continuación, el registro se elimina según la duración especificada para el parámetro de configuración `SessionLogsRetentionDurationHours` de SSM Agent.

  Si especifica este parámetro y el registro de sesiones en CloudWatch y Amazon S3 está deshabilitado, SSM Agent seguirá registrando el historial de comandos y los resultados en el archivo `ipcTempFile.log`. El archivo se eliminará según la duración especificada para el parámetro de configuración `SessionLogsRetentionDurationHours` de SSM Agent.
- none: si especifica este parámetro y el registro de sesiones en CloudWatch o Amazon S3 está habilitado, el registro en el disco funciona exactamente igual que si hubiera especificado el parámetro `disk`. SSM Agent necesita el archivo temporal cuando se habilita el registro de sesiones en CloudWatch o Amazon S3.

  Si especifica este parámetro y el registro de sesiones en CloudWatch y Amazon S3 está deshabilitado, SSM Agent no creará el archivo `ipcTempFile.log`.

Utilice el siguiente procedimiento para activar o desactivar la creación del archivo de registro temporal `ipcTempFile.log` en el disco cuando se inicie una sesión.

Activación o desactivación de la creación del archivo de registro temporal Session Manager en el disco

1. Instale SSM Agent en su instancia o actualícelo a la versión 3.2.2086 o superior. Para obtener información sobre cómo comprobar el número de versión del agente, consulte Verificación del número de versión de SSM Agent. Para obtener información sobre cómo instalar el agente

de manera manual, busque el procedimiento correspondiente a su sistema operativo en las siguientes secciones:

- Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux
- Instalación y desinstalación manual de SSM Agent en instancias de EC2 para macOS
- Cómo instalar y desinstalar de forma manual SSM Agent en instancias de EC2 para Windows Server

2. Conéctese a su instancia y busque el archivo `amazon-ssm-agent.json` en la siguiente ubicación.

   - Linux: /etc/amazon/ssm/
   - macOS: /opt/aws/ssm/
   - Windows Server: C:\Program Files\Amazon\SSM

   Si el archivo `amazon-ssm-agent.json` no existe, copie el contenido de `amazon-ssm-agent.json.template` en un archivo nuevo del mismo directorio. Asigne al nuevo archivo el nombre `amazon-ssm-agent.json`.

3. En el parámetro `SessionLogsDestination`, especifique `none` o `disk`. Guarde los cambios.
4. Reinicie SSM Agent.

Si especificó `disk` en el parámetro `SessionLogsDestination`, puede comprobar que SSM Agent crea el archivo de registro temporal al iniciar una nueva sesión y, a continuación, localizando `ipcTempFile.log` en la siguiente ubicación:

- Linux: /var/lib/amazon/ssm/*target ID*/session/orchestration/*session ID*/Standard_Stream/ipcTempFile.log
- macOS: /opt/aws/ssm/data/*target ID*/session/orchestration/*session ID*/Standard_Stream/ipcTempFile.log
- Windows Server: C:\ProgramData\Amazon\SSM\InstanceData*target ID*\session\orchestration*session ID*\Standard_Stream\ipcTempFile.log

> **Note**
>
> De forma predeterminada, el archivo de registro temporal se guarda en la instancia durante 14 días.

Si desea actualizar el parámetro `SessionLogsDestination` en varias instancias, le recomendamos que cree un documento de SSM que especifique la nueva configuración. A continuación, puede usar Run Command de Systems Manager para implementar el cambio en sus instancias. Para obtener más información, consulte Writing your own AWS Systems Manager documents (blog) y Ejecución de comandos en nodos administrados.

Ajuste del tiempo de almacenamiento del archivo de registro temporal de Session Manager en el disco

Tras habilitar el registro de Session Manager en CloudWatch o Amazon S3, todos los comandos ejecutados durante una sesión (y los resultado de estos) se registran en un archivo temporal del disco de la instancia de destino. El archivo temporal se llama `ipcTempFile.log`. Durante una sesión, o una vez finalizada, Session Manager carga este registro temporal en CloudWatch o S3. A continuación, el registro temporal se elimina según la duración especificada para el parámetro de configuración `SessionLogsRetentionDurationHours` de SSM Agent. De forma predeterminada, el archivo de registro temporal se guarda en la instancia durante 14 días en la siguiente ubicación:

- Linux: /var/lib/amazon/ssm/*target ID*/session/orchestration/*session ID*/Standard_Stream/ipcTempFile.log
- macOS: /opt/aws/ssm/data/*target ID*/session/orchestration/*session ID*/Standard_Stream/ipcTempFile.log
- Windows Server: C:\ProgramData\Amazon\SSM\InstanceData*target ID*\session\orchestration*session ID*\Standard_Stream\ipcTempFile.log

Utilice el siguiente procedimiento para ajustar el tiempo de almacenamiento del archivo de registro de Session Manager en el disco.

Ajuste del tiempo de almacenamiento del archivo **`ipcTempFile.log`** en el disco

1. Conéctese a su instancia y busque el archivo `amazon-ssm-agent.json` en la siguiente ubicación.

- Linux: /etc/amazon/ssm/
- macOS: /opt/aws/ssm/
- Windows Server: C:\Program Files\Amazon\SSM

Si el archivo `amazon-ssm-agent.json` no existe, copie el contenido de `amazon-ssm-agent.json.template` en un archivo nuevo del mismo directorio. Asigne al nuevo archivo el nombre `amazon-ssm-agent.json`.

2. Cambie el valor de `SessionLogsRetentionDurationHours` al número de horas deseado. Si `SessionLogsRetentionDurationHours` se establece en 0, el archivo de registro temporal se creará durante la sesión y se eliminará al finalizarla. Esta configuración debe garantizar que el archivo de registro no persista una vez finalizada la sesión.
3. Guarde los cambios.
4. Reinicie SSM Agent.

Inhabilitación del registro de Session Manager en Registros de CloudWatch y Amazon S3

Puede utilizar la consola de Systems Manager o la AWS CLI para deshabilitar el registro de la sesión en la cuenta.

Para deshabilitar el registro de la sesión (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Session Manager.
3. Elija la pestaña Preferences (Preferencias) y, a continuación, Edit (Editar).
4. Para deshabilitar el registro de CloudWatch, en la sección Registro de CloudWatch, desactive la casilla Habilitar.
5. Para deshabilitar el registro de S3, en la sección Registro de S3, desactive la casilla Habilitar.
6. Seleccione Save.

Para deshabilitar el registro de sesiones (AWS CLI)

Para deshabilitar el registro de la sesión mediante la AWS CLI, siga las instrucciones en Actualizar preferencias de Session Manager (línea de comandos).

En el archivo JSON, asegúrese de que las entradas de `s3BucketName` y `cloudWatchLogGroupName` no contengan valores. Por ejemplo:

```
"inputs": {
        "s3BucketName": "",
        ...
        "cloudWatchLogGroupName": "",
        ...
    }
```

Como alternativa, puede eliminar todas las entradas de `S3*` y `cloudWatch*` del archivo JSON para deshabilitar el registro.

> **Note**
>
> Según la configuración, después de deshabilitar CloudWatch o S3, es posible que SSM Agent siga generando un archivo de registro temporal en el disco. Para obtener información sobre cómo deshabilitar el registro, consulte Configuración del registro de sesiones en disco.

Esquema del documento de Session

La siguiente información describe los elementos de esquema de un documento de Session. AWS Systems Manager Session Manager utiliza los documentos de Session para determinar qué tipo de sesión iniciar, como una sesión estándar, una sesión de reenvío de puertos o una sesión para ejecutar un comando interactivo.

schemaVersion

Versión del esquema del documento de Session. Los documentos de Session solo admiten la versión 1.0.

Tipo: cadena

Obligatorio: sí

description

Una descripción que especifique para el documento de Session. Por ejemplo, “Documento para iniciar la sesión de reenvío de puertos con Session Manager”.

Tipo: cadena

Requerido: no

sessionType

El tipo de sesión que se utiliza para establecer el documento de Session.

Tipo: cadena

Obligatorio: sí

Valores válidos: `InteractiveCommands` | `NonInteractiveCommands` | `Port` | `Standard_Stream`

inputs

Las preferencias de sesión que se van a utilizar para las sesiones establecidas mediante este documento de Session. Este elemento es necesario para los documentos de Session que se utilizan para crear sesiones `Standard_Stream`.

Tipo: StringMap

Obligatorio: no

s3BucketName

El bucket de Amazon Simple Storage Service (Amazon S3) al que desea enviar los registros de las sesiones al finalizarlas.

Tipo: cadena

Requerido: no

s3KeyPrefix

El prefijo que se debe utilizar al enviar registros al bucket de Amazon S3 que usted especificó en el elemento de entrada `s3BucketName` Para obtener más información acerca del uso de un prefijo compartido con almacenamiento de objetos en Amazon S3, consulte How do I use folders in an S3 bucket? en la Guía del usuario de Amazon Simple Storage Service.

Tipo: cadena

Requerido: no

s3EncryptionEnabled

Si se establece en `true`, el bucket de Amazon S3 especificado en el elemento de entrada `s3BucketName` debe estar cifrado.

Tipo: Booleano

Obligatorio: sí

cloudWatchLogGroupName

El nombre del grupo de los Registros de Amazon CloudWatch (Registros de CloudWatch) al que desea enviar los registros de las sesiones al finalizarlas.

Tipo: cadena

Requerido: no

cloudWatchEncryptionEnabled

Si se establece en `true`, el grupo de registros que especificó en el elemento de entrada `cloudWatchLogGroupName` debe estar cifrado.

Tipo: Booleano

Obligatorio: sí

cloudWatchStreamingEnabled

Si se establece en `true`, se envía un flujo continuo de registros de datos de sesiones al grupo de registros que especificó en el elemento de entrada `cloudWatchLogGroupName`. Si se establece en `false`, los registros de las sesiones se envían al grupo de registros que especificó en el elemento de entrada `cloudWatchLogGroupName` al finalizar las sesiones.

Tipo: Booleano

Obligatorio: sí

kmsKeyId

El ID de la AWS KMS key que desee utilizar para cifrar aún más los datos entre los equipos cliente locales y los nodos administrados de Amazon Elastic Compute Cloud (Amazon EC2) a los que se conecte.

Tipo: cadena

Requerido: no

runAsEnabled

Si se establece en `true`, debe especificar una cuenta de usuario que exista en los nodos administrados a los que se conectará en el elemento de entrada `runAsDefaultUser`. De lo contrario, las sesiones no se iniciarán. De forma predeterminada, las sesiones se inician utilizando la cuenta `ssm-user` creada por AWS Systems Manager SSM Agent. La característica Ejecutar como solo se admite para conectarse a nodos administrados de Linux y macOS.

Tipo: Booleano

Obligatorio: sí

runAsDefaultUser

El nombre de la cuenta de usuario con la que se iniciarán las sesiones en los nodos administrados de Linux y macOS cuando el elemento de entrada `runAsEnabled` se establezca en `true`. La cuenta de usuario que especifique para este elemento de entrada debe existir en los nodos administrados a los que se conectará; de lo contrario, las sesiones no podrán iniciarse.

Tipo: cadena

Requerido: no

idleSessionTimeout

La cantidad de tiempo de inactividad que desea permitir antes de que una sesión se termine. Esta entrada se mide en minutos.

Tipo: cadena

Valores válidos: de 1 a 60

Obligatorio: no

maxSessionDuration

La cantidad máxima de tiempo que desea permitir antes de que una sesión se termine. Esta entrada se mide en minutos.

Tipo: cadena

Valores válidos: 1-1440

Obligatorio: no

shellProfile

Las preferencias que especificó por sistema operativo para que se apliquen dentro de las sesiones, como las preferencias de shell, las variables de entorno, los directorios de trabajo y la ejecución de varios comandos cuando se inicia una sesión.

Tipo: StringMap

Obligatorio: no

windows

Las preferencias de shell, las variables de entorno, los directorios de trabajo y los comandos que especifique para las sesiones en los nodos administrados de Windows Server.

Tipo: cadena

Requerido: no

linux

Las preferencias del intérprete de comandos, las variables de entorno, los directorios de trabajo y los comandos que especifique para las sesiones en los nodos administrados de Linux y macOS.

Tipo: cadena

Requerido: no

parameters

Un objeto que define los parámetros que acepta el documento. Para obtener más información acerca de cómo definir los parámetros de los documentos, consulte parámetros en Elementos de datos de nivel superior. En el caso de los parámetros a los que suele hacer referencia, le recomendamos almacenarlos en Systems Manager Parameter Store y, luego, hacer referencia a ellos. Puede hacer referencia a los parámetros de Parameter Store `String` y `StringList` en esta sección de un documento. No puede hacer referencia a los parámetros de Parameter Store `SecureString` en esta sección de un documento. Puede hacer referencia a un parámetro de Parameter Store mediante el siguiente formato.

```
{{ssm:parameter-name}}
```

Para obtener más información acerca de Parameter Store, consulte AWS Systems Manager Parameter Store.

Tipo: StringMap

Obligatorio: no

properties

Un objeto cuyos valores especificados se utilizan en la operación `StartSession` de la API.

Para los documentos de Session que se utilizan para las sesiones `InteractiveCommands`, el objeto de propiedades incluye los comandos que se ejecutarán en los sistemas operativos que usted especifique. Además, puede determinar si los comandos se ejecutan como `root` mediante la propiedad booleana `runAsElevated`. Para obtener más información, consulte Restricción del acceso a los comandos de una sesión.

Para los documentos de Session que se utilizan para las sesiones `Port`, el objeto de propiedades contiene el número de puerto al que se debe redirigir el tráfico. Para ver un ejemplo, consulte más adelante en este tema el ejemplo de documento de Session de tipo `Port`.

Tipo: StringMap

Obligatorio: no

Ejemplo de documento de Session de tipo `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"
        }
    }
}
```

Ejemplo de documento de Session de tipo `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
        }
    }
}
```

Ejemplo de documento de Session de tipo `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"
    }
}
```

Ejemplo de documento de Session con caracteres especiales

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
      }
   }
}
```

Solución de problemas de Session Manager

Utilice la siguiente información como ayuda para solucionar problemas con AWS Systems Manager Session Manager.

Temas

- AccessDeniedException cuando se llama a la operación TerminateSession
- Se produjo un error inesperado en el proceso del documento: se agotó el tiempo de espera del operador documental.
- Session Manager no se puede conectar desde la consola de Amazon EC2
- Sin permiso para iniciar una sesión
- SSM Agent no está en línea
- Sin permiso para cambiar preferencias de sesiones
- Nodo administrado no disponible o no configurado para Session Manager

- Complemento de Session Manager no encontrado
- Complemento de Session Manager no agregado de manera automática a la ruta de la línea de comandos (Windows)
- El complemento Session Manager no responde
- TargetNotConnected
- Aparece una pantalla en blanco después de iniciar una sesión
- El nodo administrado deja de responder durante las sesiones de larga ejecución
- Se ha producido un error (InvalidDocument) al llamar a la operación StartSession

AccessDeniedException cuando se llama a la operación TerminateSession

Problema: cuando se intenta terminar una sesión, Systems Manager devuelve el siguiente error:

```
An error occurred (AccessDeniedException) when calling the TerminateSession operation:
User: <user_arn> is not authorized to perform: ssm:TerminateSession on resource:
<ssm_session_arn> because no identity-based policy allows the ssm:TerminateSession
 action.
```

Solución A: confirme que la última versión del complemento de Session Manager esté instalada en el nodo.

Introduzca el siguiente comando en el terminal y pulse Intro.

```
session-manager-plugin --version
```

Solución B: instale o vuelva a instalar la última versión del complemento.

Para obtener más información, consulte Instalación del complemento de Session Manager para la AWS CLI.

Solución C: intente restablecer la conexión con el nodo.

Compruebe que el nodo responda a las solicitudes. Intente restablecer la sesión. O, si es necesario, abra la consola de Amazon EC2 y compruebe el estado de la instancia en ejecución.

Se produjo un error inesperado en el proceso del documento: se agotó el tiempo de espera del operador documental.

Problema: al iniciar una sesión en un host Linux, Systems Manager devuelve el siguiente error:

```
document process failed unexpectedly: document worker timed out,
check [ssm-document-worker]/[ssm-session-worker] log for crash reason
```

Si configuró el registro SSM Agent, como se describe en Visualización de registros de SSM Agent, puede ver más detalles en el registro de depuración. En este caso, Session Manager muestra la siguiente entrada de registro:

```
failed to create channel: too many open files
```

Este error generalmente indica que hay demasiados procesos de trabajo de Session Manager en ejecución y que el sistema operativo subyacente ha alcanzado un límite. Tiene dos opciones para resolver este problema.

Solución A: aumentar el límite de notificación de archivos del sistema operativo

Para aumentar el límite, ejecute el siguiente comando desde un host Linux independiente. Este comando utiliza Run Command de Systems Manager. El valor especificado aumenta `max_user_instances` a 8192. Este valor es considerablemente superior al valor predeterminado de 128, pero no sobrecargará los recursos del host:

```
aws ssm send-command --document-name AWS-RunShellScript \
--instance-id i-02573cafcfEXAMPLE  --parameters \
"commands=sudo sysctl fs.inotify.max_user_instances=8192"
```

Solución B: reducir las notificaciones de archivos utilizadas por Session Manager en el host de destino

Ejecute el siguiente comando desde un host Linux independiente para enumerar las sesiones que se ejecutan en el host de destino:

```
aws ssm describe-sessions --state Active --filters key=Target,value=i-02573cafcfEXAMPLE
```

Revise la salida del comando para identificar las sesiones que ya no se necesitan. Puede terminar esas sesiones mediante la ejecución del siguiente comando desde un host Linux independiente:

```
aws ssm terminate-session –session-id session ID
```

Opcionalmente, una vez que no haya más sesiones en ejecución en el servidor remoto, podrá liberar recursos adicionales mediante la ejecución del siguiente comando desde un host Linux

independiente. Este comando termina todos los procesos de Session Manager que se ejecutan en el host remoto y, en consecuencia, todas las sesiones con el host remoto. Antes de ejecutar este comando, verifique que no haya sesiones en curso que desee conservar:

```
aws ssm send-command --document-name AWS-RunShellScript \
            --instance-id i-02573cafcfEXAMPLE --parameters \
'{"commands":["sudo kill $(ps aux | grep ssm-session-worker | grep -v grep | awk
 '"'"'{print $2}'"'"')"]}'
```

Session Manager no se puede conectar desde la consola de Amazon EC2

Problema: luego de crear una instancia nueva, en el botón Conectar > en la pestaña Administrador de sesiones de la consola de Amazon Elastic Compute Cloud (Amazon EC2) no aparece la opción para conectarse.

Solución A: creación de un perfil de instancia. Si aún no lo ha hecho (tal y como se indica en la información de la pestaña Administrador de sesiones de la consola de EC2), cree un perfil de instancia de AWS Identity and Access Management (IAM) mediante Quick Setup. Quick Setup es una herramienta de AWS Systems Manager.

Session Manager requiere un perfil de instancia de IAM para conectarse a la instancia. Puede crear un perfil de instancia y asignarlo a la instancia mediante la creación de una configuración de administración de host conQuick Setup. Una configuración de administración de host crea un perfil de instancia con los permisos necesarios y lo asigna a la instancia. Una configuración de administración de host también habilita otras herramientas de Systems Manager y crea roles de IAM para ejecutar esas herramientas. El uso de Quick Setup o de las herramientas habilitadas por la configuración de administración de host es gratuito. Abra Quick Setup y cree una configuración de administración de host.

> **Important**
>
> Luego de crear la configuración de administración de host, Amazon EC2 puede tardar varios minutos en registrar el cambio y actualizar la pestaña Administrador de sesiones. Si la pestaña no muestra el botón Conectar después de dos o tres minutos, actualice su instancia. Si sigue sin ver la opción para conectarse, abra Configuración Rápida y verifique que solo tenga una configuración de administración de hosts. Si hay dos, elimine la configuración anterior y espere unos minutos.

Si sigue sin poder conectarse después de crear una configuración de administración de host o si se produce un error, incluido un error relacionado con SSM Agent, consulte una de las siguientes soluciones:

- Solución B: no hay un error, pero aún no se puede conectar
- Solución C: error por la falta del SSM Agent

Solución B: no hay un error, pero aún no se puede conectar

Si creó la configuración de administración de host, esperó varios minutos antes de intentar conectarse y sigue sin poder hacerlo, es posible que tenga que aplicar de manera manual la configuración de administración de host a la instancia. Utilice el siguiente procedimiento para actualizar la configuración de administración de host con Quick Setup y aplicar los cambios a una instancia.

Actualización de una configuración de administración de host mediante Quick Setup

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Quick Setup.
3. En la lista Configuraciones, elija la configuración Administración de host que creó.
4. Elija Accionesy, luego, seleccione Editar configuración.
5. Cerca de la parte inferior de la sección Objetivos, en Elija cómo desea segmentar las instancias, seleccione Manual.
6. En la sección Instancias, elija la instancia que creó.
7. Elija Actualizar.

Espere unos minutos a que EC2 actualice la pestaña Administrador de sesiones. Si sigue sin poder conectarse o si se produce un error, revise las demás soluciones para este problema.

Solución C: error por la falta del SSM Agent

Si no pudo crear una configuración de administración de host con Quick Setup o si se produjo un error que indicaba que SSM Agent no estaba instalado, es posible que deba instalar SSM Agent de manera manual en la instancia. SSM Agent es un software de Amazon que permite a Systems Manager conectarse a la instancia medianteSession Manager. SSM Agent está instalado de forma predeterminada en la mayoría de las imágenes de máquina de Amazon (AMI). Si la instancia

se creó a partir de una AMI no estándar o una AMI anterior, es posible que deba instalar el agente de manera manual. Para conocer el procedimiento de instalación de SSM Agent, consulte el siguiente tema correspondiente al sistema operativo de la instancia.

- Windows Server
- macOS
- AlmaLinux
- Amazon Linux 1
- Amazon Linux 2 y AL2023
- CentOS
- CentOS Stream
- Debian Server
- Oracle Linux
- Red Hat Enterprise Linux
- Rocky Linux
- SUSE Linux Enterprise Server
- Ubuntu Server

Si se presentan problemas con SSM Agent, consulte Solución de problemas de SSM Agent.

Sin permiso para iniciar una sesión

Problema: intenta iniciar una sesión, pero el sistema le indica que no tiene los permisos necesarios.

- Solución: un administrador de sistema no le ha concedido los permisos de política de AWS Identity and Access Management (IAM) para iniciar sesiones de Session Manager. Para obtener información, consulte Control del acceso de las sesiones de usuario a las instancias.

SSM Agent no está en línea

Problema: ve un mensaje en la pestaña Session Manager de la instancia de Amazon EC2 que indica: “SSM Agent no está en línea. El SSM Agent no pudo conectarse a un punto de conexión de Systems Manager para registrarse en el servicio”.

Solución: SSM Agent es un software de Amazon que se ejecuta en instancias de Amazon EC2 para que Session Manager pueda conectarse a ellas. Si ve este error, significa SSM Agent no puede

establecer una conexión con el punto de conexión de Systems Manager. Los posibles orígenes del problema podrían estar en las restricciones del firewall, en los problemas de enrutamiento o la falta de conectividad a Internet. Para resolver esta situación, investigue los problemas de conectividad de la red. Para obtener más información, consulte Solución de problemas de SSM Agent y Solución de problemas de disponibilidad de nodos administrados. Para obtener información sobre los puntos de conexión de Systems Manager, consulte Cuotas y puntos de conexión de AWS Systems Manager en la Referencia general de AWS.

Sin permiso para cambiar preferencias de sesiones

Problema: intenta actualizar las preferencias de sesión globales para su organización, pero el sistema le indica que no tiene los permisos necesarios para hacerlo.

- Solución: un administrador de sistema no le ha concedido los permisos de política de IAM para configurar las preferencias de Session Manager. Para obtener más información, consulta Concesión o denegación de permisos de usuario para actualizar preferencias de Session Manager.

Nodo administrado no disponible o no configurado para Session Manager

Problema 1: desea iniciar una sesión en la página de la consola Start a session (Iniciar una sesión), pero un nodo administrado no está en la lista.

- Solución A: El nodo administrado al que desea conectarse podría no haberse configurado para AWS Systems Manager. Para obtener más información, consulte Configuración de la consola unificada de Systems Manager para una organización.

  > **Note**
  >
  > Si AWS Systems Manager SSM Agent ya se está ejecutando en un nodo administrado cuando adjunte el perfil de instancias de IAM, es posible que tenga que reiniciar el agente antes de que la instancia aparezca en la página Start a session (Iniciar una sesión) de la consola.

- Solución B: la configuración del proxy que aplicó a SSM Agent en el nodo administrado puede ser incorrecta. Si la configuración del proxy es incorrecta, el nodo administrado no podrá alcanzar los puntos de conexión de servicio necesarios o el nodo se podría notificar como un sistema operativo diferente a Systems Manager. Para obtener más información, consulte Configuración de SSM

Agent para utilizar un proxy en nodos de Linux y Configurar el SSM Agent para usar un proxy para las instancias de Windows Server.

Problema 2: un nodo administrado al que desea conectarse está en la lista de la página Start a session (Iniciar una sesión) de la consola, pero la página notifica que “The instance you selected isn't configured to use Session Manager” (La instancia que ha seleccionado no está configurada para utilizar Session Manager).

- Solución A: el nodo administrado se ha configurado para usarse con el servicio de Systems Manager, pero el perfil de instancias de IAM adjunto al nodo podría no incluir los permisos para la herramienta Session Manager. Para obtener información, consulte Verificación o creación de un perfil de instancias de IAM con permisos de Session Manager.
- Solución B: el nodo administrado no está ejecutando una versión de SSM Agent que admita Session Manager. Actualice SSM Agent en el nodo a la versión 2.3.68.0 o una posterior.

  Actualice SSM Agent de forma manual en un nodo administrado siguiendo los pasos que se describen en Cómo instalar y desinstalar de forma manual SSM Agent en instancias de EC2 para Windows Server, Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux o Instalación y desinstalación manual de SSM Agent en instancias de EC2 para macOS, en función del sistema operativo.

  También puede usar el documento de Run Command `AWS-UpdateSSMAgent` para actualizar la versión del agente en una o varios nodos administrados a la vez. Para obtener más información, consulta Actualización de SSM Agent mediante Run Command.

  > **Tip**
  >
  > Para mantener siempre actualizado el agente, le recomendamos actualizar el SSM Agent a la versión más reciente según un programa automatizado que defina utilizando cualquiera de los siguientes métodos:
  >
  > - Ejecutar `AWS-UpdateSSMAgent` como parte de una asociación de State Manager. Para obtener más información, consulta Tutorial: actualización automática de SSM Agent con AWS CLI.
  > - Ejecute `AWS-UpdateSSMAgent` como parte de un periodo de mantenimiento. Para obtener información acerca de cómo trabajar con periodos de mantenimiento, consulte

Crear y administrar periodos de mantenimiento mediante la consola y Tutorial: cree y configure un periodo de mantenimiento mediante la AWS CLI.

- Solución C: el nodo administrado no puede alcanzar los puntos de conexión de servicio requeridos. Puede mejorar la posición de seguridad de los nodos administrados mediante el uso de puntos de conexión de interfaz con tecnología AWS PrivateLink para conectarse a los puntos de conexión de Systems Manager. La alternativa a usar un punto de conexión de interfaz es permitir el acceso a Internet saliente en los nodos administrados. Para obtener más información, consulte Uso de PrivateLink para configurar un punto de enlace de la VPC para Session Manager.
- Solución D: el nodo administrado dispone de recursos de memoria o CPU limitados. Aunque el nodo administrado podría de otro modo ser funcional, si el nodo no tiene suficientes recursos disponibles, no podrá establecer una sesión. Para obtener más información, consulte Solución de problemas de una instancia inaccesible.

Complemento de Session Manager no encontrado

Para utilizar la AWS CLI para ejecutar comandos de sesión, el complemento de Session Manager también debe estar instalado en su equipo local. Para obtener más información, consulta Instalación del complemento de Session Manager para la AWS CLI.

Complemento de Session Manager no agregado de manera automática a la ruta de la línea de comandos (Windows)

Cuando instala el complemento de Session Manager en Windows, el archivo ejecutable `session-manager-plugin` debería agregarse de forma automática a la variable de entorno `PATH` del sistema operativo. Si el comando no funciona después de ejecutarlo para verificar si el complemento de Session Manager se ha instalado correctamente (`aws ssm start-session --target` *`instance-id`*), es posible que deba configurarlo de forma manual mediante el procedimiento que se indica a continuación.

Para modificar la variable PATH (Windows)

1. Pulse la tecla de Windows e ingrese **`environment variables`**.
2. Elija Edit environment variables for your account (Edición de las variables de entorno de esta cuenta).
3. Elija PATH y después Editar.

4. Añada rutas al campo Variable value (Valor de variable), separadas por punto y coma, como se muestra en este ejemplo: *C:\existing\path*;***C:\new\path***

   C:\existing\path representa el valor que ya está en el campo. ***C:\new\path*** representa la ruta que desea añadir, como se muestra en estos ejemplos.

   - Máquinas de 64 bits: `C:\Program Files\Amazon\SessionManagerPlugin\bin\`
   - Máquinas de 32 bits: `C:\Program Files (x86)\Amazon\SessionManagerPlugin\bin\`
5. Elija OK (Aceptar) dos veces para aplicar la nueva configuración.
6. Cierre los símbolos del sistema en ejecución y vuelva a abrirlos.

El complemento Session Manager no responde

Durante una sesión de remisión de puertos, el tráfico podría dejar de remitirse si tiene un software antivirus instalado en el equipo local. En algunos casos, el software antivirus interfiere con el complemento de Session Manager que causa interbloqueos en el proceso. Para resolver este problema, permita o excluya el complemento de Session Manager del software antivirus. Para obtener información sobre la ruta de instalación predeterminada para el complemento de Session Manager, consulte Instalación del complemento de Session Manager para la AWS CLI.

TargetNotConnected

Problema: intenta iniciar una sesión, pero el sistema devuelve el mensaje de error “An error occurred (TargetNotConnected) when calling the StartSession operation: *InstanceID* isn't connected” (Se produjo un error [DestinoNoConectado] al llamar a la operación StartSession: el ID de la instancia no está conectado).

- Solución A: este error se devuelve cuando el nodo administrado de destino especificado para la sesión no está completamente configurado para su uso con Session Manager. Para obtener más información, consulta Configuración de Session Manager.
- Solución B: este error también se devuelve si intenta iniciar una sesión en un nodo administrado que se encuentra en otra Cuenta de AWS o Región de AWS.

Aparece una pantalla en blanco después de iniciar una sesión

Problema: ha iniciado una sesión y Session Manager muestra una pantalla en blanco.

- Solución A: este problema puede darse cuando el volumen raíz del nodo administrado está lleno. Debido a la falta de espacio en disco, el SSM Agent ya no funciona en el nodo. Para resolver este problema, utilice Amazon CloudWatch para recopilar las métricas y los registros de los sistemas operativos. Para obtener información, consulte Recopilación de métricas, registros y seguimientos con el agente de CloudWatch en la Guía del usuario de Amazon CloudWatch.
- Solución B: puede aparecer una pantalla en blanco si accedió a la consola mediante un enlace que incluye un par de punto de enlace y región que no coinciden. Por ejemplo, en la siguiente dirección URL de la consola, `us-west-2` es el punto de enlace especificado, pero `us-west-1` es la Región de AWS especificada.

```
https://us-west-2.console.aws.amazon.com/systems-manager/session-manager/sessions?
region=us-west-1
```

- Solución C: el nodo administrado se conecta a Systems Manager mediante los puntos de conexión de la VPC, y las preferencias de Session Manager registran la salida de la sesión en un bucket de Amazon S3 o en grupo de Registros de Amazon CloudWatch, pero no existe ningún punto de conexión de la puerta de enlace de `s3` ni un punto de conexión de la interfaz de `logs` en la VPC. Se necesita un punto de conexión de `s3` con el formato **`com.amazonaws.`*`region`*`.s3`** si los nodos administrados se conectan a Systems Manager mediante puntos de conexión de la VPC y sus preferencias de Session Manager escriben la salida de la sesión en un bucket de Amazon S3. De manera alternativa, se requiere un punto de conexión de `logs` con el formato **`com.amazonaws.`*`region`*`.logs`** si los nodos administrados se conectan a Systems Manager mediante los puntos de conexión de la VPC, y sus preferencias de Session Manager registran la salida de la sesión en un grupo de Registros de CloudWatch. Para obtener más información, consulte Creación de puntos de enlace de la VPC para Systems Manager.
- Solución D: se ha eliminado el grupo de registros o el bucket de Amazon S3 que especificó en sus preferencias de sesión. Para resolver este problema, actualice sus preferencias de sesión con un grupo de registros o un bucket de S3 válidos.
- Solución E: el grupo de registros o el bucket de Amazon S3 que especificó en sus preferencias de sesión no está cifrado, pero ha establecido el elemento de entrada `cloudWatchEncryptionEnabled` o `s3EncryptionEnabled` en `true`. Para resolver este problema, actualice sus preferencias de sesión con un grupo de registros o un bucket de Amazon S3 que estén cifrados o establezca el elemento de entrada `cloudWatchEncryptionEnabled` o `s3EncryptionEnabled` en `false`. Este escenario solo se aplica a los clientes que crean preferencias de sesión mediante las herramientas de línea de comandos.

El nodo administrado deja de responder durante las sesiones de larga ejecución

Problema: el nodo administrado deja de responder o se bloquea durante una sesión de larga ejecución.

Solución: reducir la duración de la retención de registros de SSM Agent para Session Manager.

Para reducir la duración de la retención de registros de SSM Agent de las sesiones

1. Busque `amazon-ssm-agent.json.template` en el directorio `/etc/amazon/ssm/` para Linux o `C:\Program Files\Amazon\SSM` para Windows.
2. Copie el contenido de `amazon-ssm-agent.json.template` en un nuevo archivo dentro del mismo directorio denominado `amazon-ssm-agent.json`.
3. Disminuya el valor predeterminado de `SessionLogsRetentionDurationHours` en la propiedad de `SSM` y guarde el archivo.
4. Restablecer SSM Agent

Se ha producido un error (InvalidDocument) al llamar a la operación StartSession

Problema: Aparece el siguiente error al iniciar una sesión empleando la 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.
```

Solución: El documento SSM que especificó para el parámetro `--document-name` no es un documento Session. Utilice el siguiente procedimiento para ver una lista de documentos Session en la AWS Management Console.

Para ver una lista de documentos Session

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. En la lista Categorías, seleccione Documentos Session.

AWS Systems Manager State Manager

State Manager, una herramienta de AWS Systems Manager, es un servicio de administración de configuración seguro y escalable que automatiza el proceso de mantener los nodos administrados y otros recursos de AWS en el estado que defina. Para comenzar a utilizar State Manager, abra la consola de Systems Manager. En el panel de navegación, elija State Manager.

> **Note**
>
> State Manager y Maintenance Windows pueden realizar algunos tipos similares de actualizaciones en los nodos administrados. La opción que elija dependerá de si necesita automatizar la conformidad del sistema o realizar tareas de alta prioridad y urgencia durante los periodos que especifique.
> Para obtener más información, consulte Elección entre State Manager y Maintenance Windows.

¿Cómo puede State Manager beneficiar a mi organización?

Mediante documentos de Systems Manager previamente configurados (documentos de SSM), State Manager ofrece los siguientes beneficios para la administración de nodos:

- Arrancar nodos con software específico en el inicio.
- Descargar y actualizar agentes en una programación definida, incluido SSM Agent.
- Configure los valores de red.
- Unir nodos a un dominio de Microsoft Active Directory.
- Ejecutar scripts en nodos administrados de Windows, Linux y macOS a lo largo de su ciclo de vida.

Para administrar la desviación de la configuración entre otros recursos de AWS, puede utilizar Automatización, una herramienta de Systems Manager, con State Manager para llevar a cabo los siguientes tipos de tareas:

- Adjuntar un rol de Systems Manager a las instancias de Amazon Elastic Compute Cloud (Amazon EC2) para transformarlas en nodos administrados.
- aplicar las reglas de entrada y salida que desee para un grupo de seguridad
- crear o eliminar copias de seguridad de Amazon DynamoDB

- crear o eliminar instantáneas de Amazon Elastic Block Store (Amazon EBS)
- desactivar los permisos de lectura y escritura en los buckets de Amazon Simple Storage Service (Amazon S3)
- Iniciar, reiniciar o detener los nodos administrados y las instancias de Amazon Relational Database Service (Amazon RDS).
- aplicar revisiones a las AMIs de Linux, macOS y Windows

Para obtener información acerca del uso de State Manager con manuales de procedimientos de Automation, consulte Programación de automatizaciones con asociaciones de State Manager.

¿Quién debe utilizar State Manager?

State Manager es una solución adecuada para cualquier cliente de AWS que quiera mejorar la administración y la gobernanza de sus recursos de AWS y reducir la desviación de la configuración.

¿Cuáles son las características de State Manager?

Estas son algunas de las características clave de State Manager:

- Asociaciones de State Manager

  Una asociación de State Manager es una configuración que asigna a sus recursos de AWS. La configuración define el estado que desea mantener en los recursos. Por ejemplo, una asociación puede especificar que el software antivirus debe estar instalado y ejecutándose en un nodo administrado, o bien que determinados puertos deben estar cerrados.

  Una asociación especifica una programación del momento en que aplicar la configuración y los destinos para la asociación. Por ejemplo, una asociación para software antivirus puede ejecutarse una vez al día en todos los nodos administrados de una Cuenta de AWS. Si el software no está instalado en un nodo, la asociación podría exigir a State Manager que lo instale. Si el software está instalado, pero el servicio no se está ejecutando, la asociación podría exigir a State Manager que inicie el servicio.
- Opciones de programación flexibles

  State Manager ofrece las siguientes opciones de programación cuando se ejecuta una asociación:

  - Procesamiento inmediato o retrasado

Al crear una asociación, de forma predeterminada, el sistema la ejecuta de inmediato en los recursos especificados. Después de la ejecución inicial, la asociación se ejecuta en intervalos de acuerdo con la programación que definió.

Puede exigir a State Manager que no ejecute una asociación inmediatamente mediante la opción Apply association only at the next specified Cron interval (Aplicar la asociación solo en el siguiente intervalo cron especificado) de la consola o el parámetro `ApplyOnlyAtCronInterval` de la línea de comandos.

- Expresiones cron y rate

  Al crear una asociación, especifica una programación para el momento en que State Manager aplica la configuración. State Manager admite la mayoría de expresiones cron y rate estándar para programar el momento en que se ejecuta una asociación. State Manager también admite expresiones cron que incluyen un día de la semana y el signo de número (#) para designar el día x de un mes para ejecutar una asociación y el signo (L) para indicar el último día X del mes.

  > **Note**
  >
  > State Manager actualmente no admite especificar meses en expresiones cron para asociaciones.

  Para tener un mayor control sobre el momento en el que se ejecuta una asociación, por ejemplo, si desea ejecutar una asociación dos días después de la revisión del martes, puede especificar un desplazamiento. Un desplazamiento define los días que hay que esperar después del día programado para ejecutar una asociación.

  Para obtener información acerca de cómo crear expresiones cron y rate, consulte Referencia: expresiones cron y rate para Systems Manager.

- Varias opciones de destino

  Una asociación especifica también sus destinos. State Manager admite indicar los destinos de los recursos de AWS mediante etiquetas, AWS Resource Groups, ID de nodos individuales o todos los nodos administrados de la Región de AWS y la Cuenta de AWS actuales.

- Compatibilidad con Amazon S3

Almacene la salida del comando de las ejecuciones de asociaciones en el bucket de Amazon S3 de su elección. Para obtener más información, consulte Trabajo con asociaciones en Systems Manager.

- Compatibilidad con EventBridge

  Esta herramienta de Systems Manager se admite como un tipo de evento y un tipo de destino en las reglas de Amazon EventBridge. Para obtener más información, consulte Monitoreo de eventos de Systems Manager con Amazon EventBridge y Referencia: patrones y tipos de eventos de Amazon EventBridge para Systems Manager.

¿Se cobra por usar State Manager?

State Manager está disponible sin costo adicional.

Temas

- Funcionamiento de State Manager
- Trabajo con asociaciones en Systems Manager
- Cómo crear asociaciones que ejecutan archivos MOF
- Creación de asociaciones que ejecuten manuales de estrategia de Ansible
- Creación de asociaciones que ejecuten recetas de Chef
- Tutorial: actualización automática de SSM Agent con AWS CLI
- Tutorial: actualización automática de los controladores PV en las instancias de EC2 de Windows Server

Más información

- Combatir la desviación de la configuración con Amazon EC2 Systems Manager y Windows PowerShell DSC
- Configure Amazon EC2 Instances in an Auto Scaling Group Using State Manager (Configuración de instancias de Amazon EC2 en un grupo de escalado automáticomediante State Manager)

Funcionamiento de State Manager

State Manager, una herramienta de AWS Systems Manager, es un servicio seguro y escalable que automatiza el proceso de mantener los nodos administrados en una infraestructura híbrida y multinube en el estado que defina.

Así es como funciona State Manager:

1. Determine el estado que desea aplicar a sus recursos de AWS.

   ¿Desea garantizar que los nodos administrados estén configurados con aplicaciones específicas, como, por ejemplo, las aplicaciones antivirus o malware? ¿Desea automatizar el proceso de actualización de SSM Agent u otros paquetes de AWS como `AWSPVDriver`? ¿Necesita garantizar que los puertos específicos se cierren o abran? Para comenzar a utilizar State Manager, determine el estado que desea aplicar a los recursos de AWS. El estado que desea aplicar determina qué documento de SSM se utiliza para crear una asociación de State Manager.

   Una asociación de State Manager es una configuración que asigna a sus recursos de AWS. La configuración define el estado que desea mantener en los recursos. Por ejemplo, una asociación puede especificar que el software antivirus debe estar instalado y ejecutándose en un nodo administrado, o bien que determinados puertos deben estar cerrados.

   Una asociación especifica una programación del momento en que aplicar la configuración y los destinos para la asociación. Por ejemplo, una asociación para software antivirus puede ejecutarse una vez al día en todos los nodos administrados de una Cuenta de AWS. Si el software no está instalado en un nodo, la asociación podría exigir a State Manager que lo instale. Si el software está instalado, pero el servicio no se está ejecutando, la asociación podría exigir a State Manager que inicie el servicio.

2. Determine si un documento de SSM preconfigurado puede ayudarlo a crear el estado deseado en los recursos de AWS.

   Systems Manager incluye decenas de documentos de SSM preconfigurados que puede utilizar para crear una asociación. Los documentos preconfigurados están listos para llevar a cabo tareas comunes, como instalar aplicaciones, configurar Amazon CloudWatch, ejecutar automatizaciones de AWS Systems Manager, ejecutar scripts de PowerShell y Shell, y unir nodos administrados a un dominio de Directory Service para Active Directory.

   Puede ver todos los documentos de SSM en la consola de Systems Manager. Elija el nombre de un documento para obtener más información acerca de cada uno de ellos. A continuación, se incluyen dos ejemplos: `AWS-ConfigureAWSPackage` y `AWS-InstallApplication`.

3. Cree una asociación.

Puede crear una asociación mediante la consola de Systems Manager, AWS Command Line Interface (AWS CLI), AWS Tools for Windows PowerShell (Tools for Windows PowerShell) o la API de Systems Manager. Al crear una asociación, debe especificar la siguiente información:

- Un nombre para la asociación.
- Los parámetros del documento de SSM (por ejemplo, la ruta a la aplicación que desee instalar o el script que se va a ejecutar en los nodos).
- Destinos para la asociación. Puede definir el destino de los nodos administrados especificando etiquetas, eligiendo identificadores de nodo individuales o eligiendo un grupo en AWS Resource Groups. También puede definir el destino de todos los nodos administrados en la Región de AWS y la Cuenta de AWS actuales. Si sus destinos incluyen más de 1.000 nodos, el sistema utiliza un mecanismo de limitación horaria. Esto significa que es posible que vea imprecisiones en el recuento de agregación de estados, ya que este proceso se ejecuta por hora y solo cuando cambia el estado de ejecución de un nodo.
- Una programación para indicar cuándo o con qué frecuencia se aplicará el estado. Puede especificar una expresión cron o rate. Para obtener más información acerca de la creación de programas usando expresiones Cron y Rate, consulte Expresiones cron y rate para asociaciones.

> **Note**
>
> State Manager actualmente no admite especificar meses en expresiones cron para asociaciones.

Cuando ejecuta el comando para crear la asociación, Systems Manager vincula la información que se especifica (programación, destinos, documento de SSM y parámetros) a los recursos de destino. El estado de la asociación inicialmente muestra Pending (Pendiente) pues el sistema intenta llegar a todos los destinos y aplicar inmediatamente el estado especificado en la asociación.

> **Note**
>
> Si crea una nueva asociación que está programada para ejecutarse mientras todavía se está ejecutando una asociación anterior, se agota el tiempo de espera de la asociación y se ejecuta la nueva asociación.

Systems Manager indica el estado de la solicitud para crear asociaciones en los recursos. Puede consultar los detalles de estado en la consola o, en el caso de los nodos administrados, mediante la operación DescribeInstanceAssociationsStatus de la API. Si elige escribir el resultado del comando en Amazon Simple Storage Service (Amazon S3) cuando crea una asociación, también podrá consultar el resultado en el bucket de Simple Storage Service (Amazon S3) que haya especificado.

Para obtener más información, consulte Trabajo con asociaciones en Systems Manager.

> **Note**
>
> Las operaciones de la API que inicia el documento SSM durante la ejecución de una asociación no se registran en AWS CloudTrail.

4. Monitorización y actualización.

Después de crear la asociación, State Manager vuelve a aplicar la configuración de acuerdo con la programación definida en la asociación. Puede ver el estado de sus asociaciones en la página de State Manager de la consola o llamando directamente al ID de asociación generado por Systems Manager cuando creó dicha asociación. Para obtener más información, consulte Visualización de los historiales de asociación. Puede actualizar los documentos de asociación y volver a aplicarlos según sea necesario. También puede crear varias versiones de una asociación. Para obtener más información, consulte Cómo editar y crear una nueva versión de una asociación.

Cómo saber cuándo se aplican las asociaciones a los recursos

Cuando crea una asociación, especifica un documento de SSM que define la configuración, una lista de recursos de destino y una programación para aplicar la configuración. De manera predeterminada, State Manager ejecuta la asociación cuando se crea, y luego según su programación. State Manager también intenta ejecutar la asociación en las siguientes situaciones:

- Edición de asociación: State Manager ejecuta la asociación después de que un usuario edite y guarde los cambios realizados en cualquiera de los siguientes campos de la asociación: `DOCUMENT_VERSION`, `PARAMETERS`, `SCHEDULE_EXPRESSION`, `OUTPUT_S3_LOCATION`.
- Edición de documento: State Manager ejecuta la asociación después de que un usuario edite y guarde los cambios realizados en el documento SSM que define el estado de configuración de la

asociación. Concretamente, la asociación se ejecuta después de realizar las siguientes ediciones en el documento:

- Un usuario especifica una nueva versión del documento `$DEFAULT` y la asociación se creó utilizando la versión `$DEFAULT`.
- Un usuario actualiza un documento y la asociación se creó utilizando la versión `$LATEST`.
- Un usuario elimina el documento que se especificó cuando se creó la asociación.

- Inicio manual: State Manager ejecuta la asociación cuando la inicia el usuario desde la consola de Systems Manager o mediante programación.
- Cambios de destino: State Manager ejecuta la asociación después de que se produzca alguna de las siguientes actividades en un nodo de destino:
  - Un nodo administrado se conecta por primera vez.
  - Un nodo administrado se conecta después de perder una ejecución de asociación programada.
  - Un nodo administrado se conecta después de haber estado detenido durante más de 30 días.

> **Note**
>
> State Manager no supervisa los documentos o paquetes utilizados en las asociaciones entre Cuentas de AWS. Si actualiza un documento o paquete en una cuenta, la actualización no hará que la asociación se ejecute en la segunda cuenta. Debe ejecutar la asociación de forma manual en la segunda cuenta.

Cómo impedir que las asociaciones se ejecuten cuando cambia un destino

En algunos casos, es posible que no quiera que una asociación se ejecute cuando un destino formado por nodos gestionados cambia, sino únicamente de acuerdo con la programación especificada.

> **Note**
>
> La ejecución de un manual de procedimientos de automatización tiene un costo. Si la asociación con un manual de procedimientos de automatización se dirige a todas las instancias de su cuenta y lanza un gran número de instancias con regularidad, el manual

de procedimientos se ejecuta en cada una de las instancias cuando se lanza. Esto puede provocar un aumento de los gastos de automatización.

Para evitar que se ejecute una asociación cuando cambian los destinos de esa asociación, seleccione la casilla Aplicar la asociación solo en el siguiente intervalo cron especificado. Esta casilla se encuentra en el área Especificar programación de las páginas Crear asociación y Editar asociación.

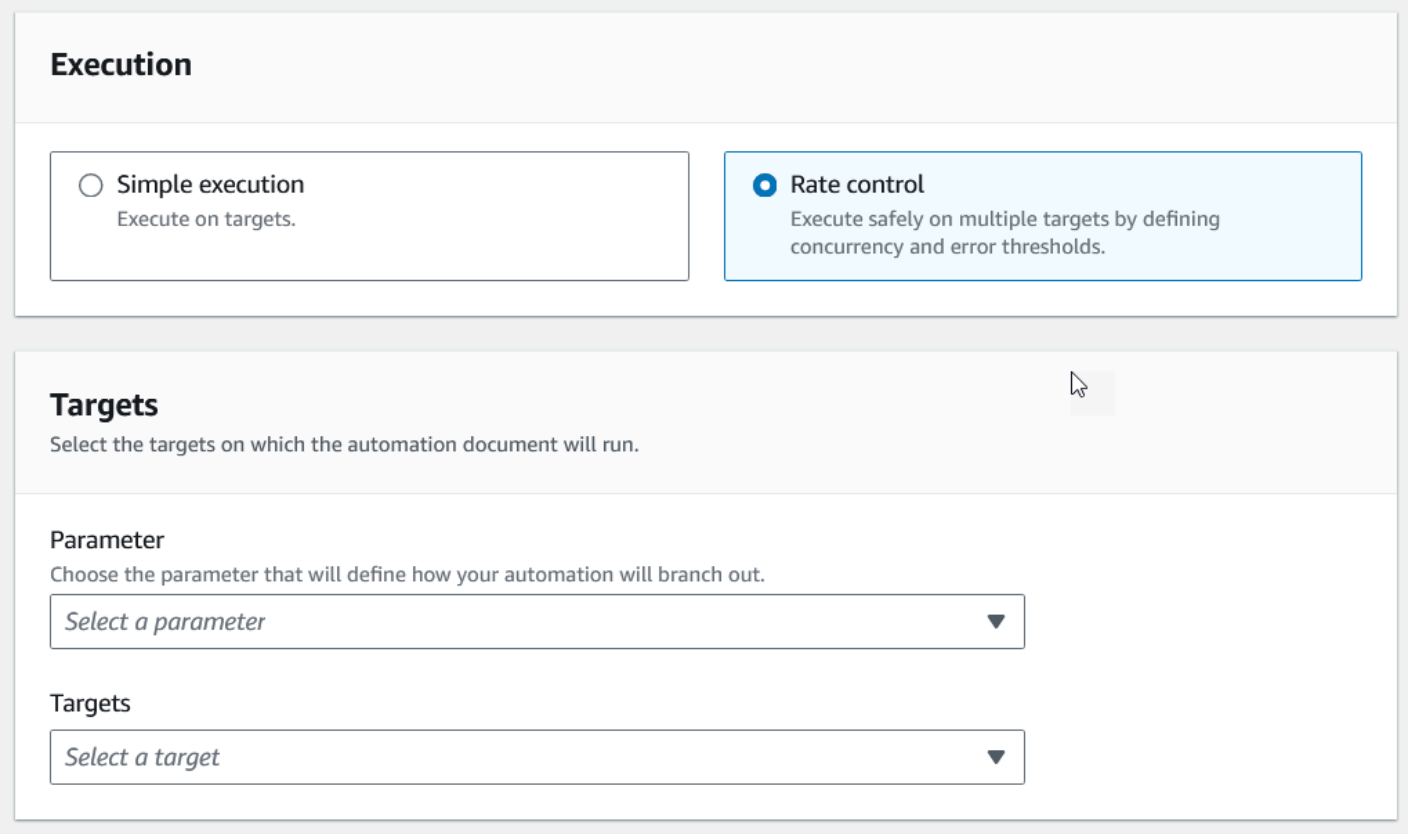
Esta opción se aplica a las asociaciones que incorporan un manual de procedimientos de automatización o un documento de SSM.

Acerca de las actualizaciones de destino con los manuales de procedimientos de Automatización

Para que las asociaciones que se crean con los manuales de procedimientos de Automatización se apliquen cuando se detecten nuevos nodos de destino, se deben cumplir las siguientes condiciones:

- La asociación debe haberla creado una configuración de Quick Setup. Quick Setup es una herramienta de AWS Systems Manager. Actualmente, no se admiten asociaciones creadas por otros procesos.
- El manual de procedimientos de Automatización se debe dirigir explícitamente al tipo de recurso `AWS::EC2::Instance` o `AWS::SSM::ManagedInstance`.
- La asociación debe especificar los parámetros y los destinos.

  En la consola, los campos Parámetro y Destinos se muestran al seleccionar una ejecución de control de velocidad.



Cuando use las acciones de la API CreateAssociation, CreateAssociationBatch o UpdateAssociation, puede especificar estos valores con las entradas `AutomationTargetParameterName` y `Targets`. En cada una de estas acciones de la API, también puede configurar el parámetro `ApplyOnlyAtCronInterval` en `true` para evitar que la asociación se ejecute cada vez que cambia un destino.

Para obtener información sobre el uso de la consola para controlar cuándo se ejecutan las asociaciones, incluidos los detalles para evitar costos inesperadamente elevados en las ejecuciones de la automatización, consulte Cómo saber cuándo se aplican las asociaciones a los recursos.

Trabajo con asociaciones en Systems Manager

En esta sección se describe cómo crear y administrar las asociaciones de State Manager mediante la consola de AWS Systems Manager, AWS Command Line Interface (AWS CLI) y Herramientas de AWS para PowerShell.

Temas

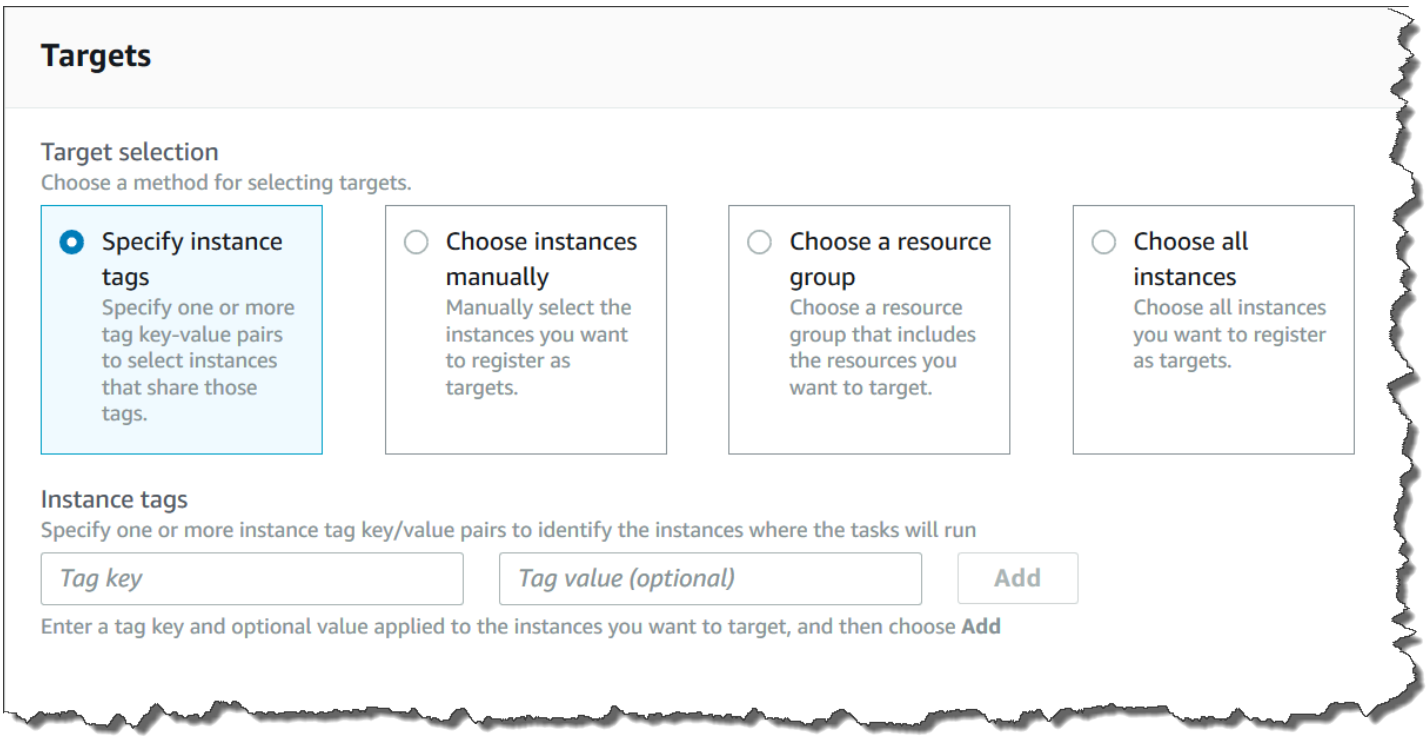
- Cómo comprender los controles de frecuencia y destinos en las asociaciones de State Manager
- Cómo crear asociaciones
- Cómo editar y crear una nueva versión de una asociación
- Eliminación de una asociación
- Ejecución de grupos de Auto Scaling con asociaciones
- Visualización de los historiales de asociación
- Uso de asociaciones mediante IAM

Cómo comprender los controles de frecuencia y destinos en las asociaciones de State Manager

Este tema describe las características de State Manager que lo ayudan a implementar una asociación en decenas o cientos de nodos mientras controla la cantidad de nodos que ejecutan la asociación a la hora programada. State Manager es una herramienta de AWS Systems Manager.

Cómo utilizar destinos

Cuando crea una asociación de State Manager, usted elige los nodos que desea configurar con la asociación en la sección Targets (Destinos) de la consola de Systems Manager, como se muestra aquí.



Si crea una asociación mediante una herramienta de línea de comandos como la AWS Command Line Interface (AWS CLI), especifique el parámetro `targets`. La definición del destino de los nodos le permite configurar decenas, cientos o miles de nodos con una asociación sin tener que especificar ni elegir ID de nodo individuales.

Cada nodo administrado puede ser el objetivo de un máximo de 20 asociaciones.

State Manager incluye las siguientes opciones de destino al crear una asociación.

Specify tags (Especificar etiquetas)

Utilice esta opción para especificar una clave de etiqueta y (opcionalmente) un valor de etiqueta asignado a los nodos. Al ejecutar la solicitud, el sistema localiza e intenta crear la asociación en todos los nodos que coinciden con la clave y el valor de etiqueta especificados. Si ha especificado varios valores de etiquetas, la asociación se dirige hacia cualquier nodo con al menos uno de esos valores de etiquetas. Cuando el sistema crea inicialmente la asociación, la ejecuta. Después de esta ejecución inicial, el sistema ejecuta la asociación de acuerdo con la programación especificada.

Si crea nuevos nodos y asigna la clave y el valor de etiqueta especificados a esos nodos, el sistema aplica automáticamente la asociación, la ejecuta inmediatamente y, a continuación, la ejecuta de acuerdo con la programación. Esto se aplica cuando la asociación utiliza un documento de comando o política y no se aplica si la asociación utiliza un manual de procedimientos de Automation. Si elimina las etiquetas especificadas de un nodo, el sistema dejará de ejecutar la asociación en esos nodos.

> **Note**
>
> Si usa manuales de procedimiento de automatización con State Manager y la limitación de etiquetado le impide alcanzar un objetivo específico, considere usar manuales de procedimiento de automatización con Amazon EventBridge. Para obtener más información, consulte Ejecución de automatizaciones a partir de eventos EventBridge. Para obtener información acerca del uso de State Manager con manuales de procedimientos de Automation, consulte Programación de automatizaciones con asociaciones de State Manager.

Como práctica recomendada, recomendamos usar etiquetas al crear asociaciones que usen un documento de comando o de política. También recomendamos usar etiquetas al crear asociaciones

para ejecutar grupos de escalado automático. Para obtener más información, consulte Ejecución de grupos de Auto Scaling con asociaciones.

> **Note**
>
> Observe la siguiente información.
>
> - Al crear una asociación en la AWS Management Console que se dirija a nodos mediante etiquetas, solo puede especificar una clave de etiqueta para una asociación de automatización y cinco claves de etiqueta para una asociación de comandos. Todas las claves de etiqueta especificadas en la asociación deben estar asignadas actualmente al nodo. Si no lo están, State Manager no se dirige al nodo para establecer una asociación.
> - Si quiere usar la consola y desea dirigirse a sus nodos mediante más de una clave de etiqueta para una asociación de automatización y cinco claves de etiqueta para una asociación de comando, asigne las claves de etiqueta a un grupo de AWS Resource Groups y agréguele los nodos. A continuación, puede elegir la opción Grupo de recursos en la lista de objetivos al crear la asociación con State Manager.
> - Puede especificar un máximo de cinco claves de etiqueta mediante AWS CLI. Si utiliza las AWS CLI, todas las claves de etiqueta especificadas en el comando `create-association` deben estar asignadas actualmente al nodo. Si no lo están, State Manager no se dirige al nodo para establecer una asociación.

Cómo elegir los nodos manualmente

Utilice esta opción para seleccionar manualmente los nodos en los que desea crear la asociación. El panel Instances (Instancias) muestra todos los nodos administrados de Systems Manager de la Cuenta de AWS y Región de AWS actuales. Puede seleccionar manualmente tantos nodos como desee. Cuando el sistema crea inicialmente la asociación, la ejecuta. Después de esta ejecución inicial, el sistema ejecuta la asociación de acuerdo con la programación especificada.

> **Note**
>
> Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

Cómo elegir un grupo de recursos

Utilice esta opción para crear una asociación en todos los nodos devueltos por una consulta de AWS Resource Groups basada en etiquetas o por una consulta de AWS CloudFormation basada en la pila.

A continuación, puede encontrar detalles acerca del establecimiento del destino de grupos de recursos para una asociación.

- Si agrega nuevos nodos a un grupo, el sistema asigna automáticamente los nodos a la asociación que segmenta el grupo de recursos. El sistema aplica la asociación a los nodos cuando detecta el cambio. Después de esta ejecución inicial, el sistema ejecuta la asociación de acuerdo con la programación especificada.
- Si crea una asociación dirigida a un grupo de recursos y el tipo de recurso `AWS::SSM::ManagedInstance` se especificó para ese grupo y, por diseño, la asociación se ejecuta en instancias de Amazon Elastic Compute Cloud (Amazon EC2) y nodos que no sean de EC2 en un entorno híbrido y multinube.

  Y viceversa. Si crea una asociación dirigida a un grupo de recursos y el tipo de recurso `AWS::EC2::Instance` se especificó para ese grupo y, por diseño, la asociación se ejecuta en instancias (Amazon EC2) y nodos que no sean de EC2 en un entorno híbrido y multinube.
- Si crea una asociación que se dirige a un grupo de recursos, ese grupo de recursos no debe tener asignadas más de cinco claves de etiqueta o más de cinco valores especificados para cualquier clave de etiqueta individual. Si se da cualquiera de estas condiciones en las etiquetas y claves asignadas al grupo de recursos, la asociación no se ejecuta y devuelve un error `InvalidTarget`.
- Si crea una asociación que se dirige a un grupo de recursos mediante etiquetas, no puede elegir la opción (valor vacío) para el valor de la etiqueta.
- Si elimina un grupo de recursos, todas las instancias de ese grupo dejarán de ejecutar la asociación. Como práctica recomendada, elimine las asociaciones que establece el destino del grupo.
- Como máximo, puede definir el destino de un único grupo de recursos para una asociación. No se admiten grupos múltiples o anidados.
- Después de crear una asociación, State Manager actualiza periódicamente la asociación con información sobre los recursos del grupo de recursos. Si agrega nuevos recursos a un grupo de recursos, la programación que indica cuándo el sistema aplica la asociación a los nuevos recursos depende de varios factores. Puede determinar el estado de la asociación en la página de State Manager de la consola de Systems Manager.

> **Warning**
>
> Un usuario, grupo o rol de AWS Identity and Access Management (IAM) con permiso para crear una asociación que establece el destino de un grupo de recursos de instancias de Amazon EC2 tiene automáticamente control de nivel de raíz de todas las instancias del grupo. Solo se debe permitir a los administradores de confianza crear asociaciones.

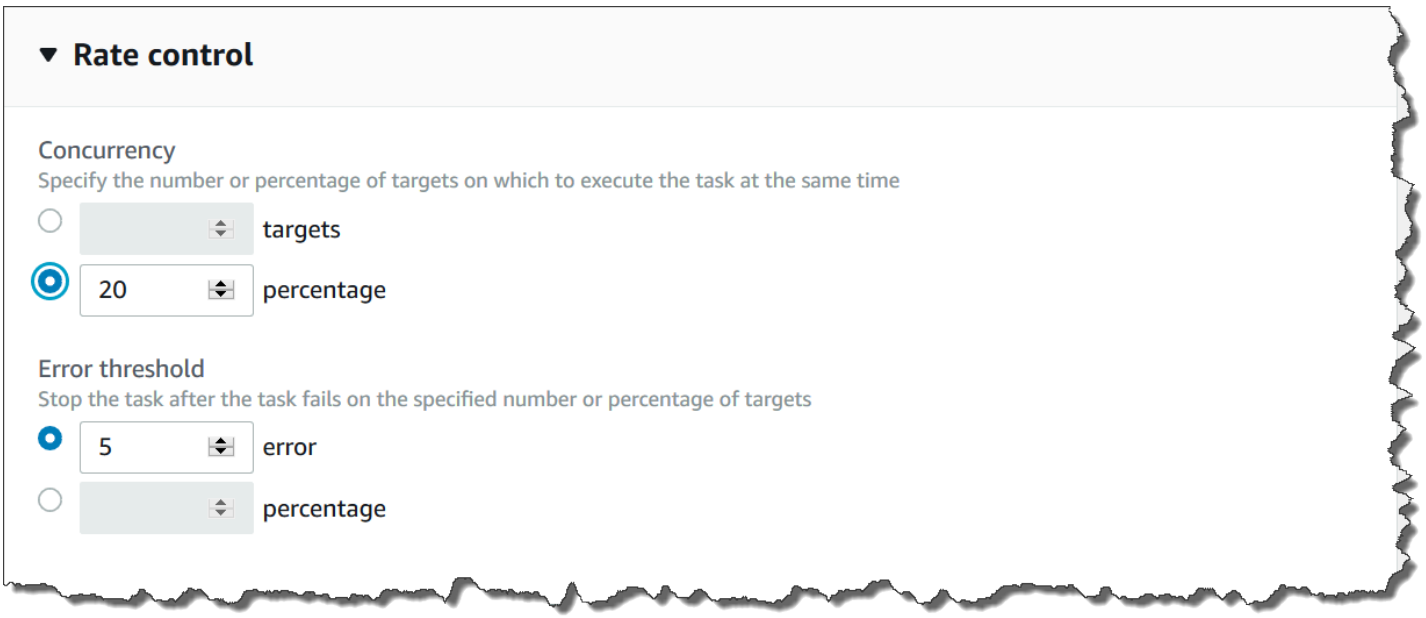
Para obtener más información acerca de Resource Groups, consulte ¿Qué es AWS Resource Groups? en la Guía del usuario de AWS Resource Groups.

Choose all nodes (Elegir todos los nodos)

Utilice esta opción para definir el destino de todos los nodos de la Cuenta de AWS y Región de AWS actuales. Cuando ejecuta la solicitud, el sistema localiza e intenta crear la asociación en todos los nodos de la Cuenta de AWS y Región de AWS actuales. Cuando el sistema crea inicialmente la asociación, la ejecuta. Después de esta ejecución inicial, el sistema ejecuta la asociación de acuerdo con la programación especificada. Si crea nuevos nodos, el sistema aplica automáticamente la asociación, la ejecuta inmediatamente y, a continuación, la ejecuta de acuerdo con la programación.

Cómo utilizar controles de velocidad

Puede controlar la ejecución de una asociación en los nodos especificando un valor de simultaneidad y un umbral de error. El valor de simultaneidad especifica cuántos nodos pueden ejecutar la asociación a la vez. Un umbral de error especifica cuántas ejecuciones de asociaciones pueden producir un error antes de que Systems Manager envíe un comando a cada nodo configurado con esa asociación para detener su ejecución. El comando deja de ejecutar la asociación hasta la próxima ejecución programada. Las características de umbral de simultaneidad y error se denominan colectivamente controles de frecuencia.



Simultaneidad

La simultaneidad ayuda a limitar el impacto en los nodos al permitirle especificar que solo un determinado número de nodos pueden procesar una asociación a la vez. Puede especificar un número absoluto de nodos, por ejemplo 20, o un porcentaje del conjunto de nodos de destino, por ejemplo, 10 %.

La simultaneidad de State Manager tiene las siguientes restricciones y limitaciones:

- Si decide crear una asociación mediante el uso de destinos, pero no quiere especificar un valor de simultaneidad, State Manager fuerza automáticamente una simultaneidad máxima de 50 nodos.
- Si los nodos nuevos que coinciden con los criterios de destino se conectan al mismo tiempo que se ejecuta una asociación que utiliza simultaneidad, los nuevos nodos ejecutarán la asociación si no se supera el valor de simultaneidad. Si el valor de simultaneidad se supera, los nodos se pasan por alto durante el intervalo de ejecución de la asociación actual. Los nodos ejecutarán la asociación durante el siguiente intervalo programado si cumplen los requisitos de simultaneidad.
- Si actualiza una asociación que utiliza la simultaneidad y uno o varios nodos están procesando esa asociación cuando se actualiza, entonces se podrá completar cualquier nodo que esté ejecutando la asociación. Aquellas asociaciones que no se hayan iniciado se detienen. Después de finalizar la ejecución de las asociaciones, todos los nodos de destino ejecutarán inmediatamente la asociación de nuevo, ya que se ha actualizado. Cuando la asociación se ejecuta de nuevo, el valor de simultaneidad se aplica.

Umbrales de error

Un umbral de error especifica cuántas ejecuciones de asociaciones pueden fallar antes de que Systems Manager envíe un comando a cada nodo configurado con esa asociación. El comando deja de ejecutar la asociación hasta la próxima ejecución programada. Puede especificar un número absoluto de errores, por ejemplo, 10 o un porcentaje del destino definido, por ejemplo, el 10 %.

Si especifica un número absoluto de tres errores, por ejemplo, State Manager envía el comando de detención cuando se devuelve el cuarto error. Si se especifica 0, State Manager envía el comando de detención tras el primer resultado de error que se devuelva.

Si especifica un umbral de error del 10 % para 50 asociaciones, State Manager envía el comando de detención cuando se devuelve el sexto error. Las asociaciones que ya se están ejecutando cuando se alcanza un umbral de errores tienen permiso para completarse, pero algunas de ellas también pueden generar un error. Para asegurarse de que no haya más errores que el número especificado para el umbral de error, establezca el valor de Concurrency (Simultaneidad) en 1 de modo que las asociaciones sigan procesándose de una en una.

Los umbrales de error de State Manager tienen en cuenta las siguientes restricciones y limitaciones:

- Los umbrales de error se aplican para el intervalo actual.
- La información sobre cada error, incluidos los detalles de nivel de paso, se registran en el historial de asociaciones.
- Si decide crear una asociación mediante el uso de destinos, pero no quiere especificar un umbral de error, State Manager forzará automáticamente un umbral de error del 100 %.

Cómo crear asociaciones

State Manager, una herramienta de AWS Systems Manager, lo ayuda a mantener los recursos de AWS en el estado que defina y a reducir la desviación de la configuración. Para ello, State Manager utiliza asociaciones. Una asociación es una configuración que asigna a los recursos de AWS. La configuración define el estado que desea mantener en los recursos. Por ejemplo, una asociación puede especificar que el software antivirus debe estar instalado y ejecutándose en un nodo administrado, o bien que determinados puertos deben estar cerrados.

Una asociación especifica una programación del momento en que aplicar la configuración y los destinos para la asociación. Por ejemplo, una asociación para software antivirus puede ejecutarse una vez al día en todos los nodos administrados de una Cuenta de AWS. Si el software no está instalado en un nodo, la asociación podría exigir a State Manager que lo instale. Si el software está

instalado, pero el servicio no se está ejecutando, la asociación podría exigir a State Manager que inicie el servicio.

> **Warning**
>
> Al crear una asociación, puede elegir un grupo de recursos de AWS de nodos administrados como destino de la asociación. Si un usuario, grupo o rol de AWS Identity and Access Management (IAM) que tiene permiso para crear una asociación que establece el destino de un grupo de recursos de nodos administrados, dicho usuario, grupo o rol tiene automáticamente control de nivel de raíz de todos los nodos del grupo. Solo se permite a los administradores de confianza crear asociaciones.

Objetivos de asociación y controles de tasas

Una asociación especifica qué nodos administrados, o destinos, deben recibir la asociación. State Manager incluye varias características para ayudarlo a definir el destino de los nodos administrados y controlar cómo se implementa la asociación en esos destinos. Para obtener más información acerca de los controles de velocidad y destinos, consulte Cómo comprender los controles de frecuencia y destinos en las asociaciones de State Manager.

Cómo etiquetar asociaciones

Puede asignar etiquetas a una asociación al crearla mediante una herramienta de línea de comandos como la AWS CLI o Herramientas de AWS para PowerShell. No se admite agregar etiquetas a una asociación mediante la consola de Systems Manager.

Cómo correr asociaciones

De forma predeterminada, State Manager ejecuta una asociación inmediatamente después de crearla y, a continuación, según la programación que haya definido.

El sistema también ejecuta asociaciones de acuerdo con las siguientes reglas:

- State Manager intenta ejecutar la asociación en todos los nodos de destino o especificados durante un intervalo.
- Si una asociación no se ejecuta durante un intervalo (porque, por ejemplo, un valor de simultaneidad limita el número de nodos que podría procesar la asociación simultáneamente), State Manager intenta ejecutar la asociación durante el intervalo siguiente.

- State Manager ejecuta la asociación después de realizar cambios en la configuración, los nodos de destino, los documentos o los parámetros de la asociación. Para obtener más información, consulte Cómo saber cuándo se aplican las asociaciones a los recursos
- State Manager registra el historial de todos los intervalos omitidos. Puede ver el historial en la pestaña Execution History (Historial de ejecución).

Asociaciones de programación

Puede programar las asociaciones para que se ejecuten a intervalos básicos, por ejemplo, cada 10 horas, o puede crear programaciones más avanzadas mediante expresiones de frecuencia y cron personalizadas. También puede impedir que las asociaciones se ejecuten al crearlas por primera vez.

Cómo utilizar las expresiones cron y rate para programar ejecuciones de asociaciones

Además de las expresiones estándar de cron y rate, State Manager también admite expresiones de cron que incluyen un día de la semana y el signo de número (#) para designar el n día de un mes para ejecutar una asociación. A continuación, se incluye un ejemplo en el que se ejecuta una programación cron el tercer martes de cada mes a las 23.30 h UTC:

```
cron(30 23 ? * TUE#3 *)
```

A continuación, se incluye un ejemplo que se ejecuta el segundo jueves de cada mes a medianoche (UTC):

```
cron(0 0 ? * THU#2 *)
```

State Manager también admite el signo (L) para indicar el último día X del mes. A continuación, se incluye un ejemplo en el que se ejecuta una programación cron el último martes de cada mes a medianoche (UTC):

```
cron(0 0 ? * 3L *)
```

Para tener un mayor control sobre el momento en el que se ejecuta una asociación, por ejemplo, si desea ejecutar una asociación dos días después de la revisión del martes, puede especificar un desplazamiento. Un desplazamiento define los días que hay que esperar después del día programado para ejecutar una asociación. Por ejemplo, si especificó una programación cron de `cron(0 0 ? * THU#2 *)`, puede especificar el número 3 en el campo Desplazamiento de programación para ejecutar la asociación cada domingo después del segundo jueves del mes.

> **Note**
>
> Para utilizar los desplazamientos, seleccione Aplicar la asociación solo en el siguiente intervalo Cron especificado en la consola o especifique el parámetro `ApplyOnlyAtCronInterval` desde la línea de comandos. Cuando cualquiera de estas opciones está activada, State Manager no ejecuta la asociación inmediatamente después de crearla.

Para obtener más información acerca de las expresiones Cron y Rate, consulte Referencia: expresiones cron y rate para Systems Manager.

Cómo crear una asociación (consola)

El siguiente procedimiento describe cómo utilizar la consola de Systems Manager para crear una asociación de State Manager.

> **Note**
>
> Observe la siguiente información.
>
> - Este procedimiento describe cómo crear una asociación que utilice un documento `Command` o un documento `Policy` para dirigirse a nodos administrados. Para obtener información sobre cómo crear una asociación que utilice un manual de ejecución de Automation para dirigirse a nodos u otros tipos de recursos de AWS, consulte Programación de automatizaciones con asociaciones de State Manager.
> - Al crear una asociación, puede especificar un máximo de cinco claves de etiqueta mediante la AWS Management Console. Todas las claves de etiqueta especificadas para la asociación deben estar asignadas actualmente al nodo. Si no lo están, State Manager no se dirige al nodo para establecer la asociación.

Cómo crear una asociación de State Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija State Manager.
3. Elija Crear asociación.

4. Escriba un nombre en el campo Nombre.
5. En la lista Document (Documento), elija la opción junto al nombre de un documento. Observe el tipo de documento. Este procedimiento se aplica a los documentos `Command` y `Policy`. Para obtener información acerca de cómo crear una asociación que utiliza un manual de procedimientos de Automation, consulte Programación de automatizaciones con asociaciones de State Manager.

   > **Important**
   >
   > State Manager no admite la ejecución de asociaciones que utilizan una nueva versión de un documento si dicho documento se comparte desde otra cuenta. State Manager siempre ejecuta la versión `default` de un documento si se comparte desde otra cuenta, aunque la consola de Systems Manager muestra que se procesó una versión nueva. Si desea ejecutar una asociación con una versión nueva de un documento compartido desde otra cuenta, debe establecer la versión del documento en `default`.

6. En Parameters (Parámetros), especifique los parámetros de entrada requeridos.
7. (Opcional) Elija una alarma de CloudWatch para aplicarla a la asociación de monitoreo.

   > **Note**
   >
   > Tenga en cuenta la siguiente información sobre este paso.
   >
   > - La lista de alarmas muestra un máximo de 100 alarmas. Si no ve su alarma en la lista, utilice la AWS Command Line Interface para crear la asociación. Para obtener más información, consulte Cómo rear una asociación (línea de comandos).
   > - Para adjuntar una alarma de CloudWatch a su comando, la entidad principal de IAM que crea la asociación debe tener permiso para la acción `iam:createServiceLinkedRole`. Para obtener más información sobre las alarmas de CloudWatch, consulte Uso de alarmas de Amazon CloudWatch.
   > - Tenga en cuenta que si la alarma se activa, no se ejecutarán automatizaciones o invocaciones de comandos pendiente.

8. En Targets (Destinos), elija una opción. Para obtener más información acerca del uso de los destinos, consulte Cómo comprender los controles de frecuencia y destinos en las asociaciones de State Manager.

> **Note**
>
> Para que las asociaciones que se crean con los manuales de procedimientos de Automatización se apliquen cuando se detecten nuevos nodos de destino, se deben cumplir algunas condiciones. Para obtener más información, consulta Acerca de las actualizaciones de destino con los manuales de procedimientos de Automatización.

9. En la sección Specify schedule (Especificar programación), elija On Schedule (De forma programada) o No schedule (Sin programación). Si elige On Schedule (De forma programada), utilice los botones proporcionados para crear una programación Cron o Rate para la asociación.

   Si no desea que una asociación se ejecute inmediatamente después de crearla, elija Aplicar la asociación solo en el siguiente intervalo cron especificado.

10. (Opcional) En el campo Schedule offset (Desplazamiento de la programación), especifique un número comprendido entre 1 y 6.

11. En la sección Advanced options (Opciones avanzadas), seleccione la opción Compliance severity (Severidad de la conformidad) para elegir un nivel de severidad para la asociación y utilice Change Calendars (Calendarios de cambios) para elegir un calendario de cambios para la asociación.

    Los informes de conformidad indican si el estado de asociación es conforme o no conforme, junto con el nivel de gravedad que se indique aquí. Para obtener más información, consulte Acerca de la conformidad de las asociaciones de State Manager.

    El calendario de cambios determina cuándo se ejecuta la asociación. Si el calendario está cerrado, la asociación no se aplica. Si el calendario está abierto, la asociación se ejecuta en consecuencia. Para obtener más información, consulte AWS Systems Manager Change Calendar.

12. En la sección Rate control (Control de velocidad), elija las opciones para controlar cómo se ejecuta la asociación en varios nodos. Para obtener más información sobre el uso de controles de velocidad, consulte Cómo comprender los controles de frecuencia y destinos en las asociaciones de State Manager.

    En la sección Simultaneidad, elija una opción:

    - Elija Targets (Destinos) para introducir un número absoluto de destinos que pueda ejecutar la asociación de forma simultánea.

- Elija porcentaje para introducir un porcentaje del destino definido que puede ejecutar la asociación de forma simultánea.

En la sección Umbral de error, elija una opción:

- Elija errors (errores) para especificar un número absoluto de errores permitidos antes de que State Manager deje de ejecutar asociaciones en más destinos.
- Elija percentage (porcentaje) para especificar un porcentaje de errores permitidos antes de que State Manager deje de ejecutar asociaciones en más destinos.

13. (Opcional) En Output options (Opciones de salida), para guardar la salida del comando en un archivo, seleccione el cuadro Enable writing output to S3 (Permitir la escritura de salida en S3). Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancias asignado al nodo administrado, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, verifique que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

A continuación, se presentan los permisos mínimos necesarios para activar la salida de Amazon S3 para una asociación. Puede restringir aún más el acceso al adjuntar políticas de IAM a usuarios o roles dentro de una cuenta. Como mínimo, un perfil de instancias de Amazon EC2 debe tener un rol de IAM con la política administrada `AmazonSSMManagedInstanceCore` y la siguiente política insertada.

JSON

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

```
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:PutObjectAcl"
            ],
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*"
        }
    ]
}
```

Para obtener permisos mínimos, el bucket de Amazon S3 que exporte debe tener la configuración predeterminada definida por la consola de Amazon S3. Para obtener más información acerca de la creación de buckets de Amazon S3, consulte la sección Creación de un bucket en la Guía del usuario de Amazon S3.

> **Note**
>
> Las operaciones de la API que inicia el documento SSM durante la ejecución de una asociación no se registran en AWS CloudTrail.

14. Elija Crear asociación.

> **Note**
>
> Si elimina la asociación creada, la asociación ya no se ejecutará en ningún destino de dicha asociación.

Cómo rear una asociación (línea de comandos)

En el siguiente procedimiento, se describe cómo utilizar la AWS CLI (en Linux o Windows) o Tools for PowerShell para crear una asociación de State Manager. Esta sección incluye varios ejemplos que muestran cómo utilizar los controles de velocidad y destinos. Los controles de velocidad y destinos le permiten asignar una asociación a decenas o cientos de nodos mientras controla la ejecución de esas asociaciones. Para obtener más información acerca de los controles de velocidad y destinos, consulte Cómo comprender los controles de frecuencia y destinos en las asociaciones de State Manager.

> **Important**
>
> Este procedimiento describe cómo crear una asociación que utilice un documento `Command` o un documento `Policy` para dirigirse a nodos administrados. Para obtener información sobre cómo crear una asociación que utilice un manual de ejecución de Automation para dirigirse a nodos u otros tipos de recursos de AWS, consulte Programación de automatizaciones con asociaciones de State Manager.

Antes de empezar

El parámetro `targets` es una matriz de criterios de búsqueda que establece el destino de los nodos mediante la combinación de `Key` y `Value` que especifique. Si tiene previsto crear una asociación en decenas o cientos de nodos mediante el parámetro `targets`, revise las siguientes opciones de establecimiento de destino antes de comenzar el procedimiento.

Direccione los nodos específicos mediante la definición de ID

```
--targets Key=InstanceIds,Values=instance-id-1,instance-id-2,instance-id-3
```

```
--targets
 Key=InstanceIds,Values=i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE,i-07782c72faEXAMPLE
```

Establecer el destino de las instancias mediante etiquetas de

```
--targets Key=tag:tag-key,Values=tag-value-1,tag-value-2,tag-value-3
```

```
--targets Key=tag:Environment,Values=Development,Test,Pre-production
```

Direccione los nodos mediante AWS Resource Groups

```
--targets Key=resource-groups:Name,Values=resource-group-name
```

```
--targets Key=resource-groups:Name,Values=WindowsInstancesGroup
```

Direccione todas las instancias de la Cuenta de AWS y la Región de AWS actuales

```
--targets Key=InstanceIds,Values=*
```

Note

Observe la siguiente información.

- State Manager no admite la ejecución de asociaciones que utilizan una nueva versión de un documento si dicho documento se comparte desde otra cuenta. State Manager siempre ejecuta la versión `default` de un documento si se comparte desde otra cuenta, aunque la consola de Systems Manager muestra que se procesó una versión nueva. Si desea ejecutar una asociación con una versión nueva de un documento compartido desde otra cuenta, debe configurar la versión del documento en`default`.
- State Manager no admite el parámetro `IncludeChildOrganizationUnits` para TargetLocation.
- Puede especificar un máximo de cinco claves de etiqueta mediante AWS CLI. Si utiliza las AWS CLI, todas las claves de etiqueta especificadas en el comando `create-association` deben estar asignadas actualmente al nodo. Si no lo están, State Manager no se dirige al nodo para establecer una asociación.
- Al crear una asociación, especifica cuándo se ejecuta el programa. Especifique el programa mediante una expresión Cron o Rate. Para obtener más información acerca de las expresiones Cron y Rate, consulte Expresiones cron y rate para asociaciones.
- Para que las asociaciones que se crean con los manuales de procedimientos de Automatización se apliquen cuando se detecten nuevos nodos de destino, se deben cumplir algunas condiciones. Para obtener más información, consulta Acerca de las actualizaciones de destino con los manuales de procedimientos de Automatización.

Cómo crear una asociación

1. Si aún no lo ha hecho, instale y configure la AWS CLI o Herramientas de AWS para PowerShell.

   Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.

2. Utilice el formato siguiente para crear un comando que crea una asociación de State Manager. Reemplace cada *`example resource placeholder`* con su propia información.

   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"
```

En el siguiente ejemplo se crea una asociación en los nodos etiquetados con `"Environment,Linux"`. La asociación utiliza el documento `AWS-UpdateSSMAgent` para actualizar SSM Agent en los nodos de destino a las 2:00 h UTC todos los domingos por la mañana. Esta asociación se ejecuta de forma simultánea en un máximo de 10 nodos en cualquier momento. Además, esta asociación deja de ejecutarse en más nodos durante un intervalo de ejecución determinado si el recuento de errores es superior a 5. Para los informes de conformidad, a esta asociación se le asigna un nivel de gravedad medio.

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
```

El ejemplo siguiente se dirige a los ID de nodo mediante la especificación de un valor de comodín (*). Esto permite que Systems Manager cree una asociación en todos los nodos de la Cuenta de AWS y Región de AWS actuales. Esta asociación se ejecuta de forma simultánea en un máximo de 10 nodos en cualquier momento. Además, esta asociación deja de ejecutarse en más nodos durante un intervalo de ejecución determinado si el recuento de errores es superior a 5. Para los informes de conformidad, a esta asociación se le asigna un nivel de gravedad medio. Esta asociación utiliza un desplazamiento de programación, lo que significa que se ejecuta dos días después de la programación cron especificada. También incluye el parámetro `ApplyOnlyAtCronInterval`, que es necesario para utilizar el desplazamiento de programación, lo que significa que la asociación no se ejecutará inmediatamente después de crearla.

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
```

En el siguiente ejemplo, se crea una asociación en los nodos de Resource Groups. El grupo se denomina "HR-Department". La asociación utiliza el documento `AWS-UpdateSSMAgent` para actualizar SSM Agent en los nodos de destino a las 2:00 h UTC todos los domingos por la mañana. Esta asociación se ejecuta de forma simultánea en un máximo de 10 nodos en cualquier momento. Además, esta asociación deja de ejecutarse en más nodos durante un

intervalo de ejecución determinado si el recuento de errores es superior a 5. Para los informes de conformidad, a esta asociación se le asigna un nivel de gravedad medio. Esta asociación se ejecuta en la programación cron especificada. No se ejecuta inmediatamente después de su creación.

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
```

En el siguiente ejemplo, se crea una asociación que se ejecuta en nodos etiquetados con un ID de nodo específico. La asociación utiliza el documento de SSM Agent para actualizar SSM Agent en los nodos de destino una vez cuando el calendario de cambios está abierto. La asociación verifica el estado del calendario cuando se ejecuta. Si el calendario se cierra al momento del lanzamiento y la asociación solo se ejecuta una vez, no se volverá a ejecutar porque la ventana de ejecución de asociación ha pasado. Si el calendario está abierto, la asociación se ejecuta en consecuencia.

> **Note**
>
> Si agrega nuevos nodos a las etiquetas o los grupos de recursos en los que actúa una asociación cuando se cierra el calendario de cambios, la asociación se aplica a esos nodos una vez que se abre el calendario de cambios.

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)"
```

En el siguiente ejemplo, se crea una asociación que se ejecuta en nodos etiquetados con un ID de nodo específico. La asociación utiliza el documento de SSM Agent para actualizar SSM Agent en los nodos de destino a las 2:00 h todos los domingos. Esta asociación solo se ejecuta en la programación cron especificada cuando el calendario de cambios está abierto. Cuando se crea la asociación, verifica el estado del calendario. Si el calendario está cerrado, la asociación no se aplica. Cuando el intervalo para aplicar la asociación comienza a las 2:00 h del domingo, la asociación verifica si el calendario está abierto. Si el calendario está abierto, la asociación se ejecuta en consecuencia.

> **Note**
>
> Si agrega nuevos nodos a las etiquetas o los grupos de recursos en los que actúa una asociación cuando se cierra el calendario de cambios, la asociación se aplica a esos nodos una vez que se abre el calendario de cambios.

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 elimina la asociación creada, la asociación ya no se ejecutará en ningún destino de dicha asociación. Además, si especificó el parámetro `apply-only-at-cron-interval`, puede restablecer esta opción. Para ello, especifique el parámetro `no-apply-only-at-cron-interval` cuando actualice la asociación desde la línea de comandos. Este parámetro obliga a la asociación a ejecutarse inmediatamente después de actualizar la asociación y de acuerdo con el intervalo especificado.

Cómo editar y crear una nueva versión de una asociación

Puede editar una asociación de State Manager para especificar un nombre nuevo, la programación, el nivel de severidad, los destinos u otros valores. En el caso de asociaciones basadas en documentos del tipo de comando SSM, también puede elegir escribir la salida del comando en un bucket de Amazon Simple Storage Service (Amazon S3). Después de editar una asociación, State Manager crea una nueva versión. Puede ver distintas versiones después de la edición, tal como se describe en los siguientes procedimientos.

> **Note**
>
> Para que las asociaciones que se crean con los manuales de procedimientos de Automatización se apliquen cuando se detecten nuevos nodos de destino, se deben cumplir algunas condiciones. Para obtener más información, consulta Acerca de las actualizaciones de destino con los manuales de procedimientos de Automatización.

Los siguientes procedimientos describen cómo editar y crear una nueva versión de una asociación mediante la consola de Systems Manager, AWS Command Line Interface (AWS CLI) y Herramientas de AWS para PowerShell (Tools for PowerShell).

> **Important**
>
> State Manager no admite la ejecución de asociaciones que utilizan una nueva versión de un documento si dicho documento se comparte desde otra cuenta. State Manager ejecuta siempre la versión `default` de un documento si se comparte desde otra cuenta, aunque la consola de Systems Manager muestre que se procesó una versión nueva. Si desea ejecutar una asociación con una versión nueva de un documento compartido desde otra cuenta, debe configurar la versión del documento en `default`.

Cómo crear una asociación (consola)

En el siguiente procedimiento, se describe cómo utilizar la consola de Systems Manager para editar y crear una nueva versión de una asociación.

> **Note**
>
> En el caso de las asociaciones que utilizan documentos de comandos de SSM, no manuales de procedimientos de automatización, este procedimiento requiere que tenga acceso de escritura a un bucket de Amazon S3 existente. Si no ha utilizado Amazon S3, debe tener en cuenta que se le cobrará por utilizar Amazon S3. Para obtener información sobre cómo crear un bucket, consulte Creación de un bucket.

Cómo editar una asociación de State Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija State Manager.
3. Elija una asociación existente y luego elija Editar.
4. Reconfigure la asociación para que cumpla los requisitos actuales.

   Para obtener información sobre las opciones de asociación con los documentos de `Policy` y de `Command`, consulte Cómo crear asociaciones. Para obtener información acerca de las opciones de asociación con manuales de procedimientos de Automation, consulte Programación de automatizaciones con asociaciones de State Manager.
5. Elija Save changes (Guardar cambios).
6. (Opcional) Para ver información sobre asociaciones, en la página Asociaciones, elija el nombre de la asociación que editó y, a continuación, elija la pestaña Versiones. El sistema enumera cada versión de la asociación que ha creado y editado.
7. (Opcional) Para ver los resultados de las asociaciones basadas en documentos de `Command` de SSM, haga lo siguiente:

   a. Abra la consola de Amazon S3 en https://console.aws.amazon.com/s3.

   b. Elija el nombre del bucket de Simple Storage Service (Amazon S3) que especificó para almacenar la información de salida del comando y, a continuación, elija la carpeta denominada con el ID del nodo que ejecutó la asociación. (si eligió almacenar información de salida en una carpeta del bucket, ábrala primero).

   c. Profundice varios niveles, a través de la carpeta `awsrunPowerShell`, hasta el archivo `stdout`.

   d. Elija Open o Download para ver el nombre de host.

Cómo editar una asociación (línea de comandos)

En el siguiente procedimiento se describe cómo utilizar la AWS CLI (en Linux o Windows) o Herramientas de AWS para PowerShell para editar y crear una nueva versión de una asociación.

Cómo editar una asociación de State Manager

1. Si aún no lo ha hecho, instale y configure la AWS CLI o Herramientas de AWS para PowerShell.

Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.

2. Utilice el siguiente formato para crear un comando para editar y crear una nueva versión de una asociación de State Manager existente. Reemplace cada *example resource placeholder* con su propia información.

> **Important**
>
> Cuando llama a `update-association`, el sistema elimina todos los parámetros opcionales de la solicitud y sobrescribe la asociación con valores nulos para esos parámetros. Este comportamiento es así por diseño. Debe especificar todos los parámetros opcionales de la llamada, incluso si no cambia los parámetros. Esto incluye el parámetro `--name`. Antes de llamar a esta acción, le recomendamos que llame a la operación de la API `describe-association` y tome nota de todos los parámetros opcionales necesarios para su llamada a `update-association`.

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
```

En el siguiente ejemplo se actualiza una asociación existente para cambiar el nombre a `TestHostnameAssociation2`. La nueva versión de asociación se ejecuta cada hora y escribe la salida de los comandos en el bucket de Amazon S3 especificado.

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=amzn-s3-demo-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=amzn-s3-demo-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 amzn-s3-demo-bucket `
  -S3Location_OutputS3KeyPrefix logs `
  -S3Location_OutputS3Region us-east-1 `
  -ScheduleExpression "cron(0 */1 * * ? *)"
```

En el siguiente ejemplo se actualiza una asociación existente para cambiar el nombre a `CalendarAssociation`. La nueva asociación se ejecuta cuando el calendario está abierto y escribe la salida del comando en el bucket de Amazon S3 especificado.

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=amzn-s3-demo-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=amzn-s3-demo-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 amzn-s3-demo-bucket `
  -CalendarNames "arn:aws:ssm:us-east-1:123456789012:document/testCalendar2"
```

En el siguiente ejemplo se actualiza una asociación existente para cambiar el nombre a `MultiCalendarAssociation`. La nueva asociación se ejecuta cuando los calendarios están abiertos y escribe la salida del comando en el bucket de Amazon S3 especificado.

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=amzn-s3-demo-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=amzn-s3-demo-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 amzn-s3-demo-bucket `
  -CalendarNames "arn:aws:ssm:us-east-1:123456789012:document/testCalendar1"
 "arn:aws:ssm:us-east-2:123456789012:document/testCalendar2"
```

3. Para ver la nueva versión de la asociación, ejecute el siguiente comando.

   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 *
   ```

   El sistema devuelve información similar a la siguiente.

   Linux & macOS

   ```
   {
       "AssociationDescription": {
           "ScheduleExpression": "cron(0 */1 * * ? *)",
   ```

```
        "OutputLocation": {
            "S3Location": {
                "OutputS3KeyPrefix": "logs",
                "OutputS3BucketName": "amzn-s3-demo-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": "amzn-s3-demo-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}
```

Eliminación de una asociación

Use el siguiente procedimiento para eliminar una asociación usando la consola AWS Systems Manager.

Eliminación de una asociación

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija State Manager.
3. Seleccione una asociación y, a continuación, elija Eliminar.

Puede eliminar varias asociaciones en una sola operación al ejecutar una automatización desde la consola AWS Systems Manager. Al seleccionar varias asociaciones para eliminarlas, State Manager abre la página de inicio del manual de procedimientos de automatización con los ID de asociación introducidos como valores de los parámetros de entrada.

Para eliminar varias asociaciones en una sola operación

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.

2. En el panel de navegación, elija State Manager.
3. Seleccione cada asociación que desee eliminar y elija Eliminar.
4. (Opcional) En el área de Parámetros de entrada adicionales, seleccione el nombre de recurso de Amazon (ARN) para el rol de asunción que desea que utilice la automatización durante la ejecución. Para crear un nuevo rol de asunción, elija Crear.
5. Seleccione Enviar.

Ejecución de grupos de Auto Scaling con asociaciones

La práctica recomendada al momento de utilizar asociaciones para ejecutar grupos de Auto Scaling es utilizar destinos de etiqueta. Si no se utilizan etiquetas, es posible que se alcance el límite de asociación.

Si todos los nodos están etiquetados con la misma clave y valor, solo necesita una asociación para ejecutar el grupo de Auto Scaling. En el siguiente procedimiento, se describe cómo crear dicha asociación.

Para crear una asociación que ejecute grupos de Auto Scaling

1. Asegúrese de que todos los nodos del grupo de Auto Scaling estén etiquetados con la misma clave y valor. Para obtener más instrucciones acerca del etiquetado de nodos, consulte Etiquetado de grupos e instancias de Auto Scaling en la Guía del usuario de AWS Auto Scaling.
2. Cree una asociación siguiendo el procedimiento indicado en Trabajo con asociaciones en Systems Manager.

   Si está trabajando en la consola, elija Specify instance tags (Especificar etiquetas de instancias) en el campo Targets (Destinos). Para Instance tags (Etiquetas de instancia), ingrese la clave y el valor de la Etiqueta de su grupo de Auto Scaling.

   Si utiliza AWS Command Line Interface (AWS CLI), especifique `--targets Key=tag:`*`tag-key`*`,Values=`*`tag-value`* donde la clave y el valor coinciden con lo que etiquetó sus nodos.

Visualización de los historiales de asociación

Puede ver todas las ejecuciones correspondientes a un ID de asociación específico mediante la operación DescribeAssociationExecutions de la API. Utilice esta operación para ver el estado, el estado detallado, los resultados, la hora de la última ejecución y para obtener más información acerca de una asociación de State Manager. State Manager es una herramienta de AWS Systems

Manager. Esta operación de la API también incluye filtros para ayudarlo a encontrar las asociaciones que cumplan los criterios que especifique. Por ejemplo, puede especificar una fecha y hora exactas y utilizar un filtro GREATER_THAN para ver las ejecuciones que se procesaron después dicha fecha y hora.

Si, por ejemplo, se produce un error en la ejecución de una asociación, puede desglosar los detalles de una ejecución específica mediante la operación DescribeAssociationExecutionTargets de la API. Esta operación muestra los recursos, como, por ejemplo, los ID de nodo, en los que se ejecutó la asociación y los diferentes estados de la asociación. A continuación, puede ver en qué recurso o nodo no se pudo ejecutar la asociación. Con el ID de recurso puede ver los detalles de la ejecución del comando para ver en qué paso de un comando se ha producido el error.

En los ejemplos de esta sección, también se incluye información sobre cómo utilizar la operación StartAssociationsOnce de la API para ejecutar una asociación una sola vez al momento de la creación. Puede utilizar esta operación de la API cuando investigue ejecuciones de asociaciones que produzcan errores. Si ve que se ha producido un error en una asociación, puede hacer un cambio en el recurso y, a continuación, ejecutar inmediatamente la asociación para ver si el cambio en el recurso permite que la asociación se ejecute correctamente.

> **Note**
>
> Las operaciones de la API que inicia el documento SSM durante la ejecución de una asociación no se registran en AWS CloudTrail.

Visualización de los historiales de asociación (consola)

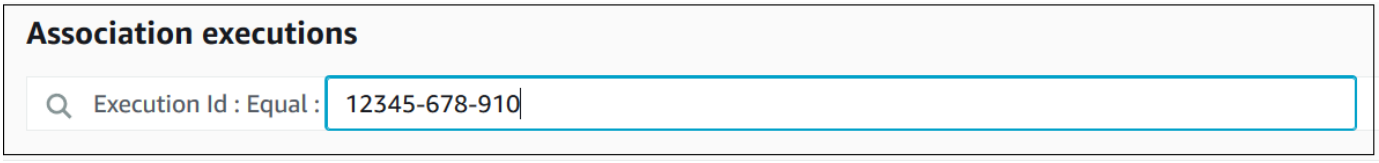
Utilice el siguiente procedimiento para ver el historial de ejecución de un ID de asociación específico y, a continuación, los detalles de ejecución de uno o varios recursos.

Para ver el historial de ejecución de un ID de asociación específico

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. Elija State Manager.
3. En el campo Association id (ID de asociación), elija la asociación cuyo historial desea ver.
4. Elija el botón View details (Ver detalles).
5. Elija la pestaña Execution history (Historial de ejecución).

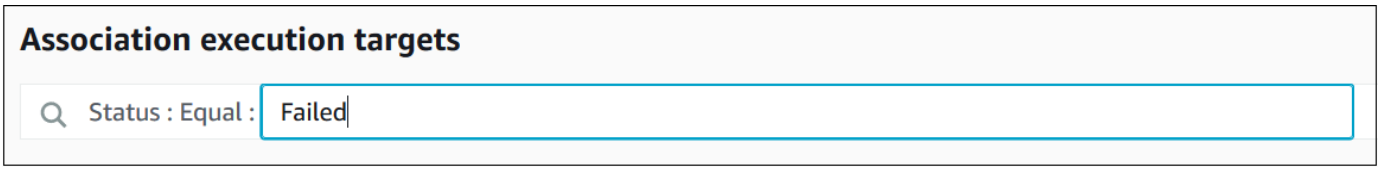
6. Elija la asociación para la que desea ver los detalles de ejecución en el nivel de recursos. Por ejemplo, elija una asociación con el estado Failed (Error). Una vez hecho esto, podrá ver los detalles de ejecución para los nodos en que no se pudo ejecutar la asociación.

   Utilice los filtros del cuadro de búsqueda para localizar la ejecución cuyos detalles desea ver.



7. Elija un ID de ejecución. Se abre la página Association execution targets (Destinos de la ejecución de la asociación). Esta página muestra todos los recursos en que se ha ejecutado la asociación.
8. Elija un ID de recurso para ver información específica sobre dicho recurso.

   Utilice los filtros del cuadro de búsqueda para localizar el recurso cuyos detalles desea ver.



9. Si está investigando una asociación que no se ha podido ejecutar, puede utilizar el botón Apply association now (Aplicar asociación ahora) para ejecutar una asociación una sola vez al momento de la creación. Una vez que haya realizado los cambios en el recurso en que no se pudo ejecutar la asociación, elija el enlace Association ID (ID de asociación) en la ruta de navegación.
10. Haga clic en el botón Apply association now (Aplicar asociación ahora). Cuando finalice la ejecución, verifique que la ejecución de la asociación se ha realizado correctamente.

Visualización de los historiales de asociación (línea de comandos)

En el siguiente procedimiento se describe cómo utilizar la AWS Command Line Interface (AWS CLI) (en Linux o Windows) o Herramientas de AWS para PowerShell para ver el historial de ejecución de un ID de asociación determinado. A continuación, el procedimiento describe cómo ver los detalles de ejecución de uno o varios recursos.

Para ver el historial de ejecución de un ID de asociación específico

1. Si aún no lo ha hecho, instale y configure la AWS CLI o Herramientas de AWS para PowerShell.

Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.

2. Ejecute el siguiente comando para ver una lista de las ejecuciones de un determinado ID de asociación.

   Linux & macOS

   ```
   aws ssm describe-association-executions \
     --association-id ID \
     --filters Key=CreatedTime,Value="2018-04-10T19:15:38.372Z",Type=GREATER_THAN
   ```

   > **Note**
   >
   > Este comando incluye un filtro para limitar los resultados únicamente a las ejecuciones que han tenido lugar después de una fecha y hora determinadas. Si desea ver todas las ejecuciones de un ID de asociación específico, elimine el parámetro `--filters` y el valor `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**
   >
   > Este comando incluye un filtro para limitar los resultados únicamente a las ejecuciones que han tenido lugar después de una fecha y hora determinadas. Si desea ver todas las ejecuciones de un ID de asociación específico, elimine el parámetro `--filters` y el valor `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**
>
> Este comando incluye un filtro para limitar los resultados únicamente a las ejecuciones que han tenido lugar después de una fecha y hora determinadas. Si desea ver todas las ejecuciones de un ID de asociación específico, elimine el parámetro `-Filter` y el valor `@{"Key"="CreatedTime";"Value"="2019-06-01T19:15:38.372Z";"Type"="GREAT`

El sistema devuelve información similar a la siguiente.

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
```

Puede limitar los resultados mediante el uso de uno o varios filtros. En el siguiente ejemplo, se devuelven todas las asociaciones que se ejecutaron antes de una fecha y hora determinada.

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"}
```

El comando siguiente devuelve todas las asociaciones que se ejecutaron correctamente después de una fecha y hora determinadas.

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. Ejecute el comando siguiente para ver todos los destinos a los que se aplicó una ejecución específica.

   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
   ```

   Puede limitar los resultados mediante el uso de uno o varios filtros. En el ejemplo siguiente, se devuelve información sobre todos los destinos en que no se pudo ejecutar la asociación específica.

   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"
    }
```

En el ejemplo siguiente, se devuelve información sobre un nodo administrado específico en que no se pudo ejecutar una asociación.

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 está investigando una asociación que no se ha podido ejecutar, puede utilizar la operación StartAssociationsOnce de la API para ejecutar una asociación inmediatamente una sola vez. Después de cambiar el recurso en el que no se pudo ejecutar la asociación, ejecute el comando siguiente para ejecutar la asociación inmediatamente una sola vez.

   Linux & macOS

   ```
   aws ssm start-associations-once \
     --association-id ID
   ```

   Windows

   ```
   aws ssm start-associations-once ^
     --association-id ID
   ```

   PowerShell

   ```
   Start-SSMAssociationsOnce `
     -AssociationId ID
   ```

Uso de asociaciones mediante IAM

State Manager, una herramienta de AWS Systems Manager, utiliza destinos para elegir con qué instancias configura sus asociaciones. Originalmente, las asociaciones se crearon especificando un nombre de documento (`Name`) y un ID de instancia (`InstanceId`). Esto creó una asociación entre

un documento y una instancia o nodo administrados. Las asociaciones solían ser identificadas por estos parámetros. Estos parámetros ahora están obsoletos, pero siguen siendo compatibles. Los recursos `instance` y `managed-instance` se agregaron como recursos a acciones con `Name` y `InstanceId`.

El comportamiento de aplicación de políticas de AWS Identity and Access Management (IAM) depende del tipo de recurso especificado. Los recursos para operaciones de State Manager solo se aplican en función de la solicitud aprobada. State Manager no realiza una verificación profunda de las propiedades de los recursos de su cuenta. Una solicitud solo se valida con recursos de política si el parámetro de solicitud presenta los recursos de política especificados. Por ejemplo, si especifica una instancia en el bloque de recursos, la política se aplica si la solicitud utiliza el parámetro `InstanceId`. El parámetro `Targets` para cada recurso de la cuenta no está verificado para ese `InstanceId`.

Los siguientes son algunos casos con comportamiento confuso:

- DescribeAssociation, DeleteAssociation y UpdateAssociation utilizan recursos `instance`, `managed-instance` y `document` para especificar la forma obsoleta de referirse a asociaciones. Esto incluye todas las asociaciones creadas con el parámetro obsoleto `InstanceId`.
- CreateAssociation, CreateAssociationBatch y UpdateAssociation utilizan recursos `instance` y `managed-instance` para especificar la forma obsoleta de referirse a asociaciones. Esto incluye todas las asociaciones creadas con el parámetro obsoleto `InstanceId`. El tipo de recurso `document` es parte de la forma obsoleta de referirse a asociaciones y es una propiedad real de una asociación. Esto significa que puede crear políticas de IAM con los permisos `Allow` o `Deny` para las acciones `Create` y `Update` según el nombre del documento.

Para obtener más información acerca del uso de políticas de IAM con Systems Manager, consulte Administración de identidades y accesos en AWS Systems Manager o Acciones, recursos y claves de condiciones de AWS Systems Manager en la Referencia de autorizaciones de servicio.

Cómo crear asociaciones que ejecutan archivos MOF

Puede ejecutar archivos Managed Object Format (MOF) para aplicar un estado deseado en nodos administrados de Windows Server con State Manager, una herramienta de AWS Systems Manager, mediante el documento `AWS-ApplyDSCMofs` de SSM. El documento `AWS-ApplyDSCMofs` cuenta con dos modos de ejecución. En el primer modo, puede configurar la asociación para analizar e informar si los nodos administrados se encuentran en el estado deseado definido en los archivos MOF especificados. En el segundo, puede ejecutar los archivos MOF y cambiar la configuración de

los nodos en función de los recursos y sus valores definidos en los archivos MOF. El documento `AWS-ApplyDSCMofs` le permite descargar y ejecutar los archivos de configuración MOF desde Amazon Simple Storage Service (Amazon S3), un recurso compartido o un sitio web seguro con un dominio HTTPS.

State Manager registra e informa del estado de cada archivo MOF durante cada ejecución de asociación. State Manager también informa de la salida de cada ejecución de archivo MOF como un evento de conformidad que puede ver en la página Conformidad de AWS Systems Manager.

La ejecución de archivos MOF se basa en Windows PowerShell Desired State Configuration (PowerShell DSC). PowerShell DSC es una plataforma declarativa que se utiliza para la configuración, la implementación y la administración de los sistemas Windows. PowerShell DSC permite a los administradores describir, en documentos de texto sencillo llamados "configuraciones de DSC", cómo desean configurar un servidor. Una configuración de PowerShell DSC es un script de PowerShell especializado que indica qué es lo que se debe hacer, pero no cómo. La ejecución de la configuración produce un archivo MOF. El archivo MOF se puede aplicar a uno o varios servidores para lograr la configuración deseada para esos servidores. Los recursos de PowerShell DSC realizan la tarea de aplicar la configuración. Para obtener más información, consulte Windows PowerShell Desired State Configuration Overview.

Temas

- Cómo utilizar Amazon S3 para almacenar artefactos
- Cómo resolver credenciales en archivos MOF
- Cómo utilizar tokens en archivos MOF
- Requisitos previos para la creación de asociaciones que ejecutan archivos MOF
- Cómo crear una asociación que ejecuta archivos MOF
- Solución de problemas al crear asociaciones que ejecutan archivos MOF
- Cómo visualizar los detalles de cumplimiento de recursos de DSC

Cómo utilizar Amazon S3 para almacenar artefactos

Si utiliza Amazon S3 para almacenar los módulos de PowerShell, archivos MOF, informes de conformidad o informes de estado, el rol de AWS Identity and Access Management (IAM) que utiliza SSM Agent de AWS Systems Manager debe tener permisos `ListBucket` y `GetObject` en el bucket. Si no proporciona estos permisos, el sistema devuelve un errorAcceso denegado. Aquí hay información importante sobre el almacenamiento de artefactos en Amazon S3.

- Si el bucket está en otra Cuenta de AWS, cree una política de recursos de bucket que concede a la cuenta (o al rol de IAM) los permisos `GetObject` y `ListBucket`.
- Si desea utilizar recursos de DSC personalizados, puede descargar estos recursos desde un bucket de Amazon S3. También puede instalarlos automáticamente desde la galería de PowerShell.
- Si utiliza Amazon S3 como fuente del módulo, cargue el módulo como un archivo Zip que distingue entre mayúsculas y minúsculas en el siguiente formato: *ModuleName_ModuleVersion*.zip. Por ejemplo: MiMódulo_1.0.0.zip.
- Todos los archivos deben estar en la raíz del bucket. Las estructuras de carpeta no son compatibles.

Cómo resolver credenciales en archivos MOF

Las credenciales se resuelven mediante AWS Secrets Manager o AWS Systems Manager Parameter Store. Esto permite configurar la rotación de credenciales automática. Esto también le permite a DSC propagar automáticamente las credenciales a los servidores sin tener que volver a implementar los archivos MOF.

Para utilizar un secreto de AWS Secrets Manager en una configuración, cree un objeto PSCredential donde el nombre de usuario sean los valores de SecretId o SecretARN del secreto que contiene la credencial. Puede especificar cualquier valor del contraseña. El valor se omite. A continuación se muestra un ejemplo.

```
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
       }
    }
}
```

Compile los archivos usando la configuración MOF PsAllowPlaintextPassword en los datos de configuración. Esto es correcto, ya que la credencial solo contiene una etiqueta.

En Secrets Manager, asegúrese de que el nodo tenga acceso a GetSecretValue en una política administrada de IAM y, de forma opcional, en la política de recursos de secretos si existe. Para trabajar con DSC, el secreto debe tener el siguiente formato.

```
{ 'Username': 'a_name', 'Password': 'a_password' }
```

El secreto puede tener otras propiedades (por ejemplo, las propiedades utilizadas para rotación), pero debe tener, como mínimo, el nombre de usuario y la contraseña.

Es recomendable que utilice un método de rotación de varios usuarios, donde tenga dos nombres de usuario y contraseñas diferentes, y la función de AWS Lambda de rotación cambia entre ellos. Este método permite varias cuentas activas al mismo tiempo, lo que elimina el riesgo de bloquear usuarios durante la rotación.

Cómo utilizar tokens en archivos MOF

Los tokens le ofrecen la posibilidad de modificar valores de propiedades de recursos después de compilar los MOF. Esto le permite volver a utilizar los archivos MOF comunes en varios servidores que requieren configuraciones similares.

La sustitución de tokens solo funciona para las propiedades de recursos de tipo `String`. Sin embargo, si el recurso tiene una propiedad de nodo CIM anidada, también resuelve tokens de propiedades `String` en ese nodo CIM. No se puede utilizar la sustitución de tokens para numerales o matrices.

Por ejemplo, considere un escenario en el que utiliza el recurso xComputerManagement y desea cambiar el nombre del equipo con DSC. Normalmente, necesitaría un archivo MOF dedicado para esa máquina. Sin embargo, gracias a la compatibilidad con token, puede crear un solo archivo MOF y aplicarlo a todos los nodos. En la propiedad `ComputerName`, en lugar de codificar de forma rígida el nombre del equipo en el MOF, puede utilizar un token de tipo de etiqueta de instancia. El valor se resuelve durante el análisis de MOF. Consulte el siguiente ejemplo.

```
Configuration MyConfig
{
    xComputer Computer
    {
        ComputerName = '{tag:ComputerName}'
```

```
    }
}
```

A continuación, configure una etiqueta en el nodo administrado en la consola de Systems Manager o una etiqueta de Amazon Elastic Compute Cloud (Amazon EC2) en la consola de Amazon EC2. Al ejecutar el documento, el script sustituye el token {tag:ComputerName} por el valor de la etiqueta de instancia.

También puede combinar varias etiquetas en una sola propiedad, como se muestra en el ejemplo siguiente.

```
Configuration MyConfig
{
    File MyFile
    {
        DestinationPath = '{env:TMP}\{tag:ComputerName}'
        Type = 'Directory'
    }
}
```

Hay cinco tipos diferentes de tokens que puede utilizar:

- tag: etiquetas de los nodos administrados o Amazon EC2.
- tagb64: igual que tag, pero el sistema usa base64 para descodificar el valor. Esto le permite utilizar caracteres especiales en los valores de etiqueta.
- env: resuelve las variables de entorno.
- ssm: valores de Parameter Store. Solo se admiten los tipos String y SecureString.
- tagssm: igual que tag, pero si la etiqueta no se ha establecido en el nodo, el sistema intenta resolver el valor de un parámetro de Systems Manager con el mismo nombre. Esto resulta útil en situaciones en las que desea un valor global predeterminado, pero quiere poder anularlo en un solo nodo (por ejemplo, las implementaciones únicas).

A continuación, se muestra un ejemplo de Parameter Store que utiliza el tipo de token `ssm`.

```
File MyFile
{
    DestinationPath = "C:\ProgramData\ConnectionData.txt"
    Content = "{ssm:%servicePath%/ConnectionData}"
```

```
}
```

Los tokens desempeñan un papel importante a la hora de reducir el código redundante haciendo que los archivos MOF sean genéricos y reutilizables. Si puede evitar el archivo MOF específico de servidor, no tendrá que usar un servicio de creación de MOF. Un servicio de creación de MOF aumenta los costos, disminuye el tiempo de aprovisionamiento y aumenta el riesgo de desviaciones de configuración entre nodos agrupados debido a que se instalan diferentes versiones del módulo en el servidor de compilación cuando se compilan los MOF.

Requisitos previos para la creación de asociaciones que ejecutan archivos MOF

Antes de crear una asociación que ejecuta archivos MOF, compruebe que los nodos administrados tienen los siguientes requisitos previos instalados:

- Windows PowerShell, versión 5.0 o posterior. Para obtener más información, consulte Requisitos del sistema de Windows PowerShell en Microsoft.com.
- AWS Tools for Windows PowerShell versión 3.3.261.0 o posterior.
- Versión 2.2 o posterior de SSM Agent.

Cómo crear una asociación que ejecuta archivos MOF

Cómo crear una asociación que ejecuta archivos MOF

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija State Manager.
3. Elija State Manager y, a continuación, Create association (Crear asociación).
4. Escriba un nombre en el campo Nombre. Esto es opcional, pero recomendable. Un nombre puede ayudarle a entender el objetivo de la asociación cuando la creó. No se permiten espacios en el nombre.
5. En la lista Document (Documento), elija **`AWS-ApplyDSCMofs`**.
6. En la sección Parameters (Parámetros), especifique los parámetros de entrada opcionales y obligatorios.

    a. Mofs To Apply (MOF a aplicar): especifique uno o varios archivos MOF que se ejecutarán cuando se ejecute esta asociación. Use comas para separar una lista de archivos MOF. Puede especificar las siguientes opciones para encontrar el archivo MOF.

- Un nombre bucket de Amazon S3. Los nombres de bucket deben utilizar letras en minúscula. Especifique esta información mediante el siguiente formato.

  ```
  s3:amzn-s3-demo-bucket:MOF_file_name.mof
  ```

  Si desea especificar una Región de AWS, utilice el siguiente formato.

  ```
  s3:bucket_Region:amzn-s3-demo-bucket:MOF_file_name.mof
  ```

- Un sitio web seguro. Especifique esta información mediante el siguiente formato.

  ```
  https://domain_name/MOF_file_name.mof
  ```

  A continuación se muestra un ejemplo.

  ```
  https://www.example.com/TestMOF.mof
  ```

- Un sistema de archivos en un recurso compartido. Especifique esta información mediante el siguiente formato.

  ```
  \server_name\shared_folder_name\MOF_file_name.mof
  ```

  A continuación se muestra un ejemplo.

  ```
  \StateManagerAssociationsBox\MOFs_folder\MyMof.mof
  ```

b. Service Path (Ruta de servicio): una ruta de servicio es un prefijo de bucket de Amazon S3 en la que desea escribir informes y la información de estado (opcional). Una ruta de servicio es una ruta de las etiquetas basadas en parámetros de Parameter Store. Al resolver etiquetas basadas en parámetros, el sistema utiliza {ssm:%servicePath %/*parameter_name*} para inyectar el valor de servicePath en el nombre del parámetro. Por ejemplo, si la ruta de servicio es "WebServers/Production", los sistemas resuelven el parámetro como: WebServers/Production/*parameter_name*. Esto es útil para cuando se ejecutan varios entornos en la misma cuenta.

c. Report Bucket Name (Nombre del bucket de informes): ingrese el nombre de un bucket de Amazon S3 en el que desea escribir datos de conformidad (opcional). Los informes se guardan en este bucket en formato JSON.

> **Note**
>
> Puede incluir la región en la que se encuentre el bucket como prefijo del nombre del bucket. A continuación, se muestra un ejemplo: us-west-2:MyMOFBucket. Si utiliza un proxy para puntos de enlace de Amazon S3 en una región específica que no incluya us-east-1, incluya como prefijo el nombre del bucket con una región. Si el nombre del bucket no es el prefijo, se detecta automáticamente la región del bucket utilizando el punto de enlace us-east-1.

d. Mof Operation Mode (Modo de funcionamiento de MOF): elija el comportamiento de State Manager cuando se ejecute la asociación **`AWS-ApplyDSCMofs`**:

- Apply (Aplicar): corrige las configuraciones de nodos que no son conformes.
- ReportOnly: no corrige las configuraciones de nodos, sino que registra todos los datos de conformidad e informa de los nodos que no son conformes.

e. Status Bucket Name (Nombre del bucket de estado): ingrese el nombre de un bucket de Amazon S3 en el que desea escribir información de estado de ejecución de MOF (opcional). Estos informes de estado son resúmenes de singleton de la ejecución de conformidad más reciente de un nodo. Esto significa que el informe se sobrescribirá la próxima vez que la asociación ejecute archivos MOF.

> **Note**
>
> Puede incluir la región en la que se encuentre el bucket como prefijo del nombre del bucket. A continuación se muestra un ejemplo: `us-west-2:amzn-s3-demo-bucket` Si utiliza un proxy para puntos de enlace de Amazon S3 en una región específica que no incluya us-east-1, incluya como prefijo el nombre del bucket con una región. Si el nombre del bucket no es el prefijo, se detecta automáticamente la región del bucket utilizando el punto de enlace us-east-1.

f. Module Source Bucket Name (Nombre del bucket de origen del módulo): ingrese el nombre de un bucket de Amazon S3 que contiene archivos de módulos de PowerShell (opcional). Si especifica None (Ninguno), elija True (Verdadero) para la siguiente opción, Allow PS Gallery Module Source (Permitir el origen de módulos de la galería de PowerShell).

> **Note**
>
> Puede incluir la región en la que se encuentre el bucket como prefijo del nombre del bucket. A continuación se muestra un ejemplo: `us-west-2:amzn-s3-demo-bucket` Si utiliza un proxy para puntos de enlace de Amazon S3 en una región específica que no incluya us-east-1, incluya como prefijo el nombre del bucket con una región. Si el nombre del bucket no es el prefijo, se detecta automáticamente la región del bucket utilizando el punto de enlace us-east-1.

g. Allow PS Gallery Module Source (Permitir origen de módulos de la galería de PowerShell): elija Verdadero para descargar los módulos de PowerShell de https://www.powershellgallery.com/ (opcional). Si elige False (Falso), especifique un origen para la opción anterior, ModuleSourceBucketName.

h. Proxy Uri (URI de proxy): utilice esta opción para descargar archivos MOF desde un servidor proxy (opcional).

i. Reboot Behavior (Comportamiento de reinicio): especifique uno de los siguientes comportamientos de reinicio si la ejecución de su archivo MOF requiere reiniciar: (opcional)

- AfterMof: reinicia el nodo una vez que se hayan completado todas las ejecuciones de MOF. Aunque varias ejecuciones de MOF solicitan reinicios, el sistema espera hasta que se hayan completado todas las ejecuciones de MOF para reiniciarse.
- Immediately (Inmediatamente): reinicia el nodo cada vez que lo solicita una ejecución de MOF. Si se ejecutan varios archivos MOF que solicitan reinicios, los nodos se reinician varias veces.
- Never (Nunca): los nodos no se reinician, incluso si la ejecución de MOF solicita expresamente un reinicio.

j. Use Computer Name For Reporting (Utilizar nombre de equipo para informes): habilite esta opción para utilizar el nombre del equipo cuando se crean informes de información de conformidad (opcional). El valor predeterminado es false (falso), lo que significa que el sistema utiliza el ID de nodo cuando se crean informes de conformidad.

k. Turn on Verbose Logging (Activar registros detallados): le recomendamos activar el registro detallado a la hora de implementar archivos MOF por primera vez (opcional).

> **Important**
>
> Cuando se permite, el registro detallado escribe más datos a su bucket de Amazon S3 que con el registro de ejecución de asociaciones estándar. Esto puede reducir el rendimiento y conllevar cargos de almacenamiento superiores en Amazon S3. Para mitigar los problemas de tamaños de almacenamiento, le recomendamos que active las políticas de ciclo de vida en su bucket de Amazon S3. Para obtener más información, consulte How Do I Create a Lifecycle Policy for an S3 Bucket? en la Guía del usuario de Amazon Simple Storage Service.

l. Turn on Debug Logging (Activar registro de depuración): le recomendamos que active el registro de depuración para solucionar los problemas de errores de MOF (opcional). También le recomendamos que desactive esta opción para hacer un uso normal.

> **Important**
>
> Cuando se permite, el registro detallado escribe más datos en su bucket de Amazon S3 que con el registro de ejecución de asociaciones estándar. Esto puede reducir el rendimiento y conllevar cargos de almacenamiento superiores en Amazon S3. Para mitigar los problemas de tamaños de almacenamiento, le recomendamos que active las políticas de ciclo de vida en su bucket de Amazon S3. Para obtener más información, consulte How Do I Create a Lifecycle Policy for an S3 Bucket? en la Guía del usuario de Amazon Simple Storage Service.

m. Compliance Type (Tipo de conformidad): especifique el tipo de conformidad que se utilizará al crear informes de información de conformidad (opcional). El valor de conformidad predeterminado es Custom:DSC. Si crea varias asociaciones que ejecutan archivos MOF, asegúrese de especificar un tipo de conformidad distinto para cada asociación. Si no lo hace, cada asociación adicional que utilice Custom:DSC sobrescribe los datos de conformidad existentes.

n. Pre Reboot Script (Script de reinicio previo): especifique un script a ejecutar si la configuración indica que es necesario reiniciar. El script se ejecuta antes del reinicio. El script debe ser de una sola línea. Separe las líneas adicionales mediante el uso de punto y coma.

7. En la sección Targets (Destinos), elija Specifying tags (Especificación de etiquetas) o Manually Selecting Instance (Selección manual de la instancia). Si decide seleccionar como destino

recursos mediante el uso de etiquetas, introduzca una clave de etiqueta y un valor de etiqueta en los campos correspondientes. Para obtener más información acerca del uso de destinos, consulte Cómo comprender los controles de frecuencia y destinos en las asociaciones de State Manager.

8. En la sección Specify schedule (Especificar programación), elija On Schedule (De forma programada) o No schedule (Sin programación). Si elige On Schedule (De forma programada), utilice los botones proporcionados para crear una programación Cron o Rate para la asociación.
9. En la sección Advanced options (Opciones avanzadas):

   - En Compliance severity (Gravedad de conformidad), elija un nivel de seguridad para la asociación. Los informes de conformidad indican si el estado de asociación es conforme o no conforme, junto con el nivel de gravedad que se indique aquí. Para obtener más información, consulte Acerca de la conformidad de las asociaciones de State Manager.
10. En la sección Rate control (Control de frecuencia), configure las opciones para ejecutar asociaciones de State Manager a través de la flota de nodos administrados. Para obtener más información sobre estas opciones, consulte Cómo comprender los controles de frecuencia y destinos en las asociaciones de State Manager.

    En la sección Simultaneidad, elija una opción:

    - Elija Targets (Destinos) para introducir un número absoluto de destinos que pueda ejecutar la asociación de forma simultánea.
    - Elija porcentaje para introducir un porcentaje del destino definido que puede ejecutar la asociación de forma simultánea.

    En la sección Umbral de error, elija una opción:

    - Elija errors (errores) para introducir un número absoluto de errores permitidos antes de que State Manager deje de ejecutar asociaciones en más destinos.
    - Elija percentage (porcentaje) para introducir un porcentaje de errores permitidos antes de que State Manager deje de ejecutar asociaciones en más destinos.
11. (Opcional) En Output options (Opciones de salida), para guardar la salida del comando en un archivo, seleccione el cuadro Enable writing output to S3 (Permitir la escritura de salida en S3). Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancias asignado al nodo administrado, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, verifique que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

12. Elija Crear asociación.

State Manager crea y ejecuta inmediatamente la asociación en los nodos o destinos especificados. Después de la ejecución inicial, la asociación se ejecuta en intervalos de acuerdo con la programación que definió y de acuerdo con las reglas siguientes:

- State Manager ejecuta asociaciones en nodos que estén en línea cuando el intervalo comienza y omite los nodos sin conexión.
- State Manager intenta ejecutar la asociación en todos los nodos configurados durante un intervalo.
- Si una asociación no se ejecuta durante un intervalo (porque, por ejemplo, un valor de simultaneidad limita el número de nodos que podría procesar la asociación simultáneamente), State Manager intenta ejecutar la asociación durante el intervalo siguiente.
- State Manager registra el historial de todos los intervalos omitidos. Puede ver el historial en la pestaña Execution History (Historial de ejecución).

> **Note**
>
> `AWS-ApplyDSCMofs` es un documento de Systems Manager Command. Esto significa que también puede ejecutar este documento mediante Run Command, una herramienta de AWS Systems Manager. Para obtener más información, consulte AWS Systems Manager Run Command.

Solución de problemas al crear asociaciones que ejecutan archivos MOF

Esta sección contiene información para ayudarle a solucionar los problemas que surjan al crear asociaciones que ejecutan archivos MOF.

Cómo activar el registro mejorado

Como primer paso para la solución de problemas, active el registro mejorado. Más concretamente, realice lo siguiente:

1. Compruebe que la asociación se ha configurado para escribir el resultado del comando en Amazon S3 o los Registros de Amazon CloudWatch (CloudWatch).
2. Establezca el parámetro Enable Verbose Logging (Habilitar registro detallado) en True.
3. Establezca el parámetro Enable Debug Logging (Habilitar registro de depuración) en True.

Con el registro de depuración y el detallado activados, el archivo de salida Stdout incluye información detallada acerca de la ejecución de scripts. Este archivo de salida puede ayudarle a identificar donde el script ha producido un error. El archivo de Stderr contiene errores que se produjeron durante la ejecución de scripts.

Problemas comunes al crear asociaciones que ejecutan archivos MOF

Esta sección contiene información sobre problemas comunes que pueden ocurrir al crear asociaciones que ejecutan archivos MOF y los pasos que hay que seguir para solucionar dichos problemas.

Mi MOF no se aplicó.

Si State Manager no pudo aplicar la asociación a los nodos, comience a revisar el archivo de salida Stderr. Este archivo puede ayudarle a entender la causa raíz del problema. Además, compruebe lo siguiente:

- El nodo tiene los permisos de acceso requeridos a todos los buckets de Simple Storage Service (Amazon S3) relacionados con MOF. En concreto:
  - Permisos s3:GetObject: necesarios para los archivos MOF de buckets de Amazon S3 privados, así como módulos personalizados en buckets de Amazon S3.
  - Permiso s3:PutObject: necesario para escribir los informes de conformidad y el estado de conformidad en los buckets de Amazon S3.

- Si utiliza etiquetas, asegúrese de que el nodo tenga la política de IAM necesaria. Para usar las etiquetas, el rol de IAM debe tener una política que permita las acciones `ec2:DescribeInstances` y `ssm:ListTagsForResource`.
- Asegúrese de que el nodo tiene las etiquetas esperadas o los parámetros de SSM asignados.
- Asegúrese de que las etiquetas o los parámetros de SSM no se hayan escrito erróneamente.
- Pruebe a aplicar el archivo MOF localmente en el nodo para asegurarse de que no hay un problema con el archivo MOF.

Parece que mi MOF tuvo errores, pero Systems Manager se ejecutó correctamente.

Si el documento `AWS-ApplyDSCMofs` se ha ejecutado correctamente, el estado de ejecución de Systems Manager muestra Success (Correcto). Este estado no refleja el estado de conformidad del nodo en función de los requisitos de configuración del archivo MOF. Para ver el estado de conformidad de los nodos, consulte los informes de conformidad. Puede ver un informe JSON en el bucket de informes de Amazon S3. Esto se aplica tanto a las ejecuciones de Run Command como a las de State Manager. Además, para State Manager, puede ver detalles de conformidad en la página de conformidad de Systems Manager.

Estados de Stderr: error de resolución de nombres tras intentar conectar con el servicio

Este error indica que el script no puede tener acceso a un servicio remoto. Lo más probable es que el script no pueda conectar con Amazon S3. Este problema se produce en la mayoría de los casos cuando el script intenta escribir informes de conformidad o estados de conformidad en el bucket de Amazon S3 facilitado en los parámetros de documentos. Normalmente, este error se produce cuando un entorno informático utiliza un firewall o un proxy transparente que incluya una lista de permitidos. Para resolver este problema, siga estos pasos:

- Utilice la sintaxis de buckets específica de la región para todos los parámetros de buckets de Amazon S3. Por ejemplo, el parámetro Mofs to Apply (MOF para aplicar) debe tener el siguiente formato:

  s3:*región-bucket*:*amzn-s3-demo-bucket*:*nombre-archivo-mof*.mof.

  A continuación se muestra un ejemplo: `s3:us-west-2:amzn-s3-demo-bucket:my-mof.mof`

  Los nombres de buckets de origen de módulos, estado e informes deben tener el siguiente formato.

región-bucket:*amzn-s3-demo-bucket*. A continuación se muestra un ejemplo: `us-west-1:amzn-s3-demo-bucket;`

- Si la sintaxis específica de la región no corrige el problema, asegúrese de que los nodos de destino pueden obtener acceso a Simple Storage Service (Amazon S3) en la región deseada. Para comprobar esto:

  1. Busque el nombre del punto de enlace para Amazon S3 en la región de Amazon S3 adecuada. Para obtener más información, consulte Puntos de conexión de Amazon S3 Service en la Referencia general de Amazon Web Services.
  2. Inicie sesión en el nodo de destino y ejecute el siguiente comando de ping.

     ```
     ping s3.s3-region.amazonaws.com
     ```

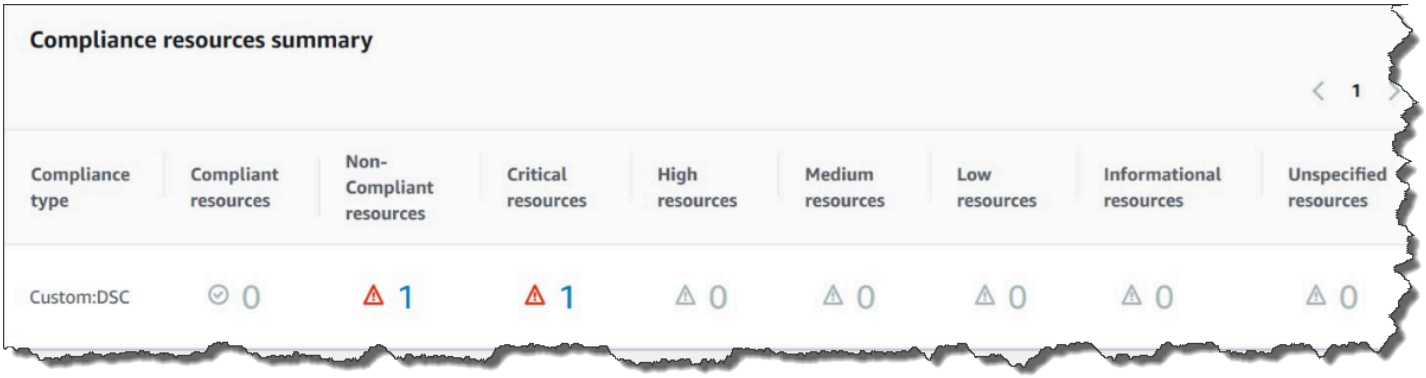
     Si el ping no se ejecuta correctamente, significa que Simple Storage Service (Amazon S3) está fuera de servicio, que un firewall/proxy transparente está bloqueando el acceso a la región de Amazon S3 o que el nodo no tiene acceso a Internet.

Cómo visualizar los detalles de cumplimiento de recursos de DSC

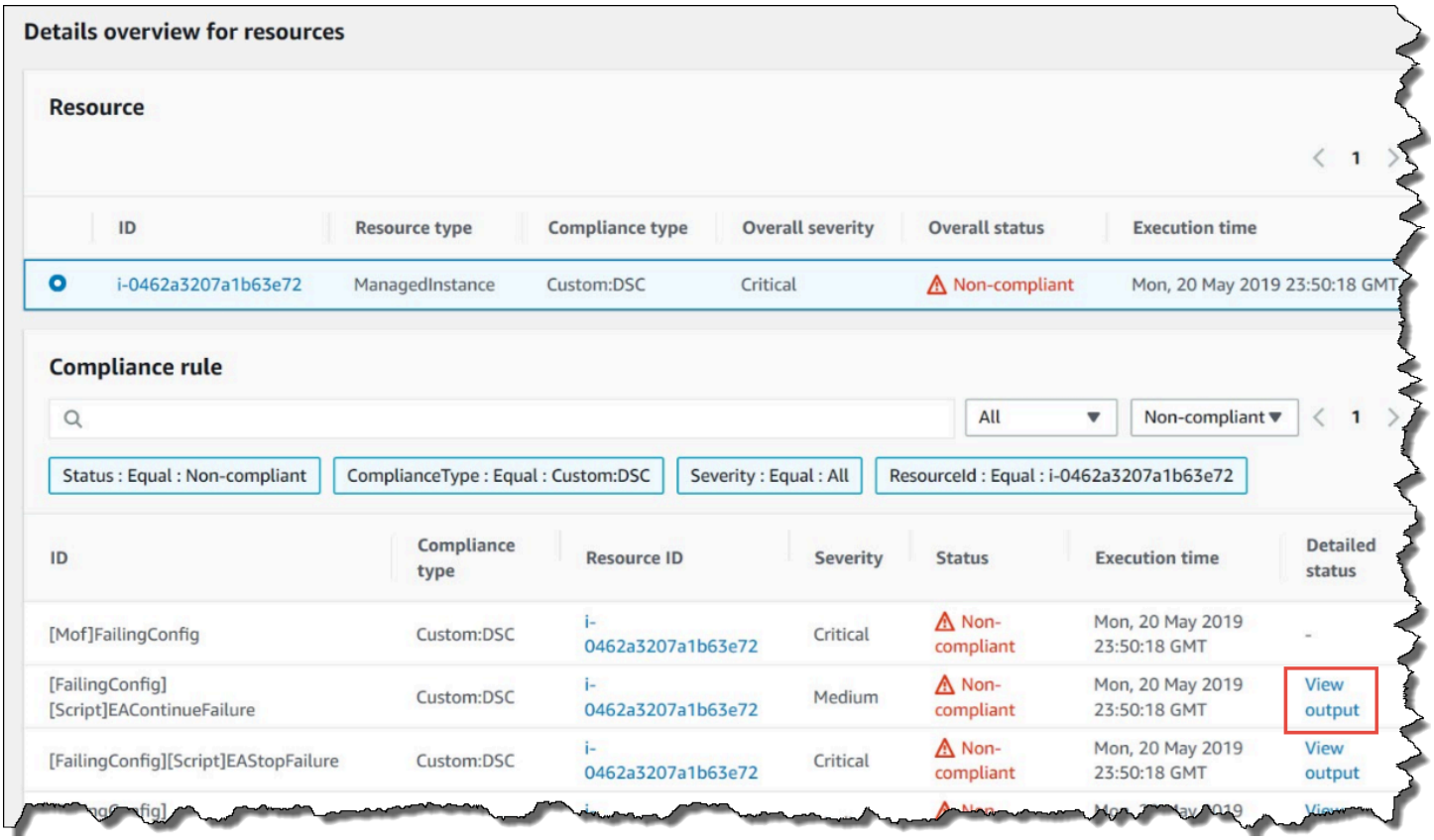
Systems Manager capta información de conformidad de los errores de los recursos de DSC en el bucket de estado de Amazon S3 Status Bucket que especificó cuando ejecutó el documento `AWS-ApplyDSCMofs`. La búsqueda de información acerca de los errores del recurso de DSC en un bucket de Amazon S3 puede llevar tiempo. En su lugar, puede ver esta información en la página Compliance (Conformidad) de Systems Manager.

La sección Compliance resources summary (Resumen de los recursos de cumplimiento muestra una cuenta de los recursos que fallaron. En el siguiente ejemplo, el ComplianceType (Tipo de cumplimiento) es Custom:DSC (Personalizado:DSC) y un recurso no conforme.

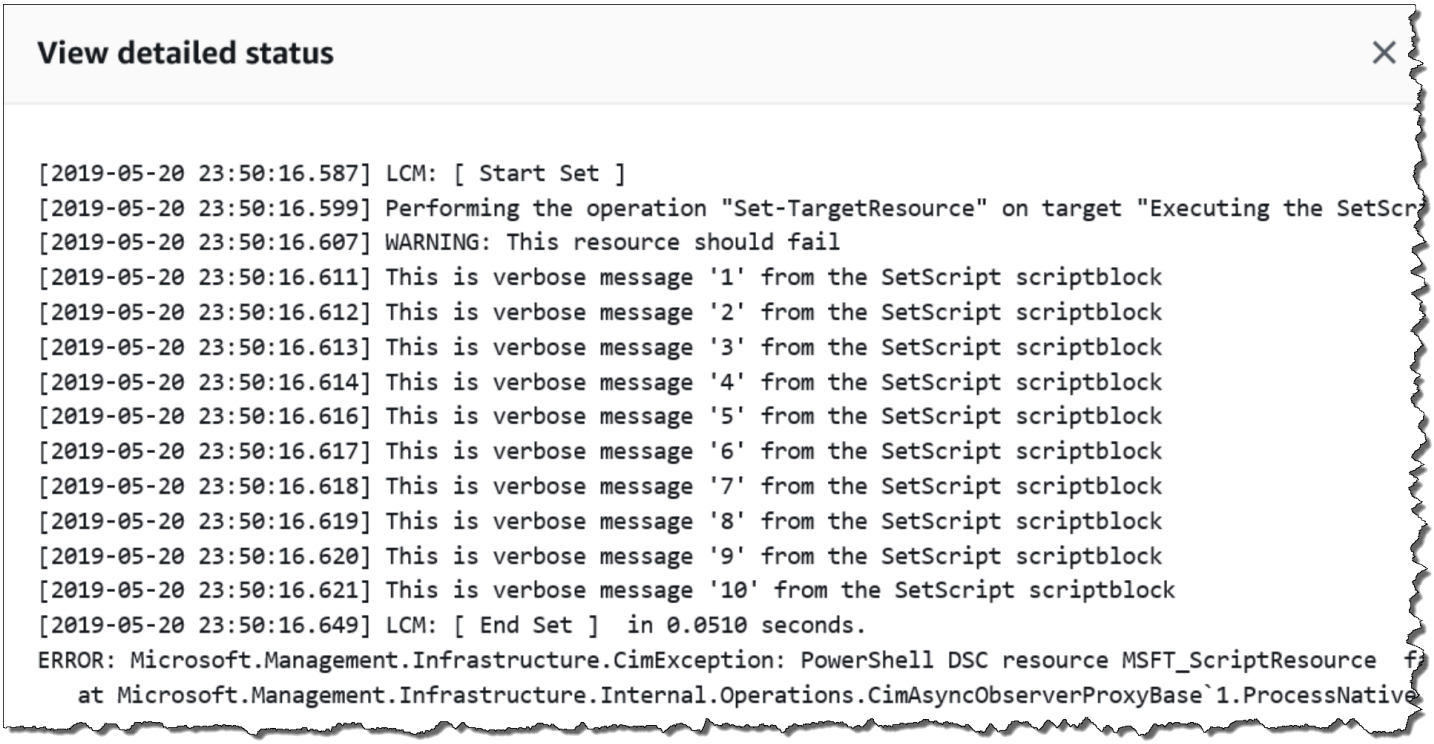
> **Note**
>
> Custom:DSC es el valor predeterminado ComplianceType en el documento `AWS-ApplyDSCMofs`. Este valor se puede personalizar.



La sección Details overview for resources (Información general de los detalles de los recursos) muestra información sobre el recurso de AWS con el recurso de DSC no conforme. En esta sección también se incluye el nombre de MOF, los pasos de ejecución del script y, si procede, un enlace de View output (Ver salida) para ver la información del estado detallada.



El enlace View output (Ver salida) muestra los últimos 4000 caracteres del estado detallado. Systems Manager comienza con la excepción como primer elemento y, a continuación, vuelve a examinar los mensajes detallados y agrega tantos como pueda hasta que alcanza la cuota de 4000 caracteres. Este proceso muestra los mensajes de registros que salieron antes de lanzar la excepción. Estos mensajes son los más importantes para la resolución de errores.



Para obtener más información acerca de cómo ver la información de cumplimiento, consulte Conformidad de AWS Systems Manager.

Situaciones que afectan a los informes del cumplimiento

Si la asociación de State Manager falla, no se notifican datos de cumplimiento. En concreto, si un MOF falla en el procesamiento, Systems Manager no notifica ningún elemento de conformidad ya que las asociaciones fallan. Por ejemplo, si Systems Manager intenta descargar un MOF de un bucket de Simple Storage Service (Amazon S3) para el que el nodo no tiene permiso de acceso, la asociación falla y no se notifican datos de conformidad.

Si un recurso en un segundo MOF falla, Systems Manager sí notifica los datos de conformidad. Por ejemplo, si un MOF intenta crear un archivo en una unidad que no existe, Systems Manager notifica la conformidad porque el documento de `AWS-ApplyDSCMofs` se puede procesar completamente, lo que significa que la asociación funciona correctamente.

Creación de asociaciones que ejecuten manuales de estrategia de Ansible

Puede crear asociaciones de State Manager que ejecuten manuales de estrategia de Ansible mediante el documento `AWS-ApplyAnsiblePlaybooks` de SSM. State Manager es una herramienta de AWS Systems Manager. Este documento ofrece los siguientes beneficios para ejecutar cuadernos de trabajo:

- Compatibilidad con la ejecución de cuadernos de trabajo complejos
- Soporte para descargar manuales de estrategias de GitHub y Amazon Simple Storage Service (Amazon S3)
- Compatibilidad con la estructura de cuaderno de trabajo comprimido
- Registro optimizado
- Capacidad para especificar qué cuaderno de trabajo ejecutar cuando se empaquetan los cuadernos de trabajo

> **Note**
>
> Systems Manager incluye dos documentos de SSM que le permiten crear asociaciones State Manager que ejecutan manuales de estrategia de Ansible: `AWS-RunAnsiblePlaybook` y `AWS-ApplyAnsiblePlaybooks`. El documento `AWS-RunAnsiblePlaybook` está obsoleto. Sigue estando disponible en Systems Manager para fines heredados. Le recomendamos que utilice el documento `AWS-ApplyAnsiblePlaybooks` debido a las mejoras que se describen aquí.
>
> Las asociaciones que ejecutan manuales de estrategia de Ansible no son compatibles con macOS.

Compatibilidad con la ejecución de cuadernos de trabajo complejos

El documento `AWS-ApplyAnsiblePlaybooks` admite cuadernos de trabajo complejos agrupados, ya que copia toda la estructura de archivos en un directorio local antes de ejecutar el cuaderno de trabajo principal especificado. Puede proporcionar libros de trabajo de origen en archivos Zip o en una estructura de directorios. El archivo Zip o directorio se pueden almacenar en GitHub o Amazon S3.

Compatibilidad con la descarga de cuadernos de trabajo desde GitHub

El documento `AWS-ApplyAnsiblePlaybooks` utiliza el complemento `aws:downloadContent` para descargar archivos del cuaderno de trabajo. Los archivos se pueden almacenar en GitHub un único archivo o como un conjunto combinado de archivos de manual de estrategias. Para descargar contenido desde GitHub, especifique información sobre el repositorio de GitHub en formato JSON. A continuación se muestra un ejemplo.

```
{
```

```
    "owner":"TestUser",
    "repository":"GitHubTest",
    "path":"scripts/python/test-script",
    "getOptions":"branch:master",
    "tokenInfo":"{{ssm-secure:secure-string-token}}"
}
```

Compatibilidad para descargar cuadernos de trabajo desde Simple Storage Service (Amazon S3)

También puede almacenar y descargar manuales de estrategias de Ansible en Amazon S3 como un único archivo .zip o una estructura de directorios. Para descargar contenido desde Amazon S3, especifique la ruta al archivo. A continuación se incluyen dos ejemplos.

Ejemplo 1: descargar un archivo de cuaderno de trabajo específico

```
{
    "path":"https://s3.amazonaws.com/amzn-s3-demo-bucket/playbook.yml"
}
```

Ejemplo 2: descargar el contenido de un directorio

```
{
    "path":"https://s3.amazonaws.com/amzn-s3-demo-bucket/ansible/webservers/"
}
```

> **Important**
>
> Si especifica Amazon S3, el perfil de instancia de AWS Identity and Access Management (IAM) de los nodos administrados debe incluir permisos para el bucket de S3. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager.

Compatibilidad con la estructura de cuaderno de trabajo comprimido

El documento `AWS-ApplyAnsiblePlaybooks` le permite ejecutar archivos .zip comprimidos en el paquete descargado. El documento comprueba si los archivos descargados contienen un archivo comprimido en formato .zip. Si se encuentra un archivo .zip, el documento descomprime automáticamente el archivo y, a continuación, ejecuta la automatización de Ansible especificada.

Registro optimizado

El documento `AWS-ApplyAnsiblePlaybooks` incluye un parámetro opcional para especificar distintos niveles de registro. Especifique -v para un nivel de detalle bajo, -vv o -vvv para un nivel de detalle medio y -vvvv para el registro de nivel de depuración. Estas opciones se asignan directamente a las opciones de detalle de Ansible.

Capacidad para especificar qué cuaderno de trabajo ejecutar cuando se empaquetan los cuadernos de trabajo

El documento `AWS-ApplyAnsiblePlaybooks` incluye un parámetro obligatorio para especificar qué cuaderno de trabajo ejecutar cuando se agrupan varios cuadernos de trabajo. Esta opción proporciona flexibilidad para ejecutar cuadernos de trabajo para admitir diferentes casos de uso.

Comprensión de las dependencias instaladas

Si especifica True (Verdadero) para el parámetro InstallDependencies, Systems Manager verifica que los nodos tengan las siguientes dependencias instaladas:

- Ubuntu Server/Debian Server: Apt-get (administración de paquetes), Python 3, Ansible, Unzip
- Amazon Linux: Ansible
- RHEL: Python 3, Ansible, Unzip

Si no se encuentra una o varias de estas dependencias, Systems Manager las instala automáticamente.

Crear una asociación que ejecute manuales de estrategias de Ansible (consola)

En el siguiente procedimiento se describe cómo utilizar la consola de Systems Manager para crear una asociación de State Manager que ejecuta manuales de estrategias de Ansible mediante el documento de `AWS-ApplyAnsiblePlaybooks`.

Crear una asociación que ejecute manuales de estrategias de Ansible (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija State Manager.
3. Elija State Manager y, a continuación, Create association (Crear asociación).
4. En Name (Nombre), especifique un nombre que le ayude a recordar el objetivo de la asociación.
5. En la lista Document (Documento), elija **`AWS-ApplyAnsiblePlaybooks`**.

6. En la sección Parameters (Parámetros), en Source Type (Tipo de origen), elija GitHub o S3.

   GitHub

   Si elige GitHub, ingrese la información del repositorio en el siguiente formato.

   ```
   {
      "owner":"user_name",
      "repository":"name",
      "path":"path_to_directory_or_playbook_to_download",
      "getOptions":"branch:branch_name",
      "tokenInfo":"{{(Optional)_token_information}}"
   }
   ```

   S3

   Si elige S3, ingrese la información de la ruta en el siguiente formato.

   ```
   {
      "path":"https://s3.amazonaws.com/path_to_directory_or_playbook_to_download"
   }
   ```

7. En Install Dependencies (Instalar dependencias), elija una opción.
8. (Opcional) En Playbook File (Archivo de cuaderno de trabajo), escriba un nombre de archivo. Si el cuaderno de trabajo está contenido en un archivo Zip, especifique una ruta relativa al archivo Zip.
9. (Opcional) En Variables adicionales, escriba las variables que desee que State Manager envíe a Ansible en el tiempo de ejecución.
10. (Opcional) En Check (Comprobar),elija una opción.
11. (Opcional) En Verbose (Detalle), elija una opción.
12. En Targets (Destinos), elija una opción. Para obtener más información acerca del uso de los destinos, consulte Cómo comprender los controles de frecuencia y destinos en las asociaciones de State Manager.
13. En la sección Specify schedule (Especificar programación), elija On Schedule (De forma programada) o No schedule (Sin programación). Si elige On Schedule (De forma programada), utilice los botones proporcionados para crear una programación Cron o Rate para la asociación.
14. En la sección Opciones avanzadas, en Gravedad de conformidad, elija un nivel de gravedad para la asociación. Los informes de conformidad indican si el estado de asociación es

conforme o no conforme, junto con el nivel de gravedad que se indique aquí. Para obtener más información, consulte Acerca de la conformidad de las asociaciones de State Manager.

15. En la sección Rate control (Control de frecuencia), configure las opciones para ejecutar asociaciones de State Manager a través de una flota de nodos administrados. Para obtener más información sobre el uso de controles de frecuencia, consulte Cómo comprender los controles de frecuencia y destinos en las asociaciones de State Manager.

    En la sección Simultaneidad, elija una opción:

    - Elija Targets (Destinos) para introducir un número absoluto de destinos que pueda ejecutar la asociación de forma simultánea.
    - Elija porcentaje para introducir un porcentaje del destino definido que puede ejecutar la asociación de forma simultánea.

    En la sección Umbral de error, elija una opción:

    - Elija errors (errores) para especificar un número absoluto de errores permitidos antes de que State Manager deje de ejecutar asociaciones en más destinos.
    - Elija percentage (porcentaje) para especificar un porcentaje de errores permitidos antes de que State Manager deje de ejecutar asociaciones en más destinos.

16. (Opcional) En Output options (Opciones de salida), para guardar la salida del comando en un archivo, seleccione el cuadro Enable writing output to S3 (Permitir la escritura de salida en S3). Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

    > **Note**
    >
    > Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancias asignado al nodo administrado, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, verifique que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

17. Elija Crear asociación.

> **Note**
>
> Si utiliza las etiquetas para crear una asociación en uno o varios nodos de destino y después elimina las etiquetas de un nodo, dicho nodo ya no ejecutará la asociación. El nodo se desvincula del documento de State Manager.

Crear una asociación que ejecute manuales de estrategias de Ansible (CLI)

En el siguiente procedimiento se describe cómo utilizar AWS Command Line Interface (AWS CLI) para crear una asociación de State Manager que ejecute manuales de estrategias de Ansible mediante el documento `AWS-ApplyAnsiblePlaybooks`.

Crear una asociación que ejecute manuales de estrategias de Ansible (CLI)

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Ejecute uno de los siguientes comandos para crear una asociación que ejecute manuales de estrategias de Ansible dirigiendo nodos mediante etiquetas. Reemplace cada *example resource placeholder* con su propia información. El comando (A) especifica GitHub como tipo de origen. El comando (B) especifica Amazon S3 como tipo de origen.

   (A) Origen de GitHub

   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"
```

A continuación se muestra un ejemplo.

```
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 *)"
```

(B) Origen de S3

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"
```

A continuación se muestra un ejemplo.

```
aws ssm create-association --name "AWS-ApplyAnsiblePlaybooks" \
    --targets "Key=tag:OS,Values=Linux" \
    --parameters '{"SourceType":["S3"],"SourceInfo":["{\"path\":\"https://
s3.amazonaws.com/amzn-s3-demo-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**
>
> Las asociaciones de State Manager no admiten todas las expresiones cron y de frecuencia. Para obtener más información acerca de la creación de expresiones cron y de frecuencia para asociaciones, consulte Referencia: expresiones cron y rate para Systems Manager.

El sistema intenta crear la asociación en los nodos y aplicar inmediatamente el estado.

3. Ejecute el siguiente comando para ver un estado actualizado de la asociación que acaba de crear.

```
aws ssm describe-association --association-id "ID"
```

Creación de asociaciones que ejecuten recetas de Chef

Puede crear asociaciones de State Manager que ejecuten recetas de Chef con el documento `AWS-ApplyChefRecipes` de SSM. State Manager es una herramienta de AWS Systems Manager. Puede dirigirse a nodos administrados por Systems Manager basados en Linux con el documento `AWS-ApplyChefRecipes` de SSM. Este documento ofrece los siguientes beneficios para ejecutar recetas de Chef:

- Admite múltiples versiones de Chef (de Chef 11 a Chef 18).
- Instala automáticamente el software cliente de Chef en los nodos de destino.
- Opcionalmente, ejecuta comprobaciones de conformidad de Systems Manager en nodos de destino y almacena los resultados de las comprobaciones de conformidad en un bucket de Amazon Simple Storage Service (Amazon S3).
- Ejecuta varios libros de recetas y recetas en una sola ejecución del documento.
- Opcionalmente, ejecuta recetas en modo `why-run`, para mostrar qué recetas cambian en nodos de destino sin realizar cambios.
- Opcionalmente aplica atributos JSON personalizados a las ejecuciones de `chef-client`.
- De forma opcional, aplica atributos JSON personalizados desde un archivo fuente que se almacena en la ubicación que especifique.

Puede utilizar buckets de Git, GitHub, HTTP o Amazon S3 como fuentes para descargar los libros de recetas y recetas de Chef que especifique en un documento `AWS-ApplyChefRecipes`.

> **Note**
>
> Las asociaciones que ejecutan recetas de Chef no son compatibles con macOS.

Introducción

Antes de crear un documento `AWS-ApplyChefRecipes`, prepare sus libros de recetas de Chef y su repositorio de libros de recetas. Si aún no tiene un libro de recetas de Chef que desea utilizar, puede comenzar usando un libro de recetas `HelloWorld` de prueba que AWS ha preparado para usted. El documento `AWS-ApplyChefRecipes` ya apunta a este libro de recetas de forma predeterminada. Sus libros de recetas deben configurarse de forma similar a la siguiente estructura de directorios. En el siguiente ejemplo, `jenkins` y `nginx` son ejemplos de libros de recetas de Chef que están disponibles en Chef Supermarket en el sitio web de Chef.

Aunque AWS no puede admitir oficialmente libros de recetas en el sitio web de Chef Supermarket muchos de ellos trabajan con el documento `AWS-ApplyChefRecipes`. Los siguientes son ejemplos de criterios para determinar cuándo se está probando un libro de recetas de la comunidad:

- El libro de recetas debe admitir los sistemas operativos basados en Linux de los nodos Systems Manager administrados a los que se dirige.
- El libro de recetas debe ser válido para la versión cliente de Chef (de Chef 11 a Chef 18) que utilice.
- El libro de recetas es compatible con Chef Infra Client y no requiere un servidor Chef.

Compruebe que puede comunicarse con el sitio web de `Chef.io` para que los libros de recetas que especifique en la lista de ejecución se puedan instalar cuando se ejecute el documento de Systems Manager (documento de SSM). Se admite el uso de una carpeta `cookbooks`, pero no es necesario; puede almacenar libros de recetas directamente bajo el nivel raíz.

```
<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

Antes de crear una asociación State Manager que ejecute recetas de Chef, tenga en cuenta que la ejecución del documento instala el software cliente de Chef en los nodos administrados por Systems Manager, a menos que establezca el valor de la versión de cliente de Chef a `None`. Esta operación utiliza un script de instalación de Chef para instalar componentes de Chef en su nombre. Antes de ejecutar un documento `AWS-ApplyChefRecipes`, asegúrese de que su empresa pueda cumplir con los requisitos legales aplicables, incluidos los términos de licencia aplicables al uso del software Chef. Para obtener más información, consulte el sitio web de Chef.

Systems Manager puede entregar informes de conformidad a un bucket de S3, a la consola de Systems Manager o hacer que los resultados de conformidad estén disponibles en respuesta

a los comandos de la API de Systems Manager. Para ejecutar informes de conformidad de Systems Manager, el perfil de instancias adjuntado a nodos administrados por Systems Manager debe tener permisos para escribir en el bucket de S3. El perfil de instancias debe tener permisos para poder utilizar la API `PutComplianceItem` de Systems Manager. Para obtener más información acerca de la conformidad de Systems Manager, consulte Conformidad de AWS Systems Manager.

Registro de la ejecución del documento

Cuando ejecuta un documento de Systems Manager (documento de SSM) mediante una asociación de State Manager, puede configurar la asociación para elegir la salida de la ejecución del documento y puede enviar la salida a Amazon S3 o los Registros de Amazon CloudWatch (Registros de CloudWatch). Para facilitar la solución de problemas cuando una asociación ha terminado de ejecutarse, compruebe que la asociación esté configurada para escribir la salida del comando en un bucket de Amazon S3 o los Registros de CloudWatch. Para obtener más información, consulte Trabajo con asociaciones en Systems Manager.

Aplicación de atributos JSON a los destinos al ejecutar una receta

Puede especificar los atributos JSON para que su cliente de Chef los aplique a los nodos de destino durante una ejecución de asociación. Al configurar la asociación, puede proporcionar JSON sin procesar o la ruta a un archivo JSON almacenado en Amazon S3.

Utilice los atributos JSON cuando desee personalizar la forma en que se ejecuta la receta sin tener que modificar la propia receta, por ejemplo:

- Anular un número reducido de atributos

  Use un JSON personalizado para evitar tener que mantener varias versiones de una receta para adaptarse a pequeñas diferencias.
- Proporcionar los valores de las variables

  Use un JSON personalizado para especificar los valores que pueden cambiar de una ejecución a otra. Por ejemplo, si sus libros de recetas de Chef configuran una aplicación de terceros que acepta pagos, puede usar un JSON personalizado para especificar la URL del punto de conexión de pago.

Especificar atributos en JSON sin procesar

El siguiente es un ejemplo del formato que puede usar para especificar atributos JSON personalizados para su receta de Chef.

```
{"filepath":"/tmp/example.txt", "content":"Hello, World!"}
```

Especificación de una ruta a un archivo JSON

El siguiente es un ejemplo del formato que puede usar para especificar la ruta a los atributos JSON personalizados para su receta de Chef.

```
{"sourceType":"s3", "sourceInfo":"someS3URL1"}, {"sourceType":"s3",
 "sourceInfo":"someS3URL2"}
```

Usar Git como fuente para el libro de recetas

El documento `AWS-ApplyChefRecipes` utiliza el complemento aws:downloadContent para descargar libros de recetas de Chef. Para descargar contenido desde Git, especifique la información sobre el repositorio de Git en formato JSON, como se muestra en el siguiente ejemplo. Reemplace cada *example-resource-placeholder* con su propia información.

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

Usar GitHub como fuente de libro de recetas

El documento `AWS-ApplyChefRecipes` utiliza el complemento aws:downloadContent para descargar libros de recetas. Para descargar contenido desde GitHub, especifique la información sobre el repositorio de GitHub en formato JSON, como se muestra en el siguiente ejemplo. Reemplace cada *example-resource-placeholder* con su propia información.

```
{
   "owner":"TestUser",
   "repository":"GitHubCookbookRepository",
```

```
   "path":"cookbooks/HelloWorld",
   "getOptions":"branch:refs/head/main",
   "tokenInfo":"{{ssm-secure:token-secure-string-parameter}}"
}
```

Usar HTTP como fuente para el libro de recetas

Puede almacenar los libros de recetas de Chef en una ubicación HTTP personalizada ya sea como un solo archivo `.zip`, un archivo `tar.gz` o como una estructura de directorios. Para descargar contenido desde HTTP, especifique la ruta al archivo o al directorio en formato JSON, como se muestra en el siguiente ejemplo. Reemplace cada *`example-resource-placeholder`* con su propia información.

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

Uso de Amazon S3 como fuente de libro de recetas

También, puede almacenar y descargar libros de recetas de Chef en Amazon S3 como un único archivo `.zip`, un archivo `tar.gz` o como una estructura de directorios. Para descargar contenido desde Amazon S3, especifique la ruta al archivo en formato JSON, como se muestra en los ejemplos siguientes. Reemplace cada *`example-resource-placeholder`* con su propia información.

Ejemplo 1: descargar un libro de recetas específico

```
{
   "path":"https://s3.amazonaws.com/chef-cookbooks/HelloWorld.zip"
}
```

Ejemplo 2: descargar el contenido de un directorio

```
{
   "path":"https://s3.amazonaws.com/chef-cookbooks-test/HelloWorld"
}
```

> **Important**
>
> Si especifica Simple Storage Service (Amazon S3), el perfil de instancias de AWS Identity and Access Management (IAM) en los nodos administrados debe configurarse con la política `AmazonS3ReadOnlyAccess`. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager.

Crear una asociación que ejecute recetas de Chef (consola)

En el siguiente procedimiento se describe cómo utilizar la consola de Systems Manager para crear una asociación de State Manager que ejecuta libros de recetas de Chef mediante el documento de `AWS-ApplyChefRecipes`.

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija State Manager.
3. Elija State Manager y, a continuación, Create association (Crear asociación).
4. En Name (Nombre), escriba un nombre que le ayude a recordar el objetivo de la asociación.
5. En la lista Document (Documento), elija **`AWS-ApplyChefRecipes`**.
6. En Parámetros, en Tipo de origen, seleccione Git, GitHub, HTTP o S3.
7. En Información sobre el origen, introduzca la información sobre la fuente del libro de recetas en el formato adecuado para el Tipo de origen que seleccionó en el paso 6. Para obtener más información, consulte los temas siguientes:
   - the section called “Usar Git como fuente para el libro de recetas”
   - the section called “Usar GitHub como fuente de libro de recetas”
   - the section called “Usar HTTP como fuente para el libro de recetas”
   - the section called “Uso de Amazon S3 como fuente de libro de recetas”
8. En Run list (Lista de ejecución), enumera las recetas que desea ejecutar en el siguiente formato, separando cada receta con una coma como se muestra. No incluya un espacio después de la coma. Reemplace cada *`example-resource-placeholder`* con su propia información.

```
recipe[cookbook-name1::recipe-name],recipe[cookbook-name2::recipe-name]
```

9. (Opcional) Especifique los atributos JSON personalizados que desee que el cliente de Chef pase a los nodos de destino.

   a. En Contenido de atributos JSON, agregue cualquier atributo que desee que el cliente de Chef pase a los nodos de destino.

   b. En Contenido de atributos JSON, agregue las rutas a los atributos que desee que el cliente de Chef pase a los nodos de destino.

   Para obtener más información, consulte the section called “Aplicación de atributos JSON a los destinos al ejecutar una receta”.

10. Para Versión de cliente de Chef, especifique una versión de Chef. Los valores válidos son de `11` a `18` o `None`. Si especifica un número entre `11` y `18` (inclusive), Systems Manager instala la versión correcta del cliente de Chef en los nodos de destino. Si especifica `None`, Systems Manager no instala el cliente de Chef en nodos de destino antes de ejecutar las recetas del documento.

11. (Opcional) En Argumentos del cliente deChef Chef, especifique argumentos adicionales que sean compatibles con la versión de Chef que esté utilizando. Para obtener más información acerca de los argumentos admitidos, ejecute `chef-client -h` en un nodo que esté ejecutando el cliente de Chef.

12. (Opcional) Active Why-Run para mostrar los cambios que se realizaron en los nodos de destino si se ejecutan las recetas, sin cambiar realmente los nodos de destino.

13. Para Compliance severity (Severidad de conformidad), elija la severidad de los resultados de conformidad de configuración de Systems Manager que desee informar. Los informes de conformidad indican si el estado de asociación es conforme o no conforme, junto con el nivel de gravedad que especifique. Los informes de conformidad se almacenan en un bucket de S3 que se especifica como valor del parámetro de Bucket del informe de conformidad (paso 14). Para obtener más información acerca de la conformidad, consulte Detalles sobre el cumplimiento en esta guía.

    Los análisis de conformidad miden la desviación entre la configuración especificada en las recetas de Chef y los recursos de nodo. Los valores válidos son `Critical`, `High`, `Medium`, `Low`, `Informational`, `Unspecified` o `None`. Para omitir los informes de cumplimiento, elija `None`.

14. En Compliance type (Tipo de conformidad), especifique el tipo de conformidad para el que desea que se informe de los resultados. Los valores válidos son `Association` para asociaciones de State Manager o `Custom:`*`custom-type`*. El valor predeterminado es `Custom:Chef`.

15. En Bucket de informe de conformidad, ingrese el nombre de un bucket de S3 en el que almacene información sobre cada ejecución de Chef realizada por este documento, incluidos los resultados de configuración de recursos y de conformidad.
16. En Rate control (Control de frecuencia), configure las opciones para ejecutar asociaciones de State Manager a través de una flota de nodos administrados. Para obtener más información sobre el uso de controles de frecuencia, consulte Cómo comprender los controles de frecuencia y destinos en las asociaciones de State Manager.

    En Concurrency (Simultáneamente), elija una opción:

    - Elija Targets (Destinos) para introducir un número absoluto de destinos que pueda ejecutar la asociación de forma simultánea.
    - Elija porcentaje para introducir un porcentaje del destino definido que puede ejecutar la asociación de forma simultánea.

    En Error threshold (Umbral de error), elija una opción:

    - Elija errors (errores) para especificar un número absoluto de errores permitidos antes de que State Manager deje de ejecutar asociaciones en más destinos.
    - Elija percentage (porcentaje) para especificar un porcentaje de errores permitidos antes de que State Manager deje de ejecutar asociaciones en más destinos.
17. (Opcional) En Output options (Opciones de salida), para guardar la salida del comando en un archivo, seleccione el cuadro Enable writing output to S3 (Permitir la escritura de salida en S3). Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

    > **Note**
    >
    > Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancias asignado al nodo administrado, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, verifique que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.
18. Elija Crear asociación.

Crear una asociación que ejecute recetas de Chef (CLI)

En el siguiente procedimiento se describe cómo utilizar la AWS Command Line Interface (AWS CLI) para crear una asociación de State Manager que ejecute cuadernos de trabajo de Chef mediante el documento `AWS-ApplyChefRecipes`.

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Ejecute uno de los siguientes comandos para crear una asociación que ejecute libros de recetas de Chef en nodos de destino que contengan las etiquetas especificadas. Use el comando adecuado para el tipo de fuente y el sistema operativo de su libro de recetas. Reemplace cada *`example-resource-placeholder`* con su propia información.

   a. Origen de 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. Origen de GitHub

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"
```

A continuación se muestra un ejemplo.

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. Origen de 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. Origen de 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": ["amzn-s3-demo-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": ["amzn-s3-demo-bucket"]}' ^
    --association-name "name" ^
    --schedule-expression "cron_or_rate_expression"
```

A continuación se muestra un ejemplo.

Linux & macOS

```
aws ssm create-association --name "AWS-ApplyChefRecipes" \
    --targets "Key=tag:OS,Values= Linux" \
    --parameters '{"SourceType":["S3"],"SourceInfo":["{\"path
\":\"https://s3.amazonaws.com/amzn-s3-demo-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/amzn-s3-demo-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 *)"
```

El sistema crea la asociación y, a menos que la expresión cron o rate especificada lo impida, el sistema ejecuta la asociación en los nodos de destino.

> **Note**
>
> Las asociaciones de State Manager no admiten todas las expresiones cron y de frecuencia. Para obtener más información acerca de la creación de expresiones cron y de frecuencia para asociaciones, consulte Referencia: expresiones cron y rate para Systems Manager.

3. Ejecute el siguiente comando para ver el estado de la asociación que acaba de crear.

```
aws ssm describe-association --association-id "ID"
```

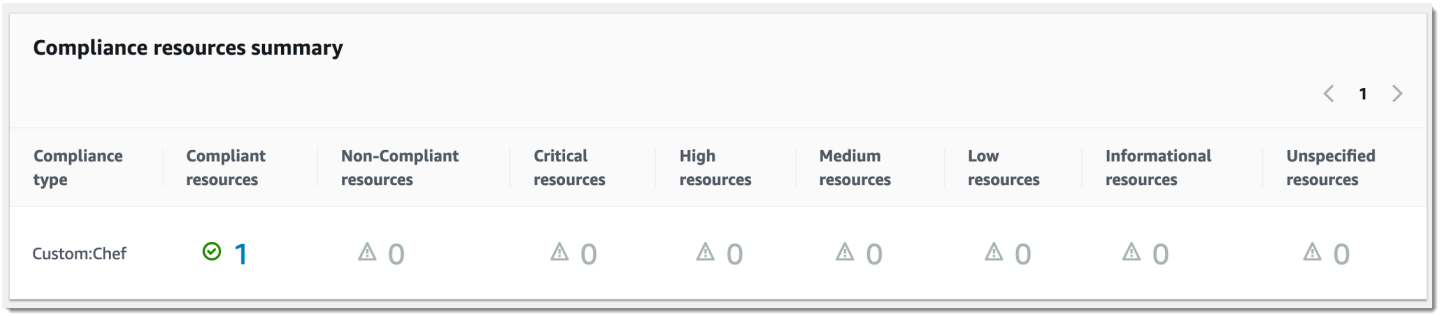
Visualización de los detalles de cumplimiento del recurso de Chef

Systems Manager captura información de conformidad sobre los recursos administrados por Chef en el valor Bucket de informe de conformidad de Amazon S3 que especificó cuando ejecutó el documento `AWS-ApplyChefRecipes`. La búsqueda de información acerca de los errores del recurso de Chef en un bucket de S3 puede llevar tiempo. En su lugar, puede ver esta información en la página Compliance (Conformidad) de Systems Manager.

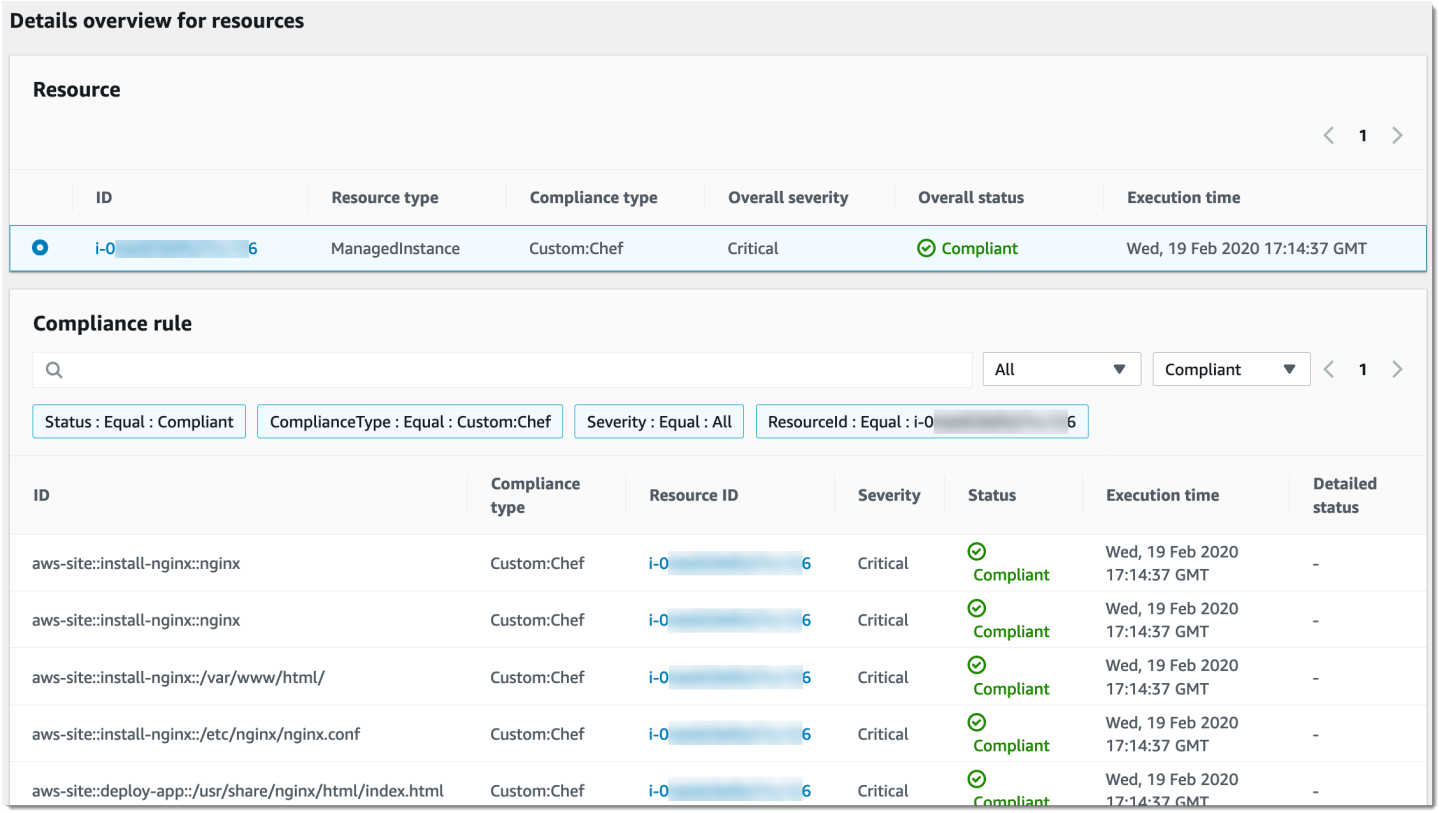
Un análisis de conformidad de Systems Manager recopila información acerca de los recursos de los nodos administrados que se crearon o se registraron en la ejecución más reciente de Chef. Los recursos pueden incluir archivos, directorios, servicios de `systemd`, paquetes de `yum`, archivos con plantillas, paquetes de `gem` y libros de recetas dependientes, entre otros.

La sección Compliance resources summary (Resumen de los recursos de cumplimiento muestra una cuenta de los recursos que fallaron. En el siguiente ejemplo, el Tipo de conformidad es Personalizado: Chef y un recurso no conforme.

> **Note**
>
> `Custom:Chef` es el valor ComplianceType predeterminado del documento `AWS-ApplyChefRecipes`. Este valor se puede personalizar.



La sección Details overview for resources (Información general de los detalles de los recursos) muestra información sobre el recurso de AWS que no cumple los requisitos. Esta sección también incluye el tipo de recursos de Chef con el que se ejecutó la conformidad, la gravedad del problema, el estado de conformidad y los vínculos a información adicional cuando corresponda.



View output (Ver salida) muestra los últimos 4000 caracteres del estado detallado. Systems Manager comienza con la excepción como primer elemento, encuentra mensajes detallados y los muestra hasta que alcanza la cuota de 4000 caracteres. Este proceso muestra los mensajes de registros que salieron antes de lanzar la excepción. Estos mensajes son los más importantes para la resolución de errores.

Para obtener más información acerca de cómo ver la información de cumplimiento, consulte Conformidad de AWS Systems Manager.

> **Important**
>
> Si la asociación de State Manager falla, no se notifican datos de cumplimiento. Por ejemplo, si Systems Manager intenta descargar un libro de recetas de Chef de un bucket de S3 para el cual el nodo no tiene permiso de acceso, la asociación falla y Systems Manager no notifica datos de conformidad.

Tutorial: actualización automática de SSM Agent con AWS CLI

El siguiente procedimiento le guía por el proceso de crear una asociación de State Manager con AWS Command Line Interface. La asociación actualiza automáticamente el SSM Agent según la programación que especifique. Para obtener más información acerca de SSM Agent, consulte Uso de SSM Agent. Para personalizar la programación de actualización de SSM Agent mediante la consola, consulte Actualización automática de SSM Agent.

Si desea recibir notificaciones sobre actualizaciones de SSM Agent, suscríbase a la página de SSM Agent Release Notes en GitHub.

Antes de empezar

Antes de completar el siguiente procedimiento, compruebe que tiene al menos una instancia de Amazon Elastic Compute Cloud (Amazon EC2) en ejecución para Linux, macOS o Windows Server y que esté configurada para Systems Manager. Para obtener más información, consulte Configuración de nodos administrados para AWS Systems Manager.

Si crea una asociación mediante la AWS CLI o AWS Tools for Windows PowerShell, utilice el parámetro `--Targets` para definir las instancias, tal y como se muestra en el siguiente ejemplo. No utilice el parámetro `--InstanceID`. El parámetro `--InstanceID` es un parámetro heredado.

Si desea crear una asociación para actualizar automáticamente el SSM Agent

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.
2. Ejecute el siguiente comando para crear una asociación dirigiéndose a instancias con las etiquetas de Amazon Elastic Compute Cloud (Amazon EC2). Reemplace cada *`example resource placeholder`* con su propia información. El parámetro `Schedule` establece una programación para ejecutar la asociación todos los domingos a las 2:00. (UTC).

   Las asociaciones de State Manager no admiten todas las expresiones cron y de frecuencia. Para obtener más información acerca de la creación de expresiones cron y de frecuencia para asociaciones, consulte Referencia: expresiones cron y rate para 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 *)"
```

Puede dirigir varias instancias especificando los ID de instancia en una lista separada por comas.

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 *)"
```

Puede especificar la versión de SSM Agent que desea actualizar.

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
```

El sistema devuelve información similar a la siguiente.

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

El sistema intenta crear la asociación en las instancias y aplicar el estado después de la creación. El estado de la asociación muestra `Pending`.

3. Ejecute el siguiente comando para ver un estado actualizado de la asociación que creó.

```
aws ssm list-associations
```

Si las instancias no están ejecutando la versión más reciente del SSM Agent, el estado muestra `Failed`. Cuando se publica una nueva versión del SSM Agent, la asociación instala automáticamente el nuevo agente y el estado es `Success` (Correcto).

Tutorial: actualización automática de los controladores PV en las instancias de EC2 de Windows Server

Las Amazon Machine Images (AMIs) de Windows para Amazon incluyen un conjunto de controladores que permiten el acceso a hardware virtualizado. Amazon Elastic Compute Cloud (Amazon EC2) utiliza estos controladores para mapear el almacén de instancias y los volúmenes de Amazon Elastic Block Store (Amazon EBS) a sus dispositivos. Se recomienda instalar los últimos controladores para mejorar la estabilidad y el rendimiento de las instancias EC2 en Windows Server. Para obtener más información acerca de los controladores PV, consulte Controladores de AWS PV.

En la siguiente explicación, se muestra cómo configurar una asociación de State Manager para descargar e instalar automáticamente los nuevos controladores de AWS PV cuando estén disponibles. State Manager es una herramienta de AWS Systems Manager.

Antes de empezar

Antes de completar el siguiente procedimiento, verifique que tiene al menos una instancia de Amazon EC2 para Windows Server en ejecución que esté configurada para Systems Manager. Para obtener más información, consulte Configuración de nodos administrados para AWS Systems Manager.

Para crear una asociación de State Manager que actualice automáticamente los controladores PV

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija State Manager.
3. Elija Crear asociación.
4. En el campo Nombre, escriba un nombre descriptivo para la asociación.
5. En la lista Document (Documento), elija `AWS-ConfigureAWSPackage`.
6. En la sección Parámetros, realice lo siguiente:
   - En Acción, elija Instalar.

- En Tipo de instalación, elija Desinstalar y volver a instalar.

> **Note**
>
> Las actualizaciones locales no son compatibles con este paquete. Debe desinstalarse y reinstalarse.

- En Nombre, escriba **`AWSPVDriver`**.

No es necesario introducir nada para la versión y los argumentos adicionales.

7. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

> **Tip**
>
> Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

> **Note**
>
> Si elige dirigirse a las instancias mediante etiquetas y especifica etiquetas que se mapean a instancias de Linux, la asociación se realiza correctamente en la instancia de Windows, pero no en las instancias de Linux. El estado general de la asociación muestra Failed.

8. En el área Especificar la programación, elija si desea ejecutar la asociación según la programación que usted configure o solo una vez. Los controladores PV actualizados se lanzan varias veces al año, por lo que puede programar la asociación para que se ejecute una vez al mes, si lo desea.
9. En el área Opciones avanzadas, en Gravedad de conformidad, elija un nivel de gravedad para la asociación. Los informes de conformidad indican si el estado de asociación es conforme o no conforme, junto con el nivel de gravedad que se indique aquí. Para obtener más información, consulte Acerca de la conformidad de las asociaciones de State Manager.
10. En Rate control (Control de velocidad):

- En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

> **Note**
>
> Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

- En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

11. (Opcional) En Output options (Opciones de salida), para guardar la salida del comando en un archivo, seleccione el cuadro Enable writing output to S3 (Permitir la escritura de salida en S3). Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancias asignado al nodo administrado, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, verifique que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

12. (Opcional) En la sección de alarma de CloudWatch, en Nombre de alarma, elija una alarma de CloudWatch existente para aplicarla a su tarea de monitoreo.

> **Note**
>
> Tenga en cuenta la siguiente información sobre este paso.

- La lista de alarmas muestra un máximo de 100 alarmas. Si no ve su alarma en la lista, utilice la AWS Command Line Interface para crear la asociación. Para obtener más información, consulte Cómo rear una asociación (línea de comandos).
- Para adjuntar una alarma de CloudWatch a su comando, la entidad principal de IAM que crea la asociación debe tener permiso para la acción `iam:createServiceLinkedRole`. Para obtener más información sobre las alarmas de CloudWatch, consulte Uso de alarmas de Amazon CloudWatch.
- Tenga en cuenta que si la alarma se activa, no se ejecutarán automatizaciones o invocaciones de comandos pendiente.

13. Elija Crear asociación y, a continuación, Cerrar. El sistema intenta crear la asociación en las instancias y aplicar inmediatamente el estado.

    Si ha creado la asociación en una o varias instancias de Amazon EC2 en Windows Server, el estado cambia a Success (Correcto). Si las instancias no están configuradas para Systems Manager, o si se ha dirigido accidentalmente a instancias de Linux, el estado muestra Failed (Error).

    Si el estado es Failed (Error), elija el ID de asociación, elija la pestaña Resources (Recursos) y, a continuación, asegúrese de que la asociación se ha creado correctamente en las instancias EC2 en Windows Server. Si las instancias EC2 en Windows Server tienen el estado Failed (Error), asegúrese de que SSM Agent se está ejecutando en la instancia y que esta se ha configurado como un rol de AWS Identity and Access Management (IAM) en Systems Manager. Para obtener más información, consulte Configuración de la consola unificada de Systems Manager para una organización.

Herramientas de administración de cambios de AWS Systems Manager

AWS Systems Manager proporciona las siguientes herramientas para hacer cambios en los recursos de AWS.

Temas

- AWS Systems Manager Automation
- AWS Systems Manager Change Calendar

- AWS Systems Manager Change Manager
- Documentos de AWS Systems Manager
- AWS Systems Manager Maintenance Windows
- AWS Systems Manager Quick Setup

AWS Systems Manager Automation

Automatización, una herramienta de AWS Systems Manager, simplifica las tareas comunes de mantenimiento, implementación y corrección para Servicios de AWS como Amazon Elastic Compute Cloud (Amazon EC2), Amazon Relational Database Service (Amazon RDS), Amazon Redshift, Amazon Simple Storage Service (Amazon S3), etc. Para comenzar a utilizar Automation, abra la consola de Systems Manager. En el panel de navegación, elija automatización.

Automation lo ayuda a crear soluciones automatizadas para implementar, configurar y administrar recursos de AWS a escala. Con Automation, tiene un control pormenorizado de la simultaneidad de sus automatizaciones. Esto significa que puede especificar a cuántos recursos desea dirigirse simultáneamente y cuántos errores pueden producirse antes de que se detenga una automatización.

Para ayudarlo a comenzar a utilizar Automation, AWS desarrolla y mantiene varios manuales de procedimientos predefinidos. Según su caso de uso, puede utilizar estos manuales de procedimientos predefinidos que realizan diversas tareas o puede crear sus propios manuales de procedimientos personalizados que se adapten mejor a sus necesidades. Para monitorear el progreso y el estado de las automatizaciones, puede utilizar la consola de Automatización de Systems Manager o la herramienta de la línea de comandos que prefiera. Automation también se integra con Amazon EventBridge para ayudarlo a crear una arquitectura basada en eventos a escala.

¿Cómo puede beneficiar a mi organización Automation?

Automation ofrece los siguientes beneficios:

- Compatibilidad con scripting en el contenido del manual de procedimientos

  Mediante la acción `aws:executeScript`, puede ejecutar funciones personalizadas de Python y PowerShell directamente desde sus manuales de procedimientos. Esto le proporciona mayor flexibilidad a la hora de crear manuales de procedimientos personalizados porque puede completar varias tareas que otras acciones de Automation no admiten. También tiene un mayor control sobre la lógica del manual de procedimientos. Para ver un ejemplo de cómo se puede utilizar esta

acción y cómo puede ayudar a mejorar una solución automatizada existente, consulte Creación de manuales de procedimientos de Automation.

- Ejecutar automatizaciones en varias Cuentas de AWS y Regiones de AWS desde una ubicación centralizada

  Los administradores pueden ejecutar automatizaciones de recursos en varias cuentas y regiones desde la consola de Systems Manager.

- Mejora de la seguridad de las operaciones

  Los administradores disponen de un lugar centralizado para conceder y denegar el acceso a manuales de procedimientos. Si utiliza solo políticas de AWS Identity and Access Management (IAM), puede controlar qué usuarios individuales o grupos de la organización pueden utilizar la Automation y a qué manuales de procedimientos pueden acceder.

- Automatizar las tareas de TI habituales

  La automatización de tareas comunes puede ayudar a mejorar la eficiencia operativa, cumplir los estándares organizativos y reducir los errores del operador. Por ejemplo, puede utilizar el manual de procedimientos `AWS-UpdateCloudFormationStackWithApproval` para actualizar los recursos que se implementaron con una plantilla de AWS CloudFormation. La actualización aplica una nueva plantilla. Puede configurar Automation para solicitar la aprobación de uno o varios usuarios de antes de que la actualización se ponga en marcha.

- Realización de tareas disruptivas en lote de forma segura

  Automation incluye características, como controles de frecuencia, que le permiten controlar la implementación de una automatización en su flota al especificar un valor de simultaneidad y un umbral de error. Para obtener más información acerca del uso de controles de frecuencias, consulte Ejecución de operaciones automatizadas a escala.

- Simplificar tareas complejas

  Automation proporciona manuales de procedimientos predefinidos que agilizan tareas complejas y lentas, como la creación de Amazon Machine Images maestras (AMIs). Por ejemplo, puede utilizar los manuales de procedimientos `AWS-UpdateLinuxAmi` y `AWS-UpdateWindowsAmi` para crear AMIs maestras a partir de una AMI de origen. Mediante los manuales de procedimientos puede ejecutar scripts personalizados antes y después de la aplicación de actualizaciones. Asimismo, puede incluir paquetes de software específicos en la instalación o excluirlos de ella. Para ver ejemplos de cómo ejecutar estos manuales de procedimientos, consulte Tutoriales.

- Definir restricciones para las entradas

Puede definir restricciones en manuales de procedimientos personalizados para limitar los valores que Automation aceptará para un parámetro de entrada concreto. Por ejemplo, `allowedPattern` solo aceptará valores para un parámetro de entrada que coincidan con la expresión regular que defina. Si especifica `allowedValues` para un parámetro de entrada, solo se aceptan los valores especificados en el manual de procedimientos.

- Registre la salida de las acciones de automatización en Amazon CloudWatch Logs

  Para cumplir los requisitos operativos y de seguridad de su organización, es posible que tenga que proporcionar un registro de los scripts que se ejecutan durante un manual de procedimientos. Con CloudWatch Logs, puede acceder a los archivos de registros de diversos Servicios de AWS, monitorearlos y almacenarlos. Puede utilizar la salida de la acción `aws:executeScript` almacenada en el grupo de registros de CloudWatch Logs para depurar y resolver problemas. Los datos de registro se pueden enviar al grupo de registros con o sin cifrado de AWS KMS usando su clave de KMS. Para obtener más información, consulte Registro de salida de acción de Automation con CloudWatch Logs.

- Integración con Amazon EventBridge

  Automation se admite como tipo de destino en las normas de Amazon EventBridge. Esto significa que puede activar manuales de procedimientos mediante eventos. Para obtener más información, consulte Monitoreo de eventos de Systems Manager con Amazon EventBridge y Referencia: patrones y tipos de eventos de Amazon EventBridge para Systems Manager.

- Compartir prácticas recomendadas de la organización

  Puede definir prácticas recomendadas para la administración de recursos, tareas de operaciones y más en manuales de procedimientos que comparte entre cuentas y regiones.

¿Quién debe utilizar Automation?

- Todo cliente de AWS que desea mejorar su eficiencia operativa a escala, reducir los errores asociados a la intervención manual y reducir el tiempo de resolución de problemas comunes.
- Expertos en infraestructura que deseen automatizar las tareas de implementación y configuración.
- Administradores que deseen resolver problemas comunes de forma fiable, mejorar la eficiencia de la solución de problemas y reducir las operaciones repetitivas.
- Usuarios que deseen automatizar una tarea que normalmente realizan de forma manual.

¿Qué es una automatización?

Una automatización se compone de todas las tareas que se definen en un manual de procedimientos y que realiza el servicio de Automation. Automation utiliza los siguientes componentes para ejecutar flujos de trabajo de automatización.

Concepto	Detalles
Manual de procedimientos de automatización	Un manual de procedimientos de Automatización de Systems Manager define la automatización (las acciones que Systems Manager realiza en los nodos administrados y los recursos de AWS). Automation incluye varios manuales de procedimientos predefinidos que se pueden utilizar para realizar tareas comunes, como reiniciar una o más instancias de Amazon EC2, o crear una Amazon Machine Image (AMI). También puede crear sus propios manuales de procedimientos. Los manuales de procedimientos utilizan YAML o JSON, e incluyen los pasos y parámetros que especifique. Los pasos se ejecutan en orden secuencial. Para obtener más información, consulte Creación de sus propios manuales de procedimientos. Los manuales de procedimientos son documentos de Systems Manager de tipo `Automation` , a diferencia de los documentos de `Command`, `Policy` y `Session`. Los manuales de procedimientos admiten la versión de esquema 0.3. Los documentos de comandos utilizan las versiones de esquema 1.2, 2.0 o 2.2. Los documentos de políticas utilizan la versión de esquema 2.0 o una posterior.

Concepto	Detalles
Acción de Automation	La automatización definida en un manual de procedimientos incluye uno o más pasos. Cada paso está asociado a una acción concreta. La acción determina las entradas, el comportamiento y las salidas del paso. Los pasos se definen en la sección `mainSteps` de su manual de procedimientos. La automatización admite 20 tipos de acción distintos. Para obtener más información, consulte Referencia de acciones de Automatización de Systems Manager.
Cuota de Automation	Cada Cuenta de AWS puede ejecutar 100 automatizaciones de manera simultánea. Esto incluye automatizaciones secundarias (automatizaciones iniciadas por otra automatización) y automatizaciones de control de frecuencia. Si intenta ejecutar más automatizaciones, Systems Manager agrega las automatizaciones adicionales a una cola y muestra el estado Pending (Pendiente). Esta cuota se puede ajustar empleando simultaneidad adaptativa. Para obtener más información, consulte Permitir que Automation se adapte a sus necesidades de simultaneidad. Para obtener más información sobre la ejecución de las automatizaciones, consulte Ejecución de una operación automatizada con la tecnología de automatización de Systems Manager.

Concepto	Detalles
Cuota de cola de Automation	Si intenta ejecutar más automatizaciones que las permitidas por el límite de automatizaciones simultáneas, las automatizaciones posterior es se agregarán a una cola. Cada Cuenta de AWS puede mantener en cola 5000 automatiz aciones. Cuando se completa una automatiz ación (o alcanza un estado terminal), se inicia la primera automatización que se encuentra en la cola.
Cuota de automatización de control de frecuencia	Cada Cuenta de AWS puede ejecutar 25 automatizaciones de control de frecuenci a de manera simultánea. Si intenta ejecutar más automatizaciones de control de frecuenci a que las permitidas por el límite de automatiz aciones simultáneas de control de frecuencia, Systems Manager agrega las automatizaciones de control de frecuencia posteriores a una cola y muestra el estado Pending (Pendient e). Para obtener más información acerca de la ejecución de automatizaciones de control de frecuencia, consulte Ejecución de operaciones automatizadas a escala.
Cuota de cola de automatización de control de frecuencia	Si intenta ejecutar más automatizaciones que las permitidas por el límite de automatiz aciones simultáneas de control de frecuencia, las automatizaciones posteriores se agregarán a una cola. Cada Cuenta de AWS puede mantener en cola 1000 automatizaciones de control de frecuencia. Cuando se completa una automatización (o alcanza un estado terminal) , se inicia la primera automatización que se encuentra en la cola.

Temas

- Configuración de Automation
- Ejecución de una operación automatizada con la tecnología de automatización de Systems Manager
- Ejecución de una automatización que requiere aprobaciones
- Ejecución de operaciones automatizadas a escala
- Ejecución de automatizaciones en varias cuentas y Regiones de AWS
- Ejecución de automatizaciones a partir de eventos EventBridge
- Ejecutar una automatización paso a paso
- Programación de automatizaciones con asociaciones de State Manager
- Programación de automatizaciones con periodos de mantenimiento
- Referencia de acciones de Automatización de Systems Manager
- Creación de sus propios manuales de procedimientos
- Referencia del manual de procedimientos de Automatización de Systems Manager
- Tutoriales
- Obtenga información sobre los estados devueltos por la automatización de Systems Manager
- Solución de problemas de Automatización de Systems Manager

Configuración de Automation

Para configurar Automatización, una herramienta de AWS Systems Manager, debe verificar el acceso de los usuarios al servicio de Automatización y configurar los roles según la situación de manera que el servicio pueda llevar a cabo acciones en sus recursos. También le recomendamos que opte por el modo de simultaneidad adaptable en sus preferencias de Automation. La simultaneidad adaptativa escala automáticamente la cuota de automatización para satisfacer sus necesidades. Para obtener más información, consulte Permitir que Automation se adapte a sus necesidades de simultaneidad.

Para garantizar el acceso adecuado a AWS Systems Manager Automation, revise los siguientes requisitos de usuario y rol de servicio.

Comprobación del acceso del usuario para manuales de procedimientos

Verifique que tiene permiso para usar manuales de procedimientos. Si su usuario, grupo o rol tiene asignados permisos de administrador, entonces tendrá acceso a Automatización de Systems

Manager. Si no tiene permisos de administrador, un administrador debe concederle permiso mediante la asignación de la política administrada `AmazonSSMFullAccess` o de una política que proporcione permisos comparables a su usuario, grupo o rol.

> **Important**
>
> La política de IAM `AmazonSSMFullAccess` concede permisos para las acciones de Systems Manager. Sin embargo, algunos manuales de procedimientos necesitan permisos para otros servicios, como el manual de procedimientos `AWS-ReleaseElasticIP`, que requiere permisos de IAM para `ec2:ReleaseAddress`. Por lo tanto, debe revisar las acciones realizadas en un manual de procedimientos a fin de asegurarse de que se asignen a su usuario, grupo o rol los permisos necesarios para realizar las acciones incluidas en el manual.

Configuración del acceso de un rol de servicio (rol de asunción) para automatizaciones

Las automatizaciones se pueden iniciar en el contexto de un rol de servicio (o rol de asunción). Esto permite al servicio realizar acciones en su nombre. Si no especifica un rol de asunción, Automation utiliza el contexto del usuario que invocó la automatización.

Sin embargo, en las siguientes situaciones es necesario especificar un rol de servicio para Automation:

- Cuando se desea restringir los permisos de un usuario para un recurso, pero se desea que el usuario ejecute una automatización que requiere permisos más avanzados. En este escenario, puede crear un rol de servicio con permisos más avanzados y permitir al usuario que ejecute la automatización.
- Cuando crea una asociación de Systems Manager State Manager que ejecuta un manual de procedimientos.
- Cuando tenga operaciones que espera que se ejecuten durante más de 12 horas.
- Cuando ejecuta un manual de procedimientos que no es propiedad de Amazon y que utiliza la acción `aws:executeScript` para llamar una operación de la API de AWS o para actuar sobre un recurso de AWS. Para obtener más información, consulta Permisos para utilizar los manuales de procedimientos.

Si precisa crear un rol de servicio para Automation, puede emplear uno de los siguientes métodos.

Temas

- Creación de un rol de servicio para Automation mediante AWS CloudFormation
- Cree los roles de servicio para la automatización mediante la consola
- Permitir que Automation se adapte a sus necesidades de simultaneidad
- Implementación de controles de cambio para Automatización

Creación de un rol de servicio para Automation mediante AWS CloudFormation

Puede crear un rol de servicio para Automatización, una herramienta de AWS Systems Manager, a partir de una plantilla de AWS CloudFormation. Después de crear el rol de servicio, puede especificar el rol de servicio en manuales de procedimientos mediante el parámetro `AutomationAssumeRole`.

Crear el rol de servicio utilizando AWS CloudFormation

Utilice el siguiente procedimiento a fin de crear el rol de AWS Identity and Access Management (IAM) necesario para la Automatización de Systems Manager con AWS CloudFormation.

Para crear el rol de IAM necesario

1. Descargue y descomprima el archivo `AWS-SystemsManager-AutomationServiceRole.zip`. Esta carpeta incluye el archivo de plantilla `AWS-SystemsManager-AutomationServiceRole.yaml` de AWS CloudFormation.
2. Abra la consola de AWS CloudFormation en https://console.aws.amazon.com/cloudformation.
3. Elija Create Stack.
4. En la sección Specify template (Especificar plantilla) seleccione Upload a template file (Cargar un archivo de plantilla).
5. Elija Browse (Examinar) y, a continuación, elija el archivo de plantilla de AWS CloudFormation `AWS-SystemsManager-AutomationServiceRole.yaml`.
6. Elija Siguiente.
7. En la página Especificar los detalles de la pila, escriba un nombre en el campo Nombre de la pila.
8. En la página Configure stack options (Configurar opciones de la pila) no es necesario seleccionar ninguna opción. Elija Siguiente.
9. En la página Review (Revisar), desplácese hacia abajo y elija la opción I acknowledge that AWS CloudFormation might create IAM resources (Confirmo que puede crear recursos de IAM).
10. Seleccione Crear.

CloudFormation muestra el estado CREATE_IN_PROGRESS durante tres minutos aproximadamente. El estado cambia a CREATE_COMPLETE una vez que se haya creado la pila y los roles estén listos para su uso.

> **Important**
>
> Si ejecuta un flujo de trabajo de automatización que invoca otros servicios mediante un rol de servicio de AWS Identity and Access Management (IAM), tenga en cuenta que el rol de servicio debe configurarse con el permiso necesario para invocar dichos servicios. Este requisito se aplica a todos los manuales de procedimientos de automatización de AWS (manuales de `AWS-*`), como los manuales de procedimientos `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup` y `AWS-RestartEC2Instance`, por nombrar algunos. Este requisito también se aplica a cualquier manual de procedimientos de automatización personalizado que cree para llamar otros Servicios de AWS mediante acciones que llaman a otros servicios. Por ejemplo, si utiliza las acciones `aws:executeAwsApi`, `aws:createStack` o `aws:copyImage`, configure el rol de servicio con el permiso necesario para invocar dichos servicios. Puede conceder permisos a otros Servicios de AWS mediante la incorporación de una política insertada de IAM al rol. Para obtener más información, consulte (Opcional) Agregar una política insertada de Automatización o una política administrada por el cliente para invocar otros Servicios de AWS.

Copia de la información de roles para Automation

Utilice el siguiente procedimiento para copiar información sobre el rol de perfil de instancia y el rol de servicio de Automation desde la consola de AWS CloudFormation. Debe especificar estos roles cuando utiliza un manual de procedimientos.

> **Note**
>
> No es necesario que copie la información de los roles con este procedimiento si ejecuta los manuales de procedimientos `AWS-UpdateLinuxAmi` o `AWS-UpdateWindowsAmi`. Estos manuales de procedimientos ya tienen los roles obligatorios especificados como valores predeterminados. Los roles especificados en estos manuales de procedimientos usan las políticas administradas de IAM.

Para copiar los nombres de rol

1. Abra la consola de AWS CloudFormation en https://console.aws.amazon.com/cloudformation.
2. Seleccione el nombre de pila de Automation que creó en el procedimiento anterior en Stack name (Nombre de la pila).
3. Elija la pestaña Recursos.
4. Elija el enlace Physical ID para AutomationServiceRole. La consola de IAM se abre en un resumen del rol del servicio de Automation.
5. Copie el nombre de recurso de Amazon (ARN) situado junto a Role ARN (ARN del rol). El ARN es similar al que se muestra a continuación: `arn:aws:iam::12345678:role/AutomationServiceRole`
6. Pegue el ARN en un archivo de texto para utilizarlo más adelante.

Ha terminado de configurar el rol de servicio de Automation. Ahora, puede utilizar el ARN del rol de servicio de Automation en sus manuales de procedimientos.

Cree los roles de servicio para la automatización mediante la consola

Si necesita crear un rol de servicio para Automatización, una herramienta de AWS Systems Manager, complete las siguientes tareas. Para obtener más información acerca de cuándo se necesita un rol de servicio para Automation, consulte Configuración de Automation.

Tareas

- Tarea 1: crear un rol de servicio para Automation
- Tarea 2: asociar la política iam:PassRole al rol de Automation

Tarea 1: crear un rol de servicio para Automation

Siga este procedimiento a fin de crear un rol de servicio (o rol de asunción) para la Automatización de Systems Manager.

> **Note**
>
> También puede utilizar este rol en manuales de procedimientos, como `AWS-CreateManagedLinuxInstance`. La utilización de este rol o del nombre de recurso de Amazon (ARN) de un rol de AWS Identity and Access Management (IAM) en los manuales

de procedimientos permite a Automation realizar acciones en su entorno, como lanzar instancias nuevas y realizar acciones en su nombre.

Para crear una función de IAM y permitir que Automation la asuma

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, seleccione Roles y luego seleccione Create role.
3. En Select type of trusted entity (Seleccionar tipo de entidad de confianza), elija AWS service (Servicio de AWS).
4. En la sección Elija un caso de uso, elija Systems Manager y, a continuación, elija Next: Permissions (Siguiente: Permisos).
5. En la página Attached permissions policy (Política de permisos adjunta), busque la política AmazonSSMAutomationRole, elíjala y, después, seleccione Next: Review (Siguiente: Revisión).
6. En la página Review (Revisar), ingrese un nombre en el cuadro Role name (Nombre de rol) y, a continuación, escriba una descripción.
7. Elija Create role. El sistema le devuelve a la página Roles.
8. En la página Roles, elija el rol que acaba de crear para abrir la página Summary (Resumen). Anote los valores de Role Name (Nombre de rol) y Role ARN (ARN de rol). Especificará el ARN de rol al asociar la política iam:PassRole a su cuenta de IAM en el procedimiento siguiente. También puede especificar el nombre del rol y el ARN en los manuales de procedimientos.

> **Note**
>
> La política `AmazonSSMAutomationRole` asigna al rol de Automation permiso para acceder a un subconjunto de funciones de AWS Lambda en su cuenta. Estas funciones comienzan por "Automation". Si piensa utilizar Automation con funciones de Lambda, el ARN de Lambda debe utilizar el siguiente formato: `"arn:aws:lambda:*:*:function:Automation*"`
> Si dispone de funciones de Lambda cuyos ARN no utilizan este formato, también debe adjuntar una política de Lambda adicional al rol de automatización, como la política AWSLambdaRole. La política o el rol adicionales deben proporcionar un acceso más amplio a las funciones de Lambda en la Cuenta de AWS.

Después de crear su rol de servicio, le recomendamos que modifique la política de confianza para ayudar a evitar el problema del suplente confuso entre servicios. El problema de la sustitución confusa es una cuestión de seguridad en la que una entidad que no tiene permiso para realizar una acción puede obligar a una entidad con más privilegios a realizar la acción. En AWS, la suplantación entre servicios puede dar lugar al problema de la sustitución confusa. La suplantación entre servicios puedes producirse cuando un servicio (el servicio que lleva a cabo las llamadas) llama a otro servicio (el servicio al que se llama). El servicio que lleva a cabo las llamadas se puedes manipular para utilizar sus permisos a fin de actuar en función de los recursos de otro cliente de una manera en la que no debe tener permiso para acceder. Para evitarlo, AWS proporciona herramientas que lo ayudan a proteger sus datos para todos los servicios con entidades principales de servicio a las que se les ha dado acceso a los recursos de su cuenta.

Le recomendamos que utilice las claves de contexto de condición global `aws:SourceArn` y `aws:SourceAccount` en las políticas de recursos a fin de limitar los permisos que Automation le concede a otro servicio para el recurso. Si el valor de `aws:SourceArn` no contiene el ID de cuenta, como un ARN de bucket de Amazon S3, debe utilizar ambas claves de contexto de condición global para limitar los permisos. Si utiliza claves de contexto de condición global y el valor de `aws:SourceArn` contiene el ID de cuenta, el valor de `aws:SourceAccount` y la cuenta en el valor de `aws:SourceArn` deben utilizar el mismo ID de cuenta cuando se utiliza en la misma instrucción de política. Utiliza `aws:SourceArn` si desea que solo se asocie un recurso al acceso entre servicios. Utiliza `aws:SourceAccount` si quiere permitir que cualquier recurso de esa cuenta se asocie al uso entre servicios. El valor de `aws:SourceArn` debe ser el ARN para las ejecuciones de automatización. Si no conoce el ARN completo del recurso o si especifica varios recursos, utilice la clave de condición de contexto global `aws:SourceArn` con comodines (*) para las partes desconocidas del ARN. Por ejemplo, `arn:aws:ssm:*:123456789012:automation-execution/*`.

En el ejemplo siguiente, se muestra cómo se pueden utilizar las claves de contexto de condición global `aws:SourceArn` y `aws:SourceAccount` para Automation para evitar el problema del suplente confuso.

JSON

```
{
  "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/*"
        }
      }
    }
  ]
}
```

Para modificar una política de confianza de rol

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. Seleccione Roles en el panel de navegación.
3. En la lista de roles de su cuenta, elija el nombre del rol de servicio de Automation.
4. Elija la pestaña Relaciones de confianza y, a continuación, Editar relación de confianza.
5. Edite la política de confianza mediante las claves de contexto de condición global `aws:SourceArn` y `aws:SourceAccount` para Automation a fin de evitar el problema de la sustitución confusa.
6. Para guardar los cambios, elija Update Trust Policy (Actualizar política de confianza).

(Opcional) Agregar una política insertada de Automatización o una política administrada por el cliente para invocar otros Servicios de AWS

Si ejecuta una automatización que invoca otros Servicios de AWS mediante un rol de servicio de IAM, el rol de servicio debe configurarse con el permiso necesario para invocar dichos servicios. Este requisito se aplica a todos los manuales de procedimientos de Automatización de AWS (manuales de `AWS-*`), como los manuales de procedimientos `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup` y `AWS-RestartEC2Instance`, por nombrar algunos. Este requisito también se aplica a cualquier manual de procedimientos personalizado que cree para invocar

otros Servicios de AWS mediante acciones que llaman a otros servicios. Por ejemplo, si utiliza las acciones `aws:executeAwsApi`, `aws:CreateStack` o `aws:copyImage`, entre otras, debe configurar el rol de servicio con el permiso necesario para invocar dichos servicios. Puede conceder permisos a otros Servicios de AWS mediante la incorporación de una política insertada de IAM o política administrada por el cliente, al rol.

Para integrar una política insertada de un rol de servicio (consola de IAM)

1. Inicie sesión en AWS Management Console y abra la consola IAM en https://console.aws.amazon.com/iam/.
2. Seleccione Roles en el panel de navegación.
3. En la lista, seleccione el nombre del rol que desee editar.
4. Elija la pestaña Permisos.
5. En la lista desplegable para Agregar permisos, elija Asociar políticas o Crear política insertada.
6. Si elige Asociar políticas, active la casilla de verificación situada al lado de la política que desea agregar y elija Agregar permisos.
7. Si elige Crear política insertada, elija la pestaña JSON.
8. Ingrese un documento de política JSON para los Servicios de AWS que desee invocar. A continuación, se muestran dos documentos de política JSON a modo de ejemplo.

   Ejemplo de PutObject y GetObject de Amazon S3

   JSON

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

Ejemplo de CreateSnapshot y DescribeSnapShots de Amazon EC2

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"ec2:CreateSnapshot",
         "Resource":"*"
      },
      {
         "Effect":"Allow",
         "Action":"ec2:DescribeSnapshots",
         "Resource":"*"
      }
   ]
}
```

Para obtener información acerca del lenguaje de la política de IAM, consulte Referencia de políticas JSON de IAM en la Guía del usuario de IAM.

9. Cuando haya terminado, elija Review policy (Revisar política). El validador de políticas notifica los errores de sintaxis.
10. En la página Review policy (Revisar política), complete el campo Name (Nombre) de la política que está creando. Revise el Summary (Resumen) de la política para ver los permisos concedidos por su política. A continuación, elija Create policy (Crear política) para guardar su trabajo.
11. Una vez que cree una política insertada, se integra de manera automática a su rol.

Tarea 2: asociar la política iam:PassRole al rol de Automation

Siga este procedimiento para asociar la política `iam:PassRole` al rol de servicio de Automation. Esto permite al servicio Automatización transferir el rol a otros servicios o herramientas de Systems Manager a la hora de ejecutar las automatizaciones.

Para asociar la política iam:PassRole al rol de Automation

1. En la página Summary del rol que acaba de crear, elija la pestaña Permissions.
2. Elija Agregar política insertada.
3. En la página Create policy (Crear política), elija la pestaña Visual editor (Editor visual).
4. Elija Service (Servicio) y, a continuación, IAM.
5. Elija Select actions (Seleccionar acciones).
6. En el cuadro de texto Filter actions (Filtrar acciones), escriba **`PassRole`** y, a continuación, elija la opción PassRole.
7. Seleccione Recursos. Compruebe que esté seleccionado Specific (Específicos) y elija Add ARN (Añadir ARN).
8. En el campo Specify ARN for role (Especificar el ARN del rol), pegue el ARN del rol de Automation que copió al final de la tarea 1. El sistema rellena los campos Account (Cuenta) y Role name with path (Nombre del rol con ruta).

   > **Note**
   >
   > Si desea que el rol de servicio de Automation adjunte un rol de perfil de instancia de IAM a una instancia de EC2, debe agregar el ARN del rol de perfil de instancia de IAM. Esto permite que el rol de servicio de Automation transfiera el rol de perfil de instancia de IAM a la instancia de EC2 de destino.

9. Elija Add (Agregar).
10. Elija Review policy (Revisar política).
11. En la página Review Policy (Revisar política), ingrese un nombre y, a continuación, elija Create policy (Crear la política).

Permitir que Automation se adapte a sus necesidades de simultaneidad

De forma predeterminada, Automation le permite ejecutar hasta 100 automatizaciones simultáneas a la vez. Automation también proporciona una configuración opcional que puede utilizar para ajustar automáticamente la cuota de automatización de la simultaneidad. Con esta configuración, la cuota de automatización de la simultaneidad puede acomodar hasta 500 automatizaciones simultáneas, según los recursos disponibles.

> **Note**
>
> Si su automatización llama a operaciones de API, el escalado adaptativo a sus destinos puede dar lugar a excepciones de limitación. Si se producen excepciones de limitación periódicas cuando se ejecutan automatizaciones con la simultaneidad adaptativa activada, es posible que deba solicitar aumentos de cuota para la operación de la API si esta opción está disponible.

Para activar la simultaneidad adaptativa con la AWS Management Console

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Automation.
3. Elija la pestaña Preferences (Preferencias) y, a continuación, Edit (Editar).
4. Marque la casilla situada junto a Enable adaptive concurrency (Activar la simultaneidad adaptativa).
5. Seleccione Guardar.

Para activar la simultaneidad adaptativa con la línea de comandos

- Abra la AWS CLI o Tools para Windows PowerShell y ejecute el siguiente comando para activar la simultaneidad adaptativa en su cuenta en la región solicitante.

  Linux & macOS

  ```
  aws ssm update-service-setting \
      --setting-id /ssm/automation/enable-adaptive-concurrency \
      --setting-value True
  ```

  Windows

  ```
  aws ssm update-service-setting ^
      --setting-id /ssm/automation/enable-adaptive-concurrency ^
      --setting-value True
  ```

PowerShell

```
Update-SSMServiceSetting `
    -SettingId "/ssm/automation/enable-adaptive-concurrency" `
    -SettingValue "True"
```

Implementación de controles de cambio para Automatización

De forma predeterminada, Automatización le permite utilizar manuales de procedimiento sin restricciones de fecha y hora. Al integrar Automatización con Change Calendar, puede implementar controles de cambios en todas las automatizaciones de su Cuenta de AWS. Con esta configuración, las entidades principales de AWS Identity and Access Management (IAM) de su cuenta solo pueden ejecutar automatizaciones durante los periodos que su calendario de cambios permite. Para obtener más información sobre el uso de Change Calendar, consulte Uso de Change Calendar.

Para activar los controles de cambios (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Automation.
3. Elija la pestaña Preferences (Preferencias) y, a continuación, Edit (Editar).
4. Seleccione la casilla de verificación situada junto a Activar la integración de Change Calendar.
5. En la lista desplegable Elegir un calendario de cambios, seleccione el calendario de cambios que desee que siga Automatización.
6. Seleccione Guardar.

Ejecución de una operación automatizada con la tecnología de automatización de Systems Manager

De forma predeterminada, cuando se ejecuta una automatización, se hace en el contexto del usuario que inicia la automatización. Esto significa que, por ejemplo, si el usuario tiene permisos de administrador, la automatización se ejecuta con permisos de administrador y tiene acceso pleno a los recursos que configura la automatización. Como práctica recomendada de seguridad, le recomendamos que ejecute la automatización con un rol de servicio de IAM, que también se conoce como un rol de asunción, configurado con la política administrada AmazonSSMAutomationRole.

Es posible que tenga que agregar políticas de IAM adicionales al rol asumido para usar diferentes manuales de procedimientos. El uso de un rol de servicio de IAM para ejecutar la automatización se denomina administración delegada.

Cuando se utiliza un rol de servicio, se puede ejecutar la automatización en los recursos de AWS, pero el usuario que ejecutó la automatización tiene el acceso restringido a dichos recursos (o no puede acceder a ellos). Por ejemplo, puede configurar un rol de servicio y utilizarlo con Automation para reiniciar una o más instancias de Amazon Elastic Compute Cloud (Amazon EC2). Automatización es una herramienta de AWS Systems Manager. La automatización reinicia las instancias, pero el rol de servicio no concede al usuario permiso para acceder a dichas instancias.

Puede especificar un rol de servicio en el tiempo de ejecución de una automatización, o bien, puede crear manuales de procedimientos personalizados y especificar el rol de servicio directamente en el manual. Si especifica un rol de servicio, ya sea en el tiempo de ejecución o en un manual de procedimientos, el servicio se ejecuta en el contexto del rol de servicio especificado. Si no especifica un rol de servicio, el sistema crea una sesión temporal en el contexto del usuario y ejecuta la automatización.

> **Note**
>
> Debe especificar un rol de servicio para las automatizaciones que espera que se ejecuten durante más de 12 horas. Si inicia una automatización cuya ejecución tarda mucho tiempo en el contexto de un usuario, la sesión temporal del usuario caduca después de 12 horas.

La administración delegada garantiza mayor control y seguridad de los recursos de AWS. También permite tener una mejor experiencia en las auditorías, ya que las acciones se realizan sobre los recursos a través de un rol de servicio centralizado en lugar de varias cuentas de IAM.

Antes de empezar

Antes de completar los siguientes procedimientos, cree el rol de servicio de IAM y configure una relación de confianza para Automatización, una herramienta de AWS Systems Manager. Para obtener más información, consulte Tarea 1: crear un rol de servicio para Automation.

Los siguientes procedimientos describes cómo utilizar la consola de Systems Manager o su herramienta de línea de comandos preferida para ejecutar una automatización sencilla.

Ejecución de una automatización sencilla (consola)

El siguiente procedimiento describe cómo utilizar la consola de Systems Manager para ejecutar una automatización sencilla.

Para ejecutar una automatización sencilla

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Automatización y, después, seleccione Ejecutar automatización.
3. En la lista Documento de automatización, elija un manual de procedimientos. Elija una o más opciones en el panel Categorías de documentos para filtrar documentos SSM según su propósito. Para ver un manual de procedimientos que le pertenezca, seleccione la pestaña De mi propiedad. Para ver un manual de procedimientos que se haya compartido con su cuenta, elija la pestaña Compartido conmigo. Para ver todos los manuales de procedimientos, seleccione la pestaña Todos los documentos.

   > **Note**
   >
   > Puede ver información acerca de un manual de procedimientos al seleccionar su nombre.

4. En la sección Detalles del documento, verifique que Versión del documento esté establecido como la versión que desea ejecutar. El sistema incluye las siguientes opciones de versión:

   - Versión predeterminada en tiempo de ejecución: seleccione esta opción si el manual de procedimientos de automatización se actualiza de forma periódica y se asigna una nueva versión predeterminada.
   - Última versión en tiempo de ejecución: seleccione esta opción si el manual de procedimientos de automatización se actualiza de forma periódica y desea ejecutar la versión que se ha actualizado más recientemente.
   - 1 (Predeterminado): seleccione esta opción para ejecutar la primera versión del documento, que es la predeterminada.
5. Elija Siguiente.
6. En la sección Modo de ejecución, seleccione Ejecución sencilla.
7. En la sección Parámetros de entrada, especifique las entradas necesarias: De forma opcional, puede elegir un rol de servicio de IAM de la lista AutomationAssumeRole.

8. (Opcional) Elija una alarma de CloudWatch para aplicarla a la automatización de monitoreo. Para adjuntar una alarma de CloudWatch a su automatización, la entidad principal de IAM que ejecuta esta última debe tener permiso para la acción `iam:createServiceLinkedRole`. Para obtener más información sobre las alarmas de CloudWatch, consulte Uso de alarmas de Amazon CloudWatch. Tenga en cuenta que si la alarma se activa, la automatización se detiene. Si usa AWS CloudTrail, verá la llamada a la API en el registro de seguimiento.
9. Elija Ejecutar.

La consola muestra el estado de la automatización. Si no se logra ejecutar la automatización, consulte Solución de problemas de Automatización de Systems Manager.

Ejecución de una automatización sencilla (línea de comandos)

El siguiente procedimiento describe cómo utilizar la AWS CLI (en Linux o Windows) o las Herramientas de AWS para PowerShell para ejecutar una automatización sencilla.

Para ejecutar una automatización sencilla

1. Si aún no lo ha hecho, instale y configure la AWS CLI o las Herramientas de AWS para PowerShell.

   Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.
2. Ejecute el siguiente comando para iniciar una automatización sencilla. Reemplace cada *example resource placeholder* con su propia información.

   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
```

A continuación, se muestra un ejemplo en el que se utiliza el manual de procedimientos `AWS-RestartEC2Instance` para reiniciar la instancia de EC2 especificada.

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"}
```

El sistema devuelve información similar a la siguiente.

Linux & macOS

```
{
    "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0123456789ab"
}
```

Windows

```
{
```

```
    "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0123456789ab"
}
```

PowerShell

```
4105a4fc-f944-11e6-9d32-0123456789ab
```

3. Ejecute el siguiente comando para recuperar el estado de la automatización.

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"}
```

El sistema devuelve información similar a la siguiente.

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                     : {}
```

Ejecución de una automatización que requiere aprobaciones

Los siguientes procedimientos describen cómo utilizar la consola de AWS Systems Manager y la AWS Command Line Interface (AWS CLI) para ejecutar una automatización con aprobaciones mediante una ejecución sencilla. La automatización utiliza la acción de automatización `aws:approve`, lo que detiene de forma temporal la automatización hasta que las entidades principales designadas aprueben o denieguen la acción. La automatización se ejecuta en el contexto del usuario actual. Esto significa que no tiene que configurar más permisos de IAM siempre y cuando cuente con el permiso necesario para usar el manual de procedimientos y cualquier acción que este solicite. Si tiene permisos de administrador en IAM, ya cuenta con el permiso necesario para usar este manual de procedimientos.

Antes de empezar

Además de las entradas estándares requeridas por el manual de procedimientos, la acción `aws:approve` necesita los siguientes dos parámetros:

- Una lista de aprobadores. La lista de aprobadores debe contener al menos un aprobador en forma de nombre de usuario o ARN de usuario. Si se proporcionan varios aprobadores, también debe especificarse el recuento de aprobaciones mínimo correspondiente en el manual de procedimientos.
- El ARN de un tema de Amazon Simple Notification Service (Amazon SNS). El nombre del tema de Amazon SNS debe empezar con `Automation`.

En este procedimiento, se da por hecho que ya ha creado un tema de Amazon SNS, lo que es necesario para entregar una solicitud de aprobación. Para obtener información, consulte Creación de un tema en la Guía para desarrolladores de Amazon Simple Notification Service.

Ejecución de una automatización con aprobadores (consola)

Para ejecutar una automatización con aprobadores

El siguiente procedimiento describe cómo utilizar la consola de Systems Manager para ejecutar una automatización con aprobadores.

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Automatización y, después, seleccione Ejecutar automatización.
3. En la lista Documento de automatización, elija un manual de procedimientos. Elija una o más opciones en el panel Categorías de documentos para filtrar documentos SSM según su propósito. Para ver un manual de procedimientos que le pertenezca, seleccione la pestaña De mi propiedad. Para ver un manual de procedimientos que se haya compartido con su cuenta, elija la pestaña Compartido conmigo. Para ver todos los manuales de procedimientos, seleccione la pestaña Todos los documentos.

   > **Note**
   >
   > Puede ver información acerca de un manual de procedimientos al seleccionar su nombre.

4. En la sección Detalles del documento, verifique que Versión del documento esté establecido como la versión que desea ejecutar. El sistema incluye las siguientes opciones de versión:

- Versión predeterminada en tiempo de ejecución: seleccione esta opción si el manual de procedimientos de automatización se actualiza de forma periódica y se asigna una nueva versión predeterminada.
- Última versión en tiempo de ejecución: seleccione esta opción si el manual de procedimientos de automatización se actualiza de forma periódica y desea ejecutar la versión que se ha actualizado más recientemente.
- 1 (Predeterminado): seleccione esta opción para ejecutar la primera versión del documento, que es la predeterminada.

5. Elija Siguiente.
6. En la página Ejecutar documento de automatización, seleccione Ejecución simple.
7. En la sección Parámetros de entrada, especifique los parámetros de entrada necesarios.

   Por ejemplo, si selecciona el manual de procedimientos **`AWS-StartEC2InstanceWithApproval`**, debe especificar o seleccionar los ID de instancias para el parámetro InstanceId.
8. En la sección Aprobadores, especifique los nombres de usuario o los ARN de usuario de los aprobadores para la acción de automatización.
9. En la sección SNSTopicARN, especifique el ARN de tema de SNS que utilizará para enviar la notificación de aprobación. El nombre de tema de SNS debe empezar por Automation.
10. De forma opcional, puede elegir un rol de servicio de IAM de la lista AutomationAssumeRole. Si se indican más de 100 cuentas y regiones, debe especificar el `AWS-SystemsManager-AutomationAdministrationRole`.
11. Elija Ejecutar automatización.

El aprobador especificado recibe una notificación de Amazon SNS con detalles para aprobar o rechazar la automatización. Esta acción de aprobación es válida durante 7 días a partir de la fecha de emisión y puede emitirse a través de la consola de Systems Manager o la AWS Command Line Interface (AWS CLI).

Si decide aprobar la automatización, esta sigue ejecutando los pasos incluidos en el manual de procedimientos especificado. La consola muestra el estado de la automatización. Si no se logra ejecutar la automatización, consulte Solución de problemas de Automatización de Systems Manager.

Para aprobar o denegar una automatización

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Automation y, a continuación, seleccione la automatización que se ejecutó en el procedimiento anterior.
3. Seleccione Acciones y, a continuación, Aprobar/denegar.
4. Seleccione Aprobar o Denegar y, si lo desea, proporcione un comentario.
5. Elija Enviar.

Ejecución de una automatización con aprobadores (línea de comandos)

El siguiente procedimiento describe cómo utilizar la AWS CLI (en Linux o Windows) o las Herramientas de AWS para PowerShell para ejecutar una automatización con aprobadores.

Para ejecutar una automatización con aprobadores

1. Si aún no lo ha hecho, instale y configure la AWS CLI o las Herramientas de AWS para PowerShell.

   Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.
2. Utilice el siguiente comando para ejecutar una automatización con aprobadores. Reemplace cada *example resource placeholder* con su propia información. En la sección del nombre del documento, especifique un manual de procedimientos que incluya la acción de automatización `aws:approve`.

   En `Approvers`, especifique los nombres de usuario o los ARN de usuario de los aprobadores para la acción. En `SNSTopic`, especifique el ARN de tema de SNS que desea utilizar para enviar la notificación de aprobación. El nombre del tema de Amazon SNS debe empezar con `Automation`.

   > **Note**
   >
   > Los nombres específicos de los valores de los parámetros para los aprobadores y el tema de SNS dependen de los valores especificados en el manual de procedimientos que elija.

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"
    }
```

El sistema devuelve información similar a la siguiente.

Linux & macOS

```
{
    "AutomationExecutionId": "df325c6d-b1b1-4aa0-8003-6cb7338213c6"
}
```

Windows

```
{
    "AutomationExecutionId": "df325c6d-b1b1-4aa0-8003-6cb7338213c6"
```

```
}
```

PowerShell

```
df325c6d-b1b1-4aa0-8003-6cb7338213c6
```

Para aprobar una automatización

- Ejecute el siguiente comando para aprobar una automatización. Reemplace cada *example resource placeholder* con su propia información.

  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"}
  ```

  No se obtienen resultados si el comando se ejecuta correctamente.

Para denegar una automatización

- Ejecute el siguiente comando para denegar una automatización. Reemplace cada *example resource placeholder* con su propia información.

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"}
```

No se obtienen resultados si el comando se ejecuta correctamente.

Ejecución de operaciones automatizadas a escala

Con Automatización de AWS Systems Manager, puede ejecutar automatizaciones en una flota de recursos de AWS mediante el uso de destinos. Además, puede controlar la implementación de la automatización en su flota al especificar un valor de simultaneidad y un umbral de error. Las características de umbral de simultaneidad y error se denominan colectivamente controles de frecuencia. El valor de simultaneidad determina la cantidad de recursos que pueden ejecutar la automatización de forma simultánea. Automation también proporciona un modo de simultaneidad adaptativa que puede elegir. La simultaneidad adaptativa escala automáticamente su cuota de automatización desde 100 automatizaciones que se ejecutan simultáneamente hasta 500. El umbral de error determina la cantidad de automatizaciones que pueden fallar antes de que Systems Manager deje de enviar la automatización a otros recursos.

Para obtener más información acerca de la simultaneidad y los umbrales de error, consulte Control de las automatizaciones a escala. Para obtener más información sobre los destinos, consulte Asignación de objetivos de una automatización.

Los siguientes procedimientos muestran cómo activar una simultaneidad adaptativa y cómo ejecutar una automatización con controles de frecuencia y destinos a través de la consola de Systems Manager y la AWS Command Line Interface (AWS CLI).

Ejecución de una automatización con controles de frecuencia y destinos (consola)

El siguiente procedimiento describe cómo utilizar la consola de Systems Manager para ejecutar una automatización con controles de frecuencia y destinos.

Para ejecutar una automatización con controles de frecuencia y destinos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Automatización y, después, seleccione Ejecutar automatización.
3. En la lista Documento de automatización, elija un manual de procedimientos. Elija una o más opciones en el panel Categorías de documentos para filtrar documentos SSM según su propósito. Para ver un manual de procedimientos que le pertenezca, seleccione la pestaña De mi propiedad. Para ver un manual de procedimientos que se haya compartido con su cuenta, elija la pestaña Compartido conmigo. Para ver todos los manuales de procedimientos, seleccione la pestaña Todos los documentos.

   > **Note**
   >
   > Puede ver información acerca de un manual de procedimientos al seleccionar su nombre.

4. En la sección Detalles del documento, verifique que Versión del documento esté establecido como la versión que desea ejecutar. El sistema incluye las siguientes opciones de versión:

   - Versión predeterminada en tiempo de ejecución: seleccione esta opción si el manual de procedimientos de automatización se actualiza de forma periódica y se asigna una nueva versión predeterminada.
   - Última versión en tiempo de ejecución: seleccione esta opción si el manual de procedimientos de automatización se actualiza de forma periódica y desea ejecutar la versión que se ha actualizado más recientemente.

- 1 (Predeterminado): seleccione esta opción para ejecutar la primera versión del documento, que es la predeterminada.

5. Elija Siguiente.
6. En la sección Modo de ejecución, seleccione Control de velocidad. Debe utilizar este modo o Varias cuentas y regiones si desea utilizar controles de frecuencia y destinos.
7. En la sección Destinos, seleccione cómo quiere que se indiquen los recursos de AWS donde desea ejecutar la automatización como destino. Estas opciones son obligatorias.

   a. Use la lista Parámetro para elegir un parámetro. Los elementos de la lista Parámetro se determinan a partir de los parámetros en el manual de procedimientos de automatización que seleccionó al inicio de este procedimiento. Al elegir un parámetro, se define el tipo de recurso en el que se ejecuta el flujo de trabajo de automatización.

   b. Utilice la lista Destinos para elegir cómo indicar los recursos de destino.

      i. Si eligió indicar recursos como destino mediante el uso de valores de parámetro, ingrese el valor del parámetro que eligió en la sección Parámetros de entrada.

      ii. Si eligió los recursos de destino con AWS Resource Groups, entonces elija el nombre del grupo de la lista Grupo de recursos.

      iii. Si eligió indicar recursos de destino mediante el uso de etiquetas, introduzca la clave de etiqueta y, opcionalmente, un valor de etiqueta en los campos correspondientes. Elija Agregar.

      iv. Si desea ejecutar un manual de procedimientos de automatización en todas las instancias de la Cuenta de AWS y la Región de AWS actuales, seleccione Todas las instancias.

8. En la sección Parámetros de entrada, especifique las entradas necesarias: De forma opcional, puede elegir un rol de servicio de IAM de la lista AutomationAssumeRole.

> **Note**
>
> Es posible que no tenga que elegir algunas de las opciones de la sección Input parameters (Parámetros de entrada). Esto se debe a que ha indicado recursos como destino a través de etiquetas o un grupo de recursos. Por ejemplo, si eligió el manual de procedimientos `AWS-RestartEC2Instance`, no necesita especificar ni elegir los ID de instancia en la sección Parámetros de entrada. La ejecución de Automation localiza las

instancias que se deben reiniciar a través de las etiquetas o los grupos de recursos que haya especificado.

9. Utilice las opciones de la sección Control de velocidad para restringir el número de recursos de AWS que pueden ejecutar la automatización dentro de cada par de cuenta-región.

   En la sección Simultaneidad, elija una opción:

   - Seleccione destinos para introducir un número absoluto de destinos que pueden ejecutar el flujo de trabajo de Automation simultáneamente.
   - Seleccione porcentaje para introducir un porcentaje del destino definido que puede ejecutar el flujo de trabajo de Automation simultáneamente.

10. En la sección Umbral de error, elija una opción:

    - Elija errores para introducir un número absoluto de errores permitidos antes de que Automation deje de enviar el flujo de trabajo a otros recursos.
    - Elija porcentaje para introducir un porcentaje de errores permitidos antes de que Automation deje de enviar el flujo de trabajo a otros recursos.

11. (Opcional) Elija una alarma de CloudWatch que desee aplicar a la automatización para fines de monitoreo. Para adjuntar una alarma de CloudWatch a su automatización, la entidad principal de IAM que ejecuta esta última debe tener permiso para la acción `iam:createServiceLinkedRole`. Para obtener más información sobre las alarmas de CloudWatch, consulte Uso de alarmas de Amazon CloudWatch. Tenga en cuenta que si la alarma se activa, la automatización se detiene. Si usa AWS CloudTrail, verá la llamada a la API en el registro de seguimiento.

12. Elija Ejecutar.

Para ver las automatizaciones que inició la automatización de control de frecuencia, en el panel de navegación, seleccione Automation y, a continuación, elija Mostrar automatizaciones secundarias.

Ejecución de una automatización con controles de frecuencia y destinos (línea de comandos)

El siguiente procedimiento describe cómo utilizar la AWS CLI (en Linux o Windows) o las Herramientas de AWS para PowerShell para ejecutar una automatización con controles de frecuencia y destinos.

Para ejecutar una automatización con controles de frecuencia y destinos

1. Si aún no lo ha hecho, instale y configure la AWS CLI o las Herramientas de AWS para PowerShell.

   Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.

2. Ejecute el siguiente comando para ver una lista de documentos.

   Linux & macOS

   ```
   aws ssm list-documents
   ```

   Windows

   ```
   aws ssm list-documents
   ```

   PowerShell

   ```
   Get-SSMDocumentList
   ```

   Anote el nombre del manual de procedimientos que desea usar.

3. Ejecute el siguiente comando para ver detalles acerca del manual de procedimientos. Reemplace *nombre del runbook* con el nombre del manual de procedimientos cuyos detalles desee ver. Además, indique el nombre de parámetro (por ejemplo, `InstanceId`) que desee utilizar para la opción `--target-parameter-name`. Este parámetro determina el tipo de recurso en el que se ejecuta la automatización.

   Linux & macOS

   ```
   aws ssm describe-document \
       --name runbook name
   ```

   Windows

   ```
   aws ssm describe-document ^
       --name runbook name
   ```

PowerShell

```
Get-SSMDocumentDescription `
    -Name runbook name
```

4. Cree un comando que utilice las opciones de control de frecuencia y destinos que desee ejecutar. Reemplace cada *example resource placeholder* con su propia información.

   Indicar destino mediante etiquetas

   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%"
```

Indicar destino mediante valores de parámetros

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%"
```

Indicar destino mediante 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%"
```

Indicar todas las instancias de Amazon EC2 en la Cuenta de AWS y la Región de AWS actuales

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%"
```

El comando devuelve un ID de ejecución. Copie este ID en el portapapeles. Puede utilizar este ID para ver el estado de la automatización.

Linux & macOS

```
{
```

```
    "AutomationExecutionId": "a4a3c0e9-7efd-462a-8594-01234EXAMPLE"
}
```

Windows

```
{
    "AutomationExecutionId": "a4a3c0e9-7efd-462a-8594-01234EXAMPLE"
}
```

PowerShell

```
a4a3c0e9-7efd-462a-8594-01234EXAMPLE
```

5. Ejecute el siguiente comando para ver la automatización. Reemplace cada *automation execution ID* con su propia información.

   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. Ejecute el siguiente comando para ver detalles acerca del progreso de la automatización. Reemplace cada *automation execution ID* con su propia información.

   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
```

El sistema devuelve información similar a la siguiente.

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**
>
> También puede monitorear el estado de la automatización en la consola. En la lista Ejecuciones de automatización, elija la automatización que acaba de ejecutar y, a continuación, seleccione la pestaña Execution steps. Esta pestaña muestra el estado de las acciones de la automatización.

Asignación de objetivos de una automatización

Utilice el parámetro `Targets` para establecer rápidamente cuáles son los recursos a los que se dirige una automatización. Por ejemplo, si desea ejecutar una automatización que reinicie las instancias administradas, en lugar de seleccionar manualmente decenas de ID de instancias en la consola o escribirlos en un comando, puede indicar las instancias como destino mediante la especificación de etiquetas de Amazon Elastic Compute Cloud (Amazon EC2) con el parámetro `Targets`.

Al ejecutar una automatización que utiliza un destino, AWS Systems Manager crea una automatización secundaria para cada destino. Por ejemplo, si indica volúmenes de Amazon Elastic Block Store (Amazon EBS) como destino mediante la especificación de etiquetas, y dichas etiquetas llevan a 100 volúmenes de Amazon EBS, Systems Manager crea 100 automatizaciones secundarias. La automatización principal se completa cuando todas las automatizaciones secundarias alcanzan un estado final.

> **Note**
>
> Todas las automatizaciones secundarias procesan de manera automática cada uno de los `input parameters` que especifique en el tiempo de ejecución (ya sea en la sección Input parameters [Parámetros de entrada] de la consola o a través de la opción `parameters` de la línea de comandos).

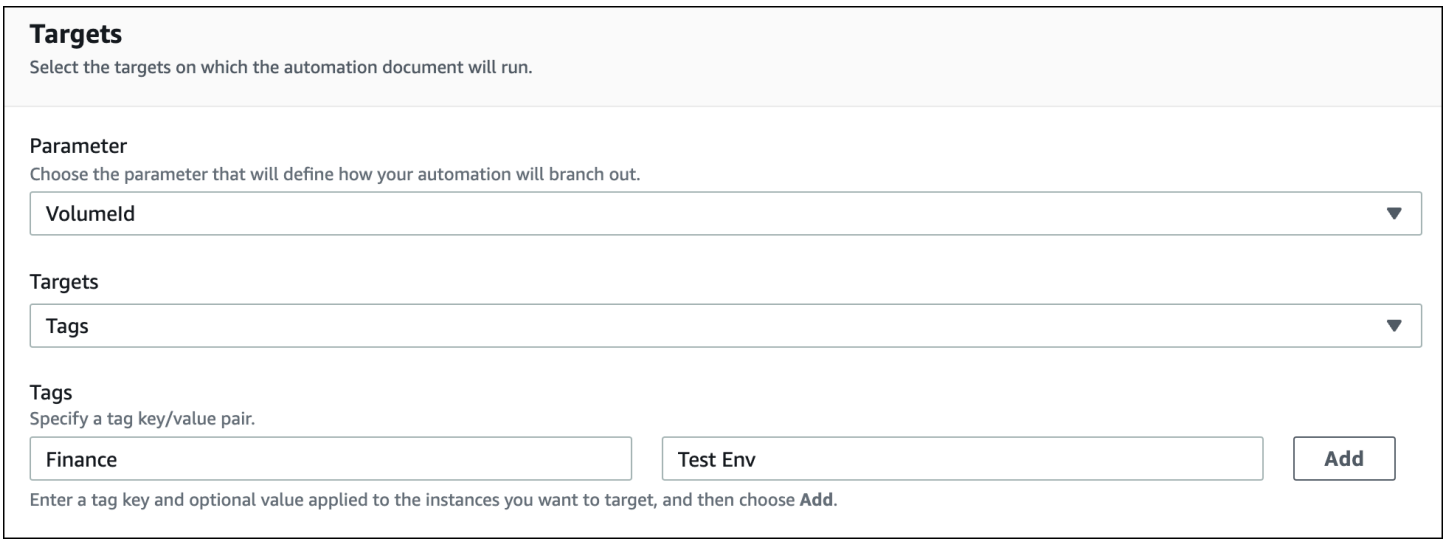
Puede indicar recursos como destino para una automatización a través de etiquetas, grupos de recursos y valores de parámetros. Además, puede utilizar la opción `TargetMaps` para indicar varios valores de parámetros de destino desde la línea de comandos o un archivo. En la siguiente sección se describe cada una de estas opciones de destino de forma más detallada.

Especificación de una etiqueta como destino

Puede especificar una sola etiqueta como destino de una automatización. Muchos recursos de AWS admiten etiquetas, incluidos las instancias de Amazon Elastic Compute Cloud (Amazon EC2) y Amazon Relational Database Service (Amazon RDS), los volúmenes y las instantáneas de Amazon Elastic Block Store (Amazon EBS), los grupos de recursos y los buckets de Amazon Simple Storage Service (Amazon S3), por nombrar algunos. Puede ejecutar rápidamente una automatización en los recursos de AWS especificando una etiqueta como destino. Una etiqueta es un par clave-valor, como Sistema_operativo:Linux o Departamento:Finanzas. Si asigna un nombre específico a un recurso, entonces también puede utilizar la palabra "Name" como una clave y el nombre del recurso como el valor.

Cuando se especifica una etiqueta como destino de una automatización, también especifica un parámetro de destino. El parámetro de destino utiliza la opción `TargetParameterName`. Al elegir un parámetro de destino, define el tipo de recurso en el que se ejecuta la automatización. El parámetro de destino que especifique con la etiqueta tiene que ser un parámetro válido definido en el manual de procedimientos. Por ejemplo, si desea indicar docenas de instancias EC2 de destino mediante el uso de etiquetas, elija el parámetro de destino `InstanceId`. Al elegir este parámetro, define las instancias como el tipo de recurso para la automatización. Al crear un manual de procedimientos personalizado, debe especificar Tipo de destino como `/AWS::EC2::Instance` para asegurarse de que solo se utilicen instancias. De lo contrario, se seleccionarán todos los recursos con la misma etiqueta. Al especificar instancias como destino con una etiqueta, es posible que se incluyan instancias finalizadas.

En la siguiente captura de pantalla, se utiliza el manual de procedimientos `AWS-DetachEBSVolume`. El parámetro de destino lógico es `VolumeId`.



El manual de procedimientos `AWS-DetachEBSVolume` también incluye una propiedad especial denominada Target type (Tipo de destino), que se establece en `/AWS::EC2::Volume`. Esto significa que, si el par de etiqueta-clave `Finance:TestEnv` devuelve diferentes tipos de recursos (por ejemplo, instancias EC2, volúmenes de Amazon EBS, instantáneas de Amazon EBS), solo se usarán los volúmenes de Amazon EBS.

Important

Los nombres de los parámetros de destino distinguen entre mayúsculas y minúsculas. Si ejecuta automatizaciones a través de la AWS Command Line Interface (AWS CLI) o las AWS Tools for Windows PowerShell, debe ingresar el nombre de parámetro de destino tal y como está definido en el manual de procedimientos. Si no lo hace, el sistema devuelve un error `InvalidAutomationExecutionParametersException`. Puede utilizar la operación de la API DescribeDocument para ver información sobre los parámetros de destino disponibles en un manual de procedimientos específico. A continuación, se muestra un ejemplo de comando de la AWS CLI que proporciona información acerca del documento `AWS-DeleteSnapshot`.

```
aws ssm describe-document \
    --name AWS-DeleteSnapshot
```

A continuación se muestran algunos ejemplos de comandos de AWS CLI que tienen como destino recursos mediante el uso de una etiqueta.

Ejemplo 1: especificación de una etiqueta como destino utilizando un par clave-valor para reiniciar instancias de Amazon EC2

En este ejemplo, se reinician todas las instancias de Amazon EC2 que estén etiquetadas con una clave Department y un valor HumanResources. El parámetro de destino utiliza el parámetro InstanceId del manual de procedimientos. En el ejemplo, se utiliza un parámetro adicional para ejecutar Automation mediante el uso de un rol de servicio de Automation (también denominado rol de asunción).

```
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"
```

Ejemplo 2: especificación de una etiqueta como destino utilizando un par clave-valor para eliminar instantáneas de Amazon EBS

En el siguiente ejemplo, se utiliza el manual de procedimientos `AWS-DeleteSnapshot` para eliminar todas las instantáneas con una clave Name y un valor January2018Backups. El parámetro de destino usa el parámetro VolumeId.

```
aws ssm start-automation-execution \
    --document-name AWS-DeleteSnapshot \
    --targets Key=tag:Name,Values=January2018Backups \
    --target-parameter-name VolumeId
```

Indicar AWS Resource Groups de destino

Puede indicar un solo grupo de recursos de AWS como destino de una automatización. Systems Manager crea una automatización secundaria para todos los objetos del grupo de recursos de destino.

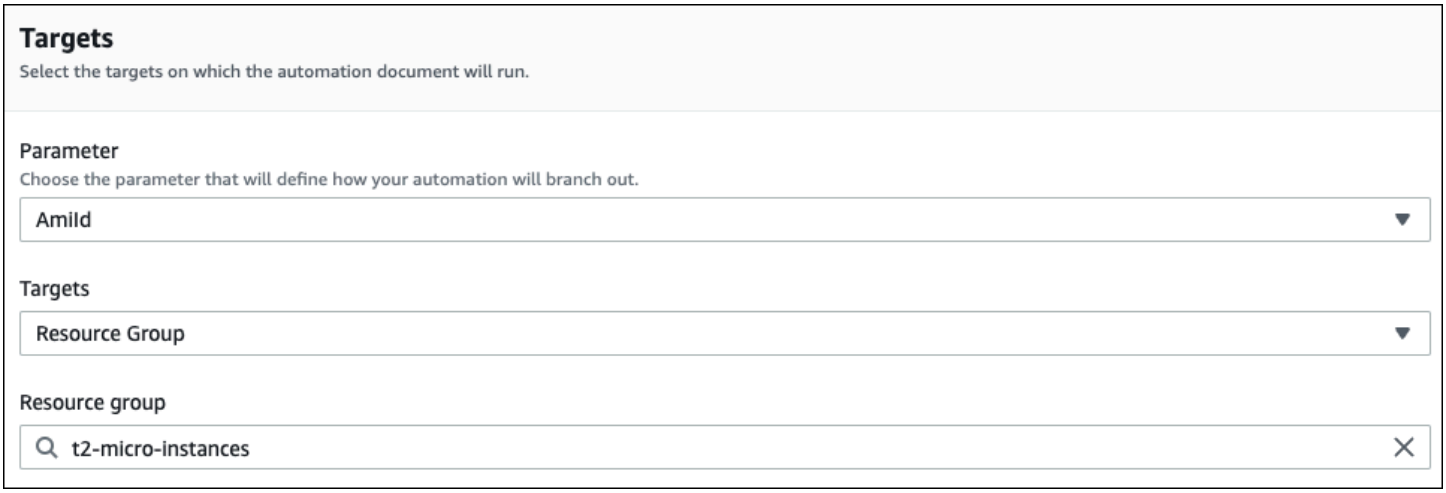
Por ejemplo, supongamos que uno de los grupos de recursos se llama PatchedAMIs. Este grupo de recursos incluye una lista de 25 Amazon Machine Images (AMIs) de Windows a las cuales se aplican

revisiones de forma rutinaria. Si ejecuta una automatización que utiliza el manual de procedimientos `AWS-CreateManagedWindowsInstance` e indica este grupo de recursos como destino, Systems Manager crea una automatización secundaria para cada una de las 25 AMIs. Esto significa que, al indicar el grupo de recursos PatchedAMIs como destino, la automatización creará 25 instancias a partir de una lista de AMIs a las cuales se han aplicado revisiones. La automatización principal se completa cuando todas las automatizaciones secundarias finalizan el procesamiento o alcanzan un estado final.

El siguiente comando de AWS CLI se aplica al ejemplo del grupo de recursos PatchAMIs. El comando adopta el parámetro AmiId para la opción `--target-parameter-name`. El comando no incluye un parámetro adicional que defina el tipo de instancia que se debe crear a partir de cada AMI. El manual de procedimientos `AWS-CreateManagedWindowsInstance` recurre a la opción predeterminada de tipo de instancia t2.medium, por lo que este comando crearía 25 instancias de Amazon EC2 t2.medium para Windows Server.

```
aws ssm start-automation-execution \
    --document-name AWS-CreateManagedWindowsInstance \
    --targets Key=ResourceGroup,Values=PatchedAMIs  \
    --target-parameter-name AmiId
```

En el siguiente ejemplo de consola, se utiliza un grupo de recursos llamado t2-micro-instances.



Indicar valores de parámetros de destino

También puede indicar el valor de un parámetro de destino. Ingresa `ParameterValues` como clave y, a continuación, el valor del recurso específico donde desee que se ejecute la automatización. Si especifica varios valores, Systems Manager ejecuta una automatización secundaria en cada valor especificado.

Por ejemplo, supongamos que su manual de procedimientos incluye un parámetro InstanceID. Si indica los valores del parámetro InstanceID como destino a la hora de ejecutar Automation, Systems Manager ejecuta una automatización secundaria para el valor de ID de cada instancia especificada. La automatización principal se habrá completado cuando la automatización termine de ejecutar cada instancia especificada o cuando se produzca un error en la automatización. Puede indicar un máximo de 50 valores de parámetros de destino.

El siguiente ejemplo utiliza el manual de procedimientos `AWS-CreateImage`. El nombre del parámetro de destino es InstanceId. La clave utiliza ParameterValues. Los valores son dos ID de instancias de Amazon EC2. Este comando crea una automatización para cada instancia, lo cual produce una AMI a partir de cada instancia.

```
aws ssm start-automation-execution
    --document-name AWS-CreateImage \
    --target-parameter-name InstanceId \
    --targets Key=ParameterValues,Values=i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE
```

> **Note**
>
> `AutomationAssumeRole` no es un parámetro válido. No elija este elemento al ejecutar una automatización que tiene como destino el valor de un parámetro.

Indicar mapas de valores de parámetros de destino

La opción `TargetMaps` amplía su capacidad de indicar `ParameterValues` como destino. Puede introducir una matriz de valores de parámetros usando `TargetMaps` en la línea de comandos. Puede especificar un máximo de 50 valores de parámetros en la línea de comandos. Si desea ejecutar comandos que especifiquen más de 50 valores de parámetros, puede introducir los valores en un archivo JSON. A continuación, puede llamar al archivo desde la línea de comandos.

> **Note**
>
> La consola no admite la opción `TargetMaps`.

Utilice el siguiente formato para especificar varios valores de parámetros con la opción `TargetMaps` en un comando. Reemplace cada *`example resource placeholder`* con su propia información.

```
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 desea especificar más de 50 valores de parámetros para la opción `TargetMaps`, especifique los valores en un archivo mediante el siguiente formato JSON. El uso de un archivo JSON también mejora la legibilidad al proporcionar varios valores de parámetros.

```
[

    {“parameter”: "value", “parameter 2”: "value", “parameter 3”: "value"},

    {“parameter 4”: "value", “parameter 5”: "value", "parameter 6": "value"}

]
```

Guarde el archivo con la extensión de archivo .json. Puede llamar al archivo con el siguiente comando. Reemplace cada *example resource placeholder* con su propia información.

```
aws ssm start-automation-execution \
    --document-name runbook name \
    --parameters input parameters \
    --target-maps path to file/file name.json
```

También puede descargar el archivo de un bucket de Amazon Simple Storage Service (Amazon S3), siempre y cuando tenga permiso para leer datos del bucket. Utilice el siguiente formato de comando. Reemplace cada *example resource placeholder* con su propia información.

```
aws ssm start-automation-execution \
    --document-name runbook name \
    --target-maps http://amzn-s3-demo-bucket.s3.amazonaws.com/file_name.json
```

A continuación se muestra un ejemplo de escenario para ayudarle a comprender la opción `TargetMaps`. En este escenario, un usuario quiere crear instancias de Amazon EC2 de diferentes tipos a partir de diferentes AMIs. Para realizar esta tarea, el usuario crea un manual de procedimientos denominado AMI_Testing. Este manual de procedimientos define dos parámetros de entrada: `instanceType` e `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"
  ]
}
```

A continuación, el usuario especifica los siguientes valores de parámetros de destino en un archivo denominado `AMI_instance_types.json`.

```
[
  {
```

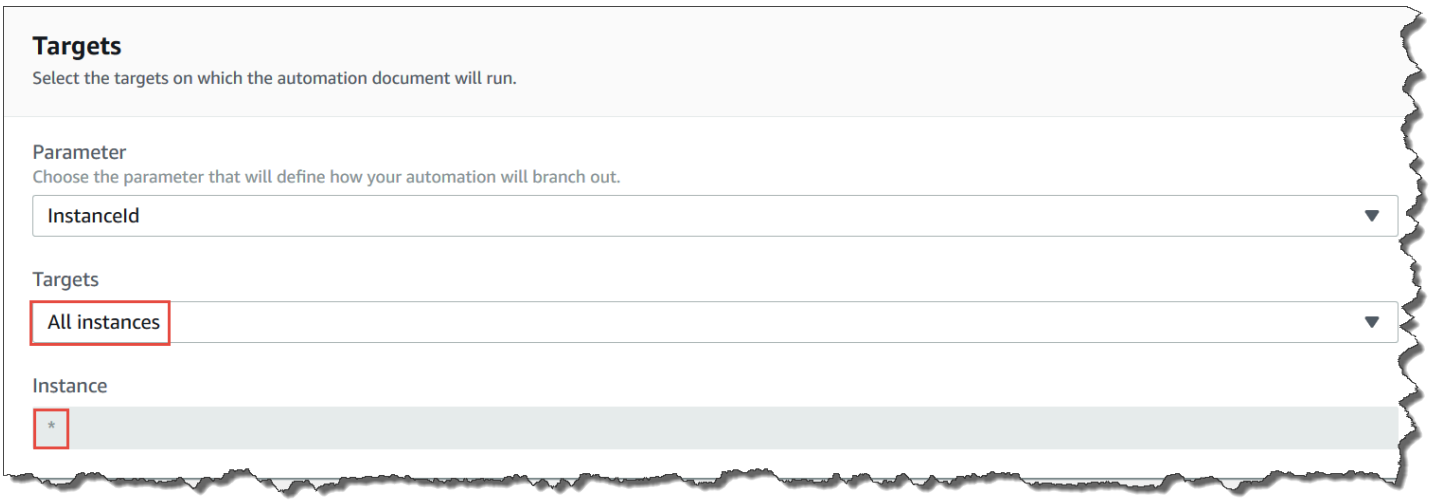
```
    "instanceType" : ["t2.micro"],
    "imageId" : ["ami-b70554c8"]
  },
  {
    "instanceType" : ["t2.small"],
    "imageId" : ["ami-b70554c8"]
  },
  {
    "instanceType" : ["t2.medium"],
    "imageId" : ["ami-cfe4b2b0"]
  },
  {
    "instanceType" : ["t2.medium"],
    "imageId" : ["ami-cfe4b2b0"]
  },
  {
    "instanceType" : ["t2.medium"],
    "imageId" : ["ami-cfe4b2b0"]
  }
]
```

El usuario puede ejecutar la automatización y crear las cinco instancias EC2 definidas en `AMI_instance_types.json` mediante la ejecución del siguiente comando.

```
aws ssm start-automation-execution \
    --document-name AMI_Testing \
    --target-parameter-name imageId \
    --target-maps file:///home/TestUser/workspace/runinstances/AMI_instance_types.json
```

Indicar todas las instancias de Amazon EC2

Puede ejecutar una automatización en todas las instancias de Amazon EC2 en la Cuenta de AWS y Región de AWS actuales si elige Todas las instancias en la lista Destinos. Por ejemplo, si desea reiniciar todas las instancias administradas de Amazon EC2, su Cuenta de AWS y la Región de AWS actual, puede elegir el manual de procedimientos **`AWS-RestartEC2Instance`** y, luego, Todas las instancias en la lista Destinos.



Después de seleccionar All instances (Todas las instancias), Systems Manager completa el campo Instance (Instancia) con un asterisco (*) y lo marca como no disponible para los cambios (el campo aparece atenuado). Systems Manager también hace que el campo InstanceId del campo Input parameters (Parámetros de entrada) deje de estar disponible para los cambios. Hacer que estos campos no estén disponibles para los cambios es un comportamiento esperado si elige indicar como destino todas las instancias.

Control de las automatizaciones a escala

Puede controlar la implementación de una automatización en una flota de recursos de AWS mediante la especificación de un valor de simultaneidad y un umbral de error. La simultaneidad y el umbral de error se denominan colectivamente controles de frecuencia.

Simultaneidad

Utilice la simultaneidad para especificar la cantidad de recursos que pueden ejecutar una automatización de forma simultánea. La simultaneidad ayuda a limitar el impacto o el tiempo de inactividad en sus recursos cuando se procesa una automatización. Puede especificar un número absoluto de recursos, por ejemplo, 20 o un porcentaje del destino definido, por ejemplo, el 10 %.

El sistema de colas entrega la automatización a un solo recurso y espera hasta que se complete la invocación inicial antes de enviar la automatización a dos o más recursos. El sistema envía la automatización a más recursos de manera exponencial hasta que se alcanza el valor de simultaneidad.

Umbrales de error

Utilice un umbral de error para especificar la cantidad de automatizaciones que pueden presentar error antes de que AWS Systems Manager deje de enviar la automatización a otros recursos. Puede especificar un número absoluto de errores, por ejemplo, 10 o un porcentaje del destino definido, por ejemplo, el 10 %.

Si especifica un número absoluto de 3 errores, por ejemplo, el sistema dejará de ejecutar la automatización cuando se reciba el cuarto error. Si especifica 0, el sistema dejará de ejecutar la automatización en otros destinos una vez que se reciba el primer resultado de error.

Por ejemplo, si envía una automatización a 50 instancias y establece el umbral de error en 10 %, el sistema dejará de enviar el comando a otras instancias cuando se reciba el quinto error. Las invocaciones que ya están ejecutando una automatización cuando se alcanza un umbral de error tienen permiso para completar el procesamiento, pero algunas de estas automatizaciones también podrían presentar un error. Si debe asegurarse de que no se produzcan más errores que el número especificado para el umbral de error, establezca el valor Concurrency (Simultaneidad) en 1 de modo que las automatizaciones se procesen de una en una.

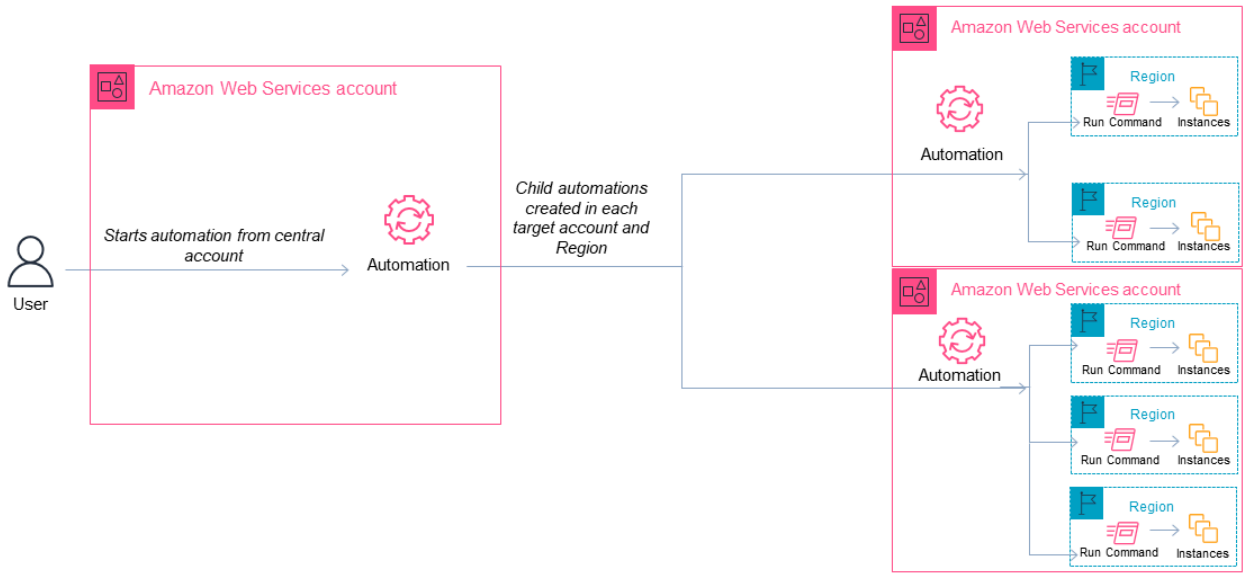
Ejecución de automatizaciones en varias cuentas y Regiones de AWS

Puede ejecutar automatizaciones de AWS Systems Manager en varias Regiones de AWS y Cuentas de AWS, o unidades organizativas de AWS Organizations desde una cuenta central. Automatización es una herramienta de AWS Systems Manager. La ejecución de automatizaciones en varias regiones, cuentas u OU reduce la cantidad de tiempo necesario para administrar los recursos de AWS, a la vez que mejora la seguridad de su entorno informático.

Por ejemplo, puede hacer lo siguiente con manuales de procedimientos de automatización:

- Implementar revisiones y actualizaciones de seguridad de forma centralizada.
- Corregir la desviación del cumplimiento en las configuraciones de VPC o en políticas del bucket de Amazon S3.
- Administrar recursos a escala, como instancias de Amazon Elastic Compute Cloud (Amazon EC2).

En el siguiente diagrama, se muestra un ejemplo de un usuario que ejecuta el manual de procedimientos `AWS-RestartEC2Instances` en varias regiones y cuentas desde una cuenta central. La automatización localiza las instancias a través de las etiquetas especificadas en las regiones y las cuentas determinadas.



Elija una cuenta central para Automatización

Si desea ejecutar automatizaciones en todas las unidades organizativas, la cuenta central debe tener permisos para enumerar todas las cuentas de las mismas. Esto solo es posible desde una cuenta de administrador delegado o desde la cuenta de administración de la organización. Le recomendamos que siga las prácticas recomendadas de AWS Organizations y use una cuenta de administrador delegado. Para obtener más información acerca de las prácticas recomendadas de AWS Organizations, consulte Prácticas recomendadas para la cuenta de administración en la Guía del usuario de AWS Organizations. Para crear una cuenta de administrador delegado de Systems Manager, puede utilizar el comando `register-delegated-administrator` con la AWS CLI, tal y como se muestra en el siguiente ejemplo.

```
aws organizations register-delegated-administrator \
    --account-id delegated admin account ID \
    --service-principal ssm.amazonaws.com
```

Si quiere ejecutar automatizaciones en varias cuentas que no estén administradas por AWS Organizations, recomendamos crear una cuenta dedicada a la administración de automatizaciones. Ejecutar todas las automatizaciones entre cuentas desde una cuenta dedicada simplifica la administración de permisos de IAM, la solución de problemas y crea una capa de separación entre

las operaciones y la administración. Este enfoque también se recomienda si usa AWS Organizations, pero solo se quiere dirigirse a cuentas individuales y no a unidades organizativas.

Cómo funciona la ejecución de automatizaciones

La ejecución de automatizaciones en varias regiones, cuentas u OU funciona de la siguiente manera:

1. Inicie sesión en la cuenta que desee configurar como cuenta central de Automatización.
2. Utilice el procedimiento Configuración de permisos de administración de la cuenta para la automatización de varias regiones y cuentas en este tema para crear los siguientes roles de IAM:
   - **`AWS-SystemsManager-AutomationAdministrationRole`**: este rol concede a los usuarios permiso para ejecutar automatizaciones en varias cuentas y unidades organizativas.
   - **`AWS-SystemsManager-AutomationExecutionRole`**: este rol concede a los usuarios permiso para ejecutar automatizaciones en cuentas determinadas.
3. Elija el manual de procedimientos, las regiones, las cuentas o las OU donde desea ejecutar la automatización.

   > **Note**
   >
   > Asegúrese de que la unidad organizativa de destino contenga las cuentas deseadas. Si elige un manual de procedimientos personalizado, este debe compartirse con todas las cuentas de destino. Para obtener información acerca de cómo compartir manuales de procedimiento, consulte Uso compartido de documentos de SSM. Para obtener información acerca del uso de manuales de procedimientos compartidos, consulte Uso de documentos de SSM compartidos.

4. Ejecute la automatización.

   > **Note**
   >
   > Al ejecutar automatizaciones en varias regiones, cuentas u OU, la automatización que ejecuta desde la cuenta principal inicia las automatizaciones secundarias en cada una de las cuentas de destino. La automatización en la cuenta principal tiene pasos `aws:executeAutomation` para cada una de las cuentas de destino.

5. Utilice las operaciones de la API GetAutomationExecution, DescribeAutomationStepExecutions y DescribeAutomationExecutions desde la consola de AWS Systems Manager o la AWS CLI para supervisar el progreso de la automatización. La salida de los pasos para la automatización en su

cuenta principal será el `AutomationExecutionId` de las automatizaciones secundarias. Para ver la salida de las automatizaciones secundarias creadas en las cuentas de destino, asegúrese de especificar la cuenta, la región y el `AutomationExecutionId` adecuados en su solicitud.

Configuración de permisos de administración de la cuenta para la automatización de varias regiones y cuentas

Utilice el siguiente procedimiento para crear los roles de IAM necesarios para la automatización de Systems Manager de varias regiones y cuentas mediante AWS CloudFormation. Este procedimiento describe cómo crear el rol **`AWS-SystemsManager-AutomationAdministrationRole`**. Solo tiene que crear este rol en la cuenta central de Automatización. Este procedimiento también describe cómo crear el rol **`AWS-SystemsManager-AutomationExecutionRole`**. Debe crear este rol en todas las cuentas que desea indicar como destino para ejecutar automatizaciones de varias regiones y cuentas. Se recomienda utilizar AWS CloudFormation StackSets para crear el rol **`AWS-SystemsManager-AutomationExecutionRole`** en las cuentas que desee indicar como destino para ejecutar automatizaciones de varias regiones y cuentas.

Para crear el rol de automatización de IAM requerido en las automatizaciones de varias regiones y cuentas mediante el uso de AWS CloudFormation

1. Descargue y descomprima `AWS-SystemsManager-AutomationAdministrationRole.zip`. O, si sus cuentas están administradas por AWS Organizations, `AWS-SystemsManager-AutomationAdministrationRole(org).zip`. Este archivo contiene el archivo de plantilla `AWS-SystemsManager-AutomationAdministrationRole.yaml` de AWS CloudFormation.
2. Abra la consola de AWS CloudFormation en https://console.aws.amazon.com/cloudformation.
3. Elija Crear pila.
4. En la sección Especificar plantilla elija Cargar un archivo de plantilla.
5. Elija Elegir archivo y, a continuación, elija el archivo de plantilla `AWS-SystemsManager-AutomationAdministrationRole.yaml` de AWS CloudFormation.
6. Elija Siguiente.
7. En la página Especificar los detalles de la pila, escriba un nombre en el campo Nombre de la pila.
8. Elija Siguiente.
9. En la página Configurar opciones de la pila, ingrese los valores de las opciones que desea utilizar. Elija Siguiente.

10. En la página Revisar, desplácese hacia abajo y elija la opción I acknowledge that AWS CloudFormation might create IAM resources with custom names.
11. Elija Crear pila.

AWS CloudFormation muestra el estado CREATE_IN_PROGRESS durante tres minutos aproximadamente. El estado cambia a CREATE_COMPLETE.

Tiene que repetir el siguiente procedimiento en todas las cuentas que desee indicar como destino para ejecutar automatizaciones de varias regiones y cuentas.

Para crear el rol de automatización de IAM requerido en las automatizaciones de varias regiones y cuentas mediante el uso de AWS CloudFormation

1. Descargue `AWS-SystemsManager-AutomationExecutionRole.zip`. O, si sus cuentas están administradas por AWS Organizations, `AWS-SystemsManager-AutomationExecutionRole (org).zip`. Este archivo contiene el archivo de plantilla `AWS-SystemsManager-AutomationExecutionRole.yaml` de AWS CloudFormation.
2. Abra la consola de AWS CloudFormation en https://console.aws.amazon.com/cloudformation.
3. Elija Crear pila.
4. En la sección Especificar plantilla elija Cargar un archivo de plantilla.
5. Elija Elegir archivo y, a continuación, elija el archivo de plantilla `AWS-SystemsManager-AutomationExecutionRole.yaml` de AWS CloudFormation.
6. Elija Siguiente.
7. En la página Especificar los detalles de la pila, escriba un nombre en el campo Nombre de la pila.
8. En la sección Parámetros, en el campo AdminAccountId, escriba el ID de la cuenta central de Automatización.
9. Si configura este rol para un entorno de AWS Organizations, hay otro campo en la sección denominado OrganizationID. Ingrese el ID de su organización de AWS.
10. Elija Siguiente.
11. En la página Configurar opciones de la pila, ingrese los valores de las opciones que desea utilizar. Elija Siguiente.
12. En la página Revisar, desplácese hacia abajo y elija la opción I acknowledge that AWS CloudFormation might create IAM resources with custom names.
13. Elija Crear pila.

AWS CloudFormation muestra el estado CREATE_IN_PROGRESS durante tres minutos aproximadamente. El estado cambia a CREATE_COMPLETE.

Ejecución de una automatización en varias regiones y cuentas (consola)

El siguiente procedimiento describe cómo utilizar la consola de Systems Manager para ejecutar una automatización de varias regiones y cuentas desde la cuenta de administración de Automation.

Antes de empezar

Antes de completar el siguiente procedimiento, tenga en cuenta la siguiente información:

- El usuario o el rol que utilice para ejecutar una automatización de varias regiones o cuentas debe tener el permiso `iam:PassRole` para el rol `AWS-SystemsManager-AutomationAdministrationRole`.
- las OU o los ID de la Cuenta de AWS donde desee ejecutar la automatización
- las regiones admitidas por Systems Manager en las que desea ejecutar la automatización
- la clave y el valor de la etiqueta, o el nombre del grupo de recursos donde desea ejecutar la automatización

Para ejecutar una automatización en varias regiones y cuentas

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Automatización y, después, seleccione Ejecutar automatización.
3. En la lista Documento de automatización, elija un manual de procedimientos. Elija una o más opciones en el panel Categorías de documentos para filtrar documentos SSM según su propósito. Para ver un manual de procedimientos que le pertenezca, seleccione la pestaña De mi propiedad. Para ver un manual de procedimientos que se haya compartido con su cuenta, elija la pestaña Compartido conmigo. Para ver todos los manuales de procedimientos, seleccione la pestaña Todos los documentos.

   > **Note**
   >
   > Puede ver información acerca de un manual de procedimientos al seleccionar su nombre.

4. En la sección Detalles del documento, verifique que Versión del documento esté establecido como la versión que desea ejecutar. El sistema incluye las siguientes opciones de versión:

   - Versión predeterminada en tiempo de ejecución: seleccione esta opción si el manual de procedimientos de automatización se actualiza de forma periódica y se asigna una nueva versión predeterminada.
   - Última versión en tiempo de ejecución: seleccione esta opción si el manual de procedimientos de automatización se actualiza de forma periódica y desea ejecutar la versión que se ha actualizado más recientemente.
   - 1 (Predeterminado): seleccione esta opción para ejecutar la primera versión del documento, que es la predeterminada.

5. Elija Siguiente.
6. En la página Ejecutar documento de automatización, elija Varias cuentas y regiones.
7. En la sección Cuentas y regiones de destino, utilice el campo Accounts and organizational (OUs) para especificar las diferentes Cuentas de AWS o unidades organizativas de AWS donde desea ejecutar la automatización. Si indica varias cuentas u OU, sepárelas con comas.
8. Utilice la lista de Regiones de AWS para elegir una o más regiones en las que desee ejecutar la automatización.
9. Utilice las opciones Multi-Region and account rate control para restringir la automatización a un número limitado de cuentas que se ejecutan en un número limitado de regiones. Estas opciones no restringen el número de recursos de AWS que pueden ejecutar las automatizaciones.

   a. En la sección Simultaneidad de ubicación (par de cuenta y región, seleccione una opción para restringir el número de automatizaciones que se pueden ejecutar en varias cuentas y regiones al mismo tiempo. Por ejemplo, si decide ejecutar una automatización en cinco (5) Cuentas de AWS, que se encuentran en cuatro (4) Regiones de AWS, Systems Manager ejecuta las automatizaciones en un total de 20 pares de cuenta-región. Puede utilizar esta opción para especificar un número absoluto, como **2**, para que la automatización solo se ejecute en dos pares de cuenta-región a la vez. También puede especificar un porcentaje de pares cuenta-región que se pueden ejecutar al mismo tiempo. Por ejemplo, con 20 pares de cuenta-región, si especifica un 20 %, la automatización se ejecuta de manera simultánea en un máximo de cinco (5) pares de cuenta-región.

      - Elija destinos para ingresar un número absoluto de pares de cuenta-región que pueden ejecutar la automatización simultáneamente.

- Elija porcentaje para ingresar un porcentaje del número total de pares de cuenta-región que pueden ejecutar la automatización simultáneamente.

b. En la sección Umbral de error, elija una opción:

- Elija errores para ingresar un número absoluto de errores permitidos antes de que Automation deje de enviar la automatización a otros recursos.
- Elija porcentaje para ingresar un porcentaje de errores permitidos antes de que Automation deje de enviar la automatización a otros recursos.

10. En la sección Destinos, seleccione cómo quiere que se indiquen los recursos de AWS donde desea ejecutar la automatización como destino. Estas opciones son obligatorias.

a. Use la lista Parámetro para elegir un parámetro. Los elementos de la lista Parámetro se determinan a partir de los parámetros en el manual de procedimientos de automatización que seleccionó al inicio de este procedimiento. Al elegir un parámetro, se define el tipo de recurso en el que se ejecuta el flujo de trabajo de automatización.

b. Utilice la lista Destinos para elegir cómo indicar los recursos de destino.

i. Si eligió indicar recursos como destino mediante el uso de valores de parámetro, ingrese el valor del parámetro que eligió en la sección Parámetros de entrada.

ii. Si eligió los recursos de destino con AWS Resource Groups, entonces elija el nombre del grupo de la lista Grupo de recursos.

iii. Si eligió indicar recursos de destino mediante el uso de etiquetas, introduzca la clave de etiqueta y, opcionalmente, un valor de etiqueta en los campos correspondientes. Elija Agregar.

iv. Si desea ejecutar un manual de procedimientos de automatización en todas las instancias de la Cuenta de AWS y la Región de AWS actuales, seleccione Todas las instancias.

11. En la sección Parámetros de entrada, especifique las entradas necesarias: De forma opcional, puede elegir un rol de servicio `AWS-SystemsManager-AutomationAdministrationRole` de IAM de la lista AutomationAssumeRole.

> **Note**
>
> Es posible que no tenga que elegir algunas de las opciones de la sección Parámetros de entrada. Esto se debe a que ha indicado recursos de destino en varias regiones y cuentas mediante etiquetas o un grupo de recursos. Por ejemplo, si eligió el manual de

procedimientos `AWS-RestartEC2Instance`, no necesita especificar ni elegir los ID de instancia en la sección Parámetros de entrada. La automatización localiza las instancias que desea reiniciar a través de las etiquetas que ha especificado.

12. (Opcional) Elija una alarma de CloudWatch que desee aplicar a la automatización para fines de monitoreo. Para adjuntar una alarma de CloudWatch a su automatización, la entidad principal de IAM que ejecuta esta última debe tener permiso para la acción `iam:createServiceLinkedRole`. Para obtener más información sobre las alarmas de CloudWatch, consulte Uso de alarmas de Amazon CloudWatch. Tenga en cuenta que si la alarma se activa, la automatización se cancela y se ejecutará cualquier paso `OnCancel` que haya definido. Si usa AWS CloudTrail, verá la llamada a la API en el registro de seguimiento.
13. Utilice las opciones de la sección Control de velocidad para restringir el número de recursos de AWS que pueden ejecutar la automatización dentro de cada par de cuenta-región.

    En la sección Simultaneidad, elija una opción:

    - Seleccione destinos para introducir un número absoluto de destinos que pueden ejecutar el flujo de trabajo de Automation simultáneamente.
    - Seleccione porcentaje para introducir un porcentaje del destino definido que puede ejecutar el flujo de trabajo de Automation simultáneamente.
14. En la sección Umbral de error, elija una opción:

    - Elija errores para introducir un número absoluto de errores permitidos antes de que Automation deje de enviar el flujo de trabajo a otros recursos.
    - Elija porcentaje para introducir un porcentaje de errores permitidos antes de que Automation deje de enviar el flujo de trabajo a otros recursos.
15. Elija Ejecutar.

Ejecución de Automatización en varias regiones y cuentas (línea de comandos)

El siguiente procedimiento describe cómo utilizar la AWS CLI (en Linux o Windows) o las Herramientas de AWS para PowerShell para ejecutar la automatización en varias regiones y cuentas desde la cuenta de administración de Automation.

Antes de empezar

Antes de completar el siguiente procedimiento, tenga en cuenta la siguiente información:

- las OU o los ID de la Cuenta de AWS donde desee ejecutar la automatización
- las regiones admitidas por Systems Manager en las que desea ejecutar la automatización
- la clave y el valor de la etiqueta, o el nombre del grupo de recursos donde desea ejecutar la automatización

Para ejecutar una automatización en varias regiones y cuentas

1. Si aún no lo ha hecho, instale y configure la AWS CLI o las Herramientas de AWS para PowerShell.

   Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.

2. Utilice el siguiente formato a fin de crear un comando para ejecutar una automatización en varias regiones y cuentas. Reemplace cada *example resource placeholder* con su propia información.

   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" }
```

A continuación, se presentan varios ejemplos.

Ejemplo 1: en este ejemplo se reinician las instancias de EC2 en tres regiones de toda la organización de AWS Organizations. Para ello, céntrese en el ID raíz de la organización e incluya las unidades organizativas secundarias.

Linux & macOS

```
aws ssm start-automation-execution \
        --document-name "AWS-RestartEC2Instance" \
        --target-parameter-name InstanceId \
        --targets '[{"Key":"AWS::EC2::Instance","Values":["*"]}]' \
        --target-locations '[{
            "Accounts": ["r-example"],
            "IncludeChildOrganizationUnits": true,
            "Regions": ["us-east-1", "us-east-2", "us-west-2"]
        }]'
```

Windows

```
aws ssm start-automation-execution \
        --document-name "AWS-RestartEC2Instance" ^
```

```
        --target-parameter-name InstanceId ^
        --targets '[{"Key":"AWS::EC2::Instance","Values":["*"]}]' ^
        --target-locations '[{
            "Accounts": ["r-example"],
            "IncludeChildOrganizationUnits": true,
            "Regions": ["us-east-1", "us-east-2", "us-west-2"]
        }]'
```

PowerShell

```
Start-SSMAutomationExecution `
        -DocumentName "AWS-RestartEC2Instance" `
        -TargetParameterName "InstanceId" `
        -Targets '[{"Key":"AWS::EC2::Instance","Values":["*"]}]'
        -TargetLocation @{
            "Accounts"="r-example";
            "Regions"="us-east-1", "us-east-2", "us-west-2";
            "IncludeChildOrganizationUnits"=true}
```

Ejemplo 2: en este ejemplo se reinician las instancias de EC2 específicas en distintas cuentas y regiones.

Linux & macOS

```
aws ssm start-automation-execution \
        --document-name "AWS-RestartEC2Instance" \
        --target-parameter-name InstanceId \
        --target-locations '[{
            "Accounts": ["123456789012"],
            "Targets": [{
                "Key":"ParameterValues",
                "Values":["i-02573cafcfEXAMPLE", "i-0471e04240EXAMPLE"]
            }],
            "TargetLocationMaxConcurrency": "100%",
            "Regions": ["us-east-1"]
        }, {
            "Accounts": ["987654321098"],
            "Targets": [{
                "Key":"ParameterValues",
                "Values":["i-07782c72faEXAMPLE"]
            }],
```

```
            "TargetLocationMaxConcurrency": "100%",
            "Regions": ["us-east-2"]
        }]'
```

Windows

```
aws ssm start-automation-execution ^
        --document-name "AWS-RestartEC2Instance" ^
        --target-parameter-name InstanceId ^
        --target-locations '[{
            "Accounts": ["123456789012"],
            "Targets": [{
                "Key":"ParameterValues",
                "Values":["i-02573cafcfEXAMPLE", "i-0471e04240EXAMPLE"]
            }],
            "TargetLocationMaxConcurrency": "100%",
            "Regions": ["us-east-1"]
        }, {
            "Accounts": ["987654321098"],
            "Targets": [{
                "Key":"ParameterValues",
                "Values":["i-07782c72faEXAMPLE"]
            }],
            "TargetLocationMaxConcurrency": "100%",
            "Regions": ["us-east-2"]
        }]'
```

PowerShell

```
Start-SSMAutomationExecution `
        -DocumentName "AWS-RestartEC2Instance" `
        -TargetParameterName "InstanceId" `
        -Targets '[{"Key":"AWS::EC2::Instance","Values":["*"]}]'
        -TargetLocation @({
            "Accounts"="123456789012",
            "Targets"= @{
                "Key":"ParameterValues",
                "Values":["i-02573cafcfEXAMPLE", "i-0471e04240EXAMPLE"]
            },
            "TargetLocationMaxConcurrency"="100%",
            "Regions"=["us-east-1"]
        }, {
            "Accounts"="987654321098",
```

```
            "Targets": @{
                "Key":"ParameterValues",
                "Values":["i-07782c72faEXAMPLE"]
            },
            "TargetLocationMaxConcurrency": "100%",
            "Regions"=["us-east-2"]
        })
```

Ejemplo 3: en este ejemplo se reinician las instancias de EC2 en las cuentas `123456789012` y `987654321098`, que se encuentran en las regiones `us-east-2` y `us-west-1`. Las instancias deben estar etiquetadas con el valor de par de claves de etiqueta `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" }
```

Ejemplo 4: en este ejemplo se reinician las instancias de EC2 en las cuentas `123456789012` y `987654321098`, que se encuentran en la región `eu-central-1`. Las instancias deben pertenecer al grupo de recursos `prod-instances` de AWS.

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" }
```

Ejemplo 5: en este ejemplo se reinician las instancias de EC2 de la unidad organizativa `ou-1a2b3c-4d5e6c` de AWS. Las instancias se encuentran en las regiones `us-west-1` y `us-west-2`. Las instancias deben pertenecer al grupo de recursos `WebServices` de AWS.

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" }
```

El sistema devuelve información similar a la siguiente.

Linux & macOS

```
{
        "AutomationExecutionId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
    }
```

Windows

```
{
        "AutomationExecutionId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
    }
```

PowerShell

```
4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE
```

3. Ejecute el siguiente comando para ver detalles de la automatización. Reemplace *automation execution ID* con su propia información.

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. Ejecute el siguiente comando para ver detalles sobre el progreso de la automatización.

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**
>
> También puede monitorear el estado de la automatización en la consola. En la lista Ejecuciones de automatización, elija la automatización que acaba de ejecutar y, a

continuación, seleccione la pestaña Execution steps. Esta pestaña muestra el estado de las acciones de la automatización.

Más información

Aplicación de revisiones centralizada en varias cuentas y regiones con AWS Systems Manager Automation

Ejecución de automatizaciones a partir de eventos EventBridge

Puede iniciar una automatización al indicar un manual de procedimientos como destino de un evento de Amazon EventBridge. Puede iniciar las automatizaciones según una programación o cuando se produzca un evento específico del sistema de AWS. Por ejemplo, supongamos que crea un manual de procedimientos denominado BootStrapInstances, el cual instala software en una instancia cuando esta se inicia. Para especificar el manual de procedimientos BootStrapInstances (y la automatización correspondiente) como destino de un evento de EventBridge, primero debe crear una nueva regla de EventBridge. (A continuación se muestra un ejemplo de regla: Service name: EC2, Event Type: EC2 Instance State-change Notification, Specific state(s): running, Any instance). A continuación, utiliza los siguientes procedimientos para indicar el manual de procedimientos BootStrapInstances como destino del evento a través de la consola de EventBridge y la AWS Command Line Interface (AWS CLI). Cuando se inicia una instancia nueva, el sistema ejecuta la automatización e instala el software.

Para obtener más información acerca de la creación de manuales de procedimientos, consulte Creación de sus propios manuales de procedimientos.

Creación de un evento de EventBridge que utiliza un manual de procedimientos (consola)

Utilice el siguiente procedimiento para configurar un manual de procedimientos como destino de un evento de EventBridge.

Para configurar un manual de procedimientos como destino de una regla de eventos de EventBridge

1. Abra la consola de Amazon EventBridge en https://console.aws.amazon.com/events/.
2. En el panel de navegación, seleccione Reglas.
3. Elija Creación de regla.
4. Escriba un nombre y una descripción para la regla.

Una regla no puede tener el mismo nombre que otra regla de la misma región y del mismo bus de eventos.

5. En Bus de eventos, seleccione el bus de eventos que desea asociar a esta regla. Si desea que esta regla responda a eventos coincidentes procedentes de su propia Cuenta de AWS, seleccione default (predeterminado). Cuando un Servicio de AWS en su cuenta emite un evento, siempre va al bus de eventos predeterminado de su cuenta.
6. Elija cómo se activa la regla.

Para crear una regla basada en…	Haga lo siguiente...	
Evento	a. En Tipo de regla, elija Regla con un patrón de evento. b. Elija Siguiente. c. En Origen del evento, elija Eventos de AWS o eventos de socios de EventBridge. d. En la sección Event pattern (Patrón de eventos), realice una de las siguientes acciones: • Para utilizar una plantilla a fin de crear el patrón de eventos, elija Event pattern form (Formulario de patrón de eventos) y después Event source (Origen del evento), AWS service (Servicio de) y Event type (Tipo de evento). Si elige	

Para crear una regla basada en…	Haga lo siguiente...	
	All Events (Todos los eventos) como el tipo de evento, todos los eventos emitidos por este Servicio de AWS coincidirán con la regla. Para personalizar la plantilla, seleccione Patron personalizado (editor JSON) y realice los cambios. • Para utilizar un patrón de eventos personalizado, elija Custom pattern (JSON editor) (Patrón personalizado [editor JSON]) y cree su patrón de evento.	

Para crear una regla basada en…	Haga lo siguiente...	
Programación	a. En Rule type (Tipo de regla), elija Schedule (Programación). b. Elija Siguiente. c. En Schedule pattern (Programar patrón), realice una de las siguientes acciones: • Para utilizar una expresión cron para definir la programación, elija A fine-grained schedule that runs at a specific time, such as 8:00 a.m. (Una programación detallada que se ejecuta a una hora específica, como las 8:00 h). PST on the first Monday of every month (PST el primer lunes de cada mes) e ingrese la expresión cron. • Para utilizar una expresión rate para definir la programación, elija A schedule that runs at a regular rate, such as every 10 minutes (Una programación que se ejecuta a una frecuenci	

Para crear una regla basada en…	Haga lo siguiente...	
	a regular, como cada 10 minutos) e ingrese la expresión rate.	

7. Elija Siguiente.
8. En Target types (Tipos de destino), elija AWS service.
9. Para Select a target (Seleccione un destino), elija Systems Manager Automation (Automatización de Systems Manager).
10. En Document (Documento), elija un manual de procedimientos para utilizarlo cuando se invoque el destino.
11. En la sección Configure automation parameter(s) (Configurar parámetros de automatización), mantenga los valores de parámetro predeterminados (si están disponibles) o escriba sus propios valores.

> **Note**
>
> Para crear un objetivo, debe especificar un valor para cada uno de los parámetros obligatorios. Si no lo hace, el sistema crea la regla, pero esta no se ejecutará.

12. Si hay muchos tipos de destino, EventBridge necesita permisos para enviar eventos al destino. En estos casos, EventBridge puede crear el rol de IAM necesario para que se ejecute la regla. Realice una de las siguientes acciones:

    - Para crear un rol de IAM automáticamente, elija Creación de un nuevo rol para este recurso específico.
    - Para utilizar un rol de IAM que haya creado antes, elija Use existing role (Usar rol existente) y seleccione el rol existente del menú desplegable. Tenga en cuenta que es posible que deba actualizar la política de confianza del rol de IAM para incluir EventBridge. A continuación, se muestra un ejemplo:

    JSON

```
{
```

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

13. Elija Siguiente.
14. (Opcional) Introduzca una o varias etiquetas para la regla. Para obtener más información, consulte Etiquetado de los recursos de Amazon EventBridge en la Guía del usuario de Amazon EventBridge.
15. Elija Siguiente.
16. Revise los detalles de la regla y seleccione Creación de regla.

Creación de un evento de EventBridge que utiliza un manual de procedimientos (línea de comandos)

El siguiente procedimiento describe cómo utilizar la AWS CLI (en Linux o Windows) o las Herramientas de AWS para PowerShell para crear una regla de eventos de EventBridge y configurar un manual de procedimientos como destino.

Para configurar un manual de procedimientos como destino de una regla de eventos de EventBridge

1. Si aún no lo ha hecho, instale y configure la AWS CLI o las Herramientas de AWS para PowerShell.

   Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.
2. Cree un comando para especificar una regla nueva de eventos de EventBridge. Reemplace cada *example resource placeholder* con su propia información.

   Se activa en función de una programación

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"
```

En el siguiente ejemplo, se crea una regla de eventos de EventBridge que se activa cada día a las 9:00 h (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 * * ? *)"
```

Se activa en función de un evento

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"]}'
```

En el siguiente ejemplo, se crea una regla de eventos de EventBridge que se activa cuando se cambia el estado de una instancia de EC2 en la región.

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"]}'
```

El comando devuelve detalles de la nueva regla de EventBridge similares a los siguientes.

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. Cree un comando para especificar un manual de procedimientos como destino de la regla de eventos de EventBridge que creó en el paso 2. Reemplace cada *example resource placeholder* con su propia información.

   Linux & macOS

   ```
   aws events put-targets \
   --rule rule name \
   --targets '{"Arn": " arn:aws:ssm:region:account ID:automation-definition/runbook
    name","Input":"{\"Message\":[\"{\\\"Key\\\":\\\"key name\\\",\\\"Values\\\":[\\
   ```

```
\"value\\\"]}\"]}","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":"{\"Message\":[\"{\\\"Key\\\":\\\"key name\\\",\\\"Values\\\":[\\
\"value\\\"]}\"]}","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
```

En el siguiente ejemplo, se crea un destino de evento de EventBridge que activa el ID de instancia especificado mediante el manual de procedimientos `AWS-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
```

El sistema devuelve información similar a la siguiente.

Linux & macOS

```
{
"FailedEntries": [],
"FailedEntryCount": 0
}
```

Windows

```
{
"FailedEntries": [],
"FailedEntryCount": 0
}
```

PowerShell

No se obtienen resultados si el comando se ejecuta satisfactoriamente para PowerShell.

Ejecutar una automatización paso a paso

Los siguientes procedimientos describen cómo utilizar la consola de AWS Systems Manager y la AWS Command Line Interface (AWS CLI) para ejecutar una automatización mediante el modo de ejecución manual. Al utilizar el modo de ejecución manual, la automatización se inicia en el estado En espera y se detiene en el estado En espera entre los diferentes pasos. Esto le permite controlar cuándo debe continuar con la automatización, lo que resulta útil si necesita revisar el resultado de un paso antes de continuar.

La automatización se ejecuta en el contexto del usuario actual. Esto significa que no tiene que configurar más permisos de IAM siempre y cuando cuente con el permiso necesario para usar el manual de procedimientos y cualquier acción que este solicite. Si tiene permisos de administrador en IAM, ya cuenta con el permiso necesario para ejecutar esta automatización.

Ejecución de una automatización paso a paso (consola)

El siguiente procedimiento muestra cómo utilizar la consola de Systems Manager para ejecutar manualmente una automatización paso a paso.

Para ejecutar una automatización paso a paso

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Automatización y, después, seleccione Ejecutar automatización.
3. En la lista Documento de automatización, elija un manual de procedimientos. Elija una o más opciones en el panel Categorías de documentos para filtrar documentos SSM según su propósito. Para ver un manual de procedimientos que le pertenezca, seleccione la pestaña De mi propiedad. Para ver un manual de procedimientos que se haya compartido con su cuenta, elija la pestaña Compartido conmigo. Para ver todos los manuales de procedimientos, seleccione la pestaña Todos los documentos.

> **Note**
>
> Puede ver información acerca de un manual de procedimientos al seleccionar su nombre.

4. En la sección Detalles del documento, verifique que Versión del documento esté establecido como la versión que desea ejecutar. El sistema incluye las siguientes opciones de versión:

   - Versión predeterminada en tiempo de ejecución: seleccione esta opción si el manual de procedimientos de automatización se actualiza de forma periódica y se asigna una nueva versión predeterminada.
   - Última versión en tiempo de ejecución: seleccione esta opción si el manual de procedimientos de automatización se actualiza de forma periódica y desea ejecutar la versión que se ha actualizado más recientemente.
   - 1 (Predeterminado): seleccione esta opción para ejecutar la primera versión del documento, que es la predeterminada.

5. Elija Siguiente.
6. En la sección Modo de ejecución, seleccione Ejecución manual.
7. En la sección Parámetros de entrada, especifique las entradas necesarias: De forma opcional, puede elegir un rol de servicio de IAM de la lista AutomationAssumeRole.
8. Elija Ejecutar.
9. Seleccione Ejecutar este paso cuando esté preparado para iniciar el primer paso de la automatización. La automatización realiza el paso uno y se detiene antes de ejecutar los siguientes pasos especificados en el manual de procedimientos que eligió en el paso 3 de este procedimiento. Si el manual de procedimientos tiene varios pasos, debe seleccionar Ejecutar este paso para que se continúe con cada paso de la automatización. Cada vez que se seleccione Ejecutar este paso se ejecuta la acción.

> **Note**
>
> La consola muestra el estado de la automatización. Si no se logra ejecutar un paso de la automatización, consulte Solución de problemas de Automatización de Systems Manager.

10. Tras realizar todos los pasos especificados en el manual de procedimientos, seleccione Completar y ver resultados para finalizar la automatización y ver los resultados.

Ejecución de una automatización paso a paso (línea de comandos)

El siguiente procedimiento describe cómo utilizar la AWS CLI (en Linux, macOS o Windows) o Herramientas de AWS para PowerShell para ejecutar manualmente una automatización paso a paso.

Para ejecutar una automatización paso a paso

1. Si aún no lo ha hecho, instale y configure la AWS CLI o las Herramientas de AWS para PowerShell.

   Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.

2. Ejecute el siguiente comando para iniciar una automatización manual. Reemplace cada *example resource placeholder* con su propia información.

   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
   ```

A continuación, se muestra un ejemplo en el que se utiliza el manual de procedimientos `AWS-RestartEC2Instance` para reiniciar la instancia de EC2 especificada.

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"}
```

El sistema devuelve información similar a la siguiente.

Linux & macOS

```
{
    "AutomationExecutionId": "ba9cd881-1b36-4d31-a698-0123456789ab"
}
```

Windows

```
{
    "AutomationExecutionId": "ba9cd881-1b36-4d31-a698-0123456789ab"
}
```

PowerShell

```
ba9cd881-1b36-4d31-a698-0123456789ab
```

3. Ejecute el siguiente comando cuando esté listo para iniciar el primer paso de la automatización. Reemplace cada *example resource placeholder* con su propia información. La automatización realiza el paso uno y se detiene antes de ejecutar los siguientes pasos especificados en el manual de procedimientos que eligió en el paso 1 de este procedimiento. Si el manual de procedimientos tiene varios pasos, debe ejecutar el siguiente comando para que se continúe con cada paso de la automatización.

   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"}
   ```

   No se obtienen resultados si el comando se ejecuta correctamente.

4. Ejecute el siguiente comando para recuperar el estado de cada ejecución de un paso en la automatización.

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
```

El sistema devuelve información similar a la siguiente.

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. Ejecute el siguiente comando para completar la automatización una vez que se hayan completado todos los pasos especificados en el manual de procedimientos elegido. Reemplace cada *example resource placeholder* con su propia información.

   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
   ```

   No se obtienen resultados si el comando se ejecuta correctamente.

Programación de automatizaciones con asociaciones de State Manager

Para iniciar una automatización, puede crear una asociación de State Manager a un manual de procedimientos. State Manager es una herramienta de AWS Systems Manager. Mediante la creación de una asociación de State Manager a un manual de procedimientos, puede indicar diferentes tipos de recursos de AWS como destino. Por ejemplo, puede crear asociaciones que implementen un estado deseado en un recurso de AWS, incluidas las siguientes:

- adjuntar un rol de Systems Manager a las instancias de Amazon Elastic Compute Cloud (Amazon EC2) para transformarlas en instancias administradas
- aplicar las reglas de entrada y salida que desee para un grupo de seguridad
- crear o eliminar copias de seguridad de Amazon DynamoDB
- crear o eliminar instantáneas de Amazon Elastic Block Store (Amazon EBS)
- desactivar los permisos de lectura y escritura en los buckets de Amazon Simple Storage Service (Amazon S3)
- activar, reiniciar o detener las instancias administradas y las instancias de Amazon Relational Database Service (Amazon RDS)
- aplicar revisiones a las AMIs de Linux, macOS y Windows

Utilice los siguientes procedimientos para crear una asociación de State Manager que ejecute una automatización a través de la consola de AWS Systems Manager y la AWS Command Line Interface (AWS CLI). Para obtener información general sobre las asociaciones e información sobre la creación de una asociación que utilice un documento de `Command` o un documento de `Policy` de SSM, consulte Cómo crear asociaciones.

Antes de empezar

Tenga en cuenta los siguientes detalles importantes antes de ejecutar una automatización con State Manager:

- Antes de crear una asociación que use un manual de procedimientos, verifique que haya configurado los permisos para Automatización, una herramienta de AWS Systems Manager. Para obtener más información, consulte Configuración de Automation.
- Las asociaciones de State Manager que usan manuales de procedimientos contribuyen al número máximo de automatizaciones que se ejecutan a la vez en su Cuenta de AWS. Puede tener como máximo 100 automatizaciones en ejecución simultánea. Para obtener más información, consulte Service Quotas de Systems Manager en Referencia general de Amazon Web Services.
- Al ejecutar una automatización, State Manager no registra las operaciones de la API iniciadas por la automatización en AWS CloudTrail.
- Systems Manager crea de forma automática un rol vinculado a servicios de modo que State Manager tenga permiso para llamar las operaciones de la API de Automatización de Systems Manager. Si lo desea, puede crear el rol vinculado al servicio usted mismo. Para ello, ejecute el siguiente comando en la AWS CLI o Herramientas de AWS para 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
```

Para obtener más información acerca de los roles vinculados a servicios, consulte Cómo utilizar roles vinculados a servicios de Systems Manager.

Creación de una asociación que ejecuta una automatización (consola)

El siguiente procedimiento describe cómo utilizar la consola de Systems Manager para crear una asociación de State Manager que ejecuta una automatización.

Para crear una asociación de State Manager que ejecute una automatización

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija State Manager y, a continuación, elija Create association (Crear asociación).
3. Escriba un nombre en el campo Nombre. Esto es opcional, pero recomendable.
4. En la lista Document (Documento), elija un manual de procedimientos. Utilice la barra de búsqueda para filtrar por manuales de procedimientos Document type : Equal : Automation (Tipo de documento : Igual : Automation). Para ver más manuales de procedimientos, utilice los números que se encuentran a la derecha de la barra de búsqueda.

> **Note**
>
> Puede ver información acerca de un manual de procedimientos al seleccionar su nombre.

5. Seleccione Simple execution (Ejecución sencilla) para ejecutar la instancia de Automation en uno o varios destinos mediante la especificación del ID de recurso de dichos destinos. Seleccione Rate control (Control de frecuencia) para ejecutar la instancia de Automation en una flota de recursos de AWS mediante la especificación de una opción de indicación de destino, como etiquetas o AWS Resource Groups. También puede controlar la operación de la automatización en sus recursos mediante la especificación de la simultaneidad y los umbrales de error.

   Si ha seleccionado Rate control (Control de frecuencia), se muestra la sección Targets (Destinos).

6. En la sección Targets (Destinos), seleccione un método para los recursos de indicación de destino.

   a. (Obligatorio) En la lista Parameter (Parámetro), seleccione un parámetro. Los elementos de la lista Parameter (Parámetro) se determinan a partir de los parámetros en el manual de procedimientos que seleccionó al inicio de este procedimiento. Al elegir un parámetro, define el tipo de recurso en el que se ejecuta la automatización.

   b. (Obligatorio) En la lista Targets (Destinos), seleccione un método para indicar los recursos de destino.

      - Resource Group (Grupo de recursos): seleccione el nombre del grupo en la lista Resource Group (Grupo de recursos). Para obtener más información acerca de la indicación de AWS Resource Groups como destino en los manuales de procedimientos, consulte Indicar AWS Resource Groups de destino.
      - Tags (Etiquetas): introduzca la clave de etiqueta y, si lo desea, el valor de etiqueta en los campos proporcionados. Elija Agregar. Para obtener más información acerca de cómo indicar las etiquetas como destino en los manuales de procedimientos, consulte Especificación de una etiqueta como destino.
      - Parameter Values (Valores de parámetros): introduzca valores en la sección Input parameters (Parámetros de entrada) . Si especifica varios valores, Systems Manager ejecuta una automatización secundaria en cada valor especificado.

Por ejemplo, supongamos que su manual de procedimientos incluye un parámetro InstanceID. Si indica los valores del parámetro InstanceID como destino a la hora de ejecutar la automatización, Systems Manager ejecuta una automatización secundaria para el valor de ID de cada instancia especificada. La automatización principal se habrá completado cuando la automatización termine de ejecutar cada instancia especificada o cuando se produzca un error en la automatización. Puede indicar un máximo de 50 valores de parámetros de destino. Para obtener más información acerca de cómo indicar los valores de parámetros como destino en los manuales de procedimientos, consulte Indicar valores de parámetros de destino.

7. En la sección Parámetros de entrada, especifique los parámetros de entrada necesarios.

   Si decidió indicar recursos como destino a través de etiquetas o un grupo de recursos, es posible que no se vea obligado a elegir algunas de las opciones de la sección Input parameters (Parámetros de entrada). Por ejemplo, si eligió el manual de procedimientos `AWS-RestartEC2Instance` y decidió indicar las instancias como destino a través de etiquetas, entonces no necesita especificar ni elegir los ID de instancias en la sección Input parameters (Parámetros de entrada). La automatización localiza las instancias que desea reiniciar a través de las etiquetas que ha especificado.

   > **Important**
   >
   > Debe especificar un ARN de rol en el campo AutomationAssumeRole. State Manager utiliza el rol de asunción para llamar los Servicios de AWS especificados en el manual de procedimientos y ejecutar las asociaciones de Automation en su nombre.

8. En la sección Specify schedule (Especificar programación), seleccione On Schedule (De forma programada) si desea ejecutar la asociación periódicamente. Si selecciona esta opción, utilice las opciones proporcionadas para crear la programación mediante expresiones cron o rate. Para obtener más información acerca de las expresiones cron y rate para State Manager, consulte Expresiones cron y rate para asociaciones.

   > **Note**
   >
   > Las expresiones rate son el mecanismo de programación preferido para las asociaciones de State Manager que utilizan manuales de procedimientos. Las expresiones rate ofrecen una mayor flexibilidad para la ejecución de asociaciones en caso de que se alcance el número máximo de automatizaciones en ejecución

simultánea. Con una programación de frecuencia, Systems Manager puede volver a intentar completar la automatización poco después de recibir una notificación que indique que las automatizaciones simultáneas han alcanzado el número máximo y se han limitado.

Seleccione No schedule (Sin programación) si desea ejecutar la asociación una sola vez.

9. (Opcional) En la sección Rate Control (Control de frecuencia), elija las opciones Concurrency (Simultaneidad) y Error threshold (Umbral de error) para controlar la implementación de la automatización en todos los recursos de AWS.

   a. En la sección Simultaneidad, elija una opción:

      - Elija targets (destinos) para ingresar un número absoluto de destinos que pueden ejecutar la automatización simultáneamente.
      - Elija percentage (porcentaje) para ingresar un porcentaje del conjunto de destinos que puede ejecutar la automatización simultáneamente.

   b. En la sección Umbral de error, elija una opción:

      - Elija errors (errores) para ingresar un número absoluto de errores permitidos antes de que Automation deje de enviar la automatización a otros recursos.
      - Elija percentage (porcentaje) para ingresar un porcentaje de errores permitidos antes de que Automation deje de enviar la automatización a otros recursos.

   Para obtener más información sobre el uso de controles de frecuencia y destinos con Automation, consulte Ejecución de operaciones automatizadas a escala.

10. Elija Crear asociación.

> **Important**
>
> Al crear una asociación, esta se ejecuta de inmediato en los destinos especificados. A continuación, la asociación se ejecuta en función de la expresión cron o rate que haya seleccionado. Si seleccionó No schedule (Sin programación), la asociación no volverá a ejecutarse.

Creación de una asociación que ejecuta una automatización (línea de comandos)

El siguiente procedimiento describe cómo utilizar la AWS CLI (en Linux o Windows) o las Herramientas de AWS para PowerShell para crear una asociación de State Manager que ejecute una automatización.

Antes de empezar

Antes de completar el siguiente procedimiento, asegúrese de haber creado un rol de servicio de IAM que contenga los permisos necesarios para ejecutar el manual de procedimientos y haya configurado una relación de confianza para Automatización, una herramienta de AWS Systems Manager. Para obtener más información, consulte Tarea 1: crear un rol de servicio para Automation.

Para crear una asociación que ejecute una automatización

1. Si aún no lo ha hecho, instale y configure la AWS CLI o las Herramientas de AWS para PowerShell.

   Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.

2. Ejecute el siguiente comando para ver una lista de documentos.

   Linux & macOS

   ```
   aws ssm list-documents
   ```

   Windows

   ```
   aws ssm list-documents
   ```

   PowerShell

   ```
   Get-SSMDocumentList
   ```

   Anote el nombre del manual de procedimientos que desea utilizar para la asociación.

3. Ejecute el siguiente comando para ver detalles acerca del manual de procedimientos. En el siguiente comando, reemplace *runbook name* con su propia información.

Linux & macOS

```
aws ssm describe-document \
--name runbook name
```

Especifique el nombre de parámetro (por ejemplo, `InstanceId`) que desea utilizar para la opción `--automation-target-parameter-name`. Este parámetro determina el tipo de recurso en el que se ejecuta la automatización.

Windows

```
aws ssm describe-document ^
--name runbook name
```

Especifique el nombre de parámetro (por ejemplo, `InstanceId`) que desea utilizar para la opción `--automation-target-parameter-name`. Este parámetro determina el tipo de recurso en el que se ejecuta la automatización.

PowerShell

```
Get-SSMDocumentDescription `
-Name runbook name
```

Especifique el nombre de parámetro (por ejemplo, `InstanceId`) que desea utilizar para la opción `AutomationTargetParameterName`. Este parámetro determina el tipo de recurso en el que se ejecuta la automatización.

4. Cree un comando que ejecute una automatización mediante una asociación de State Manager. Reemplace cada *example resource placeholder* con su propia información.

   Indicar destino mediante etiquetas

   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 crea una asociación mediante la AWS CLI, utilice el parámetro `--targets` en las instancias de destino para la asociación. No utilice el parámetro `--instance-id`. El parámetro `--instance-id` es un parámetro heredado.

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 crea una asociación mediante la AWS CLI, utilice el parámetro `--targets` en las instancias de destino para la asociación. No utilice el parámetro `--instance-id`. El parámetro `--instance-id` es un parámetro heredado.

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 crea una asociación mediante la Herramientas de AWS para PowerShell, utilice el parámetro `Target` en las instancias de destino para la asociación. No utilice el parámetro `InstanceId`. El parámetro `InstanceId` es un parámetro heredado.

Indicar destino mediante valores de parámetros

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"
```

Indicar destino mediante 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"
```

Indicar varias cuentas y regiones como destino

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"]
```

El comando devuelve detalles de la nueva asociación similares a los siguientes.

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 utiliza las etiquetas para crear una asociación en una o varias instancias de destino, elimine las etiquetas de una instancia (dicha instancia ya no ejecutará la asociación). La instancia se desvincula del documento de State Manager.

Automatizaciones de solución de problemas ejecutadas por asociaciones de State Manager

La Automatización de Systems Manager implementa un límite de 100 automatizaciones simultáneas y 1000 automatizaciones en cola por cuenta, por región. Si una asociación de State Manager que usa un manual de procedimientos tiene el estado Failed (Error) y el estado detallado AutomationExecutionLimitExceeded, es posible que su automatización haya alcanzado el límite. Como resultado, Systems Manager limita las automatizaciones. Para resolver este problema, siga estos pasos:

- Utilice una expresión rate o cron diferente para la asociación. Por ejemplo, si la asociación está programada para ejecutarse cada 30 minutos, cambie la expresión para que se ejecute cada hora o cada dos horas.
- Elimine las automatizaciones existentes con el estado Pending (Pendiente). Si elimina estas automatizaciones, se vaciará la cola actual.

Programación de automatizaciones con periodos de mantenimiento

Puede iniciar una automatización mediante la configuración de un manual de procedimientos como una tarea registrada para un periodo de mantenimiento. Al registrar el manual de procedimientos como una tarea registrada, el periodo de mantenimiento ejecuta la automatización durante el periodo de mantenimiento programado.

Por ejemplo, supongamos que crea un manual de procedimientos denominado `CreateAMI` que genera una Amazon Machine Image (AMI) de las instancias registradas como destinos en el periodo de mantenimiento. Para especificar el manual de procedimientos `CreateAMI` (y la automatización

correspondiente) como una tarea registrada de un periodo de mantenimiento, primero debe crear un periodo de mantenimiento y registrar los destinos. A continuación, utilice el siguiente procedimiento para especificar el documento `CreateAMI` como una tarea registrada en el periodo de mantenimiento. Cuando se inicia el periodo de mantenimiento durante el periodo programado, el sistema ejecuta la automatización y crea una AMI de los destinos registrados.

Para obtener información acerca de la creación de manuales de procedimientos de Automation, consulte Creación de sus propios manuales de procedimientos. Automatización es una herramienta de AWS Systems Manager.

Utilice los siguientes procedimientos para configurar una automatización como una tarea registrada para un periodo de mantenimiento a través de la consola de AWS Systems Manager, la AWS Command Line Interface (AWS CLI) o las AWS Tools for Windows PowerShell.

Registro de una tarea de automatización en un periodo de mantenimiento (consola)

El siguiente procedimiento describe cómo utilizar la consola de Systems Manager para configurar una automatización como una tarea registrada para un periodo de mantenimiento.

Antes de empezar

Antes de realizar el siguiente procedimiento, debe crear un período de mantenimiento y registrar al menos un destino. Para obtener más información, consulte los siguientes procedimientos:

- Crear y configurar un periodo de mantenimiento mediante la consola.
- Asignar destinos a un periodo de mantenimiento mediante la consola

Para configurar una automatización como una tarea registrada para un periodo de mantenimiento

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación de la izquierda, seleccione Maintenance Windows y, a continuación, el período de mantenimiento con el que desee registrar una tarea de Automation.
3. Elija Acciones. A continuación, seleccione Register Automation task (Registrar tarea de Automation) para ejecutar la automatización que desee en los destinos mediante el uso de un manual de procedimientos.
4. En Name (Nombre), escriba un nombre para la tarea.
5. En Descripción, escriba una descripción.

6. En Document (Documento), elija el manual de procedimientos que defina las tareas que se ejecutarán.
7. En Document version (Versión de documento), elija la versión del manual de procedimientos que se utilizará.
8. En Task priority (Prioridad de tarea), especifique una prioridad para esta tarea. 1 es la prioridad más alta. Las tareas de un período de mantenimiento se programan por orden de prioridad; las tareas que tengan la misma prioridad se programan en paralelo.
9. En la sección Targets (Destinos), si el manual de procedimientos que eligió ejecuta tareas sobre recursos, identifique los destinos en los que desea ejecutar esta automatización. Para ello, especifique las etiquetas o seleccione las instancias de forma manual.

   > **Note**
   >
   > Si desea pasar los recursos a través de parámetros de entrada en lugar de destinos, no necesita especificar un destino de periodo de mantenimiento.
   > En muchos casos, no es necesario especificar de forma explícita un destino para una tarea de automatización. Por ejemplo, suponga que crea una tarea de tipo Automation para actualizar una Amazon Machine Image (AMI) para Linux mediante el manual de procedimientos `AWS-UpdateLinuxAmi`. Cuando se ejecuta la tarea, la AMI se actualiza con los paquetes de distribución de Linux y el software de Amazon disponibles más recientes. Las instancias nuevas que se crearon a partir de la AMI ya tienen estas actualizaciones instaladas. Como el ID de la AMI que se actualizará se especifica en los parámetros de entrada del manual de procedimientos, no es necesario volver a especificar un destino en la tarea del periodo de mantenimiento.

   Para obtener información acerca de las tareas del periodo de mantenimiento que no requieren destinos, consulte the section called “Registro de tareas del periodo de mantenimiento sin destinos”.
10. (Opcional) En Control de velocidad:

    > **Note**
    >
    > Si la tarea que ejecuta no especifica destinos, no es necesario que especifique controles de frecuencia.

- En Concurrency (Simultaneidad), especifique un número o un porcentaje de destinos en los que desee ejecutar la automatización de forma simultánea.

  Si ha seleccionado destinos mediante la elección de pares de clave-valor para la etiqueta y no está seguro de la cantidad de destinos que utilizarán las etiquetas seleccionadas, limite el número de automatizaciones que se pueden ejecutar a la vez mediante la especificación de un porcentaje.

  Mientras transcurre el periodo de mantenimiento, se inicia una nueva automatización por cada destino. Hay un límite de 100 automatizaciones simultáneas por Cuenta de AWS. Si especifica una tasa de simultaneidad mayor que 100, las automatizaciones simultáneas que superen el valor 100 se agregarán automáticamente a la cola de automatizaciones. Para obtener más información, consulte Service Quotas de Systems Manager en la Referencia general de Amazon Web Services.
- En Error threshold (Umbral de error), especifique cuándo desea que se detenga la ejecución de la automatización en otros destinos después de que se presentan errores en un número o un porcentaje de destinos. Por ejemplo, si especifica tres errores, Systems Manager deja de ejecutar las automatizaciones cuando se recibe el cuarto error. Los destinos que sigan procesando la automatización también pueden enviar errores.

11. En la sección Input Parameters (Parámetros de entrada), especifique los parámetros para el manual de procedimientos. En el caso de los manuales de procedimientos, el sistema completará automáticamente algunos de los valores. Puede conservar o reemplazar estos valores.

> **Important**
>
> Si así lo desea, puede especificar un rol de asunción de Automation para los manuales de procedimientos. Si no especifica ningún rol para este parámetro, la automatización asume el rol de servicio del periodo de mantenimiento que elija en el paso 11. Por tanto, debe asegurarse de que el rol de servicio del periodo de mantenimiento que elija tenga los permisos de AWS Identity and Access Management (IAM) adecuados para realizar las acciones definidas en el manual de procedimientos.
> Por ejemplo, el rol vinculado al servicio para Systems Manager no tiene el permiso de IAM `ec2:CreateSnapshot`, que es necesario para usar el manual de procedimientos `AWS-CopySnapshot`. En este caso, debe utilizar un rol de servicio del período de mantenimiento personalizado o especificar un rol de asunción de Automation con

los permisos `ec2:CreateSnapshot`. Para obtener más información, consulta Configuración de Automation.

12. En el área IAM service role (Rol de servicio de IAM), elija un rol con el fin de proporcionar permisos a Systems Manager para inicie la automatización.

    Para crear un rol de servicio personalizado para tareas de periodo de mantenimiento, consulte Configuración de Maintenance Windows.
13. Seleccione Register Automation task (Registrar tarea de Automation).

Registro de una tarea de Automation en un periodo de mantenimiento (línea de comandos)

El siguiente procedimiento describe cómo utilizar la AWS CLI (en Linux o Windows) o las Herramientas de AWS para PowerShell para configurar una automatización como una tarea registrada para un periodo de mantenimiento.

Antes de empezar

Antes de realizar el siguiente procedimiento, debe crear un período de mantenimiento y registrar al menos un destino. Para obtener más información, consulte los siguientes procedimientos:

- Paso 1: cree el periodo de mantenimiento mediante la AWS CLI.
- Paso 2: registre un nodo de destino con el periodo de mantenimiento mediante la AWS CLI

Para configurar una automatización como una tarea registrada para un periodo de mantenimiento

1. Si aún no lo ha hecho, instale y configure la AWS CLI o las Herramientas de AWS para PowerShell.

   Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.
2. Cree un comando para configurar una automatización como una tarea registrada para un periodo de mantenimiento. Reemplace cada *`example resource placeholder`* con su propia información.

   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 configura una automatización como una tarea registrada a través de la AWS CLI, utilice el parámetro `--Task-Invocation-Parameters` para especificar los parámetros que se pasarán a una tarea cuando se ejecute. No utilice el parámetro `--Task-Parameters`. El parámetro `--Task-Parameters` es un parámetro heredado.
>
> En el caso de las tareas del periodo de mantenimiento sin un destino especificado, no puede proporcionar valores para `--max-errors` ni `--max-concurrency`. En su lugar, el sistema inserta un valor de marcador 1, el cual podría notificarse en la respuesta a los comandos, como describe-maintenance-window-tasks y get-maintenance-window-task. Estos valores no afectan la ejecución de la tarea y se pueden ignorar.
>
> Para obtener información acerca de las tareas del periodo de mantenimiento que no requieren destinos, consulte Registro de tareas del periodo de mantenimiento sin destinos.

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 configura una automatización como una tarea registrada a través de la AWS CLI, utilice el parámetro `--task-invocation-parameters` para especificar los parámetros que se pasarán a una tarea cuando se ejecute. No utilice el parámetro `--task-parameters`. El parámetro `--task-parameters` es un parámetro heredado.
>
> En el caso de las tareas del periodo de mantenimiento sin un destino especificado, no puede proporcionar valores para `--max-errors` ni `--max-concurrency`. En su lugar, el sistema inserta un valor de marcador 1, el cual podría notificarse en la respuesta a los comandos, como describe-maintenance-window-tasks y get-maintenance-window-task. Estos valores no afectan la ejecución de la tarea y se pueden ignorar.
>
> Para obtener información acerca de las tareas del periodo de mantenimiento que no requieren destinos, consulte Registro de tareas del periodo de mantenimiento sin destinos.

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 configura una automatización como una tarea registrada a través de las Herramientas de AWS para PowerShell, utilice el parámetro `-Automation_Parameter` para especificar los parámetros que se pasarán a una tarea cuando se ejecute. No utilice el parámetro `-TaskParameters`. El parámetro `-TaskParameters` es un parámetro heredado.
>
> En el caso de las tareas del periodo de mantenimiento sin un destino especificado, no puede proporcionar valores para `-MaxError` ni `-MaxConcurrency`. En su lugar, el sistema inserta un valor de marcador de posición 1, el cual podría notificarse en la respuesta a los comandos, como `Get-SSMMaintenanceWindowTaskList` y `Get-SSMMaintenanceWindowTask`. Estos valores no afectan la ejecución de la tarea y se pueden ignorar.
>
> Para obtener información acerca de las tareas del periodo de mantenimiento que no requieren destinos, consulte Registro de tareas del periodo de mantenimiento sin destinos.

En el siguiente ejemplo, se configura una automatización como una tarea registrada en un periodo de mantenimiento con prioridad 1. También demuestra la omisión de las opciones `--targets`, `--max-errors` y `--max-concurrency` para la tarea de un periodo de mantenimiento sin destino. La automatización utiliza el manual de procedimientos `AWS-StartEC2Instance` y el rol de asunción de Automation especificado para activar instancias EC2 registradas como destinos en el periodo de mantenimiento. El periodo de mantenimiento ejecuta la automatización de manera simultánea en 5 instancias como máximo en cualquier momento. Además, la tarea registrada deja de ejecutarse en más instancias durante un intervalo determinado si el recuento de errores es superior a 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
```

El comando devuelve detalles de la nueva tarea registrada similares a los siguientes.

Linux & macOS

```
{
"WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
}
```

Windows

```
{
"WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
}
```

PowerShell

```
4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE
```

3. Para ver la tarea registrada, ejecute el siguiente comando. Reemplace *maintenance windows ID* con su propia información.

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
```

El sistema devuelve información similar a la siguiente.

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
```

Referencia de acciones de Automatización de Systems Manager

Esta referencia describe las acciones de Automation que puede especificar en un manual de procedimientos de automatización. Automatización es una herramienta de AWS Systems Manager. Estas acciones no se pueden utilizar en otros tipos de documentos de Systems Manager (SSM). Para obtener información acerca de los complementos para otros tipos de documentos de SSM, consulte Referencia de complementos del documento de comandos.

Automatización de Systems Manager ejecuta los pasos definidos en los manuales de procedimientos de Automation. Cada paso está asociado a una acción concreta. La acción determina las entradas, el comportamiento y las salidas del paso. Los pasos se definen en la sección `mainSteps` de su manual de procedimientos.

No necesita especificar las salidas de una acción o paso. Las salidas están predeterminadas por la acción asociada al paso. Al especificar entradas de paso en los manuales de procedimientos, puede referenciar una o más salidas de un paso anterior. Por ejemplo, puede hacer que la salida de `aws:runInstances` esté disponible para una acción `aws:runCommand` posterior. También puede referenciar las salidas de pasos anteriores en la sección `Output` del manual de procedimientos.

Important

Si ejecuta un flujo de trabajo de automatización que invoca otros servicios mediante un rol de servicio de AWS Identity and Access Management (IAM), tenga en cuenta que el rol de servicio debe configurarse con el permiso necesario para invocar dichos servicios. Este requisito se aplica a todos los manuales de procedimientos de automatización de AWS (manuales de `AWS-*`), como los manuales de procedimientos `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup` y `AWS-RestartEC2Instance`, por nombrar algunos. Este requisito también se aplica a cualquier manual de procedimientos de automatización personalizado que cree para llamar otros Servicios de AWS mediante acciones que llaman a otros servicios. Por ejemplo, si utiliza las acciones `aws:executeAwsApi`, `aws:createStack` o `aws:copyImage`, configure

el rol de servicio con el permiso necesario para invocar dichos servicios. Puede conceder permisos a otros Servicios de AWS mediante la incorporación de una política insertada de IAM al rol. Para obtener más información, consulte (Opcional) Agregar una política insertada de Automatización o una política administrada por el cliente para invocar otros Servicios de AWS.

Temas

- Propiedades compartidas por todas las acciones
- aws:approve: detener una automatización para la aprobación manual
- aws:assertAwsResourceProperty: confirmar el estado de un recurso o un evento de AWS
- aws:branch: ejecutar pasos de automatización condicionales
- aws:changeInstanceState: cambiar o confirmar el estado de la instancia
- aws:copyImage: copiar o cifrar una Amazon Machine Image
- aws:createImage: crear una Amazon Machine Image
- aws:createStack: crear una pila de AWS CloudFormation
- aws:createTags: crear etiquetas para recursos de AWS
- aws:deleteImage: eliminar una Amazon Machine Image
- aws:deleteStack: eliminar una pila de AWS CloudFormation
- aws:executeAutomation: ejecutar otra automatización
- aws:executeAwsApi: llamar y ejecutar operaciones de la API de AWS
- aws:executeScript: ejecutar un script
- aws:executeStateMachine: ejecutar una máquina de estado de AWS Step Functions
- aws:invokeWebhook: invocar una integración de webhook de Automation
- aws:invokeLambdaFunction: invocar una función de AWS Lambda
- aws:loop — Repita los pasos en una automatización
- aws:pause: detener una automatización
- aws:runCommand: ejecutar un comando en una instancia administrada
- aws:runInstances: lanzar una instancia de Amazon EC2
- aws:sleep: retrasar una automatización
- aws:updateVariable — Actualiza el valor de una variable del manual de procedimientos
- aws:waitForAwsResourceProperty: esperar una propiedad de recurso de AWS

- Variables del sistema de Automation

Propiedades compartidas por todas las acciones

Las propiedades comunes son parámetros u opciones que se encuentran en todas las acciones. Algunas opciones definen el comportamiento de un paso, como cuánto tiempo debe esperar para que se complete un paso y qué hacer si el paso produce un error. Las siguientes propiedades son comunes a todas las acciones.

description

Información que proporciona para describir el propósito de un manual de procedimientos o un paso.

Tipo: cadena

Requerido: no

name

Un identificador que debe ser único en todos los nombres de paso del manual de procedimientos.

Tipo: cadena

Patrón permitido: [a-zA-Z0-9_]+$

Obligatorio: sí

action

El nombre de la acción que ejecutará el paso. `aws:runCommand`: ejecutar un comando en una instancia administrada es un ejemplo de una acción que puede especificar aquí. En este documento, se proporciona información detallada sobre todas las acciones disponibles.

Tipo: cadena

Obligatorio: sí

maxAttempts

El número de veces que se debe reintentar el paso en caso de error. Si el valor es mayor que 1, no se considerará que hay un error en el paso hasta que todos los reintentos produzcan errores. El valor predeterminado es 1.

Tipo: entero

Obligatorio: no

timeoutSeconds

El valor de tiempo de espera del paso. Si se alcanza el tiempo de espera y el valor de `maxAttempts` es mayor que 1, no se considera que se ha agotado el tiempo de espera del paso hasta que se hayan realizado todos los reintentos.

Tipo: entero

Obligatorio: no

onFailure

Indica si la automatización se debe detener, continuar o ir a otro paso en caso de error. El valor predeterminado para esta opción es abort (anular).

Tipo: cadena

Valores válidos: Abort | Continue | step:*`nombre_del_paso`*

Obligatorio: no

onCancel

Indica a qué paso debe ir la automatización en caso de que un usuario la cancele. Automation ejecuta el flujo de trabajo de cancelación durante un máximo de dos minutos.

Tipo: cadena

Valores válidos: Abort | step:*`step_name`*

Obligatorio: no

La propiedad `onCancel` no admite pasar a las siguientes acciones:

- `aws:approve`
- `aws:copyImage`
- `aws:createImage`
- `aws:createStack`
- `aws:createTags`
- `aws:loop`
- `aws:pause`

- `aws:runInstances`
- `aws:sleep`

isEnd

Esta opción detiene una automatización al final de un paso determinado. La automatización se detiene tanto si el paso genera un error como si se realiza correctamente. El valor predeterminado es false.

Tipo: Booleano

Valores válidos: true | false

Obligatorio: no

nextStep

Especifica qué paso de una automatización se debe procesar después de finalizar correctamente un paso.

Tipo: cadena

Requerido: no

isCritical

Designa un paso como crítico para la finalización correcta de Automation. Si un paso con esta designación genera un error, Automation considera que el estado final de la Automation ha generado un error. Esta propiedad solo se evalúa si la define de forma explícita en el paso. Si la propiedad `onFailure` está establecida en `Continue` en un paso, el valor adopta la opción predeterminada false. De lo contrario, el valor predeterminado para esta opción es true.

Tipo: Booleano

Valores válidos: true | false

Obligatorio: no

inputs

Las propiedades específicas de la acción.

Tipo: mapa

Obligatorio: sí

Ejemplo

```
---
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`: detener una automatización para la aprobación manual

Detiene temporalmente una automatización hasta que las entidades principales designadas aprueben o rechacen la acción. Después de que se alcanza el número necesario de aprobaciones, se reanuda la automatización. Puede insertar el paso de aprobación en cualquier lugar de la sección `mainSteps` de su manual de procedimientos.

> **Note**
>
> Esta acción no admite automatizaciones de varias cuentas ni de regiones. El tiempo de espera predeterminado para esta acción es de 7 días (604 800 segundos) y el valor máximo

es de 30 días (2 592 000 segundos). Puede limitar o ampliar el tiempo de espera mediante la especificación del parámetro `timeoutSeconds` para un paso `aws:approve`.

En el siguiente ejemplo, la acción `aws:approve` detiene temporalmente la automatización hasta que un aprobador la acepte o la rechace. Tras la aprobación, la automatización ejecuta un comando PowerShell sencillo.

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

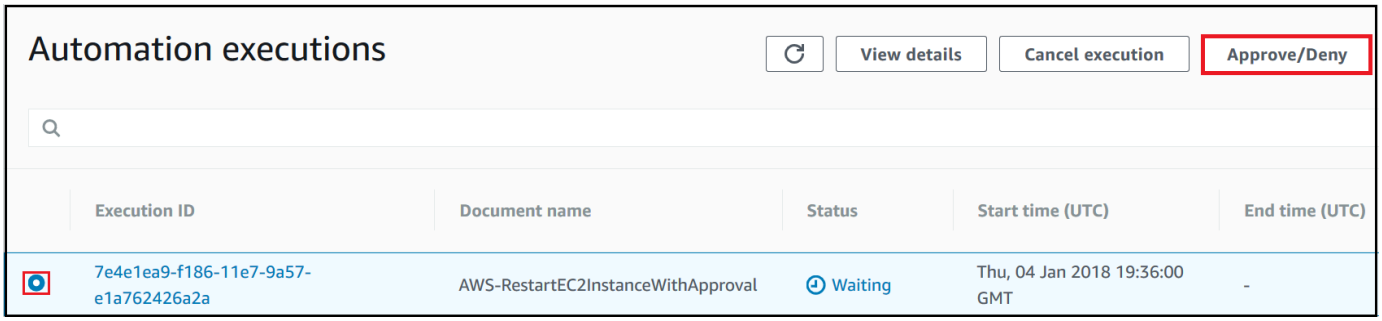
```
{
   "description":"RunInstancesDemo1",
   "schemaVersion":"0.3",
   "assumeRole":"{{ assumeRole }}",
   "parameters":{
      "assumeRole":{
         "type":"String"
      },
      "message":{
         "type":"String"
      }
   },
   "mainSteps":[
      {
         "name":"approve",
         "action":"aws:approve",
         "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"
               ]
            }
         }
```

```
        }
    ]
}
```

Puede aprobar o denegar las instancias de Automation pendientes de aprobación en la consola.

Para aprobar o denegar las automatizaciones en espera

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija automatización.
3. Elija la opción junto a una automatización cuyo estado sea En espera.



4. Elija Aprobar/Denegar.
5. Revise los detalles de Automation.
6. Elija Aprobar o Denegar, escriba un comentario opcional y, a continuación, elija Enviar.

Ejemplo de entrada

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

El nombre de recurso de Amazon (ARN) de un tema de Amazon Simple Notification Service (Amazon SNS) para aprobaciones de Automation. Al especificar un paso `aws:approve` en un manual de procedimientos, Automation envía un mensaje a este tema con el cual se informa a las entidades principales que deben aprobar o rechazar un paso de Automation. El título del tema de Amazon SNS debe tener el prefijo “Automation”.

Tipo: cadena

Requerido: no

Mensaje

La información que desea incluir en el tema de Amazon SNS cuando se envía la solicitud de aprobación. La longitud máxima del mensaje es de 4096 caracteres.

Tipo: cadena

Requerido: no

MinRequiredApprovals

El número mínimo de aprobaciones requeridas para reanudar la automatización. Si no especifica un valor, el sistema utiliza el valor predeterminado de uno. El valor de este parámetro debe ser un número positivo. El valor de este parámetro no puede superar el número de aprobadores definidos por el parámetro `Approvers`.

Tipo: entero

Requerido: no

Approvers

Una lista de entidades principales autenticadas de AWS que pueden aprobar o rechazar la acción. El número máximo de aprobadores es 10. Puede especificar entidades principales mediante cualquiera de los formatos siguientes:

- Un nombre de usuario
- Un ARN de usuario
- un ARN de rol de IAM
- Un ARN de rol de asunción de IAM

Tipo: StringList

Obligatorio: sí

EnhancedApprovals

Esta entrada solo se utiliza para las plantillas de Change Manager. Una lista de las entidades principales autenticadas de AWS que pueden aprobar o rechazar la acción, el tipo de entidad principal de IAM y el número mínimo de aprobadores. A continuación, se muestra un ejemplo:

```
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
```

Tipo: StringList

Obligatorio: sí

Salida

ApprovalStatus

El estado de aprobación del paso. El estado puede ser uno de los siguientes: Approved, Rejected o Waiting. "Waiting" significa que Automation está esperando la entrada de los aprobadores.

Tipo: cadena

ApproverDecisions

Un mapa JSON que incluye la decisión de aprobación de cada aprobador.

Tipo: MapList

`aws:assertAwsResourceProperty`: confirmar el estado de un recurso o un evento de AWS

La acción `aws:assertAwsResourceProperty` le permite confirmar el estado de un recurso o de un evento específico para determinado paso de Automation.

Para más ejemplos sobre cómo usar esta acción, consulte Ejemplos adicionales de manuales de procedimientos.

Entrada

Las entradas se definen con la operación de la API que elija.

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"
    ]
  }
}
```

Servicio

El espacio de nombres del Servicio de AWS que contiene la operación de la API que desea ejecutar. Por ejemplo, el espacio de nombres para Systems Manager es `ssm`. El espacio de nombres para Amazon EC2 es `ec2`. Puede ver una lista de espacios de nombres de Servicio de AWS admitidos en la sección Available Services (Servicios disponibles) de la Referencia de los comandos de la AWS CLI.

Tipo: cadena

Obligatorio: sí

API

El nombre de la operación de la API que desea ejecutar. Puede ver las operaciones de la API (también llamadas métodos) si elige un servicio en el panel de navegación ubicado a la izquierda, en la siguiente página: Services Reference. Elija un método en la sección Cliente para el servicio que desea invocar. Por ejemplo, todas las operaciones de la API (los métodos) para Amazon Relational Database Service (Amazon RDS) se indican en la siguiente página: métodos de Amazon RDS.

Tipo: cadena

Obligatorio: sí

Entradas de la operación de la API

Una o más entradas de la operación de la API. Puede ver las entradas disponibles (también llamadas parámetros) eligiendo un servicio en el panel de navegación izquierdo en la siguiente página de referencia de servicios. Elija un método en la sección Cliente para el servicio que desea invocar. Por ejemplo, todos los métodos para Amazon RDS se indican en la siguiente página: métodos de Amazon RDS. Elija el método describe_db_instances y desplácese hacia abajo para ver los parámetros disponibles, como, por ejemplo, DBInstanceIdentifier, Name y Values. Utilice el formato siguiente para especificar más de una entrada.

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"
}
```

Tipo: se determina a partir de la operación de la API elegida

Obligatorio: sí

PropertySelector

El elemento JSONPath a un atributo específico en el objeto de respuesta. Puede ver los objetos de respuesta eligiendo un servicio en el panel de navegación izquierdo en la siguiente página de referencia de servicios. Elija un método en la sección Cliente para el servicio que desea invocar. Por ejemplo, todos los métodos para Amazon RDS se indican en la siguiente página: métodos de Amazon RDS. Elija el método describe_db_instances y desplácese hasta la sección Response Structure (Estructura de respuesta). DBInstances aparece como un objeto de respuesta.

Tipo: cadena

Obligatorio: sí

DesiredValues

El estado previsto o el estado en el que continuará la automatización. Si especifica un valor booleano, debe utilizar una letra mayúscula como Verdadero o Falso.

Tipo: StringList

Obligatorio: sí

`aws:branch`: ejecutar pasos de automatización condicionales

La acción `aws:branch` le permite crear una automatización dinámica que evalúa diferentes elecciones en un solo paso y, a continuación, salta a otro paso en el manual de procedimientos en función de los resultados de dicha evaluación.

Cuando se especifica la acción `aws:branch` para un paso, se especifican `Choices` que la automatización debe evaluar. Las `Choices` pueden basarse en un valor que especificó en la sección `Parameters` del manual de procedimientos o en un valor dinámico generado como la salida del paso anterior. La automatización evalúa cada elección mediante una expresión booleana. Si la primera elección es true, la automatización saltará al paso designado para esa elección. Si la primera elección es false, la automatización evaluará la siguiente elección. La automatización sigue evaluando cada elección hasta que procese una elección true. A continuación, la automatización saltará al paso designado para la elección true.

Si ninguna de las elecciones es true, la automatización comprueba si el paso contiene un valor `default`. Un valor predeterminado define un paso al cual la automatización debe saltar si ninguna de las elecciones es true. Si no se especifica un valor `default` para el paso, la automatización procesará el siguiente paso en el manual de procedimientos.

La acción `aws:branch` admite evaluaciones de elecciones complejas mediante una combinación de operadores `And`, `Not` y `Or`. Para obtener más información acerca de cómo utilizar `aws:branch`, así como ejemplos de manuales de procedimientos y ejemplos que utilizan diferentes operadores, consulte Uso de instrucciones condicionales en manuales de procedimientos.

Entrada

Especifique una o más `Choices` en un paso. Las `Choices` pueden basarse en un valor que especificó en la sección `Parameters` del manual de procedimientos o en un valor dinámico

generado como la salida del paso anterior. A continuación se muestra un ejemplo de YAML que evalúa un parámetro.

```
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
```

A continuación se muestra un ejemplo de YAML que evalúa la salida de un paso anterior.

```
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
```

Elecciones

Una o más expresiones que la Automation debe evaluar a la hora de determinar el siguiente paso que procesar. Las elecciones se evalúan mediante una expresión booleana. Cada elección debe definir las siguientes opciones:

- NextStep: el siguiente paso en el manual de procedimientos que se debe procesar si la elección designada es true.
- Variable: especifique el nombre de un parámetro que se define en la sección `Parameters` del manual de procedimientos. O bien, especifique un objeto de salida de un paso anterior del manual de procedimientos. Para obtener más información sobre cómo crear variables para `aws:branch`, consulte Acerca de la creación de la variable de salida.
- Operation: los criterios utilizados para evaluar la elección. La acción `aws:branch` admite las siguientes operaciones:

  Operaciones de cadena

  - StringEquals
  - EqualsIgnoreCase
  - StartsWith
  - EndsWith
  - Contiene

  Operaciones numéricas

  - NumericEquals
  - NumericGreater
  - NumericLesser
  - NumericGreaterOrEquals
  - NumericLesser
  - NumericLesserOrEquals

  Operación booleana

  - BooleanEquals

  > **Important**
  >
  > Al crear un manual de procedimientos, el sistema valida cada operación del manual. Si no se admite una operación, el sistema devuelve un error cuando intenta crear el manual de procedimientos.

Predeterminado

El nombre de un paso al que debe saltar la automatización si ninguna de las `Choices` es true.

Tipo: cadena

Requerido: no

> **Note**
>
> La acción `aws:branch` admite los operadores `And`, `Or` y `Not`. Para ver ejemplos de `aws:branch` que utilizan operadores, consulte Uso de instrucciones condicionales en manuales de procedimientos.

`aws:changeInstanceState`: cambiar o confirmar el estado de la instancia

Cambia o confirma el estado de la instancia.

Esta acción se puede utilizar en el modo de confirmación (no ejecuta la API para cambiar el estado, sino para comprobar que la instancia se encuentre en el estado deseado). Para utilizar el modo de aserción, establezca el parámetro `CheckStateOnly` en true. Este modo es útil al ejecutar el comando Sysprep en Windows, que es un comando asíncrono que puede ejecutar en segundo plano durante mucho tiempo. Puede asegurarse de que la instancia esté detenida antes de crear una Amazon Machine Image (AMI).

> **Note**
>
> El valor del tiempo de espera predeterminado para esta acción es de 3600 segundos (una hora). Puede limitar o ampliar el tiempo de espera mediante la especificación del parámetro `timeoutSeconds` para un paso `aws:changeInstanceState`.

Entrada

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

Los ID de las instancias.

Tipo: StringList

Obligatorio: sí

CheckStateOnly

Si es false, establece el estado de la instancia en el valor deseado. Si es true, confirma el estado deseado mediante sondeo.

Valor predeterminado: `false`

Tipo: Booleano

Requerido: no

DesiredState

El estado deseado. Cuando se establece en `running`, esta acción espera hasta que el estado de Amazon EC2 sea `Running`, el estado de la instancia sea `OK` y el estado del sistema sea `OK` antes de completarse.

Tipo: cadena

Valores válidos: `running` | `stopped` | `terminated`

Obligatorio: sí

Force

Si se establece, obliga a las instancias a detenerse. Las instancias no tienen la oportunidad de vaciar las memorias cachés o los metadatos de los sistemas de archivos. Si utiliza esta opción, debe realizar los procedimientos de comprobación y reparación del sistema de archivos. Esta opción no se recomienda para las instancias EC2 de Windows Server.

Tipo: Booleano

Requerido: no

AdditionalInfo

Reservado.

Tipo: cadena

Requerido: no

Salida

Ninguna

`aws:copyImage`: copiar o cifrar una Amazon Machine Image

Copia una Amazon Machine Image (AMI) de cualquier Región de AWS en la región actual. Esta acción también puede cifrar la nueva AMI.

Entrada

Esta acción admite la mayoría de los parámetros `CopyImage`. Para obtener más información, consulte CopyImage.

En el siguiente ejemplo, se crea una copia de una AMI en la región de Seúl (`SourceImageID`: ami-0fe10819. `SourceRegion`: ap-northeast-2). La nueva AMI se copia en la región en la que inició la acción de Automation. La AMI copiada se cifrará porque la marca `Encrypted` opcional se ha configurado en `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

La región en la que se encuentra la AMI de origen.

Tipo: cadena

Obligatorio: sí

SourceImageId

El ID de la AMI que se copiará de la región de origen.

Tipo: cadena

Obligatorio: sí

ImageName

El nombre de la nueva imagen.

Tipo: cadena

Obligatorio: sí

ImageDescription

Una descripción de la imagen de destino.

Tipo: cadena

Requerido: no

Encriptado

Cifre la AMI de destino.

Tipo: Booleano

Requerido: no

KmsKeyId

El nombre de recurso de Amazon (ARN) completo de la AWS KMS key que se utilizará al cifrar las instantáneas de una imagen durante una operación de copia. Para obtener más información, consulte CopyImage.

Tipo: cadena

Requerido: no

ClientToken

Un identificador con distinción entre mayúsculas y minúsculas único que proporciona para garantizar la idempotencia de la solicitud. Para obtener más información, consulte CopyImage.

Tipo: cadena

Requerido: no

Salida

ImageId

El ID de la imagen copiada.

ImageState

El estado de la imagen copiada.

Valores válidos: `available` | `pending` | `failed`

`aws:createImage`: crear una Amazon Machine Image

Crea una Amazon Machine Image (AMI) a partir de una instancia que está en ejecución, deteniéndose o detenida, y sondea para que `ImageState` esté `available`.

Entrada

Esta acción es compatible con los siguientes parámetros de `CreateImage`. Para obtener más información, consulte 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

El ID de la instancia.

Tipo: cadena

Obligatorio: sí

ImageName

El nombre de la imagen.

Tipo: cadena

Obligatorio: sí

ImageDescription

Una descripción de la imagen.

Tipo: cadena

Requerido: no

NoReboot

Un literal booleano.

De forma predeterminada, Amazon Elastic Compute Cloud (Amazon EC2) intenta apagar y reiniciar la instancia antes de crear la imagen. Si la opción No reboot (Sin reiniciar) se establece en `true`, Amazon EC2 no apaga la instancia antes de crear la imagen. Cuando se utiliza esta opción, no se puede garantizar la integridad del sistema de archivos en la imagen creada.

Si no desea que la instancia se ejecute después de crear una AMI a partir de ella, primero debe usar la acción `aws:changeInstanceState`: cambiar o confirmar el estado de la instancia para detener la instancia y, a continuación, usar la acción `aws:createImage` con la opción NoReboot establecida en `true`.

Tipo: Booleano

Requerido: no

BlockDeviceMappings

Los dispositivos de bloques para la instancia.

Tipo: mapa

Requerido: no

Salida

ImageId

El ID de la imagen recién creada.

Tipo: cadena

ImageState

El estado actual de la imagen. Si el estado está disponible, la imagen se registra correctamente y se puede utilizar para iniciar una instancia.

Tipo: cadena

`aws:createStack`: crear una pila de AWS CloudFormation

Crea una pila de AWS CloudFormation a partir de una plantilla.

Para obtener información adicional acerca de la creación de pilas de CloudFormation, consulte CreateStack en la Referencia de la API de AWS CloudFormation.

Entrada

YAML

```
name: makeStack
action: aws:createStack
maxAttempts: 1
onFailure: Abort
inputs:
  Capabilities:
```

```
  - CAPABILITY_IAM
  StackName: myStack
  TemplateURL: http://s3.amazonaws.com/amzn-s3-demo-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/amzn-s3-demo-bucket/
myStackTemplate",
        "TimeoutInMinutes": 5,
        "Parameters": [
          {
            "ParameterKey": "LambdaRoleArn",
            "ParameterValue": "{{LambdaAssumeRole}}"
          },
          {
            "ParameterKey": "createdResource",
            "ParameterValue": "createdResource-{{automation:EXECUTION_ID}}"
          }
    }
}
```

Capacidades

Una lista de valores que especifica para que CloudFormation pueda crear determinadas pilas. Algunas plantillas de pila incluyen recursos que pueden afectar los permisos de Cuenta de AWS. Para estas pilas, debe reconocer explícitamente sus capacidades especificando este parámetro.

Los valores válidos son `CAPABILITY_IAM`, `CAPABILITY_NAMED_IAM` y `CAPABILITY_AUTO_EXPAND`.

CAPABILITY_IAM y CAPABILITY_NAMED_IAM

Si tiene recursos de IAM, puede especificar cualquiera de las dos capacidades. Si tiene recursos de IAM con nombres personalizados, debe especificar `CAPABILITY_NAMED_IAM`. Si no especifica este parámetro, esta acción devuelve un error `InsufficientCapabilities`. Los siguientes recursos requieren que especifique `CAPABILITY_IAM` o `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 su plantilla de pila contiene estos recursos, le recomendamos que revise todos los permisos asociados con ellos y edite sus permisos, si es necesario.

Para obtener más información, consulte Reconocimiento de recursos de IAM en plantillas de AWS CloudFormation.

CAPABILITY_AUTO_EXPAND

Algunas plantillas contienen macros. Las macros realizan procesamiento personalizado en las plantillas; esto puede incluir desde acciones sencillas como operaciones de búsqueda y reemplazo hasta amplias transformaciones de plantillas completas. Debido a esto, los usuarios suelen crear un conjunto de cambios a partir de la plantilla procesada, de modo que puedan revisar los cambios resultantes de las macros antes de proceder a crear la pila. Si la plantilla de pila contiene una o más macros y decide crear una pila directamente a partir de la plantilla procesada, sin revisar primero los cambios resultantes en un conjunto de cambios, deberá confirmar esta prestación.

Para obtener más información, consulte Uso de macros de AWS CloudFormation para realizar un procesamiento personalizado en plantillas en la Guía del usuario de AWS CloudFormation.

Tipo: matriz de cadenas

Valores válidos: `CAPABILITY_IAM | CAPABILITY_NAMED_IAM | CAPABILITY_AUTO_EXPAND`

Obligatorio: no

ClientRequestToken

Identificador exclusivo de esta solicitud CreateStack. Especifique este token si establece maxAttempts en este paso en un valor mayor que 1. Al especificar este token, CloudFormation sabe que no está intentando crear una pila nueva con el mismo nombre.

Tipo: cadena

Requerido: no

Limitaciones de longitud: longitud mínima de 1. Longitud máxima de 128.

Patrón: [a-zA-Z0-9][-a-zA-Z0-9]*

DisableRollback

Se configura en `true` para desactivar la restauración de la pila si se produce un error al crearla.

Condicional: puede especificar el parámetro `DisableRollback` o `OnFailure`, pero no ambos.

Valor predeterminado: `false`

Tipo: Booleano

Obligatorio: no

NotificationARNs

Los ARN de temas de Amazon Simple Notification Service (Amazon SNS) utilizados para publicar eventos relacionados con la pila. Puede encontrar ARN de temas de SNS con la consola de Amazon SNS, https://console.aws.amazon.com/sns/v3/home.

Tipo: matriz de cadenas

Miembros de la matriz: número máximo de 5 elementos.

Obligatorio: no

OnFailure

Determina la acción que se realizará si se produce un error en la creación de la pila. Debe especificar `DO_NOTHING`, `ROLLBACK` o `DELETE`.

Condicional: puede especificar el parámetro `OnFailure` o `DisableRollback`, pero no ambos.

Valor predeterminado: `ROLLBACK`

Tipo: cadena

Valores válidos: `DO_NOTHING | ROLLBACK | DELETE`

Obligatorio: no

Parámetros

Una lista de estructuras `Parameter` que especifican los parámetros de entrada de la pila. Para obtener más información, consulte el tipo de datos Parameter.

Tipo: matriz de objetos Parameter

Obligatorio: no

ResourceTypes

Los tipos de recurso de plantilla para los que tiene permiso para trabajar con ellos para esta acción de creación de la pila. Por ejemplo: `AWS::EC2::Instance`, `AWS::EC2::*` o `Custom::MyCustomInstance`. Utilice la siguiente sintaxis para describir los tipos de recurso de plantilla.

- Para todos los recursos de AWS:

```
AWS::*
```

- Para todos los recursos personalizados:

```
Custom::*
```

- Para un recurso personalizado específico:

```
Custom::logical_ID
```

- Para todos los recursos de un Servicio de AWS concreto:

```
AWS::service_name::*
```

- Para un recurso de AWS específico:

```
AWS::service_name::resource_logical_ID
```

Si la lista de tipos de recursos no incluye un recurso que está va a crear, se produce un error en la creación de la pila. De forma predeterminada, CloudFormation concede permisos a todos los tipos de recursos. IAM utiliza este parámetro para las claves de condición específicas de CloudFormation en las políticas de IAM. Para obtener más información, consulte Control del acceso con AWS Identity and Access Management.

Tipo: matriz de cadenas

Limitaciones de longitud: longitud mínima de 1. La longitud máxima es de 256 caracteres.

Obligatorio: no

RoleARN

El nombre de recurso de Amazon (ARN) de un rol de (IAM) que CloudFormation asume al crear la pila. CloudFormation utiliza las credenciales del rol para realizar llamadas en su nombre. CloudFormation siempre utiliza este rol para todas las operaciones futuras en la pila. Siempre que los usuarios tengan permiso para operar en la pila, CloudFormation utiliza este rol, aunque los usuarios no tengan permiso para transmitirlo. Asegúrese de que el rol concede la menor cantidad de privilegios.

Si no especifica un valor, CloudFormation utiliza el rol que se había asociado anteriormente a la pila. Si no hay ningún rol disponible, CloudFormation utiliza una sesión temporal que se genera a partir de sus credenciales de usuario.

Tipo: cadena

Limitaciones de longitud: longitud mínima de 20. La longitud máxima es de 2048 caracteres.

Obligatorio: no

StackName

El nombre que está asociado a la pila. El nombre debe ser único en la región en la que se crea la pila.

> **Note**
>
> El nombre de una pila solo puede contener caracteres alfanuméricos (con distinción de mayúsculas y minúsculas) y guiones. Debe comenzar con un carácter alfabético y no puede tener más de 128 caracteres.

Tipo: cadena

Obligatorio: sí

StackPolicyBody

Estructura que contiene el cuerpo de políticas de la pila. Para obtener más información, consulte Evitar actualizaciones en los recursos de la pila.

Condicional: puede especificar el parámetro `StackPolicyBody` o `StackPolicyURL`, pero no ambos.

Tipo: cadena

Limitaciones de longitud: longitud mínima de 1. La longitud máxima es de 16384 caracteres.

Obligatorio: no

StackPolicyURL

Ubicación de un archivo que contiene la política de la pila. La URL debe apuntar a una política ubicada en un bucket de &S3; en la misma región que el stack. El tamaño de archivo máximo permitido para la política de la pila es de 16 KB.

Condicional: puede especificar el parámetro `StackPolicyBody` o `StackPolicyURL`, pero no ambos.

Tipo: cadena

Limitaciones de longitud: longitud mínima de 1. La longitud máxima es de 1350 caracteres.

Obligatorio: no

Etiquetas

Pares clave-valor para asociar con este stack. CloudFormation también propaga estas etiquetas a los recursos que se crean en la pila. Puede especificar un número máximo de 10 etiquetas.

Tipo: matriz de objetos Tag

Obligatorio: no

TemplateBody

Estructura que contiene el cuerpo de la plantilla con una longitud mínima de 1 byte y una longitud máxima de 51.200 bytes. Para obtener más información, consulte Anatomía de la plantilla.

Condicional: puede especificar el parámetro `TemplateBody` o `TemplateURL`, pero no ambos.

Tipo: cadena

Limitaciones de longitud: longitud mínima de 1.

Obligatorio: no

TemplateURL

Ubicación de un archivo que contiene el cuerpo de la plantilla. La URL debe apuntar a una plantilla que se encuentre en un bucket de S3. El tamaño máximo permitido para la plantilla es de 460.800 bytes. Para obtener más información, consulte Anatomía de la plantilla.

Condicional: puede especificar el parámetro `TemplateBody` o `TemplateURL`, pero no ambos.

Tipo: cadena

Limitaciones de longitud: longitud mínima de 1. La longitud máxima es de 1024 caracteres.

Obligatorio: no

TimeoutInMinutes

Periodo de tiempo que puede transcurrir antes de que el estado de la pila se convierta en `CREATE_FAILED`. Si no se ha establecido `DisableRollback` o se ha establecido en `false`, se restaurará la pila.

Tipo: entero

Rango válido: valor mínimo de 1.

Obligatorio: no

Salidas

StackId

Identificador único de la pila.

Tipo: cadena

StackStatus

Estado actual de la pila.

Tipo: cadena

Valores válidos: `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`

Obligatorio: sí

StackStatusReason

Mensaje de éxito o error asociado con el estado de la pila.

Tipo: cadena

Requerido: no

Para obtener más información, consulte CreateStack.

Consideraciones de seguridad

Para poder usar la acción `aws:createStack`, debe asignar la siguiente política al rol de asunción de Automation de IAM. Para obtener más información sobre el rol de asunción, consulte Tarea 1: crear un rol de servicio para Automation.

JSON

```
{
```

```
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "sqs:*",
            "cloudformation:CreateStack",
            "cloudformation:DescribeStacks"
         ],
         "Resource":"*"
      }
   ]
}
```

`aws:createTags`: crear etiquetas para recursos de AWS

Crea etiquetas nuevas para instancias de Amazon Elastic Compute Cloud (Amazon EC2) o instancias administradas de AWS Systems Manager.

Entrada

Esta acción admite la mayoría de los parámetros `CreateTags` de Amazon EC2 y `AddTagsToResource` de Systems Manager. Para obtener más información, consulte CreateTags y AddTagsToResource.

En el siguiente ejemplo, se muestra cómo etiquetar una Amazon Machine Image (AMI) y una instancia como recursos de producción para un determinado departamento.

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

Los ID de los recursos que se van a etiquetar. Si el tipo de recurso no es “EC2”, este campo solo puede contener un único elemento.

Tipo: lista de cadenas

Obligatorio: sí

Etiquetas

Las etiquetas para asociarlas con los recursos.

Tipo: lista de mapas

Obligatorio: sí

Tipo de recurso

El tipo de los recursos que se van a etiquetar. Si no se suministra, se usa el valor predeterminado "EC2".

Tipo: cadena

Requerido: no

Valores válidos: `EC2` | `ManagedInstance` | `MaintenanceWindow` | `Parameter`

Salida

Ninguna

`aws:deleteImage`: eliminar una Amazon Machine Image

Elimina la Amazon Machine Image (AMI) especificada y todas las instantáneas relacionadas.

Entrada

Esta acción solo admite un parámetro. Para obtener más información, consulte la documentación de DeregisterImage y 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

El ID de la imagen que se va a eliminar.

Tipo: cadena

Obligatorio: sí

Salida

Ninguna

`aws:deleteStack`: eliminar una pila de AWS CloudFormation

Elimina una pila de AWS CloudFormation.

Entrada

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

Un identificador único para esta solicitud `DeleteStack`. Especifique este token si tiene previsto intentar completar las solicitudes otra vez para que CloudFormation sepa que no está intentando eliminar una pila con el mismo nombre. Puede intentar completar las solicitudes `DeleteStack` otra vez para verificar que CloudFormation las ha recibido.

Tipo: cadena

Limitaciones de longitud: longitud mínima de 1. Longitud máxima de 128.

Patrón: [a-zA-Z][-a-zA-Z0-9]*

Obligatorio: no

RetainResources.member.N

Esta entrada solo se aplica a las pilas que se encuentran en un estado `DELETE_FAILED`. Una lista de los ID de recurso lógico en el caso de los recursos que desea conservar. Durante la eliminación, CloudFormation elimina la pila, pero no los recursos que se conservan.

La retención de recursos resulta útil cuando no se puede eliminar un recurso, como un bucket de &S3; vacío, pero desea eliminar el stack.

Tipo: matriz de cadenas

Obligatorio: no

RoleARN

El nombre de recurso de Amazon (ARN) de un rol de AWS Identity and Access Management (IAM) que CloudFormation asume para crear la pila. CloudFormation utiliza las credenciales del rol para realizar llamadas en su nombre. CloudFormation siempre utiliza este rol para todas las operaciones futuras en la pila. Siempre que los usuarios tengan permiso para operar en la pila, CloudFormation utiliza este rol, aunque los usuarios no tengan permiso para transmitirlo. Asegúrese de que el rol concede la menor cantidad de privilegios.

Si no especifica un valor, CloudFormation utiliza el rol que se había asociado anteriormente a la pila. Si no hay ningún rol disponible, CloudFormation utiliza una sesión temporal que se genera a partir de sus credenciales de usuario.

Tipo: cadena

Limitaciones de longitud: longitud mínima de 20. La longitud máxima es de 2048 caracteres.

Obligatorio: no

StackName

El nombre o el ID de pila único que está asociado con la pila.

Tipo: cadena

Obligatorio: sí

Consideraciones de seguridad

Para poder usar la acción `aws:deleteStack`, debe asignar la siguiente política al rol de asunción de Automation de IAM. Para obtener más información sobre el rol de asunción, consulte Tarea 1: crear un rol de servicio para Automation.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "sqs:*",
            "cloudformation:DeleteStack",
            "cloudformation:DescribeStacks"
         ],
         "Resource":"*"
      }
   ]
}
```

`aws:executeAutomation`: ejecutar otra automatización

Ejecuta una automatización secundaria mediante la llamada a un manual de procedimientos secundario. Con esta acción, puede crear manuales de procedimientos para sus operaciones

más comunes y puede referenciar esos manuales durante una automatización. Esta acción puede simplificar los manuales de procedimientos, ya que se elimina la necesidad de duplicar pasos en manuales similares.

La automatización secundaria se ejecuta en el contexto del usuario que ha iniciado la automatización principal. Esto significa que la automatización secundaria utiliza el mismo rol o usuario de AWS Identity and Access Management (IAM) que el usuario que inició la primera automatización.

> **Important**
>
> Si especifica parámetros en una automatización secundaria que use un rol de asunción (un rol que usa la política iam:passRole), el usuario o el rol que hayan iniciado la automatización principal deben tener permiso para transmitir el rol de asunción especificado en la automatización secundaria. Para obtener más información sobre la configuración de un rol de asunción para Automation, consulte Cree los roles de servicio para la automatización mediante la consola.

Entrada

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

El nombre del manual de procedimientos secundario que se ejecutará durante el paso. Para ver manuales de procedimientos en la misma Cuenta de AWS, especifique el nombre del manual. Para ver manuales de procedimientos compartidos desde una Cuenta de AWS diferente, especifique el nombre de recurso de Amazon (ARN) del manual. Para obtener información acerca del uso de manuales de procedimientos compartidos, consulte Uso de documentos de SSM compartidos.

Tipo: cadena

Obligatorio: sí

DocumentVersion

La versión del manual de procedimientos secundario que se ejecutará. Si no se especifica, Automation ejecuta la versión predeterminada del manual de procedimientos.

Tipo: cadena

Requerido: no

MaxConcurrency

El número máximo de destinos que pueden ejecutar esta tarea en paralelo. Puede especificar un número, como 10, o un porcentaje, como 10 %.

Tipo: cadena

Requerido: no

MaxErrors

Número de errores permitidos antes de que el sistema deje de ejecutar la automatización en destinos adicionales. Puede especificar un número absoluto de errores, por ejemplo, 10 o un porcentaje del destino definido, por ejemplo, el 10 %. Si especifica 3, por ejemplo, el sistema dejará de ejecutar la automatización una vez que se reciba el cuarto error. Si especifica 0, el sistema dejará de ejecutar la automatización en otros destinos una vez que se reciba el primer resultado de error. Si ejecuta una automatización en 50 recursos y establece `MaxErrors` al 10 %, el sistema dejará de ejecutar la automatización en destinos adicionales una vez que se reciba el sexto error.

Las automatizaciones que ya se están ejecutando cuando se alcanza un umbral de `MaxErrors` tienen permiso para completarse, pero es posible que algunas de ellas también presenten un error. Si necesita asegurarse de que no se produzcan más errores en las automatizaciones más allá del valor de `MaxErrors` especificado, establezca `MaxConcurrency` en 1 de modo que las automatizaciones se procesen de una en una.

Tipo: cadena

Requerido: no

RuntimeParameters

Parámetros obligatorios para el manual de procedimientos secundario. El mapeo usa el formato siguiente: {"parámetro1" : "valor1", "parámetro2" : "valor2" }

Tipo: mapa

Obligatorio: no

Etiquetas

Metadatos opcionales que se asignan a un recurso. Puede especificar un máximo de cinco etiquetas para una automatización.

Tipo: MapList

Obligatorio: no

TargetLocations

Una ubicación es una combinación de Regiones de AWS o Cuentas de AWS en las que desea ejecutar la automatización. Se debe especificar 1 elemento como mínimo y 100 como máximo.

Al especificar un valor para este parámetro, las salidas no se devuelven a la automatización principal. Si es necesario, debe realizar llamadas posteriores a las operaciones de la API para recuperar la salida de las automatizaciones secundarias.

Tipo: MapList

Obligatorio: no

TargetMaps

Lista de asignaciones de clave-valor de los parámetros del documento para indicar los recursos de destino. Tanto `Targets` como `TargetMaps` no se pueden especificar juntos.

Tipo: MapList

Obligatorio: no

TargetParameterName

Nombre del parámetro utilizado como el recurso de destino para la automatización con frecuencia controlada. Es obligatorio si especifica `Targets`.

Tipo: cadena

Requerido: no

Destinos

Lista de asignaciones de clave-valor para indicar recursos como destino. Es obligatorio si especifica `TargetParameterName`.

Tipo: MapList

Obligatorio: no

Output

Output

La salida generada por la automatización secundaria. Puede hacer referencia a la salida mediante el formato siguiente: *`Secondary_Automation_Step_Name`*.Output

Tipo: StringList

A continuación se muestra un ejemplo:

```
- 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

El ID de la automatización secundaria.

Tipo: cadena

Estado

El estado de la automatización secundaria.

Tipo: cadena

`aws:executeAwsApi`: llamar y ejecutar operaciones de la API de AWS

Llama y ejecuta operaciones de la API de AWS. Se admiten la mayoría de las operaciones de la API, aunque no todas se han puesto a prueba. No se admiten las operaciones de la API de streaming, como GetObject. Si no estás seguro de si una operación de API que quiere usar es una operación de streaming, revise la documentación de Boto3 del servicio para determinar si una API requiere entradas o salidas de streaming. Actualizamos periódicamente la versión de Boto3 que utiliza esta acción. Sin embargo, tras el lanzamiento de una nueva versión de Boto3, los cambios pueden tardar varias semanas en reflejarse en esta acción. Cada acción `aws:executeAwsApi` puede ejecutarse hasta un máximo de 25 segundos. Para más ejemplos sobre cómo usar esta acción, consulte Ejemplos adicionales de manuales de procedimientos.

Entradas

Las entradas se definen con la operación de la API que elija.

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"
        }
    ]
}
```

Servicio

El espacio de nombres del Servicio de AWS que contiene la operación de la API que desea ejecutar. Puede ver una lista de espacios de nombres de espacios de Servicio de AWS admitidos en la sección Available services (Servicios disponibles) de AWS SDK para Python (Boto3). El espacio de nombres se encuentra en la sección Cliente. Por ejemplo, el espacio de nombres para Systems Manager es `ssm`. El espacio de nombres para Amazon Elastic Compute Cloud (Amazon EC2) es `ec2`.

Tipo: cadena

Obligatorio: sí

API

El nombre de la operación de la API que desea ejecutar. Puede ver las operaciones de la API (también llamadas métodos) si elige un servicio en el panel de navegación ubicado a la izquierda, en la siguiente página: Services Reference. Elija un método en la sección Cliente para el servicio que desea invocar. Por ejemplo, todas las operaciones de la API (los métodos) para Amazon Relational Database Service (Amazon RDS) se indican en la siguiente página: métodos de Amazon RDS.

Tipo: cadena

Obligatorio: sí

Entradas de la operación de la API

Una o más entradas de la operación de la API. Puede ver las entradas disponibles (también llamadas parámetros) eligiendo un servicio en el panel de navegación izquierdo en la siguiente página de referencia de servicios. Elija un método en la sección Cliente para el servicio que desea invocar. Por ejemplo, todos los métodos para Amazon RDS se indican en la siguiente página: métodos de Amazon RDS. Elija el método describe_db_instances y desplácese hacia

abajo para ver los parámetros disponibles, como, por ejemplo, DBInstanceIdentifier, Name y 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"
}
```

Tipo: se determina a partir de la operación de la API elegida

Obligatorio: sí

Salidas

El usuario especifica las salidas en función de la respuesta de la operación de la API elegida.

Nombre

Un nombre para la salida.

Tipo: cadena

Obligatorio: sí

Selector

El elemento JSONPath a un atributo específico en el objeto de respuesta. Puede ver los objetos de respuesta eligiendo un servicio en el panel de navegación izquierdo en la siguiente página de

referencia de servicios. Elija un método en la sección Cliente para el servicio que desea invocar. Por ejemplo, todos los métodos para Amazon RDS se indican en la siguiente página: métodos de Amazon RDS. Elija el método describe_db_instances y desplácese hasta la sección Response Structure (Estructura de respuesta). DBInstances aparece como un objeto de respuesta.

Tipo: Entero, Booleano, Cadena, StringList, StringMap o MapList

Obligatorio: sí

Tipo

El tipo de datos del elemento de respuesta.

Tipo: varía

Obligatorio: sí

`aws:executeScript`: ejecutar un script

Ejecuta el script Python o PowerShell proporcionado mediante el uso del tiempo de ejecución y el controlador especificados. Cada acción `aws:executeScript` puede ejecutarse hasta un máximo de 600 segundos (10 minutos). Puede limitar el tiempo de espera mediante la especificación del parámetro `timeoutSeconds` para un paso `aws:executeScript`.

Utilice instrucciones de devolución en la función para agregar salidas a la carga útil de salida. Para ver ejemplos sobre cómo definir salidas para la acción `aws:executeScript`, consulte Ejemplo 2: manual de procedimientos con scripts. También puede enviar la salida de acciones `aws:executeScript` de los manuales de procedimientos al grupo de registros de Amazon CloudWatch Logs que especifique. Para obtener más información, consulte Registro de salida de acción de Automation con CloudWatch Logs.

Si desea enviar la salida desde acciones `aws:executeScript` a los Registros de CloudWatch o si los scripts que especifica para la llamada de acciones `aws:executeScript` las operaciones de la API de AWS, siempre se requiere un rol de servicio de AWS Identity and Access Management (IAM) (o asumir un rol) para ejecutar el manual de procedimientos.

La acción `aws:executeScript` contiene los siguientes módulos de PowerShell Core preinstalados:

- Microsoft.PowerShell.Host
- Microsoft.PowerShell.Management

- Microsoft.PowerShell.Security
- Microsoft.PowerShell.Utility
- PackageManagement
- PowerShellGet

Para utilizar módulos de PowerShell Core que no estén preinstalados, el script debe instalar el módulo con la marca `-Force`, como se muestra en el siguiente comando. No se admite el módulo `AWSPowerShell.NetCore`. Reemplace *`ModuleName`* con el módulo que desee instalar.

```
Install-Module ModuleName -Force
```

Para utilizar cmdlets de PowerShell Core en el script, se recomienda utilizar los módulos de `AWS.Tools`, como se muestra en los siguientes comandos. Reemplace cada *`example resource placeholder`* con su propia información.

- Cmdlets de Amazon S3

```
Install-Module AWS.Tools.S3 -Force
Get-S3Bucket -BucketName amzn-s3-demo-bucket
```

- Cmdlets de Amazon EC2

```
Install-Module AWS.Tools.EC2 -Force
Get-EC2InstanceStatus -InstanceId instance-id
```

- Cmdlets de AWS Tools for Windows PowerShell comunes o independientes del servicio

```
Install-Module AWS.Tools.Common -Force
Get-AWSRegion
```

Si el script inicializa nuevos objetos además de usar cmdlets de PowerShell Core, también debe importar el módulo como se muestra en el siguiente comando.

```
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
```

Para obtener ejemplos de instalación e importación de módulos de `AWS.Tools`, y de uso de cmdlets de PowerShell Core en los manuales de procedimientos, consulte Experiencia de diseño visual para manuales de procedimientos de automatización.

Input

Proporcione la información necesaria para ejecutar el script. Reemplace cada *`example resource placeholder`* con su propia información.

> **Note**
>
> El archivo adjunto de un script de Python puede ser un archivo .py o uno .zip que contenga el script. Los scripts de PowerShell deben almacenarse en archivos .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"
    }
}
```

Tiempo de ejecución

El lenguaje del tiempo de ejecución que se utilizará para ejecutar el script proporcionado. `aws:executeScript` es compatible con scripts de 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`), PowerShell Core 6.0 (`dotnetcore2.1`), PowerShell 7.0 (`dotnetcore3.1`) y PowerShell 7.4 (`dotnet8`).

Valores admitidos: **`python3.7`** | **`python3.8`** | **`python3.9`** | **`python3.10`** | **`python3.11`** | **`PowerShell Core 6.0`** | **`PowerShell 7.0`** | **`PowerShell 7.4`**

Tipo: cadena

Obligatorio: sí

> **Note**
>
> Para los tiempos de ejecución de Python, el entorno proporciona 512 MB de memoria y 512 MB de espacio en disco. Para los tiempos de ejecución de PowerShell, el entorno proporciona 1024 MB de memoria y 512 MB de espacio en disco.

Controlador

Nombre de la función. Debe asegurarse de que la función definida en el controlador tenga dos parámetros, `events` y `context`. El tiempo de ejecución de PowerShell no admite este parámetro.

Tipo: cadena

Requerido: Sí (Python) | No admitido (PowerShell)

InputPayload

Un objeto JSON o YAML que se pasará al primer parámetro del controlador. Se puede usar para pasar los datos de entrada al script.

Tipo: cadena

Requerido: no

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: "python3.11"
        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": "ExamplePython"
            }
            print(f"Adding tag {tag} to instance id {instanceId}")
            ec2.create_tags(
                Resources=[instanceId],
                Tags=[tag]
            )
            return tag
```

```
    outputs:
      - Type: String
        Name: TagKey
        Selector: $.Payload.Key
outputs:
  - tagInstance.TagKey
```

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
    isEnd: true
    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 = "ExamplePowerShell"

        Write-Information "Adding tag key: $($tag.Key) and value: $($tag.Value)
 to instance id $($input.instanceId)"
        New-EC2Tag -Resource $input.instanceId -Tag $tag
```

```
        return $tag
    outputs:
      - Type: String
        Name: TagKey
        Selector: $.Payload.Key
outputs:
  - tagInstance.TagKey
```

Script

Un script insertado que desea ejecutar durante la automatización.

Tipo: cadena

Requerido: No (Python) | Sí (PowerShell)

Archivo adjunto

Nombre de un archivo de script independiente o archivo .zip que puede invocarse mediante la acción. Especifique el mismo valor que el `Name` del archivo adjunto de documento que especifique en el parámetro de solicitud `Attachments`. Para obtener más información, consulte Archivos adjuntos en la referencia de la API de AWS Systems Manager. Si proporciona un script mediante un archivo adjunto, también debe definir una sección `files` en los elementos de nivel superior del manual de procedimientos. Para obtener más información, consulte Versión de esquema 0.3.

Si desea invocar un archivo para Python, use el formato `filename.method_name` en `Handler`.

> **Note**
>
> El archivo adjunto de un script de Python puede ser un archivo .py o uno .zip que contenga el script. Los scripts de PowerShell deben almacenarse en archivos .zip.

Al incluir bibliotecas de Python en su archivo adjunto, recomendamos que agregue un archivo `__init__.py` vacío en cada directorio del módulo. Esto le permite importar los módulos desde la biblioteca del archivo adjunto dentro del contenido de su script. Por ejemplo: `from library import module`

Tipo: cadena

Requerido: no

Output

Carga

La representación JSON del objeto devuelta por su función. Se devuelven hasta 100 KB. Si genera una lista, incluirá un máximo de 100 elementos.

Uso de archivos adjuntos con aws:executeScript

Los archivos adjuntos proporcionan una forma eficaz de empaquetar y reutilizar scripts complejos, varios módulos y dependencias externas con sus acciones `aws:executeScript`. Use archivos adjuntos cuando necesite hacer lo siguiente:

- Empaquetar varios módulos de Python o scripts de PowerShell juntos.
- Reutilizar la misma lógica de script en varios manuales de procedimientos.
- Incluir bibliotecas o dependencias externas en sus scripts.
- Mantener limpia la definición del manual de procedimientos separando la lógica de los scripts complejos.
- Compartir paquetes de scripts entre equipos o flujos de trabajo de automatización.

Estructura y empaquetado de los archivos adjuntos

Puede adjuntar archivos individuales o paquetes zip que contengan varios archivos. La estructura depende de su caso de uso:

Archivos adjuntos de un único archivo

En el caso de scripts sencillos, puede adjuntar un único archivo `.py` (Python) o un archivo `.zip` que contenga un único script de PowerShell.

Paquetes de varios módulos

Para una automatización compleja que requiera varios módulos, cree un paquete zip con la siguiente estructura recomendada:

```
my-automation-package.zip
### main.py                    # Entry point script
### utils/
#   ### __init__.py           # Required for Python module imports
```

```
#   ### helper_functions.py   # Utility functions
#   ### aws_operations.py     # AWS-specific operations
### config/
#   ### __init__.py
#   ### settings.py           # Configuration settings
### requirements.txt          # Optional: document dependencies
```

> **Important**
>
> Para los paquetes de Python, debe incluir un archivo `__init__.py` vacío en cada directorio que contenga los módulos de Python. Esto le permite importar módulos utilizando la sintaxis de importación estándar de Python, como `from utils import helper_functions`.

Estructura de paquetes de PowerShell

Los archivos adjuntos de PowerShell deben empaquetarse en archivos zip con la siguiente estructura:

```
my-powershell-package.zip
### Main.ps1                  # Entry point script
### Modules/
#   ### HelperFunctions.ps1   # Utility functions
#   ### AWSOperations.ps1     # AWS-specific operations
### Config/
    ### Settings.ps1          # Configuration settings
```

Crear manuales de procedimientos con archivos adjuntos

Siga estos pasos para crear manuales de procedimientos que utilicen archivos adjuntos:

1. Cargar el archivo adjunto a Amazon S3

   Cargue el archivo de script o el paquete zip en un bucket de S3 al que pueda acceder su rol de automatización. Observe el URI de S3 que se usará en el siguiente paso.

   ```
   aws s3 cp my-automation-package.zip s3://my-automation-bucket/scripts/
   ```

2. Calcular la suma de verificación del archivo adjunto

   Calcule la suma de verificación SHA-256 de su archivo adjunto para la verificación de seguridad:

```
# Linux/macOS
shasum -a 256 my-automation-package.zip

# Windows PowerShell
Get-FileHash -Algorithm SHA256 my-automation-package.zip
```

3. Definir la sección de archivos en su manual de procedimientos

   Agregue una sección `files` en el nivel superior de su manual de procedimientos para hacer referencia a su archivo adjunto:

```
files:
  my-automation-package.zip:
    sourceType: "S3"
    sourceInfo:
      path: "s3://my-automation-bucket/scripts/my-automation-package.zip"
      checksums:
        sha256: "your-calculated-checksum-here"
```

4. Hacer referencia al archivo adjunto en su paso de executeScript

   Utilice el parámetro `Attachment` para hacer referencia al archivo cargado:

```
- name: runMyScript
  action: aws:executeScript
  inputs:
    Runtime: python3.11
    Handler: main.process_data
    Attachment: my-automation-package.zip
    InputPayload:
      inputData: "{{InputParameter}}"
```

Ejemplos de archivos adjuntos aws:executeScript

Los siguientes ejemplos muestran diferentes maneras de usar los archivos adjuntos con la acción `aws:executeScript`.

Ejemplo 1: Archivo adjunto único

En este ejemplo, se muestra cómo utilizar un único archivo Python como adjunto para procesar los datos de la instancia EC2.

Archivo adjunto: process_instance.py

Cree un archivo Python con el siguiente contenido:

```
import boto3
import json

def process_instance_data(events, context):
    """Process EC2 instance data and return formatted results."""
    try:
        instance_id = events.get('instanceId')
        if not instance_id:
            raise ValueError("instanceId is required")

        ec2 = boto3.client('ec2')

        # Get instance details
        response = ec2.describe_instances(InstanceIds=[instance_id])
        instance = response['Reservations'][0]['Instances'][0]

        # Format the response
        result = {
            'instanceId': instance_id,
            'instanceType': instance['InstanceType'],
            'state': instance['State']['Name'],
            'availabilityZone': instance['Placement']['AvailabilityZone'],
            'tags': {tag['Key']: tag['Value'] for tag in instance.get('Tags', [])}
        }

        print(f"Successfully processed instance {instance_id}")
        return result

    except Exception as e:
        print(f"Error processing instance: {str(e)}")
        raise
```

Manual de procedimientos completo

Este es el manual de procedimientos completo que usa el archivo adjunto único:

```
description: Process EC2 instance data using single file attachment
schemaVersion: '0.3'
assumeRole: '{{AutomationAssumeRole}}'
```

```
parameters:
  AutomationAssumeRole:
    type: String
    description: (Required) IAM role for automation execution
  InstanceId:
    type: String
    description: (Required) EC2 instance ID to process

files:
  process_instance.py:
    sourceType: "S3"
    sourceInfo:
      path: "s3://my-automation-bucket/scripts/process_instance.py"
      checksums:
        sha256: "abc123def456..."

mainSteps:
  - name: processInstance
    action: aws:executeScript
    inputs:
      Runtime: python3.11
      Handler: process_instance.process_instance_data
      Attachment: process_instance.py
      InputPayload:
        instanceId: '{{InstanceId}}'
    outputs:
      - Type: StringMap
        Name: InstanceData
        Selector: $.Payload

outputs:
  - processInstance.InstanceData
```

Ejemplo 2: Paquete de varios módulos

En este ejemplo, se muestra el uso de un paquete zip que contiene varios módulos de Python para operaciones complejas de bucket de S3.

Estructura del paquete

Cree un paquete zip con la siguiente estructura:

```
s3-operations.zip
### main.py
```

```
### utils/
#   ### __init__.py
#   ### s3_helper.py
#   ### validation.py
### config/
    ### __init__.py
    ### settings.py
```

main.py (punto de entrada)

El script principal que orquesta las operaciones:

```
from utils.s3_helper import S3Operations
from utils.validation import validate_bucket_name
from config.settings import get_default_settings

def cleanup_s3_bucket(events, context):
    """Clean up S3 bucket based on specified criteria."""
    try:
        bucket_name = events.get('bucketName')
        max_age_days = events.get('maxAgeDays', 30)

        # Validate inputs
        if not validate_bucket_name(bucket_name):
            raise ValueError(f"Invalid bucket name: {bucket_name}")

        # Initialize S3 operations
        s3_ops = S3Operations()
        settings = get_default_settings()

        # Perform cleanup
        deleted_objects = s3_ops.delete_old_objects(
            bucket_name,
            max_age_days,
            settings['dry_run']
        )

        result = {
            'bucketName': bucket_name,
            'deletedCount': len(deleted_objects),
            'deletedObjects': deleted_objects[:10],  # Return first 10 for brevity
            'dryRun': settings['dry_run']
        }
```

```
        print(f"Cleanup completed for bucket {bucket_name}")
        return result

    except Exception as e:
        print(f"Error during S3 cleanup: {str(e)}")
        raise
```

Solución de problemas de aws:executeScript

Utilice la siguiente guía para resolver los problemas habituales relacionados con los archivos adjuntos `aws:executeScript`:

Errores de importación del módulo

Si recibe errores de importación al utilizar paquetes de varios módulos haga lo siguiente:

- Asegúrese de haber incluido un archivo vacío `__init__.py` en cada directorio que contenga los módulos de Python.
- Compruebe que las instrucciones de importación coincidan con la estructura real de archivos y directorios del paquete zip.
- Utilice las importaciones relativas (por ejemplo, `from .utils import helper`) o las importaciones absolutas (por ejemplo, `from utils import helper`) de forma coherente.

Errores de archivo adjunto no encontrado

Si la automatización no encuentra el archivo adjunto, haga lo siguiente:

- Compruebe que el valor del parámetro `Attachment` coincida exactamente con la clave de su sección `files`.
- Compruebe que la ruta del bucket de S3 y el nombre del archivo sean correctos en la sección `files`.
- Asegúrese de que su función de automatización tenga el permiso `s3:GetObject` para la ubicación de S3 del archivo adjunto.
- Compruebe que la suma de verificación en su manual de procedimientos coincida con la suma de verificación real del archivo.

Errores de función del controlador

Si recibe errores relacionados con el controlador, haga lo siguiente:

- Para Python: utilice el formato `filename.function_name` en el parámetro `Handler` (por ejemplo, `main.process_data`).
- Asegúrese de que su función controladora acepte exactamente dos parámetros: `events` y `context`.
- Para PowerShell: no especifique ningún parámetro `Handler`; el script se ejecuta directamente.

Errores de ejecución de scripts

Si el script falla durante la ejecución, haga lo siguiente:

- Consulte el historial de ejecución de la automatización para ver los mensajes de error detallados y los seguimientos de la pila.
- Use instrucciones `print()` (Python) o `Write-Information` (PowerShell) para agregar resultados de depuración.
- Compruebe que se concedan todos los permisos de AWS necesarios a su función de automatización.
- Pruebe la lógica del script localmente antes de empaquetarlo como un archivo adjunto.

Gestión de errores y códigos de salida

Para gestionar correctamente los errores y devolver los códigos de salida, haga lo siguiente:

- En Python: use `raise Exception("error message")` para indicar un error en el script.
- En PowerShell: utilice `throw "error message"` o `Write-Error` para indicar un error.
- Devuelva datos estructurados de sus funciones para proporcionar información detallada sobre el éxito o el error.
- Utilice bloques try-catch para gestionar las excepciones de manera elegante y proporcionar mensajes de error significativos.

`aws:executeStateMachine`: ejecutar una máquina de estado de AWS Step Functions

Ejecuta una máquina de estado de AWS Step Functions.

Entrada

Esta acción admite la mayoría de los parámetros de la operación de la API de Step Functions StartExecution.

Permisos de AWS Identity and Access Management (IAM) necesarios

- `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

El nombre de recurso de Amazon (ARN) de la máquina de estado de Step Functions.

Tipo: cadena

Obligatorio: sí

name

El nombre de la ejecución.

Tipo: cadena

Requerido: no

input

Una cadena que contiene los datos de entrada JSON de la ejecución.

Tipo: cadena

Requerido: no

Salidas

Las siguientes salidas están predefinidas para esta acción.

executionArn

El ARN de la ejecución.

Tipo: cadena

input

La cadena que contiene los datos de entrada JSON de la ejecución. Las restricciones de longitud se aplican al tamaño de la carga y se expresan como bytes en codificación UTF-8.

Tipo: cadena

name

El nombre de la ejecución.

Tipo: cadena

salida

Los datos de salida JSON de la ejecución. Las restricciones de longitud se aplican al tamaño de la carga y se expresan como bytes en codificación UTF-8.

Tipo: cadena

startDate

La fecha en que se inicia la ejecución.

Tipo: cadena

stateMachineArn

El ARN de la máquina indicada ejecutada.

Tipo: cadena

estado

El estado actual de la ejecución.

Tipo: cadena

stopDate

Si la ejecución ya ha finalizado, la fecha en que se detuvo la ejecución.

Tipo: cadena

`aws:invokeWebhook`: invocar una integración de webhook de Automation

Invoca la integración de webhook de Automation especificada. Para obtener información acerca de la creación de integraciones de Automation, consulte Crear integraciones webhook para Automation.

> **Note**
>
> Para utilizar la acción `aws:invokeWebhook`, su usuario o rol de servicio debe permitir las siguientes acciones:
>
> - ssm:GetParameter
> - kms:Decrypt
>
> El permiso para la operación `Decrypt` de AWS Key Management Service (AWS KMS) solo es necesario si utiliza una clave administrada por el cliente con el fin de cifrar el parámetro de la integración.

Entrada

Proporcione la información de la integración de Automation que desea que invoque.

YAML

```
action: "aws:invokeWebhook"
inputs:
 IntegrationName: "exampleIntegration"
```

```
 Body: "Request body"
```

JSON

```
{
    "action": "aws:invokeWebhook",
    "inputs": {
        "IntegrationName": "exampleIntegration",
        "Body": "Request body"
    }
}
```

IntegrationName

El nombre de la integración de Automation. Por ejemplo, `exampleIntegration`. La integración que especifique debe existir previamente.

Tipo: cadena

Obligatorio: sí

Cuerpo

La carga que desea enviar cuando se invoca la integración de webhook.

Tipo: cadena

Requerido: no

Salida

Respuesta

El texto recibido de la respuesta del proveedor de webhook.

ResponseCode

El código del estado HTTP recibido de la respuesta del proveedor de webhook.

`aws:invokeLambdaFunction`: invocar una función de AWS Lambda

Invoca la función de AWS Lambda especificada.

> **Note**
>
> Cada acción `aws:invokeLambdaFunction` puede ejecutarse hasta un máximo de 300 segundos (5 minutos). Puede limitar el tiempo de espera mediante la especificación del parámetro `timeoutSeconds` para un paso `aws:invokeLambdaFunction`.

Input

Esta acción admite la mayoría de los parámetros invocados del servicio Lambda. Para obtener más información, consulte Invoke.

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

El nombre de la función de Lambda. Esta función debe existir.

Tipo: cadena

Obligatorio: sí

Qualifier

La versión de función o nombre de alias.

Tipo: cadena

Requerido: no

InvocationType

El tipo de invocación. El valor predeterminado es `RequestResponse`.

Tipo: cadena

Valores válidos: `Event` | `RequestResponse` | `DryRun`

Requerido: no

LogType

Si el valor predeterminado es `Tail`, el tipo de invocación debe ser `RequestResponse`. Lambda devuelve los últimos 4 KB de datos de registro generados por la función de Lambda, con codificación Base64.

Tipo: cadena

Valores válidos: `None` | `Tail`

Requerido: no

ClientContext

La información específica del cliente.

Requerido: no

InputPayload

Un objeto JSON o YAML que se pasará al primer parámetro del controlador. Puede utilizar esta entrada para transferir datos a la función. Esta entrada proporciona más flexibilidad y compatibilidad que la entrada de `Payload` heredada. Si define `InputPayload` y `Payload` para la acción, `InputPayload` tiene prioridad y el valor de `Payload` no se utiliza.

Tipo: StringMap

Requerido: no

Carga

Una cadena JSON que se pasará al primer parámetro del controlador. Puede utilizar estos datos de entrada para pasarlos a la función. Recomendamos utilizar la entrada de `InputPayload` para garantizar una mayor funcionalidad.

Tipo: cadena

Requerido: no

Output

StatusCode

El código de estado HTTP.

FunctionError

Si está presente, indica que se ha producido un error durante la ejecución de la función. Se incluyen detalles sobre el error en la carga de la respuesta.

LogResult

Los registros con codificación Base64 para la invocación de la función de Lambda. Los registros están presentes solo si el tipo de invocación es `RequestResponse` y se han solicitado.

Carga

La representación JSON del objeto devuelto por la función de Lambda. Payload está presente solo si el tipo de invocación es `RequestResponse`.

Lo que aparece a continuación es una parte del manual de procedimientos `AWS-PatchInstanceWithRollback` que muestra cómo hacer referencia a las salidas de la acción `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` — Repita los pasos en una automatización

Esta acción se repite en un subconjunto de pasos de un manual de procedimientos de automatización. Puede elegir un bucle de estilo `do while` o `for each`. Para construir un bucle `do while`, utilice el parámetro de entrada `LoopCondition`. Para construir un bucle `for each`, utilice los parámetros de entrada `Iterators` y `IteratorDataType`. Cuando utilice una acción `aws:loop`, especifique únicamente el parámetro de entrada `Iterators` o `LoopCondition`. El número máximo de veces que se van a ejecutar es de 100.

La propiedad `onCancel` solo se puede utilizar para los pasos definidos dentro de un bucle. La propiedad `onCancel` no es compatible con la acción `aws:loop`. La propiedad `onFailure` se puede usar para una acción `aws:loop`, pero solo se usará si se produce un error inesperado que

provoque un error en el paso. Si define propiedades `onFailure` para los pasos de un bucle, la acción `aws:loop` hereda esas propiedades y reacciona en consecuencia cuando se produce un error.

Ejemplos

A continuación se muestran ejemplos de cómo construir los distintos tipos de acciones de bucle.

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
```

Entrada

La entrada es la siguiente.

Iteradores

La lista de elementos sobre los que se deben iterar los pasos. El número máximo de iteradores es 100.

Tipo: StringList

Requerido: no

IteratorDataType (Tipo de datos de iteradores)

Un parámetro opcional para especificar el tipo de datos del `Iterators`. Se puede proporcionar un valor para este parámetro junto con el parámetro de entrada `Iterators`. Si no especifica un valor para este parámetro y `Iterators`, luego debe especificar un valor para el parámetro `LoopCondition`.

Tipo: cadena

Valores válidos: Boolean | Integer | String | StringMap

Predeterminado: String

Requerido: no

LoopCondition

Consta de una `Variable` y una condición de operador a evaluar. Si no especifica un valor para este parámetro, debe especificar un valor para los parámetros `Iterators` y `IteratorDataType`. Puede utilizar evaluaciones de operadores complejas mediante una combinación de operadores `And`, `Not` y `Or`. La condición se evalúa una vez completados los pasos del ciclo. Si la condición es `true` y el valor `MaxIterations` no se ha alcanzado, los pasos del bucle se vuelven a ejecutar. Las condiciones del operador son las siguientes:

Operaciones de cadena

- StringEquals

- EqualsIgnoreCase
- StartsWith
- EndsWith
- Contiene

Operaciones numéricas

- NumericEquals
- NumericGreater
- NumericLesser
- NumericGreaterOrEquals
- NumericLesser
- NumericLesserOrEquals

Operación booleana

- BooleanEquals

Tipo: StringMap

Requerido: no

MaxIterations (Iteraciones máximas)

Número máximo de veces que se van a ejecutar los pasos del ciclo. Una vez que se alcanza el valor especificado para esta entrada, el bucle deja de ejecutarse incluso si `LoopCondition` aún es `true` o si quedan objetos en el parámetro `Iterators`.

Tipo: entero

Valores válidos: 1 - 100

Requerido: no

Pasos

La lista de pasos que se van a ejecutar en el bucle. Funcionan como un manual de procedimientos anidado. En estos pasos, puede acceder al valor del iterador actual de un bucle `for each` mediante la sintaxis `{{loopStepName.CurrentIteratorValue}}`. También puede acceder a un valor entero de la iteración actual para ambos tipos de bucles mediante la sintaxis `{{loopStepName.CurrentIteration}}`.

Tipo: lista de pasos

Obligatorio: sí

Salida

Currentiteration (Iteración actual)

La iteración del bucle actual como número entero. Los valores de iteración comienzan en 1.

Tipo: entero

CurrentiteratorValue (Valor del iterador actual)

El valor del iterador actual como una cadena. Esta salida solo está presente en los bucles `for each`.

Tipo: cadena

`aws:pause`: detener una automatización

Esta acción detiene la automatización. Una vez detenida, el estado de la automatización es Waiting (En espera). Para continuar con la automatización, utilice la operación de la API SendAutomationSignal con el tipo de señal `Resume`. Se recomienda utilizar las acciones `aws:sleep` o `aws:approve` para tener un control más detallado de los flujos de trabajo.

> **Note**
>
> El tiempo de espera predeterminado para esta acción es de 7 días (604 800 segundos) y el valor máximo es de 30 días (2 592 000 segundos). Puede limitar o ampliar el tiempo de espera mediante la especificación del parámetro `timeoutSeconds` para un paso `aws:pause`.

Input

La entrada es la siguiente.

YAML

```
name: pauseThis
```

```
action: aws:pause
timeoutSeconds: 1209600
inputs: {}
```

JSON

```
{
    "name": "pauseThis",
    "action": "aws:pause",
    "timeoutSeconds": "1209600",
    "inputs": {}
}
```

Output

Ninguno

`aws:runCommand`: ejecutar un comando en una instancia administrada

Ejecuta los comandos especificados.

> **Note**
>
> Automation solo admite la salida de una acción de AWS Systems Manager Run Command. Un manual de procedimientos puede incluir varias acciones de Run Command, pero la salida solo se admite para una acción a la vez.

Input

Esta acción admite la mayoría de los parámetros de comando. Para obtener más información, consulte 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 el documento de tipo Command le pertenece a usted o a AWS, especifique el nombre del documento. Si está utilizando un documento que una Cuenta de AWS diferente compartió con usted, especifique el nombre de recurso de Amazon (ARN) del documento. Para obtener más información acerca del uso de documentos compartidos, consulte Uso de documentos de SSM compartidos.

Tipo: cadena

Obligatorio: sí

InstanceIds

Los ID de instancia donde desea que se ejecute el comando. Puede especificar un máximo de 50 ID.

También puede utilizar el pseudoparámetro `{{RESOURCE_ID}}` en lugar de los ID de instancias para ejecutar el comando en todas las instancias del grupo de destino. Para obtener más

información sobre pseudoparámetros, consulte Utilización de pseudoparámetros en el registro de las tareas del periodo de mantenimiento.

Otra alternativa es enviar comandos a una flota de instancias con el parámetro `Targets`. El parámetro `Targets` acepta etiquetas de Amazon Elastic Compute Cloud (Amazon EC2). Para obtener más información acerca de cómo utilizar el parámetro `Targets`, consulte Ejecución de comandos a escala.

Tipo: StringList

Obligatorio: no (Si no especifica InstanceIds ni utiliza el pseudoparámetro `{{RESOURCE_ID}}`, debe especificar el parámetro `Targets`).

Destinos

Una matriz de criterios de búsqueda que indica instancias como destino mediante el uso de una combinación de clave-valor que usted especifique. `Targets` es obligatorio si no se proporciona uno o más ID de instancia en la llamada. Para obtener más información acerca de cómo utilizar el parámetro `Targets`, consulte Ejecución de comandos a escala.

Tipo: MapList (El esquema del mapa en la lista debe coincidir con el objeto). Para obtener información, consulte Target en la Referencia de la API de AWS Systems Manager.

Obligatorio: no (Si no especifica `Targets`, debe especificar InstanceIds o utilizar el pseudoparámetro `{{RESOURCE_ID}}`).

A continuación se muestra un ejemplo.

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"
            ]
        }
    }
}
```

Parámetros

Los parámetros obligatorios y opcionales especificados en el documento.

Tipo: mapa

Obligatorio: no

CloudWatchOutputConfig

Las opciones de configuración para enviar la salida del comando a Amazon CloudWatch Logs. Para obtener más información acerca de cómo enviar la salida de un comando a CloudWatch Logs, consulte Configuración de Registros de Amazon CloudWatch para Run Command.

Tipo: StringMap (El esquema del mapa debe coincidir con el objeto. Para obtener más información, consulte CloudWatchOutputConfig en la Referencia de la API de AWS Systems Manager).

Obligatorio: no

A continuación se muestra un ejemplo.

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
        }
```

```
    }
}
```

Comentario

Información definida por el usuario sobre el comando.

Tipo: cadena

Requerido: no

DocumentHash

El hash del documento.

Tipo: cadena

Requerido: no

DocumentHashType

El tipo del hash.

Tipo: cadena

Valores válidos: `Sha256` | `Sha1`

Obligatorio: no

NotificationConfig

Las configuraciones para enviar notificaciones.

Obligatorio: no

OutputS3BucketName

El nombre del bucket de S3 para las respuestas de salida del comando. Su nodo administrado debe tener permisos para el bucket de S3 con el objetivo de registrar correctamente la salida.

Tipo: cadena

Requerido: no

OutputS3KeyPrefix

El prefijo .

Tipo: cadena

Requerido: no

ServiceRoleArn

El ARN del rol de AWS Identity and Access Management (IAM).

Tipo: cadena

Requerido: no

TimeoutSeconds

La cantidad de segundos que hay que esperar para que un comando entregue el resultado a AWS Systems Manager SSM Agent en una instancia. Si SSM Agent en la instancia no recibe el comando antes de que se alcance el valor especificado, el estado del comando se transforma en `Delivery Timed Out`.

Tipo: entero

Obligatorio: no

Valores válidos: 30-2592000

Output

CommandId

El ID del comando.

Estado

El estado del comando.

ResponseCode

El código de respuesta del comando. Si el documento que ejecuta tiene más de 1 paso, no se devuelve ningún valor para esta salida.

Output

La salida del comando. Si dirige el comando a una etiqueta o a varias instancias, no se devuelve ningún valor de salida. Puede usar las operaciones de la API `GetCommandInvocation` y `ListCommandInvocations` para recuperar resultados para instancias individuales.

`aws:runInstances`: lanzar una instancia de Amazon EC2

Lanza una instancia nueva de Amazon Elastic Compute Cloud (Amazon EC2).

Entrada

La acción admite la mayoría de los parámetros de la API. Para obtener más información, consulte la documentación de la 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

Reservado.

Tipo: cadena

Requerido: no

BlockDeviceMappings

Los dispositivos de bloques para la instancia.

Tipo: MapList

Requerido: no

ClientToken

El identificador para garantizar la instancia idempotente de la solicitud.

Tipo: cadena

Requerido: no

DisableApiTermination

Activa o desactiva la terminación de la API de la instancia.

Tipo: Booleano

Requerido: no

EbsOptimized

Activa o desactiva la optimización de Amazon Elastic Block Store (Amazon EBS).

Tipo: Booleano

Requerido: no

IamInstanceProfileArn

El nombre de recurso de Amazon (ARN) del perfil de instancia de AWS Identity and Access Management (IAM).

Tipo: cadena

Requerido: no

IamInstanceProfileName

El nombre del perfil de instancia de IAM correspondiente a la instancia.

Tipo: cadena

Requerido: no

ImageId

El ID de la Amazon Machine Image (AMI).

Tipo: cadena

Obligatorio: sí

InstanceInitiatedShutdownBehavior

Indica si la instancia se detiene o termina al cerrarse el sistema.

Tipo: cadena

Requerido: no

InstanceType

El tipo de instancia.

> **Note**
>
> Si no se proporciona el valor de tipo de instancia, se emplea el tipo de instancia m1.small.

Tipo: cadena

Requerido: no

KernelId

El ID del kernel.

Tipo: cadena

Requerido: no

KeyName

El nombre del par de claves.

Tipo: cadena

Requerido: no

MaxInstanceCount

El número máximo de instancias que se van a lanzar.

Tipo: cadena

Requerido: no

MetadataOptions

Opciones de metadatos de la instancia. Para obtener más información, consulte InstanceMetadataOptionsRequest.

Tipo: StringMap

Requerido: no

MinInstanceCount

El número mínimo de instancias que se van a lanzar.

Tipo: cadena

Requerido: no

Supervisión

Activa o desactiva el monitoreo detallado.

Tipo: Booleano

Requerido: no

NetworkInterfaces

Las interfaces de red.

Tipo: MapList

Requerido: no

Placement

La ubicación de la instancia.

Tipo: StringMap

Requerido: no

PrivateIpAddress

La dirección IPv4 principal.

Tipo: cadena

Requerido: no

RamdiskId

El ID del disco RAM.

Tipo: cadena

Requerido: no

SecurityGroupIds

Los ID de los grupos de seguridad para la instancia.

Tipo: StringList

Requerido: no

SecurityGroups

Los nombres de los grupos de seguridad para la instancia.

Tipo: StringList

Requerido: no

SubnetId

El ID de subred.

Tipo: cadena

Requerido: no

TagSpecifications

Las etiquetas que aplicar a los recursos durante el lanzamiento. Solo puede etiquetar instancias y volúmenes en el momento del lanzamiento. Las etiquetas especificadas se aplican a todas las instancias o volúmenes que se crean durante el lanzamiento. Para etiquetar una instancia después de que se haya lanzado, utilice la acción `aws:createTags`: crear etiquetas para recursos de AWS.

Tipo MapList (Para obtener más información, consulte TagSpecification).

Requerido: no

UserData

Un script proporcionado como un valor literal de cadena. Si se escribe un valor literal, debe estar codificado en Base64.

Tipo: cadena

Requerido: no

Salida

InstanceIds

Los ID de las instancias.

InstanceStates

El estado actual de la instancia.

`aws:sleep`: retrasar una automatización

Retrasa una automatización durante un periodo determinado. Esta acción utiliza el formato de fecha y hora de la Organización Internacional de Normalización (ISO, por sus siglas en inglés) 8601. Para obtener más información sobre este formato de fecha y hora, consulte ISO 8601.

Entrada

Puede retrasar una automatización durante un periodo determinado.

YAML

```
name: sleep
action: aws:sleep
inputs:
  Duration: PT10M
```

JSON

```
{
   "name":"sleep",
   "action":"aws:sleep",
   "inputs":{
      "Duration":"PT10M"
   }
}
```

También puede retrasar la automatización hasta una fecha y una hora específicas. Si la fecha y hora especificadas ha transcurrido, la acción avanza de forma inmediata.

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**
>
> La automatización admite un tiempo de espera máximo de 604 799 segundos (7 días).

Duración

Una duración ISO 8601. No puede especificar una duración negativa.

Tipo: cadena

Requerido: no

Timestamp

Una marca de tiempo ISO 8601. Si no especifica un valor para este parámetro, debe especificar un valor para el parámetro `Duration`.

Tipo: cadena

Requerido: no

Salida

Ninguna

`aws:updateVariable` — Actualiza el valor de una variable del manual de procedimientos

Esta acción actualiza el valor de una variable del manual de procedimientos. El tipo de datos del valor debe coincidir con el tipo de datos de la variable que desea actualizar. Las conversiones de tipos de datos no son compatibles. La propiedad `onCancel` no es compatible con la acción `aws:updateVariable`.

Entrada

La entrada es la siguiente.

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"]
    }
}
```

Nombre

El nombre de la variable cuyo valor desea actualizar. Debe usar el formato `variable:variable name`

Tipo: cadena

Obligatorio: sí

Valor

El nuevo valor que se va a asignar a la variable. El valor debe coincidir con el tipo de datos de la variable. Las conversiones de tipos de datos no son compatibles.

Tipo: Boolean | Integer | MapList | String | StringList | StringMap

Obligatorio: sí

Restricciones:

- MapList (Lista de mapas) puede contener un número máximo de 200 elementos.
- La clave puede tener una longitud mínima de 1 y una longitud máxima de 50.
- StringList puede tener un número mínimo de 0 elementos y un número máximo de 50 elementos.
- La cadena puede tener una longitud mínima de 1 y una longitud máxima de 512.

Salida

Ninguna

`aws:waitForAwsResourceProperty`: esperar una propiedad de recurso de AWS

La acción `aws:waitForAwsResourceProperty` permite a su automatización esperar hasta alcanzar un estado de recurso o de evento específico antes de continuar con la automatización. Para más ejemplos sobre cómo usar esta acción, consulte Ejemplos adicionales de manuales de procedimientos.

> **Note**
>
> El valor del tiempo de espera predeterminado para esta acción es de 3600 segundos (una hora). Puede limitar o ampliar el tiempo de espera mediante la especificación del parámetro `timeoutSeconds` para un paso `aws:waitForAwsResourceProperty`. Para obtener más información y ejemplos sobre cómo usar esta acción, consulte Administración de los tiempos de espera en los manuales de procedimientos.

Entrada

Las entradas se definen con la operación de la API que elija.

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"
    ]
  }
}
```

Servicio

El espacio de nombres del Servicio de AWS que contiene la operación de la API que desea ejecutar. Por ejemplo, el espacio de nombres para AWS Systems Manager es `ssm`. El espacio de nombres para Amazon Elastic Compute Cloud (Amazon EC2) es `ec2`. Puede ver una lista de espacios de nombres de Servicio de AWS admitidos en la sección Available Services (Servicios disponibles) de la Referencia de los comandos de la AWS CLI.

Tipo: cadena

Obligatorio: sí

API

El nombre de la operación de la API que desea ejecutar. Puede ver las operaciones de la API (también llamadas métodos) si elige un servicio en el panel de navegación ubicado a la izquierda, en la siguiente página: Services Reference. Elija un método en la sección Cliente para el servicio que desea invocar. Por ejemplo, todas las operaciones de la API (los métodos) para Amazon Relational Database Service (Amazon RDS) se indican en la siguiente página: métodos de Amazon RDS.

Tipo: cadena

Obligatorio: sí

Entradas de la operación de la API

Una o más entradas de la operación de la API. Puede ver las entradas disponibles (también llamadas parámetros) eligiendo un servicio en el panel de navegación izquierdo en la siguiente página de referencia de servicios. Elija un método en la sección Cliente para el servicio que desea invocar. Por ejemplo, todos los métodos para Amazon RDS se indican en la siguiente página: métodos de Amazon RDS. Elija el método describe_db_instances y desplácese hacia abajo para ver los parámetros disponibles, como, por ejemplo, DBInstanceIdentifier, Name y 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"
}
```

Tipo: se determina a partir de la operación de la API elegida

Obligatorio: sí

PropertySelector

El elemento JSONPath a un atributo específico en el objeto de respuesta. Puede ver los objetos de respuesta eligiendo un servicio en el panel de navegación izquierdo en la siguiente página de referencia de servicios. Elija un método en la sección Cliente para el servicio que desea invocar. Por ejemplo, todos los métodos para Amazon RDS se indican en la siguiente página: métodos de Amazon RDS. Elija el método describe_db_instances y desplácese hasta la sección Response Structure (Estructura de respuesta). DBInstances aparece como un objeto de respuesta.

Tipo: cadena

Obligatorio: sí

DesiredValues

El estado previsto o el estado en el que continuará la automatización.

Tipo: MapList, StringList

Obligatorio: sí

Variables del sistema de Automation

Los manuales de procedimientos de AWS Systems Manager Automation usan las siguientes variables. En el código fuente JSON del manual de procedimientos `AWS-UpdateWindowsAmi`, puede ver un ejemplo de cómo se usan estas variables.

Para ver el código fuente JSON del manual de procedimientos **`AWS-UpdateWindowsAmi`**

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. En la lista de documentos, utilice la barra de búsqueda o los números que se encuentran a la derecha de la barra de búsqueda para seleccionar el manual de procedimientos **`AWS-UpdateWindowsAmi`**.
4. Elija la pestaña Content.

Variables del sistema

Los manuales de procedimientos de Automation admiten las siguientes variables del sistema.

Variable	Detalles
`global:ACCOUNT_ID`	El ID de la Cuenta de AWS del usuario o el rol donde se ejecuta Automatización.
`global:DATE`	La fecha (en el tiempo de ejecución) con el formato aaaa-MM-dd.

Variable	Detalles
`global:DATE_TIME`	La fecha y la hora (en el tiempo de ejecución) con el formato aaaa-MM-dd_HH.mm.ss.
`global:AWS_PARTITION`	Partición en la que se encuentra el recurso. Para las Regiones de AWS estándar, la partición es `aws`. Para los recursos en otras particiones, la partición es `aws-`*`partition name`* . Por ejemplo, la partición de los recursos de la región AWS GovCloud (EE. UU. Oeste) es `aws-us-gov` .
`global:REGION`	La región en la que se ejecuta el manual de procedimientos. Por ejemplo, us-east-2.

Variables de Automation

Los manuales de procedimientos de Automation admiten las siguientes variables de la automatización.

Variable	Detalles
`automation:EXECUTION_ID`	El identificador único asignado a la automatización actual. Por ejemplo, `1a2b3c-1a2b3c-1a2b3c-1a2b3c1a2b3c1a2b3c` .

Temas

- Terminología
- Escenarios admitidos
- Escenarios no admitidos

Terminología

Los siguientes términos describen cómo se resuelven las variables y los parámetros.

Plazo	Definición	Ejemplo
ARN constante	Un nombre de recurso de Amazon (ARN) válido sin variables.	`arn:aws:iam::123456789012:role/roleName`
Parámetro del manual de procedimientos	Un parámetro definido en el nivel del manual de procedimientos (por ejemplo, `instanceId`). El parámetro se utiliza en un reemplazo de cadena básica. Su valor se proporciona en el tiempo de ejecución de inicio.	`{ "description": "Create Image Demo", "version": "0.3", "assumeRole": "Your_Automation_Assume_Role_ARN ", "parameters":{ "instanceId": { "type": "String", "description": "Instance to create image from" } }`
Variable del sistema	Una variable general que se sustituye en el manual de procedimientos cuando se evalúa cualquiera de sus partes.	`"activities": [ { "id": "copyImage", "activityType": "AWS-CopyImage", "maxAttempts": 1, "onFailure": "Continue", "inputs": { "ImageName": "{{imageName}}", "SourceImageId": "{{sourceImageId}}", "SourceRegion": "{{sourceRegion}}",`

Plazo	Definición	Ejemplo
		"Encrypted": true, "ImageDescription": "Test CopyImage Description created on **{{global:DATE}}** " } }]

Plazo	Definición	Ejemplo

Variable de Automation	Una variable relacionada con la automatización que se sustituye en el manual de procedimientos cuando se evalúa cualquier parte del manual.	{
		"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}}** ”
		]
		}
		}
		}

Plazo	Definición	Ejemplo

Parámetro de Systems Manager	Un variable definida en AWS Systems Manager Parameter Store. No se puede referenciar directamente en la entrada de paso. Es posible que se requieran permisos para acceder al parámetro.	`description: Launch new Windows test instance`
		`schemaVersion: '0.3'`
		`assumeRole: '{{AutomationAssumeRole}}'`
		`parameters:`
		`  AutomationAssumeRole:`
		`    type: String`
		`    default: ''`
		`    description: >-`
		`      (Required) The ARN of the role that allows Automation to perform the`
		`      actions on your behalf. If no role is specified, Systems Manager`
		`      Automation uses your IAM permissions to run this runbook.`
		`  LatestAmi:`
		`    type: String`
		`    default: >-`
		`      {{ssm:/aws/service/ami-windows-latest/Windows_Server-2016-English-Full-Base}}`
		`    description: The latest Windows Server 2016 AMI queried from the public parameter.`
		`mainSteps:`
		`  - name: launchInstance`
		`    action: 'aws:runInstances'`
		`    maxAttempts: 3`

Plazo	Definición	Ejemplo
		timeoutSeconds: 1200 onFailure: Abort inputs: ImageId: '{{LatestAmi}}' ...

Escenarios admitidos

Escenario	Comentarios	Ejemplo
ARN constante `assumeRole` en la creación.	Se lleva a cabo una comprobación de autenticación para verificar que el usuario que realiza la llamada tiene permiso para pasar el `assumeRole` indicado.	{ "description": "Test all Automation resolvable parameters", "schemaVersion": "0.3", "assumeRole": ***"arn:aws:iam::123456789012:role/roleName"*** , "parameters": { ...
Parámetro de manual de procedimientos suministrado para `AssumeRole` cuando se inicia la automatización.	Se debe definir en la lista de parámetros del manual de procedimientos.	{ "description": "Test all Automation resolvable parameters", "schemaVersion": "0.3", "assumeRole": **"{{dynamicARN}}"** , "parameters": { ...

Escenario	Comentarios	Ejemplo

Valor proporcionado para el parámetro del manual de procedimientos en el inicio.	El cliente proporciona el valor que se usará para un parámetro. Las entradas suministradas en el inicio deben estar definidas en la lista de parámetros del manual de procedimientos.	<pre>...
		"parameters": {
		    "amiId": {
		      "type": "String",
		      "default":
		 "*ami-12345678* ",
		      "description":
		 "list of commands to
		 run as part of first
		 step"
		    },
		...</pre> Las entradas en la ejecución de automatización de inicio incluyen: `{"amiId" : ["`*ami-12345678*` "] }`

Escenario	Comentarios	Ejemplo

Parámetro de Systems Manager que se referencia en el contenido del manual de procedimientos.	La variable existe dentro de la cuenta del cliente o es un parámetro de acceso público, y el `AssumeRole` para el manual de procedimientos tiene acceso a la variable. Se lleva a cabo una comprobación en el momento de la creación para confirmar que `AssumeRole` tiene acceso. No se puede referenciar de manera directa el parámetro en la entrada de paso.	<pre>...
		parameters:
		    LatestAmi:
		    type: String
		    default: >-
		      {{ssm:/aws/service/ami-windows-latest/Windows_Server-2016-English-Full-Base}}
		    description: The latest Windows Server 2016 AMI queried from the public parameter.
		mainSteps:
		  - name: launchInstance
		    action: 'aws:runInstances'
		    maxAttempts: 3
		    timeoutSeconds: 1200
		    onFailure: Abort
		    inputs:
		      ImageId: '{{LatestAmi}}'
		...</pre>

Escenario	Comentarios	Ejemplo
Variable del sistema a la que se hace referencia en la definición del paso	Una variable de sistema se sustituye en el manual de procedimientos cuando se inicia la automatización. El valor inyectado en el manual de procedimientos es relativo al momento en el que se produce la sustitución. Por ejemplo, el valor de una variable de tiempo inyectada en el paso 1 es diferente del valor inyectado en el paso 3 debido al tiempo que se tarda en ejecutar los pasos intermedios. No es necesario que las variables del sistema se establezcan en la lista de parámetros del manual de procedimientos.	... "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}}}"] } } }, ...

Escenario	Comentarios	Ejemplo
Variable de Automation a la que se hace referencia en la definición del paso.	No es necesario que las variables de Automation se establezcan en la lista de parámetros del manual de procedimientos. La única variable de Automation admitida es automation:EXECUTION_ID.	... "mainSteps": [{ "name": "invokeLambdaFunction", "action": "aws:invokeLambdaFunction", "maxAttempts": 1, "onFailure": "Continue", "inputs": { "FunctionName": "Hello-World-LambdaFunction", "Payload" : "{ "executionId" : "{{automation:EXECUTION_ID}}" }" } } ...

Escenario	Comentarios	Ejemplo

Hacer referencia a la salida del paso anterior en la definición del paso siguiente.	Es el redireccionamiento de parámetros. Se hace referencia a la salida de un paso anterior con la sintaxis `{{stepName.OutputName}}` . El cliente no puede usar esta sintaxis para los parámetros del manual de procedimientos. Esto se resuelve cuando se ejecuta el paso de la referencia. El parámetro no se incluye en la lista de parámetros del manual de procedimientos.	`...`
		`"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"`
		`}`
		`}`
		
		`...`

Escenarios no admitidos

Escenario	Comentario	Ejemplo

Parámetro de Systems Manager proporcionado para `assumeRole` en la creación	No admitido.	`...`
		
		`{`
		`  "description":`
		` "Test all Automation`
		` resolvable parameter`
		`s",`
		`  "schemaVersion":`
		` "0.3",`
		`  "assumeRole":`
		` "{{ssm:administrato`
		`rRoleARN}}  ",`
		`  "parameters": {`
		
		`...`

Parámetro de Systems Manager que se referencia de manera directa en la entrada de paso	Devuelve la excepción `InvalidDocumentCon tent` en el momento de la creación.	`...`
		`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}}'`
		`...`

Definición de paso variable	La definición de un paso en el manual de procedimientos	`...`

Escenario	Comentario	Ejemplo

	se construye a través de variables.	"mainSteps": [
		{
		"name": "LaunchInstance",
		"action": "aws:runInstances",
		"**{{attemptModel}}** ": 1,
		"onFailure": "Continue",
		"inputs": {
		"ImageId": "*ami-12345678* ",
		"MinInstanceCount": 1,
		"MaxInstanceCount": 2
		}
		
		...
		
		User supplies input :
		{ "attemptModel" :
		"**minAttempts** " }

Escenario	Comentario	Ejemplo
Referencia cruzada de los parámetros del manual de procedimientos	El usuario proporciona un parámetro de entrada en el inicio, que es una referencia a otro parámetro del manual de procedimientos.	... "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} }" }, ...

Escenario	Comentario	Ejemplo
Expansión multinivel	El manual de procedimientos define una variable que toma el nombre de una variable. Se encuentra dentro de los delimitadores de variable (es decir, {{ }}) y se expande al valor de dicho parámetro/ variable.	... "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}}*** }}"] }

Escenario	Comentario	Ejemplo
		... Note: The customer intention here would be to run a command of "echo {Hello world}"

Escenario	Comentario	Ejemplo
Referencia a la salida de un paso de manual de procedimientos que es un tipo de variable diferente	El usuario referencia la salida de un paso de manual de procedimientos anterior en el paso siguiente. La salida es un tipo de variable que no cumple los requisitos de la acción en el paso siguiente.	... mainSteps: - name: getImageId action: aws:executeAwsApi inputs: Service: ec2 Api: DescribeImages Filters: - Name: "name" Values: - "{{ImageName}}" outputs: - Name: ImageIdList Selector: "$.Images" Type: "StringList" - name: copyMyImages action: aws:copyImage maxAttempts: 3 onFailure: Abort inputs: SourceImageId: {{getImageId.ImageIdList}} SourceRegion: ap-northeast-2 ImageName: Encrypted Copies of LAMP base AMI in ap-northeast-2 Encrypted: true ... Note: You must provide the type required by the Automation action. In this case, aws:copyImage requires a "String" type variable but the preceding step

Escenario	Comentario	Ejemplo
		`outputs a "StringList" type variable.`

Creación de sus propios manuales de procedimientos

Un manual de procedimientos de automatización define las acciones que Systems Manager realiza en las instancias administradas y en otros recursos de AWS cuando se ejecuta una automatización. Automatización es una herramienta de AWS Systems Manager. Un manual de procedimientos contiene uno o más pasos que se ejecutan en orden secuencial. Cada paso se construye en torno a una sola acción. La salida de un paso se puede utilizar como entrada en un paso posterior.

El proceso de ejecución de estas acciones y sus pasos se denomina automatización.

Los tipos de acción admitidos en los manuales de procedimientos le permiten automatizar una amplia variedad de operaciones en su entorno de AWS. Por ejemplo, con el tipo de acción `executeScript`, puede insertar un script de Python o PowerShell directamente en el manual de procedimientos. (Al crear un manual de procedimientos personalizado, puede agregar el script insertado o adjuntarlo desde un bucket de S3 o desde el equipo local). Puede automatizar la administración de sus recursos de AWS CloudFormation mediante el uso de los tipos de acción `deleteStack` y `createStack`. Además, mediante el tipo de acción `executeAwsApi`, un paso puede ejecutar cualquier operación de la API en cualquier Servicio de AWS, incluida la creación o la eliminación de recursos de AWS, el inicio de otros procesos, la creación de notificaciones y muchas más operaciones.

Para obtener una lista de los 20 tipos de acción admitidos para Automation, consulte Referencia de acciones de Automatización de Systems Manager.

Automation AWS Systems Manager proporciona varios manuales de procedimientos con pasos predefinidos que se pueden utilizar para realizar tareas comunes, como reiniciar una o más instancias de Amazon Elastic Compute Cloud (Amazon EC2), o crear una Amazon Machine Image (AMI). También puede crear sus propios manuales de procedimientos y compartirlos con otras Cuentas de AWS, o volverlos públicos para todos los usuarios de Automation.

Los manuales de procedimientos se escriben con YAML o JSON. Sin embargo, con el Generador de documentos en la consola de Automatización de Systems Manager, puede crear un manual de procedimientos sin tener que utilizar YAML o JSON nativo.

> **Important**
>
> Si ejecuta un flujo de trabajo de automatización que invoca otros servicios mediante un rol de servicio de AWS Identity and Access Management (IAM), tenga en cuenta que el rol de servicio debe configurarse con el permiso necesario para invocar dichos servicios. Este requisito se aplica a todos los manuales de procedimientos de automatización de AWS (manuales de `AWS-*`), como los manuales de procedimientos `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup` y `AWS-RestartEC2Instance`, por nombrar algunos. Este requisito también se aplica a cualquier manual de procedimientos de automatización personalizado que cree para llamar otros Servicios de AWS mediante acciones que llaman a otros servicios. Por ejemplo, si utiliza las acciones `aws:executeAwsApi`, `aws:createStack` o `aws:copyImage`, configure el rol de servicio con el permiso necesario para invocar dichos servicios. Puede conceder permisos a otros Servicios de AWS mediante la incorporación de una política insertada de IAM al rol. Para obtener más información, consulte (Opcional) Agregar una política insertada de Automatización o una política administrada por el cliente para invocar otros Servicios de AWS.

Para obtener información acerca de las acciones que puede especificar en un manual de procedimientos, consulte Referencia de acciones de Automatización de Systems Manager.

Para obtener información acerca del uso de AWS Toolkit for Visual Studio Code para crear manuales de procedimientos, consulte Uso de documentos de Automatización de Systems Manager en la Guía del usuario de AWS Toolkit for Visual Studio Code.

Para obtener información acerca del uso del diseñador visual para crear un manual de procedimientos personalizado, consulte Experiencia de diseño visual para manuales de procedimientos de automatización.

Contenido

- Experiencia de diseño visual para manuales de procedimientos de automatización
  - Antes de empezar
  - Descripción general de la interfaz de experiencia de diseño visual

        - Navegador de acciones
        - Canvas
        - Formulario
        - Métodos abreviados de teclado
    - Uso de la experiencia de diseño visual
        - Cree de un flujo de trabajo del manual de procedimientos
        - Diseñe un manual de procedimientos
        - Actualice su manual de procedimientos
        - Exporte su manual de procedimientos
    - Configuración de entradas y salidas para sus acciones
        - Proporcione datos de entrada para una acción
        - Defina los datos de salida de una acción
    - Manejo de errores con la experiencia de diseño visual
        - Vuelva a intentar la acción en caso de error
        - Tiempos de espera
        - Acciones fallidas
        - Acciones canceladas
        - Acciones cruciales
        - Finalización de acciones
    - Tutorial: cómo crear un manual de procedimientos utilizando la experiencia de diseño visual
        - Paso 1: navegue hasta la experiencia de diseño visual
        - Paso 2: Cree un flujo de trabajo
        - Paso 3: Revise el código generado automáticamente
        - Paso 4: Ejecute su nuevo manual de procedimientos
        - Paso 5: Eliminar
- Creación de manuales de procedimientos de Automation
    - Identifique su caso de uso
    - Configure el entorno de desarrollo.
    - Desarrolle contenido para el manual de procedimientos
    - Ejemplo 1: creación de manuales de procedimientos principal y secundario

        - Cree el manual de procedimientos secundario
        - Crear el manual de procedimientos principal
    - Ejemplo 2: manual de procedimientos con scripts
    - Ejemplos adicionales de manuales de procedimientos
        - Implementación de la arquitectura de VPC y de controladores de dominio de Microsoft Active Directory
        - Restauración de un volumen raíz a partir de la última instantánea
        - Creación de una AMI y de una copia entre regiones
- Creación de parámetros de entrada que rellenan recursos de AWS
- Uso del Generador de documentos para crear un manual de procedimientos
    - Crear un manual de procedimientos con el Generador de documentos
    - Crear un manual de procedimientos que ejecute scripts
- Uso de scripts en manuales de procedimientos
    - Permisos para utilizar los manuales de procedimientos
    - Incorporación de scripts a los manuales de procedimientos
    - Restricciones de script para los manuales de procedimientos
- Uso de instrucciones condicionales en manuales de procedimientos
    - Uso de la acción aws:branch
        - Creación de un paso aws:branch en un manual de procedimientos
            - Acerca de la creación de la variable de salida
        - Manuales de procedimientos aws:branch de ejemplo
        - Creación de automatizaciones con bifurcación complejas a través de operadores
    - Ejemplos de cómo usar opciones condicionales
- Uso de salidas de acción como entradas
    - Uso de JSONPath en un manual de procedimientos
- Crear integraciones webhook para Automation
    - Creación de integraciones (consola)
    - Creación de integraciones (línea de comandos)
    - Creación de webhooks para integraciones
- Administración de los tiempos de espera en los manuales de procedimientos

Experiencia de diseño visual para manuales de procedimientos de automatización

La automatización AWS Systems Manager proporciona una experiencia de diseño visual low-code que le ayuda a crear manuales de procedimientos de automatización. La experiencia de diseño visual proporciona una interfaz de arrastrar y soltar con la opción de añadir su propio código para que pueda crear y editar manuales con mayor facilidad. Con la experiencia en diseño visual, tiene las siguientes opciones:

- Controle instrucciones condicionales.
- Controle cómo se filtran o transforman la entrada y la salida para cada acción.
- Configure la gestión de errores.
- Cree prototipos de nuevos manuales de procedimientos.
- Utilice sus prototipos de manuales de procedimientos como punto de partida para el desarrollo local con el AWS Toolkit for Visual Studio Code.

Al crear o editar un manual de procedimientos, puede acceder a la experiencia de diseño visual desde la consola de Automatización. A medida que crea un manual de procedimientos, la experiencia de diseño visual valida su trabajo y genera código automáticamente. Puede revisar el código generado o exportarlo para su desarrollo local. Cuando haya terminado, puede guardar el manual de procedimientos, ejecutarlo y examinar los resultados en la consola de automatización de Systems Manager.

Antes de empezar

Para utilizar la experiencia de diseño visual, necesita una Cuenta de AWS y credenciales que proporcionen los permisos correctos para cualquier recurso que quiera utilizar.

En la experiencia de diseño visual, la automatización se integra con Amazon CodeGuru Security para ayudarlo a detectar infracciones y vulnerabilidades de las políticas de seguridad en sus scripts Python. Para utilizar esta característica para acciones `aws:executeScript`, su política AWS Identity and Access Management (de IAM) debe incluir los siguientes permisos:

JSON

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

```
        {
            "Effect": "Allow",
            "Action": [
                "codeguru-security:CreateUploadUrl",
                "codeguru-security:CreateScan",
                "codeguru-security:GetScan",
                "codeguru-security:GetFindings"
            ],
            "Resource": "*"
        }
    ]
}
```

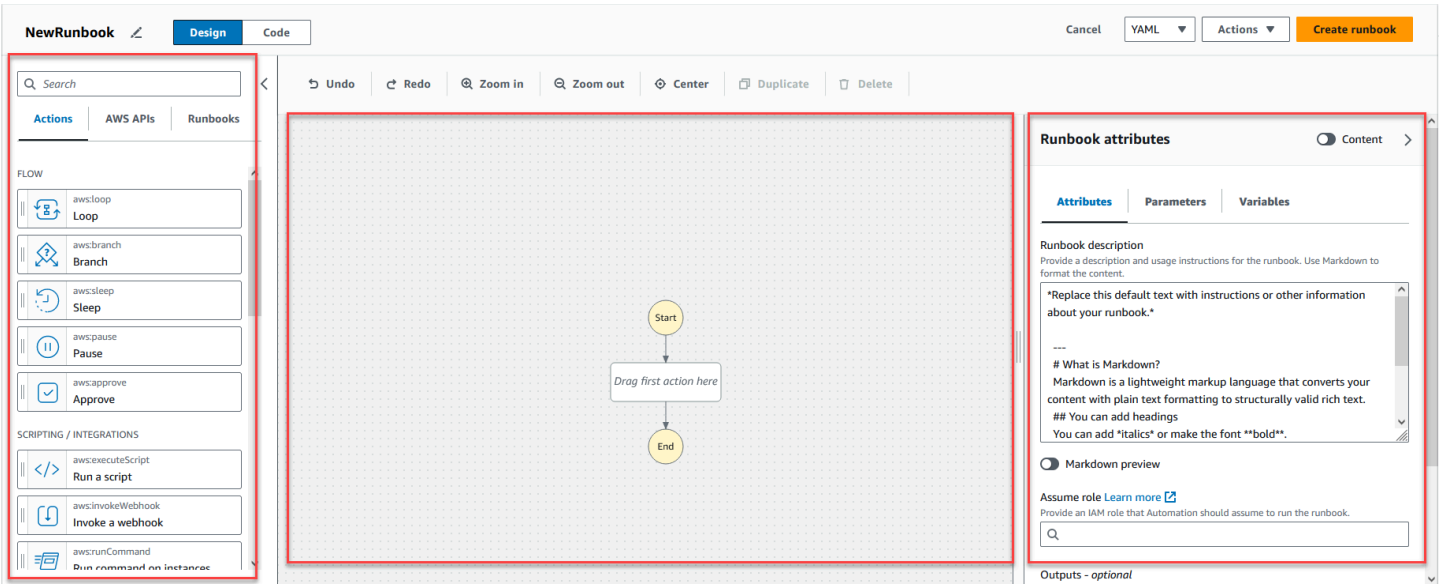
Temas

- Descripción general de la interfaz de experiencia de diseño visual
- Uso de la experiencia de diseño visual
- Configuración de entradas y salidas para sus acciones
- Manejo de errores con la experiencia de diseño visual
- Tutorial: cómo crear un manual de procedimientos utilizando la experiencia de diseño visual

Descripción general de la interfaz de experiencia de diseño visual

La experiencia de diseño visual de la automatización de Systems Manager consiste en un diseñador visual de flujos de trabajo low-code que le ayuda a crear manuales de procedimientos de automatización.

Conozca la experiencia de diseño visual con una descripción general de los componentes de la interfaz:



- El navegador de Acciones contiene las pestañas Acciones, AWSAPI y Runbooks.
- El lienzo es donde usted arrastra y suelta las acciones en el gráfico del flujo de trabajo, se cambia el orden de las acciones y se seleccionan las acciones que se van a configurar o ver.
- El panel Formulario es el lugar donde puede ver y editar las propiedades de cualquier acción que haya seleccionado en el lienzo. Seleccione el botón Contenido para ver el formato YAML o JSON de su manual de procedimientos, con la acción actualmente seleccionada resaltada.

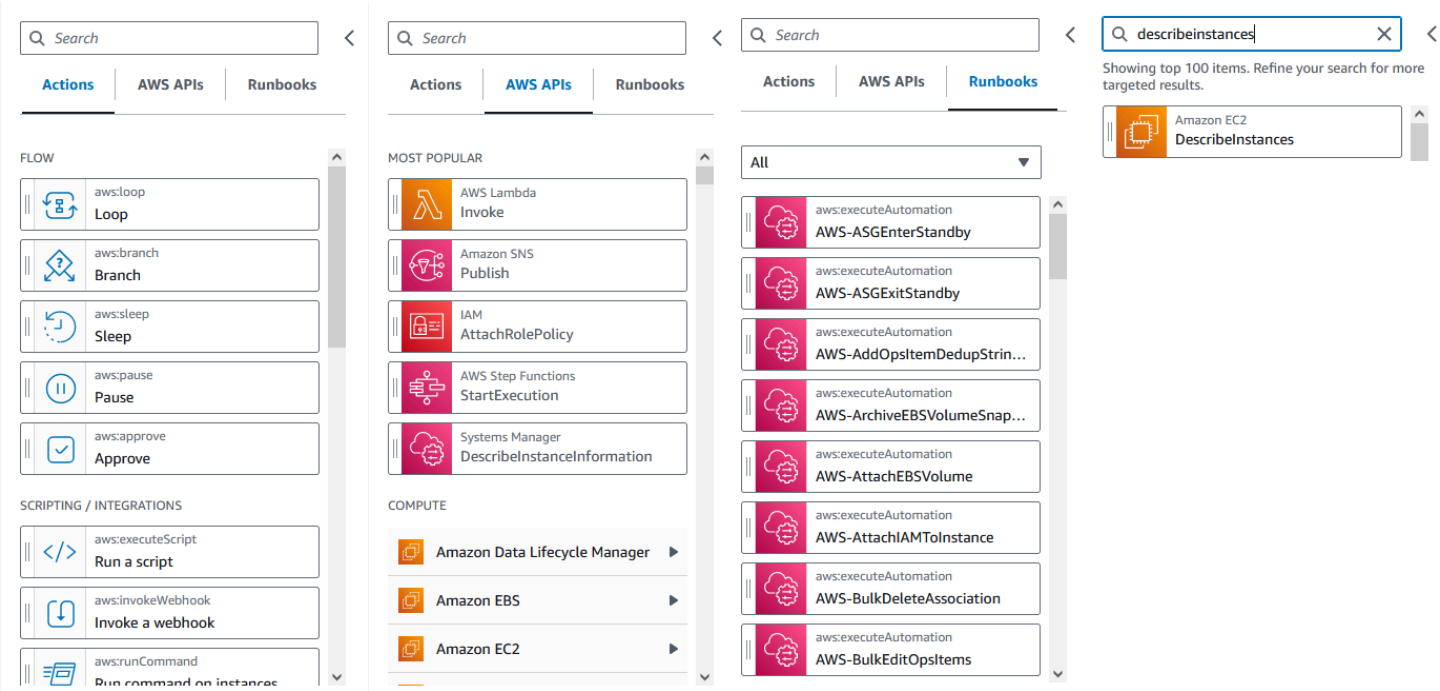
Los enlaces de Información abren un panel con información contextual cuando necesita ayuda. Estos paneles también incluyen enlaces a temas relacionados en la documentación de automatización de Systems Manager.

Navegador de acciones

Desde el navegador de Acciones, puede seleccionar acciones para arrastrarlas y soltarlas en su gráfico de flujo de trabajo. Puede buscar todas las acciones mediante el campo de búsqueda situado en la parte superior del navegador de Acciones. El navegador de Acciones contiene las siguientes pestañas:

- La pestaña Acciones proporciona una lista de acciones de automatización que puede arrastrar y soltar en el gráfico de flujo de trabajo del manual de procedimientos en el lienzo.
- La pestaña AWS de APIs proporciona una lista de AWS APIs que puede arrastrar y soltar en el gráfico de flujo de trabajo de su manual de procedimientos en el lienzo.

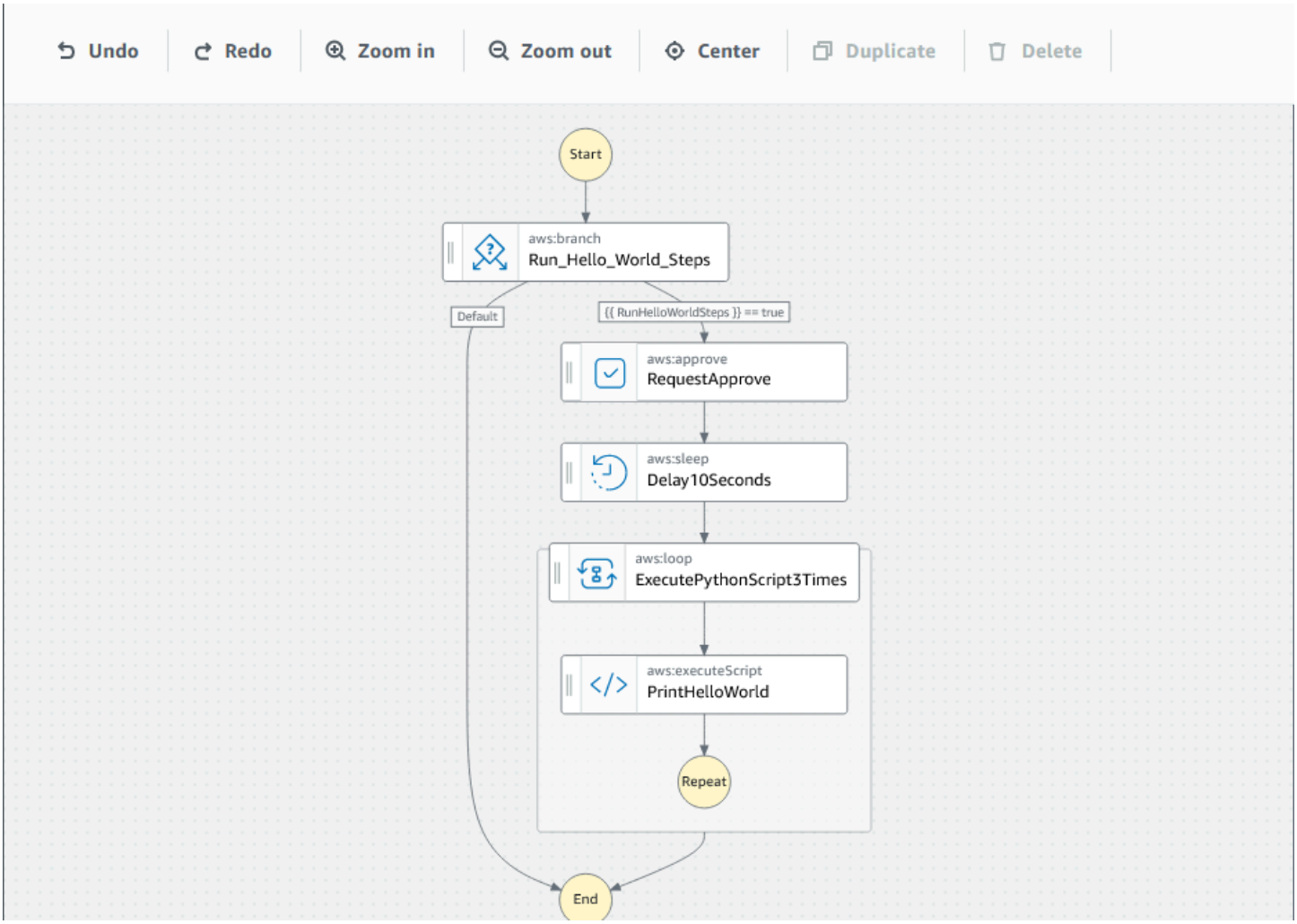
- La pestaña Runbooks incluye varios manuales de procedimientos reutilizables y listos para usar como bloques de construcción que puede usar para una variedad de casos de uso. Por ejemplo, puede utilizar los manuales de procedimientos para realizar tareas de corrección habituales en las instancias de Amazon EC2 de su flujo de trabajo sin tener que volver a crear las mismas acciones.



Canvas

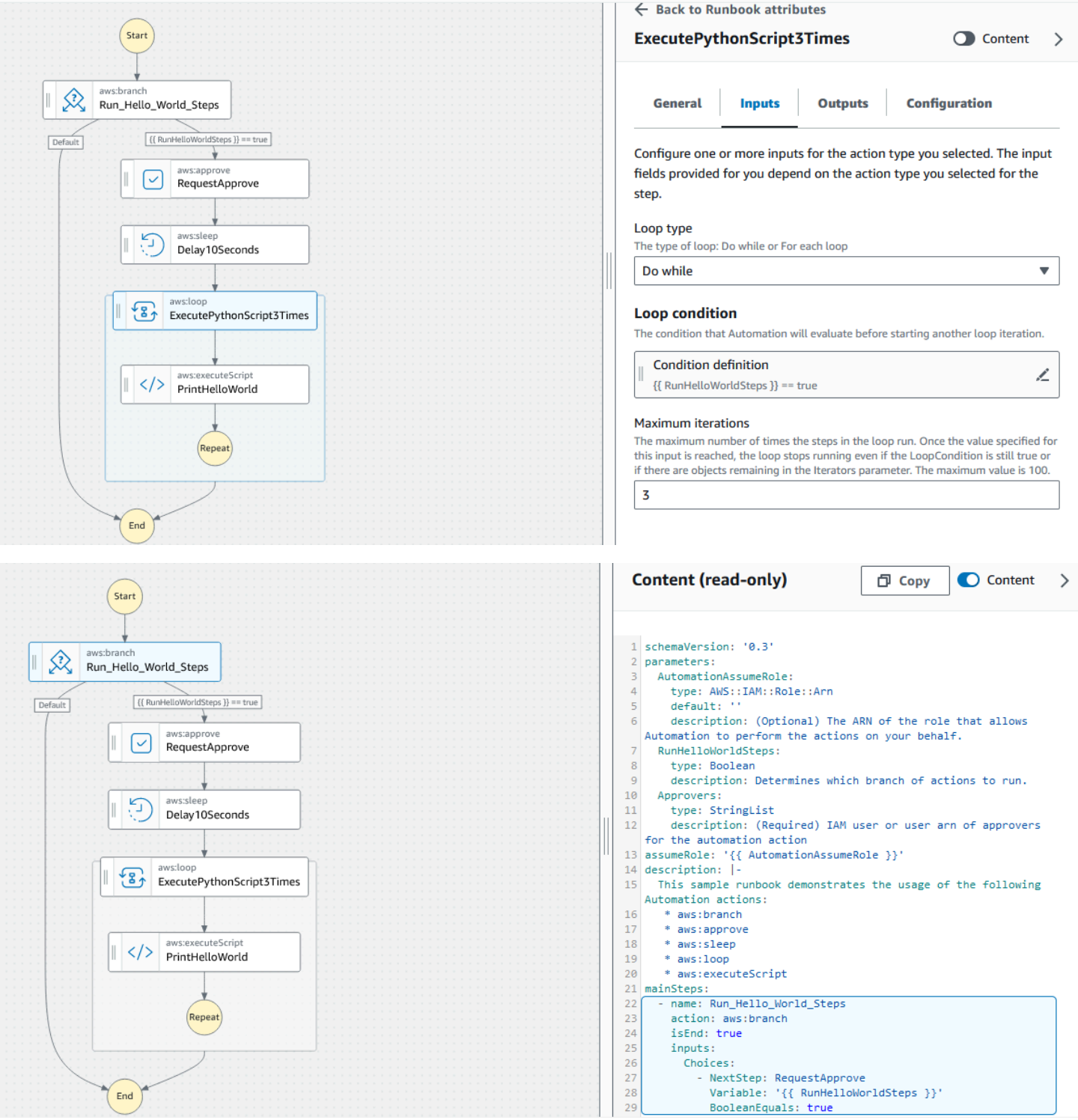
Después de elegir una acción para añadirla a su automatización, arrástrela al lienzo y suéltela en el gráfico de flujo de trabajo. También puede arrastrar y soltar acciones para moverlas a diferentes lugares del flujo de trabajo de su manual de procedimientos. Si su flujo de trabajo es complejo, es posible que no pueda verlo en su totalidad en el panel del lienzo. Use los controles de la parte superior del lienzo para acercar o alejar la imagen. Para ver diferentes partes de un flujo de trabajo, puede arrastrar el gráfico del flujo de trabajo al lienzo.

Arrastre una acción desde el navegador de Acciones y suéltela en el gráfico de flujo de trabajo de su manual de procedimientos. Una línea muestra dónde se colocará en su flujo de trabajo. Para cambiar el orden de una acción, puede arrastrarla a un lugar diferente de su flujo de trabajo. La nueva acción se ha añadido a su flujo de trabajo y su código se genera automáticamente.



Formulario

Después de añadir una acción a su flujo de trabajo del manual de procedimientos, puede configurarla para que se adapte a su caso de uso. Elija la acción que desee configurar y verá sus parámetros y opciones en el panel Formulario. También puedes ver el código YAML o JSON pulsando el botón Contenido. El código asociado a la acción que ha seleccionado aparece resaltado.



Métodos abreviados de teclado

La experiencia de diseño visual es compatible con los atajos de teclado que se muestran en la siguiente tabla.

Método abreviado de teclado	Método abreviado de teclado
Ctrl + Z	Deshacer la última operación.
Ctrl + Mayús + Z	Rehacer la última operación.
Alt + 0	Centrar el flujo de trabajo en el lienzo.
Ctrl + A	Eliminar todos los estados seleccionados.
Ctrl + A	Elimine todos los estados

Método abreviado de teclado	Método abreviado de teclado
seleccionados.	seleccionados.
Ctrl+D	Ctrl+D
estado seleccionado.	estado seleccionado.

Uso de la experiencia de diseño visual

Aprenda a crear, editar y ejecutar flujos de trabajo de manual de procedimientos utilizando la experiencia de diseño visual. Una vez que el flujo de trabajo esté listo, puede guardarlo o exportarlo. También puede utilizar la experiencia de diseño visual para crear prototipos rápidamente.

Cree de un flujo de trabajo del manual de procedimientos

1. Inicie sesión en la consola de automatización de Systems Manager.
2. Seleccione Crear runbook.
3. En el cuadro Nombre , escriba un nombre para el manual de procedimientos, por ejemplo,*MyNewRunbook*.
4. Junto a los botones Diseño y Código, seleccione el icono del lápiz e introduzca un nombre para el manual de procedimientos.

Ahora puede diseñar un flujo de trabajo para su nuevo manual de procedimientos.

Diseñe un manual de procedimientos

Para diseñar un flujo de trabajo de manual de procedimientos utilizando la experiencia de diseño visual, arrastre una acción de automatización desde el navegador de Acciones al lienzo y sitúela en el lugar que desee en el flujo de trabajo del manual de procedimientos. También puede reordenar las acciones de su flujo de trabajo arrastrándolas a una ubicación diferente. A medida que arrastra una

acción al lienzo, aparece una línea en el lugar donde puede colocar la acción en el flujo de trabajo. Cuando una acción se coloca en el lienzo, su código se genera automáticamente y se añade al contenido del manual de procedimientos.

Si sabe el nombre de la acción que quiere añadir, utilice el cuadro de búsqueda situado en la parte superior del navegador de Acciones para buscarla.

Después de colocar una acción en el lienzo, configúrela mediante el panel Formulario de la derecha. Este panel contiene las pestañas General, Entradas, Salidas y Configuración para cada acción de automatización o acción de API que coloque en el lienzo. Por ejemplo, la pestaña General contiene las siguientes secciones:

- El Nombre del paso identifica el paso. Especifique un valor único para el nombre del paso.
- La Descripción le ayuda a describir lo que está haciendo la acción en el flujo de trabajo de su manual de procedimientos.

La pestaña Entradas contiene campos que varían en función de la acción. Por ejemplo, la acción de automatización de `aws:executeScript` contiene las siguientes secciones:

- El Tiempo de ejecución es el lenguaje a usar para ejecutar el script proporcionado.
- El Controlador es el nombre de su función. Debe asegurarse de que la función definida en el controlador tenga dos parámetros: `events` y `context`. El tiempo de ejecución de PowerShell no admite este parámetro.
- El Script es un script insertado que desea ejecutar durante el flujo de trabajo.
- (Opcional) El Archivo adjunto es para scripts independientes o archivos.zip que la acción puede invocar. Este parámetro es obligatorio para los manuales de procedimientos JSON.

La pestaña Salidas le ayuda a especificar los valores que desea generar de una acción. Puede hacer referencia a los valores de salida en acciones posteriores de su flujo de trabajo o generar resultados para fines de registro. No todas las acciones tendrán una pestaña de Salidas porque no todas las acciones admiten salidas. Por ejemplo, la acción `aws:pause` no admite salidas. En el caso de las acciones que son compatibles con salidas, la pestaña Salidas consta de las siguientes secciones:

- El Nombre es el nombre que se utilizará para el valor de salida. Puede hacer referencia a los resultados en acciones posteriores de su flujo de trabajo.
- El Selector es una expresión de cadena JSONPath que comienza con `"$."` que se utiliza para seleccionar uno de varios componentes dentro de un elemento JSON.

- El Tipo es el tipo de datos del valor de salida. Por ejemplo, escriba tipo de dato `String` o `Integer`.

La pestaña Configuración contiene propiedades y opciones que pueden utilizar todas las acciones de automatización. La acción consta de las secciones siguientes:

- La propiedad Cantidad máxima de intentos es el número de veces que se reintenta una acción si se produce un error.
- La propiedad Tiempo de espera en segundos especifica el valor de tiempo de espera de una acción.
- La propiedad Es crítico determina si el error de la acción detiene toda la automatización.
- La propiedad Siguiente paso determina la siguiente acción que debe realizar la automatización en el manual de procedimientos.
- La propiedad En caso de error determina la siguiente acción de la automatización en el manual de procedimientos en caso de que la acción falle.
- La propiedad Al cancelar determina la siguiente acción de la automatización en el manual de procedimientos si un usuario cancela la acción.

Para eliminar una acción, puedes usar la barra de herramientas situada sobre el lienzo o hacer clic con el botón derecho del ratón y seleccionar Eliminar acción.

A medida que el flujo de trabajo vaya creciendo, es posible que no quepa en el lienzo. Pruebe una de las siguientes opciones para que el flujo de trabajo encaje en el lienzo:

- Use los controles de los paneles laterales para cambiar el tamaño de los paneles o cerrarlos.
- Utilice la barra de herramientas situada en la parte superior del lienzo para acercar o alejar el gráfico del flujo de trabajo.

Actualice su manual de procedimientos

Puedes actualizar el flujo de trabajo de un manual de procedimientos existente creando una nueva versión del manual. Las actualizaciones de sus manuales de procedimientos se pueden realizar utilizando la experiencia de diseño visual o editando el código directamente. Para actualizar un manual de procedimientos existente, siga el siguiente procedimiento:

1. Inicie sesión en la consola de automatización de Systems Manager.

2. Elija el manual de procedimientos que desea actualizar.
3. Elija Create new version (Crear nueva versión).
4. La experiencia de diseño visual tiene dos paneles: un panel de códigos y un panel de flujo de trabajo visual. Elija Diseño en el panel de flujo de trabajo visual para editar su flujo de trabajo con la experiencia de diseño visual. Cuando haya terminado, elija Crear nueva versión para guardar los cambios y salir.
5. (Opcional) Use el panel de códigos para editar el contenido del manual de procedimientos en YAML o JSON.

Exporte su manual de procedimientos

Para exportar el código YAML o JSON del flujo de trabajo de su manual de procedimientos y también un gráfico del flujo de trabajo, siga el siguiente procedimiento:

1. Elija tu manual de procedimientos en la consola Documentos.
2. Elija Create new version (Crear nueva versión).
3. En el menú desplegable Acciones, elige si quiere exportar el gráfico o el manual de procedimientos y el formato que prefiera.

Configuración de entradas y salidas para sus acciones

Cada acción de automatización responde en función de las entradas que recibe. En la mayoría de los casos, se pasa el resultado a las acciones siguientes. En la experiencia de diseño visual, puede configurar los datos de entrada y salida de una acción en las pestañas de Entradas y Salidas del panel de Formulario.

Para obtener información detallada acerca de cómo definir y utilizar la salida para las acciones de automatización, consulte Uso de salidas de acción como entradas.

Proporcione datos de entrada para una acción

Cada acción de automatización tiene una o más entradas para las que debe proporcionar un valor. El valor que proporcione para la entrada de una acción viene determinado por el tipo de datos y el formato que acepte la acción. Por ejemplo, las acciones `aws:sleep` requieren un valor de cadena con formato ISO 8601 para la entrada `Duration`.

Por lo general, en el flujo de trabajo del manual de procedimientos se utilizan acciones que devuelven los resultados que se desean utilizar en acciones posteriores. Es importante que se

asegure de que los valores de entrada son correctos para evitar errores en el flujo de trabajo del manual de procedimientos. Los valores de entrada también son importantes porque determinan si la acción devuelve el resultado esperado. Por ejemplo, al usar la acción `aws:executeAwsApi`, debe asegurarse de que está proporcionando el valor correcto para la operación de la API.

Defina los datos de salida de una acción

Algunas acciones de automatización devuelven la salida después de realizar las operaciones definidas. Las acciones que devuelven salidas tienen resultados predefinidos o le permiten definir las salidas usted mismo. Por ejemplo, la acción `aws:createImage` tiene salidas predefinidas que devuelven un `ImageId` y `ImageState`. Comparativamente, con la acción `aws:executeAwsApi`, puede definir las salidas que desea obtener de la operación de API especificada. Como resultado, puede devolver uno o más valores de una sola operación de API para usarlos en acciones posteriores.

Para definir sus propios resultados para una acción de automatización, debe especificar el nombre del resultado, el tipo de datos y el valor del resultado. Para seguir utilizando la `aws:executeAwsApi` acción como ejemplo, supongamos que llama a la operación de `DescribeInstances` API desde Amazon EC2. En este ejemplo, desea devolver o generar el `State` de una instancia de Amazon EC2 y ramificar el flujo de trabajo de su manual de procedimientos en función del resultado. Puede elegir un nombre para la salida **`InstanceState`** y usar el tipo de datos **`String`**.

El proceso para definir el valor real de la salida varía según la acción. Por ejemplo, si utiliza la acción `aws:executeScript`, debe utilizar argumentos de `return` en las funciones para proporcionar datos a las salidas. Con otras acciones como `aws:executeAwsApiaws:waitForAwsResourceProperty`, y `aws:assertAwsResourceProperty`, se requiere un `Selector`. El `Selector`, o `PropertySelector` como lo denominan algunas acciones, es una cadena JSONPath que se utiliza para procesar la respuesta JSON de una operación de API. Es importante entender cómo está estructurado el objeto de respuesta JSON de una operación de API para poder seleccionar el valor correcto para la salida. Con la operación `DescribeInstances` de API mencionada anteriormente, consulte el siguiente ejemplo de respuesta JSON:

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

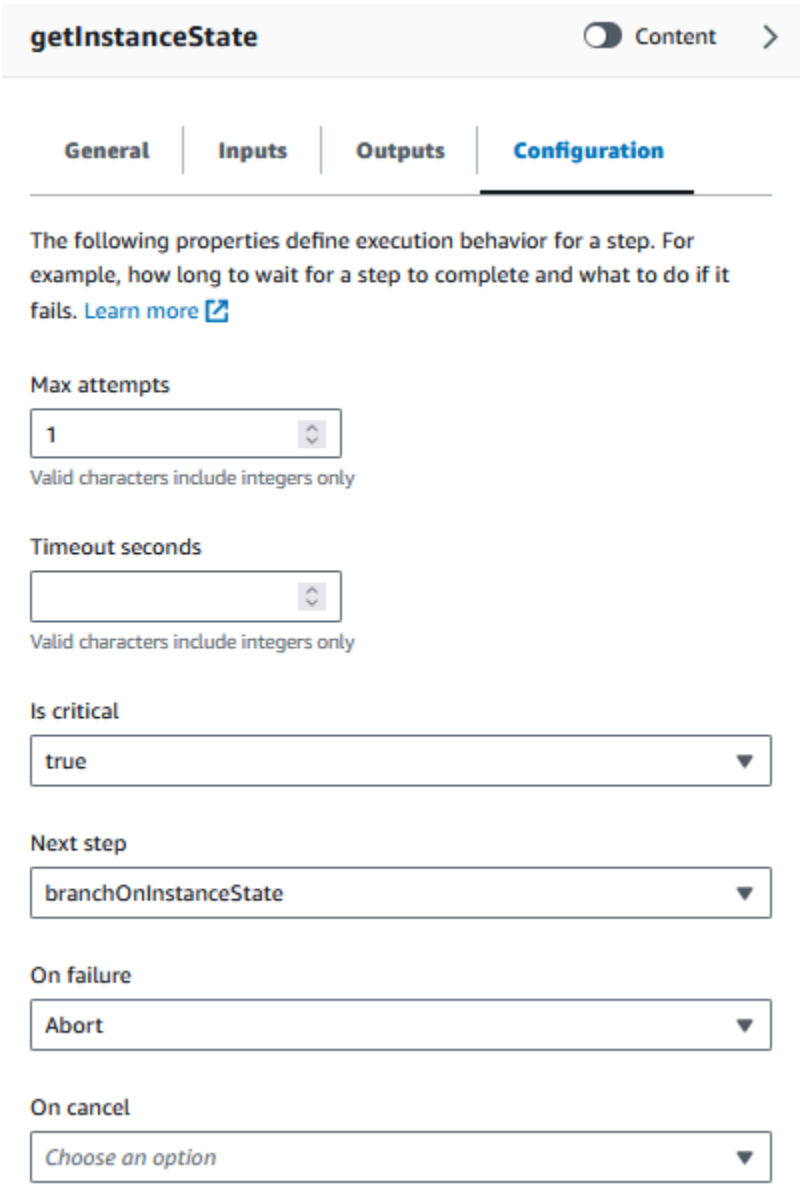
En el objeto de respuesta JSON, la instancia `State` está anidada en un objeto `Instances`, que está anidado en el objeto `Reservations`. Para devolver el valor de la instancia `State`, use la siguiente cadena para el `Selector` para que el valor se pueda usar en nuestra salida: **`$.Reservations[0].Instances[0].State.Name`**.

Para hacer referencia a un valor de salida en acciones posteriores del flujo de trabajo de su manual de procedimientos, se utiliza el siguiente formato: `{{ StepName.NameOfOutput }}`. Por ejemplo,

`{{ GetInstanceState.InstanceState }}`. En la experiencia de diseño visual, puede elegir valores de salida para usarlos en acciones posteriores utilizando el menú desplegable de entrada. Al utilizar salidas en acciones posteriores, el tipo de datos de la salida debe coincidir con el tipo de datos de la entrada. En este resultado de ejemplo, la `InstanceState` salida es una `String`. Por lo tanto, para usar el valor en la entrada de una acción posterior, la entrada debe aceptar una `String`.

Manejo de errores con la experiencia de diseño visual

De forma predeterminada, cuando una acción informa de un error, la automatización detiene por completo el flujo de trabajo del manual de procedimientos. Esto se debe a que el valor predeterminado de la propiedad `onFailure` en todas las acciones es `Abort`. Puedes configurar la forma en que la automatización gestiona los errores en el flujo de trabajo de su manual de procedimientos. Incluso si ha configurado la gestión de errores, es posible que algunos errores provoquen un error en la automatización. Para obtener más información, consulte Solución de problemas de Automatización de Systems Manager. En la experiencia de diseño visual, la gestión de errores se configura en el panel de Configuración.



Vuelva a intentar la acción en caso de error

Para volver a intentar una acción en caso de error, especifique un valor para la propiedad Cantidad máxima de intentos. El valor predeterminado es 1. Si especifica un valor mayor a 1, no se considerará que hay un error en la acción hasta que todos los reintentos produzcan errores.

Tiempos de espera

Puede configurar un tiempo de espera para las acciones para establecer el número máximo de segundos que la acción puede ejecutarse antes de que se produzca un error. Para configurar un tiempo de espera, introduzca el número de segundos que debe esperar la acción antes de que se produzca un error en la propiedad Tiempo de espera en segundos. Si se alcanza el tiempo de espera

y el la acción tiene un valor de `Max attempts` que es mayor que 1, no se considera que se ha agotado el tiempo de espera hasta que se reintente completo.

Acciones fallidas

De forma predeterminada, cuando se produce un error en una acción, la automatización detiene por completo el flujo de trabajo del manual de procedimientos. Puede modificar este comportamiento especificando un valor alternativo para la propiedad En caso de error de las acciones del manual de procedimientos. Si desea que el flujo de trabajo continúe con el siguiente paso del manual de procedimientos, elija Continuar. Si desea que el flujo de trabajo salte a un paso posterior diferente del manual de procedimientos, elija Paso y, a continuación, introduzca el nombre del paso.

Acciones canceladas

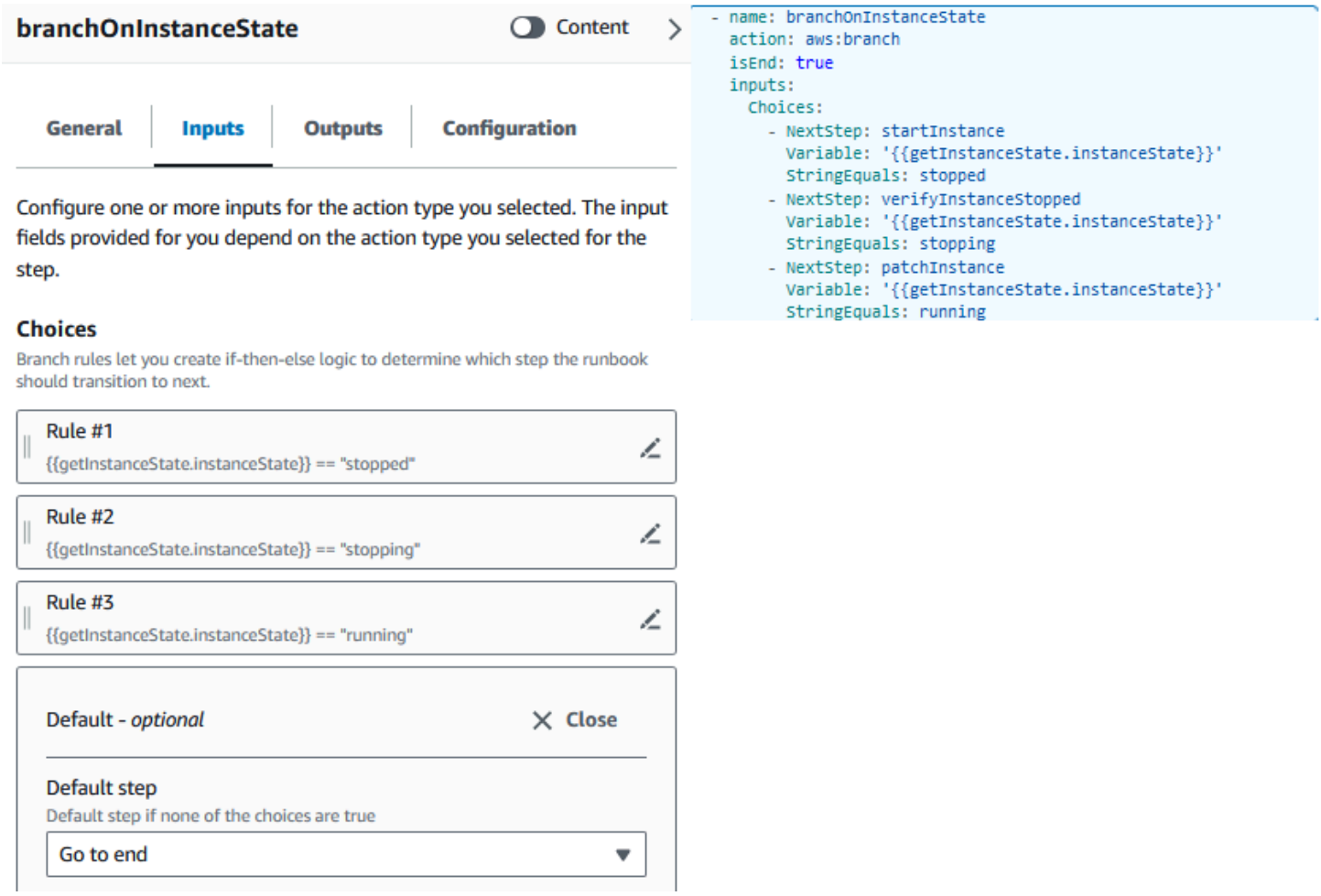
De forma predeterminada, cuando un usuario cancela una acción, la automatización detiene por completo el flujo de trabajo del manual de procedimientos. Puede modificar este comportamiento especificando un valor alternativo para la propiedad Al cancelar de las acciones de su manual de procedimientos. Si desea que el flujo de trabajo salte a un paso posterior diferente del manual de procedimientos, elija Paso y, a continuación, introduzca el nombre del paso.

Acciones cruciales

Puede designar una acción como crítica, lo que significa que determina el estado general de los informes de su automatización. Si un paso con esta designación genera un error, la automatización informa que el estado final es `Failed` independientemente del éxito de otras acciones. Para configurar una acción como crítica, deje el valor predeterminado como Verdadero para la propiedad Es crítico.

Finalización de acciones

La propiedad Finaliza detiene una automatización al final de la acción determinada. El valor predeterminado de esta propiedad es `false`. Si configura esta propiedad para una acción, la automatización se detiene tanto si la acción se realiza correctamente como si no. Esta propiedad se utiliza con mayor frecuencia con acciones `aws:branch` para gestionar valores de entrada inesperados o indefinidos. En el siguiente ejemplo, se muestra un manual de procedimientos que espera un estado de instancia igual a `running`, `stopping` o. `stopped` Si una instancia está en un estado diferente, la automatización finaliza.



Tutorial: cómo crear un manual de procedimientos utilizando la experiencia de diseño visual

En este tutorial, aprenderá los conceptos básicos del trabajo con la experiencia de diseño visual que proporciona la automatización de Systems Manager. En la experiencia de diseño visual, puede crear un manual de procedimientos que utilice múltiples acciones. Utilice la característica de arrastrar y soltar para organizar las acciones en el lienzo. También puede buscar, seleccionar y configurar estas acciones. Luego, puedes ver el código YAML generado automáticamente para el flujo de trabajo de su manual de procedimientos, salir de la experiencia de diseño visual, ejecutar el manual de procedimientos y revisar los detalles de la ejecución.

En este tutorial también se muestra cómo actualizar el manual de procedimientos y ver la nueva versión. Al final del tutorial, realizará un paso de limpieza y eliminará su manual de procedimientos.

Después de completar este tutorial, sabrá cómo usar la experiencia de diseño visual para crear un manual de procedimientos. También sabrá cómo actualizar, ejecutar y eliminar su manual de procedimientos.

> **Note**
>
> Antes de empezar este tutorial, asegúrese de completar Configuración de Automation.

Temas

- Paso 1: navegue hasta la experiencia de diseño visual
- Paso 2: Cree un flujo de trabajo
- Paso 3: Revise el código generado automáticamente
- Paso 4: Ejecute su nuevo manual de procedimientos
- Paso 5: Eliminar

Paso 1: navegue hasta la experiencia de diseño visual

1. Inicie sesión en la consola de automatización de Systems Manager.
2. Elija Crear runbook de automatización.

Paso 2: Cree un flujo de trabajo

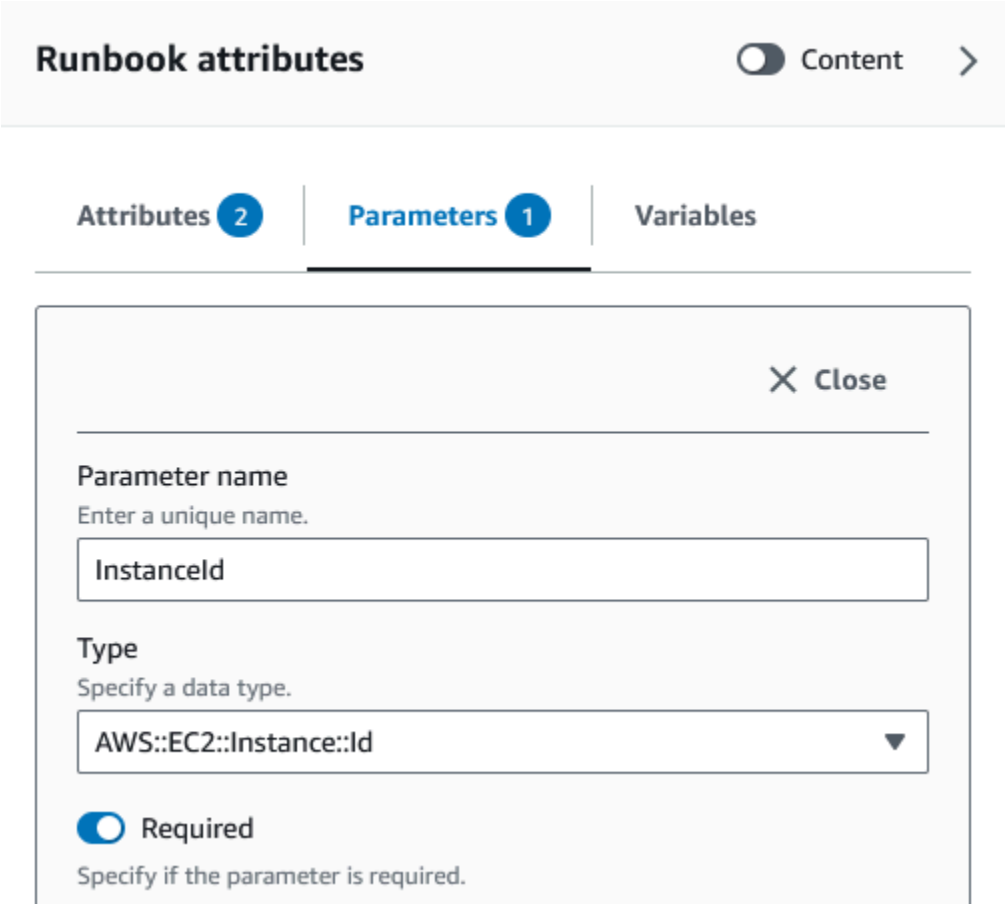
En la experiencia de diseño visual, un flujo de trabajo es una representación gráfica de su manual de procedimientos en el lienzo. Puede utilizar la experiencia de diseño visual para definir, configurar y examinar las acciones individuales de su manual de procedimientos.

Para crear un flujo de trabajo

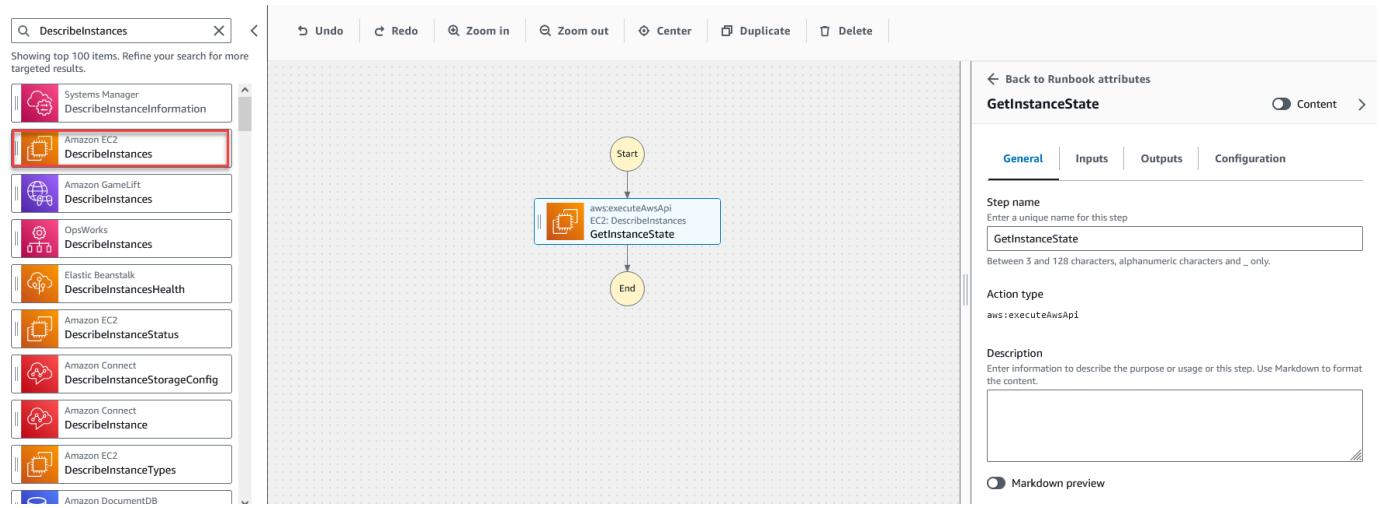
1. Junto a los botones Diseño y Código, seleccione el icono del lápiz e introduzca un nombre para el manual de procedimientos. En este tutorial, escriba **`VisualDesignExperienceTutorial`**.



2. En la sección Atributos del documento del panel Formulario, expanda el menú desplegable Parámetros de entrada y seleccione Agregar un parámetro.
   a. En el campo Nombre del parámetro, introduzca **`InstanceId`**.
   b. En el menú desplegable Tipo, elija AWS::EC2::Instance.
   c. Seleccione el botón Requerido.



3. En el navegador de API de AWS, ingrese **`DescribeInstances`** en la barra de búsqueda.
4. Arrastre una acción Amazon EC2 – DescribeInstances al lienzo vacío.
5. Para Nombre del paso, ingrese un valor. Para este tutorial, puede utilizar el nombre **`GetInstanceState`**.



a. Amplíe el menú desplegable de Entradas adicionales y, en el campo Nombre de entrada, introduzca **`InstanceIds`**.

b. Seleccione la pestaña Entradas.

c. En el campo Valor de entrada, elija la entrada del documento **`InstanceId`**. Esto hace referencia al valor del parámetro de entrada que haya creado al principio del procedimiento. Como la entrada InstanceIds de la acción `DescribeInstances` acepta valores `StringList`, debes escribir la entrada de InstanceId entre corchetes. El YAML del valor de entrada debe coincidir con lo siguiente: **`['{{ InstanceId }}']`**.

d. En la pestaña Salidas, seleccione Añadir una salida e **`InstanceState`** introdúzcala en el campo Nombre.

e. En el campo Selector, introduzca **`$.Reservations[0].Instances[0].State.Name`**.

f. En el menú desplegable Tipo, seleccione Cadena.

6. Arrastre una acción Ramificación desde el navegador Acciones y suéltela debajo del paso **`GetInstanceState`**.

7. Para Nombre del paso, escriba un valor. Para este tutorial, use el nombre **`BranchOnInstanceState`**.

   Para definir la lógica de ramificación, haga lo siguiente:

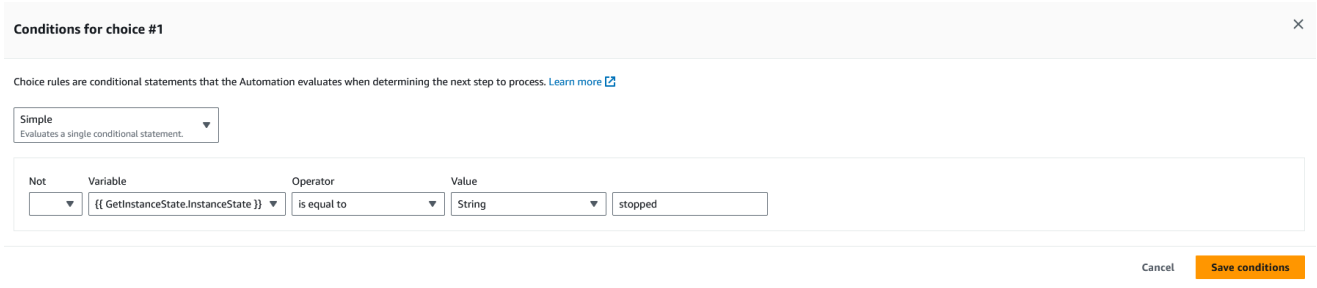
   a. Elija el estado **`Branch`** en el lienzo. Luego, en Entradas y Opciones, seleccione el icono del lápiz para editar la Regla #1.

   b. Elija Agregar condiciones.

   c. En el cuadro de diálogo Condiciones de la regla #1, elija la salida del paso **`GetInstanceState.InstanceState`** en el menú desplegable Variable.

   d. En Operador, seleccione es igual a.

   e. Para Valor, escoja Cadena de la lista desplegable. Escriba **`stopped`**.



   f. Seleccione Guardar condiciones.

g. Seleccione Agregar nueva regla.

h. Elija Añadir condiciones para la Regla #2.

i. En el cuadro de diálogo Condiciones de la regla #2, elija la salida del paso **`GetInstanceState.InstanceState`** en el menú desplegable Variable.

j. En Operador, seleccione es igual a.

k. Para Valor, escoja Cadena de la lista desplegable. Escriba **`stopping`**.

l. Seleccione Guardar condiciones.

m. Seleccione Agregar nueva regla.

n. Para la Regla #3, seleccione Añadir condiciones.

o. En el cuadro de diálogo Condiciones de la regla #3, elija la salida del paso **`GetInstanceState.InstanceState`** en el menú desplegable Variable.

p. En Operador, seleccione es igual a.

q. Para Valor, escoja Cadena de la lista desplegable. Escriba **`running`**.

r. Seleccione Guardar condiciones.

s. En la Regla predeterminada, seleccione Ir al final para el Paso predeterminado.

8. Arrastre una acción para cambiar el estado de la instancia hasta el cuadro vacío Arrastrar la acción aquí bajo la condición {{getInstanceState.instanceState}} == “stopped”.

   a. Para el Nombre del paso, introduzca **`StartInstance`**.

   b. En la pestaña Entradas, bajo ID de instancia, escoja el valor de entrada del documento InstanceId del desplegable.

   c. Para el Estado deseado, especifique **`running`**.

9. Arrastre una acción Esperar un recurso de AWS hasta el cuadro vacío Arrastrar la acción aquí bajo la condición {{getInstanceState.instanceState}} == “stopping”.

10. Para Nombre del paso, escriba un valor. Para este tutorial, use el nombre **`WaitForInstanceStop`**.

    a. Para el campo Servicio, elija Amazon EC2.

    b. Para el campo API, elija Describir instancias.

    c. Para el campo Selector de propiedades, introduzca **`$.Reservations[0].Instances[0].State.Name`**.

    d. Para el parámetro Valores deseados, introduzca **`["stopped"]`**.

e. En la pestaña Configuración de la acción Esperar por la detención de la instancia, escoja Iniciar instancia del desplegable Paso siguiente.

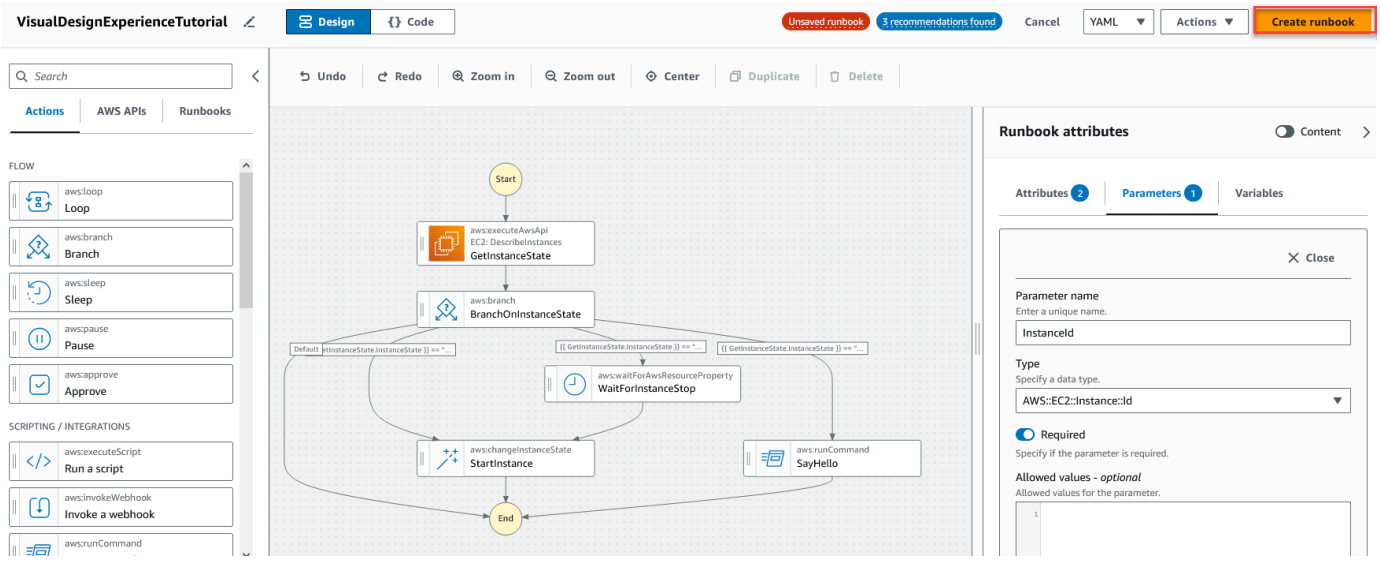
11. Arrastre una acción Ejecutar el comando en instancias hasta el cuadro vacío Arrastrar acción aquí bajo la condición {{getInstanceState.instanceState}} == “running”.
12. Para el Nombre del paso, introduzca **`SayHello`**.

    a. En la pestaña Entradas, introduzca **`AWS-RunShellScript`** para el parámetro Nombre del documento.

    b. Para ID de instancia, elija el valor de entrada del documento ID de instancia en el menú desplegable.

    c. Expanda el desplegable Entradas adicionales y, en el menú desplegable Nombre de entrada, escoja Parámetros.

    d. En el campo Valor de entrada, introduzca **`{"commands": "echo 'Hello World'"}`**.
13. Revisa el manual de procedimientos completo en el lienzo y seleccione Crear runbook para guardar el manual de procedimientos del tutorial.



Paso 3: Revise el código generado automáticamente

Al arrastrar y soltar acciones desde el navegador de Acciones al lienzo, la experiencia de diseño visual compone automáticamente el contenido YAML o JSON de su manual de procedimientos en tiempo real. Puede ver y editar este código. Para ver el código generado automáticamente, seleccione Código para los botones Diseño y Código.

Paso 4: Ejecute su nuevo manual de procedimientos

Después de crear su manual de procedimientos, puede ejecutar la automatización.

Para ejecutar su nuevo manual de procedimientos de automatización

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Automatización y, después, seleccione Ejecutar automatización.
3. En la lista Documento de automatización, elija un manual de procedimientos. Elija una o más opciones en el panel Categorías de documentos para filtrar documentos SSM según su propósito. Para ver un manual de procedimientos que le pertenezca, seleccione la pestaña De mi propiedad. Para ver un manual de procedimientos que se haya compartido con su cuenta, elija la pestaña Compartido conmigo. Para ver todos los manuales de procedimientos, seleccione la pestaña Todos los documentos.

   > **Note**
   >
   > Puede ver información acerca de un manual de procedimientos al seleccionar su nombre.

4. En la sección Detalles del documento, verifique que Versión del documento esté establecido como la versión que desea ejecutar. El sistema incluye las siguientes opciones de versión:

   - Versión predeterminada en tiempo de ejecución: seleccione esta opción si el manual de procedimientos de automatización se actualiza de forma periódica y se asigna una nueva versión predeterminada.
   - Última versión en tiempo de ejecución: seleccione esta opción si el manual de procedimientos de automatización se actualiza de forma periódica y desea ejecutar la versión que se ha actualizado más recientemente.
   - 1 (Predeterminado): seleccione esta opción para ejecutar la primera versión del documento, que es la predeterminada.

5. Elija Siguiente.
6. En la página Ejecutar el runbook de automatización, elija Ejecución simple.
7. En la sección Parámetros de entrada, especifique las entradas necesarias: De forma opcional, puede elegir un rol de servicio de IAM de la lista AutomationAssumeRole.

8. (Opcional) Elija una alarma de Amazon CloudWatch que desee aplicar a la automatización para fines de monitoreo. Para adjuntar una alarma de CloudWatch a su automatización, la entidad principal de IAM que ejecuta esta última debe tener permiso para la acción `iam:createServiceLinkedRole`. Para obtener más información sobre las alarmas de CloudWatch, consulte Uso de alarmas de Amazon CloudWatch. Si la alarma se activa, la automatización se detiene. Si usa AWS CloudTrail, verá la llamada a la API en el registro de seguimiento.
9. Elija Ejecutar.

Paso 5: Eliminar

Para eliminar el manual de procedimientos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. Elija la pestaña De mi propiedad.
4. Busque el manual de procedimientos de Experiencia de diseño visual.
5. Seleccione el botón en la página de la tarjeta del documento y, a continuación, Eliminar documento del desplegable Acciones.

Creación de manuales de procedimientos de Automation

Cada manual de procedimientos en Automatización, una herramienta de AWS Systems Manager, define una automatización. Los manuales de procedimientos de Automation definen las acciones que se realizan durante una automatización. En el contenido del manual de procedimientos, defina los parámetros de entrada, las salidas y las acciones que Systems Manager realiza en las instancias administradas y en los recursos de AWS.

Automation incluye varios manuales de procedimientos predefinidos que se pueden utilizar para realizar tareas comunes, como reiniciar una o más instancias de Amazon Elastic Compute Cloud (Amazon EC2), o crear una Amazon Machine Image (AMI). Sin embargo, los casos de uso pueden ir más allá de las capacidades de los manuales de procedimientos predefinidos. Si este es el caso, puede crear sus propios manuales de procedimientos y modificarlos de acuerdo con sus necesidades.

Un manual de procedimientos consta de acciones de automatización, parámetros para esas acciones y parámetros de entrada que usted especifica. El contenido de un manual de procedimientos está escrito en YAML o JSON. Si no conoce YAML ni JSON, recomendamos usar el diseñador visual o aprender más sobre cualquiera de los lenguajes de marcado antes de intentar crear su propio manual de procedimientos. Para obtener más información sobre el diseñador visual, consulte Experiencia de diseño visual para manuales de procedimientos de automatización.

Las siguientes secciones lo ayudarán a crear su primer manual de procedimientos.

Identifique su caso de uso

El primer paso para crear un manual de procedimientos es identificar su caso de uso. Por ejemplo, ha programado el manual de procedimientos `AWS-CreateImage` de manera que se ejecute diariamente en todas las instancias de Amazon EC2 de producción. Al final del mes, decide tener más imágenes de las necesarias para los puntos de recuperación. Desde entonces, desea eliminar automáticamente la AMI más antigua de una instancia de Amazon EC2 al crear una AMI nueva. Para ello, crea un nuevo manual de procedimientos que hace lo siguiente:

1. Ejecuta la acción `aws:createImage` y especifica el ID de instancia en la descripción de la imagen.
2. Ejecuta la acción `aws:waitForAwsResourceProperty` para sondear el estado de la imagen hasta que sea `available`.
3. Después de que el estado de la imagen se haya establecido como `available`, la acción `aws:executeScript` ejecuta un script de Python personalizado que recopila los ID de todas las imágenes asociadas a su instancia de Amazon EC2. El script hace esto mediante el filtrado, con el ID de instancia en la descripción de la imagen que especificó en el momento de la creación. A continuación, el script ordena la lista de ID de imagen en función de la `creationDate` de la imagen y selecciona el ID de la AMI más antigua.
4. Por último, la acción `aws:deleteImage` se ejecuta para eliminar la AMI más antigua con el ID de la salida del paso anterior.

En este escenario, ya estaba usando el manual de procedimientos `AWS-CreateImage`, pero descubrió que el caso de uso requería más flexibilidad. Esta es una situación común, ya que puede haber superposición entre los manuales de procedimientos y las acciones de automatización. Como resultado, es posible que tenga que ajustar qué manuales de procedimientos o acciones usa para abordar su caso de uso.

Por ejemplo, las acciones `aws:executeScript` y `aws:invokeLambdaFunction` le permiten ejecutar scripts personalizados como parte de la automatización. A la hora de elegir una, es posible que prefiera `aws:invokeLambdaFunction` debido a los lenguajes adicionales de tiempo de ejecución compatibles. Sin embargo, es posible que prefiera `aws:executeScript` porque le permite crear contenido de script directamente en los manuales de procedimientos YAML y proporcionar contenido de script como archivos adjuntos para los manuales de procedimientos JSON. También podría considerar que `aws:executeScript` es más simple en cuanto a la configuración de AWS Identity and Access Management (IAM). Como utiliza los permisos proporcionados en el `AutomationAssumeRole`, `aws:executeScript` no requiere un rol de ejecución de la función de AWS Lambda adicional.

En cualquier escenario, una acción puede proporcionar más flexibilidad o funcionalidad adicional que la otra. Por lo tanto, recomendamos que revise los parámetros de entrada disponibles para el manual de procedimientos o la acción que desea utilizar para determinar cuál se ajusta mejor a su caso de uso y sus preferencias.

Configure el entorno de desarrollo.

Después de identificar el caso de uso y los manuales de procedimientos predefinidos o las acciones de automatización que desea utilizar en el manual, debe configurar el entorno de desarrollo para el contenido del manual de procedimientos. Para desarrollar el contenido de su manual de procedimientos, recomendamos usar el AWS Toolkit for Visual Studio Code en lugar de la consola de documentos de Systems Manager.

El Toolkit for VS Code es una extensión de código abierto para Visual Studio Code (VS Code) que ofrece más características que la consola de documentos de Systems Manager. Entre las características útiles, se incluyen la validación de esquemas para YAML y JSON, los fragmentos para tipos de acciones de automatización y la compatibilidad con la capacidad de completar de manera automática para distintas opciones en YAML y JSON.

Para obtener más información acerca de la instalación de Toolkit for VS Code, consulte Instalación de AWS Toolkit for Visual Studio Code. Para obtener más información acerca del uso de Toolkit for VS Code para desarrollar manuales de procedimientos, consulte Uso de documentos de Automatización de Systems Manager en la Guía del usuario de AWS Toolkit for Visual Studio Code.

Desarrolle contenido para el manual de procedimientos

Con su caso de uso identificado y su entorno configurado, ya está listo para desarrollar el contenido para su manual de procedimientos. Su caso de uso y sus preferencias determinarán en gran medida

las acciones de automatización o los manuales de procedimientos que utilice en el contenido de su manual. Algunas acciones admiten solo un subconjunto de parámetros de entrada si se comparan con otra acción que le permite llevar a cabo una tarea similar. Otras acciones tienen salidas específicas, como `aws:createImage`, donde algunas acciones le permiten definir sus propias salidas, tales como `aws:executeAwsApi`.

Si no está seguro de cómo se usa una acción concreta en el manual de procedimientos, le recomendamos que revise la entrada correspondiente a la acción en la Referencia de acciones de Automatización de Systems Manager. También recomendamos que revise el contenido de manuales de procedimientos predefinidos para ver ejemplos reales de cómo se utilizan estas acciones. Para obtener más ejemplos de la aplicación real de los manuales de procedimientos, consulte Ejemplos adicionales de manuales de procedimientos.

Con el fin de demostrar las diferencias en materia de simplicidad y flexibilidad que proporciona el contenido del manual de procedimientos, los siguientes tutoriales brindan un ejemplo de cómo aplicar revisiones a grupos de instancias de Amazon EC2 por etapas:

- the section called “Ejemplo 1: creación de manuales de procedimientos principal y secundario”: en este ejemplo, se utilizan dos manuales de procedimientos en una relación de manuales de tipo principal-secundario. El manual de procedimientos principal inicia una automatización de control de frecuencia del manual de procedimientos secundario.
- the section called “Ejemplo 2: manual de procedimientos con scripts”: en este ejemplo, se muestra cómo puede realizar las mismas tareas del ejemplo 1 mediante la condensación del contenido en un único manual de procedimientos y a través del uso de scripts.

Ejemplo 1: creación de manuales de procedimientos principal y secundario

En el siguiente ejemplo, se muestra cómo crear dos manuales de procedimientos que apliquen revisiones a grupos etiquetados de instancias de Amazon Elastic Compute Cloud (Amazon EC2) por etapas. Estos manuales de procedimientos se utilizan en una relación de manuales de tipo principal-secundario, por la cual se usa el manual de procedimientos principal para iniciar una automatización de control de frecuencia del manual de procedimientos secundario. Para obtener más información acerca de las automatizaciones de control de frecuencia, consulte Ejecución de operaciones automatizadas a escala. Para obtener más información acerca de las acciones de automatización que se utilizan en este ejemplo, consulte Referencia de acciones de Automatización de Systems Manager.

Cree el manual de procedimientos secundario

En este manual de procedimientos de ejemplo, se aborda el siguiente escenario. Emily se desempeña como ingeniera en sistemas en AnyCompany Consultants, LLC. Debe configurar la aplicación de revisiones para grupos de instancias de Amazon Elastic Compute Cloud (Amazon EC2) que alojen bases de datos primarias y secundarias. Las aplicaciones acceden a estas bases de datos las 24 horas del día, por lo que una de las instancias de base de datos siempre debe estar disponible.

Ella decide que lo mejor que puede hacer es aplicar revisiones a las instancias por etapas. Primero se aplicará revisiones al grupo principal de instancias de base de datos y, luego, al grupo secundario. Además, para evitar incurrir en costos adicionales por dejar en ejecución instancias que se habían detenido anteriormente, Emily quiere que las instancias a las que se aplicaron revisiones se devuelvan a su estado original antes de que se apliquen las revisiones.

Emily identifica los grupos primarios y secundarios de instancias de base de datos a través de las etiquetas asociadas a las instancias. Ella decide crear un manual de procedimientos principal que inicia una automatización de control de frecuencia de un manual de procedimientos secundario. Al hacerlo, puede indicar como destino las etiquetas asociadas a los grupos primarios y secundarios de las instancias de base de datos y, a la vez, administrar la simultaneidad de las automatizaciones secundarias. Después de revisar los documentos disponibles de Systems Manager (SSM) para aplicar revisiones, elige el documento `AWS-RunPatchBaseline`. Mediante el uso de este documento de SSM, sus colegas pueden revisar la información de conformidad de las revisiones asociados una vez finalizada la operación de aplicación de revisiones.

Para empezar a crear el contenido de su manual de procedimientos, Emily revisa las acciones de automatización disponibles y comienza a crear el contenido del manual de procedimientos secundario de la siguiente manera:

1. En primer lugar, proporciona valores para el esquema y la descripción del manual de procedimientos, y define los parámetros de entrada para el manual de procedimientos secundario.

   A través del parámetro `AutomationAssumeRole`, Emily y sus colegas pueden utilizar un rol de IAM existente que permite a Automation realizar las acciones en el manual de procedimientos en su nombre. Emily utiliza el parámetro `InstanceId` para determinar a qué instancia se le debe aplicar una revisión. De forma opcional, los parámetros `Operation`, `RebootOption` y `SnapshotId` se pueden utilizar para proporcionar valores a los parámetros del documento para `AWS-RunPatchBaseline`. Para evitar que se proporcionen valores no válidos a esos parámetros del documento, define los `allowedValues` como se necesiten.

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. Con los elementos de nivel superior definidos, Emily continúa con la creación de las acciones que componen los `mainSteps` del manual de procedimientos. El primer paso genera el estado actual de la instancia de destino especificada en el parámetro de entrada `InstanceId` mediante la acción `aws:executeAwsApi`. La salida de esta acción se utiliza en acciones posteriores.

   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. En lugar de realizar un inicio manual y un seguimiento del estado original de cada instancia a la que se debe aplicar una revisión, Emily utiliza la salida de la acción anterior para bifurcar la automatización en función del estado de la instancia de destino. Esto permite que la automatización ejecute diferentes pasos dependiendo de las condiciones definidas en la acción `aws:branch` y, además, mejora la eficiencia general de la automatización sin intervención manual.

   Si el estado de la instancia ya es `running`, la automatización continúa con la aplicación de revisiones a la instancia con el documento `AWS-RunPatchBaseline` mediante la acción `aws:runCommand`.

   Si el estado de la instancia es `stopping`, la automatización realiza un sondeo para que la instancia alcance el estado `stopped` con la acción `aws:waitForAwsResourceProperty`, activa la instancia con la acción `executeAwsApi` y realiza un sondeo para que la instancia alcance el estado `running` antes de aplicarle revisiones.

   Si el estado de la instancia es `stopped`, la automatización activa la instancia y realiza un sondeo para que alcance el estado `running` antes de aplicarle revisiones con las mismas acciones.

   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. Una vez finalizada la operación de aplicación de revisiones, Emily quiere que la automatización devuelva la instancia de destino al mismo estado en que estaba antes de que comenzara la automatización. Para hacerlo, usa la salida de la primera acción otra vez. La automatización se bifurca en función del estado original de la instancia de destino con la acción `aws:branch`. Si, anteriormente, la instancia estaba en cualquier estado distinto de `running`, se detiene. De lo contrario, si el estado de la instancia es `running`, se termina la automatización.

   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 revisa el contenido del manual de procedimientos secundario completado y crea el manual en la misma Cuenta de AWS y Región de AWS que las instancias de destino. Ya está lista para seguir con la creación del contenido del manual de procedimientos principal. A continuación, se muestra el contenido del manual de procedimientos secundario completado.

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}}"
            ]
         }
      }
   ]
}
```

Para obtener más información acerca de las acciones de automatización que se utilizan en este ejemplo, consulte Referencia de acciones de Automatización de Systems Manager.

Crear el manual de procedimientos principal

En este manual de procedimientos de ejemplo, se continúa con el escenario descrito en la sección anterior. Ahora que Emily ha creado el manual de procedimientos secundario, comienza a generar el contenido del manual de procedimientos principal de la siguiente manera:

1. En primer lugar, proporciona valores para el esquema y la descripción del manual de procedimientos, y define los parámetros de entrada para el manual de procedimientos principal.

A través del parámetro `AutomationAssumeRole`, Emily y sus colegas pueden utilizar un rol de IAM existente que permite a Automation realizar las acciones en el manual de procedimientos en su nombre. Emily utiliza los parámetros `PatchGroupPrimaryKey` y `PatchGroupPrimaryValue` para especificar la etiqueta asociada al grupo principal de instancias de base de datos a las cuales se aplicarán revisiones. Utiliza los parámetros `PatchGroupSecondaryKey` y `PatchGroupSecondaryValue` para especificar la etiqueta asociada al grupo secundario de instancias de base de datos a las cuales se aplicarán revisiones.

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. Con los elementos de nivel superior definidos, Emily continúa con la creación de las acciones que componen los `mainSteps` del manual de procedimientos.

   La primera acción inicia una automatización de control de frecuencia con el manual de procedimientos secundario que acaba de crear y que indica como destino instancias asociadas a la etiqueta que se especificó en los parámetros de entrada `PatchGroupPrimaryKey` y `PatchGroupPrimaryValue`. Utiliza los valores proporcionados para los parámetros de entrada

con el fin de especificar la clave y el valor de la etiqueta asociada al grupo principal de instancias de base de datos a las que desea aplicar revisiones.

Una vez finalizada la primera automatización, la segunda acción inicia otra automatización de control de frecuencia con el manual de procedimientos secundario que indica como destino instancias asociadas a la etiqueta que se especificó en los parámetros de entrada `PatchGroupSecondaryKey` y `PatchGroupSecondaryValue`. Utiliza los valores proporcionados para los parámetros de entrada con el fin de especificar la clave y el valor de la etiqueta asociada al grupo secundario de instancias de base de datos a las que desea aplicar revisiones.

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 revisa el contenido del manual de procedimientos principal completado y crea el manual en la misma Cuenta de AWS y Región de AWS que las instancias de destino. Ahora, está lista para probar sus manuales de procedimientos y así asegurarse de que la automatización funciona como desea antes de implementarlos en su entorno de producción. A continuación, se muestra el contenido del manual de procedimientos principal completado.

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"
         }
      }
   ]
}
```

Para obtener más información acerca de las acciones de automatización que se utilizan en este ejemplo, consulte Referencia de acciones de Automatización de Systems Manager.

Ejemplo 2: manual de procedimientos con scripts

En este manual de procedimientos de ejemplo, se aborda el siguiente escenario. Emily se desempeña como ingeniera en sistemas en AnyCompany Consultants, LLC. Anteriormente, creó dos manuales de procedimientos que se utilizan en una relación de manuales de tipo principal-secundario a fin de aplicar parches a grupos de instancias de Amazon Elastic Compute Cloud (Amazon EC2) que alojan bases de datos primarias y secundarias. Las aplicaciones acceden a estas bases de datos las 24 horas del día, por lo que una de las instancias de base de datos siempre debe estar disponible.

Teniendo en cuenta este requisito, creó una solución que aplica parches a las instancias en etapas mediante el documento `AWS-RunPatchBaseline` de Systems Manager (SSM). Mediante el uso de este documento de SSM, sus colegas pueden revisar la información de conformidad de los parches asociados una vez finalizada la operación de aplicación de parches.

Primero se aplican parches al grupo principal de instancias de base de datos y, luego, al grupo secundario. Además, para evitar incurrir en costos adicionales por dejar en ejecución instancias que se habían detenido anteriormente, Emily se aseguró de que la automatización devolviera las instancias a las que se aplicaron parches a su estado original antes de que se aplicaran los parches. Emily utilizó etiquetas asociadas a los grupos primarios y secundarios de instancias de base de datos para identificar a qué instancias se deben aplicar parches en el orden deseado.

Su solución automatizada existente funciona, pero, de ser posible, quiere mejorarla. Para ayudar con el mantenimiento del contenido del manual de procedimientos y facilitar la solución de problemas, le gustaría condensar la automatización en un solo manual de procedimientos y simplificar el número de parámetros de entrada. Además, le gustaría evitar la creación de múltiples automatizaciones secundarias.

Después de que Emily revisa las acciones de automatización disponibles, determina que puede mejorar su solución mediante la acción `aws:executeScript` para ejecutar sus scripts de Python personalizados. Ahora, comienza a crear el contenido del manual de procedimientos de la siguiente manera:

1. En primer lugar, proporciona valores para el esquema y la descripción del manual de procedimientos, y define los parámetros de entrada para el manual de procedimientos principal.

A través del parámetro `AutomationAssumeRole`, Emily y sus colegas pueden utilizar un rol de IAM existente que permite a Automation realizar las acciones en el manual de procedimientos en su nombre. A diferencia del ejemplo 1, ahora el parámetro `AutomationAssumeRole` es obligatorio en lugar de opcional. Debido a que este manual de procedimientos incluye acciones `aws:executeScript`, siempre se requiere un rol de servicio de AWS Identity and Access Management (IAM) (o rol de asunción). Este requisito es necesario, ya que algunos de los scripts de Python especificados para las acciones llaman las operaciones de la API de AWS.

Emily utiliza los parámetros `PrimaryPatchGroupTag` y `SecondaryPatchGroupTag` para especificar las etiquetas asociadas al grupo principal y secundario de instancias de base de datos a las cuales se aplicarán parches. Para simplificar los parámetros de entrada necesarios, decide utilizar los parámetros `StringMap` en lugar de varios parámetros `String`, como hizo con el manual de procedimientos del ejemplo 1. De forma opcional, los parámetros `Operation`, `RebootOption` y `SnapshotId` se pueden utilizar para proporcionar valores a los parámetros del documento para `AWS-RunPatchBaseline`. Para evitar que se proporcionen valores no válidos a esos parámetros del documento, define los `allowedValues` como se necesiten.

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. Con los elementos de nivel superior definidos, Emily continúa con la creación de las acciones que componen los `mainSteps` del manual de procedimientos. El primer paso recopila los ID de todas las instancias asociadas a la etiqueta especificada en el parámetro `PrimaryPatchGroupTag` y genera un parámetro `StringMap` que contiene el ID de instancia y su estado actual. La salida de esta acción se utiliza en acciones posteriores.

Tenga en cuenta que el parámetro de entrada `script` no es compatible con los manuales de procedimientos JSON. Los manuales de procedimientos JSON deben proporcionar contenido de script a través del parámetro de entrada `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 utiliza la salida de la acción anterior en otra acción `aws:executeScript` para verificar que todas las instancias asociadas a la etiqueta especificada en el parámetro `PrimaryPatchGroupTag` se encuentren en el estado `running`.

   Si el estado de la instancia ya es `running` o `shutting-down`, el script sigue recorriendo las instancias restantes.

   Si el estado de la instancia es `stopping`, el script realiza un sondeo para que la instancia llegue al estado `stopped` y activa la instancia.

Si el estado de la instancia es `stopped`, el script activa la instancia.

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 verifica que todas las instancias asociadas a la etiqueta especificada en el parámetro `PrimaryPatchGroupTag` ya estén en funcionamiento o se encuentren en el estado `running`. A continuación, usa otro script para verificar que todas las instancias, incluidas las que se activaron con la acción anterior, hayan alcanzado el estado `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 usa dos scripts más para devolver los valores `String` individuales de la clave y el valor de la etiqueta especificada en el parámetro `PrimaryPatchGroupTag`. Los valores que devuelven estas acciones le permiten proporcionar valores al parámetro `Targets` para el documento `AWS-RunPatchBaseline` de manera directa. A continuación, la automatización continúa con la aplicación de parches a la instancia con el documento `AWS-RunPatchBaseline` a través de la acción `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. Una vez finalizada la operación de aplicación de parches, Emily quiere que la automatización devuelva las instancias de destino asociadas a la etiqueta que se especificó en el parámetro `PrimaryPatchGroupTag` al mismo estado en que estaban antes de que se iniciara la automatización. Para hacerlo, usa otra vez la salida de la primera acción en un script. En función del estado original de la instancia de destino, si la instancia estaba anteriormente en un estado distinto de `running`, se detiene. En cambio, si el estado de la instancia es `running`, el script sigue recorriendo las instancias 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. La operación de aplicación de parches se completa para las instancias asociadas a la etiqueta especificada en el parámetro `PrimaryPatchGroupTag`. Ahora, Emily duplica todas las acciones anteriores en el contenido de su manual de procedimientos para indicar como destino las instancias asociadas a la etiqueta que se especificó en el parámetro `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 revisa el contenido del manual de procedimientos en script completado y crea el manual en la misma Cuenta de AWS y Región de AWS que las instancias de destino. Ahora, está lista para probar su manual de procedimientos para asegurarse de que la automatización funciona como desea antes de implementarlo en su entorno de producción. A continuación, se muestra el contenido del manual de procedimientos completado con scripts.

   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":"..."
         }
      }
   ]
}
```

Para obtener más información acerca de las acciones de automatización que se utilizan en este ejemplo, consulte Referencia de acciones de Automatización de Systems Manager.

Ejemplos adicionales de manuales de procedimientos

El siguiente manual de procedimientos de ejemplo demuestra cómo puede usar las acciones de automatización de AWS Systems Manager para automatizar tareas comunes de implementación, solución de problemas y mantenimiento.

> **Note**
>
> Los manuales de procedimientos de ejemplo de esta sección se proporcionan para demostrar cómo puede crear manuales de procedimientos personalizados para satisfacer sus necesidades operativas específicas. Estos manuales de procedimientos no están diseñados para su uso en entornos de producción tal como están. Sin embargo, puede personalizarlos para su propio uso.

Ejemplos

- Implementación de la arquitectura de VPC y de controladores de dominio de Microsoft Active Directory
- Restauración de un volumen raíz a partir de la última instantánea

- Creación de una AMI y de una copia entre regiones

Implementación de la arquitectura de VPC y de controladores de dominio de Microsoft Active Directory

Para aumentar la eficiencia y estandarizar las tareas comunes, puede elegir automatizar las implementaciones. Esto resulta útil si suele implementar la misma arquitectura en varias cuentas y Regiones de AWS de manera regular. La automatización de las implementaciones de arquitectura también puede reducir la posibilidad de que se produzcan los errores humanos inherentes a los procesos manuales. AWS Systems Manager Las acciones de Automation pueden ayudarlo a lograrlo. Automatización es una herramienta de AWS Systems Manager.

El siguiente manual de procedimientos de AWS Systems Manager de ejemplo realiza estas acciones:

- Recupera la última Amazon Machine Image (AMI) de Windows Server 2016 con Parameter Store de Systems Manager para usarla al momento de lanzar las instancias de EC2 que se configurarán como controladores de dominio. Parameter Store es una herramienta de AWS Systems Manager.
- Utiliza la acción de automatización `aws:executeAwsApi` a fin de llamar varias operaciones de la API de AWS para crear la arquitectura de VPC. Las instancias del controlador de dominio se inician en subredes privadas y se conectan a Internet mediante una puerta de enlace NAT. Esto permite que el SSM Agent en las instancias acceda a los puntos de enlace necesarios de Systems Manager.
- Utiliza la acción de automatización `aws:waitForAwsResourceProperty` a fin de confirmar que las instancias lanzadas por la acción anterior estén `Online` para AWS Systems Manager.
- Utiliza la acción de automatización `aws:runCommand` para configurar las instancias lanzadas como controladores de dominio de 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..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..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

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 $_",
                "}"
              ]
            }
          }
        }
      ]
    }
```

Restauración de un volumen raíz a partir de la última instantánea

El sistema operativo en un volumen raíz puede dañarse por varias razones. Por ejemplo, después de una operación de revisióno, las instancias podrían arrancar de manera incorrecta debido a un kernel o un registro dañado. La automatización de tareas comunes de solución de problemas, como la restauración de un volumen raíz a partir de la última instantánea tomada antes de la operación de aplicación de revisiones, puede reducir el tiempo de inactividad y agilizar los esfuerzos de solución de problemas. AWS Systems Manager Las acciones de Automation pueden ayudarlo a lograrlo. Automatización es una herramienta de AWS Systems Manager.

El siguiente manual de procedimientos de AWS Systems Manager de ejemplo realiza estas acciones:

- Utiliza la acción de automatización `aws:executeAwsApi` para recuperar detalles del volumen raíz de la instancia.
- Utiliza la acción de automatización `aws:executeScript` para recuperar la última instantánea del volumen raíz.
- Utiliza la acción de automatización `aws:branch` para continuar la automatización si se encuentra una instantánea para el volumen raíz.

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"
                }
            }
        }
    }
```

Creación de una AMI y de una copia entre regiones

La creación de una Amazon Machine Image (AMI) a partir de una instancia es un proceso que se utiliza para realizar copias de seguridad y tareas de recuperación. En una arquitectura de recuperación de desastres, también puede copiar una AMI en otra Región de AWS, si así lo desea. La automatización de tareas de mantenimiento habituales puede reducir el tiempo de inactividad si un problema requiere la conmutación por error. AWS Systems Manager Las acciones de Automation pueden ayudarlo a lograrlo. Automatización es una herramienta de AWS Systems Manager.

El siguiente manual de procedimientos de AWS Systems Manager de ejemplo realiza estas acciones:

- Utiliza la acción de automatización `aws:executeAwsApi` para crear una AMI.
- Utiliza la acción de automatización `aws:waitForAwsResourceProperty` para confirmar la disponibilidad de la AMI.
- Utiliza la acción de automatización `aws:executeScript` para copiar la AMI en la región de destino.

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"
                }
            }
        }
    }
```

Creación de parámetros de entrada que rellenan recursos de AWS

Automatización, una herramienta de Systems Manager, rellena los recursos de AWS en la AWS Management Console que concuerdan con el tipo de recurso que el usuario defina para un parámetro de entrada. Los recursos en su Cuenta de AWS que concuerden con el tipo de recurso se muestran en una lista desplegable para que elija. Puede definir tipos de parámetros de entrada para instancias de Amazon Elastic Compute Cloud (Amazon EC2), buckets de Amazon Simple Storage Service (Amazon S3) y roles de AWS Identity and Access Management (IAM). Las definiciones de tipo admitidas y las expresiones regulares que se utilizan para localizar los recursos coincidentes son las siguientes:

- `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/.*$`
- `List<AWS::IAM::Role::Arn>` - `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):iam::[0-9]{12}:role/.*$`

A continuación, se muestra un ejemplo de los tipos de parámetros de entrada definidos en el contenido del 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
      }
   ]
}
```

Uso del Generador de documentos para crear un manual de procedimientos

Si los manuales de procedimientos públicos de AWS Systems Manager no admiten todas las acciones que desea realizar en sus recursos de AWS, puede crear sus propios manuales. Para crear un manual de procedimientos personalizado, puede crear manualmente un archivo de formato YAML o JSON local con las acciones de automatización adecuadas. Como alternativa, puede usar el Generador de documentos en la consola de Automatización de Systems Manager para crear un manual de procedimientos personalizado.

Con el Generador de documentos, puede agregar acciones de automatización a su manual de procedimientos personalizado y proporcionar los parámetros necesarios sin tener que usar la sintaxis

JSON o YAML. Después de agregar pasos y crear el manual de procedimientos, el sistema convierte las acciones que ha agregado al formato YAML que Systems Manager puede utilizar para ejecutar la automatización.

Los manuales de procedimientos admiten el uso de Markdown, un lenguaje de marcado que le permite agregar descripciones de estilo Wiki a manuales de procedimientos y pasos individuales dentro del manual. Para obtener más información acerca del uso de Markdown, consulte Uso de Markdown en AWS.

Crear un manual de procedimientos con el Generador de documentos

Antes de empezar

Le recomendamos que lea acerca de las diferentes acciones que puede usar dentro de un manual de procedimientos. Para obtener más información, consulte Referencia de acciones de Automatización de Systems Manager.

Para crear un manual de procedimientos con el Generador de documentos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. Elija Create automation (Crear automatización).
4. En Name (Nombre), ingrese un nombre descriptivo para el manual de procedimientos.
5. En Document description (Descripción del documento), proporcione la descripción del estilo Markdown del manual de procedimientos. Puede proporcionar instrucciones para utilizar el manual de procedimientos, pasos numerados o cualquier otro tipo de información para describir el manual de procedimientos. Consulte el texto predeterminado para obtener información sobre cómo dar formato al contenido.

   > **Tip**
   >
   > Alterna entre Hide preview (Ocultar vista previa) y Show preview (Mostrar vista previa) para ver el aspecto del contenido de la descripción mientras redacta.

6. De forma opcional, en Assume role (Asumir rol), escriba el nombre o ARN de un rol de servicio que realizará acciones en su nombre. Si no especifica un rol, Automation utiliza los permisos de acceso del usuario que ejecuta la automatización.

> **Important**
>
> Para los manuales de procedimientos que no son propiedad de Amazon y que utilizan la acción `aws:executeScript`, es necesario especificar un rol. Para obtener más información, consulte Permisos para utilizar los manuales de procedimientos.

7. (Opcional) En Outputs (Salidas), ingrese cualquier salida para la automatización de este manual de procedimientos a fin de que esté disponible para otros procesos.

   Por ejemplo, si el manual de procedimientos crea una AMI nueva, puede especificar ["CreateImage.ImageId"] y, a continuación, utilizar esta salida para crear instancias nuevas en una automatización posterior.

8. De forma opcional, expanda la sección Input parameters (Parámetros de entrada) y haga lo siguiente.

   1. En Parameter name (Nombre de parámetro), ingrese un nombre descriptivo para el parámetro del manual de procedimientos que está creando.
   2. En Type (Tipo), elija un tipo para el parámetro, como `String` o `MapList`.
   3. En Required (Requerido), realice una de las acciones siguientes:
      - Elija Yes (Sí) si se debe proporcionar un valor para este parámetro del manual de procedimientos en el tiempo de ejecución.
      - Elija No si el parámetro no es necesario y, de forma opcional, ingrese el valor de un parámetro predeterminado en Default value (Valor predeterminado).
   4. En Description (Descripción), ingrese una descripción para el parámetro del manual de procedimientos.

   > **Note**
   >
   > Para agregar más parámetros del manual de procedimientos, elija Add a parameter (Agregar un parámetro). Para quitar un parámetro del manual de procedimientos, elija el botón X (quitar).

9. (Opcional) Expanda la sección Target type (Tipo de destino) y elija un tipo de destino para definir los tipos de recursos en los que se puede ejecutar la automatización. Por ejemplo, para usar un manual de procedimientos en instancias EC2, elija `/AWS::EC2::Instance`.

> **Note**
>
> Si especifica un valor de “/”, el manual de procedimientos puede ejecutarse en todos los tipos de recursos. Para obtener una lista de los tipos de recursos válidos, consulte la Referencia de tipos de recursos de AWS en la Guía del usuario AWS CloudFormation.

10. (Opcional) Expanda la sección Document tags (Etiquetas del documento) e ingrese uno o más pares de clave-valor de etiqueta para aplicarlos al manual de procedimientos. Las etiquetas facilitan la identificación, la organización y la búsqueda de recursos.
11. En la sección Step 1 (Paso 1) proporcione la siguiente información.

    - En Step name (Nombre del paso), escriba un nombre descriptivo para el primer paso de la automatización.
    - En Action type (Tipo de acción), seleccione el tipo de acción que desea utilizar para este paso.

      Para obtener una lista e información acerca de los tipos de acción disponibles, consulte Referencia de acciones de Automatización de Systems Manager.
    - En Description (Descripción), escriba una descripción del paso de automatización. Puede usar Markdown para dar formato al texto.
    - Según el Action type (Tipo de acción) seleccionado, introduzca las entradas necesarias para el tipo de acción en la sección Step inputs (Entradas de paso). Por ejemplo, si ha seleccionado la acción `aws:approve`, debe especificar un valor para la propiedad `Approvers`.

      Para obtener información acerca de los campos de entrada de pasos, consulte la entrada de Referencia de acciones de Automatización de Systems Manager correspondiente al tipo de acción seleccionado. Por ejemplo: `aws:executeStateMachine`: ejecutar una máquina de estado de AWS Step Functions.
    - (Opcional) En Additional inputs (Entradas adicionales), proporcione los valores de entrada adicionales que se necesitan para el manual de procedimientos. Los tipos de entrada disponibles dependen del tipo de acción seleccionado para el paso. (Tenga en cuenta que algunos tipos de acción requieren valores de entrada).

> **Note**
>
> Para agregar más entradas, elija Add optional input (Agregar entrada opcional). Para eliminar una entrada, elija el botón X (eliminar).

- (Opcional) En Outputs (Salidas), ingrese cualquier salida que corresponda a este paso a fin de que esté disponible para otros procesos.

> **Note**
>
> Las Outputs (Salidas) no están disponibles para todos los tipos de acción.

- (Opcional) Expanda la sección Common properties (Propiedades comunes) y especifique propiedades para las acciones que son comunes a todas las acciones de automatización. Por ejemplo, para Timeout seconds (Segundos de tiempo de espera), puede proporcionar un valor en segundos para especificar cuánto tiempo se puede ejecutar el paso antes de que se detenga.

  Para obtener más información, consulte Propiedades compartidas por todas las acciones.

> **Note**
>
> Para agregar más pasos, seleccione Add step (Agregar paso) y repita el procedimiento para crear un paso. Para quitar un paso, elija Remove step (Quitar paso).

12. Elija Create automation (Crear automatización) para guardar el manual de procedimientos.

Crear un manual de procedimientos que ejecute scripts

El siguiente procedimiento muestra cómo usar el Generador de documentos en la consola de Automation de AWS Systems Manager para crear un manual de procedimientos personalizado que ejecuta un script.

El primer paso del manual de procedimientos que crea ejecuta un script para lanzar una instancia de Amazon Elastic Compute Cloud (Amazon EC2). El segundo paso ejecuta otro script para monitorear que el estado de la instancia pase a ser ok. A continuación, se informa el estado general `Success` para la automatización.

Antes de empezar

Asegúrese de haber completado los pasos siguientes:

- Verifique que tiene privilegios de administrador o que se le han concedido los permisos adecuados para acceder a Systems Manager en AWS Identity and Access Management (IAM).

  Para obtener más información, consulte Comprobación del acceso del usuario para manuales de procedimientos.
- Compruebe que tiene un rol de servicio de IAM para Automation (también conocido como rol de asunción) en su Cuenta de AWS. El rol es necesario porque esta explicación utiliza la acción `aws:executeScript`.

  Para obtener información acerca de la creación de esta función, consulte Configuración del acceso de un rol de servicio (rol de asunción) para automatizaciones.

  Para obtener información acerca del requisito de rol de servicio de IAM para ejecutar `aws:executeScript`, consulte Permisos para utilizar los manuales de procedimientos.
- Compruebe que tiene permiso para lanzar instancias EC2.

  Para obtener información, consulte IAM y Amazon EC2 en la Guía del usuario de Amazon EC2.

Para crear un manual de procedimientos personalizado que ejecute scripts mediante el Generador de documentos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. Elija Create automation (Crear automatización).
4. En Name (Nombre), escriba este nombre descriptivo para el manual de procedimientos: **`LaunchInstanceAndCheckStatus`**.
5. (Opcional) En Document description (Descripción del documento), reemplace el texto predeterminado con una descripción de este manual de procedimientos utilizando Markdown. A continuación, se muestra un ejemplo.

```
##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. En Assume role (Rol de asunción), ingrese el ARN del rol de servicio de IAM para Automation (Rol de asunción) para la automatización, en el formato **`arn:aws:iam::111122223333:role/AutomationServiceRole`**. Sustituya el ID de la Cuenta de AWS por 111122223333.

   El rol que especifique se utiliza para proporcionar los permisos necesarios para iniciar la automatización.

   > **Important**
   >
   > Para los manuales de procedimientos que no son propiedad de Amazon y que utilizan la acción `aws:executeScript`, es necesario especificar un rol. Para obtener más información, consulte Permisos para utilizar los manuales de procedimientos.

7. Expanda Input parameters (Parámetros de entrada) y haga lo siguiente.

   1. En Parameter name (Nombre de parámetro), introduzca **`imageId`**.
   2. En Type (Tipo), elija **`String`**.
   3. En Required (Requerido), elija No.
   4. En Default value (Valor predeterminado), escriba lo siguiente.

      ```
      {{ ssm:/aws/service/ami-amazon-linux-latest/amzn-ami-hvm-x86_64-gp2 }}
      ```

> **Note**
>
> Este valor lanza una instancia de Amazon EC2 con el ID de la última Amazon Machine Image (AMI) de Amazon Linux 1. Si desea utilizar una AMI diferente, reemplace el valor con el ID de su AMI.

5. En Description (Descripción), escriba lo siguiente.

```
(Optional) The AMI ID to use for launching the instance. The default value uses
 the latest released Amazon Linux AMI ID.
```

8. Seleccione Add a parameter (Añadir un parámetro) para crear el segundo parámetro, **`tagValue`**, y escriba lo siguiente.

   1. En Parameter name (Nombre de parámetro), introduzca **`tagValue`**.
   2. En Type (Tipo), elija **`String`**.
   3. En Required (Requerido), elija No.
   4. En Default value (Valor predeterminado), introduzca **`LaunchedBySsmAutomation`**. Esto agrega el valor de par de claves de etiqueta `Name:LaunchedBySsmAutomation` a la instancia.
   5. En Description (Descripción), escriba lo siguiente.

```
(Optional) The tag value to add to the instance. The default value is
 LaunchedBySsmAutomation.
```

9. Seleccione Add a parameter (Añadir un parámetro) para crear el tercer parámetro, **`instanceType`**, y escriba lo siguiente.

   1. En Parameter name (Nombre de parámetro), introduzca **`instanceType`**.
   2. En Type (Tipo), elija **`String`**.
   3. En Required (Requerido), elija No.
   4. En Default value (Valor predeterminado), introduzca **`t2.micro`**.
   5. En Parameter description (Descripción del parámetro), escriba lo siguiente.

```
(Optional) The instance type to use for the instance. The default value is
 t2.micro.
```

10. Expanda Target type (Tipo de destino) y elija **"/"**.
11. (Opcional) Expanda Document tags (Etiquetas de documento) para aplicar etiquetas de recursos al manual de procedimientos. En Tag key (Clave de etiqueta), escriba **Purpose** y en Tag value (Valor de etiqueta), escriba **LaunchInstanceAndCheckState**.
12. En la sección Step 1 (Paso 1) siga los pasos siguientes.

    1. En Step name (Nombre del paso), ingrese este nombre descriptivo para el primer paso de la automatización: **LaunchEc2Instance**.
    2. En Action type (Tipo de acción), seleccione Run a script) (**aws:executeScript**).
    3. En Description (Descripción), escriba una descripción para el paso de automatización, como la siguiente.

    ```
    **About This Step**

       This step first launches an EC2 instance using the ```aws:executeScript```
     action and the provided script.
    ```

    4. Amplíe Inputs (Entradas).
    5. En Runtime (Entorno de ejecución), elija el lenguaje del entorno de ejecución que se va a usar para ejecutar el script proporcionado.
    6. En Handler (Controlador), escriba **launch_instance**. Este es el nombre de la función declarado en el siguiente script.

    > **Note**
    >
    > Esto no es necesario para PowerShell.

    7. En Script, reemplace el contenido predeterminado por lo siguiente. Asegúrese de que el script coincide con el valor del entorno de ejecución.

    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. Amplíe Additional inputs (Entradas adicionales).
9. En Input name (Nombre de entrada), elija InputPayload. En Input Value (Valor de entrada), introduzca los siguientes datos YAML.

```
image_id: "{{ imageId }}"
    tag_value: "{{ tagValue }}"
    instance_type: "{{ instanceType }}"
```

13. Expanda Outputs (Salidas) y realice lo siguiente:

- En Nombre, escriba **`payload`**.
- Para Selector, escriba **`$.Payload`**.
- En Type (Tipo), elija `StringMap`.

14. Seleccione Add step (Agregar paso) para agregar un segundo paso al manual de procedimientos. El segundo paso consulta el estado de la instancia iniciada en el paso 1 y espera hasta que el estado devuelto sea `ok`.
15. En la sección Step 2 (Paso 2) haga lo siguiente.

1. En Step name (Nombre del paso), ingrese este nombre descriptivo para el segundo paso de la automatización: **`WaitForInstanceStatusOk`**.
2. En Action type (Tipo de acción), seleccione Run a script) (**`aws:executeScript`**).
3. En Description (Descripción), escriba una descripción para el paso de automatización, como la siguiente.

```
**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. En Runtime (Entorno de ejecución), elija el lenguaje del entorno de ejecución que se va a usar para ejecutar el script proporcionado.
5. En Handler (Controlador), escriba **`poll_instance`**. Este es el nombre de la función declarado en el siguiente script.

> **Note**
>
> Esto no es necesario para PowerShell.

6. En Script, reemplace el contenido predeterminado por lo siguiente. Asegúrese de que el script coincide con el valor del entorno de ejecución.

   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. Amplíe Additional inputs (Entradas adicionales).
8. En Input name (Nombre de entrada), elija InputPayload. En Input value (Valor de entrada), introduzca lo siguiente:

```
{{ LaunchEc2Instance.payload }}
```

16. Elija Create automation (Crear automatización) para guardar el manual de procedimientos.

Uso de scripts en manuales de procedimientos

Los manuales de procedimientos de Automation admiten la ejecución de scripts como parte de la automatización. Automatización es una herramienta de AWS Systems Manager. Mediante los manuales de procedimientos, puede ejecutar scripts directamente en AWS sin crear un entorno informático independiente para ejecutar los scripts. Dado que los manuales de procedimientos pueden ejecutar pasos de script junto con otros tipos de pasos de automatización, como las aprobaciones, usted puede intervenir manualmente en situaciones críticas o ambiguas. Puede enviar la salida desde las acciones `aws:executeScript` en sus manuales de procedimientos a Amazon CloudWatch Logs. Para obtener más información, consulte Registro de salida de acción de Automation con CloudWatch Logs.

Permisos para utilizar los manuales de procedimientos

Para utilizar un manual de procedimientos, Systems Manager debe utilizar los permisos de un rol de AWS Identity and Access Management (IAM). El método que Automation utiliza para

determinar qué permisos de rol utilizar depende de algunos factores y de si un paso utiliza la acción `aws:executeScript`.

Para los manuales de procedimientos que no utilizan `aws:executeScript`, Automation emplea una de dos fuentes de permisos:

- Los permisos de un rol de servicio de IAM, o rol de asunción, que se especifica en el manual de procedimientos o se transfiere como parámetro.
- Si no se especifica ningún rol de servicio de IAM, los permisos del usuario que inició la automatización.

Sin embargo, cuando un paso de un manual de procedimientos incluye la acción `aws:executeScript`, siempre se requiere un rol de servicio (rol de asunción) de IAM si el script de Python o PowerShell especificado para la acción llama cualquier operación de la API de AWS. Automation comprueba este rol en el siguiente orden:

- Los permisos de un rol de servicio de IAM, o rol de asunción, que se especifica en el manual de procedimientos o se transfiere como parámetro.
- Si no se encuentra ningún rol, Automation intenta ejecutar el script de Python o PowerShell especificado para `aws:executeScript` sin ningún permiso. Si el script llama una operación de la API de AWS (por ejemplo, la operación `CreateImage` de Amazon EC2) o intenta actuar sobre un recurso de AWS (como una instancia de EC2), el paso que contiene el script produce un error y Systems Manager devuelve un mensaje al respecto.

Incorporación de scripts a los manuales de procedimientos

Puede agregar scripts a los manuales de procedimientos si los inserta en un paso del manual de procedimientos. También puede adjuntar scripts al manual de procedimientos al cargarlos desde el equipo local o mediante la especificación del bucket de Amazon Simple Storage Service (Amazon S3) donde se encuentren los scripts. Una vez que se complete un paso que ejecuta un script, la salida del script estará disponible como un objeto JSON, el cual luego puede utilizar como entrada para los pasos posteriores del manual de procedimientos. Para obtener más información sobre la acción de `aws:executeScript` y cómo utilizar los archivos adjuntos para los scripts, consulte `aws:executeScript`: ejecutar un script.

Restricciones de script para los manuales de procedimientos

Los manuales de procedimientos imponen un límite de cinco archivos adjuntos. Los scripts pueden adoptar la forma de scripts de Python (.py) o scripts de PowerShell Core (.ps1), o adjuntarse como contenido dentro de un archivo .zip.

Uso de instrucciones condicionales en manuales de procedimientos

De forma predeterminada, los pasos que defina en la sección `mainSteps` de un manual de procedimientos se ejecutan en orden secuencial. Cuando finaliza una acción, comienza la siguiente acción especificada en la sección `mainSteps`. Asimismo, si una acción no se ejecuta, falla toda la automatización (de forma predeterminada). Puede utilizar la acción de automatización `aws:branch` y las opciones de manual de procedimientos que se describen en esta sección para crear automatizaciones que aplican la bifurcación condicional. Esto significa que puede crear automatizaciones que salten a otro paso después de evaluar diferentes elecciones o de responder de forma dinámica a los cambios cuando se completa un paso. A continuación, se muestra una lista de las opciones que puede utilizar para crear automatizaciones dinámicas:

- **`aws:branch`**: esta acción de automatización le permite crear una automatización dinámica que evalúa varias elecciones en un solo paso y, a continuación, salta a otro paso en el manual de procedimientos en función de los resultados de dicha evaluación.
- **`nextStep`**: esta opción especifica qué paso de una automatización se debe procesar después de finalizar correctamente un paso.
- **`isEnd`**: esta opción detiene una automatización al final de un paso determinado. El valor predeterminado para esta opción es false.
- **`isCritical`**: esta opción designa un paso como crítico para la finalización correcta de la automatización. Si un paso con esta designación genera un error, Automation informa que el estado final de la automatización es `Failed`. El valor predeterminado para esta opción es `true`.
- **`onFailure`**: esta opción indica si la automatización debe detenerse, continuar, o bien, pasar a otro paso en caso de error. El valor predeterminado para esta opción es abort (anular).

En la siguiente sección, se describe la acción de automatización `aws:branch`. Para obtener más información acerca de las opciones `nextStep`, `isEnd`, `isCritical` y `onFailure`, consulte Manuales de procedimientos `aws:branch` de ejemplo.

Uso de la acción **aws:branch**

La acción `aws:branch` ofrece las opciones de bifurcación condicional más dinámicas para las automatizaciones. Como se ha indicado anteriormente, esta acción permite que la automatización evalúe varias condiciones en un solo paso y, a continuación, salte a un nuevo paso en función de los resultados de dicha evaluación. La acción `aws:branch` funciona como una instrucción `IF-ELIF-ELSE` en programación.

A continuación, se muestra un ejemplo YAML de un paso `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
```

Cuando se especifica la acción `aws:branch` para un paso, se especifican `Choices` que la automatización debe evaluar. La automatización puede evaluar `Choices` según el valor de un parámetro que especificó en la sección `Parameters` del manual de procedimientos. La automatización también puede evaluar `Choices` en función de la salida de un paso anterior.

La automatización evalúa cada elección mediante una expresión booleana. Si la evaluación determina que la primera elección es `true`, la automatización saltará al paso designado para esa elección. Si la evaluación determina que la primera elección es `false`, entonces la automatización evaluará la siguiente elección. Si el paso incluye tres o más `Choices`, la automatización evalúa cada elección en orden secuencial hasta que evalúa una que sea `true`. A continuación, la automatización saltará al paso designado para la elección `true`.

Si ninguna de las `Choices` es `true`, la automatización comprueba si el paso contiene un valor `Default`. Un valor `Default` define un paso al cual la automatización debe saltar si ninguna de las elecciones es `true`. Si no se especifica un valor `Default` para el paso, la automatización procesará el siguiente paso en el manual de procedimientos.

A continuación, se muestra un paso `aws:branch` en YAML llamado chooseOSfromParameter. Este paso incluye dos `Choices`: (`NextStep: runWindowsCommand`) y (`NextStep: runLinuxCommand`). La automatización evalúa estas `Choices` a fin de determinar qué comando ejecutar para el sistema operativo adecuado. La `Variable` para cada elección utiliza `{{OSName}}`, que es un parámetro que el autor del manual de procedimientos definió en la sección `Parameters` del manual.

```
mainSteps:
- name: chooseOSfromParameter
  action: aws:branch
  inputs:
    Choices:
    - NextStep: runWindowsCommand
      Variable: "{{OSName}}"
      StringEquals: Windows
    - NextStep: runLinuxCommand
      Variable: "{{OSName}}"
      StringEquals: Linux
```

A continuación, se muestra un paso `aws:branch` en YAML llamado chooseOSfromOutput. Este paso incluye dos `Choices`: (`NextStep: runPowerShellCommand`) y (`NextStep: runShellCommand`). La automatización evalúa estas `Choices` a fin de determinar qué comando ejecutar para el sistema operativo adecuado. La `Variable` para cada elección utiliza `{{GetInstance.platform}}`, que es la salida de un paso anterior en el manual de procedimientos. En este ejemplo también se incluye una opción llamada `Default`. Si la automatización evalúa ambas `Choices` y ninguna de ellas es `true`, saltará a un paso llamado `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
```

Creación de un paso **aws:branch** en un manual de procedimientos

Al crear un paso `aws:branch` en un manual de procedimientos, debe definir las `Choices` que la automatización deberá evaluar para determinar a qué paso debe pasar a continuación. Tal y como se ha mencionado anteriormente, las `Choices` se evalúan mediante una expresión booleana. Cada elección debe definir las siguientes opciones:

- NextStep: el siguiente paso en el manual de procedimientos que se debe procesar si la elección designada es `true`.
- Variable: especifique el nombre de un parámetro que se define en la sección `Parameters` del manual de procedimientos, una variable que se define en la sección `Variables` o un objeto de salida de un paso anterior.

  Especifique los valores variables con el siguiente formato.

  `Variable: "{{variable name}}"`

  Especifique los valores de parámetro con el siguiente formato.

  `Variable: "{{parameter name}}"`

  Especifique las variables de objetos de salida con el siguiente formato.

  `Variable: "{{previousStepName.outputName}}"`

  > **Note**
  >
  > La creación de la variable de salida se describe con más detalle en la siguiente sección, Acerca de la creación de la variable de salida.

- Operation: los criterios utilizados para evaluar la elección, como por ejemplo `StringEquals: Linux`. La acción `aws:branch` admite las siguientes operaciones:

  Operaciones de cadena

  - StringEquals
  - EqualsIgnoreCase
  - StartsWith

- EndsWith
- Contiene

Operaciones numéricas

- NumericEquals
- NumericGreater
- NumericLesser
- NumericGreaterOrEquals
- NumericLesser
- NumericLesserOrEquals

Operación booleana

- BooleanEquals

> **Important**
>
> Al crear un manual de procedimientos, el sistema valida cada operación del manual. Si no se admite una operación, el sistema devuelve un error cuando intenta crear el manual de procedimientos.

- Default: define un paso alternativo al que la automatización debe saltar si ninguna de las `Choices` son `true`.

> **Note**
>
> Si no desea especificar un valor `Default`, puede especificar la opción `isEnd`. Si ninguna de las `Choices` es `true` y no se especifica ningún valor `Default`, la automatización se detiene al final del paso.

Utilice las siguientes plantillas para construir el paso `aws:branch` en los manuales de procedimientos. Reemplace cada *`example resource placeholder`* con su propia información.

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"
         }
      }
   ]
}
```

Acerca de la creación de la variable de salida

Para crear una elección `aws:branch` que haga referencia a la salida de un paso anterior, debe identificar el nombre del paso anterior y el nombre del campo de salida. A continuación, combine los nombres del paso y el campo con el siguiente formato.

`Variable: "{{previousStepName.outputName}}"`

Por ejemplo, el primer paso del siguiente ejemplo se denomina `GetInstance`. Y, a continuación, debajo de `outputs`, hay un campo llamado `platform`. En el segundo paso (`ChooseOSforCommands`), el autor quiere hacer referencia a la salida del campo platform como una variable. Para crear la variable, solo tiene que combinar el nombre del paso (GetInstance) y el nombre del campo de salida (platform) para crear `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
```

En este ejemplo, se muestra cómo se crea *"Variable": "{{ describeInstance.Platform }}"* a partir del paso anterior y el resultado.

```
- 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
```

Manuales de procedimientos **`aws:branch`** de ejemplo

A continuación, se muestran algunos ejemplos de manuales de procedimientos que utilizan `aws:branch`.

Ejemplo 1: uso de **`aws:branch`** con una variable de salida para ejecutar comandos según el tipo de sistema operativo

En el primer paso de este ejemplo (`GetInstance`), el autor del manual de procedimientos utiliza la acción `aws:executeAwsApi` para llamar la operación `ssm DescribeInstanceInformation` de la API. El autor utiliza esta acción para determinar el tipo de sistema operativo que utiliza una instancia. La acción `aws:executeAwsApi` devuelve el ID de instancia y el tipo de plataforma.

En el segundo paso (`ChooseOSforCommands`), el autor utiliza la acción `aws:branch` con dos `Choices` (`NextStep: runPowerShellCommand`) y (`NextStep: runShellCommand`). La automatización evalúa el sistema operativo de la instancia a través de la salida del paso anterior

(`Variable: "{{GetInstance.platform}}"`). La automatización salta a un paso para el sistema operativo designado.

```
---
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
```

Ejemplo 2: uso de **`aws:branch`** con una variable de parámetro para ejecutar comandos según el tipo de sistema operativo

El autor del manual de procedimientos define varias opciones de parámetros al principio del manual, en la sección `parameters`. Un parámetro se llama `OperatingSystemName`. En el primer paso (`ChooseOS`), el autor utiliza la acción `aws:branch` con dos `Choices` (`NextStep: runWindowsCommand`) y (`NextStep: runLinuxCommand`). La variable para estas `Choices` hace referencia a la opción del parámetro especificada en la sección de parámetros (`Variable: "{{OperatingSystemName}}"`). Cuando el usuario ejecuta este manual de procedimientos, especifica un valor en el tiempo de ejecución para `OperatingSystemName`. La automatización utiliza el parámetro de tiempo de ejecución durante la evaluación de `Choices`. La automatización salta a un paso para el sistema operativo designado según el parámetro de tiempo de ejecución especificado para `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
```

Creación de automatizaciones con bifurcación complejas a través de operadores

Puede crear automatizaciones con bifurcación complejas mediante el uso de los operadores `And`, `Or` y `Not` en los pasos `aws:branch`.

El operador “And”

Utilice el operador `And` cuando desee que varias variables sean `true` para una elección. En el siguiente ejemplo, la primera elección evalúa si una instancia está en ejecución, `running`, y utiliza el sistema operativo `Windows`. Si la evaluación de ambas variables es da como resultado true, la automatización salta al paso `runPowerShellCommand`. Si una o más de las variables es `false`, la automatización evaluará las variables de la segunda elección.

```
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
```

El operador “Or”

Utilice el operador `Or` cuando desee que cualquiera de entre varias variables sea verdadera para una elección. En el siguiente ejemplo, la primera elección evalúa si una cadena de parámetro es `Windows` y si la salida de un paso AWS Lambda es verdadera. Si la evaluación determina que alguna de estas variables es true, la automatización salta al paso `RunPowerShellCommand`. Si las dos variables son false, la automatización evaluará las variables de la segunda elección.

```
- Or:
  - Variable: "{{parameter1}}"
    StringEquals: Windows
  - Variable: "{{BooleanParam1}}"
    BooleanEquals: true
  NextStep: RunPowershellCommand
- Or:
  - Variable: "{{parameter2}}"
    StringEquals: Linux
  - Variable: "{{BooleanParam2}}"
    BooleanEquals: true
  NextStep: RunShellScript
```

El operador “Not”

Utilice el operador `Not` cuando desee saltar a un paso definido cuando una variable no sea verdadera. En el siguiente ejemplo, la primera elección evalúa si una cadena de parámetro es `Not Linux`. Si la evaluación determina que la variable no es Linux, la automatización salta al paso `sleep2`. Si la evaluación de la primera elección determina que es Linux, entonces la automatización evaluará la siguiente elección.

```
mainSteps:
- name: switch
  action: aws:branch
  inputs:
    Choices:
    - NextStep: sleep2
      Not:
        Variable: "{{testParam}}"
        StringEquals: Linux
    - NextStep: sleep1
      Variable: "{{testParam}}"
      StringEquals: Windows
    Default:
      sleep3
```

Ejemplos de cómo usar opciones condicionales

En esta sección, se incluyen diferentes ejemplos de cómo utilizar opciones dinámicas en un manual de procedimientos. Cada ejemplo de esta sección amplía el siguiente manual de procedimientos. Este manual de procedimientos tiene dos acciones. La primera acción se denomina

`InstallMsiPackage`. Utiliza la `aws:runCommand` acción para instalar una aplicación en una Windows Server instancia. La segunda acción se denomina `TestInstall`. Utiliza la acción `aws:invokeLambdaFunction` para probar la aplicación instalada si esta se instala correctamente. El paso uno especifica on`Failure: Abort`. Esto significa que, si la aplicación no se instala correctamente, la automatización se detiene antes del paso dos.

Ejemplo 1: manual de procedimientos con dos acciones lineales

```
---
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
...
```

Creación de una automatización dinámica que salta a diferentes pasos con la opción **onFailure**

En el siguiente ejemplo, se utilizan las opciones `onFailure: step:step name`, `nextStep` e `isEnd` para crear una automatización dinámica. En este ejemplo, si la acción `InstallMsiPackage` produce un error, la automatización salta a una acción denominada PostFailure (`onFailure: step:PostFailure`) para ejecutar una función de AWS Lambda con la que se realiza alguna acción en caso de que se produzca un error durante la instalación. Si la instalación se realiza correctamente, la automatización salta a la acción TestInstall (`nextStep: TestInstall`). Tanto el paso `TestInstall` como el paso `PostFailure` utilizan la opción `isEnd` (`isEnd: true`) de modo que la automatización se finaliza cuando se haya completado alguno de esos pasos.

> **Note**
>
> Uso de la opción `isEnd` en el último paso de la sección `mainSteps` es opcional. Si el último paso no salta a otros pasos, la automatización se detiene después de ejecutar la acción en el último paso.

Ejemplo 2: una automatización dinámica que salta a diferentes pasos

```
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**
>
> Antes de procesar un manual de procedimientos, el sistema verifica que no cree un bucle infinito. Si se detecta un bucle infinito, Automation devuelve un error y un rastro en forma de círculo que muestra los pasos que crean el bucle.

Creación de una automatización dinámica que define pasos críticos

Puede especificar que un paso es de importancia crítica para el éxito general de la automatización. Si un paso crítico presenta error, Automation informa que el estado de la automatización es `Failed`, incluso aunque uno o más pasos se hayan ejecutado correctamente. En el siguiente ejemplo, se indica que debe ejecutarse el paso VerifyDependencies si se produce un error en el paso InstallMsiPackage (`onFailure: step:VerifyDependencies`). También se indica que el paso `InstallMsiPackage` no es crítico (`isCritical: false`). En este ejemplo, si la aplicación no se instala correctamente, la Automation procesa el paso `VerifyDependencies` para determinar si faltan una o varias dependencias, lo que ha provocado que la aplicación no se instale.

Ejemplo 3: definición de pasos críticos para la automatización

```
---
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
...
```

Uso de salidas de acción como entradas

Varias acciones de automatización devuelven salidas predefinidas. Puede pasar estas salidas como entradas a pasos posteriores de su manual de procedimientos utilizando el formato `{{stepName.outputName}}`. También puede definir salidas personalizadas para las acciones de automatización en sus manuales de procedimientos. Esto le permite ejecutar scripts o invocar operaciones API para otros Servicios de AWS una vez para que pueda reutilizar los valores como entradas en acciones posteriores. Los tipos de parámetros de los manuales de procedimientos son estáticos. Esto significa que el tipo de parámetro no se puede cambiar después de definirlo. Para definir la salida de un paso, proporcione los siguientes campos:

- Nombre: (Requeridas) El nombre de la salida que se utiliza para hacer referencia al valor de salida en los pasos posteriores.
- Selector: (Requerido) La expresión JSONPath que se utiliza para determinar el valor de salida.
- Tipo: (Opcional) El tipo de datos del valor devuelto por el campo selector. Los tipo de valores válidos son `String`, `Integer`, `Boolean`, `StringList`, `StringMap`, `MapList`. El valor predeterminado es `String`.

Si el valor de una salida no coincide con el tipo de datos que especificó, la automatización intentará convertir el tipo de datos. Por ejemplo, si el valor devuelto es un `Integer`, pero el valor `Type` especificado es `String`, el valor de salida final es un valor `String`. Las siguientes conversiones de tipos son compatibles:

- Los valores `String` se pueden convertir en `StringList`, `Integer` y `Boolean`.
- Los valores `Integer` se pueden convertir en `String` y `StringList`.
- Los valores `Boolean` se pueden convertir en `String` y `StringList`.
- Los valores `StringList`, `IntegerList`, o `BooleanList` que contienen un elemento se pueden convertir en `String`, `Integer`, o `Boolean`.

Cuando se utilizan parámetros con acciones de automatización, el tipo de parámetro no se puede cambiar dinámicamente dentro de la entrada de una acción.

Este es un ejemplo de un manual de procedimientos que demuestra cómo definir salidas de acción y hacer referencia al valor como entrada para una acción posterior. El manual de procedimientos hace lo siguiente:

- Utiliza la acción `aws:executeAwsApi` para llamar a la operación de la API de Amazon EC2 DescribeImages a fin de obtener el nombre de una AMI de Windows Server 2016 específica. Da como salida el ID de la imagen como `ImageId`.
- Utiliza la acción `aws:executeAwsApi` para llamar a la operación de la API de Amazon EC2 RunInstances a fin de lanzar una instancia que utiliza el `ImageId` del paso anterior. Da como salida el ID de la instancia como `InstanceId`.
- Utiliza la acción `aws:waitForAwsResourceProperty` para sondear la operación de API de Amazon EC2 DescribeInstanceStatus a fin de esperar hasta que la instancia alcance el estado `running`. La acción agota el tiempo de espera en 60 segundos. El paso agota el tiempo de espera si el estado de la instancia no consigue alcanzar el estado `running` después de 60 segundos de sondeo.
- Usa la acción `aws:assertAwsResourceProperty` para llamar a la operación de la API de Amazon EC2 `DescribeInstanceStatus` y confirmar que la instancia se encuentra en estado `running`. El paso presenta un error si el estado de la instancia no es `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"
```

```
...
```

Cada una de las acciones de automatización descritas anteriormente le permiten llamar una operación de la API determinada mediante la especificación del espacio de nombres de servicio, el nombre de la operación de la API, los parámetros de entrada y los parámetros de salida. Las entradas se definen con la operación de la API que elija. Puede ver las operaciones de la API (también llamadas métodos) si elige un servicio en el panel de navegación ubicado a la izquierda, en la siguiente página: Services Reference. Elija un método en la sección Cliente para el servicio que desea invocar. Por ejemplo, todas las operaciones de la API (los métodos) para Amazon Relational Database Service (Amazon RDS) se indican en la siguiente página: métodos de Amazon RDS.

Puede ver el esquema para cada acción de automatización en las siguientes ubicaciones:

- `aws:assertAwsResourceProperty`: confirmar el estado de un recurso o un evento de AWS
- `aws:executeAwsApi`: llamar y ejecutar operaciones de la API de AWS
- `aws:waitForAwsResourceProperty`: esperar una propiedad de recurso de AWS

Los esquemas incluyen descripciones de los campos obligatorios para utilizar cada acción.

Uso de los campos Selector/PropertySelector

Cada acción de Automation requiere que especifique una salida `Selector` (para `aws:executeAwsApi`) o un `PropertySelector` (para `aws:assertAwsResourceProperty` y `aws:waitForAwsResourceProperty`). Estos campos se utilizan para procesar la respuesta JSON desde una operación de la API de AWS. Estos campos utilizan la sintaxis de JSONPath.

A continuación, se muestra un ejemplo que tiene como objetivo ilustrar este concepto para la acción `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"
...
```

En `aws:executeAwsApi` del paso `getImageId`, la automatización invoca la operación de la API `DescribeImages` y recibe una respuesta de `ec2`. A continuación, la automatización aplica `Selector - "$.Images[0].ImageId"` a la respuesta de la API y asigna el valor seleccionado a la variable `ImageId` de salida. Otros pasos de la misma automatización pueden utilizar el valor de `ImageId` especificando `"{{ getImageId.ImageId }}"`.

A continuación, se muestra un ejemplo que tiene como objetivo ilustrar este concepto para la acción `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
...
```

En `aws:waitForAwsResourceProperty` del paso `waitUntilInstanceStateRunning`, la automatización invoca la operación de la API `DescribeInstanceStatus` y recibe una respuesta de `ec2`. La automatización aplica `PropertySelector - "$.InstanceStatuses[0].InstanceState.Name"` a la respuesta y comprueba si el valor devuelto especificado coincide con un valor en la lista `DesiredValues` (en este caso, `running`). El paso repite el proceso hasta que la respuesta devuelve un estado de instancia de `running`.

Uso de JSONPath en un manual de procedimientos

Una expresión JSONPath es una cadena que comienza con “$” que se utiliza para seleccionar uno de varios componentes dentro de un elemento JSON. La siguiente lista incluye información sobre los operadores de JSONPath que Automatización de Systems Manager admite:

- Elemento secundario con notación de puntos (.): utilizar con un objeto JSON. Este operador selecciona el valor de una clave específica.
- Análisis profundo (..): utilizar con un elemento JSON. Este operador analiza el nivel de elemento JSON por nivel y selecciona una lista de valores con la clave específica. El tipo de retorno de este operador siempre es una matriz JSON. En el contexto de un tipo de salida de la acción de automatización, el operador puede ser StringList o MapList.
- Índice de matriz ([ ]): utilizar con una matriz JSON. Este operador obtiene el valor de un índice específico.
- Filtro ([?(***expresión***)]): se usa con una matriz JSON. Este operador filtra los valores de la matriz JSON que coinciden con los criterios definidos en la expresión de filtro. Las expresiones de filtro solo pueden utilizar los siguientes operadores: ==,! =, >, <, >= o <=. La combinación de varias expresiones de filtro con AND (&&) u OR (||) no es compatible. El tipo de retorno de este operador siempre es una matriz JSON.

Para comprender mejor los operadores de JSONPath, revise la siguiente respuesta de JSON de la operación de la API de EC2 `DescribeInstances`. Debajo de esta respuesta se muestran algunos ejemplos con resultados diferentes si se aplican distintas expresiones JSONPath a la respuesta de la operación de la 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": []
    }
]
}
```

Ejemplo de JSONPath 1: obtener una cadena específica de una respuesta de JSON

```
JSONPath:
```

```
$.Reservations[0].Instances[0].ImageId

Returns:
"ami-12345678"

Type: String
```

Ejemplo de JSONPath 2: obtener un booleano específico de una respuesta de JSON

```
JSONPath:
$.Reservations[0].Instances[0].BlockDeviceMappings[0].Ebs.DeleteOnTermination

Returns:
true

Type: Boolean
```

Ejemplo de JSONPath 3: obtener un entero específico de una respuesta de JSON

```
JSONPath:
$.Reservations[0].Instances[0].State.Code

Returns:
16

Type: Integer
```

Ejemplo de JSONPath 4: analizar en profundidad una respuesta de JSON y, a continuación, obtener todos los valores para VolumeId como una StringList

```
JSONPath:
$.Reservations..BlockDeviceMappings..VolumeId

Returns:
[
   "vol-000000000000",
   "vol-111111111111"
]

Type: StringList
```

Ejemplo de JSONPath 5: obtener un objeto BlockDeviceMappings como un StringMap

```
JSONPath:
$.Reservations[0].Instances[0].BlockDeviceMappings[0]

Returns:
{
   "Ebs" : {
      "DeleteOnTermination" : true,
      "Status" : "attached",
      "VolumeId" : "vol-000000000000"
   },
   "DeviceName" : "/dev/xvda"
}

Type: StringMap
```

Ejemplo de JSONPath 6: analizar en profundidad una respuesta de JSON y, a continuación, obtener todos los objetos de estado como una MapList

```
JSONPath:
$.Reservations..Instances..State

Returns:
[
   {
      "Code" : 16,
      "Name" : "running"
   },
   {
      "Code" : 80,
      "Name" : "stopped"
   }
]

Type: MapList
```

Ejemplo 7 de JSONPath: filtro para instancias en el **`running`** estado

```
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
```

Ejemplo 8 de JSONPath: devuelve el **`ImageId`** de instancias que no están en el **`running`** estado

```
JSONPath:
$.Reservations..Instances[?(@.State.Name != 'running')].ImageId

Returns:
[
  "ami-12345678"
]

Type: StringList | String
```

Crear integraciones webhook para Automation

Para enviar mensajes mediante manuales de procedimientos durante una automatización, cree una integración. Las integraciones se pueden invocar durante una automatización mediante la acción `aws:invokeWebhook` en su manual de procedimientos. Si aún no ha creado un manual de procedimientos, consulte Creación de webhooks para integraciones. Para obtener más información acerca de la acción `aws:invokeWebhook`, consulte `aws:invokeWebhook`: invocar una integración de webhook de Automation.

Como se muestra en los siguientes procedimientos, puede crear una integración mediante la consola de Automatización de Systems Manager o su herramienta de la línea de comandos preferida.

Creación de integraciones (consola)

Para crear una integración para Automation (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija automatización.
3. Elija la pestaña Integrations (Integraciones).
4. Seleccione Add integration (Agregar integración) y elija Webhook.
5. Ingrese los valores obligatorios y los valores opcionales que desee incluir para la integración.
6. Elija Add (Agregar) para crear la integración.

Creación de integraciones (línea de comandos)

Para crear una integración mediante herramientas de la línea de comandos, debe crear el parámetro `SecureString` obligatorio para una integración. Automatización utiliza un espacio de nombres reservado en Parameter Store, una herramienta de Systems Manager, para almacenar información sobre su integración. Si crea una integración mediante la AWS Management Console, Automation se encarga de este proceso por usted. Después del espacio de nombres, debe especificar el tipo de integración que desea crear y, a continuación, el nombre de la integración. Actualmente, Automation admite la integración de tipos `webhook`.

Los campos admitidos para las integraciones de tipo `webhook` son las siguientes:

- Descripción
- headers
- payload
- URL

Antes de empezar

Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI) o las Herramientas de AWS para PowerShell. Para obtener información, consulte Instalación o

actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.

Para crear una integración para Automation (línea de comandos)

- Ejecute los siguientes comandos para crear el parámetro `SecureString` para una integración. Reemplace cada *example resource placeholder* con su propia información. El espacio de nombres `/d9d01087-4a3f-49e0-b0b4-d568d7826553/ssm/integrations/webhook/` está reservado en Parameter Store para las integraciones. El nombre del parámetro debe utilizar este espacio de nombres seguido del nombre de la integración. Por ejemplo, `/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"}'
```

Creación de webhooks para integraciones

Al crear webhooks con su proveedor, tenga en cuenta lo siguiente:

- El protocolo debe ser HTTPS.
- Se admiten encabezados de solicitud personalizados.
- Se puede especificar un cuerpo de la solicitud predeterminado.
- El cuerpo de la solicitud predeterminado se puede anular cuando se invoca una integración mediante la acción `aws:invokeWebhook`.

Administración de los tiempos de espera en los manuales de procedimientos

Todas las acciones de automatización comparten la propiedad `timeoutSeconds`. Esta propiedad se puede utilizar para especificar el valor de tiempo de espera de la ejecución de una acción. Además, se puede modificar el modo en que afecta a la automatización y al estado general de la ejecución el hecho de que se agote dicho tiempo para una acción. Para ello, puede definir también las propiedades compartidas `onFailure` y `isCritical` de una acción.

Por ejemplo, dependiendo del caso de uso, si se agota el tiempo de espera de una acción, tal vez prefiera que la automatización continúe con una acción diferente y que no afecte a su estado general. En este ejemplo, va a especificar con la propiedad `timeoutSeconds` el tiempo que debe esperarse antes de que se agote el tiempo de espera de la acción. A continuación, especifica la acción o el paso al que la automatización debe pasar si se agota el tiempo de espera. Especifique un valor utilizando el formato `step:step name` en la propiedad `onFailure` en lugar del valor predeterminado de `Abort`. De forma predeterminada, si se agota el tiempo de espera de una acción, el estado de ejecución de la automatización será `Timed Out`. Para evitar que el agotamiento del tiempo de espera afecte el estado de ejecución de la automatización, especifique `false` en la propiedad `isCritical`.

En el ejemplo siguiente, se muestra cómo se definen las propiedades compartidas de una acción que se describe en este escenario.

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"
}
```

Para obtener más información acerca de las propiedades que comparten todas las acciones de automatización, consulte Propiedades compartidas por todas las acciones.

Referencia del manual de procedimientos de Automatización de Systems Manager

Para ayudarlo a empezar rápidamente, AWS Systems Manager proporciona manuales de procedimientos predefinidos. Amazon Web Services, AWS Support y AWS Config mantienen estos manuales de procedimientos. La referencia del manual de procedimientos describe cada uno de los manuales predefinidos que proporcionan Systems Manager, Soporte y AWS Config. Para obtener más información, consulte Referencia del manual de procedimientos de Automatización de Systems Manager.

Tutoriales

Los siguientes tutoriales le serán de ayuda para utilizar Automation de AWS Systems Manager para abordar casos de uso comunes. Estos tutoriales demuestran cómo usar sus propios manuales de procedimientos, manuales de procedimientos predefinidos proporcionados por Automatización y otras herramientas de Systems Manager con otros Servicios de AWS.

Contenido

- Actualización de AMIs
  - Actualizar una AMI de Linux
  - Actualizar una AMI (AWS CLI) de Linux
  - Actualización de un Windows Server AMI
  - Actualice un golden AMI mediante la Automation, AWS Lambda, y Parameter Store
    - Tarea 1: crear un parámetro en Systems Manager Parameter Store
    - Tarea 2: crear un rol de IAM para AWS Lambda
    - Tarea 3: crear una función de AWS Lambda
    - Tarea 4: crear un manual de procedimientos y aplicar revisiones a la AMI
  - Actualización de las AMIs mediante Automatización y Jenkins
  - Actualización de AMIs para grupos de escalado automático
    - Crear el manual de procedimientos PatchAMIAndUpdateASG
- Uso de los manuales de procedimientos de autoservicio de AWS Support
  - Ejecutar la herramienta EC2Rescue en instancias inaccesibles
    - Funcionamiento
    - Antes de empezar

        - Concesión de permisos AWSSupport-EC2Rescue para realizar acciones en las instancias
            - Concesión de permisos mediante políticas de IAM
            - Concesión de permisos mediante una plantilla de AWS CloudFormation
        - Ejecución de Automation
    - Restablecimiento de contraseñas y claves de SSH en instancias EC2
        - Funcionamiento
        - Antes de empezar
            - Concesión de permisos a AWSSupport-EC2Rescue para realizar acciones en las instancias
                - Concesión de permisos mediante políticas de IAM
                - Concesión de permisos mediante una plantilla de AWS CloudFormation
        - Ejecución de Automation
- Transferir datos a Automatización usando transformadores de entrada

Actualización de AMIs

Los siguientes tutoriales explican cómo actualizar Amazon Machine Image (AMIs) para incluir las revisiones más recientes.

Temas

- Actualizar una AMI de Linux
- Actualizar una AMI (AWS CLI) de Linux
- Actualización de un Windows Server AMI
- Actualice un golden AMI mediante la Automation, AWS Lambda, y Parameter Store
- Actualización de las AMIs mediante Automatización y Jenkins
- Actualización de AMIs para grupos de escalado automático

Actualizar una AMI de Linux

Este tutorial de Automatización de Systems Manager le muestra cómo utilizar la consola o AWS CLI y el manual de procedimientos de `AWS-UpdateLinuxAmi` para actualizar una AMI de Linux con las revisiones más recientes de los paquetes que especifique. Automatización es una herramienta de

AWS Systems Manager. El manual de procedimientos `AWS-UpdateLinuxAmi` también automatiza la instalación de paquetes y configuraciones adicionales que sean específicos del sitio. Puede actualizar diversas distribuciones de Linux con esta explicación, incluidas Ubuntu Server, CentOS, RHEL, SLES o Amazon Linux AMIs. Para obtener una lista completa de las versiones de Linux compatibles, consulte Requisitos previos de Patch Manager.

El manual de procedimientos de `AWS-UpdateLinuxAmi` le permite automatizar las tareas de mantenimiento de imágenes sin tener que crear el manual de procedimientos en JSON o YAML. Puede utilizar el manual de procedimientos `AWS-UpdateLinuxAmi` para realizar los siguientes tipos de tareas.

- Actualizar todos los paquetes de distribución y el software de Amazon en una Amazon Machine Image (AMI) de Amazon Linux, Red Hat Enterprise Linux, Ubuntu Server, SUSE Linux Enterprise Server o CentOS. Este es el comportamiento predeterminado del manual de procedimientos.
- Instale AWS Systems Manager SSM Agent en una imagen existente para habilitar las herramientas de Systems Manager, como la ejecución remota de comandos mediante AWS Systems Manager Run Command o la recopilación de inventario de software con Inventario.
- Instalar paquetes de software adicionales.

Antes de empezar

Antes de empezar a trabajar con los manuales de procedimientos, configure los roles y, opcionalmente, EventBridge para Automation. Para obtener más información, consulte Configuración de Automation. Esta explicación también requiere que especifique el nombre de un perfil de instancia de AWS Identity and Access Management (IAM). Para obtener más información sobre cómo crear un perfil de instancia de IAM, consulte Configuración de permisos de instancia requeridos para Systems Manager.

El manual de procedimientos `AWS-UpdateLinuxAmi` acepta los siguientes parámetros de entrada.

Parámetro	Tipo	Descripción
SourceAmiId	Cadena	(Obligatorio) ID de AMI de origen.
IamInstanceProfileName	Cadena	(Obligatorio) Nombre del rol de perfil de instancia de IAM que ha creado en Configura

Parámetro	Tipo	Descripción
		ción de permisos de instancia requeridos para Systems Manager. El rol de perfil de instancia concede permiso a Automation para que realice acciones en sus instancias, como ejecutar comandos o iniciar y detener servicios. El manual de procedimientos utiliza únicamente el nombre del rol de perfil de instancia . Si especifica el nombre de recurso de Amazon (ARN), se produce un error en la automatización.
AutomationAssumeRole	Cadena	(Obligatorio) El nombre del rol de servicio de IAM que ha creado en Configuración de Automation. El rol de servicio (también denominad o rol de asunción) concede permiso a Automation para asumir el rol de IAM y realizar acciones en su nombre. Por ejemplo, el rol de servicio permite a Automation crear una AMI nueva al ejecutar la acción `aws:createImage` en un manual de procedimi entos. Para este parámetro , debe especificarse el ARN completo.

Parámetro	Tipo	Descripción
TargetAmiName	Cadena	(Opcional) El nombre de la nueva AMI después de que se cree. El nombre predeterminado es una cadena generada por el sistema que incluye el ID de la AMI de origen, así como la hora y la fecha de creación.
InstanceType	Cadena	(Opcional) El tipo de instancia que se lanzará como el host de espacio de trabajo. Los tipos de instancia varían según la región. El tipo predeterminado es t2.micro.
PreUpdateScript	Cadena	(Opcional) La URL de un script que se ejecutará antes de que se apliquen las actualizaciones. El valor predeterminado (\"none\") es no ejecutar un script.
PostUpdateScript	Cadena	(Opcional) La URL de un script que se ejecutará después de que se apliquen las actualizaciones de paquete. El valor predeterm inado (\"none\") es no ejecutar un script.

Parámetro	Tipo	Descripción
IncludePackages	Cadena	(Opcional) Actualizar solo estos paquetes designados. De forma predeterminada (\"all\"), se aplican todas las actualizaciones disponibles.
ExcludePackages	Cadena	(Opcional) Nombres de los paquetes a los que no se aplicarán las actualizaciones, en todas las condiciones. De forma predeterminada (\"none\"), no se excluye ningún paquete.

Pasos de Automation

El manual de procedimientos `AWS-UpdateLinuxAmi` incluye las siguientes acciones de automatización de forma predeterminada.

Paso 1: launchInstance (acción **`aws:runInstances`**)

En este paso, se lanza una instancia con los datos de usuario de Amazon Elastic Compute Cloud (Amazon EC2) y un rol de perfil de instancia de IAM. Los datos de usuario instalan el SSM Agent adecuado en función del sistema operativo. La instalación del SSM Agent le permite utilizar herramientas de Systems Manager, como Run Command, State Manager e Inventario.

Paso 2: updateOSSoftware (acción **`aws:runCommand`**)

Este paso ejecuta los siguientes comandos en la instancia lanzada:

- Descarga un script de actualización de Amazon S3.
- Ejecuta un script de preactualización opcional.
- Actualiza los paquetes de distribución y el software de Amazon.
- Ejecuta un script de posactualización opcional.

El registro de ejecución se almacena en la carpeta /tmp para que el usuario la consulte más tarde.

Si desea actualizar un conjunto específico de paquetes, puede proporcionar la lista utilizando el parámetro `IncludePackages`. Cuando se proporciona, el sistema intenta actualizar únicamente estos paquetes y sus dependencias. No se realizan otras actualizaciones. De forma predeterminada, cuando no se especifica ningún paquete de inclusión, el programa actualiza todos los paquetes disponibles.

Si desea excluir la actualización de un conjunto específico de paquetes, puede proporcionar la lista al parámetro `ExcludePackages`. Si se proporciona, estos paquetes permanecen en su versión actual, independientemente de otras opciones especificadas. De forma predeterminada, cuando no se especifica ningún paquete de exclusión, no se excluye ningún paquete.

Paso 3: stopInstance (acción **`aws:changeInstanceState`**)

Este paso detiene la instancia actualizada.

Paso 4: createImage (acción **`aws:createImage`**)

Este paso crea una AMI con un nombre descriptivo que se enlaza con ID de origen y la hora de creación. Por ejemplo: “AMI generada con EC2 Automation a la(s) {{global:DATE_TIME}} a partir de {{SourceAmiId}}” donde DATE_TIME y SourceID representan variables de Automation.

Paso 5: terminateInstance (acción **`aws:changeInstanceState`**)

Este paso limpia la automatización con la terminación de la instancia en ejecución.

Output

La automatización devuelve el ID de la AMI nueva como salida.

> **Note**
>
> De forma predeterminada, cuando Automation ejecuta el manual de procedimientos `AWS-UpdateLinuxAmi`, el sistema crea una instancia temporal en la VPC predeterminada (172.30.0.0/16). Si ha eliminado la VPC predeterminada, recibirá el siguiente error: `VPC not defined 400`
> Para solucionar este problema, debe generar una copia del manual de procedimientos `AWS-UpdateLinuxAmi` y especificar un ID de subred. Para obtener más información, consulte VPC not defined 400.

Para crear una AMI con revisiones mediante Automation (AWS Systems Manager)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija automatización.
3. Elija Ejecutar automatización.
4. En la lista Automation document (Documento de Automation), elija **`AWS-UpdateLinuxAmi`**.
5. En la sección Document details (Detalles del documento), compruebe que el valor de Document version (Versión del documento) es Default version at runtime (Versión predeterminada en tiempo de ejecución).
6. Elija Siguiente.
7. En la sección Execution mode (Modo de ejecución), seleccione Simple Execution (Ejecución sencilla).
8. En la sección Input parameters (Parámetros de entrada), ingrese la información que ha recopilado en la sección Before you begin (Antes de empezar).
9. Elija Ejecutar. La consola muestra el estado de la ejecución de Automation.

Una vez finalizada la automatización, lance una instancia de prueba desde la AMI actualizada para verificar los cambios.

> **Note**
>
> Si se produce un error en cualquier paso de la automatización, la información acerca del error se muestra en la página Automation Executions (Ejecuciones de automatizaciones). La automatización está diseñada para terminar la instancia temporal después de completar correctamente todas las tareas. Si se produce un error en un paso, puede que el sistema no finalice la instancia. Por lo tanto, si se produce un error en un paso, termine manualmente la instancia temporal.

Actualizar una AMI (AWS CLI) de Linux

En esta explicación de AWS Systems Manager Automation, se muestra cómo utilizar la AWS Command Line Interface (AWS CLI) y el manual de procedimientos `AWS-UpdateLinuxAmi` de Systems Manager para aplicar revisiones de forma automática a una Amazon Machine Image (AMI) de Linux con las versiones más recientes de los paquetes que usted especifique. Automatización es

una herramienta de AWS Systems Manager. El manual de procedimientos `AWS-UpdateLinuxAmi` también automatiza la instalación de paquetes y configuraciones adicionales que sean específicos del sitio. Puede actualizar diversas distribuciones de Linux con esta explicación, incluidas Ubuntu Server, CentOS, RHEL, SLES o Amazon Linux AMIs. Para obtener una lista completa de las versiones de Linux compatibles, consulte Requisitos previos de Patch Manager.

El manual de procedimientos `AWS-UpdateLinuxAmi` le permite automatizar las tareas de mantenimiento de imágenes sin tener que crear el manual de procedimientos en JSON o YAML. Puede utilizar el manual de procedimientos `AWS-UpdateLinuxAmi` para realizar los siguientes tipos de tareas.

- Actualizar todos los paquetes de distribución y el software de Amazon en Amazon Linux Red Hat Enterprise Linux, Ubuntu Server, SLES, o Cent OS Amazon Machine Image (AMI). Este es el comportamiento predeterminado del manual de procedimientos.
- Instalar AWS Systems Manager SSM Agent en una imagen existente para habilitar las capacidades de Systems Manager, como la ejecución remota de comandos mediante AWS Systems Manager Run Command o la recopilación de inventario de software con Inventory.
- Instalar paquetes de software adicionales.

Antes de empezar

Antes de empezar a trabajar con los manuales de procedimientos, configure los roles y, opcionalmente, EventBridge para Automation. Para obtener más información, consulte Configuración de Automation. Esta explicación también requiere que especifique el nombre de un perfil de instancia de AWS Identity and Access Management (IAM). Para obtener más información sobre cómo crear un perfil de instancia de IAM, consulte Configuración de permisos de instancia requeridos para Systems Manager.

El manual de procedimientos `AWS-UpdateLinuxAmi` acepta los siguientes parámetros de entrada.

Parámetro	Tipo	Descripción
SourceAmiId	Cadena	(Obligatorio) ID de AMI de origen. Puede referenciar automáticamente el último ID de una AMI de Amazon EC2 para Linux mediante un

Parámetro	Tipo	Descripción
		parámetro de AWS Systems Manager Parameter Store público. Para obtener más información, consulte Query for the latest Amazon Linux AMI IDs using AWS Systems Manager Parameter Store.
IamInstanceProfileName	Cadena	(Obligatorio) Nombre del rol de perfil de instancia de IAM que ha creado en Configuración de permisos de instancia requeridos para Systems Manager. El rol de perfil de instancia concede permiso a Automation para que realice acciones en sus instancias, como ejecutar comandos o iniciar y detener servicios. El manual de procedimientos utiliza únicamente el nombre del rol de perfil de instancia.

Parámetro	Tipo	Descripción
AutomationAssumeRole	Cadena	(Obligatorio) El nombre del rol de servicio de IAM que ha creado en Configuración de Automation. El rol de servicio (también denominado rol de asunción) concede permiso a Automation para asumir el rol de IAM y realizar acciones en su nombre. Por ejemplo, el rol de servicio permite a Automation crear una AMI nueva al ejecutar la acción `aws:createImage` en un manual de procedimientos. Para este parámetro, debe especificarse el ARN completo.
TargetAmiName	Cadena	(Opcional) El nombre de la nueva AMI después de que se cree. El nombre predeterminado es una cadena generada por el sistema que incluye el ID de la AMI de origen, así como la hora y la fecha de creación.
InstanceType	Cadena	(Opcional) El tipo de instancia que se lanzará como el host de espacio de trabajo. Los tipos de instancia varían según la región. El tipo predeterminado es t2.micro.

Parámetro	Tipo	Descripción
PreUpdateScript	Cadena	(Opcional) La URL de un script que se ejecutará antes de que se apliquen las actualizaciones. El valor predeterminado (\"none\") es no ejecutar un script.
PostUpdateScript	Cadena	(Opcional) La URL de un script que se ejecutará después de que se apliquen las actualizaciones de paquete. El valor predeterm inado (\"none\") es no ejecutar un script.
IncludePackages	Cadena	(Opcional) Actualizar solo estos paquetes designado s. De forma predeterminada (\"all\"), se aplican todas las actualizaciones disponibles.
ExcludePackages	Cadena	(Opcional) Nombres de los paquetes a los que no se aplicarán las actualizaciones, en todas las condiciones. De forma predeterminada (\"none \"), no se excluye ningún paquete.

Pasos de Automation

El manual de procedimientos `AWS-UpdateLinuxAmi` incluye los siguientes pasos de forma predeterminada.

Paso 1: launchInstance (acción **`aws:runInstances`**)

En este paso, se lanza una instancia con los datos de usuario de Amazon Elastic Compute Cloud (Amazon EC2) y un rol de perfil de instancia de IAM. Los datos de usuario instalan SSM Agent adecuado en función del sistema operativo. La instalación del SSM Agent le permite utilizar herramientas de Systems Manager, como Run Command, State Manager e Inventario.

Paso 2: updateOSSoftware (acción **`aws:runCommand`**)

Este paso ejecuta los siguientes comandos en la instancia lanzada:

- Descarga un script de actualización de Amazon Simple Storage Service (Amazon S3).
- Ejecuta un script de preactualización opcional.
- Actualiza los paquetes de distribución y el software de Amazon.
- Ejecuta un script de posactualización opcional.

El registro de ejecución se almacena en la carpeta /tmp para que el usuario la consulte más tarde.

Si desea actualizar un conjunto específico de paquetes, puede proporcionar la lista utilizando el parámetro `IncludePackages`. Cuando se proporciona, el sistema intenta actualizar únicamente estos paquetes y sus dependencias. No se realizan otras actualizaciones. De forma predeterminada, cuando no se especifica ningún paquete de inclusión, el programa actualiza todos los paquetes disponibles.

Si desea excluir la actualización de un conjunto específico de paquetes, puede proporcionar la lista al parámetro `ExcludePackages`. Si se proporciona, estos paquetes permanecen en su versión actual, independientemente de otras opciones especificadas. De forma predeterminada, cuando no se especifica ningún paquete de exclusión, no se excluye ningún paquete.

Paso 3: stopInstance (acción **`aws:changeInstanceState`**)

Este paso detiene la instancia actualizada.

Paso 4: createImage (acción **`aws:createImage`**)

Este paso crea una AMI con un nombre descriptivo que se enlaza con ID de origen y la hora de creación. Por ejemplo: “AMI generada con EC2 Automation a la(s) {{global:DATE_TIME}} a partir de {{SourceAmiId}}” donde DATE_TIME y SourceID representan variables de Automation.

Paso 5: terminateInstance (acción **`aws:changeInstanceState`**)

Este paso limpia la automatización con la terminación de la instancia en ejecución.

Output

La automatización devuelve el ID de la AMI nueva como salida.

> **Note**
>
> De forma predeterminada, cuando Automation ejecuta el manual de procedimientos `AWS-UpdateLinuxAmi`, el sistema crea una instancia temporal en la VPC predeterminada (172.30.0.0/16). Si ha eliminado la VPC predeterminada, recibirá el siguiente error: `VPC not defined 400`
> Para solucionar este problema, debe generar una copia del manual de procedimientos `AWS-UpdateLinuxAmi` y especificar un ID de subred. Para obtener más información, consulte VPC not defined 400.

Para crear una AMI con revisiones mediante Automation

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Ejecute el siguiente comando para ejecutar el manual de procedimientos `AWS-UpdateLinuxAmi`. Reemplace cada *`example resource placeholder`* con su propia información.

   ```
   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'
   ```

   El comando devuelve un ID de ejecución. Copie este ID en el portapapeles. Utilizará este ID para ver el estado de la automatización.

   ```
   {
       "AutomationExecutionId": "automation execution ID"
   ```

```
}
```

3. Ejecute el siguiente comando para ver la automatización con la AWS CLI:

```
aws ssm describe-automation-executions
```

4. Ejecute el siguiente comando para ver detalles acerca del progreso de la automatización. Reemplace *automation execution ID* con su propia información.

```
aws ssm get-automation-execution --automation-execution-id automation execution ID
```

El proceso de actualización puede tardar 30 minutos o más en completarse.

> **Note**
>
> También puede monitorear el estado de la automatización en la consola. En la lista, elija la automatización que acaba de ejecutar y, a continuación, elija la pestaña Steps (Pasos). Esta pestaña le muestra el estado de las acciones de la automatización.

Una vez finalizada la automatización, lance una instancia de prueba desde la AMI actualizada para verificar los cambios.

> **Note**
>
> Si se produce un error en cualquier paso de la automatización, la información acerca del error se muestra en la página Automation Executions (Ejecuciones de automatizaciones). La automatización está diseñada para terminar la instancia temporal después de completar correctamente todas las tareas. Si se produce un error en un paso, puede que el sistema no finalice la instancia. Por lo tanto, si se produce un error en un paso, termine manualmente la instancia temporal.

Actualización de un Windows Server AMI

El manual de procedimientos `AWS-UpdateWindowsAmi` le permite automatizar las tareas de mantenimiento de imágenes en sus Amazon Machine Image (AMI) de Amazon Windows sin tener que crear el manual de procedimientos en JSON o YAML. Este manual de procedimientos es

compatible con Windows Server 2008 R2 o versiones posteriores. Puede utilizar el manual de procedimientos `AWS-UpdateWindowsAmi` para realizar los siguientes tipos de tareas.

- Instalar todas las actualizaciones de Windows y actualizar el software de Amazon (comportamiento predeterminado).
- Instalar actualizaciones de Windows específicas y actualizar el software de Amazon.
- Personalizar una AMI con sus propios scripts.

Antes de empezar

Antes de empezar a trabajar con manuales de procedimientos, configure roles para Automation a fin de agregar una política `iam:PassRole` que referencie el ARN del perfil de instancia al que desea conceder acceso. De forma opcional, configure Amazon EventBridge para Automatización, una herramienta de AWS Systems Manager. Para obtener más información, consulte Configuración de Automation. Esta explicación también requiere que especifique el nombre de un perfil de instancia de AWS Identity and Access Management (IAM). Para obtener más información sobre cómo crear un perfil de instancia de IAM, consulte Configuración de permisos de instancia requeridos para Systems Manager.

> **Note**
>
> Las actualizaciones de AWS Systems Manager SSM Agent normalmente se implementan en diferentes regiones y en distintos momentos. Al personalizar o actualizar una AMI, utilice únicamente las AMI de origen publicadas para la región en la que está trabajando. De este modo, se asegurará de que está trabajando con el último SSM Agent lanzado para esa región y evitará problemas de compatibilidad.

El manual de procedimientos `AWS-UpdateWindowsAmi` acepta los siguientes parámetros de entrada.

Parámetro	Tipo	Descripción
SourceAmiId	Cadena	(Obligatorio) ID de AMI de origen. Puede referenciar automáticamente el ID de la versión más reciente de

Parámetro	Tipo	Descripción
		la AMI de Windows Server mediante un parámetro de Systems Manager Parameter Store público. Para obtener más información, consulte Query for the latest Windows AMI IDs using AWS Systems ManagerParameter Store.
SubnetId	Cadena	(Opcional) La subred en la que quiere lanzar la instancia temporal. Debe especificar un valor para este parámetro si ha eliminado su VPC predeterminada.
IamInstanceProfileName	Cadena	(Obligatorio) Nombre del rol de perfil de instancia de IAM que ha creado en Configuración de permisos de instancia requeridos para Systems Manager. El rol de perfil de instancia concede permiso a Automation para que realice acciones en sus instancias, como ejecutar comandos o iniciar y detener servicios. El manual de procedimientos utiliza únicamente el nombre del rol de perfil de instancia.

Parámetro	Tipo	Descripción
AutomationAssumeRole	Cadena	(Obligatorio) El nombre del rol de servicio de IAM que ha creado en Configuración de Automation. El rol de servicio (también denominado rol de asunción) concede permiso a Automation para asumir el rol de IAM y realizar acciones en su nombre. Por ejemplo, el rol de servicio permite a Automation crear una AMI nueva al ejecutar la acción `aws:createImage` en un manual de procedimientos. Para este parámetro, debe especificarse el ARN completo.
TargetAmiName	Cadena	(Opcional) El nombre de la nueva AMI después de que se cree. El nombre predeterminado es una cadena generada por el sistema que incluye el ID de la AMI de origen, así como la hora y la fecha de creación.
InstanceType	Cadena	(Opcional) El tipo de instancia que se lanzará como el host de espacio de trabajo. Los tipos de instancia varían según la región. El tipo predeterminado es t2.medium.

Parámetro	Tipo	Descripción
PreUpdateScript	Cadena	(Opcional) Un script que se ejecutará antes de actualizar la AMI. Ingrese un script en el manual de procedimientos o en el tiempo de ejecución como parámetro.
PostUpdateScript	Cadena	(Opcional) Un script que se ejecutará después de actualizar la AMI. Ingrese un script en el manual de procedimientos o en el tiempo de ejecución como parámetro.
IncludeKbs	Cadena	(Opcional) Especifique uno o varios ID de artículo de la Base de conocimientos de Microsoft (KB) para incluirlos. Puede instalar varios ID utilizando valores separados por comas. Formatos válidos: KB9876543 o 9876543.
ExcludeKbs	Cadena	(Opcional) Especifique uno o varios ID de artículo de la Base de conocimientos de Microsoft (KB) para excluirlos. Puede excluir varios ID utilizando valores separados por comas. Formatos válidos: KB9876543 o 9876543.

Parámetro	Tipo	Descripción
Categorías	Cadena	(Opcional) Especifique una o más categorías de actualización. Puede filtrar las categorías usando valores separados por comas. Opciones: Critical Update, Security Update, Definitio n Update, Update Rollup, Service Pack, Tool, Update o Driver. Los formatos válidos incluyen una sola entrada, por ejemplo: Critical Update. O bien, puede especificar una lista separada por comas: Critical Update,Security Update,Definition Update.
SeverityLevels	Cadena	(Opcional) Especifique uno o varios niveles de seguridad de MSRC asociados con una actualización. Puede filtrar los niveles de gravedad usando valores separados por comas. Opciones: Critical, Important , Low, Moderate o Unspecifi ed. Los formatos válidos incluyen una sola entrada, por ejemplo: Critical. O bien, puede especificar una lista separada por comas: Critical, Important,Low.

Pasos de Automation

El manual de procedimientos `AWS-UpdateWindowsAmi` incluye los siguientes pasos de forma predeterminada.

Paso 1: launchInstance (acción **`aws:runInstances`**)

Este paso lanza una instancia con un rol de perfil de instancia de IAM desde el `SourceAmiID` especificado.

Paso 2: runPreUpdateScript (acción **`aws:runCommand`**)

Este paso le permite especificar un script como una cadena que se ejecuta antes de que se instalen las actualizaciones.

Paso 3: updateEC2Config (acción **`aws:runCommand`**)

En este paso, se utiliza el manual de procedimientos `AWS-InstallPowerShellModule` para descargar un módulo de PowerShell de AWS público. Systems Manager verifica la integridad del módulo con un hash SHA-256. A continuación, Systems Manager revisa el sistema operativo para determinar si debe actualizar EC2Launch o EC2Config. EC2Config se ejecuta desde Windows Server 2008 R2 a Windows Server 2012 R2. EC2Launch se ejecuta en Windows Server 2016.

Paso 4: updateSSMAgent (acción **`aws:runCommand`**)

Este paso actualiza SSM Agent con el manual de procedimientos `AWS-UpdateSSMAgent`.

Paso 5: updateAWSPVDriver (acción **`aws:runCommand`**)

Este paso actualiza controladores PV de AWS con el manual de procedimientos `AWS-ConfigureAWSPackage`.

Paso 6: updateAwsEnaNetworkDriver (acción **`aws:runCommand`**)

Este paso actualiza controladores de red ENA de AWS con el manual de procedimientos `AWS-ConfigureAWSPackage`.

Paso 7: installWindowsUpdates (acción **`aws:runCommand`**)

Este paso instala actualizaciones de Windows con el manual de procedimientos `AWS-InstallWindowsUpdates`. De forma predeterminada, Systems Manager busca e instala todas las actualizaciones que faltan. Puede cambiar el comportamiento predeterminado si especifica uno de los siguientes parámetros: `IncludeKbs`, `ExcludeKbs`, `Categories` o `SeverityLevels`.

Paso 8: runPostUpdateScript (acción **`aws:runCommand`**)

Este paso le permite especificar un script como una cadena que se ejecuta después de que se hayan instalado las actualizaciones.

Paso 9: runSysprepGeneralize (acción **`aws:runCommand`**)

En este paso, se utiliza el manual de procedimientos `AWS-InstallPowerShellModule` para descargar un módulo de PowerShell de AWS público. Systems Manager verifica la integridad del módulo con un hash SHA-256. A continuación, Systems Manager ejecuta sysprep mediante el uso de métodos admitidos por AWS para EC2Launch (Windows Server 2016) o EC2Config (Windows Server 2008 R2 a 2012 R2).

Paso 10: stopInstance (acción **`aws:changeInstanceState`**)

Este paso detiene la instancia actualizada.

Paso 11: createImage (acción **`aws:createImage`**)

Este paso crea una AMI con un nombre descriptivo que se enlaza con ID de origen y la hora de creación. Por ejemplo: “AMI generada con EC2 Automation a la(s) {{global:DATE_TIME}} a partir de {{SourceAmiId}}” donde DATE_TIME y SourceID representan variables de Automation.

Paso 12: TerminateInstance (acción **`aws:changeInstanceState`**)

Este paso limpia la automatización con la terminación de la instancia en ejecución.

Output

Esta sección le permite designar las salidas de diversos pasos o los valores de cualquier parámetro como la salida de Automation. De forma predeterminada, la salida es el ID de la AMI de Windows actualizada que se creó con la automatización.

> **Note**
>
> De forma predeterminada, cuando Automation ejecuta el manual de procedimientos `AWS-UpdateWindowsAmi` y crea una instancia temporal, el sistema usa la VPC predeterminada (172.30.0.0/16). Si ha eliminado la VPC predeterminada, recibirá el siguiente error:
> VPC not defined 400
> Para solucionar este problema, debe generar una copia del manual de procedimientos `AWS-UpdateWindowsAmi` y especificar un ID de subred. Para obtener más información, consulte VPC not defined 400.

Para crear una AMI de Windows con revisiones mediante Automation

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Ejecute el siguiente comando para ejecutar el manual de procedimientos `AWS-UpdateWindowsAmi`. Reemplace cada *example resource placeholder* con su propia información. El comando de ejemplo que aparece a continuación utiliza una AMI de Amazon EC2 reciente para minimizar el número de revisiones que es necesario aplicar. Si ejecuta este comando más de una vez, debe especificar un valor único para `targetAMIname`. Los nombres para las AMI deben ser únicos.

   ```
   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'
   ```

   El comando devuelve un ID de ejecución. Copie este ID en el portapapeles. Utilizará este ID para ver el estado de la automatización.

   ```
   {
       "AutomationExecutionId": "automation execution ID"
   }
   ```

3. Ejecute el siguiente comando para ver la automatización con la AWS CLI:

   ```
   aws ssm describe-automation-executions
   ```

4. Ejecute el siguiente comando para ver detalles acerca del progreso de la automatización.

   ```
   aws ssm get-automation-execution
       --automation-execution-id automation execution ID
   ```

> **Note**
>
> En función del número de revisiones aplicados, el proceso de aplicación de revisiones de Windows que se ejecuta en esta automatización de muestra puede tardar 30 minutos o más en completarse.

Actualice un golden AMI mediante la Automation, AWS Lambda, y Parameter Store

El siguiente ejemplo utiliza el modelo en el que una organización mantiene sus propias AMIs propietarias y les aplica revisiones de forma periódica, en lugar de basarse en las AMIs de Amazon Elastic Compute Cloud (Amazon EC2).

En el siguiente procedimiento se muestra cómo aplicar de forma automática las revisiones de sistema operativo (SO) a una AMI que ya se considera la AMI más actualizada o la más reciente. En el ejemplo, el valor predeterminado, el parámetro `SourceAmiId` se define mediante un parámetro de AWS Systems Manager Parameter Store que se denomina `latestAmi`. El valor de `latestAmi` se actualiza mediante una función de AWS Lambda invocada al final de la automatización. Como resultado de este proceso de Automatización, se reducen el tiempo y el esfuerzo empleados en la aplicación de revisiones a las AMIs, ya que las revisiones siempre se aplican a la AMI más actualizada. Parameter Store y Automatización son herramientas de AWS Systems Manager.

Antes de empezar

Configure los roles de Automation y, si así lo desea, Amazon EventBridge para Automation. Para obtener más información, consulte Configuración de Automation.

Contenido

- Tarea 1: crear un parámetro en Systems Manager Parameter Store
- Tarea 2: crear un rol de IAM para AWS Lambda
- Tarea 3: crear una función de AWS Lambda
- Tarea 4: crear un manual de procedimientos y aplicar revisiones a la AMI

Tarea 1: crear un parámetro en Systems Manager Parameter Store

Cree un parámetro de cadena en Parameter Store que utilice la siguiente información:

- Name (Nombre): `latestAmi`.

- Valor: un ID de AMI. Por ejemplo: `ami-188d6e0e`.

Para obtener información sobre cómo crear un parámetro de cadena de Parameter Store, consulte Creación de parámetros de Parameter Store en Systems Manager.

Tarea 2: crear un rol de IAM para AWS Lambda

Utilice el siguiente procedimiento para crear un rol de servicio de IAM para AWS Lambda. Estas políticas conceden permiso a Lambda para que actualice el valor del parámetro `latestAmi` con una función de Lambda y Systems Manager.

Para crear un rol de servicio de IAM para Lambda

1. Inicie sesión en AWS Management Console y abra la consola IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, seleccione Políticas y, a continuación, Crear política.
3. Seleccione la pestaña JSON.
4. Reemplace el contenido predeterminado por la siguiente política. Reemplace cada *example resource placeholder* por su propia información.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "logs:CreateLogGroup",
            "Resource": "arn:aws:logs:us-east-1:111122223333:*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": [
                "arn:aws:logs:us-east-1:111122223333:log-group:/aws/
lambda/function name:*"
            ]
```

```
        }
    ]
}
```

5. Elija Siguiente: etiquetas.
6. (Opcional) Agregue uno o varios pares de valor etiqueta-clave para organizar, realizar un seguimiento o controlar el acceso a esta política.
7. Elija Siguiente: Revisar.
8. En la página Review Policy (Revisar política), en Name (Nombre), escriba un nombre para la política insertada, como **amiLambda**.
9. Seleccione Crear política.
10. Repita los pasos 2 y 3.
11. Pegue la política siguiente. Reemplace cada *example resource placeholder* por su propia información.

    JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ssm:PutParameter",
            "Resource": "arn:aws:ssm:us-
east-1:111122223333:parameter/latestAmi"
        },
        {
            "Effect": "Allow",
            "Action": "ssm:DescribeParameters",
            "Resource": "*"
        }
    ]
}
```

12. Elija Siguiente: etiquetas.
13. (Opcional) Agregue uno o varios pares de valor etiqueta-clave para organizar, realizar un seguimiento o controlar el acceso a esta política.
14. Elija Siguiente: Revisar.

15. En la página Review Policy (Revisar política), en Name (Nombre), escriba un nombre para la política insertada, como **`amiParameter`**.
16. Elija Crear política.
17. En el panel de navegación, seleccione Roles y luego seleccione Crear rol.
18. Inmediatamente debajo de Caso de uso, seleccione Lambda y, a continuación, Siguiente.
19. En la página Agregar permisos, utilice el campo Buscar para localizar las dos políticas que ha creado anteriormente.
20. Elija la casilla de verificación situada junto a las políticas y, a continuación, elija Siguiente.
21. En Role name (Nombre del rol), escriba un nombre para el rol nuevo (por ejemplo, **`lambda-ssm-role`** o el nombre que prefiera).

    > **Note**
    >
    > Dado que varias entidades pueden hacer referencia al rol, no puede cambiar el nombre del rol después de crearla.

22. (Opcional) Agregue uno o varios pares clave-valor de etiqueta para organizar o controlar el acceso a este rol o realizar su seguimiento y, a continuación, elija Crear rol.

Tarea 3: crear una función de AWS Lambda

Utilice el siguiente procedimiento para crear una función de Lambda que actualice automáticamente el valor del parámetro `latestAmi`.

Cómo crear una función de Lambda

1. Inicie sesión en la AWS Management Console y abra la consola AWS Lambda en https://console.aws.amazon.com/lambda/.
2. Seleccione Creación de función.
3. En la página Crear función, elija Diseñar desde cero.
4. En Nombre de la función, introduzca **`Automation-UpdateSsmParam`**.
5. En Tiempo de ejecución, seleccione Python 3.8.
6. En Arquitectura, seleccione el tipo de procesador de equipo que Lambda utilizará para ejecutar la función, x86_64 o arm64,
7. En la sección Permisos, expanda Cambiar rol de ejecución predeterminado.

8. Elija Use an existing role (Usar un rol existente) y, a continuación, elija el rol de servicio para Lambda que creó en la tarea 2.
9. Seleccione Creación de función.
10. En la sección Origen de código de la pestaña lambda_function, elimine el código existente en el campo y, a continuación, pegue el siguiente código de ejemplo.

```
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. Elija Archivo, Guardar.
12. Para probar la función de Lambda, en el menú Prueba, elija Configurar eventos de prueba.
13. En Event Name (Nombre del evento), escriba un nombre para el evento de prueba, como **`MyTestEvent`**.
14. Reemplace el texto existente por el código JSON siguiente. Reemplace *`AMI ID`* con su propia información para establecer el valor del parámetro `latestAmi`.

```
{
   "parameterName":"latestAmi",
   "parameterValue":"AMI ID"
}
```

15. Seleccione Save.
16. Elija Test (Probar) para probar la función. En la pestaña Resultado de la ejecución, el estado debe indicarse como Correcto, junto con otros detalles sobre la actualización.

Tarea 4: crear un manual de procedimientos y aplicar revisiones a la AMI

Utilice el siguiente procedimiento para crear y ejecutar un manual de procedimientos que aplique revisiones a la AMI que ha especificado en el parámetro latestAmi. Después de que se complete la

automatización, el valor de latestAmi se actualiza con el ID de la AMI a la cual se acaba de aplicar revisiones. Las automatizaciones posteriores utilizan la AMI creada con la ejecución anterior.

Para crear y ejecutar el manual de procedimientos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. Para Crear documento, seleccione Automatización.
4. En Nombre, escriba **`UpdateMyLatestWindowsAmi`**.
5. Elija la pestaña Editor y después elija Edit (Editar).
6. Elija Aceptar cuando se le solicite.
7. En el campo Editor de documentos, reemplace el contenido predeterminado con el siguiente manual de procedimientos YAML de muestra.

```
---
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. Elija Create automation (Crear automatización).
9. En el panel de navegación, elija Automatización y, después, seleccione Ejecutar automatización.
10. En la página Choose document (Elegir documento), seleccione la pestaña Owned by me (De mi propiedad).
11. Busque el manual de procedimientos UpdateMyLatestWindowsAmi y seleccione el botón en la tarjeta UpdateMyLatestWindowsAmi.
12. Elija Siguiente.
13. Elija Simple execution (Ejecución sencilla).
14. Especifique los valores de los parámetros de entrada.
15. Elija Ejecutar.
16. Una vez que se complete la automatización, elija Parameter Store en el panel de navegación y confirme que el nuevo valor de `latestAmi` coincide con el valor que devuelve la automatización. También puede verificar que el nuevo ID de AMI coincida con la salida de Automation en la sección AMIs de la consola de Amazon EC2.

Actualización de las AMIs mediante Automatización y Jenkins

Si su organización utiliza el software Jenkins en una canalización de CI/CD, puede agregar Automatización como un paso posterior a la compilación para preinstalar las versiones de las aplicaciones en las Amazon Machine Images (AMIs). Automatización es una herramienta de AWS

Systems Manager. También puede utilizar la característica de programación de Jenkins para llamar a Automatización y crear su propia cadencia de revisiones de sistema operativo (SO).

En el siguiente ejemplo, se muestra cómo invocar a Automatización desde un servidor Jenkins que se ejecuta en las instalaciones o en Amazon Elastic Compute Cloud (Amazon EC2). Para la autenticación, el servidor Jenkins utiliza las credenciales AWS basadas en una política de (IAM) que usted crea en el ejemplo y adjunta al perfil de instancia.

> **Note**
>
> Asegúrese de seguir las prácticas recomendadas de seguridad Jenkins al configurar la instancia.

Antes de empezar

Complete las siguientes tareas antes de configurar Automatización con Jenkins:

- Complete el ejemplo Actualice un golden AMI mediante la Automation, AWS Lambda, y Parameter Store. En el siguiente ejemplo, se utiliza el manual de procedimientos UpdateMyLatestWindowsAmi creado en ese ejemplo.
- Configure los roles de IAM para Automation. Systems Manager requiere un rol de perfil de instancia y un ARN de rol de servicio para procesar automatizaciones. Para obtener más información, consulte Configuración de Automation.

Para crear una política de IAM para el servidor Jenkins

1. Inicie sesión en AWS Management Console y abra la consola IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, seleccione Políticas y, a continuación, Crear política.
3. Seleccione la pestaña JSON.
4. Reemplace cada *example resource placeholder* con su propia información.

   JSON

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

```
        {
            "Effect": "Allow",
            "Action": "ssm:StartAutomationExecution",
            "Resource": [
                "arn:aws:ssm:us-
east-1:111122223333:document/UpdateMyLatestWindowsAmi",
                "arn:aws:ssm:us-east-1:111122223333:automation-
definition/UpdateMyLatestWindowsAmi:$DEFAULT"
            ]
        }
    ]
}
```

5. Elija Revisar política.
6. En la página Review Policy (Revisar política), en Name (Nombre), escriba un nombre para la política insertada, como **`JenkinsPolicy`**.
7. Seleccione Crear política.
8. Seleccione Roles en el panel de navegación.
9. Elija el perfil de instancia que está asociado a su servidor Jenkins.
10. En la pestaña Permisos, elija Agregar permisos y, a continuación, Adjuntar políticas.
11. En la sección Otras políticas de permisos, ingrese el nombre de la política que ha creado en los pasos anteriores. Por ejemplo, JenkinsPolicy.
12. Marque la casilla de verificación situada junto a la política y, a continuación, elija Adjuntar políticas.

Utilice el siguiente procedimiento para configurar AWS CLI en su servidor Jenkins.

Configuración del servidor Jenkins para Automatización

1. Conéctese a su servidor Jenkins en el puerto 8080 con su navegador preferido para acceder a la interfaz de administración.
2. Ingrese la contraseña que se encuentra en `/var/lib/jenkins/secrets/initialAdminPassword`. Para mostrar la contraseña, ejecute el comando siguiente.

```
sudo cat /var/lib/jenkins/secrets/initialAdminPassword
```

3. El script de instalación Jenkins lo dirige a la página Personalizar Jenkins. Seleccione Install suggested plugins (Instalar complementos sugeridos).

4. Una vez que se complete la instalación, elija Credenciales de Administrador, seleccione Guardar credenciales y, a continuación, seleccione Empezar a usarJenkins.
5. En el panel de navegación de la izquierda, elija AdministrarJenkins y, a continuación, elija Administrar complementos.
6. Seleccione la pestaña Available (Disponible) y, a continuación, ingrese **`Amazon EC2 plugin`**.
7. Seleccione la casilla de verificación para **`Amazon EC2 plugin`** y, a continuación, Install without restart (Instalar sin reiniciar).
8. Una vez que se complete la instalación, seleccione Go back to the top page (Volver a la página superior).
9. Elija Administrar Jenkins y, a continuación, seleccione Administrar nodos y nubes.
10. En la sección Configurar nubes, seleccione Agregar una nube nueva y, a continuación, elija Amazon EC2.
11. Ingrese su información en los campos restantes. Asegúrese de seleccionar la opción Usar perfil de instancia de EC2 para obtener credenciales.

Utilice el siguiente procedimiento a fin de configurar su proyecto de Jenkins para invocar a Automatización.

Configuración del servidor Jenkins para invocar a Automatización

1. Abra la consola Jenkins en un navegador web.
2. Elija el proyecto que desee configurar con Automation y, a continuación, elija Configure.
3. En la pestaña Build, elija Add Build Step.
4. Elija Execute shell o Execute Windows batch command (en función de su sistema operativo).
5. En el campo Command (Comando), ejecute un comando de la AWS CLI como el siguiente. Reemplace cada *`example resource placeholder`* con su propia información.

```
aws ssm start-automation-execution \
        --document-name runbook name \
        --region Región de AWS of your source AMI \
        --parameters runbook parameters
```

El siguiente comando de ejemplo usa el manual de procedimientos UpdateMyLatestWindowsAmi y el parámetro `latestAmi` de Systems Manager creado en Actualice un golden AMI mediante la Automation, AWS Lambda, y Parameter Store.

```
aws ssm start-automation-execution \
        --document-name UpdateMyLatestWindowsAmi \
        --parameters \
            "sourceAMIid='{{ssm:latestAmi}}'" 
        --region region
```

En Jenkins, el comando se parece al ejemplo de la siguiente captura de pantalla.



6. En el proyecto Jenkins, seleccione Construir ahora. Jenkins devuelve un resultado similar al del siguiente ejemplo.



Actualización de AMIs para grupos de escalado automático

En el ejemplo siguiente se actualiza un grupo de escalado automático con una AMI a la que recién se ha aplicado revisiones. Este enfoque garantiza que las nuevas imágenes se pongan automáticamente a disposición de los entornos informáticos que utilizan grupos de escalado automático

El paso final de la automatización en este ejemplo utiliza una función de Python para crear una nueva plantilla de lanzamiento que utiliza la AMI a la que recién se ha aplicado revisiones. A

continuación, el grupo de escalado automático se actualiza para utilizar la nueva plantilla de lanzamiento. En este tipo de escenario de escalado automático, los usuarios podrían terminar las instancias existentes en el grupo de escalado automático para forzar el lanzamiento de una instancia nueva que utilice la nueva imagen. O los usuarios podrían esperar y permitir que los eventos de escala vertical y horizontal lancen instancias más recientes de forma natural.

Antes de empezar

Complete las tareas siguientes antes de comenzar este ejemplo.

- Configure los roles de IAM para Automatización, una herramienta de AWS Systems Manager. Systems Manager requiere un rol de perfil de instancia y un ARN de rol de servicio para procesar automatizaciones. Para obtener más información, consulte Configuración de Automation.

Crear el manual de procedimientos PatchAMIAndUpdateASG

Utilice el siguiente procedimiento para crear el manual de procedimientos PatchAMIAndUpdateASG que aplica revisiones a la AMI que especifica para el parámetro SourceAMI. El manual de procedimientos también actualiza un grupo de escalado automático para utilizar la última AMI a la que se ha aplicado revisiones.

Para crear y ejecutar el manual de procedimientos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. En el menú desplegable Create document (Crear documento), seleccione Automation.
4. En el campo Nombre, escriba **`PatchAMIAndUpdateASG`**.
5. Seleccione la pestaña Editor (Editor) y elija Edit (Editar).
6. Elija OK (Aceptar) cuando se le solicite y elimine el contenido en el campo Document editor (Editor de documentos).
7. En el campo Document editor (Editor de documentos), pegue el siguiente contenido del manual de procedimientos YAML de muestra.

```
---
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. Elija Create automation (Crear automatización).

9. En el panel de navegación, elija Automatización y, después, seleccione Ejecutar automatización.
10. En la página Choose document (Elegir documento), seleccione la pestaña Owned by me (De mi propiedad).
11. Busque el manual de procedimientos PatchAMIAndUpdateASG y seleccione el botón de la tarjeta PatchAMIAndUpdateASG.
12. Elija Siguiente.
13. Elija Simple execution (Ejecución sencilla).
14. Especifique los valores de los parámetros de entrada. Asegúrese de que el `SubnetId` y `SecurityGroupIds` que especifica permitan el acceso a los puntos de conexión públicos de Systems Manager o a los puntos de conexión de la interfaz para Systems Manager.
15. Elija Ejecutar.
16. Una vez finalizada la automatización, en la consola de Amazon EC2, elija Auto Scaling y, a continuación, Launch Templates (Plantillas de lanzamiento). Verifique que ve la nueva plantilla de lanzamiento y que utiliza la nueva AMI.
17. Seleccione Auto Scaling y, a continuación, Auto Scaling Groups (grupo de escalado automático). Verifique que el grupo de escalado automático utiliza la nueva plantilla de lanzamiento.
18. Termine una o más instancias de su grupo de escalado automático. Las instancias de reemplazo se lanzarán con la nueva AMI.

Uso de los manuales de procedimientos de autoservicio de AWS Support

En esta sección, se describe cómo utilizar algunas de las automatizaciones de autoservicio que creó el equipo de AWS Support. Estas automatizaciones lo ayudan a administrar sus recursos de AWS.

Flujos de trabajo de automatización de Support

Los flujos de trabajo de automatización de Support (SAW, Support Automation Workflow) son manuales de procedimientos de automatización que escribe y mantiene el equipo de AWS Support. Estos manuales de procedimientos lo ayudan a solucionar problemas comunes en sus recursos de AWS, monitorear e identificar de forma proactiva los problemas de red, recopilar y analizar registros, entre muchas otras actividades.

Los unbooks SAW utilizan el prefijo **`AWSSupport`**. Por ejemplo, `AWSSupport-ActivateWindowsWithAmazonLicense`.

Además, los clientes de Enterprise y Business Support de AWS también tienen acceso a los manuales de procedimientos que utilizan el prefijo **`AWSPremiumSupport`**. Por ejemplo, `AWSPremiumSupport-TroubleshootEC2DiskUsage`.

Para obtener más información acerca de AWS Support, consulte Introducción a AWS Support.

Temas

- Ejecutar la herramienta EC2Rescue en instancias inaccesibles
- Restablecimiento de contraseñas y claves de SSH en instancias EC2

Ejecutar la herramienta EC2Rescue en instancias inaccesibles

EC2Rescue puede ayudarlo a diagnosticar y resolver problemas de las instancias de Amazon Elastic Compute Cloud (Amazon EC2) para Linux y Windows Server. Puede ejecutar la herramienta manualmente, tal y como se describe en Uso de EC2Rescue para Linux Server y Uso de EC2Rescue para Windows Server. También puede ejecutar la herramienta de forma automática a través de Automatización de Systems Manager y el manual de procedimientos **`AWSSupport-ExecuteEC2Rescue`**. Automatización es una herramienta de AWS Systems Manager. El manual de procedimientos **`AWSSupport-ExecuteEC2Rescue`** está diseñado para realizar una combinación de acciones de Systems Manager, acciones de AWS CloudFormation y funciones de Lambda que automatizan los pasos que normalmente se necesitan para utilizar EC2Rescue.

Puede utilizar el manual de procedimientos **`AWSSupport-ExecuteEC2Rescue`** para solucionar diferentes tipos de problemas del sistema operativo (OS). No se admiten instancias con volúmenes raíz cifrados. Consulte los temas siguientes para obtener una lista completa:

Windows: consulte Acción de rescate en Uso de EC2Rescue for Windows Server con la línea de comando.

Linux y macOS: algunos módulos de EC2Rescue para Linux detectan problemas e intentan remediarlos. Para obtener más información, consulte la documentación `aws-ec2rescue-linux` de cada módulo en GitHub.

Funcionamiento

La solución de problemas de una instancia con Automation y el manual de procedimientos **`AWSSupport-ExecuteEC2Rescue`** funciona de la siguiente manera:

- Usted especifica el ID de la instancia inaccesible y activa el manual de procedimientos.

- El sistema crea una VPC temporal y, a continuación, ejecuta una serie de funciones de Lambda para configurar la VPC.
- El sistema identifica una subred para su VPC temporal en la misma zona de disponibilidad que la instancia original.
- El sistema lanza una instancia auxiliar de habilitada para SSM y temporal.
- El sistema detiene la instancia original y crea una copia de seguridad. A continuación, asocia el volumen raíz original a la instancia auxiliar.
- El sistema utiliza Run Command para ejecutar EC2Rescue en la instancia auxiliar. EC2Rescue identifica e intenta corregir los problemas en el volumen raíz original asociado. Cuando finaliza, EC2Rescue vuelve a asociar el volumen raíz a la instancia original.
- El sistema reinicia la instancia original y termina la instancia temporal. El sistema también termina la VPC temporal y las funciones de Lambda creadas al inicio de la Automation.

Antes de empezar

Antes de ejecutar la siguiente Automation, haga lo siguiente:

- Copie el ID de instancia de la instancia inaccesible. Especificará este ID en el procedimiento.
- Opcionalmente, recopile el ID de una subred en la misma zona de disponibilidad que su instancia inaccesible. La instancia de EC2Rescue se creará en esta subred. Si no especifica ninguna subred, Automation crea una nueva VPC temporal en su Cuenta de AWS. Verifique que su Cuenta de AWS tiene al menos una VPC disponible. De forma predeterminada, puede crear cinco VPC en una región. Si ya ha creado cinco VPC en la región, se produce un error en la Automation sin realizar cambios en la instancia. Para obtener más información acerca de las cuotas de Amazon VPC, consulte VPC y subredes en la Guía del usuario de Amazon VPC.
- También puede crear y especificar un rol de AWS Identity and Access Management (IAM) para Automation. Si no especifica este rol, Automation se ejecuta en el contexto del usuario que ha ejecutado la Automation.

Concesión de permisos **`AWSSupport-EC2Rescue`** para realizar acciones en las instancias

EC2Rescue necesita permiso para realizar una serie de acciones en las instancias durante la automatización. Estas acciones invocan los servicios AWS Lambda, IAM y Amazon EC2 para intentar solucionar los problemas de sus instancias de forma segura. Si dispone de permisos de nivel de administrador en su Cuenta de AWS o VPC, es probable que pueda ejecutar la automatización sin configurar permisos, tal y como se describe en esta sección. Si no dispone de permisos de nivel

de administrador, usted o un administrador deben configurarlos mediante una de las siguientes opciones.

- Concesión de permisos mediante políticas de IAM
- Concesión de permisos mediante una plantilla de AWS CloudFormation

Concesión de permisos mediante políticas de IAM

Puede adjuntar la siguiente política de IAM a su cuenta de usuario, grupo o rol como una política insertada o bien puede crear una nueva política administrada de IAM y adjuntarla a su usuario, grupo o rol. Para obtener más información sobre la adición de una política insertada a su usuario, grupo o rol, consulte Uso de políticas insertadas. Para obtener más información sobre cómo crear una política administrada, consulte Uso de políticas administradas.

> **Note**
>
> Si crea una nueva política administrada de IAM, también debe adjuntar la política administrada AmazonSSMAutomationRole para que la instancia se pueda comunicar con la API de Systems Manager.

Política de IAM para AWSSupport-EC2Rescue

Reemplace *account ID* con su propia información.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "lambda:InvokeFunction",
                "lambda:DeleteFunction",
                "lambda:GetFunction"
            ],
            "Resource": "arn:aws:lambda:*:111122223333: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::111122223333:role/AWSSupport-EC2Rescue-*",
                "arn:aws:iam::111122223333: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*",
                "autoscaling:DescribeAutoScalingInstances"
            ],
            "Resource": "*",
            "Effect": "Allow"
        }
    ]
}
```

Concesión de permisos mediante una plantilla de AWS CloudFormation

AWS CloudFormation automatiza el proceso de creación de roles y políticas de IAM a través de una plantilla preconfigurada. Utilice el siguiente procedimiento para crear los roles y las políticas de IAM necesarios para EC2Rescue Automation mediante AWS CloudFormation.

Para crear los roles y las políticas de IAM necesarios para EC2Rescue

1. Descargue AWSSupport-EC2RescueRole.zip y extraiga el archivo `AWSSupport-EC2RescueRole.json` en un directorio de su equipo local.
2. Si su Cuenta de AWS está en una partición especial, edite la plantilla para cambiar los valores de ARN por los de su partición.

   Por ejemplo, para las regiones de China, cambie todas las instancias de `arn:aws` por `arn:aws-cn`.
3. Inicie sesión en la AWS Management Console y abra la consola de AWS CloudFormation en https://console.aws.amazon.com/cloudformation.
4. Elija Create stack (Crear pila), With new resources (standard) (Con nuevos recursos [estándar]).

5. En la página Create stack (Crear pila) en Prerequisite - Prepare template (Requisito previo: preparar plantilla), elija Template is ready (La plantilla está lista).
6. En Specify template (Especificar plantilla), elija Upload a template file (Cargar un archivo de plantilla).
7. Elija Choose file (Elegir archivo), y, a continuación, busque y seleccione el archivo `AWSSupport-EC2RescueRole.json` del directorio en el que lo extrajo.
8. Elija Siguiente.
9. En la página Specify stack details (Especificar detalles de la pila) para el campo Stack name (Nombre de la pila), escriba un nombre para identificar esta pila y, a continuación, elija Next (Siguiente).
10. (Opcional) En el área Tags (Etiquetas), aplique a la pila uno o varios pares de nombre-valor de claves de etiqueta.

    Las etiquetas son metadatos opcionales que usted asigna a un recurso. Las etiquetas le permiten clasificar los recursos de diversas maneras, por ejemplo, según su finalidad, propietario o entorno. Por ejemplo, puede que desee etiquetar una pila para identificar el tipo de tareas que ejecuta, los tipos de destinos u otros recursos implicados y el entorno en el que se ejecuta.
11. Elegir Next (Siguiente)
12. En la página Review (Revisar), revise los detalles de la pila y, a continuación, desplácese hacia abajo y elija la opción I acknowledge that AWS CloudFormation might create IAM resources (Acepto que es posible que cree recursos de IAM).
13. Seleccione Creación de pila.

    AWS CloudFormation muestra el estado CREATE_IN_PROGRESS durante unos minutos. El estado cambia a CREATE_COMPLETE tras crear la pila. También puede elegir el icono de actualización para comprobar el estado del proceso de creación.
14. En la lista Stacks (Pilas), seleccione el botón situado junto a la pila que acaba de crear y, a continuación, elija la pestaña Outputs (Salidas).
15. Anote el valor de Value (Valor). Es el ARN de AssumeRole. Especifique este ARN cuando ejecute la automatización en el siguiente procedimiento, Ejecución de Automation.

Ejecución de Automation

> **Important**
>
> La siguiente automatización detiene la instancia inaccesible. La detención de la instancia puede ocasionar la pérdida de datos en los volúmenes de almacén de instancias asociados (si los hubiera). La detención de la instancia también puede provocar que cambie la IP pública, si no hay asociada ninguna IP elástica.

Para ejecutar el manual de procedimientos de Automation **`AWSSupport-ExecuteEC2Rescue`**

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija automatización.
3. Elija Ejecutar automatización.
4. En la sección Automation document (Documento de Automation), elija Owned by Amazon (Propiedad de Amazon) en la lista.
5. En la lista de manuales de procedimientos, seleccione el botón de la tarjeta que corresponde a **`AWSSupport-ExecuteEC2Rescue`** y, a continuación, elija Next (Siguiente).
6. En la página Execute automation document (Ejecutar documento de Automation), elija Simple execution (Ejecución sencilla).
7. En la sección Document details (Detalles del documento), asegúrese de que el valor de Document version (Versión del documento) sea la versión predeterminada más alta. Por ejemplo, $DEFAULT o 3 (default) (3 [predeterminada]).
8. En la sección Input parameters, especifique los siguientes parámetros:

    a. En UnreachableInstanceId, especifique la ID de la instancia inaccesible.

    b. (Opcional) En EC2RescueInstanceType, especifique un tipo de instancia para la instancia de EC2Rescue. El tipo de instancia predeterminada es `t2.medium`.

    c. En AutomationAssumeRole, si ha creado roles para esta automatización mediante el procedimiento de AWS CloudFormation descrito anteriormente en este tema, elija el ARN del AssumeRole que creó en la consola de AWS CloudFormation.

    d. (Opcional) En LogDestination, especifique un bucket de S3 si desea recopilar registros de nivel del sistema operativo mientras soluciona los problemas de la instancia. Los registros se cargan automáticamente en el bucket especificado.

e. En SubnetId, especifique una subred en una VPC existente en la misma zona de disponibilidad que la instancia inaccesible. De forma predeterminada, Systems Manager crea una nueva VPC, pero usted puede especificar una subred en una VPC existente, si así lo desea.

> **Note**
>
> Si no ve la opción para especificar un ID de subred o bucket, verifique que está usando la versión más reciente predeterminada del manual de procedimientos.

9. (Opcional) En el área Tags (Etiquetas), aplique uno o más pares de nombre-valor de claves de etiqueta para ayudar a identificar la automatización, como `Key=Purpose,Value=EC2Rescue`.
10. Elija Ejecutar.

El manual de procedimientos crea una AMI de copia de seguridad como parte de la automatización. Los demás recursos creados por la automatización se eliminan automáticamente, pero esta AMI permanece en su cuenta. La AMI recibe su nombre siguiendo esta convención:

AMI de copia de seguridad: AWSSupport-EC2Rescue:*`UnreachableInstanceId`*

Puede localizar esta AMI en la consola de Amazon EC2 mediante la búsqueda del ID de ejecución de Automation.

Restablecimiento de contraseñas y claves de SSH en instancias EC2

Puede utilizar el manual de procedimientos `AWSSupport-ResetAccess` para volver a activar de forma automática la generación de contraseñas del administrador local en las instancias de Amazon Elastic Compute Cloud (Amazon EC2) para Windows Server y generar una nueva clave de SSH en instancias de EC2 para Linux. El manual de procedimientos `AWSSupport-ResetAccess` se ha diseñado para realizar una combinación de acciones de AWS Systems Manager, acciones de AWS CloudFormation y funciones de AWS Lambda que automatizan los pasos que normalmente se requieren para restablecer la contraseña de administrador local.

Puede utilizar Automatización, una herramienta de AWS Systems Manager, con el manual de procedimientos `AWSSupport-ResetAccess` para solucionar los siguientes problemas:

Windows

Ha perdido el par de claves de EC2: si desea resolver este problema, puede utilizar el manual de procedimientos AWSSupport-ResetAccess para crear una AMI que se habilita con contraseñas con la instancia actual, lanzar una nueva instancia a partir de la AMI y seleccionar un par de claves de su propiedad.

Ha perdido la contraseña de administrador local: si desea solucionar este problema, puede utilizar el manual de procedimientos `AWSSupport-ResetAccess` para generar una nueva contraseña que se pueda descifrar con el par de claves de EC2 actual.

Linux

Ha perdido el par de claves de EC2 o configuró el acceso de SSH a la instancia con una clave que ha perdido: si desea solucionar este problema, puede utilizar el manual de procedimientos `AWSSupport-ResetAccess` a fin de crear una nueva clave de SSH para la instancia actual, lo que le permitirá conectarse de nuevo a la instancia.

> **Note**
>
> Si su instancia de EC2 para Windows Server está configurada para Systems Manager, también puede restablecer su contraseña de administrador local mediante EC2Rescue y AWS Systems Manager Run Command. Para obtener más información, consulte Uso de EC2Rescue para Windows Server con Systems Manager Run Command en la Guía del usuario de Amazon EC2.

Información relacionada

Conexión a una instancia de Linux desde Windows mediante PuTTY en la Guía del usuario de Amazon EC2

Funcionamiento

La solución de problemas de una instancia con Automation y el manual de procedimientos `AWSSupport-ResetAccess` funciona de la siguiente manera:

- Especifica el ID de la instancia y ejecuta el manual de procedimientos.
- El sistema crea una VPC temporal y, a continuación, ejecuta una serie de funciones de Lambda para configurar la VPC.

- El sistema identifica una subred para su VPC temporal en la misma zona de disponibilidad que la instancia original.
- El sistema lanza una instancia auxiliar de habilitada para SSM y temporal.
- El sistema detiene la instancia original y crea una copia de seguridad. A continuación, asocia el volumen raíz original a la instancia auxiliar.
- El sistema utiliza Run Command para ejecutar EC2Rescue en la instancia auxiliar. En Windows, EC2Rescue permite generar la contraseña para el administrador local usando EC2Config o EC2Launch en el volumen raíz original asociado. En Linux, EC2Rescue genera e introduce una nueva clave de SSH y guarda la clave privada (cifrada) en Parameter Store. Cuando finaliza, EC2Rescue vuelve a asociar el volumen raíz a la instancia original.
- El sistema crea una nueva Amazon Machine Image (AMI) de su instancia, ahora que la generación de contraseña está habilitada. Puede usar esta AMI para crear una nueva instancia de EC2 y asociar un par de claves nuevo, en caso de ser necesario.
- El sistema reinicia la instancia original y termina la instancia temporal. El sistema también termina la VPC temporal y las funciones de Lambda creadas al inicio de la Automation.
- Windows: la instancia genera una nueva contraseña que se puede descodificar en la consola de Amazon EC2 con el par de claves que la instancia tiene asignado en este momento.

  Linux: puede aplicar SSH a la instancia utilizando la clave de SSH almacenada en Systems Manager Parameter Store como /ec2rl/openssh/***instance ID***/key.

Antes de empezar

Antes de ejecutar la siguiente Automation, haga lo siguiente:

- Copie el ID de la instancia en la que desea restablecer la contraseña de administrador. Especificará este ID en el procedimiento.
- Opcionalmente, recopile el ID de una subred en la misma zona de disponibilidad que su instancia inaccesible. La instancia de EC2Rescue se creará en esta subred. Si no especifica ninguna subred, Automation crea una nueva VPC temporal en su Cuenta de AWS. Verifique que su Cuenta de AWS tiene al menos una VPC disponible. De forma predeterminada, puede crear cinco VPC en una región. Si ya ha creado cinco VPC en la región, se produce un error en la Automation sin realizar cambios en la instancia. Para obtener más información acerca de las cuotas de Amazon VPC, consulte VPC y subredes en la Guía del usuario de Amazon VPC.

- También puede crear y especificar un rol de AWS Identity and Access Management (IAM) para Automation. Si no especifica este rol, Automation se ejecuta en el contexto del usuario que ha ejecutado la Automation.

Concesión de permisos a AWSSupport-EC2Rescue para realizar acciones en las instancias

EC2Rescue necesita permiso para realizar una serie de acciones en las instancias durante la automatización. Estas acciones invocan los servicios AWS Lambda, IAM y Amazon EC2 para intentar solucionar los problemas de sus instancias de forma segura. Si dispone de permisos de nivel de administrador en su Cuenta de AWS o VPC, es probable que pueda ejecutar la automatización sin configurar permisos, tal y como se describe en esta sección. Si no dispone de permisos de nivel de administrador, usted o un administrador deben configurarlos mediante una de las siguientes opciones.

- Concesión de permisos mediante políticas de IAM
- Concesión de permisos mediante una plantilla de AWS CloudFormation

Concesión de permisos mediante políticas de IAM

Puede adjuntar la siguiente política de IAM a su cuenta de usuario, grupo o rol como una política insertada o bien puede crear una nueva política administrada de IAM y adjuntarla a su usuario, grupo o rol. Para obtener más información sobre la adición de una política insertada a su usuario, grupo o rol, consulte Uso de políticas insertadas. Para obtener más información sobre cómo crear una política administrada, consulte Uso de políticas administradas.

> **Note**
>
> Si crea una nueva política administrada de IAM, también debe adjuntar la política administrada AmazonSSMAutomationRole para que la instancia se pueda comunicar con la API de Systems Manager.

Política de IAM para **`AWSSupport-ResetAccess`**

Reemplace *`account ID`* con su propia información.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "lambda:InvokeFunction",
                "lambda:DeleteFunction",
                "lambda:GetFunction"
            ],
            "Resource": "arn:aws:lambda:*:111122223333: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::111122223333:role/AWSSupport-EC2Rescue-*",
                "arn:aws:iam::111122223333: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"
        }
    ]
}
```

Concesión de permisos mediante una plantilla de AWS CloudFormation

AWS CloudFormation automatiza el proceso de creación de roles y políticas de IAM a través de una plantilla preconfigurada. Utilice el siguiente procedimiento para crear los roles y las políticas de IAM necesarios para EC2Rescue Automation mediante AWS CloudFormation.

Para crear los roles y las políticas de IAM necesarios para EC2Rescue

1. Descargue AWSSupport-EC2RescueRole.zip y extraiga el archivo AWSSupport-EC2RescueRole.json en un directorio de su equipo local.
2. Si su Cuenta de AWS está en una partición especial, edite la plantilla para cambiar los valores de ARN por los de su partición.

   Por ejemplo, para las regiones de China, cambie todas las instancias de arn:aws por arn:aws-cn.
3. Inicie sesión en la AWS Management Console y abra la consola de AWS CloudFormation en https://console.aws.amazon.com/cloudformation.
4. Elija Create stack (Crear pila), With new resources (standard) (Con nuevos recursos [estándar]).
5. En la página Create stack (Crear pila) en Prerequisite - Prepare template (Requisito previo: preparar plantilla), elija Template is ready (La plantilla está lista).
6. En Specify template (Especificar plantilla), elija Upload a template file (Cargar un archivo de plantilla).
7. Elija Choose file (Elegir archivo), y, a continuación, busque y seleccione el archivo AWSSupport-EC2RescueRole.json del directorio en el que lo extrajo.
8. Elija Siguiente.
9. En la página Specify stack details (Especificar detalles de la pila) para el campo Stack name (Nombre de la pila), escriba un nombre para identificar esta pila y, a continuación, elija Next (Siguiente).
10. (Opcional) En el área Tags (Etiquetas), aplique a la pila uno o varios pares de nombre-valor de claves de etiqueta.

    Las etiquetas son metadatos opcionales que usted asigna a un recurso. Las etiquetas le permiten clasificar los recursos de diversas maneras, por ejemplo, según su finalidad, propietario o entorno. Por ejemplo, puede que desee etiquetar una pila para identificar el tipo de tareas que ejecuta, los tipos de destinos u otros recursos implicados y el entorno en el que se ejecuta.
11. Elegir Next (Siguiente)
12. En la página Review (Revisar), revise los detalles de la pila y, a continuación, desplácese hacia abajo y elija la opción I acknowledge that AWS CloudFormation might create IAM resources (Acepto que es posible que cree recursos de IAM).

13. AWS CloudFormation muestra el estado CREATE_IN_PROGRESS durante unos minutos. El estado cambia a CREATE_COMPLETE tras crear la pila. También puede elegir el icono de actualización para comprobar el estado del proceso de creación.
14. En la lista de pilas, elija la opción situada junto al pila que acaba de crear y, a continuación, elija la pestaña Outputs (Salidas).
15. Copie el contenido de Value (Valor). Es el ARN de AssumeRole. Especificará este ARN al ejecutar Automation.

Ejecución de Automation

El siguiente procedimiento describe cómo ejecutar el manual de procedimientos `AWSSupport-ResetAccess` a través de la consola de AWS Systems Manager.

> **Important**
>
> La siguiente automatización detiene la instancia. La detención de la instancia puede ocasionar la pérdida de datos en los volúmenes de almacén de instancias asociados (si los hubiera). La detención de la instancia también puede provocar que cambie la IP pública, si no hay asociada ninguna IP elástica. Para evitar estos cambios de configuración, use Run Command para restablecer el acceso. Para obtener más información, consulte Uso de EC2Rescue para Windows Server con Systems Manager Run Command en la Guía del usuario de Amazon EC2.

Para ejecutar el documento de AWSSupport-ResetAccess Automation

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija automatización.
3. Elija Ejecutar automatización.
4. En la sección Automation document (Documento de Automation), elija Owned by Amazon (Propiedad de Amazon) en la lista.
5. En la lista de manuales de procedimientos, seleccione el botón de la tarjeta que corresponde a AWSSupport-ResetAccess y, a continuación, elija Next (Siguiente).
6. En la página Execute automation document (Ejecutar documento de Automation), elija Simple execution (Ejecución sencilla).

7. En la sección Document details (Detalles del documento), asegúrese de que el valor de Document version (Versión del documento) sea la versión predeterminada más alta. Por ejemplo, $DEFAULT o 3 (default) (3 [predeterminada]).
8. En la sección Input parameters, especifique los siguientes parámetros:
   a. En InstanceID, especifique la ID de la instancia inaccesible.
   b. En SubnetId, especifique una subred en una VPC existente en la misma zona de disponibilidad que la instancia que haya indicado. De forma predeterminada, Systems Manager crea una nueva VPC, pero usted puede especificar una subred en una VPC existente, si así lo desea.

      > **Note**
      >
      > Si no ve la opción para especificar un ID de subred, verifique que está usando la versión más reciente predeterminada del manual de procedimientos.

   c. En EC2RescueInstanceType, especifique un tipo de instancia para la instancia de EC2Rescue, El tipo de instancia predeterminada es `t2.medium`.
   d. Para AssumeRole, si ha creado roles para esta automatización mediante el procedimiento de AWS CloudFormation descrito anteriormente en este tema, especifique el ARN de AssumeRole que anotó en la consola de AWS CloudFormation.
9. (Opcional) En el área Tags (Etiquetas), aplique uno o más pares de nombre-valor de claves de etiqueta para ayudar a identificar la automatización, como `Key=Purpose,Value=ResetAccess`.
10. Elija Ejecutar.
11. Para supervisar el progreso de la automatización, elija la automatización que se esté ejecutando, y luego la pestaña Steps (Pasos). Una vez que se haya completado la automatización, elija la pestaña Descriptions (Descripciones) y, a continuación, View output (Ver salida) para ver los resultados. Para ver la salida de pasos individuales, elija la pestaña Steps (Pasos) y, a continuación, View Outputs (Ver salidas) junto a un paso.

El manual de procedimientos crea una AMI de copia de seguridad y una AMI que se habilita con contraseñas como parte de la automatización. Los demás recursos que creó la automatización se eliminan automáticamente, pero estas AMIs permanecen en su cuenta. Las AMIs reciben su nombre según las convenciones siguientes:

- AMI de copia de seguridad: `AWSSupport-EC2Rescue:`*`InstanceID`*
- AMI habilitada con contraseña: AWSSupport-EC2Rescue: AMI habilitada con contraseña de *`Instance ID`*

Puede localizar estas AMIs buscando el ID de ejecución de Automation.

En Linux, la nueva clave privada de SSH para la instancia se guarda (cifrada) en Parameter Store. El nombre del parámetro es /ec2rl/openssh/***`instance ID`***/key.

Transferir datos a Automatización usando transformadores de entrada

Este tutorial de AWS Systems Manager Automation muestra cómo utilizar la característica de transformador de entrada de Amazon EventBridge para extraer el `instance-id` de una instancia de Amazon Elastic Compute Cloud (Amazon EC2) a partir de un evento de cambio de estado de instancia. Automatización es una herramienta de AWS Systems Manager. Utilizamos el transformador de entrada para transmitir esos datos al destino del manual de procedimientos `AWS-CreateImage` como el parámetro de entrada `InstanceId`. La regla se activa cuando alguna instancia cambia al estado `stopped`.

Para obtener más información acerca de cómo trabajar con transformadores de entrada, consulte Tutorial: utilizar el transformador de entrada para personalizar los datos que se transmiten al destino del evento en la Guía del usuario de Amazon EventBridge.

Antes de empezar

Compruebe que ha agregado los permisos y la política de confianza necesarios de EventBridge a su rol de servicio de Automatización de Systems Manager. Para obtener más información, consulte Información general sobre la administración de permisos de acceso a los recursos de EventBridge en la Guía del usuario de Amazon EventBridge.

Para utilizar transformadores de entrada con Automation

1. Abra la consola de Amazon EventBridge en https://console.aws.amazon.com/events/.
2. En el panel de navegación, seleccione Reglas.
3. Elija Creación de regla.
4. Escriba un nombre y una descripción para la regla.

   Una regla no puede tener el mismo nombre que otra regla de la misma región y del mismo bus de eventos.

5. En Bus de eventos, seleccione el bus de eventos que desea asociar a esta regla. Si desea que esta regla responda a eventos coincidentes procedentes de su propia Cuenta de AWS, seleccione default (predeterminado). Cuando un Servicio de AWS en su cuenta emite un evento, siempre va al bus de eventos predeterminado de su cuenta.
6. En Tipo de regla, elija Regla con un patrón de evento.
7. Elija Siguiente.
8. En Origen del evento, elija Eventos de AWS o eventos de socios de EventBridge.
9. En la sección Event pattern (Patrón de eventos), elija Event pattern form (Formulario de patrón de eventos).
10. En Event source (Origen del evento), elija AWS services (Servicios de).
11. En AWS service (Servicio de), elija EC2.
12. En Event Type (Tipo de evento), elija EC2 Instance State-change Notification (Notificación de cambio de estado de instancia de EC2).
13. En Specific state(s) (Estados específicos), elija stopped (detenido).
14. Elija Siguiente.
15. En Target types (Tipos de destino), elija AWS service.
16. Para Select a target (Seleccione un destino), elija Systems Manager Automation (Automatización de Systems Manager).
17. En Document (Documento), elija AWS-CreateImage.
18. Expanda la sección Configure automation parameter(s) (Configurar parámetros de automatización) y seleccione Input Transformer (Transformador de entrada).
19. En Input path (Ruta de entrada), escriba **`{"instance":"$.detail.instance-id"}`**.
20. En Template (Plantilla), escriba **`{"InstanceId":[<instance>]}`**.
21. En Execution role (Rol de ejecución), elija Use existing role (Usar rol existente) y elija su rol de servicio de Automation.
22. Elija Siguiente.
23. (Opcional) Introduzca una o varias etiquetas para la regla. Para obtener más información, consulte Etiquetado de los recursos de Amazon EventBridge en la Guía del usuario de Amazon EventBridge.
24. Elija Siguiente.
25. Revise los detalles de la regla y seleccione Creación de regla.

Obtenga información sobre los estados devueltos por la automatización de Systems Manager

AWS Systems Manager Automation brinda información detallada sobre los diferentes estados por los que pasa una acción o un paso de una automatización, y para la automatización general. Automatización es una herramienta de AWS Systems Manager. Puede monitorear los estados de la automatización con los siguientes métodos:

- Monitoree el estado de ejecución en la consola de Systems Manager Automation.
- Utilice sus herramientas de línea de comandos preferidas. Para la AWS Command Line Interface (AWS CLI), puede utilizar describe-automation-step-executions o get-automation-execution. Para las AWS Tools for Windows PowerShell, puede utilizar Get-SSMAutomationStepExecution o Get-SSMAutomationExecution.
- Configure Amazon EventBridge para responder a los cambios de estado de una acción o una automatización.

Para obtener más información sobre cómo gestionar los tiempos de espera de una automatización, consulte Administración de los tiempos de espera en los manuales de procedimientos.

Acerca de los estados de las automatizaciones

Automation brinda información detallada sobre el estado de las acciones de automatización individuales, además de la automatización general.

El estado general de la automatización puede ser diferente del estado notificado por una acción o un paso individuales, como se describe en las siguientes tablas.

Estado detallado de las acciones

Estado	Detalles
Pendiente	Todavía no se ha empezado a ejecutar el paso. Si la automatización utiliza acciones condicionales y no se cumplió la condición para ejecutar el paso, este permanece en este estado después de que se completa una automatización. Los pasos también permanecen en este

Estado	Detalles
	estado si la automatización se cancela antes de que se ejecute el paso.
InProgress	El paso se está ejecutando.
Waiting	El paso está esperando la entrada.
Success	El paso se ha completado correctamente. Se trata de un estado terminal.
TimedOut	No se completó un paso ni una aprobación antes del periodo de espera especificado. Se trata de un estado terminal.
Cancelling	El paso se encuentra en proceso de detención después de ser cancelado por un solicitante.
Cancelado	Un solicitante detuvo el paso antes de que se completara. Se trata de un estado terminal.
Con error	El paso no se completó correctamente. Se trata de un estado terminal.
Exited	Solo devuelto por la acción `aws:loop`. El bucle ciclo no se completó. Un paso dentro del bucle se movió a un paso exterior mediante las propiedades `nextStep`, `onCancel`, o `onFailure` .

Estado detallado de una automatización

Estado	Detalles
Pendiente	Todavía no se ha empezado a ejecutar la automatización.
InProgress	La automatización se está ejecutando.

Estado	Detalles
Waiting	La automatización está esperando la entrada.
Success	Se ha completado correctamente la automatiz ación. Se trata de un estado terminal.
TimedOut	No se completó un paso ni una aprobación antes del periodo de espera especificado. Se trata de un estado terminal.
Cancelling	La automatización se encuentra en proceso de detención después de ser cancelada por un solicitante.
Cancelado	Un solicitante detuvo la automatización antes de que se completara. Se trata de un estado terminal.
Con error	No se ha completado correctamente la automatización. Se trata de un estado terminal.

Solución de problemas de Automatización de Systems Manager

Utilice la siguiente información como ayuda para solucionar problemas con Automatización de AWS Systems Manager, una herramienta de AWS Systems Manager. En este tema se incluyen tareas específicas para resolver problemas basados en los mensajes de error de Automation.

Temas

- Errores de Automation comunes
- No se ha podido iniciar la ejecución de Automation
- Ejecución iniciada, pero el estado es Error
- La ejecución se ha iniciado, pero se ha agotado el tiempo de espera

Errores de Automation comunes

En esta sección se incluye información sobre los errores de Automation comunes.

VPC not defined 400

De forma predeterminada, cuando Automation ejecuta los manuales de procedimientos `AWS-UpdateLinuxAmi` o `AWS-UpdateWindowsAmi`, el sistema crea una instancia temporal en la VPC predeterminada (172.30.0.0/16). Si ha eliminado la VPC predeterminada, recibirá el siguiente error:

```
VPC not defined 400
```

Para resolver este problema, debe especificar un valor para el parámetro de entrada `SubnetId`.

No se ha podido iniciar la ejecución de Automation

Es posible que no se logre completar una automatización debido a un error de acceso denegado o un error de rol de asunción no válido si no ha configurado correctamente los roles y las políticas de AWS Identity and Access Management (IAM) para Automatización.

Acceso denegado

En los siguientes ejemplos, se describen situaciones en las que no se logra iniciar una automatización por un error de acceso denegado.

Acceso denegado a la API de Systems Manager

Mensaje de error: `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)`

- Posible causa 1: el usuario que intenta iniciar la automatización no tiene permiso para invocar la API `StartAutomationExecution`. Para resolver este problema, adjunte la política de IAM necesaria al usuario que se utilizó para iniciar la automatización.
- Posible causa 2: el usuario que intenta iniciar la automatización tiene permiso para invocar la API `StartAutomationExecution`, pero no para invocarla a través del manual de procedimientos específico. Para resolver este problema, adjunte la política de IAM necesaria al usuario que se utilizó para iniciar la automatización.

Acceso denegado debido a la falta de permisos PassRole

Mensaje de error: `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)`

El usuario que intenta iniciar la automatización no tiene el permiso PassRole para el rol de asunción. Para solucionar este problema, adjunte la política iam:PassRole al rol del usuario que intenta iniciar la automatización. Para obtener más información, consulte Tarea 2: asociar la política iam:PassRole al rol de Automation.

Rol de asunción no válido

Al ejecutar una automatización, se proporciona un rol de asunción en el manual de procedimientos o se transmite como un valor de parámetro al manual. Los diferentes tipos de errores se pueden producir si el rol de asunción no se especifica ni se configura correctamente.

Rol de asunción con formato incorrecto

Mensaje de error: `The format of the supplied assume role ARN isn't valid.` El rol de asunción tiene un formato incorrecto. Para solucionar este problema, verifique que se ha especificado un rol de asunción válido en el manual de procedimientos o como un parámetro de tiempo de ejecución al iniciar la automatización.

No se puede asumir el rol de asunción

Mensaje de error: `The defined assume role is unable to be assumed. (Service: AWSSimpleSystemsManagement; Status Code: 400; Error Code: InvalidAutomationExecutionParametersException; Request ID: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)`

- Posible causa 1: el rol de asunción no existe. Para resolver este problema, cree el rol. Para obtener más información, consulte the section called “Configuración de Automation”. Los detalles específicos para la creación de este rol se describen en el siguiente tema, Tarea 1: crear un rol de servicio para Automation.
- Posible causa 2: el rol de asunción no tiene una relación de confianza con el servicio Systems Manager. Para solucionar este problema, cree la relación de confianza. Para obtener más información, consulte No puedo asumir un rol en la Guía del usuario de IAM.

Ejecución iniciada, pero el estado es Error

Errores específicos de la acción

Los manuales de procedimientos contienen pasos que se ejecutan en orden. Cada paso invoca una o varias API de Servicio de AWS. Las API determinan las entradas, el comportamiento y las salidas del paso. Hay varios lugares en los que un error puede provocar que no se realice un paso. Los mensajes de error indican cuándo y dónde se ha producido un error.

Para ver un mensaje de error en la consola de Amazon Elastic Compute Cloud (Amazon EC2), elija el enlace View Outputs (Ver salidas) del paso con error. Para ver un mensaje de error desde la AWS CLI, llame a `get-automation-execution` y busque el atributo `FailureMessage` en un `StepExecution` con error.

En los siguientes ejemplos, se ha producido un error en un paso asociado a la acción `aws:runInstance`. Cada ejemplo explora un tipo de error distinto.

Falta la imagen

Mensaje de error: `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.`

La acción `aws:runInstances` ha recibido la acción de un `ImageId` que no existe. Para solucionar este problema, actualice el manual de procedimientos o los valores de parámetro con el ID de AMI correcto.

La política de rol de asunción no tiene permisos suficientes

Mensaje de error: `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: xxxxxxx (Service: AmazonEC2; Status Code: 403; Error Code: UnauthorizedOperation; Request ID: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)]. Please refer to Automation Service Troubleshooting Guide for more diagnosis details.`

El rol de asunción no tiene permisos suficientes para invocar la API `RunInstances` en las instancias EC2. Para solucionar este problema, adjunte una política de IAM al rol de asunción que tenga permiso para invocar la API `RunInstances`. Para obtener más información, consulte Cree los roles de servicio para la automatización mediante la consola.

Estado inesperado

Mensaje de error: `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.`

- Posible causa 1: hay un problema en la instancia o el servicio Amazon EC2. Para solucionar este problema, inicie sesión en la instancia o revise el registro del sistema de la instancia para entender por qué la instancia ha comenzado a cerrarse.
- Posible causa 2: el script de datos de usuario especificado para la acción `aws:runInstances` tiene un problema o una sintaxis incorrecta. Verifique la sintaxis del script de datos de usuario. Además, verifique que los scripts de datos de usuario no cierran la instancia, o invoque otros scripts que cierren la instancia.

Referencias de errores específicos de la acción

Cuando se produce un error en un paso, el mensaje de error podría indicar qué servicio se estaba invocando cuando se produjo el error. En la siguiente tabla se enumeran los servicios invocados por cada acción. La tabla también incluye enlaces a información acerca de cada servicio.

Acción	Servicios de AWS invocados por esta acción	Para obtener información sobre este servicio	Solución de problemas de contenido
`aws:runInstances`	Amazon EC2	Guía del usuario de Amazon EC2	Solución de problemas de las instancias EC2
`aws:changeInstanceState`	Amazon EC2	Guía del usuario de Amazon EC2	Solución de problemas de las instancias EC2

Acción	Servicios de AWS invocados por esta acción	Para obtener información sobre este servicio	Solución de problemas de contenido
`aws:runCommand`	Systems Manager	AWS Systems Manager Run Command	Solución de problemas Systems Manager Run Command
`aws:createImage`	Amazon EC2	Amazon Machine Images	
`aws:createStack`	AWS CloudFormation	Guía del usuario de AWS CloudFormation	Solución de problemas de AWS CloudFormation
`aws:deleteStack`	AWS CloudFormation	Guía del usuario de AWS CloudFormation	Solución de problemas de AWS CloudFormation
`aws:deleteImage`	Amazon EC2	Imágenes de máquina de Amazon	
`aws:copyImage`	Amazon EC2	Amazon Machine Images	
`aws:createTag`	Amazon EC2, Systems Manager	Recursos y etiquetas de EC2	
`aws:invokeLambdaFunction`	AWS Lambda	Guía para desarrolladores de AWS Lambda	Solución de problemas de Lambda

Error interno del servicio Automation

Mensaje de error: `Internal Server Error. Please refer to Automation Service Troubleshooting Guide for more diagnosis details.`

Un problema con el servicio de Automation impide que el manual de procedimientos especificado se ejecute correctamente. Para solucionar este problema, póngase en contacto con AWS Support. Proporcione el ID de ejecución y el ID de cliente, si están disponibles.

La ejecución se ha iniciado, pero se ha agotado el tiempo de espera

Mensaje de error: `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.`

Se ha agotado el tiempo de espera de la acción `aws:runInstances`. Esto puede ocurrir si la acción del paso tarda en ejecutarse más que el valor especificado para `timeoutSeconds` en el paso. Para resolver este problema, especifique un valor más largo para el parámetro `timeoutSeconds` en la acción `aws:runInstances`. Si no se resuelve el problema con esto, investigue el motivo por el cual el paso tarda más de lo esperado en ejecutarse

AWS Systems Manager Change Calendar

Change Calendar, una herramienta de AWS Systems Manager, le permite configurar intervalos de fecha y hora cuando las acciones que especifique (por ejemplo, en manuales de procedimientos de Automatización de Systems Manager) pueden llevarse a cabo o no en su Cuenta de AWS. En Change Calendar, estos intervalos se denominan eventos. Cuando crea una entrada de Change Calendar, está creando un documento de Systems Manager del tipo `ChangeCalendar`. En Change Calendar, el documento almacena datos iCalendar 2.0 en texto sin formato. Los eventos que añada a la entrada de Change Calendar pasan a formar parte del documento. Para comenzar a utilizar Change Calendar, abra la consola de Systems Manager. En el panel de navegación, elija Change Calendar.

Puede crear un calendario y sus eventos en la consola de Systems Manager. También puede importar un iCalendar (`.ics`) que ha exportado desde un proveedor de calendario de terceros compatible para agregar los eventos al calendario. Los proveedores compatibles incluyen Google Calendar, Microsoft Outlook y iCloud Calendar.

Una entrada de Change Calendar puede ser de dos tipos:

`DEFAULT_OPEN` o abierta de forma predeterminada

Todas las acciones se pueden ejecutar de forma predeterminada, excepto durante los eventos del calendario. Durante los eventos, el estado de un calendario `DEFAULT_OPEN` es `CLOSED` y los eventos están bloqueados para que se ejecuten.

`DEFAULT_CLOSED` o cerrada de forma predeterminada

Todas las acciones están bloqueadas de forma predeterminada, excepto durante los eventos del calendario. Durante los eventos, el estado de un calendario `DEFAULT_CLOSED` es `OPEN` y se permite ejecutar acciones.

Puede elegir que todos los flujos de trabajo de automatización programados, los períodos de mantenimiento y las asociaciones de State Manager se agreguen automáticamente a un calendario. También puede eliminar cualquiera de esos tipos individuales de la pantalla del calendario.

¿Quién debe utilizar Change Calendar?

- Los clientes de AWS que realizan los siguientes tipos de acciones:
  - Cree o ejecute manuales de procedimientos de automatización.
  - Cree una solicitud de cambio en Change Manager.
  - Ejecute los periodos de mantenimiento.
  - Cree una asociación en State Manager.

  Automatización, Change Manager, Maintenance Windows y State Manager son herramientas de AWS Systems Manager. Mediante la integración de estas herramientas a Change Calendar, puede permitir o bloquear estos tipos de acciones en función del estado actual del calendario de cambios que asocie a cada uno.
- Lo deben utilizar los administradores responsables de que las configuraciones de los nodos administrados de Systems Manager sean coherentes, estables y funcionales en todo momento.

Ventajas de Change Calendar

A continuación se describen los beneficios de Change Calendar.

- Revisión de los cambios antes de aplicarlos

  Una entrada de Change Calendar puede ayudar a garantizar que los cambios potencialmente destructivos para el entorno se revisen antes de aplicarlos.
- Aplicación de cambios solo durante los momentos apropiados

  Las entradas de Change Calendar ayudan a mantener el entorno estable durante los eventos. Por ejemplo, puede crear una entrada de Change Calendar para impedir los cambios cuando

espere una gran demanda de recursos; por ejemplo, durante una conferencia o una promoción de marketing pública. Una entrada de calendario también puede impedir los cambios mientras esté previsto que el soporte de administradores sea limitado; por ejemplo, durante las vacaciones o los días festivos. Puede usar una entrada de calendario para permitir los cambios excepto en determinados momentos del día o de la semana en los que disponga de soporte de administradores limitado para solucionar problemas de acciones o implementaciones fallidas.

- Obtener el estado actual o próximo del calendario

  Puede ejecutar la operación `GetCalendarState` de la API de Systems Manager para mostrar el estado actual del calendario, el estado a una hora determinada o la próxima vez programada para cambiar el estado del calendario.

  > **Note**
  >
  > La API `GetCalendarState` tiene una cuota de 10 solicitudes por segundo. Para obtener más información sobre las cuotas de Systems Manager, consulte Service Quotas de Systems Manager en Referencia general de Amazon Web Services.

- Compatibilidad con EventBridge

  Esta herramienta de Systems Manager se admite como un tipo de evento en las reglas de Amazon EventBridge. Para obtener más información, consulte Monitoreo de eventos de Systems Manager con Amazon EventBridge y Referencia: patrones y tipos de eventos de Amazon EventBridge para Systems Manager.

Temas

- Configuración de Change Calendar
- Uso de Change Calendar
- Agregado de dependencias de Change Calendar a manuales de procedimientos de Automation
- Solución de problemas de Change Calendar

Configuración de Change Calendar

Complete lo siguiente antes de usar Change Calendar, una herramienta de AWS Systems Manager.

Instalar las herramientas de línea de comandos más recientes

Instale las herramientas de línea de comandos más recientes para obtener información de estado acerca de los calendarios.

Requisito	Descripción
AWS CLI	(Opcional) Para utilizar la AWS Command Line Interface (AWS CLI) con el fin de obtener información de estado acerca de los calendarios, instale la versión más reciente de la AWS CLI en el equipo local. Para obtener más información acerca de cómo se instala o actualiza la CLI, consulte Instalación, actualización y desinstalación de AWS CLI en la Guía del usuario de AWS Command Line Interface.
Herramientas de AWS para PowerShell	(Opcional) Para utilizar Tools for PowerShell con el fin de obtener información de estado acerca de los calendarios, instale la versión más reciente de Tools for PowerShell en el equipo local. Para obtener más información acerca de cómo se instala o actualiza Tools for PowerShell, consulte Instalación de Herramientas de AWS para PowerShell en la Guía del usuario de Herramientas de AWS para PowerShell.

Configuración de permisos

Si el usuario, grupo o rol tiene asignados permisos de administrador, entonces tiene acceso a Change Calendar. Si no tiene permisos de administrador, un administrador debe concederle permiso mediante la asignación de la política administrada `AmazonSSMFullAccess` o de una política que proporcione permisos comparables a su usuario, grupo o rol.

Los siguientes permisos son necesarios para utilizar Change Calendar.

Entradas de Change Calendar

Para crear, actualizar o eliminar una entrada de Change Calendar, incluida la adición y eliminación de eventos de la entrada, debe tener una política adjunta a su usuario, grupo o rol que permita las siguientes acciones:

- `ssm:CreateDocument`
- `ssm:DeleteDocument`
- `ssm:DescribeDocument`
- `ssm:DescribeDocumentPermission`
- `ssm:GetCalendar`
- `ssm:ListDocuments`
- `ssm:ModifyDocumentPermission`
- `ssm:PutCalendar`
- `ssm:UpdateDocument`
- `ssm:UpdateDocumentDefaultVersion`

Estado del calendario

Para obtener información sobre el estado actual o próximo del calendario, debe tener una política adjunta a su usuario, grupo o rol que permita la siguiente acción:

- `ssm:GetCalendarState`

Eventos operativos

Para ver los eventos operativos, como los periodos de mantenimiento, las asociaciones y las automatizaciones planificadas, la política asociada al usuario, grupo o rol debe permitir las siguientes acciones:

- `ssm:DescribeMaintenanceWindows`
- `ssm:DescribeMaintenanceWindowExecution`
- `ssm:DescribeAutomationExecutions`
- `ssm:ListAssociations`

> **Note**
>
> Las entradas de Change Calendar que son propiedad de (es decir, que se han creado mediante) cuentas distintas de las suyas serán de solo lectura, aunque se compartan con su cuenta. Los periodos de mantenimiento, las asociaciones de State Manager y las automatizaciones no se comparten.

Uso de Change Calendar

Puede usar la consola de AWS Systems Manager para agregar, administrar o eliminar entradas de Change Calendar, una herramienta de AWS Systems Manager. También puede importar eventos de proveedores de calendarios de terceros admitidos mediante la importación de un archivo de iCalendar (`.ics`) que exportó desde el calendario fuente. Puede utilizar la operación de la API `GetCalendarState` o el comando `get-calendar-state` de AWS Command Line Interface (AWS CLI) para obtener información acerca del estado de Change Calendar en un momento específico.

Temas

- Creación de un calendario de cambios
- Creación y administración de eventos en Change Calendar
- Importación y administración de eventos desde calendarios de terceros
- Actualización de un calendario de cambios
- Uso compartido de un calendario de cambios
- Eliminación de un calendario de cambios
- Obtención del estado de un calendario de cambios

Creación de un calendario de cambios

Cuando crea una entrada en Change Calendar, una herramienta de AWS Systems Manager, está creando un documento de Systems Manager (documento de SSM) que utiliza el formato `text`.

Para crear un calendario de cambios

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.

2. En el panel de navegación, elija Change Calendar.
3. Elija Create calendar (Crear calendario).

   -o bien-

   Si se abre la página de inicio Change Calendar primero, elija Create change calendar (Creación de calendario de cambios).
4. En la página Create calendar (Crear calendario), en Calendar details (Detalles del calendario), escriba un nombre para la entrada de calendario. Los nombres de entradas de calendario solo pueden incluir letras, números, puntos, guiones y guiones bajos. El nombre debe ser lo bastante específico como para identificar el propósito de la entrada de calendario de un vistazo. Un ejemplo es **`support-off-hours`**. Este nombre no se puede actualizar después de haber creado la entrada de calendario.
5. (Opcional) En Description (Descripción), escriba una descripción para la entrada de calendario.
6. (Opcional) En el área Import calendar (Importa calendario), elija Choose file (Elegir archivo) para seleccionar un archivo de iCalendar (`.ics`) que ha exportado desde un proveedor de calendario de terceros. Al importar el archivo, los eventos al calendario se agregarán.

   Los proveedores compatibles incluyen Google Calendar, Microsoft Outlook y iCloud Calendar.

   Para obtener más información, consulte Importación de eventos de proveedores de calendarios de terceros.
7. En Calendar type (Tipo de calendario), elija una de las opciones siguientes:

   - Open by default (Abierto de forma predeterminada): el calendario se abre (las acciones de Automation se pueden ejecutar hasta que se inicie un evento) y, a continuación, se cierra durante el tiempo de un evento asociado.
   - Closed by default (Cerrado de forma predeterminada): el calendario se cierra (las acciones de Automation no se pueden ejecutar hasta que se inicie un evento), pero se abre durante el tiempo de un evento asociado.
8. (Opcional) En Eventos de administración de cambios, seleccione Agregar eventos de administración de cambios al calendario. En esta selección se muestran todos los periodos de mantenimiento programado, las asociaciones de State Manager, los flujos de trabajo de automatización y las solicitudes de cambios de Change Manager en la visualización del calendario mensual.

> **Tip**
>
> Si más adelante desea eliminar permanentemente estos tipos de eventos de la visualización del calendario, edite el calendario, desactive esta casilla y, a continuación, elija Guardar.

9. Elija Create calendar (Crear calendario).

   Una vez creada la entrada de calendario, Systems Manager muestra la entrada de calendario en la lista Change Calendar. Las columnas muestran la versión del calendario y el número de la Cuenta de AWS del propietario del calendario. La entrada de calendario no puede impedir ni permitir ninguna acción hasta que haya agregado al menos un evento. Para obtener más información sobre cómo se crea un evento, consulte Creación de un evento de Change Calendar. Para obtener información sobre la importación de eventos, consulte Importación de eventos de proveedores de calendarios de terceros.

Creación y administración de eventos en Change Calendar

Después de crear un calendario en AWS Systems Manager Change Calendar, puede crear, actualizar y eliminar eventos incluidos en el calendario abierto o cerrado. Change Calendar es una herramienta de AWS Systems Manager.

> **Tip**
>
> Como alternativa a crear eventos directamente en la consola de Systems Manager, puede importar un archivo de iCalendar (`.ics`) de una aplicación de calendario de terceros compatible. Para obtener más información, consulta Importación y administración de eventos desde calendarios de terceros.

Temas

- Creación de un evento de Change Calendar
- Actualización de un evento de Change Calendar
- Eliminación de un evento de Change Calendar

Creación de un evento de Change Calendar

Cuando agrega un evento a una entrada de Change Calendar, una herramienta de AWS Systems Manager, especifica un periodo durante el cual se suspende la acción predeterminada de la entrada de calendario. Por ejemplo, si el tipo de entrada de calendario es cerrado de forma predeterminada, el calendario estará abierto para realizar cambios durante los eventos. (Alternativamente, puede crear un evento de asesoramiento, que solo cumple un rol informativo en el calendario).

Actualmente, solo puede crear un evento de Change Calendar mediante la consola. Los eventos se agregan al documento de Change Calendar que crea cuando crea una entrada de Change Calendar.

Para crear un evento de Change Calendar

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Change Calendar.
3. En la lista de calendarios, elija el nombre de la entrada de calendario a la que desea agregar un evento.
4. En la página de detalles de la entrada de calendario, seleccione Create event (Crear evento).
5. En la página Create scheduled event (Crear evento programado), en Event details (Detalles del evento) escriba el un nombre para mostrar del evento. Los nombres de eventos solo pueden incluir letras, números, puntos, guiones y guiones bajos. El nombre debe ser lo bastante específico para identificar el propósito del evento. Un ejemplo es **`nighttime-hours`**.
6. En Description (Descripción), escriba una descripción del evento. Por ejemplo, **`The support team isn't available during these hours`**.
7. (Opcional) Si desea que este evento sirva solo como notificación visual o recordatorio, seleccione la casilla de verificación Advisory (Recordatorio). Los eventos de recordatorio no cumplen ningún rol funcional en su calendario. Solo sirven para fines informativos para aquellos que ven su calendario.
8. En Event start date (Fecha de inicio del evento), ingrese o elija un día con el formato `MM/DD/YYYY` para iniciar el evento e indique una hora en el día especificado con el formato `hh:mm:ss` (horas, minutos y segundos) para iniciar el evento.
9. En Event end date (Fecha de finalización del evento), ingrese o elija un día con el formato `MM/DD/YYYY` para finalizar el evento e indique una hora en el día especificado con el formato `hh:mm:ss` (horas, minutos y segundos) para finalizar el evento.

10. En Schedule time zone (Programar zona horaria), elija la zona horaria aplicable a las horas de inicio y finalización del evento. Puede introducir parte del nombre de una ciudad o la diferencia de zona horaria con respecto a la hora media de Greenwich (GMT) para encontrar una zona horaria más rápidamente. El valor predeterminado es la Hora Universal Coordinada (UTC).
11. (Opcional) Para crear un evento que se repita con periodicidad diaria, semanal o mensual, active Recurrence (Recurrencia) y, a continuación, especifique la frecuencia y la fecha de finalización opcionales de la repetición.
12. Elija Create scheduled event (Crear evento programado). El nuevo evento se agrega a la entrada de calendario y se muestra en la pestaña Events (Eventos) de la página de detalles de la entrada de calendario.

Actualización de un evento de Change Calendar

Utilice el siguiente procedimiento para actualizar un evento Change Calendar en la consola de AWS Systems Manager. Change Calendar es una herramienta de AWS Systems Manager.

Para actualizar un evento de Change Calendar

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Change Calendar.
3. En la lista de calendarios, elija el nombre de la entrada de calendario cuyo evento desea editar.
4. En la página de detalles de la entrada de calendario, seleccione Events (Eventos).
5. En la página del calendario, elija el evento que desea editar.

> **Tip**
>
> Utilice los botones de la parte superior izquierda para avanzar o retroceder un año o un mes. Si es preciso cambiar la zona horaria, elija la que sea correcta en la lista ubicada en la parte superior derecha.

6. En Event details (Detalles del evento), elija Edit (Editar).

   Para cambiar el nombre y la descripción del evento, agregue o reemplace los valores de texto actuales.

7. Para cambiar el valor Event start date (Fecha de inicio del evento), elija la fecha de inicio actual y, a continuación, elija una nueva fecha del calendario. Para cambiar la hora de inicio, elija la hora de inicio actual y, a continuación, elija una nueva hora de la lista.
8. Para cambiar el valor Event end date (Fecha de finalización del evento), elija la fecha actual y, a continuación, elija una nueva fecha de finalización del calendario. Para cambiar la hora de finalización, elija la hora de finalización actual y, a continuación, elija una nueva hora de la lista.
9. Para cambiar el valor de Schedule time zone (Programar zona horaria), seleccione la zona horaria aplicable a la hora de inicio y de finalización del evento. Puede introducir parte del nombre de una ciudad o la diferencia de zona horaria con respecto a la hora media de Greenwich (GMT) para encontrar una zona horaria más rápidamente. El valor predeterminado es la Hora Universal Coordinada (UTC).
10. (Opcional) Si desea que este evento sirva solo como notificación visual o recordatorio, seleccione la casilla de verificación Advisory (Recordatorio). Los eventos de recordatorio no cumplen ningún rol funcional en su calendario. Solo sirven para fines informativos para aquellos que ven su calendario.
11. Seleccione Guardar. Los cambios se muestran en la pestaña Events (Eventos) de la página de detalles de la entrada de calendario. Elija el evento que ha actualizado para ver los cambios.

Eliminación de un evento de Change Calendar

Puede eliminar un evento a la vez en Change Calendar, una herramienta de AWS Systems Manager, mediante la AWS Management Console.

> **Tip**
>
> Si cuando creó el calendario seleccionó Agregar eventos de administración de cambios al calendario, puede hacer lo siguiente:
>
> - Para ocultar temporalmente un tipo de evento de administración de cambios de la visualización del calendario, seleccione la X para el tipo en la parte superior de la vista previa mensual.
> - Para eliminar permanentemente estos tipos de la visualización del calendario, edite el calendario, desactive la casilla Agregar eventos de administración de cambios al calendario y, a continuación, seleccione Guardar. Si elimina los tipos de la visualización del calendario, no se eliminarán de la cuenta.

Para eliminar un evento de Change Calendar

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Change Calendar.
3. En la lista de calendarios, elija el nombre de la entrada de calendario cuyo evento desea eliminar.
4. En la página de detalles de la entrada de calendario, seleccione Events (Eventos).
5. En la página del calendario, elija el evento que desea eliminar.

> **Tip**
>
> Utilice los botones de la parte superior izquierda para avanzar o retroceder un año o un mes en el calendario. Si es preciso cambiar la zona horaria, elija la que sea correcta en la lista ubicada en la parte superior derecha.

6. En la página Event details (Detalles del evento), seleccione Delete (Eliminar). Cuando se le pida que confirme que desea eliminar el evento, elija Confirm (Confirmar).

Importación y administración de eventos desde calendarios de terceros

Como alternativa a la creación de eventos directamente en la consola de AWS Systems Manager, puede importar un archivo de iCalendar (`.ics`) de una aplicación de calendario de terceros compatible. El calendario puede incluir eventos importados y eventos que cree en Change Calendar, que es una herramienta de AWS Systems Manager.

Antes de empezar

Antes de intentar importar un archivo de calendario, revise los siguientes requisitos y restricciones:

Formato del archivo del calendario

Solo son compatibles los archivos de iCalendar válidos (`.ics`).

Proveedores de calendarios compatibles

Solo se admiten los archivos `.ics` exportados de los siguientes proveedores de calendarios de terceros:

- Calendario de Google (Instrucciones de exportación)
- Microsoft Outlook (Instrucciones de exportación)
- Calendario de iCloud (Instrucciones de exportación)

Tamaño del archivo

Puede importar cualquier número de archivos `.ics` válidos. Sin embargo, el tamaño total de todos los archivos importados de cada calendario no puede superar los 64 KB.

> **Tip**
>
> Para minimizar el tamaño del archivo `.ics`, asegúrese de exportar solo los detalles básicos sobre las entradas del calendario. Si es necesario, reduzca la duración del período de tiempo que va a exportar.

Zona horaria

Además de un nombre de calendario, un proveedor de calendario y al menos un evento, el archivo `.ics` que se exporta debe indicar también la zona horaria del calendario. Si no lo hace o hay un problema al identificar la zona horaria, se le pedirá que especifique una después de importar el archivo.

Limitación de eventos periódicos

El archivo `.ics` que exportó puede incluir eventos recurrentes. Sin embargo, si se ha eliminado una o varias apariciones de un evento recurrente en el calendario fuente, se produce un error en la importación.

Temas

- Importación de eventos de proveedores de calendarios de terceros
- Actualización de todos los eventos de un proveedor de calendario de terceros
- Eliminación de todos los eventos importados de un calendario de terceros

Importación de eventos de proveedores de calendarios de terceros

Utilice el siguiente procedimiento para importar un archivo de iCalendar (`.ics`) de una aplicación de calendario de terceros compatible. Los eventos que están en el archivo se incorporan a las reglas del

calendario abierto o cerrado. Puede importar un archivo a un nuevo calendario que esté creando con Change Calendar (una herramienta de AWS Systems Manager) o en un calendario existente.

Después de importar el archivo `.ics`, puede eliminar eventos individuales de este mediante el la interfaz de Change Calendar. Para obtener más información, consulta Eliminación de un evento de Change Calendar. También puede eliminar todos los eventos del calendario fuente mediante la eliminación del archivo `.ics`. Para obtener más información, consulta Eliminación de todos los eventos importados de un calendario de terceros.

Importación de eventos de proveedores de calendarios de terceros

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Change Calendar.
3. Para empezar con un calendario nuevo, elija Create calendar (Crear calendario). En el área Import calendar (Importar calendario), elija Choose file (Elegir archivo). Para obtener información sobre otros pasos para crear un calendario nuevo, consulte Creación de un calendario de cambios.

   -o bien-

   Para importar eventos de terceros a un calendario existente, elija el nombre de un calendario existente para abrirlo.
4. Elija Actions, Edit (Acciones, Editar) y, a continuación, en el área Import calendar (Importar calendario), elija Choose file (Elegir archivo).
5. Busque y seleccione el archivo `.ics` exportado en su equipo local.
6. Si se le pregunta, en Select a time zone (Seleccionar una zona horaria), seleccione qué zona horaria se aplica al calendario.
7. Seleccione Guardar.

Actualización de todos los eventos de un proveedor de calendario de terceros

Si se agregan o eliminan varios eventos del calendario fuente después de haber importado el archivo `.ics` de iCalendar, puede reflejar esos cambios en Change Calendar. En primer lugar, vuelva a exportar el calendario de origen y, a continuación, importe el archivo nuevo a Change Calendar, que es una herramienta de AWS Systems Manager. Los eventos del calendario de cambios se actualizarán para reflejar el contenido del archivo más reciente.

Para actualizar todos los eventos de un proveedor de calendario de terceros

1. En el calendario de terceros, agregue o elimine eventos tal como desee que se reflejen en Change Calendar y, a continuación, vuelva a exportar el calendario a un archivo `.ics` nuevo.
2. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
3. En el panel de navegación, elija Change Calendar.
4. Elija el nombre del calendario de la lista desde la lista de calendarios.
5. Seleccione Elegir archivo y, a continuación, busque y seleccione el archivo `.ics` de reemplazo.
6. En respuesta a la notificación sobre el reemplazo del archivo existente, elija Confirm (Confirmar).

Eliminación de todos los eventos importados de un calendario de terceros

Si ya no desea que ninguno de los eventos importados de un proveedor de terceros se incluya en el calendario, puede eliminar el archivo `.ics` importado de iCalendar.

Para eliminar todos los eventos importados de un calendario de terceros

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Change Calendar.
3. Elija el nombre del calendario de la lista desde la lista de calendarios.
4. En el área Import calendar (Importar calendario), bajo My imported calendars (Mis calendarios importados), busque el nombre del calendario importado y, a continuación, elija la Xen la tarjeta.
5. Seleccione Guardar.

Actualización de un calendario de cambios

Puede actualizar la descripción de un calendario de cambios, pero no el nombre. Aunque puede cambiar el estado predeterminado de un calendario, tenga en cuenta que esto invierte el comportamiento de las acciones de cambio durante los eventos asociados a la entrada del calendario. Por ejemplo, si cambia el estado de un calendario de Open by default (Abierto de forma predeterminada) a Closed by default (Cerrado de forma predeterminada), es posible que se realicen cambios no deseados durante los periodos de eventos durante los cuales los usuarios que crearon los eventos asociados no hayan previsto que se produzcan cambios.

Cuando actualiza un calendario de cambios, edita el documento Change Calendar que creó al crear la entrada. Change Calendar es una herramienta de AWS Systems Manager.

Actualización de un calendario de cambios

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Change Calendar.
3. En la lista de calendarios, elija el nombre del calendario que desee actualizar.
4. En la página de detalles del calendario, elija Actions, Edit (Acciones, Editar).
5. En Description (Descripción), puede cambiar el texto de la descripción. No es posible editar el nombre de un calendario de cambios.
6. Para cambiar el estado del calendario, elija un valor diferente en Calendar type (Tipo de calendario). Tenga en cuenta que esto invierte el comportamiento de las acciones de cambio durante los eventos asociados al calendario. Antes de cambiar el tipo de calendario, debe confirmar con los demás usuarios de Change Calendar que cambiar el tipo de calendario no permitirá cambios no deseados durante los eventos que ellos han creado.

   - Open by default (Abierto de forma predeterminada): el calendario se abre (las acciones de Automation se pueden ejecutar hasta que se inicie un evento) y, a continuación, se cierra durante el tiempo de un evento asociado.
   - Closed by default (Cerrado de forma predeterminada): el calendario se cierra (las acciones de Automation no se pueden ejecutar hasta que se inicie un evento), pero se abre durante el tiempo de un evento asociado.
7. Seleccione Guardar.

   El calendario no puede impedir ni permitir ninguna acción hasta que haya agregado al menos un evento. Para obtener información sobre cómo agregar un disco, consulte Creación de un evento de Change Calendar.

Uso compartido de un calendario de cambios

Puede compartir un calendario en Change Calendar, una herramienta de AWS Systems Manager, con otras Cuentas de AWS mediante el uso de la consola de AWS Systems Manager. Cuando se comparte un calendario, el calendario es de solo lectura para los usuarios de la cuenta compartida. Los periodos de mantenimiento, las asociaciones de State Manager y las automatizaciones no se comparten.

Uso compartido de un calendario de cambios

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Change Calendar.
3. En la lista de calendarios, elija el nombre del calendario que desee compartir.
4. En la página de detalles del calendario, elija la pestaña Sharing (Compartir).
5. Elija Action, Share (Acciones, Compartir).
6. En Share Calendar (Compartir calendario), para Account ID (ID de cuenta), ingrese el número de ID de una Cuenta de AWS válida y luego elija Share (Compartir).

   Los usuarios de la cuenta compartida pueden leer el calendario de cambios, pero no pueden realizar cambios.

Eliminación de un calendario de cambios

Puede eliminar un calendario en Change Calendar, una herramienta de AWS Systems Manager, mediante la consola de Systems Manager o AWS Command Line Interface (AWS CLI). Al eliminar un calendario de cambios, se eliminan todos los eventos asociados.

Eliminación de un calendario de cambios

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Change Calendar.
3. En la lista de calendarios, elija el nombre del calendario que desee eliminar.
4. En la página de detalles del calendario, seleccione Actions, Delete (Acciones, Eliminar). Cuando se le pida la confirmación para eliminar el calendario, elija Delete (Eliminar).

Obtención del estado de un calendario de cambios

Puede obtener el estado general de un calendario o el estado de un calendario en un momento específico en Change Calendar, una herramienta de AWS Systems Manager. También puede mostrar la próxima vez que el estado del calendario va a cambiar de `OPEN` a `CLOSED`, o a la inversa.

> **Note**
>
> Para obtener información sobre la integración de Change Calendar en Amazon EventBridge para la supervisión automática de los cambios de estado del calendario, consulte the section called “Integración de Change Calendar en Amazon Eventbridge”. La integración de EventBridge proporciona notificaciones basadas en eventos cuando los estados del calendario cambian, lo que complementa el enfoque basado en sondeos de la acción de la API `GetCalendarState`.

Esta tarea solo se puede realizar mediante la operación `GetCalendarState` de la API. En el procedimiento de esta sección se utiliza la AWS Command Line Interface (AWS CLI).

Para obtener el estado de un calendario de cambios

- Ejecute el siguiente comando para mostrar el estado de uno o más calendarios en un momento específico. El parámetro `--calendar-names` es obligatorio, pero `--at-time` es opcional. Reemplace cada *`example resource placeholder`* con su propia información.

  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"
  ```

  A continuación se muestra un ejemplo.

  ```
  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"
```

A continuación se muestra un ejemplo.

```
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"
```

El comando devuelve información similar a la siguiente.

```
{
    "State": "OPEN",
    "AtTime": "2020-07-30T16:18:18Z",
    "NextTransitionTime": "2020-07-31T00:00:00Z"
}
```

Los resultados muestran el estado del calendario (si el calendario es de tipo `DEFAULT_OPEN` o `DEFAULT_CLOSED`) de las entradas de calendario especificadas que son propiedad de la cuenta o compartidas con ella, en el momento especificado como valor de `--at-time`, así como en el momento de la siguiente transición. Si no agrega el parámetro `--at-time`, se utiliza la hora actual.

> **Note**
>
> Si especifica más de un calendario en una solicitud, el comando regresa el estado `OPEN` solo si todos los calendarios de la solicitud están abiertos. Si se cierran uno o más calendarios de la solicitud, el estado regresado es `CLOSED`.

Agregado de dependencias de Change Calendar a manuales de procedimientos de Automation

Para que las acciones de Automatización se adhieran a Change Calendar, una herramienta de AWS Systems Manager, deberá agregar un paso en un manual de procedimientos de Automatización que utilice la acción `aws:assertAwsResourceProperty`. Configure la acción para que ejecute `GetCalendarState`, con el fin de comprobar que una entrada de calendario especificada está

en el estado que desea (`OPEN` o `CLOSED`). El manual de procedimientos de Automation solo puede continuar con el siguiente paso si el estado del calendario es `OPEN`. A continuación se muestra un extracto de ejemplo basado en YAML de un manual de procedimientos de Automation que no puede avanzar al siguiente paso, `LaunchInstance`, a menos que el estado del calendario coincida con `OPEN`, que es el estado especificado en `DesiredValues`.

A continuación se muestra un ejemplo.

```
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
...
```

Solución de problemas de Change Calendar

Utilice la siguiente información como ayuda para solucionar problemas con Change Calendar, una herramienta de AWS Systems Manager.

Temas

- Error ‘Error en la importación del calendario’

Error ‘Error en la importación del calendario’

Problema: Al importar un archivo de iCalendar (`.ics`), el sistema informa que ocurrió un error en la importación del calendario.

- Solución 1: Asegúrese de importar un archivo que se ha exportado desde un proveedor de calendario de terceros compatible, que incluya lo siguiente:
  - Calendario de Google (Instrucciones de exportación)
  - Microsoft Outlook (Instrucciones de exportación)
  - Calendario de iCloud (Instrucciones de exportación)
- Solución 2: Si el calendario fuente contiene eventos recurrentes, asegúrese de que no se haya cancelado o eliminado ninguna ocurrencia individual del evento. En la actualidad, Change Calendar no admite la importación de eventos recurrentes con cancelaciones individuales. Para resolver el problema, elimine el evento recurrente del calendario fuente, vuelva a exportar el calendario y vuelva a importarlo en Change Calendar y, a continuación, agregue el evento recurrente mediante la interfaz de Change Calendar. Para obtener más información, consulta Creación de un evento de Change Calendar.
- Solución 3: Asegúrese de que el calendario fuente contenga al menos un evento. Las cargas de los archivos `.ics` que no contienen eventos no se realizan correctamente.
- Solución 4: Si el sistema informa de que la importación ha fallado porque el `.ics` es demasiado grande, asegúrese de exportar solo detalles básicos sobre las entradas del calendario. Si es necesario, reduzca la duración del período de tiempo que exporta.
- Solución 5: Si Change Calendar no puede determinar la zona horaria del calendario exportado cuando intenta importarlo desde la pestaña Events (Eventos), puede que reciba este mensaje: “Falló la importación del calendario. Change Calendar no se ha podido localizar una zona horaria válida. Puede importar el calendario desde el menú Edit (Editar)”. En este caso, seleccione Accions, Edit (Acciones, Editar) y, a continuación, intente importar el archivo desde la página Edit calendar (Editar calendario).
- Solución 6: No edite el archivo `.ics` antes de importar. Si se intenta modificar el contenido del archivo, se pueden dañar los datos del calendario. Si ha modificado el archivo antes de intentar importar, vuelva a exportar el calendario desde el calendario fuente y, a continuación, vuelva a intentar la carga.

AWS Systems Manager Change Manager

Change Manager, una herramienta de AWS Systems Manager, es un marco empresarial de administración de cambios con el que se pueden solicitar, aprobar, implementar e informar los cambios operativos de la configuración y la infraestructura de la aplicación. A partir de una única cuenta de administrador delegado, si utiliza AWS Organizations, puede administrar los cambios en varias Cuentas de AWS y Regiones de AWS. De forma alternativa, a través de una cuenta

local, puede administrar los cambios de una sola Cuenta de AWS. Utilice Change Manager para administrar los cambios tanto en los recursos de AWS como en los recursos locales. Para comenzar a utilizar Change Manager, abra la consola de Systems Manager. En el panel de navegación, elija Change Manager.

Con Change Manager, puede usar plantillas de cambios preaprobadas para ayudar a automatizar los procesos de cambio de sus recursos y evitar los resultados no intencionales cuando se efectúan cambios operativos. Cada plantilla de cambios especifica lo siguiente:

- Se definen uno o más manuales de procedimientos de Automation de los que el usuario puede elegir cuando vaya a crear una solicitud de cambio. Los cambios que se realizan en los recursos se definen en los manuales de procedimientos de Automation. Puede incluir manuales de procedimientos personalizados o manuales de procedimientos administrados de AWS en las plantillas de cambios que cree. Cuando un usuario crea una solicitud de cambio, puede elegir cuál de los manuales de procedimientos disponibles incluirá en la solicitud. Además, puede crear plantillas de cambios que permitan al usuario que realiza la solicitud especificar cualquier manual de procedimientos en la solicitud de cambio.
- Se indican los usuarios de la cuenta que deben revisar las solicitudes de cambio que se realizan con esa plantilla de cambios.
- Se establece el tema de Amazon Simple Notification Service (Amazon SNS) que se utilizará para notificar a los aprobadores designados que una solicitud de cambio está lista para su revisión.
- Se indica la alarma de Amazon CloudWatch que se utilizará para monitorear el flujo de trabajo del manual de procedimientos.
- Se define el tema de Amazon SNS que se utilizará para enviar notificaciones acerca de los cambios de estado de las solicitudes de cambio que se creen con la plantilla de cambios.
- Se indican las etiquetas que se aplicarán a la plantilla de cambios para utilizarlas en la clasificación en categorías y el filtro de las plantillas de cambios.
- Se establece si las solicitudes de cambio creadas a partir de la plantilla de cambios se pueden ejecutar sin un paso de aprobación (solicitudes de aprobación automática).

A través de su integración a Change Calendar, que es otra herramienta de Systems Manager, Change Manager también lo ayuda a implementar cambios de forma segura a la vez que evita conflictos de programación con los eventos empresariales importantes. La integración de Change Manager a AWS Organizations y AWS IAM Identity Center lo ayuda a administrar los cambios en toda su organización desde una única cuenta con el sistema de administración de identidades existente. Puede monitorear el progreso de los cambios desde Change Manager y auditar los

cambios operativos en toda su organización, lo que proporciona visibilidad y rendición de cuentas mejoradas.

Change Manager complementa los controles de seguridad de sus prácticas de integración continua (CI) y su metodología de entrega continua (CD). Change Manager no está diseñado para cambios realizados como parte de un proceso de lanzamiento automatizado, como, por ejemplo, una canalización de CI/CD, a menos que se requiera una excepción o una aprobación.

Cómo funciona Change Manager

Cuando se identifica la necesidad de un cambio operativo estándar o de emergencia, alguien de la organización crea una solicitud de cambio basada en una de las plantillas de cambios que se crearon para usarlas en su organización o cuenta.

Si el cambio solicitado requiere aprobaciones manuales, Change Manager notifica a los aprobadores designados mediante una notificación de Amazon SNS que indica que una solicitud de cambio está lista para su revisión. Puede designar aprobadores para las solicitudes de cambio en la plantilla de cambios o permitir que los usuarios designen a los aprobadores de la solicitud de cambio en la misma solicitud. Puede asignar diferentes revisores a diferentes plantillas. Por ejemplo, asigne un usuario, un grupo de usuarios o un rol de AWS Identity and Access Management (IAM) que deba aprobar las solicitudes de cambio en los nodos administrados y otro usuario, grupo o rol de IAM para los cambios en la base de datos. Si la plantilla de cambios permite las aprobaciones automáticas y la política de usuario de un solicitante no las prohíbe, el usuario también puede elegir ejecutar el manual de procedimientos de Automatización para su solicitud sin un paso de revisión (con la excepción de los eventos de congelación de cambios).

Puede agregar hasta cinco niveles de aprobadores a cada plantilla de cambios. Por ejemplo, puede requerir que los revisores técnicos aprueben primero una solicitud de cambio creada a partir de una plantilla de cambios y, a continuación, requiera un segundo nivel de aprobaciones por parte de uno o más administradores.

Change Manager está integrado en AWS Systems Manager Change Calendar. Cuando se aprueba un cambio solicitado, en primer lugar, el sistema determina si la solicitud entra en conflicto con otras actividades empresariales programadas. Si se detecta un conflicto, Change Manager puede bloquear el cambio o requerir aprobaciones adicionales antes de comenzar a ejecutar el flujo de trabajo del manual de procedimientos. Por ejemplo, puede permitir cambios solo durante el horario laborable para asegurarse de que los equipos estén disponibles para resolver cualquier problema inesperado. Para cualquier cambio que se solicite para ejecutarse fuera de ese horario, puede requerir una aprobación administrativa de nivel superior en forma de aprobadores de congelación de cambios.

Para los cambios de emergencia, Change Manager puede omitir el paso de verificación de Change Calendar para detectar conflictos o eventos de bloqueo después de que se aprueba una solicitud de cambio.

Cuando llegue el momento de implementar un cambio aprobado, Change Manager ejecutará el manual de procedimientos de Automation que se especifica en la solicitud de cambio asociada. Solo se permiten las operaciones definidas en las solicitudes de cambio aprobadas cuando se ejecutan los flujos de trabajo del manual de procedimientos. Este enfoque ayuda a evitar resultados no intencionales mientras se implementan los cambios.

Además de restringir los cambios que se pueden realizar cuando se ejecuta un flujo de trabajo de manual de procedimientos, Change Manager también lo ayuda a controlar los límites de simultaneidad y errores. Puede elegir cuántos recursos puede ejecutar un flujo de trabajo de manual de procedimientos a la vez, en cuántas cuentas puede ejecutarse el cambio a la vez y cuántos errores permitir antes de que el proceso se detenga y (si el manual de procedimientos incluye un script de restauración) se restaure. También puede monitorear el progreso de los cambios que se están realizando mediante las alarmas de CloudWatch.

Una vez finalizado el flujo de trabajo del manual de procedimientos, puede revisar los detalles de los cambios efectuados. Estos detalles incluyen el motivo de una solicitud de cambio, qué plantilla de cambios se utilizó, quién solicitó los cambios, quién los aprobó y cómo se implementaron.

Más información

Presentación de AWS Systems ManagerChange Manager en el Blog de noticias de AWS

¿Cómo puede Change Manager beneficiar las operaciones de mi organización?

Entre los beneficios de Change Manager se incluyen los siguientes:

- Reduzca el riesgo de interrupción del servicio y de tiempo de inactividad

  Change Manager puede hacer que los cambios operativos sean más seguros garantizando que solo se implementen los cambios aprobados cuando se ejecuta un flujo de trabajo de manual de procedimientos. Puede bloquear los cambios no planificados y los no revisados. Change Manager lo ayuda a evitar los tipos de resultados no intencionales causados por errores humanos que requieren costosas horas de investigación y acciones para deshacer dichos resultados.
- Obtenga auditorías e informes detallados sobre los historiales de cambios

Change Manager proporciona la posibilidad de realizar rendiciones de cuentas con una forma coherente de notificar y auditar los cambios realizados en toda la organización, la intención de los cambios y los detalles sobre quién los aprobó e implementó.

- Evite conflictos o infracciones de programación

  Change Manager puede detectar conflictos de programación, como los eventos festivos o los lanzamientos de nuevos productos, en función del calendario de cambios activo para su organización. Puede permitir que los flujos de trabajo del manual de procedimientos se ejecuten solo durante el horario laborable o permitirlos solo con aprobaciones adicionales.

- Adapte los requisitos de cambios a su negocio cambiante

  Durante los diferentes periodos empresariales, puede implementar distintos requisitos de administración de cambios. Por ejemplo, durante la preparación de informes de fin de mes, el periodo de presentaciones fiscales u otros periodos empresariales críticos, puede bloquear los cambios o requerir la aprobación del nivel de directores para los cambios que podrían generar riesgos operativos innecesarios.

- Administre de forma centralizada los cambios en las cuentas

  Gracias a su integración a Organizations, Change Manager le permite administrar los cambios en todas las unidades organizativas desde una única cuenta de administrador delegado. Puede activar Change Manager para usarlo en toda su organización o solo en algunas de sus unidades organizativas.

¿Quién debe utilizar Change Manager?

Change Manager es apropiado para los siguientes clientes y organizaciones de AWS:

- cualquier cliente de AWS que desee mejorar la seguridad y la gobernanza de los cambios operativos realizados en sus entornos en la nube o en las instalaciones
- las organizaciones que deseen aumentar la colaboración y la visibilidad entre los equipos, mejorar la disponibilidad de las aplicaciones evitando el tiempo de inactividad y reducir el riesgo asociado a las tareas manuales y repetitivas
- las organizaciones que deban cumplir prácticas recomendadas para la administración de cambios
- clientes que necesiten un historial totalmente auditable de los cambios realizados en la configuración y la infraestructura de sus aplicaciones

¿Cuáles son las características principales de Change Manager?

Las características principales de Change Manager incluyen las siguientes:

- Soporte integrado para las prácticas recomendadas de administración de cambios

  Con Change Manager, puede aplicar prácticas recomendadas de administración de cambios selectas a sus operaciones. Puede elegir activar las siguientes opciones:

  - verificar Change Calendar para ver si los eventos están restringidos actualmente, de manera que los cambios se realizan solo durante los periodos abiertos del calendario
  - permitir cambios durante eventos restringidos con aprobaciones adicionales de aprobadores de congelación de cambios
  - requerir que se especifiquen alarmas de CloudWatch para todas las plantillas de cambios
  - requerir que se revisen y aprueben todas las plantillas de cambios creadas en su cuenta antes de poder usarlas para crear solicitudes de cambio
- Diferentes rutas de aprobación para periodos cerrados del calendario y solicitudes de cambio de emergencia

  Puede permitir una opción para verificar Change Calendar para ver los eventos restringidos y bloquear solicitudes de cambio aprobadas hasta que se complete el evento. Sin embargo, también puede designar un segundo grupo de aprobadores, aprobadores de congelación de cambios, que pueden permitir que el cambio se realice incluso si el calendario está cerrado. También puede crear plantillas de cambios de emergencia. Las solicitudes de cambio creadas a partir de una plantilla de cambios de emergencia siguen requiriendo las aprobaciones regulares, pero no están sujetas a las restricciones del calendario ni requieren aprobaciones de congelación de cambios.
- Control de cómo y cuándo se inician los flujos de trabajo del manual de procedimientos

  Los flujos de trabajo del manual de procedimientos se pueden iniciar de acuerdo con una programación o tan pronto como se completen las aprobaciones (sujeto a las reglas de restricción del calendario).
- Compatibilidad integrada con notificaciones

  Especifique quién de su organización debe revisar y aprobar las plantillas y las solicitudes de cambios. Asigne un tema de Amazon SNS a una plantilla de cambios para enviar notificaciones a los suscriptores del tema sobre los cambios de estado de las solicitudes de cambio creadas con esa plantilla de cambios.
- Integración de AWS Systems Manager Change Calendar

Change Manager permite a los administradores restringir los cambios de programación durante periodos especificados. Por ejemplo, puede crear una política que permita cambios solo durante el horario laborable para asegurarse de que el equipo esté disponible para resolver cualquier problema. También puede restringir los cambios durante eventos empresariales importantes. Por ejemplo, las empresas minoristas pueden restringir los cambios durante los grandes eventos de ventas. También puede requerir aprobaciones adicionales durante los periodos con restricciones.

- Integración con AWS IAM Identity Center y compatibilidad con Active Directory

  Con la integración con el Centro de identidades de IAM, los miembros de su organización pueden acceder a las Cuentas de AWS y administrar sus recursos con Systems Manager basándose en una identidad de usuario común. Con el Centro de identidades de IAM, puede asignar a sus usuarios acceso a las AWS.

  La integración a Active Directory permite asignar usuarios a su cuenta de Active Directory como aprobadores de plantillas de cambios creadas para sus operaciones de Change Manager.

- Integración con las alarmas de Amazon CloudWatch

  Change Manager está integrado a las alarmas de CloudWatch. Change Manager escucha las alarmas de CloudWatch durante el flujo de trabajo del manual de procedimientos y lleva a cabo cualquier acción, incluido el envío de notificaciones, que se defina para la alarma.

- Integración con AWS CloudTrail Lake

  Al crear un almacén de datos de eventos en AWS CloudTrail Lake, puede ver información auditable sobre los cambios realizados por las solicitudes de cambio que se ejecutan en su cuenta u organización. La información del evento almacenada incluye detalles como los siguientes:

  - Las acciones de API ejecutadas
  - Los parámetros de solicitud incluidos para esas acciones
  - El usuario que ejecutó la acción
  - Los recursos que se actualizaron durante el proceso

- Integración con el AWS Organizations

  Mediante las capacidades entre cuentas proporcionadas por Organizations, puede utilizar una cuenta de administrador delegado para administrar las operaciones de Change Manager en las unidades organizativas de su organización. En su cuenta de administración de Organizations, puede especificar qué cuenta será la cuenta de administrador delegado. También puede controlar en cuáles de sus unidades organizativas se puede utilizar Change Manager.

¿Se cobra por usar Change Manager?

Sí. Change Manager tiene un precio de pago por uso. Solo paga por lo que utiliza. Para más información, consulte Precios de AWS Systems Manager.

¿Cuáles son los componentes principales de Change Manager?

Los componentes de Change Manager que se utilizan para administrar el proceso de cambios en su organización o su cuenta incluyen los siguientes:

Cuenta de administrador delegado

Si utiliza Change Manager en una organización, debe utilizar una cuenta de administrador delegado. Esta es la Cuenta de AWS designada como la cuenta para administrar las actividades de operaciones en Systems Manager, incluido Change Manager. La cuenta de administrador delegado se encarga de administrar las actividades de cambio en toda la organización. Cuando se configura la organización para utilizar Change Manager, se debe especificar cuál de sus cuentas llevará a cabo este rol. La cuenta de administrador delegado debe ser el único miembro de la unidad organizativa al que esté asignada la capacidad. La cuenta de administrador delegado no es necesaria si Change Manager se utiliza solo con una única Cuenta de AWS.

> **Important**
>
> Si utiliza Change Manager en toda una organización, se recomienda efectuar siempre los cambios desde la cuenta de administrador delegado. Si bien es posible realizar cambios desde otras cuentas de la organización, esos cambios no se notificarán ni se podrán ver desde la cuenta de administrador delegado.

Plantilla de cambios

Una plantilla de cambios es una colección de ajustes de configuración en Change Manager que definen aspectos tales como las aprobaciones requeridas, los manuales de procedimientos disponibles y las opciones de notificación para las solicitudes de cambio.

Puede requerir que las plantillas de cambios que creen los usuarios de su organización o cuenta pasen por un proceso de aprobación antes de que puedan utilizarse.

Change Manager admite dos tipos de plantillas de cambios. Para una solicitud de cambio aprobada que se basa en una plantilla de cambios de emergencia, el cambio solicitado se puede realizar

incluso si existen eventos de bloqueo en Change Calendar. Si se trata de una solicitud de cambio aprobada que se basa en una plantilla de cambios estándar, el cambio solicitado no se puede realizar si existen eventos de bloqueo en Change Calendar, a menos que se reciban aprobaciones adicionales por parte de los aprobadores de eventos de congelación de cambios designados.

Solicitud de cambio

Una solicitud de cambio se trata de una solicitud en Change Manager para ejecutar un manual de procedimientos de Automation que actualice uno o más recursos de sus entornos locales o en AWS. Una solicitud de cambio se crea usando una plantilla de cambios.

Cuando se crea una solicitud de cambio, uno o más aprobadores de su organización o cuenta deben revisar y aprobar la solicitud. Sin las aprobaciones necesarias, el flujo de trabajo del manual de procedimientos, que aplica los cambios solicitados, no puede ejecutarse.

En el sistema, las solicitudes de cambio son un tipo de OpsItem en AWS Systems Manager OpsCenter. Sin embargo, los OpsItems del tipo `/aws/changerequest` no se muestran en OpsCenter. Como OpsItems, las solicitudes de cambio están sujetas a las mismas cuotas obligatorias que otros tipos de OpsItems.

Además, para crear una solicitud de cambio mediante programación, no debe llamar a la operación `CreateOpsItem` de la API. En cambio, debe utilizar la operación `StartChangeRequestExecution` de la API. Pero en lugar de ejecutarse inmediatamente, la solicitud de cambio debe aprobarse y, además, no debe haber ningún evento de bloqueo en Change Calendar que evite que el flujo de trabajo se ejecute. La acción `StartChangeRequestExecution` recién se podrá completar cuando se hayan recibido las aprobaciones necesarias y el calendario no esté bloqueado (o se haya concedido permiso para omitir eventos de bloqueo del calendario).

Flujo de trabajo de manual de procedimientos

El flujo de trabajo del manual de procedimientos es el proceso de cambios solicitados que se realizan en los recursos de destino de su entorno en la nube o en las instalaciones. Cada solicitud de cambio designa un único manual de procedimientos de Automation que se debe utilizar para llevar a cabo el cambio solicitado. El flujo de trabajo del manual de procedimientos se lleva a cabo después de que se han otorgado todas las aprobaciones necesarias y si no hay eventos de bloqueo en Change Calendar. Si el cambio se ha programado para una fecha y hora específicas, el flujo de trabajo del manual de procedimientos no comienza sino hasta el momento programado, incluso si se han recibido todas las aprobaciones y el calendario no está bloqueado.

Temas

- Configuración de Change Manager
- Uso de Change Manager
- Auditoría y registro de la actividad de Change Manager
- Solución de problemas de Change Manager

Configuración de Change Manager

Puede utilizar Change Manager, una herramienta de AWS Systems Manager, para administrar los cambios de una organización completa, como esté configurada en AWS Organizations, o para una única Cuenta de AWS.

Si utiliza Change Manager con una organización, comience con el tema Configuración de Change Manager para una organización (cuenta de administración) y, luego, continúe con Configuración de opciones y prácticas recomendadas de Change Manager.

Si utiliza Change Manager con una única cuenta, diríjase directamente a Configuración de opciones y prácticas recomendadas de Change Manager.

> **Note**
>
> Si comienza a usar Change Manager con una única cuenta, pero esa cuenta se agrega posteriormente a una unidad organizativa para la cual Change Manager está permitido, no se tendrá en cuenta la configuración de su cuenta única.

Temas

- Configuración de Change Manager para una organización (cuenta de administración)
- Configuración de opciones y prácticas recomendadas de Change Manager
- Configuración de roles y permisos para Change Manager
- Control de acceso a flujos de trabajo de manual de procedimientos de aprobación automática

Configuración de Change Manager para una organización (cuenta de administración)

Las tareas de este tema se aplican si utiliza Change Manager, una herramienta de AWS Systems Manager, en una organización que está configurada en AWS Organizations. Si desea usar Change

Manager solo en una única Cuenta de AWS, pase al tema Configuración de opciones y prácticas recomendadas de Change Manager.

Lleve a cabo las tareas de esta sección en una Cuenta de AWS que sirva como la cuenta de administración en Organizations. Para obtener más información acerca de la cuenta de administración y otros conceptos de Organizations, consulte Terminología y conceptos de AWS Organizations.

Si necesita activar Organizations y especificar su cuenta como la cuenta de administración antes de continuar, consulte Creación y administración de una organización en la Guía del usuario de AWS Organizations.

> **Note**
>
> Este proceso de configuración no se puede efectuar en las siguientes Regiones de AWS:
>
> - UE (Milán) (eu-south-1)
> - Medio Oriente (Baréin) (me-south-1)
> - África (Ciudad del Cabo) (af-south-1)
> - Asia-Pacífico (Hong Kong) (ap-east-1)
>
> Asegúrese de trabajar en una región diferente en su cuenta de administración para este procedimiento.

Durante el procedimiento de configuración, debe llevar a cabo las siguientes tareas principales en Quick Setup, una herramienta de AWS Systems Manager.

- Tarea 1: registrar la cuenta de administrador delegado para su organización

  Las tareas relacionadas con el cambio que se llevan a cabo con Change Manager se administran en una de las cuentas miembro, que especifica que es la cuenta de administrador delegado. La cuenta de administrador delegado que se registra para Change Manager se convierte en la cuenta de administrador delegado para todas sus operaciones de Systems Manager. (Es posible que tenga cuentas de administrador delegado para otros Servicios de AWS). Su cuenta de administrador delegado para Change Manager, que no es la misma que la cuenta de administración, administra las actividades de cambio en toda la organización, incluidas las plantillas de cambios, las solicitudes de cambio y sus aprobaciones. En la cuenta de administrador

delegado, también puede especificar otras opciones de configuración para sus operaciones de Change Manager.

> **Important**
>
> La cuenta de administrador delegado debe ser el único miembro de la unidad organizativa al que se asigne la capacidad en Organizations.

- Tarea 2: definir y especificar las políticas de acceso del manual de procedimientos para los roles del solicitante de cambios o las funciones de trabajo personalizadas que desee utilizar para las operaciones de Change Manager

  Para crear solicitudes de cambio en Change Manager, los usuarios de sus cuentas miembro deben recibir permisos de AWS Identity and Access Management (IAM) que les permitan acceder solo a los manuales de procedimientos de Automation y las plantillas de cambios que usted elija que estén disponibles para ellos.

  > **Note**
  >
  > Cuando un usuario crea una solicitud de cambio, primero selecciona una plantilla de cambios. Esta plantilla de cambios puede tener varios manuales de procedimientos disponibles, pero el usuario solo puede seleccionar uno para cada solicitud de cambio. Las plantillas de cambios también se pueden configurar para permitir a los usuarios incluir cualquier manual de procedimientos disponible en sus solicitudes.

  Para otorgar los permisos necesarios, Change Manager utiliza el concepto de funciones de trabajo, que también es utilizado por IAM. Sin embargo, a diferencia de las políticas administradas de AWS para las funciones de trabajo en IAM, debe especificar tanto los nombres de las funciones de trabajo de Change Manager como los permisos de IAM para esas funciones de trabajo.

  Cuando vaya a configurar una función de trabajo, le recomendamos crear una política personalizada y proporcionar solo los permisos necesarios para llevar a cabo tareas de administración de cambios. Por ejemplo, podría especificar permisos que limiten a los usuarios a ese conjunto específico de manuales de procedimientos dependiendo de las funciones de trabajo que defina.

Por ejemplo, puede crear una función de trabajo con el nombre `DBAdmin`. Para esta función de trabajo, puede conceder solo los permisos necesarios para los manuales de procedimientos relacionados con las bases de datos de Amazon DynamoDB, como `AWS-CreateDynamoDbBackup` y `AWSConfigRemediation-DeleteDynamoDbTable`.

Como otro ejemplo, es posible que desee conceder a algunos usuarios solo los permisos necesarios para trabajar con manuales de procedimientos relacionados con los buckets de Amazon Simple Storage Service (Amazon S3), como `AWS-ConfigureS3BucketLogging` y `AWSConfigRemediation-ConfigureS3BucketPublicAccessBlock`.

El proceso de configuración en Quick Setup para Change Manager también pone a su disposición un conjunto de permisos administrativos completos de Systems Manager para que pueda aplicarlos a un rol administrativo que cree.

Cada configuración de Change Manager en Quick Setup que implemente creará una función de trabajo en su cuenta de administrador delegado con permisos para ejecutar plantillas de Change Manager y manuales de procedimientos de Automation en las unidades organizativas que haya seleccionado. Puede crear hasta 15 configuraciones de Quick Setup para Change Manager.

- Tarea 3: elegir qué cuentas miembro de su organización utilizar con Change Manager

  Puede utilizar Change Manager con todas las cuentas miembro de todas las unidades organizativas configuradas en Organizations y en todas las Regiones de AWS en las que funcionen. En cambio, si lo prefiere, puede usar Change Manager solo en algunas de sus unidades organizativas.

> **Important**
>
> Antes de comenzar este procedimiento, le recomendamos que lea sus pasos para comprender las opciones de configuración que va a elegir y los permisos que va a conceder. En particular, planifique las funciones de trabajo personalizadas que creará y los permisos que asignará a cada función de trabajo. Esto garantiza que cuando, más adelante, adjunte las políticas de función de trabajo que cree para los usuarios individuales, los grupos de usuarios o los roles de IAM, solo se les concedan los permisos que usted desea que tengan. Como práctica recomendada, comience configurando la cuenta de administrador delegado mediante el inicio de sesión para un administrador de Cuenta de AWS. A continuación,

configure las funciones de trabajo y sus permisos después de haber creado las plantillas de cambios e identificado los manuales de procedimientos que utilizará cada una.

Para configurar Change Manager para usarlo en una organización, lleve a cabo la siguiente tarea en el área de Quick Setup de la consola de Systems Manager.

Repita esta tarea para cada función de trabajo que desee crear para su organización. Cada función de trabajo que cree puede tener permisos para un conjunto diferente de unidades organizativas.

Para configurar una organización para Change Manager en la cuenta de administración de Organizations

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Quick Setup.
3. En la tarjeta Change Manager, seleccione crear.
4. En Delegated administrator account (Cuenta de administrador delegado), ingrese el ID de la Cuenta de AWS que desea utilizar para administrar las plantillas de cambios, las solicitudes de cambio y los flujos de trabajo del manual de procedimientos en Change Manager.

   Si con anterioridad ha especificado una cuenta de administrador delegado para Systems Manager, su ID ya se indicará en este campo.

   Important

   La cuenta de administrador delegado debe ser el único miembro de la unidad organizativa al que se asigne la capacidad en Organizations.
   Si la cuenta de administrador delegado registrada luego se anula de ese rol, el sistema elimina sus permisos para administrar las operaciones de Systems Manager al mismo tiempo. Tenga en cuenta que será necesario regresar a Quick Setup, designar una cuenta de administrador delegado diferente y volver a especificar todas las funciones de trabajo y los permisos.
   Si utiliza Change Manager en toda una organización, se recomienda efectuar siempre los cambios desde la cuenta de administrador delegado. Si bien es posible realizar cambios desde otras cuentas de la organización, esos cambios no se notificarán ni se podrán ver desde la cuenta de administrador delegado.

5. En la sección Permissions to request and make changes (Permisos para solicitar y realizar cambios), haga lo siguiente.

> **Note**
>
> Cada configuración de implementación que cree proporcionará la política de permisos para una sola función de trabajo. Puede regresar a Quick Setup más adelante para crear más funciones de trabajo cuando haya creado plantillas de cambios para utilizarlas en sus operaciones.

Para crear un rol administrativo: para una función de trabajo de administrador que tenga permisos de IAM para todas las acciones de AWS, realice lo siguiente.

> **Important**
>
> El otorgamiento de permisos administrativos completos a los usuarios debe realizarse con moderación y solo si sus roles requieren acceso completo a Systems Manager. Para obtener información importante acerca de los aspectos que deben tenerse en cuenta sobre la seguridad en el acceso a Systems Manager, consulte Administración de identidades y accesos en AWS Systems Manager y Prácticas recomendadas de seguridad para Systems Manager.

1. En Job function (Función de trabajo), ingrese un nombre para identificar este rol y sus permisos, como **`MyAWSAdmin`**.
2. En Role and permissions option (Opción de rol y permisos), elija Administrator permissions (Permisos de administrador).

Para crear otras funciones de trabajo: para crear un rol no administrativo, haga lo siguiente.

1. En Job function (Función de trabajo), ingrese un nombre para identificar este rol y sugerir sus permisos. El nombre que elija debe representar el alcance de los manuales de procedimientos para los que proporcionará permisos, como `DBAdmin` o `S3Admin`.
2. En Role and permissions option (Opción de rol y permisos), elija Custom permissions (Permisos personalizados).

3. En Permissions policy editor (Editor de políticas de permisos), ingrese los permisos de IAM, en formato JSON, que se concederán a esta función de trabajo.

> **Tip**
>
> Le recomendamos utilizar el editor de políticas de IAM para diseñar la política y, luego, pegar la política JSON en el campo Permissions policy (Política de permisos).

Política de muestra: administración de bases de datos de DynamoDB

Por ejemplo, puede comenzar con el contenido de la política que proporcione permisos para trabajar con los documentos de Systems Manager (documentos de SSM) a los que necesita acceder la función de trabajo. A continuación, se presenta un contenido de política de muestra que otorga acceso a todos los manuales de procedimientos de Automation administrados por AWS que se relacionen con las bases de datos de DynamoDB y dos plantillas de cambios que se han creado en la Cuenta de AWS `123456789012` de ejemplo, en la región Este de EE. UU. (Ohio) (`us-east-2`).

La política también incluye permisos para la operación StartChangeRequestExecution, que es necesaria para crear una solicitud de cambio en Change Calendar.

> **Note**
>
> Este ejemplo no es exhaustivo. Es posible que se necesiten permisos adicionales para trabajar con otros recursos de AWS, como las bases de datos y los nodos.

JSON

```
{
    "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:111122223333:document/
MyFirstDBChangeTemplate",
                "arn:aws:ssm:region:111122223333:document/
MySecondDBChangeTemplate"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "ssm:ListDocuments",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ssm:StartChangeRequestExecution",
            "Resource": "arn:aws:ssm:region:111122223333:automation-
definition/*:*"
        }
    ]
}
```

Para obtener más información acerca de las políticas de IAM, consulte Administración del acceso a los recursos de AWS y Creación de políticas de IAM en la Guía del usuario de IAM.

6. En la sección Targets (Destinos), elija si desea conceder permisos para la función de trabajo que está creando a toda la organización o solo a algunas de sus unidades organizativas.

   Si elige Entire organization (Toda la organización), continúe con el paso 9.

   Si elige Custom (Personalizar), continúe con el paso 8.
7. En la sección Target OUs (Unidades organizativas de destino), seleccione las casillas de verificación de las unidades organizativas en las que se utilizará Change Manager.
8. Seleccione Crear.

Después de que el sistema termine de configurar Change Manager para su organización, mostrará un resumen de las implementaciones. Esta información de resumen incluye el nombre del rol que se creó para la función de trabajo que configuró. Por ejemplo, `AWS-QuickSetup-SSMChangeMgr-DBAdminInvocationRole`.

> **Note**
>
> Quick Setup utiliza StackSets de AWS CloudFormation para implementar las configuraciones. También puede ver información acerca de una configuración de implementación completada en la consola de AWS CloudFormation. Para obtener más información acerca de StackSets, consulte Uso de StackSets de AWS CloudFormation en la Guía del usuario de AWS CloudFormation.

El siguiente paso consiste en configurar opciones de Change Manager adicionales. Puede completar esta tarea en su cuenta de administrador delegado o en cualquier cuenta de una unidad organizativa a la que haya permitido usar Change Manager. Puede configurar opciones, como elegir una opción de administración de identidades de usuarios, especificar qué usuarios pueden revisar y aprobar o rechazar las plantillas y las solicitudes de cambios, y elegir qué opciones de prácticas recomendadas se permitirán para su organización. Para obtener más información, consulta Configuración de opciones y prácticas recomendadas de Change Manager.

Configuración de opciones y prácticas recomendadas de Change Manager

Las tareas de esta sección se deben llevar a cabo independientemente de si Change Manager, una herramienta de AWS Systems Manager, se utiliza en toda una organización o en una única Cuenta de AWS.

Si utiliza Change Manager para una organización, puede realizar las siguientes tareas en su cuenta de administrador delegado o en cualquier cuenta de una unidad organizativa a la que haya permitido que use Change Manager.

Temas

- Tarea 1: configurar la administración de identidades de usuarios y los revisores de plantillas de Change Manager
- Tarea 2: configurar los aprobadores de eventos de congelación de cambios y las prácticas recomendadas de Change Manager
- Configuración de temas de Amazon SNS para las notificaciones de Change Manager

Tarea 1: configurar la administración de identidades de usuarios y los revisores de plantillas de Change Manager

Realice la tarea de este procedimiento la primera vez que acceda a Change Manager. Puede actualizar estos ajustes de configuración más adelante regresando a Change Manager y eligiendo Edit (Editar) en la pestaña Settings (Configuración).

Para configurar la administración de identidades de usuarios y los revisores de plantillas de Change Manager

1. Inicie sesión en la AWS Management Console.

   Si utiliza Change Manager en una organización, inicie sesión con las credenciales para su cuenta de administrador delegado. El usuario debe tener los permisos de AWS Identity and Access Management (IAM) necesarios para efectuar actualizaciones en la configuración de Change Manager.
2. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
3. En el panel de navegación, elija Change Manager.
4. En la página de inicio del servicio, en función de las opciones disponibles, lleve a cabo alguna de las siguientes operaciones:

   - Si utiliza Change Manager con AWS Organizations, elija Set up delegated account (Configurar una cuenta delegada).
   - Si utiliza Change Manager con una única Cuenta de AWS, elija Set up Change Manager (Configurar Change Manager).

-o bien-

Seleccione Create sample change request (Crear solicitud de cambio de muestra), Skip (Omitir) y, luego, la pestaña Settings (Configuración).

5. En User identity management (Administración de identidades de usuarios), elija alguna de las siguientes opciones.

   - AWS Identity and Access Management (IAM): identifique a los usuarios que realizan y aprueban solicitudes y llevan a cabo otras acciones en Change Manager mediante el uso de usuarios, grupos y roles existentes.
   - AWS IAM Identity Center (Centro de identidades de IAM): permita que el Centro de identidades de IAM cree y administre identidades o conéctese a su origen de identidades existente para identificar a los usuarios que llevan a cabo acciones en Change Manager.

6. En la sección notificación del revisor de plantillas, especifique los temas de Amazon Simple Notification Service (Amazon SNS) que se utilizarán para notificar a los revisores de plantillas que una nueva plantilla de cambios o versión de plantilla de cambios está lista para revisarse. Asegúrese de que el tema de Amazon SNS que elija esté configurado para enviar notificaciones a los revisores de plantillas.

   Para obtener más información acerca de cómo crear y configurar temas de Amazon SNS para efectuar notificaciones a los revisores de plantillas de cambios, consulte Configuración de temas de Amazon SNS para las notificaciones de Change Manager.

   1. Para especificar el tema de Amazon SNS que se utilizará para la notificación a los revisores de plantillas, elija una de las siguientes opciones:
      - Ingrese un nombre de recurso de Amazon (ARN) de SNS: en Topic ARN (ARN de tema), ingrese el ARN de un tema de Amazon SNS existente. Este tema puede estar en cualquiera de las cuentas de su organización.
      - Seleccione un tema de SNS existente: en Target notification topic (Tema de notificaciones de destino), seleccione el ARN de un tema de Amazon SNS existente en su Cuenta de AWS actual. (Esta opción no estará disponible si aún no ha creado ningún tema de Amazon SNS en su Cuenta de AWS y Región de AWS actuales).

      > **Note**
      >
      > El tema de Amazon SNS que seleccione debe estar configurado para especificar las notificaciones que envía y los suscriptores a los que se las envía. Su política

de acceso también debe conceder permisos a Systems Manager para que Change Manager pueda enviar notificaciones. Para obtener más información, consulta Configuración de temas de Amazon SNS para las notificaciones de Change Manager.

2. Seleccione Agregar notificación.

7. En la sección Change template reviewers (Revisores de plantillas de cambios), seleccione los usuarios de su organización o cuenta que revisarán las nuevas plantillas de cambios o versiones de plantillas de cambios antes de que se puedan utilizar en sus operaciones.

   Los revisores de plantillas de cambios son responsables de verificar la idoneidad y la seguridad de las plantillas que otros usuarios han enviado para su uso en los flujos de trabajo del manual de procedimientos de Change Manager.

   Seleccione los revisores de plantillas de cambios haciendo lo siguiente:

   1. Elija Agregar.
   2. Seleccione la casilla de verificación situada junto al nombre de cada usuario, grupo o rol de IAM que desee designar como revisor de plantillas de cambios.
   3. Elija Add approvers (Agregar aprobadores).

8. Elija Enviar.

Después de completar este proceso de configuración inicial, establezca los ajustes adicionales y las prácticas recomendadas de Change Manager siguiendo los pasos que se indican en Tarea 2: configurar los aprobadores de eventos de congelación de cambios y las prácticas recomendadas de Change Manager.

Tarea 2: configurar los aprobadores de eventos de congelación de cambios y las prácticas recomendadas de Change Manager

Luego de completar los pasos que se indican en Tarea 1: configurar la administración de identidades de usuarios y los revisores de plantillas de Change Manager, puede designar revisores adicionales para las solicitudes de cambio que se efectúen durante los eventos de congelación de cambios y especificar qué prácticas recomendadas disponibles desea permitir para sus operaciones de Change Manager.

Un evento de congelación de cambios significa que existen restricciones en el calendario de cambios actual (el estado del calendario en AWS Systems Manager Change Calendar es `CLOSED`). En estos casos, además de los aprobadores regulares de solicitudes de cambio, o cuando la solicitud de

cambio se crea utilizando una plantilla que permite las aprobaciones automáticas, se requiere que los aprobadores de congelación de cambios concedan permiso para que esta solicitud de cambio se ejecute. Si no lo hacen, el cambio no se procesará hasta que el estado del calendario vuelva a ser OPEN.

Para configurar los aprobadores de eventos de congelación de cambios y las prácticas recomendadas de Change Manager

1. En el panel de navegación, elija Change Manager.
2. Elija la pestaña Settings (Configuración) y, luego, Edit (Editar).
3. En la sección Approvers for change freeze events (Aprobadores de eventos de congelación de cambios), seleccione los usuarios de su organización o cuenta que podrán aprobar los cambios para que se ejecuten incluso cuando el calendario en uso de Change Calendar esté CERRADO en ese momento.

   > **Note**
   >
   > Para permitir las revisiones de congelación de cambios, debe activar la opción Check Change Calendar for restricted change events (Verificar Change Calendar para detectar eventos de cambios restringidos) en Best practices (Prácticas recomendadas).

   Seleccione los aprobadores de los eventos de congelación de cambios haciendo de la siguiente manera:

   1. Elija Agregar.
   2. Seleccione la casilla de verificación situada junto al nombre de cada usuario, grupo o rol de IAM que desee designar como aprobador de eventos de congelación de cambios.
   3. Elija Add approvers (Agregar aprobadores).
4. En la sección Best practices (Prácticas recomendadas) ubicada cerca de la parte inferior de la página, active las prácticas recomendadas que desee aplicar a cada una de las siguientes opciones.

   - Opción: check Change Calendar for restricted change events (Verificar Change Calendar para detectar eventos de cambios restringidos)

Para especificar que Change Manager verifique un calendario de Change Calendar para asegurarse de que los cambios no estén bloqueados por eventos programados, primero seleccione la casilla de verificación Enabled (Habilitado) y, a continuación, seleccione el calendario para verificar si hay eventos restringidos en la lista Change Calendar.

Para obtener más información acerca de Change Calendar, consulte AWS Systems Manager Change Calendar.

- Opción: SNS topic for approvers for closed events (Tema de SNS para aprobadores de eventos cerrados)

  1. Elija una de las siguientes opciones para especificar el tema de Amazon Simple Notification Service (Amazon SNS) de su cuenta que se utilizará para enviar notificaciones a los aprobadores durante los eventos de congelación de cambios. (Tenga en cuenta que también debe especificar aprobadores en la sección Approvers for change freeze events [Aprobadores de eventos de congelación de cambios] anterior a Best practices [Prácticas recomendadas]).

     - Ingrese un nombre de recurso de Amazon (ARN) de SNS: en Topic ARN (ARN de tema), ingrese el ARN de un tema de Amazon SNS existente. Este tema puede estar en cualquiera de las cuentas de su organización.
     - Seleccione un tema de SNS existente: en Target notification topic (Tema de notificaciones de destino), seleccione el ARN de un tema de Amazon SNS existente en su Cuenta de AWS actual. (Esta opción no estará disponible si aún no ha creado ningún tema de Amazon SNS en su Cuenta de AWS y Región de AWS actuales).

     > **Note**
     >
     > El tema de Amazon SNS que seleccione debe estar configurado para especificar las notificaciones que envía y los suscriptores a los que se las envía. Su política de acceso también debe conceder permisos a Systems Manager para que Change Manager pueda enviar notificaciones. Para obtener más información, consulta Configuración de temas de Amazon SNS para las notificaciones de Change Manager.

  2. Seleccione Agregar notificación.

- Opción: Require monitors for all templates (Requerir monitores para todas las plantillas)

Si desea asegurarse de que todas las plantillas de su organización o cuenta especifican una alarma de Amazon CloudWatch para monitorear su operación de cambio, seleccione la casilla de verificación Enabled (Habilitado).

- Opción: Require template review and approval before use (Requerir revisión y aprobación de las plantillas antes de usarlas)

  Para asegurarse de que no se creen solicitudes de cambio ni se ejecuten flujos de trabajo de manual de procedimientos sin tener como base una plantilla que se haya revisado y aprobado, seleccione la casilla de verificación Enabled (Habilitado).

5. Seleccione Save.

Configuración de temas de Amazon SNS para las notificaciones de Change Manager

Puede configurar Change Manager, una herramienta de AWS Systems Manager, para que envíe notificaciones a un tema de Amazon Simple Notification Service (Amazon SNS) por los eventos relacionados con las solicitudes y las plantillas de cambios. Lleve a cabo las siguientes tareas para recibir notificaciones por los eventos de Change Manager a los que agregue un tema.

Temas

- Tarea 1: crear un tema de Amazon SNS y suscribirse a él
- Tarea 2: actualizar la política de acceso de Amazon SNS
- Tarea 3: (Opcional) Actualizar la política de acceso de AWS Key Management Service

Tarea 1: crear un tema de Amazon SNS y suscribirse a él

En primer lugar, debe crear un tema de Amazon SNS y suscribirse a él. Para obtener más información, consulte Creating a Amazon SNS topic (Creación de un tema de Amazon SNS) y Subscribing to an Amazon SNS topic (Suscripción a un tema de Amazon SNS) en la Guía para desarrolladores de Amazon Simple Notification Service.

> **Note**
>
> Para recibir notificaciones, debe especificar el nombre de recurso de Amazon (ARN) de un tema de Amazon SNS que se encuentre en la misma Región de AWS y Cuenta de AWS que la cuenta de administrador delegado.

Tarea 2: actualizar la política de acceso de Amazon SNS

Utilice el siguiente procedimiento para actualizar la política de acceso de Amazon SNS de modo que Systems Manager pueda publicar las notificaciones de Change Manager en el tema de Amazon SNS que creó en la tarea 1. Si no completa esta tarea, Change Manager no tendrá permiso para enviar las notificaciones de los eventos para los que agrega el tema.

1. Inicie sesión en la AWS Management Console y abra la consola de Amazon SNS en https://console.aws.amazon.com/sns/v3/home.
2. En el panel de navegación, elija Temas.
3. Elija el tema que creó en la tarea 1 y, a continuación, elija Edit (Editar).
4. Expanda Política de acceso.
5. Agregue y actualice el siguiente bloque `Sid` a la política existente y sustituya cada *`espacio disponible de entrada del usuario`* con su propia información.

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

   Ingrese este bloque después del bloque `Sid` existente, y reemplace *`region`*, *`account-id`* y *`topic-name`* con los valores apropiados para el tema que ha creado.
6. Seleccione Save changes (Guardar cambios).

El sistema ahora enviará notificaciones al tema de Amazon SNS cuando se produzca el tipo de evento para el que se agregó el tema.

> **Important**
>
> Si configuró el tema de Amazon SNS con una clave de cifrado de AWS Key Management Service (AWS KMS) del lado del servidor, debe completar la tarea 3.

Tarea 3: (Opcional) Actualizar la política de acceso de AWS Key Management Service

Si activó el cifrado de AWS Key Management Service (AWS KMS) del lado del servidor para el tema de Amazon SNS, también debe actualizar la política de acceso de la AWS KMS key que eligió cuando configuró el tema. Siga el siguiente procedimiento para actualizar la política de acceso de modo que Systems Manager pueda publicar las notificaciones de aprobaciones de Change Manager en el tema de Amazon SNS que creó en la tarea 1.

1. Abra la consola de AWS KMS en https://console.aws.amazon.com/kms.
2. En el panel de navegación, elija Claves administradas por el cliente.
3. Elija el ID de la clave administrada por el cliente que eligió cuando creó el tema.
4. En la sección Key policy (Política de claves), elija Switch to policy view (Cambiar a la vista de política).
5. Seleccione Editar.
6. Escriba el siguiente bloque `Sid` después de uno de los bloques `Sid` en la política existente. Reemplace cada uno *`marcador de posición del usuario`* con información propia.

```
{
    "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. Ahora, escriba el siguiente bloque `Sid` después de uno de los bloques `Sid` en la política de recursos para ayudar a evitar el problema del suplente confuso entre servicios.

   Este bloque utiliza claves de contexto de condición global `aws:SourceArn` y `aws:SourceAccount` para limitar los permisos que Systems Manager otorga a otro servicio al recurso.

   Reemplace cada uno *`marcador de posición del usuario`* con información propia.

   JSON

```
{
    "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:us-east-1:111122223333:topic-name",
            "Condition": {
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:ssm:us-east-1:111122223333:*"
                },
                "StringEquals": {
                    "aws:SourceAccount": "111122223333"
                }
            }
        }
    ]
}
```

8. Seleccione Save changes (Guardar cambios).

Configuración de roles y permisos para Change Manager

De manera predeterminada, Change Manager no tiene permiso para realizar acciones en los recursos. Debe conceder acceso mediante un rol de servicio de AWS Identity and Access Management (IAM), o rol de asunción. Este rol permite que Change Manager pueda ejecutar de forma segura los flujos de trabajo de manuales de procedimientos especificados en una solicitud de cambio aprobada en su nombre. El rol concede a AWS Security Token Service (AWS STS) la confianza AssumeRole para Change Manager.

Proporcionando estos permisos a un rol para que actúe en nombre de los usuarios de una organización, ya no es necesario conceder ese conjunto de permisos como tal a los usuarios. Las acciones permitidas por los permisos están limitadas únicamente a las operaciones aprobadas.

Cuando los usuarios de la cuenta u organización crean una solicitud de cambio, pueden seleccionar este rol de asunción para realizar las operaciones de cambio.

Puede crear un nuevo rol de asunción para Change Manager, o bien actualizar un rol existente con los permisos necesarios.

Si necesita crear un rol de servicio para Change Manager, complete las siguientes tareas.

Tareas

- Tarea 1: Creación de una política de rol de asunción para Change Manager
- Tarea 2: Creación de un rol de asunción para Change Manager
- Tarea 3: Adición de la política iam:PassRole a otros roles
- Tarea 4: Adición de políticas insertadas a un rol de asunción para invocar a otros Servicios de AWS
- Tarea 5: Configuración del acceso de usuario a Change Manager

Tarea 1: Creación de una política de rol de asunción para Change Manager

Utilice el siguiente procedimiento para crear la política que va a adjuntar al rol de asunción de Change Manager.

Para crear una política de rol de asunción para Change Manager

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, seleccione Policies y, a continuación, seleccione Create Policy.

3. En la página Create policy (Crear política), elija la pestaña JSON y reemplace el contenido predeterminado con el siguiente, que modificará para sus propias operaciones de Change Manager en los siguientes pasos.

> **Note**
>
> Si va a crear una política para utilizarla con una sola Cuenta de AWS, no una organización con varias cuentas y Regiones de AWS, puede omitir el primer bloque de instrucciones. No se requiere el permiso `iam:PassRole` en el caso de una sola cuenta que utiliza Change Manager.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::111122223333: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:us-east-1:111122223333:automation-
definition/template-name:$DEFAULT",
                "arn:aws:ssm:us-east-1::document/template-name"
            ]
        },
```

```
        {
            "Effect": "Allow",
            "Action": [
                "ssm:ListOpsItemEvents",
                "ssm:GetOpsItem",
                "ssm:ListDocuments",
                "ssm:DescribeOpsItems"
            ],
            "Resource": "*"
        }
    ]
}
```

4. Actualice el valor `Resource` de la acción `iam:PassRole` para incluir los ARN de todas las funciones laborales definidas para la organización a las que desee conceder permisos para iniciar flujos de trabajo de manuales de procedimientos.
5. Reemplace los marcadores de posición *`region`*, *`account-id`*, *`template-name`*, *`delegated-admin-account-id`* y *`job-function`* con valores para sus operaciones de Change Manager.
6. Modifique la lista de la segunda instrucción `Resource` para incluir todas las plantillas de cambio a las que desee conceder permisos. Como alternativa, puede especificar `"Resource": "*"` para conceder permisos a todas las plantillas de cambio de la organización.
7. Elija Siguiente: etiquetas.
8. (Opcional) Agregue uno o varios pares de valor etiqueta-clave para organizar, realizar un seguimiento o controlar el acceso a esta política.
9. Elija Siguiente: Revisar.
10. En la página Review policy (Revisar política), ingrese un nombre en el cuadro Name (Nombre), como **`MyChangeManagerAssumeRole`**, y luego ingrese una descripción opcional.
11. Elija Create policy (Crear política) y vaya a Tarea 2: Creación de un rol de asunción para Change Manager.

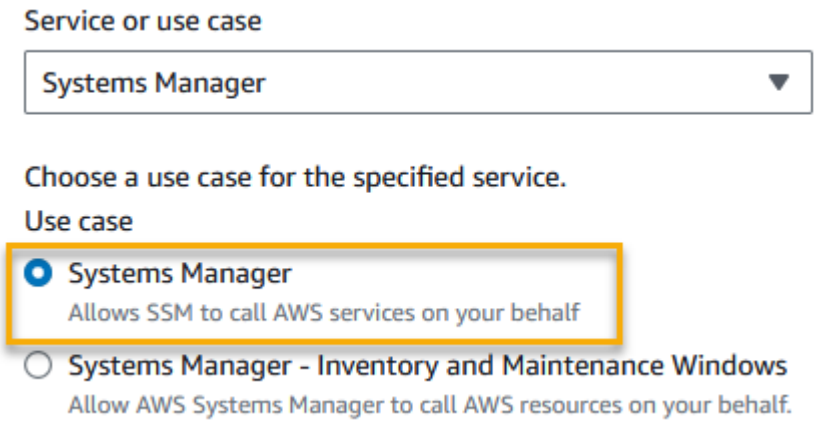
Tarea 2: Creación de un rol de asunción para Change Manager

Siga este procedimiento para crear un rol de asunción de Change Manager, un tipo de rol de servicio, para Change Manager.

Para crear un rol de asunción para Change Manager

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.

2. En el panel de navegación, seleccione Roles y luego seleccione Crear rol.
3. En Select trusted entity (Seleccionar entidad de confianza), realice las siguientes elecciones:
    1. En Trusted entity type (Tipo de entidad de confianza), elija AWS service (Servicio de)
    2. En Use cases for other Servicios de AWS, (Casos de uso de otros), elija Systems Manager
    3. Elija Systems Manager, como aparece en la siguiente imagen.



4. Elija Siguiente.
5. En la página Attached permissions policy (Política de permisos adjuntos), busque la política de rol de asunción que haya creado en Tarea 1: Creación de una política de rol de asunción para Change Manager, como, por ejemplo, **`MyChangeManagerAssumeRole`**.
6. Seleccione la casilla de verificación situada junto al nombre de la política de rol de asunción, y luego elija Next: Tags (Siguiente: Etiquetas).
7. En Role name (Nombre del rol), ingrese un nombre para el nuevo perfil de instancias, por ejemplo, **`MyChangeManagerAssumeRole`**.
8. (Opcional) En Description (Descripción), actualice la descripción de este rol de instancia.
9. (Opcional) Agregue uno o varios pares de valor etiqueta-clave para organizar, realizar un seguimiento o controlar el acceso a este rol.
10. Elija Siguiente: Revisar.
11. (Opcional) En Tags (Etiquetas), agregue uno o varios pares de valores etiqueta-clave para organizar, seguir o controlar el acceso a este rol, y luego elija Create role (Crear rol). El sistema le devuelve a la página Roles.
12. Elija Create role. El sistema le devuelve a la página Roles.
13. En la página Roles, elija el rol que acaba de crear para abrir la página Summary (Resumen).

Tarea 3: Adición de la política **`iam:PassRole`** a otros roles

Utilice el siguiente procedimiento para adjuntar la política `iam:PassRole` a un perfil de instancias de IAM o rol de servicio de IAM. (El servicio Systems Manager utiliza perfiles de instancia de IAM para comunicarse con instancias de EC2. En el caso de los nodos administrados que no son de EC2 en un entorno híbrido y multinube, se utiliza en su lugar un rol de servicio de IAM).

Al adjuntar la política `iam:PassRole`, el servicio Change Manager puede transferir permisos de rol de asunción a otros servicios o herramientas de Systems Manager cuando se ejecutan flujos de trabajo de manuales de procedimientos.

Para adjuntar la política **`iam:PassRole`** a un perfil de instancia o rol de servicio de IAM

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. Seleccione Roles en el panel de navegación.
3. Busque el rol de asunción Change Manager que haya creado, como, por ejemplo, **`MyChangeManagerAssumeRole`**, y elija su nombre.
4. En la página Summary (Resumen) del rol de asunción, elija la pestaña Permissions (Permisos).
5. Elija Add permissions, Create inline policy (Agregar permisos, Crear política insertada).
6. En la página Create policy (Crear política), elija la pestaña Visual editor (Editor visual).
7. Elija Service (Servicio) y, a continuación, IAM.
8. En el cuadro de texto Filter actions (Filtrar acciones), ingrese **`PassRole`**, y luego elija la opción PassRole.
9. Expanda Resources (Recursos). Compruebe que esté seleccionado Specific (Específicos) y elija Add ARN (Añadir ARN).
10. En el campo Specify ARN for role (Especificar ARN para el rol), ingrese el ARN del rol de perfil de instancias de IAM o del rol de servicio de IAM al que desee transferir permisos de rol de asunción. El sistema rellena los campos Account (Cuenta) y Role name with path (Nombre del rol con ruta).
11. Elija Add (Agregar).
12. Elija Review policy (Revisar política).
13. En Name (Nombre), ingrese un nombre para identificar esta política, y luego elija Create policy (Crear política).

Más información

- Configuración de permisos de instancia requeridos para Systems Manager
- Creación del rol de servicio de IAM requerido para Systems Manager en entornos híbridos y multinube

Tarea 4: Adición de políticas insertadas a un rol de asunción para invocar a otros Servicios de AWS

Si una solicitud de cambio invoca otros Servicios de AWS mediante el rol de asunción Change Manager, este debe tener permiso para invocar esos servicios. Este requisito se aplica a todos los manuales de procedimientos de AWS Automation (manuales de procedimientos AWS-*) que se puedan utilizar en una solicitud de cambio, como los manuales de procedimientos `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup` y `AWS-RestartEC2Instance`. Este requisito también se aplica a cualquier manual de procedimientos personalizado que cree para invocar otros Servicios de AWS mediante acciones que llaman a otros servicios. Por ejemplo, si utiliza las acciones `aws:executeAwsApi`, `aws:CreateStack` o `aws:copyImage`, debe configurar el rol de servicio con el permiso necesario para invocar esos servicios. Puede habilitar permisos para otros Servicios de AWS si agrega una política insertada de IAM al rol.

Para agregar una política insertada a un rol de asunción para invocar otros Servicios de AWS (consola de IAM)

1. Inicie sesión en AWS Management Console y abra la consola IAM en https://console.aws.amazon.com/iam/.
2. Seleccione Roles en el panel de navegación.
3. En la lista, elija el nombre del rol de asunción que desee actualizar, como, por ejemplo, `MyChangeManagerAssumeRole`.
4. Elija la pestaña Permisos.
5. Elija Add permissions, Create inline policy (Agregar permisos, Crear política insertada).
6. Seleccione la pestaña JSON.
7. Ingrese un documento de política JSON para los Servicios de AWS que desee invocar. A continuación se muestran dos documentos de política JSON a modo de ejemplo.

   Ejemplo de **`PutObject`** y **`GetObject`** de Amazon S3

JSON

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

Ejemplo de **`CreateSnapshot`** y **`DescribeSnapShots`** de Amazon EC2

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"ec2:CreateSnapshot",
         "Resource":"*"
      },
      {
         "Effect":"Allow",
         "Action":"ec2:DescribeSnapshots",
         "Resource":"*"
      }
   ]
}
```

Para obtener información sobre el lenguaje de las políticas de IAM, consulte Referencia de políticas JSON de IAM en la Guía del usuario de IAM.

8. Cuando haya terminado, elija Review policy (Revisar política). El validador de políticas notifica los errores de sintaxis.
9. En Name (Nombre), ingrese un nombre para identificar la política que está creando. Revise el Summary (Resumen) de la política para ver los permisos concedidos por su política. A continuación, elija Create policy (Crear política) para guardar su trabajo.
10. Una vez que cree una política insertada, se integra de manera automática a su rol.

Tarea 5: Configuración del acceso de usuario a Change Manager

Si el usuario, grupo o rol tiene asignados permisos de administrador, entonces tiene acceso a Change Manager. Si no tiene permisos de administrador, un administrador debe asignar la política administrada `AmazonSSMFullAccess` o una política que proporcione permisos comparables a su usuario, grupo o rol.

Utilice el siguiente procedimiento a fin de configurar un usuario para utilizar Change Manager. El usuario que elija tendrá permiso para configurar y ejecutar Change Manager.

Según la aplicación de identidad que utilice en su organización, puede seleccionar cualquiera de las tres opciones disponibles para configurar el acceso del usuario. Al configurar el acceso del usuario, asigne o agregue lo siguiente:

1. Asigne la política `AmazonSSMFullAccess` o una política comparable que proporcione permiso para acceder a Systems Manager.
2. Asigne la política `iam:PassRole`.
3. Agregue el ARN del rol de asunción de Change Manager que copió al final de Tarea 2: Creación de un rol de asunción para Change Manager.

Para dar acceso, agregue permisos a los usuarios, grupos o roles:

- Usuarios y grupos en AWS IAM Identity Center:

  Cree un conjunto de permisos. Siga las instrucciones de Creación de un conjunto de permisos en la Guía del usuario de AWS IAM Identity Center.
- Usuarios gestionados en IAM a través de un proveedor de identidades:

Cree un rol para la federación de identidades. Siga las instrucciones descritas en Creación de un rol para un proveedor de identidad de terceros (federación) en la Guía del usuario de IAM.

- Usuarios de IAM:
  - Cree un rol que el usuario pueda aceptar. Siga las instrucciones descritas en Creación de un rol para un usuario de IAM en la Guía del usuario de IAM.
  - (No recomendado) Adjunte una política directamente a un usuario o añada un usuario a un grupo de usuarios. Siga las instrucciones descritas en Adición de permisos a un usuario (consola) de la Guía del usuario de IAM.

Ha terminado de configurar los roles requeridos para Change Manager. A partir de ahora, puede utilizar el ARN del rol de asunción Change Manager en las operaciones de Change Manager.

Control de acceso a flujos de trabajo de manual de procedimientos de aprobación automática

En cada plantilla de cambios creada para su organización o cuenta, puede especificar si las solicitudes de cambio que se creen a partir de esa plantilla pueden ejecutarse como solicitudes de cambio de aprobación automática, lo que significa que se ejecutan automáticamente sin ningún paso de revisión (con la excepción de los eventos de congelación de cambios).

Sin embargo, es posible que desee evitar que ciertos usuarios, grupos o roles de AWS Identity and Access Management (IAM) ejecuten solicitudes de cambio de aprobación automática, incluso si una plantilla de cambios lo permite. Puede hacerlo mediante el uso de la clave de condición `ssm:AutoApprove` para la operación `StartChangeRequestExecution` en una política de IAM asignada al usuario, al grupo o al rol de IAM.

Puede agregar la siguiente política como una política insertada, donde la condición se especifica como `false`, para evitar que los usuarios ejecuten solicitudes de cambio de aprobación automática.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
            {
            "Effect": "Allow",
            "Action": "ssm:StartChangeRequestExecution",
            "Resource": "*",
```

```
            "Condition": {
                "BoolIfExists": {
                    "ssm:AutoApprove": "false"
                }
            }
        }
    ]
}
```

Para obtener más información acerca de cómo especificar políticas insertadas, consulte Políticas insertadas y Agregado y eliminación de permisos de identidad de IAM en la Guía del usuario de IAM.

Para obtener más información sobre las claves de condición de las políticas de Systems Manager, consulteClaves de condición de Systems Manager.

Uso de Change Manager

Con Change Manager, una herramienta de AWS Systems Manager, los usuarios de toda la organización o de una única Cuenta de AWS pueden llevar a cabo tareas relacionadas con los cambios para las que se les hayan otorgado los permisos necesarios. Las tareas de Change Manager son las siguientes:

- Cree, revise y apruebe o rechace las plantillas de cambios.

  Una plantilla de cambios es una colección de ajustes de configuración en Change Manager que definen aspectos tales como las aprobaciones requeridas, los manuales de procedimientos disponibles y las opciones de notificación para las solicitudes de cambio.
- Cree, revise y apruebe o rechace las solicitudes de cambios.

  Una solicitud de cambio se trata de una solicitud en Change Manager para ejecutar un manual de procedimientos de Automation que actualice uno o más recursos de sus entornos locales o en AWS. Una solicitud de cambio se crea usando una plantilla de cambios.
- Especifique qué usuarios de su organización o cuenta se pueden convertir en revisores de las plantillas y las solicitudes de cambios.
- Edite los ajustes de configuración, como, por ejemplo, cómo se administran las identidades de usuarios en Change Manager y cuáles de las opciones disponibles de prácticas recomendadas se aplican a sus operaciones de Change Manager. Para obtener más información acerca de la configuración de estos ajustes, consulte Configuración de opciones y prácticas recomendadas de Change Manager.

Temas

- Uso de las plantillas de cambios
- Uso de solicitudes de cambio
- Revisión de los detalles, las tareas y los plazos de las solicitudes de cambio (consola)
- Visualización de recuentos agregados de solicitudes de cambio (línea de comandos)

Uso de las plantillas de cambios

Una plantilla de cambios es una colección de ajustes de configuración en Change Manager que definen aspectos tales como las aprobaciones requeridas, los manuales de procedimientos disponibles y las opciones de notificación para las solicitudes de cambio.

> **Note**
>
> AWS proporciona una muestra de la plantilla de cambios Hello World que puede utilizar para probar Change Manager, una herramienta de AWS Systems Manager. Sin embargo, puede crear sus propias plantillas de cambios para definir los cambios que desea permitir en los recursos de su organización o cuenta.

Los cambios que se realizan cuando se ejecuta un flujo de trabajo de manual de procedimientos se basan en el contenido del manual de procedimientos de Automation. En cada plantilla de cambios que cree, puede incluir uno o más manuales de procedimientos de Automation de los que puede elegir el usuario que realiza una solicitud de cambio para que se ejecute durante la actualización. También puede crear plantillas de cambios que permitan a los solicitantes elegir cualquier manual de procedimientos de Automation que esté disponible para la solicitud de cambio.

Para crear una plantilla de cambios, puede utilizar la opción Builder (Generador) en la página Create template (Crear plantilla) de la consola. De forma alternativa, mediante la opción Editor (Editor), puede crear de forma manual contenido JSON o YAML con la configuración que desee para el flujo de trabajo del manual de procedimientos. También puede utilizar una herramienta de línea de comandos para crear una plantilla de cambios, con contenido JSON para la plantilla de cambios almacenada en un archivo externo.

Temas

- Prueba de la plantilla de cambios administrada por AWSHello World
- Creación de plantillas de cambios

- Revisión y aprobación o rechazo de las plantillas de cambios
- Eliminación de plantillas de cambio

Prueba de la plantilla de cambios administrada por AWS`Hello World`

Puede utilizar la plantilla de cambios de muestra `AWS-HelloWorldChangeTemplate`, que utiliza el manual de procedimientos de Automatización de muestra `AWS-HelloWorld`, para probar el proceso de revisión y aprobación una vez que haya terminado de configurar Change Manager, una herramienta de AWS Systems Manager. Esta plantilla está diseñada para probar o verificar los permisos configurados, las asignaciones de aprobadores y el proceso de aprobación. La aprobación para utilizar esta plantilla de cambios en su organización o cuenta ya ha sido otorgada por AWS. Sin embargo, cualquier solicitud de cambio basada en esta plantilla de cambios aún debe ser aprobada por los revisores de su organización o cuenta.

En lugar de efectuar cambios en un recurso, el resultado del flujo de trabajo del manual de procedimientos asociado a esta plantilla es imprimir un mensaje en la salida de un paso de Automation.

Antes de empezar

Antes de comenzar, asegúrese de haber realizado las siguientes tareas:

- Si utiliza AWS Organizations para administrar los cambios en toda una organización, lleve a cabo las tareas de configuración de la organización descritas en Configuración de Change Manager para una organización (cuenta de administración).
- Configure Change Manager para su cuenta de administrador delegado o cuenta única, tal como se describe en Configuración de opciones y prácticas recomendadas de Change Manager.

> **Note**
>
> Si activó la opción de práctica recomendada Require monitors for all templates (Requerir monitores para todas las plantillas) en la configuración de Change Manager, desactívelo temporalmente mientras prueba la plantilla de cambios Hello World.

Para probar la plantilla de cambios Hello World administrada por AWS

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.

2. En el panel de navegación, elija Change Manager.
3. Seleccione Create request (Crear solicitud).
4. Elija la plantilla de cambios denominada `AWS-HelloWorldChangeTemplate` y, luego, Next (Siguiente).
5. En Name (Nombre), ingrese un nombre para la solicitud de cambio que facilite la identificación de su finalidad, como **`MyChangeRequestTest`**.
6. Para conocer el resto de los pasos para crear la solicitud de cambio, consulte Creación de solicitudes de cambio.

Pasos a seguir a continuación

Para obtener más información acerca de cómo aprobar las solicitudes de cambio, consulte Revisión y aprobación o rechazo de las solicitudes de cambio.

Para ver el estado y los resultados de la solicitud de cambio, elija el nombre en la pestaña Requests (Solicitudes) de Change Manager.

Creación de plantillas de cambios

Una plantilla de cambios es una colección de ajustes de configuración en Change Manager que definen aspectos tales como las aprobaciones requeridas, los manuales de procedimientos disponibles y las opciones de notificación para las solicitudes de cambio.

Puede crear plantillas de cambios para sus operaciones de Change Manager, una herramienta de AWS Systems Manager, con la consola, que incluye las opciones del Creador y el Editor o las herramientas de línea de comandos.

Temas

- Acerca de las aprobaciones en las plantillas de cambios
- Creación de plantillas de cambios con el Generador
- Creación de plantillas de cambios con el Editor
- Creación de plantillas de cambios con las herramientas de línea de comandos

Acerca de las aprobaciones en las plantillas de cambios

En cada plantilla de cambios que cree, puede especificar hasta cinco niveles de aprobación para las solicitudes de cambio que se creen a partir de ella. Para cada uno de esos niveles, puede designar

hasta cinco aprobadores potenciales. Un aprobador no se limita a un solo usuario. También puede especificar un grupo de IAM o un rol de IAM como aprobador individual. En el caso de los grupos de IAM y los roles de IAM, uno o más usuarios que pertenezcan al grupo o al rol pueden dar su aprobación a fin de recibir el número total de aprobaciones necesarias para una solicitud de cambio. También puede especificar más aprobadores de los que requiere la plantilla de cambios.

Change Manager admite dos enfoques principales de las aprobaciones: las aprobaciones por nivel y las aprobaciones por línea. También es posible combinar ambos tipos en algunas situaciones. Le recomendamos que utilice solo aprobaciones por nivel en sus operaciones de Change Manager.

Per-level approvals

Recomendado. A partir del 23 de enero de 2023, Change Manager admite aprobaciones por nivel. En este modelo, para cada nivel de aprobación de la plantilla de cambios, primero se debe especificar el número de aprobaciones necesarias para ese nivel. A continuación, se especifica al menos ese número de aprobadores para el nivel y se pueden especificar más aprobadores. Sin embargo, solo el número de aprobadores por nivel que se especifique deben aprobar la solicitud de cambio. Por ejemplo, es posible especificar cinco aprobadores, pero se requieren tres aprobaciones.

Para ver ejemplos de la vista en la consola y JSON de este tipo de aprobación, consulte the section called “Ejemplo de configuración de aprobación por nivel”.

Per-line approvals

Se admite para la compatibilidad con versiones anteriores. La versión original de Change Manager solo admitía aprobaciones por línea. En este modelo, cada aprobador especificado para un nivel de aprobación se representa como una línea de aprobación. Cada aprobador tenía que aprobar una solicitud de cambio para que se aprobara en ese nivel. Antes del 23 de enero de 2023, este era el único modelo compatible para las aprobaciones. Las plantillas de cambios creadas antes de esta fecha siguen admitiendo aprobaciones por línea, pero se recomienda utilizar aprobaciones por nivel en su lugar.

Para ver ejemplos de la vista en la consola y JSON de este tipo de aprobación, consulte the section called “Ejemplo de configuración de aprobación por línea”.

Combined per-line and per-level approvals

No se recomienda. En la consola, la pestaña Editor ya no permite agregar aprobaciones por línea. Sin embargo, en algunos casos, es posible que consiga aprobaciones tanto por línea como por

nivel en una plantilla de cambios. Esto puede ocurrir si actualiza una plantilla de cambios que se creó antes del 23 de enero de 2023 o si crea o actualiza una plantilla de cambios y edita su contenido en YAML manualmente.

Para ver ejemplos de la vista en la consola y JSON de este tipo de aprobación, consulte the section called “Ejemplo de configuración de aprobación combinada por nivel y por línea”.

> **Important**
>
> Si bien es posible crear una plantilla de cambios que combine las aprobaciones por línea y por nivel, esta configuración no es recomendable ni necesaria. Prevalecerá el tipo de aprobación que requiera más aprobaciones (aprobaciones por línea o por nivel). Por ejemplo:
>
> - Si una plantilla de cambios especifica tres aprobaciones por nivel pero cinco aprobaciones por línea, se requieren cinco aprobaciones.
> - Si una plantilla de cambios especifica cuatro aprobaciones por nivel pero dos por línea, se requieren cuatro aprobaciones.

Puede crear un nivel que incluya aprobaciones por línea y por nivel editando el contenido en YAML o JSON manualmente. A continuación, la pestaña Editor muestra los controles para especificar el número de aprobaciones requerido tanto para el nivel como para las líneas individuales. Sin embargo, los nuevos niveles que agregue mediante la consola solo admiten configuraciones de aprobación por nivel.

Notificaciones y rechazos de solicitudes de cambio

Notificaciones de Amazon SNS

Cuando se crea una solicitud de cambio con su plantilla, las notificaciones se envían a los suscriptores del tema Amazon Simple Notification Service (Amazon SNS) designado para las notificaciones de ese nivel. Puede especificar el tema de la notificación en la plantilla de cambios o permitir que el usuario que crea la solicitud de cambio especifique uno.

Luego de recibir el número mínimo de aprobaciones necesarias en un nivel, se envían notificaciones a los aprobadores suscritos al tema de Amazon SNS para el siguiente nivel, y así sucesivamente.

> **Important**
>
> Asegúrese de que los roles, grupos y usuarios de IAM que designe en conjunto proporcionen suficientes aprobadores para cumplir con la cantidad requerida de aprobaciones que especifique. Por ejemplo, si designa solo un grupo de IAM como aprobador que contiene tres usuarios, no puede especificar que cinco aprobaciones sean obligatorias en ese nivel, solo tres o menos.

Rechazos de solicitudes de cambio

Independientemente del número de niveles de aprobación y aprobadores que especifique, solo se requiere un rechazo a una solicitud de cambio para evitar que se produzca el flujo de trabajo del manual de procedimientos de esa solicitud.

Ejemplos de tipos de aprobación de Change Manager

Los siguientes ejemplos muestran la vista de la consola y el contenido en JSON de los tres tipos de aprobación en Change Manager.

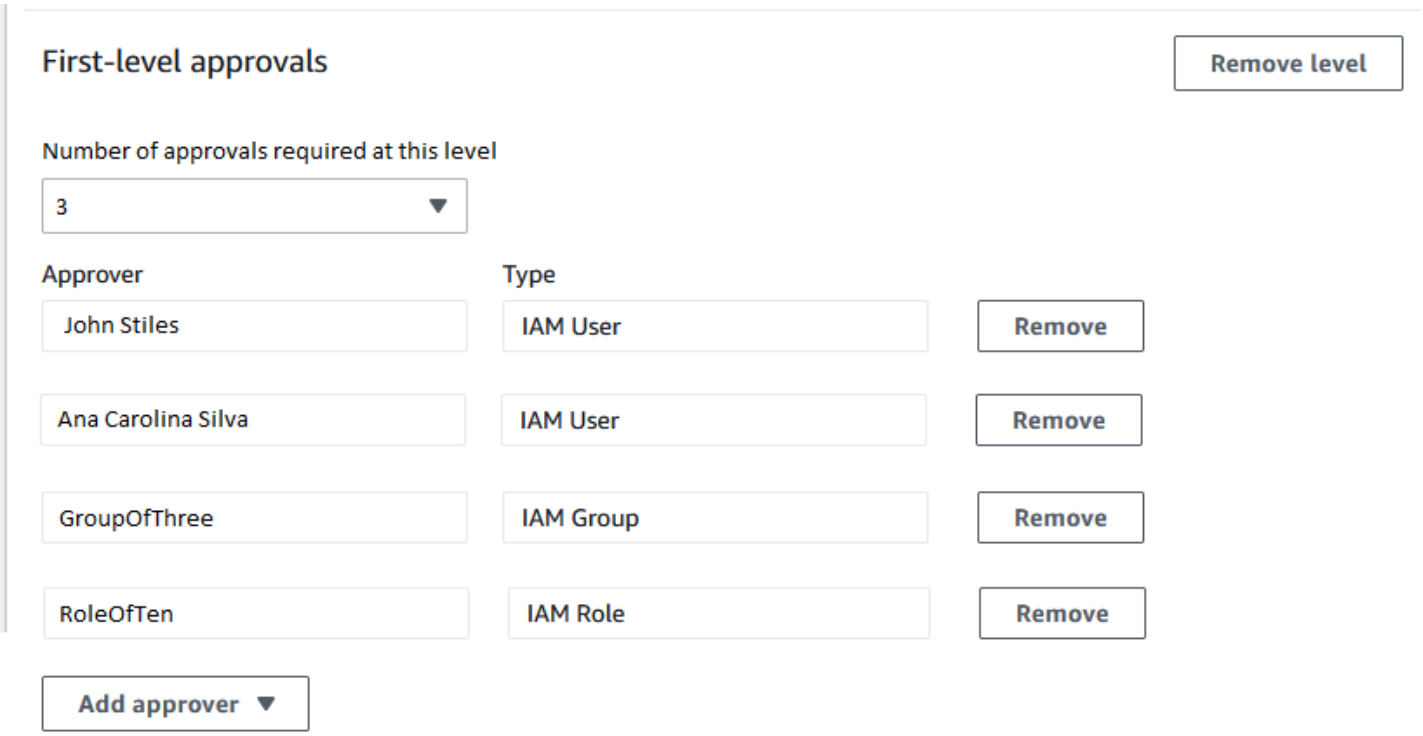
Temas

- Ejemplo de configuración de aprobación por nivel
- Ejemplo de configuración de aprobación por línea
- Ejemplo de configuración de aprobación combinada por nivel y por línea

Ejemplo de configuración de aprobación por nivel

En la configuración del nivel de aprobación por nivel que se muestra en la siguiente imagen, se requieren tres aprobaciones. Esas aprobaciones pueden provenir de cualquier combinación de usuarios, grupos y roles de IAM que se especifiquen como aprobadores. Los aprobadores especificados incluyen dos usuarios de IAM (John Stiles y Ana Carolina Silva), un grupo de usuarios que contiene tres miembros (`GroupOfThree`) y un rol de usuario que representa a diez usuarios (`RoleOfTen`).

Si los tres usuarios del grupo `GroupOfThree` aprueban la solicitud de cambio, se aprueba para ese nivel. No es necesario recibir la aprobación de cada usuario, grupo o rol. El número mínimo de aprobaciones puede provenir de cualquier combinación de aprobadores especificados. Le recomendamos que utilice aprobaciones por nivel en sus operaciones de Change Manager.



En el siguiente ejemplo, se ilustra parte del código en YAML de esta configuración.

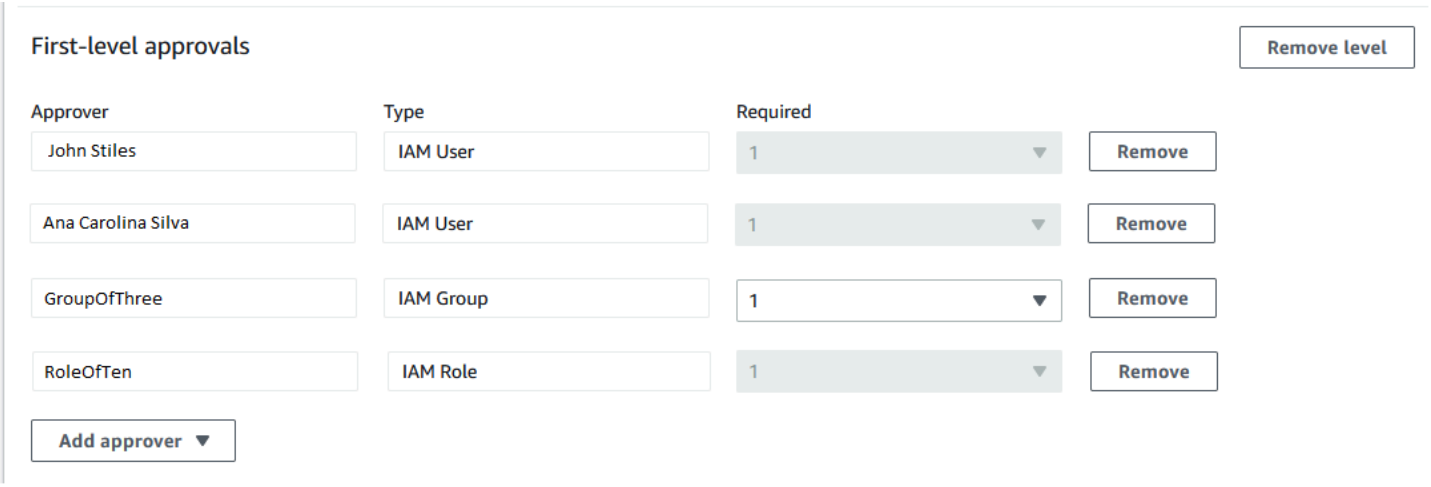
> **Note**
>
> Esta versión del código en YAML incluye una entrada adicional, `MinRequiredApprovals` (con `M` mayúscula inicial). El valor de esta entrada indica cuántas aprobaciones se requieren de entre todos los revisores disponibles. Tenga en cuenta también que el valor `minRequiredApprovals` (`m` inicial en minúscula) de cada aprobador de la lista `Approvers` es `0` (cero). Esto indica que el aprobador puede contribuir a las aprobaciones generales, pero no está obligado a hacerlo.

```
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
```

Ejemplo de configuración de aprobación por línea

En la configuración del nivel de aprobación que se muestra en la siguiente imagen, se especifican cuatro aprobadores. Estos incluyen dos usuarios de IAM (John Stiles y Ana Carolina Silva), un grupo de usuarios que contiene tres miembros (`GroupOfThree`) y un rol de usuario que representa a diez usuarios (`RoleOfTen`). Se admiten aprobaciones por línea por motivos de compatibilidad con versiones anteriores, pero no se recomiendan.



Para que la solicitud de cambio se apruebe en esta configuración de aprobación por línea, deben aprobarla todas las líneas de aprobadores: John Stiles, Ana Carolina Silva, un miembro del grupo `GroupOfThree` y un miembro del rol `RoleOfTen`.

En el siguiente ejemplo, se ilustra parte del código en YAML de esta configuración.

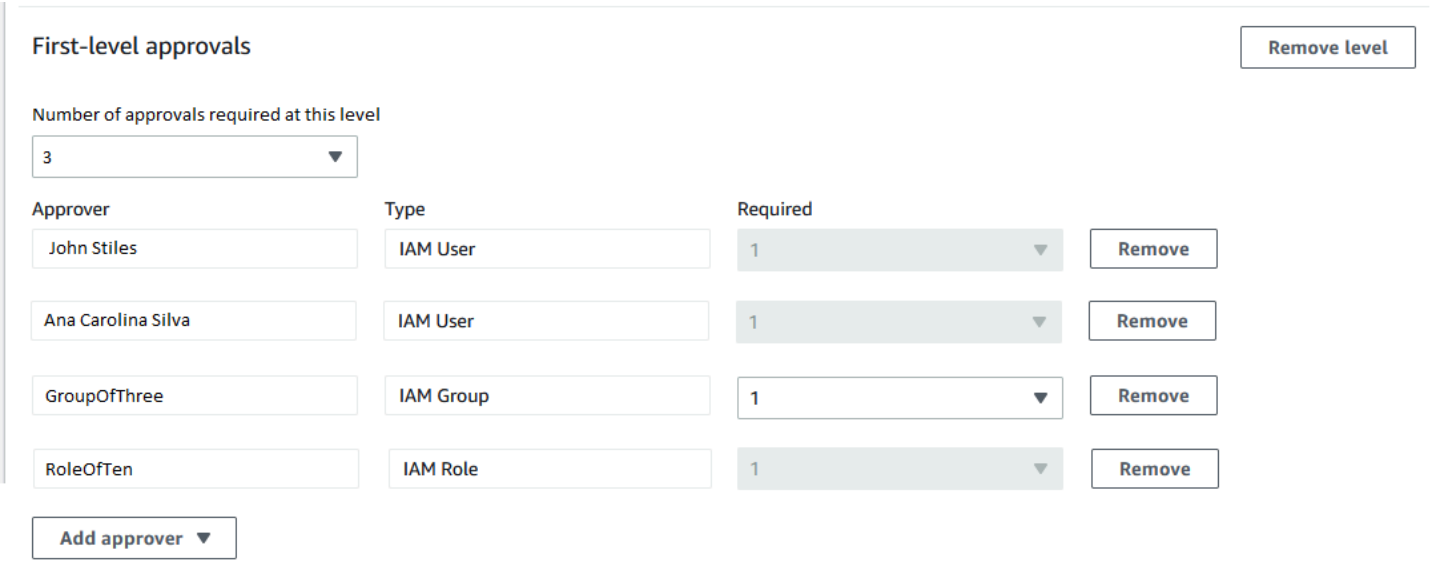
> **Note**
>
> Observe que el valor de cada aprobador `minRequiredApprovals` es 1. Esto indica que se requiere una aprobación de cada aprobador.

```
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
```

Ejemplo de configuración de aprobación combinada por nivel y por línea

En la configuración combinada de aprobación por nivel y por línea que se muestra en la siguiente imagen, se especifican tres aprobaciones para el nivel, pero se especifican cuatro aprobaciones por

línea. El tipo de aprobación que requiera más aprobaciones tendrá prioridad sobre el otro, por lo que esta configuración requiere cuatro aprobaciones. No se recomienda la aprobación combinada por nivel y por línea.




```
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
```

Temas

- Creación de plantillas de cambios con el Generador
- Creación de plantillas de cambios con el Editor
- Creación de plantillas de cambios con las herramientas de línea de comandos

Creación de plantillas de cambios con el Generador

Con el Creador de plantillas de cambios de Change Manager, una herramienta de AWS Systems Manager, puede configurar el flujo de trabajo de manual de procedimientos definido en la plantilla de cambios sin tener que utilizar la sintaxis JSON o YAML. Después de que usted especifique las opciones, el sistema convertirá la entrada al formato YAML que Systems Manager puede utilizar para ejecutar los flujos de trabajo del manual de procedimientos.

Para crear una plantilla de cambios con el Generador

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Change Manager.
3. Seleccione Crear plantilla.
4. En Name (Nombre), ingrese un nombre para la plantilla que facilite la identificación de su finalidad, como **`UpdateEC2LinuxAMI`**.
5. En la sección Change template details (Detalles de la plantilla de cambios), realice el siguiente procedimiento:

   - En Description (Descripción), proporcione una breve explicación de cómo y cuándo se utilizará la plantilla de cambios que está creando.

     Esta descripción ayuda a los usuarios que crean solicitudes de cambio a determinar si están utilizando la plantilla de cambios correcta. También ayuda a quienes revisan las solicitudes de cambio a entender si la solicitud debe aprobarse.
   - En Change template type (Tipo de plantilla de cambios), especifique si va a crear una plantilla de cambios estándar o una plantilla de cambios de emergencia.

La plantilla de cambios de emergencia se utiliza para situaciones en las que se debe realizar un cambio aunque los cambios estén bloqueados por un evento del calendario en uso por AWS Systems Manager Change Calendar. Las solicitudes de cambio creadas a partir de una plantilla de cambios de emergencia siguen requiriendo la aprobación de los aprobadores designados, pero los cambios solicitados pueden ejecutarse incluso cuando el calendario está bloqueado.

- En Runbook options (Opciones de manuales de procedimientos), especifique los manuales de procedimientos de los que los usuarios pueden elegir cuando se crea una solicitud de cambio. Puede agregar un único manual de procedimientos o varios. De forma alternativa, puede permitir a los solicitantes especificar el manual de procedimientos que se va a utilizar. En cualquiera de estos casos, solo se puede incluir un manual de procedimientos en la solicitud de cambio.
- En Runbook (Manual de procedimientos), seleccione los nombres de los manuales de procedimientos y las versiones de esos manuales de los cuales los usuarios pueden elegir para sus solicitudes de cambio. Independientemente de la cantidad de manuales de procedimientos que agregue a la plantilla de cambios, solo se puede seleccionar uno por cada solicitud de cambio.

  No se especifica ningún manual de procedimientos si antes se elige Any runbook can be used (Se puede utilizar cualquier manual de procedimientos).

  > **Tip**
  >
  > Seleccione un manual de procedimientos y una versión de manual de procedimientos y, a continuación, elija View (Ver) para examinar el contenido del manual de procedimientos en la interfaz de Systems Manager Documents.

6. En la sección Template information (Información de la plantilla), utilice Markdown para ingresar información para los usuarios que creen solicitudes de cambio a partir de esta plantilla de cambios. Hemos proporcionado un conjunto de preguntas que puede incluir para los usuarios que crean solicitudes de cambio, o bien, puede agregar otra información y preguntas en su lugar.

> **Note**
>
> Markdown es un lenguaje de marcado que le permite agregar descripciones de estilo wiki a documentos y pasos individuales dentro del documento. Para obtener más información acerca del uso de Markdown, consulte Uso de Markdown en AWS.

Recomendamos proporcionar preguntas para los usuarios sobre sus solicitudes de cambio para ayudar a los aprobadores a decidir si aprobar o no cada solicitud de cambio, como listar los pasos manuales necesarios para ejecutarse como parte del cambio y un plan de restauración.

> **Tip**
>
> Alterne entre Hide preview (Ocultar vista previa) y Show preview (Mostrar vista previa) para ver el aspecto del contenido a medida que lo redacta.

7. En la sección Change request approvals (Aprobaciones de solicitudes de cambio), realice el siguiente procedimiento:

   - (Opcional) Si desea permitir que las solicitudes de cambio creadas a partir de esta plantilla de cambios se ejecuten automáticamente, sin que los aprobadores las revisen (con la excepción de los eventos de congelación de cambios), seleccione Enable auto-approval (Habilitar la aprobación automática).

     > **Note**
     >
     > Habilitar las aprobaciones automáticas en una plantilla de cambios proporciona a los usuarios la opción de pasar por alto a los revisores. Sin embargo, los usuarios todavía pueden elegir especificar revisores cuando se crea una solicitud de cambio. Por lo tanto, aún debe especificar las opciones de revisores en la plantilla de cambios.

     > **Important**
     >
     > Si habilita la aprobación automática para una plantilla de cambios, los usuarios pueden enviar solicitudes de cambio utilizando esa plantilla que no requiere revisión por parte de los revisores antes de ejecutarse (con la excepción de los aprobadores

de eventos de congelación de cambios). Si desea restringir que un usuario, grupo o rol de IAM determinado envíe solicitudes de aprobación automática, puede utilizar una condición en una política de IAM para este fin. Para obtener más información, consulte Control de acceso a flujos de trabajo de manual de procedimientos de aprobación automática.

- En Número de aprobaciones requeridas en este nivel, elija el número de aprobaciones que deben recibir las solicitudes de cambio creadas a partir de esta plantilla de cambios para este nivel.
- Para agregar aprobadores obligatorios de primer nivel, elija Add approver (Agregar aprobador) y, luego, elija entre las siguientes opciones:
  - Template specified approvers (Aprobadores especificados en la plantilla): elija uno o más usuarios, grupos o roles de AWS Identity and Access Management (IAM) de su cuenta para que aprueben las solicitudes de cambio creadas a partir de esta plantilla de cambios. Todas las solicitudes de cambio que se creen con esta plantilla deben ser revisadas y aprobadas por cada aprobador que usted especifique.
  - Solicite aprobadores específicos: el usuario que realiza la solicitud de cambio especifica los revisores cuando efectúa la solicitud y puede elegir entre una lista de usuarios de su cuenta.

    El número que ingrese en la columna Required (Obligatorio) determinará cuántos revisores deben especificarse en una solicitud de cambio que utilice esta plantilla de cambios.

> **Important**
>
> Antes del 23 de enero de 2023, la pestaña Editor permitía especificar solo las aprobaciones por línea. Las nuevas plantillas de cambios y los nuevos niveles que agregue a las plantillas de cambios existentes mediante la pestaña Editor solo admiten aprobaciones por nivel. Le recomendamos que utilice solo aprobaciones por nivel en sus operaciones de Change Manager.
> Para obtener más información, consulte Acerca de las aprobaciones en las plantillas de cambios.

- En SNS topic to notify approvers (Tema de SNS para notificar a los aprobadores), realice el siguiente procedimiento:
  1. Elija una de las siguientes opciones para especificar el tema de Amazon Simple Notification Service (Amazon SNS) de su cuenta que se utilizará para enviar notificaciones a los aprobadores que indiquen que una solicitud de cambio está lista para que la revisen:

- Ingrese un nombre de recurso de Amazon (ARN) de SNS: en Topic ARN (ARN de tema), ingrese el ARN de un tema de Amazon SNS existente. Este tema puede estar en cualquiera de las cuentas de su organización.
- Seleccione un tema de SNS existente: en Target notification topic (Tema de notificaciones de destino), seleccione el ARN de un tema de Amazon SNS existente en su Cuenta de AWS actual. (Esta opción no estará disponible si aún no ha creado ningún tema de Amazon SNS en su Cuenta de AWS y Región de AWS actuales).
- Especifique el tema de SNS cuando se crea la solicitud de cambio: el usuario que crea una solicitud de cambio puede especificar el tema de Amazon SNS que se utilizará para efectuar las notificaciones.

> **Note**
>
> El tema de Amazon SNS que seleccione debe estar configurado para especificar las notificaciones que envía y los suscriptores a los que se las envía. Su política de acceso también debe conceder permisos a Systems Manager para que Change Manager pueda enviar notificaciones. Para obtener más información, consulta Configuración de temas de Amazon SNS para las notificaciones de Change Manager.

2. Seleccione Agregar notificación.

8. (Opcional) Para agregar un nivel adicional de aprobadores, elija Add approval level (Agregar nivel de aprobación) y elija entre aprobadores especificados en la plantilla y aprobadores especificados en la solicitud para este nivel. A continuación, elija un tema de SNS para notificar a este nivel de aprobadores.

   Una vez recibidas todas las aprobaciones emitidas por los aprobadores del primer nivel, se notifica a los aprobadores del segundo nivel y así sucesivamente.

   Puede agregar un máximo de cinco niveles de aprobadores en cada plantilla. Por ejemplo, puede requerir aprobaciones de usuarios con roles técnicos para el primer nivel y, luego, la aprobación administrativa para el segundo nivel.

9. En la sección Monitoring (Monitoreo), en CloudWatch alarm to monitor (Alarma de CloudWatch para monitorear), ingrese el nombre de una alarma de Amazon CloudWatch de la cuenta actual para monitorear el progreso de los flujos de trabajo del manual de procedimientos que se basen en esta plantilla.

> **Tip**
>
> Para crear una nueva alarma o revisar la configuración de una alarma que desea especificar, elija Open the Amazon CloudWatch console (Abrir la consola de Amazon CloudWatch). Para obtener más información acerca del uso de las alarmas de CloudWatch, consulte Using CloudWatch Alarms en la Guía del usuario de Amazon CloudWatch.

10. En la sección notificaciones, realice el siguiente procedimiento:

    1. Elija una de las siguientes opciones para especificar el tema de Amazon SNS de su cuenta que se utilizará para enviar notificaciones sobre las solicitudes de cambio creadas con esta plantilla de cambios:

       - Ingrese un nombre de recurso de Amazon (ARN) de SNS: en Topic ARN (ARN de tema), ingrese el ARN de un tema de Amazon SNS existente. Este tema puede estar en cualquiera de las cuentas de su organización.
       - Seleccione un tema de SNS existente: en Target notification topic (Tema de notificaciones de destino), seleccione el ARN de un tema de Amazon SNS existente en su Cuenta de AWS actual. (Esta opción no estará disponible si aún no ha creado ningún tema de Amazon SNS en su Cuenta de AWS y Región de AWS actuales).

       > **Note**
       >
       > El tema de Amazon SNS que seleccione debe estar configurado para especificar las notificaciones que envía y los suscriptores a los que se las envía. Su política de acceso también debe conceder permisos a Systems Manager para que Change Manager pueda enviar notificaciones. Para obtener más información, consulta Configuración de temas de Amazon SNS para las notificaciones de Change Manager.

    2. Seleccione Agregar notificación.

11. (Opcional) En la sección Tags (Etiquetas), aplique uno o más pares de nombre y valor de clave de etiqueta en la plantilla de cambios.

    Las etiquetas son metadatos opcionales que usted asigna a un recurso. Con las etiquetas puede clasificar un recurso de diferentes maneras, por ejemplo, según su finalidad, propietario o entorno. Por ejemplo, es posible que desee etiquetar una plantilla de cambios para identificar

el tipo de cambios que ella realiza y el entorno en el que se ejecuta. En este caso, puede especificar los siguientes pares de claves nombre-valor:

- `Key=TaskType,Value=InstanceRepair`
- `Key=Environment,Value=Production`

12. Seleccione Save and preview (Guardar y previsualizar).
13. Revise los detalles de la plantilla de cambios que está creando.

    Si desea modificar la plantilla de cambios antes de enviarla para su revisión, elija Actions, Edit (Acciones, Editar).

    Si está satisfecho con el contenido de la plantilla de cambios, elija Submit for review (Enviar para revisión). Los usuarios de su organización o cuenta que se hayan especificado como revisores de plantillas en la pestaña Settings (Configuración) de Change Manager reciben una notificación que indica que una nueva plantilla de cambios está en espera de su revisión.

    Si se ha especificado un tema de Amazon SNS para las plantillas de cambios, se envían notificaciones cuando la plantilla de cambios se rechaza o se aprueba. Si no recibe notificaciones relacionadas con esta plantilla de cambios, puede regresar a Change Manager más tarde para verificar su estado.

Creación de plantillas de cambios con el Editor

Siga los pasos de este tema para configurar una plantilla de cambios en Change Manager, una herramienta de AWS Systems Manager. Para ello, ingrese JSON o YAML en lugar de usar los controles de la consola.

Para crear una plantilla de cambios con el Editor

1. En el panel de navegación, elija Change Manager.
2. Seleccione Crear plantilla.
3. En Name (Nombre), ingrese un nombre para la plantilla que facilite la identificación de su finalidad, como **`RestartEC2LinuxInstance`**.
4. Arriba de Change template details (Detalles de la plantilla de cambios), elija Editor (Editor).
5. En la sección Document editor (Editor de documentos), elija Edit (Editar) y, a continuación, ingrese el contenido JSON o YAML para su plantilla de cambios.

   A continuación se muestra un ejemplo.

> **Note**
>
> El parámetro `minRequiredApprovals` se utiliza para especificar cuántos revisores de un nivel especificado deben aprobar una solicitud de cambio creada mediante esta plantilla.
> En este ejemplo, se muestran dos niveles de aprobaciones. Puede especificar hasta cinco niveles de aprobaciones, pero solo se requiere un nivel.
> En el primer nivel, el usuario específico “John-Doe” debe aprobar cada solicitud de cambio. Después de eso, tres miembros cualesquiera del rol de IAM `Admin` deben aprobar la solicitud de cambio.
> Para obtener más información acerca de la aprobación de las plantillas de cambios, consulte Acerca de las aprobaciones en las plantillas de cambios.

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. Seleccione Save and preview (Guardar y previsualizar).
7. Revise los detalles de la plantilla de cambios que está creando.

   Si desea modificar la plantilla de cambios antes de enviarla para su revisión, elija Actions, Edit (Acciones, Editar).

   Si está satisfecho con el contenido de la plantilla de cambios, elija Submit for review (Enviar para revisión). Los usuarios de su organización o cuenta que se hayan especificado como revisores de plantillas en la pestaña Settings (Configuración) de Change Manager reciben una notificación que indica que una nueva plantilla de cambios está en espera de su revisión.

   Si se ha especificado un tema de Amazon Simple Notification Service (Amazon SNS) para las plantillas de cambios, las notificaciones se envían cuando se rechaza o se aprueba la plantilla. Si no recibe notificaciones relacionadas con esta plantilla de cambios, puede regresar a Change Manager más tarde para verificar su estado.

Creación de plantillas de cambios con las herramientas de línea de comandos

En los siguientes procedimientos, se describe cómo utilizar la AWS Command Line Interface (AWS CLI) (en Linux, macOS o Windows) o las AWS Tools for Windows PowerShell para crear una solicitud de cambio en Change Manager, una herramienta de AWS Systems Manager.

Para crear una nueva plantilla de cambios

1. Si aún no lo ha hecho, instale y configure la AWS CLI o las Herramientas de AWS para PowerShell.

   Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.
2. Cree un archivo JSON en su equipo local con un nombre similar a `MyChangeTemplate.json` y, a continuación, péguele el contenido para su plantilla de cambios.

   > **Note**
   >
   > Las plantillas de cambios utilizan una versión del esquema 0.3 que no incluye la misma compatibilidad que para los manuales de procedimientos de Automation.

A continuación se muestra un ejemplo.

> **Note**
>
> El parámetro `minRequiredApprovals` se utiliza para especificar cuántos revisores de un nivel especificado deben aprobar una solicitud de cambio creada mediante esta plantilla.
> En este ejemplo, se muestran dos niveles de aprobaciones. Puede especificar hasta cinco niveles de aprobaciones, pero solo se requiere un nivel.
> En el primer nivel, el usuario específico “John-Doe” debe aprobar cada solicitud de cambio. Después de eso, tres miembros cualesquiera del rol de IAM `Admin` deben aprobar la solicitud de cambio.
> Para obtener más información acerca de la aprobación de las plantillas de cambios, consulte Acerca de las aprobaciones en las plantillas de cambios.

```
{
   "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. Ejecute el siguiente comando para crear la plantilla de cambios.

   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"
```

Para obtener más información sobre las opciones que puede especificar, consulte create-document.

El sistema devuelve información similar a la siguiente.

```
{
   "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":[

      ]
   }
}
```

Los usuarios de su organización o cuenta que se hayan especificado como revisores de plantillas en la pestaña Settings (Configuración) de Change Manager reciben una notificación que indica que una nueva plantilla de cambios está en espera de su revisión.

Si se ha especificado un tema de Amazon Simple Notification Service (Amazon SNS) para las plantillas de cambios, las notificaciones se envían cuando se rechaza o se aprueba la plantilla. Si no recibe notificaciones relacionadas con esta plantilla de cambios, puede regresar a Change Manager más tarde para verificar su estado.

Revisión y aprobación o rechazo de las plantillas de cambios

Si está especificado como revisor de plantillas de cambios en Change Manager, una herramienta de AWS Systems Manager, se le avisará cuando una nueva plantilla de cambios o una nueva versión de una plantilla de cambios esté a la espera de su revisión. Un tema de Amazon Simple Notification Service (Amazon SNS) envía las notificaciones.

> **Note**
>
> Esta funcionalidad depende de si su cuenta se ha configurado para utilizar un tema de Amazon SNS para enviar notificaciones de la revisión de las plantillas de cambios. Para obtener más información acerca de cómo especificar un tema para las notificaciones a los revisores de plantillas, consulte Tarea 1: configurar la administración de identidades de usuarios y los revisores de plantillas de Change Manager.

Para revisar la plantilla de cambios, visite el enlace que aparece en la notificación, inicie sesión en la AWS Management Console y siga los pasos descritos en este procedimiento.

Para revisar y aprobar o rechazar una plantilla de cambios

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Change Manager.
3. En la sección Change templates (Plantillas de cambios) de la parte inferior de la pestaña Overview (Información general), elija el número en Pending review (Pendiente de revisión).
4. En la lista Change templates (Plantillas de cambios), busque y elija el nombre de la plantilla de cambios que va a revisar.
5. En la página de resumen, revise el contenido propuesto de la plantilla de cambios y realice alguna de las siguientes acciones:
   - Para aprobar la plantilla de cambios, lo que permite utilizarla en las solicitudes de cambio, elija Approve (Aprobar).
   - Para rechazar la plantilla de cambios, lo que evita que se utilice en solicitudes de cambio, elija Reject (Rechazar).

Eliminación de plantillas de cambio

En este tema se describe cómo se eliminan las plantillas que ha creado en Change Manager, una herramienta de Systems Manager. Si utiliza Change Manager para una organización, este procedimiento se lleva a cabo en su cuenta de administrador autorizado.

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Change Manager.
3. Elija la pestaña Plantillas.
4. Elija el nombre de la plantilla que desea eliminar.
5. Elija Actions, Delete template. (Actiones, Eliminar plantilla).
6. En el cuadro de diálogo de confirmación, ingrese **`DELETE`** y, a continuación, elija Delete (Eliminar).

Uso de solicitudes de cambio

Una solicitud de cambio se trata de una solicitud en Change Manager para ejecutar un manual de procedimientos de Automation que actualice uno o más recursos de sus entornos locales o en AWS. Una solicitud de cambio se crea usando una plantilla de cambios.

Cuando crea una solicitud de cambio en Change Manager, una herramienta de AWS Systems Manager, uno o más aprobadores de su organización o cuenta deben revisar y aprobar la solicitud. Sin las aprobaciones necesarias, el flujo de trabajo del manual de procedimientos, que realiza los cambios solicitados, no tiene permitido ejecutarse.

Temas

- Creación de solicitudes de cambio
- Revisión y aprobación o rechazo de las solicitudes de cambio

Creación de solicitudes de cambio

Cuando crea una solicitud de cambio en Change Manager, una herramienta de AWS Systems Manager, la plantilla de cambios que selecciona suele llevar a cabo los siguientes procedimientos:

- Designa los aprobadores de la solicitud de cambio o especifica cuántas aprobaciones se requieren.
- Especifica el tema de Amazon Simple Notification Service (Amazon SNS) que se va a utilizar para notificar a los aprobadores acerca de la solicitud de cambio.
- Especifica una alarma de Amazon CloudWatch para monitorear el flujo de trabajo del manual de procedimientos para la solicitud de cambio.
- Identifica qué manuales de procedimientos de Automation puede elegir para realizar el cambio solicitado.

En algunos casos, es posible que una plantilla de cambios se configure para que usted especifique su propio manual de procedimientos de Automation que va a utilizar y, también, quién debe revisar y aprobar la solicitud.

> **Important**
>
> Si utiliza Change Manager en toda una organización, se recomienda efectuar siempre los cambios desde la cuenta de administrador delegado. Si bien es posible realizar cambios

desde otras cuentas de la organización, esos cambios no se notificarán ni se podrán ver desde la cuenta de administrador delegado.

Temas

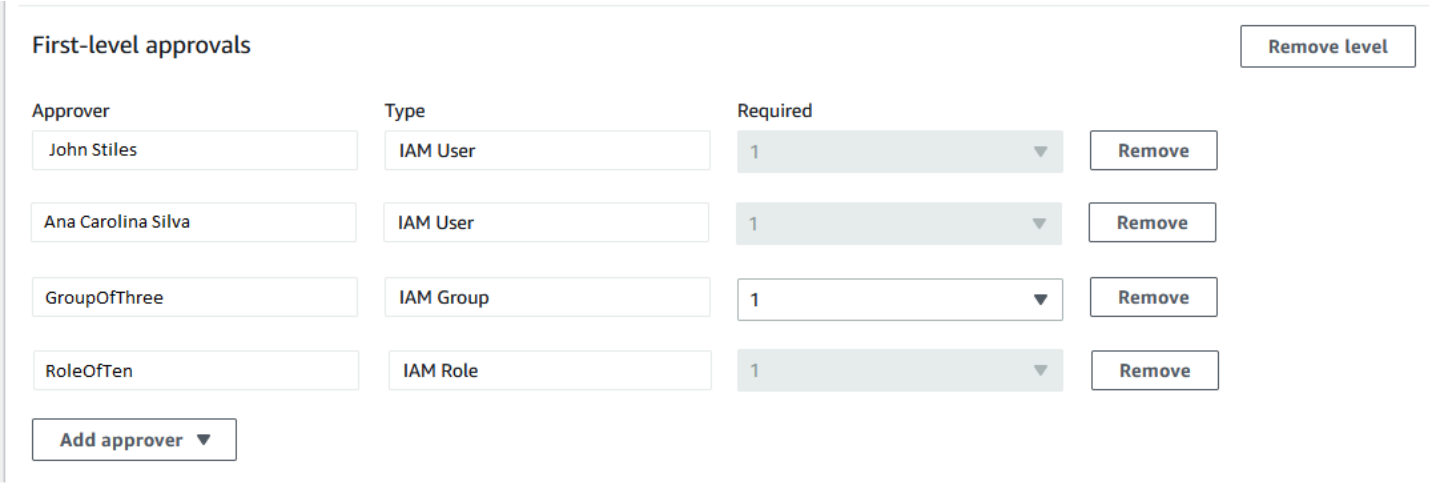
- Acerca de las aprobaciones de solicitudes de cambio
- Creación de solicitudes de cambio (consola)
- Creación de solicitudes de cambio (AWS CLI)

Acerca de las aprobaciones de solicitudes de cambio

Según los requisitos especificados en una plantilla de cambios, las solicitudes de cambio que cree a partir de ella pueden requerir la aprobación de hasta cinco niveles antes de que pueda realizarse el flujo de trabajo del manual de procedimientos de la solicitud. Para cada uno de esos niveles, el creador de la plantilla puede especificar hasta cinco posibles aprobadores. Un aprobador no se limita a un solo usuario. En este sentido, un aprobador también puede ser un grupo de IAM o un rol de IAM. En el caso de los grupos de IAM y los roles de IAM, uno o más usuarios que pertenezcan al grupo o al rol pueden dar su aprobación a fin de recibir el número total de aprobaciones necesarias para una solicitud de cambio. Los creadores de plantillas también pueden especificar más aprobadores de los que requiere la plantilla de cambios.

Flujos de trabajo de aprobación originales y aprobaciones actualizadas

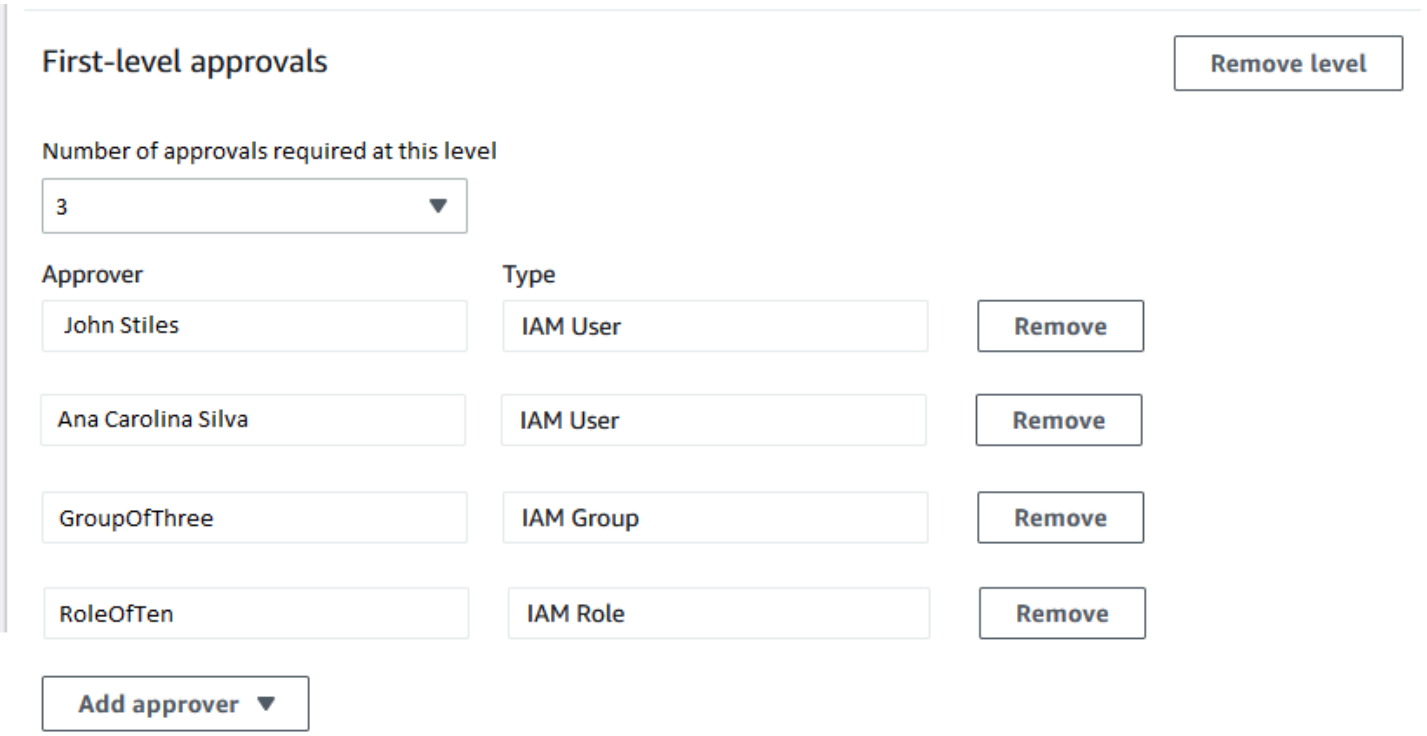
Con las plantillas de cambios creadas antes del 23 de enero de 2023, se debe recibir la aprobación de cada aprobador especificado para que la solicitud de cambio se apruebe en ese nivel. Por ejemplo, en la configuración del nivel de aprobación que se muestra en la siguiente imagen, se especifican cuatro aprobadores. Los aprobadores especificados incluyen dos usuarios (John Stiles y Ana Carolina Silva), un grupo de usuarios que contiene tres miembros (GroupOfThree) y un rol de usuario que representa a diez usuarios (RoleOfTen).



Para que la solicitud de cambio se apruebe en este nivel, deben aprobarla John Stiles, Ana Carolina Silva, un miembro del grupo `GroupOfThree` y un miembro del rol `RoleOfTen`.

Con las plantillas de cambios creadas el 23 de enero de 2023 o después, los creadores de plantillas pueden especificar el número total de aprobaciones necesarias para cada nivel de aprobación. Esas aprobaciones pueden provenir de cualquier combinación de usuarios, grupos y roles que se hayan especificado como aprobadores. Una plantilla de cambios puede requerir solo una aprobación para un nivel, pero especificar, por ejemplo, dos usuarios individuales, dos grupos y un rol como posibles aprobadores.

Por ejemplo, en el área de niveles de aprobación que se muestra en la siguiente imagen, se requieren tres aprobaciones. Los aprobadores especificados por la plantilla incluyen dos usuarios (John Stiles y Ana Carolina Silva), un grupo de usuarios que contiene tres miembros (`GroupOfThree`) y un rol de usuario que representa a diez usuarios (`RoleOfTen`).



Si los tres usuarios del grupo `GroupOfThree` aprueban la solicitud de cambio, se aprueba para ese nivel. No es necesario recibir la aprobación de cada usuario, grupo o rol. El número mínimo de aprobaciones puede provenir de cualquier combinación de posibles aprobadores.

Cuando se crea la solicitud de cambios, se envían notificaciones a los suscriptores del tema de Amazon SNS que se ha especificado para las notificaciones de aprobación en ese nivel. Es posible que el creador de la plantilla de cambios haya especificado el tema de notificación que se debe utilizar o le haya permitido especificar uno.

Una vez recibido el número mínimo de aprobaciones requeridas en un nivel, se envían notificaciones a los aprobadores que estén suscritos al tema de Amazon SNS para el siguiente nivel, y así sucesivamente.

Independientemente del número de niveles de aprobación y aprobadores que se especifiquen, solo se requiere un rechazo a una solicitud de cambio para evitar que se produzca el flujo de trabajo del manual de procedimientos de esa solicitud.

Creación de solicitudes de cambio (consola)

El siguiente procedimiento describe cómo crear una solicitud de cambio con la consola de Systems Manager.

Creación de una solicitud de cambio (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Change Manager.
3. Seleccione Create request (Crear solicitud).
4. Busque y seleccione una plantilla de cambios que desee utilizar para esta solicitud de cambio.
5. Elija Siguiente.
6. En Name (Nombre), ingrese un nombre para la solicitud de cambio que facilite la identificación de su finalidad, como **`UpdateEC2LinuxAMI-us-east-2`**.
7. En Runbook (Manual de procedimientos), seleccione el manual de procedimientos que desea utilizar para realizar el cambio solicitado.

   > **Note**
   >
   > Si la opción para seleccionar un manual de procedimientos no está disponible, se debe a que el autor de la plantilla de cambios ha especificado qué manual de procedimientos se debe utilizar.

8. En Change request information (Información de la solicitud de cambio), utilice Markdown para proporcionar información adicional sobre la solicitud de cambio para ayudar a los revisores a decidir si aprobar o rechazar la solicitud de cambio. Es posible que el autor de la plantilla que utilice haya proporcionado instrucciones o preguntas para que responda.

   > **Note**
   >
   > Markdown es un lenguaje de marcado que le permite agregar descripciones de estilo wiki a documentos y pasos individuales dentro del documento. Para obtener más información acerca del uso de Markdown, consulte Uso de Markdown en AWS.

9. En la sección Workflow start time (Hora de inicio del flujo de trabajo), elija una de las siguientes opciones:

   - Ejecutar la operación a una hora programada: en Requested start time (Hora de inicio solicitada), ingrese la fecha y la hora que propone para que se ejecute el flujo de trabajo del manual de procedimientos de esta solicitud. En Estimated end time (Hora de finalización estimada), ingrese la fecha y la hora en las que espera que se complete el flujo de trabajo

del manual de procedimientos. (Esta vez es solo una estimación que usted proporciona a los revisores).

> **Tip**
>
> Seleccione View Change Calendar (Ver calendario de cambios) para verificar si hay eventos de bloqueo durante el tiempo especificado.

- Ejecutar la operación tan pronto como sea posible después de la aprobación: si se aprueba la solicitud de cambio, el flujo de trabajo del manual de procedimientos se ejecuta tan pronto como haya un periodo sin restricciones en el que se puedan realizar los cambios.

10. En la sección Change request approvals (Aprobaciones de solicitudes de cambio), realice el siguiente procedimiento:

    1. Si se presentan las opciones de Approval type (Tipo de aprobación), elija alguna de las siguientes opciones:

       - Apruebe de forma automática: la plantilla de cambios seleccionada está configurada para permitir que las solicitudes de cambio se ejecuten automáticamente sin que los aprobadores las revisen. Continúe con el paso 11.

         > **Note**
         >
         > Los permisos especificados en las políticas de IAM que rigen el uso de Systems Manager no deben impedir que envíe solicitudes de cambio de aprobación automática para que se ejecuten automáticamente.

       - Especifique aprobadores: debe agregar uno o más usuarios, grupos o roles de IAM para que revisen y aprueben esta solicitud de cambio.

         > **Note**
         >
         > Puede elegir especificar revisores incluso si los permisos especificados en las políticas de IAM que rigen el uso de Systems Manager le permiten ejecutar solicitudes de cambio de aprobación automática.

    2. Seleccione Add approver (Agregar aprobador) y, a continuación, elija uno o más usuarios, grupos o roles de AWS Identity and Access Management (IAM) de las listas de revisores disponibles.

> **Note**
>
> Es posible que ya se hayan especificado uno o más aprobadores. Esto significa que los aprobadores obligatorios ya están especificados en la plantilla de cambios que seleccionó. Estos aprobadores no se pueden eliminar de la solicitud. Si el botón Agregar aprobador no está disponible, significa que la plantilla que ha elegido no permite agregar revisores adicionales a las solicitudes.

Para obtener más información acerca de cómo aprobar las solicitudes de cambio, consulte Acerca de las aprobaciones de solicitudes de cambio.

3. En SNS topic to notify approvers (Tema de SNS para notificar a los aprobadores), elija una de las siguientes opciones para especificar el tema de Amazon SNS de su cuenta que se utilizará para enviar notificaciones a los aprobadores que agregue a esta solicitud de cambio.

> **Note**
>
> Si la opción para especificar un tema de Amazon SNS no está disponible es porque la plantilla de cambios que seleccionó ya especifica el tema de Amazon SNS que se va a utilizar.

- Ingrese un nombre de recurso de Amazon (ARN) de SNS: en Topic ARN (ARN de tema), ingrese el ARN de un tema de Amazon SNS existente. Este tema puede estar en cualquiera de las cuentas de su organización.
- Seleccione un tema de SNS existente: en Target notification topic (Tema de notificaciones de destino), seleccione el ARN de un tema de Amazon SNS existente en su cuenta actual. (Esta opción no estará disponible si aún no ha creado ningún tema de Amazon SNS en su Cuenta de AWS y Región de AWS actuales).

> **Note**
>
> El tema de Amazon SNS que seleccione debe estar configurado para especificar las notificaciones que envía y los suscriptores a los que se las envía. Su política de acceso también debe conceder permisos a Systems Manager para que Change

Manager pueda enviar notificaciones. Para obtener más información, consulta Configuración de temas de Amazon SNS para las notificaciones de Change Manager.

4. Seleccione Agregar notificación.

11. Elija Siguiente.
12. En IAM role (Rol de IAM), seleccione un rol de IAM de su cuenta actual que tenga los permisos necesarios para ejecutar los manuales de procedimientos especificados para esta solicitud de cambio.

    Este rol también se conoce como rol de servicio, o rol de asunción, para Automation. Para obtener más información acerca de este rol, consulte Configuración de Automation.
13. En la sección Deployment location (Ubicación de implementación), elija una de las siguientes opciones:

> **Note**
>
> Si utiliza Change Manager con una única Cuenta de AWS y no con una organización configurada en AWS Organizations, no es necesario que especifique una ubicación de implementación.

- Aplique el cambio a esta cuenta: el flujo de trabajo del manual de procedimientos se ejecuta solo en la cuenta actual. Para una organización, esto significa que se ejecuta en la cuenta de administrador delegado.
- Aplique el cambio a varias unidades organizativas: en este caso, realice el siguiente procedimiento:
  1. En Accounts and organizational units (OUs) (Cuentas y unidades organizativas), ingrese el ID de una cuenta miembro de su organización en el formato **`123456789012`** o el ID de una unidad organizativa en el formato **`o-o96EXAMPLE`**.
  2. (Opcional) Para Execution role name (Nombre del rol de ejecución), ingrese el nombre del rol de IAM de la cuenta de destino o de la OU que tenga los permisos necesarios para ejecutar los manuales de procedimientos especificados para esta solicitud de cambio. Todas las cuentas de cualquier unidad organizativa que especifique deben usar el mismo nombre para este rol.
  3. (Opcional) Elija Agregar otra ubicación de destino para cada cuenta o unidad organizativa adicional que desee especificar y repita los pasos a y b.

4. En región de destinoRegión de AWS, seleccione la región en la que desea realizar el cambio; por ejemplo, `Ohio (us-east-2)` para la región Este de EE. UU. (Ohio).
5. Amplíe Rate control (Control de velocidad).

   Para Concurrency (Simultaneidad), ingrese un número y, a continuación, en la lista, seleccione si representa la cantidad o el porcentaje de cuentas en las que puede ejecutarse el flujo de trabajo del manual de procedimientos al mismo tiempo.

   En Error threshold (Límite de errores), ingrese un número y, a continuación, en la lista, seleccione si representa la cantidad o el porcentaje de cuentas en las que el flujo de trabajo del manual de procedimientos puede producir error antes de que se detenga la operación.

14. En la sección Deployment targets (Destinos de implementación), realice el siguiente procedimiento:

    1. Seleccione una de las siguientes opciones:
       - Single resource (Recurso único): el cambio debe hacerse solo para un recurso. Por ejemplo, un único nodo o una única Amazon Machine Image (AMI), según la operación definida en los manuales de procedimientos para esta solicitud de cambio.
       - Multiple resources (Recursos múltiples): en Parameter (Parámetro), seleccione uno de los parámetros disponibles de los manuales de procedimientos para esta solicitud de cambio. Esta selección refleja el tipo de recurso que se va a actualizar.

         Por ejemplo, si el manual de procedimientos de esta solicitud de cambio es `AWS-RetartEC2Instance`, puede elegir `InstanceId` y, a continuación, definir qué instancias se actualizan seleccionando una de las siguientes opciones:
         - Specify tags (Especificar etiquetas): ingrese un par de clave-valor con el que se etiquetarán todos los recursos que se van a actualizar.
         - Choose a resource group (Elegir un grupo de recursos): elija el nombre del grupo de recursos al que pertenecen todos los recursos que se van a actualizar.
         - Specify parameter values (Especificar valores de parámetros): identifique los recursos que se van a actualizar en la sección Runbook parameters (Parámetros del manual de procedimientos).
         - Target all instances (Abarcar todas las instancias): realice el cambio en todos los nodos administrados de las ubicaciones de destino.
    2. Si eligió Multiple resources (Recursos múltiples), expanda Rate control (Control de velocidad).

En Concurrency (Simultaneidad), ingrese un número y, a continuación, en la lista, seleccione si representa la cantidad o el porcentaje de destinos que el flujo de trabajo del manual de procedimientos puede actualizar al mismo tiempo.

En Error threshold (Límite de errores), ingrese un número y, a continuación, en la lista, seleccione si representa la cantidad o el porcentaje de destinos en los que la actualización puede producir error antes de que se detenga la operación.

15. Si en el paso anterior eligió Specify parameter values (Especificar valores de parámetros) para actualizar varios recursos, en la sección Runbook parameters (Parámetros del manual de procedimientos), especifique valores para los parámetros de entrada requeridos. Los valores de los parámetros que debe proporcionar se basan en el contenido de los manuales de procedimientos de Automation asociados a la plantilla de cambios que eligió.

    Por ejemplo, si la plantilla de cambios utiliza el manual de procedimientos `AWS-RetartEC2Instance`, debe ingresar uno o más ID de instancia para el parámetro InstanceId. También puede elegir Show interactive instance picker (Mostrar selector de instancias interactivo) y seleccionar las instancias disponibles una por una.
16. Elija Siguiente.
17. En la página Review and submit (Revisar y enviar), revise minuciosamente los recursos y las opciones que ha especificado para esta solicitud de cambio.

    Elija el botón Edit (Editar) en cualquier sección en la que desee realizar cambios.

    Cuando esté satisfecho con los detalles de la solicitud de cambio, elija Submit for approval (Enviar para aprobación).

Si se especificó un tema de Amazon SNS en la plantilla de cambios que eligió para la solicitud, las notificaciones se envían cuando la solicitud se rechaza o se aprueba. Si no recibe notificaciones por la solicitud, puede regresar a Change Manager para verificar su estado.

Creación de solicitudes de cambio (AWS CLI)

Puede crear una solicitud de cambio mediante AWS Command Line Interface (AWS CLI) con la especificación de opciones y parámetros para la solicitud de cambio en un archivo JSON y con el uso de la opción `--cli-input-json` para incluirlo en su comando.

Para crear una solicitud de cambio (AWS CLI)

1. Si aún no lo ha hecho, instale y configure la AWS CLI o Herramientas de AWS para PowerShell.

   Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.

2. Cree un archivo JSON en su equipo local con un nombre similar a `MyChangeRequest.json` y, a continuación, péguele el siguiente contenido.

   Reemplace *`espacio disponible`* con valores para la solicitud de cambio.

   > **Note**
   >
   > Este ejemplo JSON crea una solicitud de cambio mediante la plantilla de cambio `AWS-HelloWorldChangeTemplate` y el runbook `AWS-HelloWorld`. A fin de ayudarlo a adaptar este ejemplo para sus propias solicitudes de cambio, consulte StartChangeRequestExecution en la AWS Systems ManagerReferencia de la API para obtener información sobre todos los parámetros disponibles
   > Para obtener más información acerca de cómo aprobar las solicitudes de cambio, consulte Acerca de las aprobaciones de solicitudes de cambio.

```
{
    "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. En el directorio en el que creó el archivo JSON, ejecute el siguiente comando.

```
aws ssm start-change-request-execution --cli-input-json file://MyChangeRequest.json
```

El sistema devuelve información similar a la siguiente.

```
{
    "AutomationExecutionId": "b3c1357a-5756-4839-8617-2d2a4EXAMPLE"
}
```

Revisión y aprobación o rechazo de las solicitudes de cambio

Si está especificado como revisor de una solicitud de cambio en Change Manager, una herramienta de AWS Systems Manager, recibirá una notificación a través de un tema de Amazon Simple Notification Service (Amazon SNS) cuando una nueva solicitud de cambio esté en espera de su revisión.

> **Note**
>
> Esta funcionalidad depende de si se especificó un tema de Amazon SNS en la plantilla de cambios para enviar notificaciones de revisión. Para obtener más información, consulta Configuración de temas de Amazon SNS para las notificaciones de Change Manager.

Para revisar la solicitud de cambio, puede seguir el enlace que se incluye en la notificación o iniciar sesión en la AWS Management Console directamente y seguir los pasos descritos en este procedimiento.

> **Note**
>
> Si se asigna un tema de Amazon SNS a los revisores de una plantilla de cambios, se enviarán notificaciones a los suscriptores del tema cuando cambie el estado de la solicitud de cambio.
> Para obtener más información acerca de cómo aprobar las solicitudes de cambio, consulte Acerca de las aprobaciones de solicitudes de cambio.

Revisión y aprobación o rechazo de las solicitudes de cambio (consola)

Los siguientes procedimientos describen cómo utilizar la consola de Systems Manager para revisar y aprobar o rechazar solicitudes de cambio.

Para revisar y aprobar o rechazar una solicitud de cambio individual

1. Abra el enlace que figura en la notificación por email que recibió e inicie sesión en la AWS Management Console, que lo dirigirá a la solicitud de cambio para que la revise.
2. En la página de resumen, revise el contenido propuesto de la solicitud de cambio.

Para aprobar la solicitud de cambio, elija Approve (Aprobar). En el cuadro de diálogo, proporcione los comentarios que desee agregar a esta aprobación y, a continuación, elija Approve (Aprobar). El flujo de trabajo del manual de procedimientos representado por esta solicitud comienza a ejecutarse cuando está programado o tan pronto como los cambios no estén bloqueados por ninguna restricción.

-o bien-

Para rechazar la solicitud de cambio, elija Reject (Rechazar). En el cuadro de diálogo, proporcione los comentarios que desee agregar a este rechazo y, a continuación, elija Reject (Rechazar).

Para revisar y aprobar o rechazar solicitudes de cambio en bloque

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Change Manager.
3. Elija la pestaña Approvals (Aprobaciones).
4. (Opcional) Revise los detalles de las solicitudes pendientes de aprobación eligiendo el nombre de cada solicitud, y luego vuelva a la pestaña Approvals (Aprobaciones).
5. Seleccione la casilla de verificación de cada solicitud de cambio que desee aprobar.

   -o bien-

   Seleccione la casilla de verificación de cada solicitud de cambio que desee rechazar.
6. En el cuadro de diálogo, proporcione los comentarios que desee agregar al rechazo o la aprobación.
7. En función de si va a aprobar o rechazar las solicitudes de cambio seleccionadas, elija Approve (Aprobar) oReject (Rechazar).

Revisión y aprobación o rechazo de las solicitudes de cambio (línea de comandos)

El siguiente procedimiento describe cómo utilizar la AWS Command Line Interface (AWS CLI) (en Linux, macOS o Windows) para revisar y aprobar o rechazar una solicitud de cambio.

Para revisar y aprobar o rechazar una solicitud de cambio

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Cree un archivo JSON en su equipo local que especifique los parámetros para su llamada a la AWS CLI.

```
{
  "OpsItemFilters":
  [
    {
      "Key": "OpsItemType",
      "Values": ["/aws/changerequest"],
      "Operator": "Equal"
    }
  ],
  "MaxResults": number
}
```

   Puede filtrar los resultados de un aprobador específico indicando el nombre de recurso de Amazon (ARN) del aprobador en el archivo JSON. A continuación se muestra un ejemplo.

```
{
  "OpsItemFilters":
  [
    {
      "Key": "OpsItemType",
      "Values": ["/aws/changerequest"],
      "Operator": "Equal"
    },
    {
      "Key": "ChangeRequestByApproverArn",
      "Values": ["arn:aws:iam::account-id:user/user-name"],
      "Operator": "Equal"
    }
  ],
  "MaxResults": number
}
```

3. Ejecute el siguiente comando para ver la cantidad máxima de solicitudes de cambio que especificó en el archivo 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. Ejecute el siguiente comando para aprobar o rechazar una solicitud de cambio.

   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 se especificó un tema de Amazon SNS en la plantilla de cambios que eligió para la solicitud, las notificaciones se envían cuando la solicitud se rechaza o se aprueba. Si no recibe notificaciones por la solicitud, puede regresar a Change Manager para verificar su estado. Para obtener información acerca de otras opciones cuando utilice este comando, consulte send-automation-signal, en la sección AWS Systems Manager de la Referencia de comandos de la AWS CLI.

Revisión de los detalles, las tareas y los plazos de las solicitudes de cambio (consola)

Puede consultar la información acerca de una solicitud de cambio, incluidas las solicitudes para las que ya se han procesado los cambios, en el panel de Change Manager, una herramienta de AWS Systems Manager. Estos detalles incluyen un enlace a la operación de Automation que ejecuta los manuales de procedimientos que efectúan el cambio. Cuando se crea la solicitud, se genera un ID de ejecución de Automation, pero el proceso recién se ejecuta cuando se han otorgado todas las aprobaciones y no hay restricciones que bloqueen el cambio.

Para revisar los detalles, las tareas y los plazos de las solicitudes de cambio

1. En el panel de navegación, elija Change Manager.
2. Elija la pestaña Requests (Solicitudes).
3. En la sección Change requests (Solicitudes de cambio), busque la solicitud de cambio que desee revisar.

   Puede utilizar las opciones de Create date range (Crear rango de fechas) para limitar los resultados a un periodo específico.

   Puede filtrar las solicitudes según las siguientes propiedades:

   - `Status`
   - `Request ID`
   - `Approver`
   - `Requester`

   Por ejemplo, para ver los detalles acerca de todas las solicitudes de cambio que se han completado de forma correcta en las últimas 24 horas, realice el siguiente procedimiento:

   1. Para Create date range (Crear rango de fechas), elija 1d.
   2. En el cuadro de búsqueda, seleccione Status, CompletedWithSuccess (Estado, Completado de forma correcta).
   3. En los resultados, elija el nombre de la solicitud de cambio completada correctamente cuyos resultados desea revisar.
4. Vea información acerca de la solicitud de cambio en las siguientes pestañas:

- Request details (Detalles de la solicitud): consulte los detalles básicos de la solicitud de cambio, incluido el solicitante, la plantilla de cambios y los manuales de procedimientos de Automation seleccionados para el cambio. También puede seguir un enlace a los detalles de la operación de Automation y ver la información sobre los parámetros del manual de procedimientos especificado en la solicitud, las alarmas de Amazon CloudWatch asignadas a la solicitud de cambio, las aprobaciones y los comentarios proporcionados para la solicitud.
- Task (Tarea): consulte información acerca de la tarea en el cambio, incluido el estado de la tarea para las solicitudes de cambio completadas, los recursos de destino, los pasos de los manuales de procedimientos de Automation asociados y los detalles del límite de errores y simultaneidad.
- Timeline (Plazo): vea un resumen de todos los eventos asociados a la solicitud de cambio, enumerados por fecha y hora. El resumen indica cuándo se creó la solicitud de cambio, las acciones de los aprobadores asignados, una nota de para cuándo se programa la ejecución de las solicitudes de cambio aprobadas, los detalles del flujo de trabajo del manual de procedimientos y los cambios de estado para el proceso general de cambio y para cada paso del manual de procedimientos.
- Associated events (Eventos asociados): vea los detalles auditables sobre las solicitudes de cambio que se registran en AWS CloudTrail Lake. Los detalles incluyen qué acciones de la API se ejecutaron, los parámetros de solicitud incluidos para esas acciones, la cuenta de usuario que ejecutó la acción, los recursos actualizados durante el proceso y más.

  Cuando habilita el seguimiento de eventos de CloudTrail Lake, este crea un almacén de datos de eventos para los eventos relacionados con sus solicitudes de cambio. Los detalles del evento están disponibles para la cuenta u organización en la que se ejecutó la solicitud de cambio. Puede activar el seguimiento de eventos de CloudTrail Lake desde cualquier solicitud de cambio en su cuenta u organización. Para obtener información sobre cómo habilitar la integración de CloudTrail Lake y crear un almacén de datos de eventos, consulte Supervisión de los eventos de las solicitudes de cambio.

  > **Note**
  >
  > El uso de CloudTrail Lake conlleva un cargo. Consulte Precios de AWS CloudTrail para obtener más información.

Visualización de recuentos agregados de solicitudes de cambio (línea de comandos)

Puede consultar los recuentos agregados de solicitudes de cambio en Change Manager, una herramienta de AWS Systems Manager, mediante la operación GetOpsSummary de la API. Esta operación de la API puede regresar recuentos de una única Cuenta de AWS en una única Región de AWS o de varias cuentas y regiones.

> **Note**
>
> Si desea ver recuentos agregados de las solicitudes de cambio para varias Cuentas de AWS y Regiones de AWS, debe establecer y configurar una sincronización de datos de recursos. Para obtener más información, consulte Cómo crear una sincronización de datos de recursos para Inventory.

El procedimiento siguiente describe cómo utilizar la AWS Command Line Interface (AWS CLI) (en Linux, macOS o Windows) para ver los recuentos agregados de las solicitudes de cambio.

Para ver los recuentos agregados de las solicitudes de cambio

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.
2. Ejecute uno de los siguientes comandos.

   Única cuenta y región

   Este comando regresa un recuento de todas las solicitudes de cambio de la Cuenta de AWS y la Región de AWS para las cuales la sesión de la AWS CLI está configurada.

   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
```

La llamada regresa información similar a la siguiente.

```
{
    "Entities": [
        {
            "Data": {
                "AWS:OpsItem": {
                    "Content": [
                        {
                            "Count": "38",
                            "Status": "Open"
                        }
                    ]
                }
            }
        }
    ]
}
```

Varias cuentas o regiones

Este comando regresa un recuento de todas las solicitudes de cambio para las Cuentas de AWS y las Regiones de AWS especificadas en la sincronización de datos de recursos.

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
```

La llamada regresa información similar a la siguiente.

```
{
    "Entities": [
        {
            "Data": {
                "AWS:OpsItem": {
                    "Content": [
                        {
                            "Count": "43",
                            "Status": "Open"
                        },
                        {
                            "Count": "2",
                            "Status": "Resolved"
                        }
                    ]
                }
            }
        }
    ]
}
```

Varias cuentas y una región específica

Este comando regresa un recuento de todas las solicitudes de cambio para las Cuentas de AWS especificadas en la sincronización de datos de recursos. Sin embargo, solo regresa datos de la región especificada en el comando.

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
```

Varias cuentas y regiones con resultados agrupados por región

Este comando regresa un recuento de todas las solicitudes de cambio para las Cuentas de AWS y las Regiones de AWS especificadas en la sincronización de datos de recursos. El resultado muestra información de recuentos por región.

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"}]}]'
```

La llamada regresa información similar a la siguiente.

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

Varias cuentas y regiones con resultados agrupados por cuentas y regiones

Este comando regresa un recuento de todas las solicitudes de cambio para las Cuentas de AWS y las Regiones de AWS especificadas en la sincronización de datos de recursos. El resultado agrupa la información de recuentos por cuentas y regiones.

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"}]}]}]
```

La llamada regresa información similar a la siguiente.

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

Auditoría y registro de la actividad de Change Manager

Puede auditar la actividad de Change Manager, una herramienta de AWS Systems Manager, mediante las alarmas de Amazon CloudWatch y AWS CloudTrail.

Para obtener más información acerca de cómo auditar y registrar las opciones de Systems Manager, consulte Registro y monitorización en AWS Systems Manager.

Auditoría de la actividad de Change Manager con las alarmas de CloudWatch

Puede configurar una alarma de CloudWatch y asignarla a una plantilla de cambios. Si se cumple alguna de las condiciones definidas en la alarma, se llevan a cabo las acciones especificadas para ella. En la configuración de la alarma, puede especificar un tema de Amazon Simple Notification Service (Amazon SNS) para efectuar una notificación cuando se cumple alguna condición de la alarma.

Para obtener más información acerca de la creación de una plantilla de Change Manager, consulte Uso de las plantillas de cambios.

Para obtener más información acerca de la creación de alarmas de CloudWatch, consulte Using CloudWatch Alarms (Uso de las alarmas de CloudWatch) en la Guía del usuario de Amazon CloudWatch.

Auditoría de la actividad de Change Manager con CloudTrail

CloudTrail registra las llamadas a la API realizadas en la consola de Systems Manager, la AWS Command Line Interface (AWS CLI) y el SDK de Systems Manager. Puede ver la información en la consola de CloudTrail o en un bucket de Amazon Simple Storage Service (Amazon S3), donde se almacena. Se utiliza un bucket para todos los registros de CloudTrail de su cuenta.

Los registros de las acciones de Change Manager muestran la creación de documentos de plantillas de cambios, las aprobaciones y los rechazos de las solicitudes y las plantillas de cambios, la actividad generada por los manuales de procedimientos de Automation y mucho más. Para obtener más información acerca de cómo ver y utilizar los registros de CloudTrail de la actividad de Systems Manager, consulte Registro de llamadas a la API de AWS Systems Manager con AWS CloudTrail.

Solución de problemas de Change Manager

Utilice la siguiente información como ayuda para solucionar problemas con Change Manager, una herramienta de AWS Systems Manager.

Temas

- Error “Group {GUID} not found” (Grupo {GUID} no encontrado) durante las aprobaciones de las solicitudes de cambio cuando se usa Active Directory (grupos)

Error “Group *{GUID}* not found” (Grupo {GUID} no encontrado) durante las aprobaciones de las solicitudes de cambio cuando se usa Active Directory (grupos)

Problema: cuando AWS IAM Identity Center (Centro de identidades de IAM) se utiliza para la administración de identidades de usuarios, un miembro de un grupo de Active Directory al que se le conceden permisos de aprobación en Change Manager recibe un error de “not authorized” (no autorizado) o “group not found” (grupo no encontrado).

- Solución: cuando selecciona grupos de Active Directory en el Centro de identidades de IAM para acceder a la AWS Management Console, el sistema programa una sincronización periódica que copia la información de esos grupos de Active Directory en el Centro de identidades de IAM. Este proceso debe completarse antes de que los usuarios autorizados a través de la pertenencia a grupos de Active Directory puedan aprobar correctamente una solicitud. Para obtener más información, consulte Conexión al directorio de Microsoft AD en la Guía del usuario de AWS IAM Identity Center.

Documentos de AWS Systems Manager

Un documento de AWS Systems Manager (documento de SSM) define las acciones que Systems Manager realiza en las instancias administradas. Systems Manager incluye más de 100 documentos preconfigurados que puede utilizar especificando los parámetros en tiempo de ejecución. Los documentos preconfigurados se pueden encontrar en la consola de documentos de Systems Manager en la pestaña Owned by Amazon (Propiedad de Amazon) o si especifica Amazon para el filtro `Owner` (Propietario) al llamar a la operación de la API `ListDocuments`. Los documentos usan JSON (JavaScript Object Notation, notación de objetos de JavaScript) o YAML e incluyen los pasos y los parámetros que especifique.

Para mejorar la seguridad, a partir del 14 de julio de 2025, los documentos SSM admiten la interpolación de variables de entorno al procesar los parámetros. Esta característica, disponible en la versión 2.2 del esquema y en la versión 3.3.2746.0 o superior de SSM Agent, ayuda a prevenir los ataques de inyección de comandos.

Para comenzar a utilizar documentos de SSM, abra la consola de Systems Manager. En el panel de navegación, elija Documentos.

> **Important**
>
> En Systems Manager, un documento de SSM propiedad de Amazon es un documento creado y administrado por Amazon Web Services. Los documentos que son propiedad de Amazon incluyen un prefijo como `AWS-*` en el nombre del documento. Se considera que el propietario del documento es Amazon, no una cuenta de usuario específica de AWS. Estos documentos están a disposición del público para que los utilicen todos los usuarios.

¿Cómo puede beneficiar la herramienta Documentos a mi organización?

Documentos, una herramienta de AWS Systems Manager, ofrece los siguientes beneficios:

- Categorías de documentos

  Para ayudarlo a encontrar los documentos que necesita, elija una categoría según el tipo de documento que esté buscando. Para ampliar la búsqueda, puede elegir varias categorías del mismo tipo de documento. No se admite la selección de categorías de distintos tipos de documento. Las categorías solo se admiten para documentos propiedad de Amazon.

- Versiones de los documentos

  Puede crear y guardar distintas versiones de los documentos. A continuación, puede especificar una versión predeterminada para cada documento. La versión predeterminada de un documento se puede actualizar a una versión más reciente o revertir a una versión anterior del documento. Cuando cambia el contenido de un documento, Systems Manager incrementa automáticamente la versión del documento. Especifique la versión de un documento en la consola, los comandos de AWS Command Line Interface (AWS CLI) o las llamadas a la API para recuperar o utilizar cualquier versión de un documento.

- Personalizar documentos según sus necesidades

  Si desea personalizar los pasos y las acciones en un documento, puede crear el suyo propio. El sistema almacena el documento con su Cuenta de AWS en la Región de AWS en la que lo cree. Para obtener más información acerca de cómo crear un documento de SSM, consulte Crear contenido en el documento de SSM.

- Etiquetar documentos

  Puede etiquetar sus documentos para identificar rápidamente uno o varios documentos en función de las etiquetas que les haya asignado. Por ejemplo, puede etiquetar documentos para entornos específicos, departamentos, usuarios, grupos o períodos. También puede restringir el acceso a documentos mediante la creación de una política de AWS Identity and Access Management (IAM) que especifique las etiquetas a las que podrá obtener acceso un usuario o un grupo.

- Compartir documentos

  Puede hacer que los documentos sean públicos o compartirlos con determinadas Cuentas de AWS en la misma Región de AWS. Compartir documentos entre cuentas puede resultar útil si, por ejemplo, desea que todas las instancias de Amazon Elastic Compute Cloud (Amazon EC2) que proporcione a los clientes o a los empleados tengan la misma configuración. Además de mantener actualizadas las aplicaciones o las revisiones en las instancias, quizá le interese restringir las instancias de cliente en determinadas actividades. O bien, quizá desee asegurarse de que las instancias utilizadas por las cuentas de los empleados en toda la empresa dispongan de acceso a recursos internos específicos. Para obtener más información, consulte Uso compartido de documentos de SSM.

¿Quién debería utilizar Documentos?

- Todo cliente de AWS que desee utilizar las herramientas de Systems Manager para mejorar su eficiencia operativa a escala, reducir los errores asociados a la intervención manual y reducir el tiempo de resolución de problemas comunes.
- Expertos en infraestructura que deseen automatizar las tareas de implementación y configuración.
- Administradores que deseen resolver problemas comunes de forma fiable, mejorar la eficiencia de la solución de problemas y reducir las operaciones repetitivas.
- Usuarios que deseen automatizar una tarea que normalmente realizan de forma manual.

¿Cuáles son los tipos de documentos de SSM?

En la siguiente tabla, se describen los distintos tipos de documentos de SSM y los usos.

Tipo	Utilizar con	Detalles
ApplicationConfiguration ApplicationConfigurationSchema	AWS AppConfig	AWS AppConfig, una herramienta de AWS Systems Manager, le permite crear, administrar e implement ar rápidamente configura ciones de aplicaciones. Para almacenar los datos de configuración en un documento SSM, puede crear un documento que utilice el tipo de documento `ApplicationConfigu ration` . Para obtener más información, consulte Configuración libre en la Guía del usuario de AWS AppConfig. Si crea una configuración en un documento SSM, debe

Tipo	Utilizar con	Detalles
		especificar un esquema JSON correspondiente. El esquema utiliza el tipo de documento `ApplicationConfigu rationSchema` y, al igual que un conjunto de reglas, define las propiedades permitidas para cada ajuste de configuración de la aplicació n. Para obtener más informaci ón, consulte Acerca de los validadores en la Guía del usuario de AWS AppConfig.

Tipo	Utilizar con	Detalles
Manual de procedimientos de Automation	Automatización State Manager Maintenance Windows	Utilice los manuales de procedimientos de Automation a la hora de realizar tareas de implementación y mantenimiento comunes como, por ejemplo, crear o actualizar una Amazon Machine Image (AMI). State Manager utiliza los documentos de automatización para aplicar una configuración. Estas acciones se pueden ejecutar en uno o varios destinos en cualquier momento durante el ciclo de vida de una instancia. Maintenance Windows utiliza manuales de procedimientos de Automation para realizar tareas de implementación y mantenimiento comunes en función de la programación especificada. Todos los documentos de Automation compatibles con sistemas operativos basados en Linux también se admiten en instancias EC2 para macOS.

Tipo	Utilizar con	Detalles
Documento de calendario de cambios	Change Calendar	Change Calendar, una herramienta de AWS Systems Manager, usa el tipo de documento `ChangeCalendar`. Un documento de Change Calendar almacena una entrada de calendario y eventos asociados que pueden permitir o impedir que las acciones de Automation cambien el entorno. En Change Calendar, un documento almacena datos de iCalendar 2.0 en texto sin formato. Change Calendar no es compatible con instancias EC2 para macOS.

Tipo	Utilizar con	Detalles
Plantilla de AWS CloudFormation	AWS CloudFormation	Estas plantillas de AWS CloudFormation describen los recursos que desea aprovisionar en sus pilas de CloudFormation. El almacenamiento de plantillas de CloudFormation como documentos de Systems Manager le permite beneficiarse de las características de los documentos de Systems Manager. Estos incluyen crear y comparar varias versiones de su plantilla y compartir su plantilla con otras cuentas en la misma Región de AWS. Puede crear y editar plantillas y pilas de CloudFormation con Application Manager, una herramienta de Systems Manager. Para obtener más información, consulte Trabajo con plantillas y pilas de AWS CloudFormation en Application Manager.

Tipo	Utilizar con	Detalles
Documento de comandos	Run Command State Manager Maintenance Windows	Run Command, una herramienta de AWS Systems Manager, utiliza documentos de comandos para ejecutar comandos. State Manager, una herramienta de AWS Systems Manager, utiliza documentos de comandos para aplicar una configuración. Estas acciones se pueden ejecutar en uno o más destinos en cualquier momento durante el ciclo de vida de una instancia. Maintenance Windows, una herramienta de AWS Systems Manager, utiliza los documentos de comandos para aplicar una configuración en función de la programación especificada. La mayoría de los documentos de Command son compatibles con todos los sistemas operativos Linux y Windows Server, que, a su vez, son compatibles con Systems Manager. Los siguientes documentos de Command se admiten en instancias EC2 para macOS: • `AWS-ConfigureAWSPackage`

Tipo	Utilizar con	Detalles
		• `AWS-RunPatchBaseline` • `AWS-RunPatchBaselineAssociation` • `AWS-RunShellScript`
Plantilla de paquete de conformidad de AWS Config	AWS Config	Las plantillas de paquetes de conformidad de AWS Config son documentos con formato YAML que se utilizan para crear paquetes de conformidad que contienen la lista de reglas administradas o personalizadas de AWS Config y acciones de corrección. Para obtener más información, consulte Conformance Packs (Paquetes de conformidad).

Tipo	Utilizar con	Detalles
Documento de paquete	Distributor	En Distributor, una herramienta de AWS Systems Manager, un paquete está representado por un documento de SSM. Un documento de paquete incluye archivos de almacenamiento ZIP asociados que contienen software o recursos para instalar en las instancias administradas. La creación de un paquete en Distributor crea el documento del paquete. Distributor no es compatible con Oracle Linux ni con las instancias administradas de macOS.

Tipo	Utilizar con	Detalles
Documento de política	State Manager	Inventario, una herramienta de AWS Systems Manager, utiliza el documento de política `AWS-GatherSoftwareInventory` con una asociación de State Manager para recopilar datos de inventario de instancias administradas. Cuando crea sus propios documentos de SSM, los manuales de procedimientos de Automation y los documentos de Command son el método preferido para aplicar una política a una instancia administrada. Systems Manager Inventory y el documento de política de `AWS-GatherSoftwareInventory` se admiten en todos los sistemas operativos compatibles con Systems Manager.

Tipo	Utilizar con	Detalles
Plantilla de análisis posterior a incidentes	Análisis posterior a incidentes de Incident Manager	Incident Manager utiliza la plantilla de análisis posterior a incidentes para crear un análisis basado en prácticas recomendadas de administración de operaciones de AWS. Utilice la plantilla para crear un análisis que su equipo pueda utilizar para identificar mejoras en la respuesta a incidentes.

Tipo	Utilizar con	Detalles
Documento de sesión	Session Manager	Session Manager, una herramienta de AWS Systems Manager, utiliza documentos de sesión para determinar qué tipo de sesión iniciar, como una sesión de reenvío de puertos, una sesión para ejecutar un comando interactivo o una sesión para crear un túnel SSH. Los documentos de sesión son compatibles con todos los sistemas operativos Linux y Windows Server que, a su vez, son compatibles con Systems Manager. Los siguientes documentos de Command se admiten en instancias EC2 para macOS: • `AWS-PasswordReset` • `AWS-StartInteractiveCommand` • `AWS-StartPortForwardingSession` • `AWS-StartPortForwardingSessionToSocket` • `AWS-StartSSHSession`

Cuotas de documentos de SSM

Para obtener más información sobre las cuotas de los documentos de SSM, consulte Service Quotas de Systems Manager en la Referencia general de Amazon Web Services.

Temas

- Componentes del documento
- Crear contenido en el documento de SSM
- Trabajo con documentos
- Solución de problemas de gestión de parámetros

Componentes del documento

Esta sección incluye información sobre los componentes de los documentos de SSM.

Contenido

- Esquemas, características y ejemplos
- Elementos y parámetros de datos
- Referencia de complementos del documento de comandos

Esquemas, características y ejemplos

Los documentos de AWS Systems Manager (SSM) utilizan las siguientes versiones de esquema.

- Los documentos del tipo `Command` pueden utilizar la versión de esquema 1.2, 2.0 y 2.2. Si utiliza documentos de esquema 1.2, le recomendamos que cree documentos que utilicen la versión de esquema 2.2.
- Los documentos del tipo `Policy` deben utilizar la versión de esquema 2.0 o posterior.
- Los documentos del tipo `Automation` deben utilizar la versión de esquema 0.3.
- Los documentos del tipo `Session` deben utilizar la versión de esquema 1.0.
- Puede crear documentos en JSON o YAML.

Para obtener más información acerca del esquema del documento de `Session`, consulte Esquema del documento de Session.

Si utiliza la versión de esquema más reciente para los documentos de tipo `Command` y `Policy`, puede aprovechar las siguientes características.

Características de los documentos con la versión de esquema 2.2

Característica	Detalles
Edición de documentos	Ahora los documentos pueden actualizarse. Con la versión 1.2, cualquier actualización de un documento requería guardarlo con otro nombre.
Control de versiones automático	Cualquier actualización de un documento crea una versión nueva. No es una versión de esquema, sino una versión del documento.
Versión predeterminada	Si tiene varias versiones de un documento, puede especificar qué versión es el documento predeterminado.
Secuenciación	Los complementos o los pasos de un documento se ejecutan en el orden especific ado.
Compatibilidad multiplataforma	La compatibilidad multiplataforma le permite especificar diferentes sistemas operativos para distintos complementos dentro del mismo documento de SSM. La compatibilidad multiplat aforma utiliza el parámetro `precondition` dentro de un paso.
Interpolación de parámetros	Permite interpolar los parámetros de cadena en las variables de entorno antes de la ejecución del comando, lo que proporciona una mayor seguridad contra las inyecciones de comandos. Cuando se establece en `ENV_VAR`, el agente crea una variable de entorno denominada `SSM_`*`parameter-name`* que contiene el valor del parámetro.

> **Note**
>
> El AWS Systems Manager SSM Agent de las instancias debe mantenerse actualizado con la versión más reciente para poder utilizar las características nuevas de Systems Manager y las características del documento de SSM. Para obtener más información, consulte Actualización de SSM Agent mediante Run Command.

La siguiente tabla enumera las diferencias entre las versiones de esquema principales.

Versión 1.2	Versión 2.2 (versión más reciente)	Detalles
runtimeConfig	mainSteps	En la versión 2.2, la sección `mainSteps` sustituye a la `runtimeConfig` . La sección `mainSteps` permite a Systems Manager ejecutar los pasos de forma secuencial.
propiedades	inputs	En la versión 2.2, la sección `inputs` sustituye la sección `properties` . La sección `inputs` acepta parámetros en los pasos.
Comandos de la de	runCommand	En la versión 2.2, la sección `inputs` toma el parámetro `runCommand` en lugar del parámetro `commands`.
id	acción	En la versión 2.2, `Action` sustituye a `ID`. Se trata tan solo de un cambio de nombre.

Versión 1.2	Versión 2.2 (versión más reciente)	Detalles
no se usa	nombre	En la versión 2.2, `name` es cualquier nombre definido por el usuario para un paso.

Uso del parámetro precondition

Con la versión de esquema 2.2 o posterior, puede utilizar el parámetro `precondition` para especificar el sistema operativo de destino de cada complemento o para validar los parámetros de entrada que definió en su documento de SSM. El parámetro `precondition` admite hacer referencia a los parámetros de entrada de su documento de SSM, y `platformType` utilizando los valores de `Linux`, `MacOS`, y `Windows`. Solo el operador `StringEquals` es compatible.

En el caso de documentos que utilizan la versión de esquema 2.2 o posterior, si no se especifica `precondition`, cada complemento se ejecuta u omite en función de la compatibilidad del complemento con el sistema operativo. La compatibilidad de los complementos con el sistema operativo se evalúa antes de `precondition`. En el caso de los documentos que utilizan el esquema 2.0 o anterior, los complementos incompatibles generarán un error.

Por ejemplo, en un documento con la versión de esquema 2.2, si no se especifica `precondition` y se incluye el complemento `aws:runShellScript`, el paso se ejecuta en las instancias de Linux, pero el sistema lo omite en las instancias de Windows Server, ya que `aws:runShellScript` no es compatible con las instancias de Windows Server. Sin embargo, en el caso de un documento con versión de esquema 2.0, si especifica el complemento `aws:runShellScript` y, a continuación, ejecuta el documento en una instancia de Windows Server, se produce un error en la ejecución. Puede ver un ejemplo del parámetro de condición previa en un documento de SSM más adelante en esta sección.

Versión de esquema 2.2

Elementos de nivel superior

En el siguiente ejemplo, se muestran los elementos de nivel superior de un documento de SSM que utiliza la versión 2.2 del esquema.

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 }}"
```

```
         }
      }
   ]
}
```

Ejemplo de la versión 2.2 del esquema

En el ejemplo siguiente, se utiliza el complemento `aws:runPowerShellScript` para ejecutar un comando de PowerShell en las instancias de destino.

YAML

```
---
schemaVersion: "2.2"
description: "Example document"
parameters:
  Message:
    type: "String"
    description: "Example parameter"
    default: "Hello World"
    allowedValues:
    - "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",
         "allowedValues": ["Hello World"]
      }
```

```
   },
   "mainSteps": [
      {
         "action": "aws:runPowerShellScript",
         "name": "example",
         "inputs": {
            "timeoutSeconds": "60",
            "runCommand": [
               "Write-Output {{Message}}"
            ]
         }
      }
   ]
}
```

Ejemplos del parámetro de condición previa de una versión de esquema 2.2

La versión de esquema 2.2 ofrece compatibilidad multiplataforma. Esto significa que dentro de un mismo documento de SSM puede especificar diferentes sistemas operativos para distintos complementos. La compatibilidad multiplataforma utiliza el parámetro `precondition` dentro de un paso, tal y como se muestra en el siguiente ejemplo. También puede utilizar el parámetro `precondition` para validar los parámetros de entrada que haya definido en el documento de SSM. Puede ver esto en el segundo caso de los siguientes ejemplos.

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"
```

```
            ]
         }
      }
   ]
}
```

Ejemplo de interpolación de la versión 2.2 del esquema con versiones de SSM Agent anteriores a la 3.3.2746.0

En las versiones de SSM Agent anteriores a la 3.3.2746.0, el agente ignora el parámetro `interpolationType` y, en su lugar, realiza una sustitución de cadena sin procesar. Si hace referencia a `SSM_`*`parameter-name`* de forma explícita, debe configurarlo de forma explícita. En el siguiente ejemplo para Linux, se hace referencia explícita a la variable de entorno `SSM_Message`.

```
{
    "schemaVersion": "2.2",
    "description": "An example document",
    "parameters": {
        "Message": {
            "type": "String",
            "description": "Message to be printed",
            "default": "Hello",
            "interpolationType" : "ENV_VAR",
      "allowedPattern: "^[^"]*$"

        }
    },
    "mainSteps": [{
        "action": "aws:runShellScript",
        "name": "printMessage",
        "inputs": {
            "runCommand": [
              "if [ -z "${SSM_Message+x}" ]; then",
              "    export SSM_Message=\"{{Message}}\"",
              "fi",
              "",
              "echo $SSM_Message"
            ]
        }
    }
```

```
}
```

> **Note**
>
> `allowedPattern` no es técnicamente obligatorio si un documento SSM no utiliza doble llave: `{{ }}`

Ejemplo de la versión 2.2 del esquema State Manager

Puede utilizar el siguiente documento de SSM con State Manager, una herramienta de Systems Manager, para descargar e instalar el software antivirus ClamAV. State Manager aplica una configuración específica, lo que significa que cada vez que se ejecuta la asociación de State Manager, el sistema comprueba si el software ClamAV está instalado. En caso contrario, State Manager vuelve a ejecutar este documento.

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"
            ]
         }
      }
   ]
}
```

Ejemplo de inventario con la versión 2.2 del esquema

Puede utilizar el siguiente documento de SSM con State Manager para recopilar metadatos de inventario de las instancias.

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 }}"
         }
      }
   ]
}
```

Ejemplo de la versión 2.2 del esquema **`AWS-ConfigureAWSPackage`**

El siguiente ejemplo muestra el documento de `AWS-ConfigureAWSPackage`. La sección `mainSteps` incluye el complemento `aws:configurePackage` en el paso `action`.

> **Note**
>
> En sistemas operativos Linux, solo son compatibles los paquetes `AWSSupport-EC2Rescue` y `AmazonCloudWatchAgent`.

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 }}"
         }
      }
   ]
}
```

Versión de esquema 1.2

El siguiente ejemplo muestra los elementos de nivel superior de un documento con la versión de esquema 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"
            }
         ]
      }
   }
}
```

Ejemplo de la versión 1.2 del esquema **`aws:runShellScript`**

El siguiente ejemplo muestra el documento de SSM `AWS-RunShellScript`. La sección runtimeConfig incluye el complemento `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 }}"
                }
            ]
        }
    }
}
```

Versión de esquema 0.3

Elementos de nivel superior

El siguiente ejemplo muestra los elementos de nivel superior de un manual de procedimientos de automatización con la versión de esquema 0.3 o posterior en formato 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
        }
    }
}
```

Ejemplo de manual de procedimientos de automatización YAML

En el siguiente ejemplo, se muestra el contenido de un manual de procedimientos de automatización en formato YAML. Este ejemplo de trabajo de la versión 0.3 del esquema del documento también demuestra el uso de Markdown para dar formato a las descripciones del documento.

```
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
```

Ejemplos de gestión segura de parámetros

Los siguientes ejemplos demuestran la gestión seguro de parámetros mediante una variable de entorno `interpolationType`.

Ejecución básica de comandos de forma segura

En este ejemplo se muestra cómo gestionar de forma segura un parámetro de comando:

> **Note**
>
> `allowedPattern` no es técnicamente obligatorio en los documentos SSM que no utilizan doble llave: `{{ }}`

YAML

```
---

schemaVersion: '2.2'
description: An example document.
parameters:
  Message:
    type: String
    description: "Message to be printed"
    default: Hello
    interpolationType: ENV_VAR
    allowedPattern: "^[^"]*$"
mainSteps:
  - action: aws:runShellScript
    name: printMessage
    precondition:
      StringEquals:
        - platformType
        - Linux
    inputs:
      runCommand:
        - echo {{Message}}
```

JSON

```
{
    "schemaVersion": "2.2",
    "description": "An example document.",
    "parameters": {
        "Message": {
            "type": "String",
            "description": "Message to be printed",
            "default": "Hello",
            "interpolationType": "ENV_VAR",
            "allowedPattern": "^[^"]*$"
```

```
        }
    },
    "mainSteps": [{
        "action": "aws:runShellScript",
        "name": "printMessage",
        "precondition": {
           "StringEquals": ["platformType", "Linux"]
        },
        "inputs": {
            "runCommand": [
              "echo {{Message}}"
            ]
        }
    }]
}
```

Uso de parámetros en lenguajes interpretados

Este ejemplo demuestra la gestión segura de parámetros en Python:

YAML

```
---
schemaVersion: '2.2'
description: 'Secure Python script execution'
parameters:
  inputData:
    type: String
    description: 'Input data for processing'
    interpolationType: 'ENV_VAR'
mainSteps:
  - action: aws:runPowerShellScript
    name: runPython
    inputs:
      runCommand:
        - |
          python3 -c '
          import os
          import json

          # Safely access parameter through environment variable
          input_data = os.environ.get("SSM_inputData", "")
```

```
          # Process the data
          try:
              processed_data = json.loads(input_data)
              print(f"Successfully processed: {processed_data}")
          except json.JSONDecodeError:
              print("Invalid JSON input")
          '
```

Ejemplo de compatibilidad con versiones anteriores

En este ejemplo se muestra cómo gestionar los parámetros de forma segura y mantener la compatibilidad con versiones anteriores:

YAML

```
---
schemaVersion: '2.2'
description: 'Backwards compatible secure parameter handling'
parameters:
  userInput:
    type: String
    description: 'User input to process'
    interpolationType: 'ENV_VAR'
    allowedPattern: '^[^"]*$'

mainSteps:
  - action: aws:runShellScript
    name: processInput
    inputs:
      runCommand:
        - |
          # Handle both modern and legacy agent versions
          if [ -z "${SSM_userInput+x}" ]; then
              # Legacy agent - fall back to direct parameter reference
              export SSM_userInput="{{userInput}}"
          fi

          # Process the input securely
          echo "Processing input: $SSM_userInput"
```

> **Note**
>
> `allowedPattern` no es técnicamente obligatorio en los documentos SSM que no utilizan doble llave: `{{ }}`

Prácticas recomendadas de seguridad de parámetros

Siga estas prácticas recomendadas cuando gestione parámetros en documentos SSM:

- Utilice la interpolación de variables de entorno: utilice siempre `interpolationType: "ENV_VAR"` para los parámetros de cadena que se utilizarán en la ejecución de comandos.
- Implemente la validación de entradas: utilice `allowedPattern` para restringir los valores de los parámetros a patrones seguros.
- Gestione los sistemas heredados: incluya una lógica alternativa para las versiones de SSM Agent anteriores que no admitan la interpolación de variables de entorno.
- Escape los caracteres especiales: cuando utilice valores de parámetros en los comandos, escape correctamente los caracteres especiales para evitar que el intérprete de comandos los interprete.
- Limite el alcance de los parámetros: utilice los patrones de parámetros más restrictivos posibles para su caso de uso.

Elementos y parámetros de datos

En este tema se describen los elementos de datos que se utilizan en los documentos de SSM. La versión del esquema utilizada para crear un documento define la sintaxis y los elementos de datos que el documento acepta. Se recomienda utilizar la versión de esquema 2.2 o una versión posterior para los documentos de Command. Los manuales de procedimientos de Automation utilizan la versión de esquema 0.3. Asimismo, los manuales de procedimientos de Automation admiten el uso de Markdown, un lenguaje de marcado que le permite agregar descripciones de estilo wiki a documentos y pasos individuales dentro del documento. Para obtener más información acerca del uso de Markdown, consulte Uso de Markdown en la consola en la Guía de introducción a la AWS Management Console.

En la siguiente sección se describen los elementos de datos que puede incluir en un documento de SSM.

Elementos de datos de nivel superior

schemaVersion

La versión de esquema que utilizar.

Tipo: versión

Obligatorio: sí

description

La información que proporciona para describir el propósito del documento. También puede utilizar este campo para determinar si un parámetro requiere un valor para que se ejecute un documento o si es opcional proporcionar un valor para el parámetro. En los ejemplos de este tema, se pueden ver los parámetros obligatorios y opcionales.

Tipo: cadena

Requerido: no

parameters

Una estructura que define los parámetros que acepta el documento.

Para mejorar la seguridad al administrar los parámetros de cadena, puede utilizar la interpolación de variables de entorno si especifica la propiedad `interpolationType`. Cuando se establece en `ENV_VAR`, el sistema crea una variable de entorno denominada `SSM_`*`parameter-name`* que contiene el valor del parámetro.

A continuación se incluye un ejemplo de un parámetro que utiliza una variable de entorno `interpolationType`:

```
{
    "schemaVersion": "2.2",
    "description": "An example document.",
    "parameters": {
        "Message": {
            "type": "String",
            "description": "Message to be printed",
            "default": "Hello",
            "interpolationType" : "ENV_VAR",
            "allowedPattern": "^[^"]*$"
```

```
        }
    },
    "mainSteps": [{
        "action": "aws:runShellScript",
        "name": "printMessage",
        "precondition" : {
           "StringEquals" : ["platformType", "Linux"]
        },
        "inputs": {
            "runCommand": [
              "echo {{Message}}"
            ]
        }
    }
}
```

> **Note**
>
> `allowedPattern` no es técnicamente obligatorio en los documentos SSM que no utilizan doble llave: `{{ }}`

En el caso de los parámetros que usa con frecuencia, le recomendamos que los almacene en Parameter Store, una herramienta de AWS Systems Manager. A continuación, puede definir parámetros en el documento que hagan referencia a los parámetros de Parameter Store como su valor predeterminado. Para hacer referencia a un parámetro de Parameter Store, utilice la sintaxis siguiente.

```
{{ssm:parameter-name}}
```

Puede utilizar un parámetro que haga referencia a un parámetro de Parameter Store igual que haría con cualquier otro parámetro de documentos. En el siguiente ejemplo, el valor predeterminado del parámetro `commands` es el parámetro `myShellCommands` de Parameter Store. Al especificar el parámetro `commands` como una cadena `runCommand`, el documento ejecuta los comandos almacenados en el parámetro `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 }}"],
            interpolationType : 'ENV_VAR'
            allowedPattern: '^[^"]*$'

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 }}"],
        "interpolationType" : "ENV_VAR"
      }
    },
    "mainSteps": [
      {
        "action": "aws:runShellScript",
        "name": "runShellScriptDefaultParams",
        "inputs": {
            "runCommand": [
              "{{ commands }}"
          ]
        }
      }
    ]
  }
```

> **Note**
>
> Puede hacer referencia a los parámetros de `String` y `StringList` de Parameter Store en la sección `parameters` del documento. No puede hacer referencia a los parámetros `SecureString` de Parameter Store.

Para obtener más información acerca de Parameter Store, consulte AWS Systems Manager Parameter Store.

Tipo: estructura

La estructura `parameters` acepta los siguientes campos y valores:

- `type`: (Obligatorio) Entre los valores permitidos se incluyen los siguientes: `String`, `StringList`, `Integer` `Boolean`, `MapList` y `StringMap`. Para ver ejemplos de cada tipo, consulte Ejemplos del parámetro `type` en documentos de SSM en la siguiente sección.

  > **Note**
  >
  > Los documentos de tipo comando solo admiten los tipos de parámetros `String` y `StringList`.

- `description`: (Opcional) Una descripción del parámetro.
- `default`: (Opcional) El valor predeterminado del parámetro o una referencia a un parámetro en Parameter Store.
- `allowedValues`: (Opcional) Una matriz de valores permitidos para el parámetro. La definición de valores permitidos para el parámetro valida la entrada del usuario. Si un usuario introduce un valor que no está permitido, la ejecución no se iniciará.

  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`: (Opcional) Una expresión regular que valida si la entrada del usuario coincide con el patrón definido para el parámetro. Si la entrada del usuario no coincide con el patrón permitido, la ejecución no se iniciará.

> **Note**
>
> Systems Manager realiza dos validaciones para `allowedPattern`. La primera validación se lleva a cabo utilizando la Biblioteca regex de Java en el nivel de API cuando usa un documento. La segunda validación se lleva a cabo en SSM Agent mediante el uso de la Biblioteca regex antes de procesar el documento.

YAML

```
InstanceId:
  type: String
  description: "(Required) The instance ID to target."
  allowedPattern: "^i-(?:[a-f0-9]{8}|[a-f0-9]{17})$"
  default: ''
```

JSON

```
"InstanceId": {
  "type": "String",
  "description": "(Required) The instance ID to target.",
  "allowedPattern": "^i-(?:[a-f0-9]{8}|[a-f0-9]{17})$",
  "default": ""
```

```
}
```

- `displayType`: (Opcional) Se utiliza para mostrar `textfield` o `textarea` en la AWS Management Console. `textfield` es un cuadro de texto de línea única. `textarea` es un área de texto multilínea.
- `minItems`: (Opcional) El número mínimo de elementos permitidos.
- `maxItems`: (Opcional) El número máximo de elementos permitidos.
- `minChars`: (Opcional) El número mínimo de caracteres del parámetro permitidos.
- `maxChars`: (Opcional) El número máximo de caracteres del parámetro permitidos.
- `interpolationType`: (Opcional) Define cómo se procesan los valores de los parámetros antes de ejecutar el comando. Cuando se establece en `ENV_VAR`, el valor del parámetro está disponible como una variable de entorno denominada `SSM_`*`parameter-name`*. Esta característica ayuda a evitar la inyección de comandos al tratar los valores de los parámetros como cadenas literales.

  Tipo: cadena

  Valores válidos: `ENV_VAR`

Obligatorio: no

variables

(Solo en la versión 0.3 del esquema) Valores a los que puede hacer referencia o actualizar a lo largo de los pasos de un manual de procedimientos de automatización. Las variables son similares a los parámetros, pero difieren de forma muy importante. Los valores de los parámetros son estáticos en el contexto de un manual de procedimientos, pero los valores de las variables se pueden cambiar en el contexto del manual de procedimientos. Al actualizar el valor de una variable, el tipo de datos debe coincidir con el tipo de datos definido. Para obtener información sobre la actualización de los valores de las variables en una automatización, consulte `aws:updateVariable` — Actualiza el valor de una variable del manual de procedimientos.

Tipo: Boolean | Integer | MapList | String | StringList | StringMap

Obligatorio: no

YAML

```
variables:
    payload:
```

```
        type: StringMap
        default: "{}"
```

JSON

```
{
    "variables": [
        "payload": {
            "type": "StringMap",
            "default": "{}"
        }
    ]
}
```

runtimeConfig

(Versión de esquema 1.2 solamente) La configuración de la instancia aplicada por uno o varios complementos de Systems Manager. No se garantiza que los complementos se ejecuten en secuencia.

Tipo: diccionario<cadena,PluginConfiguration>

Obligatorio: no

mainSteps

(Solo versiones de esquema 0.3, 2.0 y 2.2) Un objeto que puede incluir varios pasos (complementos). Los complementos se definen en pasos. Los pasos se ejecutan en orden secuencial según se indica en el documento.

Tipo: diccionario<cadena,PluginConfiguration>

Obligatorio: sí

salidas

(Solo versión de esquema 0.3) Datos generados por la ejecución de este documento que puede utilizarse en otros procesos. Por ejemplo, si el documento crea una nueva AMI, puede especificar “CreateImage.ImageId” como valor de salida y, a continuación, utilizar este resultado para crear nuevas instancias en una ejecución de automatización posterior. Para obtener más información acerca de las salidas, consulte Uso de salidas de acción como entradas.

Tipo: diccionario<cadena,OutputConfiguration>

Obligatorio: no

files

(Solo versión de esquema 0.3) Los archivos de script (y sus sumas de comprobación) están asociados al documento y se ejecutan durante una ejecución de automatización. Solo se aplica a los documentos que incluyen la acción `aws:executeScript` y para los que se han especificado datos adjuntos en uno o más pasos.

Para obtener más información sobre los tiempos de ejecución compatibles con los manuales de procedimientos de Automatización, consulte `aws:executeScript`: ejecutar un script. Para obtener más información acerca de la inclusión de secuencias de comandos en documentos de Automation, consulte Uso de scripts en manuales de procedimientos y Experiencia de diseño visual para manuales de procedimientos de automatización.

Cuando se crea un manual de procedimientos de automatización con datos adjuntos, también se deben especificar los archivos de los datos adjuntos mediante la opción `--attachments` (para AWS CLI) o `Attachments` (para API y SDK). Puede especificar la ubicación del archivo para los archivos y los documentos de SSM almacenados en buckets de Amazon Simple Storage Service (Amazon S3). Para obtener más información, consulte Attachments en la referencia de la API de AWS Systems Manager.

YAML

```
---
files:
  launch.py:
    checksums:
      sha256: 18871b1311b295c43d0f...
[truncated]...772da97b67e99d84d342ef4aEXAMPLE
```

JSON

```
"files": {
    "launch.py": {
        "checksums": {
            "sha256": "18871b1311b295c43d0f...
[truncated]...772da97b67e99d84d342ef4aEXAMPLE"
        }
    }
}
```

Tipo: diccionario<cadena,FilesConfiguration>

Obligatorio: no

Ejemplos del parámetro **`type`** en documentos de SSM

Los tipos de parámetros de los documentos de SSM son estáticos. Esto significa que el tipo de parámetro no se puede cambiar después de definirlo. Cuando se utilizan parámetros con complementos de documentos de SSM, el tipo de parámetro no se puede cambiar dinámicamente dentro de la entrada de un complemento. Por ejemplo, no se puede hacer referencia a un parámetro `Integer` dentro de la entrada `runCommand` del complemento `aws:runShellScript` porque esta entrada acepta una cadena o lista de cadenas. Para utilizar un parámetro para una entrada de un complemento, el tipo de parámetro debe coincidir con el tipo aceptado. Por ejemplo, debe especificar un parámetro de tipo `Boolean` para la entrada `allowDowngrade` del complemento `aws:updateSsmAgent`. Si el tipo de parámetro no coincide con el tipo de entrada de un complemento, el documento de SSM no se valida y el sistema no crea el documento. Esto también es cierto cuando se utilizan parámetros posteriores dentro de entradas para otros complementos o acciones de AWS Systems Manager automatización. Por ejemplo, no puede hacer referencia a un parámetro `StringList` dentro de la entrada `documentParameters` del complemento `aws:runDocument`. La entrada `documentParameters` acepta un mapa de cadenas incluso si el tipo de parámetro posterior de documento de SSM es un parámetro `StringList` y coincide con el parámetro al que está haciendo referencia.

Cuando se utilizan parámetros con acciones de Automation, los tipos de parámetros no se validan cuando se crea el documento de SSM en la mayoría de los casos. Solo cuando se utiliza la acción `aws:runCommand` se validan los tipos de parámetros cuando crea el documento de SSM. En todos los demás casos, la validación de parámetros se produce durante la ejecución de la automatización cuando se verifica la entrada de una acción antes de ejecutar la acción. Por ejemplo, si el parámetro de entrada es `String` y hace referencia a él como el valor de la entrada `MaxInstanceCount` de la acción `aws:runInstances`, se crea el documento de SSM. Sin embargo, al ejecutar el documento, la automatización produce un error al validar la acción `aws:runInstances` porque la entrada `MaxInstanceCount` requiere un valor `Integer`.

A continuación, se incluyen ejemplos de cada `type` de parámetro.

Cadena

Una secuencia de cero o más caracteres Unicode escritos entre comillas. Por ejemplo, "i-1234567890abcdef0". Utilice barras diagonales invertidas para aplicar escape.

Los parámetros de cadena pueden incluir un campo opcional `interpolationType` con el valor `ENV_VAR` para permitir la interpolación de variables de entorno y mejorar la seguridad.

YAML

```
---
InstanceId:
  type: String
  description: "(Optional) The target EC2 instance ID."
  interpolationType: ENV_VAR
```

JSON

```
"InstanceId":{
  "type":"String",
  "description":"(Optional) The target EC2 instance ID.",
  "interpolationType": "ENV_VAR"
}
```

StringList

Una lista de elementos de cadena separados por comas. Por ejemplo, ["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"
}
```

Booleano

Admite solo `true` o `false`. No admite “true” o 0.

YAML

```
---
canRun:
  type: Boolean
  description: ''
  default: true
```

JSON

```
"canRun": {
  "type": "Boolean",
  "description": "",
  "default": true
}
```

Entero

Números enteros. No acepta números decimales, por ejemplo 3,14159 ni números escritos entre comillas, por ejemplo "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

Un mapeo de claves a valores. Las claves y los valores deben ser cadenas. Por ejemplo, {"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

Una lista de objetos StringMap.

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
}
```

Visualización del contenido del documento de Command de SSM

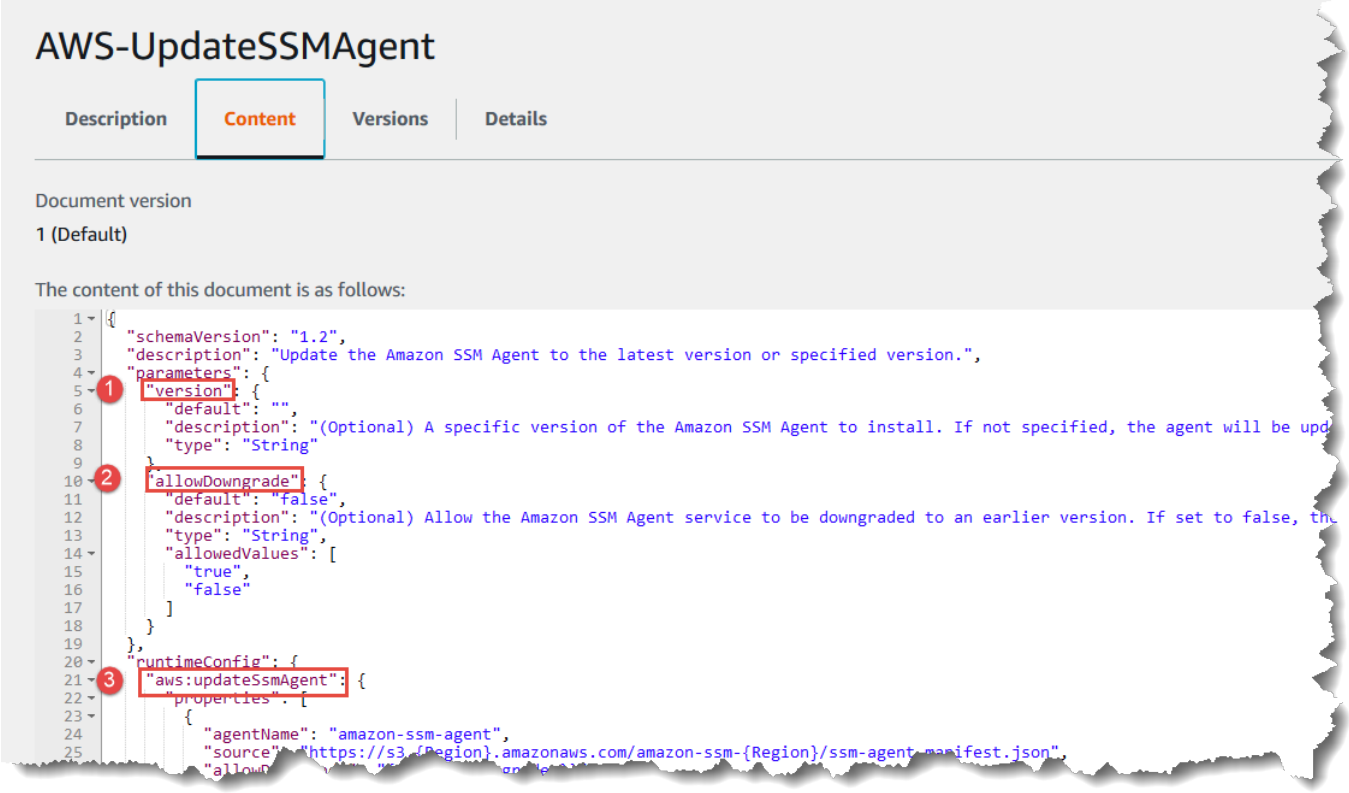
Para tener una vista previa de los parámetros necesarios y opcionales para un documento de AWS Systems Manager (SSM) Command, además de las acciones que ejecuta, puede ver el contenido del documento en la consola de Systems Manager.

Para ver el contenido del documento de SSM Command

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. En el cuadro de búsqueda, seleccione Tipo de documento y, a continuación, seleccione Comando.

4. Elija el nombre de un documento y, a continuación, la pestaña Contenido.
5. En el campo de contenido, revise los parámetros disponibles y los pasos de acción para el documento.

   Por ejemplo, en la siguiente imagen se muestra que (1) `version` y 2 `allowDowngrade` son parámetros opcionales para el documento `AWS-UpdateSSMAgent` y que la primera acción ejecutada por el documento es (3) `aws:updateSsmAgent`.



Referencia de complementos del documento de comandos

Esta referencia describe los complementos que puede especificar en un documento de tipo de comandos de AWS Systems Manager (SSM). Estos complementos no se pueden utilizar en manuales de procedimientos de Automation, que utilizan acciones de automatización. Para obtener información acerca de las acciones de AWS Systems Manager Automation, consulte Referencia de acciones de Automatización de Systems Manager.

Systems Manager determina las acciones que hay que realizar en una instancia administrada; para ello, lee el contenido de un documento de SSM. Cada documento incluye una sección de ejecución de código. En función de la versión de esquema de su documento, esta sección de ejecución de código puede incluir uno o más complementos o pasos. A efectos de este tema de ayuda, los

complementos y los pasos se denominan complementos. En esta sección, se incluye información sobre cada uno de los complementos de Systems Manager. Para obtener más información sobre los documentos, incluida la información sobre la creación de documentos y las diferencias entre las versiones de esquema, consulte Documentos de AWS Systems Manager.

En el caso de los complementos que aceptan parámetros de cadena, como `aws:runShellScript` y `aws:runPowerShellScript`, el parámetro `interpolationType` se puede utilizar para mejorar la seguridad al tratar las entradas de los parámetros como cadenas literales en lugar de comandos potencialmente ejecutables. Por ejemplo:

```
{
    "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 }}"],
        "interpolationType" : "ENV_VAR"
      }
    },
    //truncated
 }
```

> **Note**
>
> Algunos de los complementos descritos aquí se ejecutan solo en instancias de Windows Server o Linux. Se indican las dependencias de la plataforma para cada complemento. Los siguientes complementos de documentos se admiten en instancias Amazon Elastic Compute Cloud (Amazon EC2) para macOS:
>
> - `aws:refreshAssociation`
> - `aws:runShellScript`
> - `aws:runPowerShellScript`
> - `aws:softwareInventory`
> - `aws:updateSsmAgent`

Contenido

- Entradas compartidas
- aws:applications
- aws:cloudWatch
- aws:configureDocker
- aws:configurePackage
- aws:domainJoin
- aws:downloadContent
- aws:psModule
- aws:refreshAssociation
- aws:runDockerAction
- aws:runDocument
- aws:runPowerShellScript
- aws:runShellScript
- aws:softwareInventory
- aws:updateAgent
- aws:updateSsmAgent

Entradas compartidas

Solo con la versión 3.0.502 y las versiones posteriores de SSM Agent, todos los complementos pueden utilizar las siguientes entradas:

finallyStep

El último paso que desea que ejecute el documento. Si esta entrada se define para un paso, tendrá prioridad sobre un valor `exit` especificado en las entradas on`Failure` o on`Success`. Para que un paso con esta entrada se ejecute como se espera, el paso debe ser el último definido en los `mainSteps` del documento.

Tipo: Booleano

Valores válidos: `true` | `false`

Obligatorio: no

onFailure

Si especifica esta entrada para un complemento con el valor `exit` y el paso falla, el estado del paso refleja el error, y el documento no ejecuta los pasos restantes a menos que se haya definido un `finallyStep`. Si especifica esta entrada para un complemento con el valor `successAndExit` y el paso falla, el estado del paso aparece como realizado correctamente y el documento no ejecuta los pasos restantes a menos que se haya definido un `finallyStep`.

Tipo: cadena

Valores válidos: `exit` | `successAndExit`

Obligatorio: no

onSuccess

Si especifica esta entrada para un complemento y el paso se ejecuta correctamente, el documento no ejecutará los pasos restantes a menos que se haya definido un `finallyStep`.

Tipo: cadena

Valores válidos: `exit`

Obligatorio: no

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`

Instalar, reparar o desinstalar aplicaciones en una instancia de EC2. Este complemento solo se ejecuta en los sistemas operativos Windows Server.

Sintaxis

Esquema 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 }}"
      }
    }
  ]
}
```

Esquema 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 }}"
            }
         ]
      }
   }
}
```

Propiedades

acción

La acción que hay que realizar.

Tipo: enumeración

Valores válidos: `Install` | `Repair` | `Uninstall`

Obligatorio: sí

parameters

Los parámetros para el instalador.

Tipo: cadena

Requerido: no

origen

La URL del archivo `.msi` para la aplicación.

Tipo: cadena

Obligatorio: sí

sourceHash

Hash SHA256 del archivo `.msi`.

Tipo: cadena

Requerido: no

`aws:cloudWatch`

Exportar datos desde Windows Server a Amazon CloudWatch o los Registros de Amazon CloudWatch y monitorear los datos mediante las métricas de CloudWatch. Este complemento solo se ejecuta en los sistemas operativos Windows Server. Para obtener más información sobre cómo configurar la integración de CloudWatch con Amazon Elastic Compute Cloud (Amazon EC2), consulte Recopilación de métricas, registros y seguimientos con el agente de CloudWatch en la Guía del usuario de Amazon CloudWatch.

> **Important**
>
> El agente unificado de CloudWatch ha sustituido a SSM Agent como herramienta para enviar datos de registro a los Registros de Amazon CloudWatch. El complemento aws:cloudWatch del SSM Agent no es compatible. Recomendamos utilizar solo el agente de CloudWatch unificado para sus procesos de recopilación de registros. Para obtener más información, consulte los temas siguientes:
>
> - Envío de registros de nodos a los Registros de CloudWatch (agente de CloudWatch) unificado
> - Migrar la recopilación de registros del nodo de Windows Server al agente de CloudWatch
> - Recopilación de métricas, registros y seguimientos con el agente de CloudWatch en la Guía del usuario de Amazon CloudWatch.

Puede exportar y monitorizar los siguientes tipos de datos:

ApplicationEventLog

Envía datos de registro de eventos de aplicación a los Registros de CloudWatch.

CustomLogs

Envía cualquier archivo de registro basado en texto a los Registros de Amazon CloudWatch. El complemento de CloudWatch crea una huella digital para archivos de registro. El sistema asocia, a continuación, un desplazamiento de datos con cada huella digital. El complemento carga archivos cuando hay cambios, registra el desplazamiento y asocia el desplazamiento con una huella digital. Este método se utiliza para evitar una situación en la que un usuario activa el complemento, asocia el servicio con un directorio que contiene un gran número de archivos y el sistema carga todos los archivos.

> **Warning**
>
> Tenga en cuenta que si la aplicación trunca o intenta limpiar registros durante el sondeo, cualquier registro especificado para `LogDirectoryPath` pueden perder entradas. Si, por ejemplo, desea limitar el tamaño de los archivos de registro, cree un nuevo archivo de registro cuando se alcance el límite y, a continuación, continúe escribiendo datos en el nuevo archivo.

ETW

Envía datos de Seguimiento de eventos para Windows (ETW) a los Registros de CloudWatch.

IIS

Envía datos de registro de IIS a los Registros de CloudWatch.

PerformanceCounter

Envía contadores de rendimiento de Windows a CloudWatch. Puede seleccionar diferentes categorías para cargar a CloudWatch como métricas. Para cada contador de rendimiento que desee cargar, cree una sección PerformanceCounter con un ID único (por ejemplo, "PerformanceCounter2", "PerformanceCounter3", etc.) y configure sus propiedades.

> **Note**
>
> Si se detiene el SSM Agent de AWS Systems Manager o el complemento de CloudWatch, los datos del contador de rendimiento no se registran en CloudWatch. Este comportamiento es distinto de los registros personalizados o de los registros de eventos de Windows. Estos registros mantienen los datos de contador de rendimiento y los cargan a CloudWatch después de que el SSM Agent o el complemento de CloudWatch estén disponibles.

SecurityEventLog

Envía datos de registro de eventos de seguridad a los Registros de CloudWatch.

SystemEventLog

Envía datos de registro de eventos de sistema a los Registros de CloudWatch.

Puede definir los siguientes destinos para los datos:

CloudWatch

El destino al que se envían los datos de las métricas del contador de rendimiento. Puede añadir más secciones con ID únicos (por ejemplo, "CloudWatch2", "CloudWatch3", etc.) y especificar una región diferente para cada ID nuevo para enviar los mismos datos a diferentes ubicaciones.

CloudWatchLogs

El destino al que se envían los datos de registro. Puede añadir más secciones con ID únicos (por ejemplo, "CloudWatchLogs2", "CloudWatchLogs3", etc.) y especificar una región diferente para cada ID nuevo para enviar los mismos datos a diferentes ubicaciones.

Sintaxis

```
"runtimeConfig":{
        "aws:cloudWatch":{
            "settings":{
                "startType":"{{ status }}"
            },
            "properties":"{{ properties }}"
        }
    }
```

Configuración y propiedades

AccessKey

El ID de la clave de acceso de . Esta propiedad es necesaria a menos que lanzara su instancia utilizando un rol de IAM. Esta propiedad no se puede utilizar con SSM.

Tipo: cadena

Requerido: no

CategoryName

La categoría de contador de rendimiento de Performance Monitor.

Tipo: cadena

Obligatorio: sí

CounterName

El nombre del contador de rendimiento de Performance Monitor.

Tipo: cadena

Obligatorio: sí

CultureName

La configuración regional en la que se registra la marca temporal. Si CultureName está en blanco, se usa de forma predeterminada la misma configuración regional que utiliza su instancia de Windows Server.

Tipo: cadena

Valores válidos: para obtener una lista de los valores admitidos, consulte el tema National Language Support (NLS) API Reference [Referencia de la API de compatibilidad con el idioma nacional (NLS)] en el sitio web de Microsoft. No se admiten los valores div, div-MV, hu y hu-HU.

Obligatorio: no

DimensionName

Una dimensión para la métrica de Amazon CloudWatch. Si especifica `DimensionName`, también debe especificar `DimensionValue`. Estos parámetros ofrecen otra vista al enumerar las métricas. Puede utilizar la misma dimensión para varias métricas, lo que le permite ver todas las métricas que pertenezcan a una dimensión concreta.

Tipo: cadena

Requerido: no

DimensionValue

Un valor de dimensión para la métrica de Amazon CloudWatch.

Tipo: cadena

Requerido: no

Codificación

La codificación del archivo que se va a utilizar (por ejemplo, UTF-8). Utilice el nombre de codificación, no el nombre de visualización.

Tipo: cadena

Valores válidos: para obtener una lista de los valores admitidos, consulte Clase Encoding en la biblioteca de Microsoft Learn.

Obligatorio: sí

Filtro

El prefijo de los nombres de registro. Deje en blanco este parámetro para monitorear todos los archivos.

Tipo: cadena

Valores válidos: para obtener una lista de los valores admitidos, consulte el tema Propiedad FileSystemWatcherFilter en la biblioteca de MSDN.

Obligatorio: no

Flujos

Cada tipo de datos que se va a cargar, junto con el destino de los datos (CloudWatch o Registros de CloudWatch). Por ejemplo, para enviar un contador de rendimiento definido bajo `"Id": "PerformanceCounter"` al destino de CloudWatch definido bajo `"Id": "CloudWatch"`, escriba “PerformanceCounter,CloudWatch”. Del mismo modo, para enviar el registro personalizado, el registro de ETW y el registro del sistema al destino de los Registros de CloudWatch definido bajo `"Id": "ETW"`, escriba “(ETW),CloudWatchLogs”. Además, puede enviar el mismo contador de rendimiento o archivo de registro a más de un destino. Por ejemplo, para enviar el registro de la aplicación a dos destinos diferentes definidos bajo `"Id": "CloudWatchLogs"` y `"Id": "CloudWatchLogs2"`, escriba "ApplicationEventLog, (CloudWatchLogs, CloudWatchLogs2)".

Tipo: cadena

Valores válidos (origen): `ApplicationEventLog` | `CustomLogs` | `ETW` | `PerformanceCounter` | `SystemEventLog` | `SecurityEventLog`

Los valores válidos (destino): `CloudWatch` | `CloudWatchLogs` | `CloudWatch`*n* | `CloudWatchLogs`*n*

Obligatorio: sí

FullName

El nombre completo del componente.

Tipo: cadena

Obligatorio: sí

Id

Identifica el origen o el destino de los datos. Este identificador deberá ser único dentro del archivo de configuración.

Tipo: cadena

Obligatorio: sí

InstanceName

El nombre de la instancia del contador de rendimiento. No utilice un asterisco (*) para indicar todas las instancias, porque cada componente de contador de rendimiento solo admite una métrica. Puede, sin embargo utilizar _Total.

Tipo: cadena

Obligatorio: sí

Niveles

Tipos de mensajes para enviar a Amazon CloudWatch.

Tipo: cadena

Valores válidos:

- 1: solo los mensajes de error cargados.
- 2: solo los mensajes de advertencia cargados.
- 4: solo los mensajes de información cargados.

Puede agregar valores juntos para incluir más de un tipo de mensaje. Por ejemplo, 3 significa que se incluyen mensajes de error (1) y mensajes de advertencia (2). El valor 7 significa que se incluyen mensajes de error (1), mensajes de advertencia (2) y mensajes informativos (4).

Obligatorio: sí

> **Note**
>
> Los registros de seguridad de Windows deben establecer los niveles en 7.

LineCount

El número de líneas del encabezado para identificar el archivo de registro. Por ejemplo, los archivos de registros de IIS tienen encabezados prácticamente idénticos. Puede especificar 3, que leería las tres primeras líneas del encabezado del archivo de registro para identificarlo. En los archivos de registro de IIS, la tercera línea es la marca de fecha y hora, que es distinta entre los archivos de registro.

Tipo: entero

Obligatorio: no

LogDirectoryPath

Para CustomLogs, escriba la ruta donde se almacenan registros en la instancia de EC2. Para los registros de IIS, la carpeta en la que se almacenan los registros de IIS para un sitio individual (por ejemplo, C:\\inetpub\\logs\\LogFiles\\W3SVC***n***). Para los registros de IIS, solo se admite el formato de registro W3C. No se admiten los formatos IIS, NCSA y personalizados.

Tipo: cadena

Obligatorio: sí

LogGroup

El nombre de su grupo de registro. Este nombre se muestra en la pantalla Grupos de registros en la consola de CloudWatch.

Tipo: cadena

Obligatorio: sí

LogName

El nombre del archivo de registro.

1. Para encontrar el nombre del registro, en Visor de eventos, en el panel de navegación, seleccione Registros de aplicaciones y servicios.
2. En la lista de registros, haga clic con el botón derecho en el registro que desea cargar (por ejemplo, Microsoft > Windows > Copia de seguridad > Operational) y, a continuación, seleccione Create Custom View.
3. En el cuadro de diálogo Create Custom View, seleccione la pestaña XML. LogName se encuentra en la etiqueta <Select Path=> (por ejemplo, `Microsoft-Windows-Backup`). Copie este texto en el parámetro LogName.

Tipo: cadena

Valores válidos: `Application` | `Security` | `System` | `Microsoft-Windows-WinINet/Analytic`

Obligatorio: sí

LogStream

El flujo de registros de destino. Si utiliza {instance_id}, se utiliza el ID de la instancia predeterminado de esta instancia como el nombre de registro de destino.

Tipo: cadena

Valores válidos: `{instance_id}` | `{hostname}` | `{ip_address}`*`<log_stream_name>`*

Si ingresa un nombre de flujo de registro que no existe, los Registros de CloudWatch lo crea automáticamente. Puede utilizar una cadena literal o variables predefinidas ({instance_id}, {hostname}, {ip_address}), o una combinación de las tres para definir el nombre del flujo de registro.

El nombre del flujo de registro especificado en este parámetro aparece en la pantalla Log Groups > Streams for ***`<YourLogStream>`*** en la consola de CloudWatch.

Obligatorio: sí

MetricName

La métrica de CloudWatch bajo la que desea que se incluyan los datos de rendimiento.

> **Note**
>
> No utilice caracteres especiales en el nombre. Si lo hace, es posible que la métrica y las alarmas asociadas no funcionen.

Tipo: cadena

Obligatorio: sí

NameSpace

El espacio de nombres de métrica en el que desea que se escriban los datos de contadores de rendimiento.

Tipo: cadena

Obligatorio: sí

PollInterval

¿Cuántos segundos deben transcurrir antes de que se carguen nuevos datos de registro y de contador de rendimiento?

Tipo: entero

Valores válidos: establecer este en 5 o más segundos. Se recomiendan quince segundos (00:00:15).

Obligatorio: sí

Región

La Región de AWS a la que desea enviar los datos de registro. Aunque puede enviar contadores de rendimiento a una región distinta de la que envía los datos de registro, le recomendamos que ajuste este parámetro a la misma región en la que se ejecuta la instancia.

Tipo: cadena

Valores válidos: ID de regiones de las Regiones de AWS compatibles con Systems Manager y los Registros de CloudWatch, como `us-east-2`, `eu-west-1`, y `ap-southeast-1`. Para las listas de Regiones de AWS compatibles con cada servicio, consulte Puntos de conexión de Registros de Amazon CloudWatch y Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

Obligatorio: sí

SecretKey

La clave de acceso secreta de . Esta propiedad es necesaria a menos que lanzara su instancia utilizando un rol de IAM.

Tipo: cadena

Requerido: no

startType

Active o desactive CloudWatch en la instancia.

Tipo: cadena

Valores válidos: `Enabled` | `Disabled`

Obligatorio: sí

TimestampFormat

El formato de marca temporal que desea utilizar. Para obtener una lista de los valores admitidos, consulte el tema Custom Date and Time Format Strings en la biblioteca de MSDN.

Tipo: cadena

Obligatorio: sí

TimeZoneKind

Proporciona información sobre la zona horaria cuando no se incluye ninguna información sobre la zona horaria en la marca temporal del registro. Si este parámetro se deja en blanco y su marca temporal no incluye información sobre la zona horaria, los Registros de CloudWatch utilizan de forma predeterminada la zona horaria local. Se hace caso omiso a este parámetro si la marca temporal ya contiene información sobre la zona horaria.

Tipo: cadena

Valores válidos: `Local` | `UTC`

Obligatorio: no

Unidad

La unidad de medida adecuada para la métrica.

Tipo: cadena

Valores válidos: segundos | microsegundos | milisegundos | bytes | kilobytes | megabytes | gigabytes | terabytes | bits | kilobits | megabits | gigabits | terabits | porcentaje | recuento | bytes/segundo | kilobytes/segundo | megabytes/segundo | gigabytes/segundo | terabytes/segundo | bits/segundo | kilobits/segundo | megabits/segundo | gigabits/segundo | terabits/segundo | recuento/segundo o ninguno

Obligatorio: sí

`aws:configureDocker`

(Versión de esquema 2.0 o posterior) Configure una instancia para trabajar con contenedores y Docker. Este complemento es compatible con la mayoría de las variantes y los sistemas operativos Linux y Windows Server.

Sintaxis

Esquema 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 }}"
      }
    }
  ]
}
```

Entradas

acción

El tipo de acción que se va a realizar.

Tipo: enumeración

Valores válidos: `Install` | `Uninstall`

Obligatorio: sí

`aws:configurePackage`

(Versión de esquema 2.0 o posterior) Instale o desinstale un paquete Distributor de AWS Systems Manager. Puede instalar la versión más reciente, la versión predeterminada o una versión del paquete que especifique. Los paquetes proporcionados por AWS también son compatibles. Este complemento se ejecuta en los sistemas operativos Windows Server y Linux, pero no todos los paquetes disponibles se admiten en sistemas operativos Linux.

Los paquetes de AWS disponibles para Windows Server, incluyen lo siguiente: `AWSPVDriver`, `AWSNVMe`, `AwsEnaNetworkDriver`, `AwsVssComponents`, `AmazonCloudWatchAgent`, `CodeDeployAgent`, y `AWSSupport-EC2Rescue.`

Los paquetes AWS disponibles para los sistemas operativos Linux incluyen los siguientes: `AmazonCloudWatchAgent`, `CodeDeployAgent`, y `AWSSupport-EC2Rescue`.

Sintaxis

Esquema 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\"}"
         }
      }
   ]
}
```

Entradas

nombre

El nombre del paquete de AWS que se va a instalar o desinstalar. Los paquetes disponible incluyen lo siguiente: `AWSPVDriver`, `AwsEnaNetworkDriver`, `AwsVssComponents`, y `AmazonCloudWatchAgent`.

Tipo: cadena

Obligatorio: sí

acción

Instalar o desinstalar un paquete.

Tipo: enumeración

Valores válidos: `Install` | `Uninstall`

Obligatorio: sí

installationType

El tipo de instalación que se va a realizar. Si especifica `Uninstall and reinstall`, el paquete se desinstala por completo y, a continuación, se vuelve a instalar. La aplicación no estará disponible hasta que se complete la reinstalación. Si especifica `In-place update`, solo se agregan archivos nuevos o modificados a la instalación existente según las instrucciones que proporcione en un script de actualización. La aplicación permanece disponible durante todo el proceso de actualización. La opción `In-place update` no es compatible con paquetes publicados de AWS. `Uninstall and reinstall` es el valor predeterminado.

Tipo: enumeración

Valores válidos: `Uninstall and reinstall` | `In-place update`

Obligatorio: no

additionalArguments

Una cadena JSON con los parámetros adicionales que se deben proporcionar a los scripts de instalación, desinstalación o actualización. Cada parámetro debe tener el prefijo `SSM_`. Puede hacer referencia a un parámetro Parameter Store en sus argumentos adicionales mediante el uso de la convención `{{ssm:`*`parameter-name`*`}}`. Para utilizar el parámetro adicional en el script de instalación, desinstalación o actualización, debe hacer referencia al parámetro como una variable de entorno utilizando la sintaxis adecuada para el sistema operativo. Por ejemplo, en PowerShell, se hace referencia al argumento `SSM_arg` como `$Env:SSM_arg`. No hay límite en el número de argumentos que define, pero la entrada de argumento adicional tiene un límite de 4096 caracteres. Este límite incluye todas las claves y los valores que defina.

Tipo: StringMap

Obligatorio: no

versión

Una versión específica del paquete que se va a instalar o desinstalar. Si se va a instalar, el sistema instala la versión más reciente publicada, de forma predeterminada. Si se va a desinstalar, el sistema desinstala la versión instalada, de forma predeterminada. Si no se encuentra ninguna versión instalada, se descarga la versión más reciente publicada y se ejecuta la acción de desintalación.

Tipo: cadena

Requerido: no

`aws:domainJoin`

Unir una instancia de EC2 a un dominio. Este complemento se ejecuta en los sistemas operativos Linux y Windows Server. Este complemento cambia el nombre de host de las instancias de Linux al formato EC2AMAZ-*XXXXXXX*. Para obtener más información acerca de cómo unir instancias de EC2, consulte Cómo unir una instancia de EC2 al Directorio de Microsoft AD administrado de AWS en la Guía de administración de AWS Directory Service.

Sintaxis

Esquema 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
  hostname:
    description: "(Optional) The hostname you want to assign to the node."
    type: String
mainSteps:
- action: aws:domainJoin
  name: domainJoin
  inputs:
    directoryId: "{{ directoryId }}"
    directoryName: "{{ directoryName }}"
    directoryOU: "{{ directoryOU }}"
    dnsIpAddresses: "{{ dnsIpAddresses }}"
    hostname: "{{ hostname }}"
```

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"
    },
    "hostname": {
        "description": "(Optional) The hostname you want to assign to the node.",
        "type": "String"
```

```
      }
  },
  "mainSteps": [
    {
      "action": "aws:domainJoin",
      "name": "domainJoin",
      "inputs": {
        "directoryId": "{{ directoryId }}",
        "directoryName": "{{ directoryName }}",
        "directoryOU":"{{ directoryOU }}",
        "dnsIpAddresses":"{{ dnsIpAddresses }}",
        "hostname":"{{ hostname }}"
      }
    }
  ]
}
```

Esquema 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 }}"
         }
      }
```

```
    }
}
```

Propiedades

directoryId

El ID del directorio.

Tipo: cadena

Obligatorio: sí

Ejemplo: "directoryId": "d-1234567890"

directoryName

El nombre del dominio.

Tipo: cadena

Obligatorio: sí

Ejemplo: "directoryName": "example.com"

directoryOU

La unidad organizativa.

Tipo: cadena

Requerido: no

Ejemplo: "directoryOU": "OU=test,DC=example,DC=com"

dnsIpAddresses

Las direcciones IP de los servidores DNS.

Tipo: StringList

Obligatorio: sí

Ejemplo: "dnsIpAddresses": ["198.51.100.1"," 198.51.100.2"]

hostname

El nombre de host que desea asignar al nodo. Si no se proporciona, no se cambiará el nombre de las instancias de Windows, mientras que las instancias de Linux utilizarán el patrón de nomenclatura predeterminado. Si se proporciona, las instancias de Windows usarán el valor exacto proporcionado, mientras que, para las instancias de Linux, este servirá como prefijo (a menos que `keepHostName` se establezca como “true”).

Tipo: cadena

Requerido: no

keepHostName

Determina si se cambia el nombre de host de las instancias de Linux cuando se unen al dominio. Este es un parámetro exclusivo de Linux. Los nombres de los hosts de Linux se cambiarán de forma predeterminada (sin entradas para `hostname`, `hostnameNumAppendDigits` y con `keepHostName` como “false”) siguiendo el patrón EC2AMAZ-XXXXXX. Cuando se establece como “true”, mantiene el nombre de host original e ignora las entradas para `hostname` y `hostnameNumAppendDigits`.

Tipo: Booleano

Obligatorio: no

hostnameNumAppendDigits

Define el número de dígitos numéricos aleatorios que se añadirán después del valor del nombre de host. Este es un parámetro exclusivo de Linux y se utiliza junto con el parámetro `hostname`. Se ignora si no se proporciona el `hostname`.

Tipo: cadena

Valores permitidos: de 1 a 5

Obligatorio: no

Ejemplos

Para ver ejemplos, consulte Join an Amazon EC2 Instance to your AWS Managed Microsoft AD en la Guía de administración de AWS Directory Service.

`aws:downloadContent`

(Versión de esquema 2.0 o posterior) Descargue documentos y scripts de SSM de ubicaciones remotas. Los repositorios GitHub Enterprise no son compatibles. Este complemento es compatible con los sistemas operativos Linux y Microsoft Windows Server.

Sintaxis

Esquema 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 }}"
      }
    }
  ]
}
```

Entradas

sourceType

La fuente de descarga. Systems Manager admite los siguientes tipos de fuente para la descarga de scripts y documentos de SSM: `GitHub`, `Git`, `HTTP`, `S3`, y `SSMDocument`.

Tipo: cadena

Obligatorio: sí

sourceInfo

La información necesaria para recuperar el contenido del origen necesario.

Tipo: StringMap

Obligatorio: sí

En sourceType **`GitHub,`** especifique lo siguiente:

- propietario: el propietario del repositorio.
- repositorio: el nombre del repositorio.
- ruta: la ruta del archivo o directorio que desea descargar.
- getOptions: opciones adicionales para recuperar contenido de una rama que no sea maestra o de una confirmación específica en el repositorio. getOptions se pueden omitir si está utilizando la última confirmación en la rama maestra. Si su repositorio se creó después del 1 de octubre

de 2020, la rama predeterminada podría llamarse principal en lugar de maestra. En este caso, deberá especificar valores para el parámetro getOptions.

Este parámetro utiliza el siguiente formato:

- branch:refs/heads/*branch_name*

  El valor predeterminado es `master`.

  Para especificar una ramificación no predeterminada, utilice el siguiente formato:

  branch:refs/heads/*branch_name*

- commitID:*commitID*

  El valor predeterminado es `head`.

  Para utilizar la versión del documento SSM en una confirmación que no sea la última, especifique el ID de confirmación completo. Por ejemplo:

  ```
  "getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",
  ```

- tokenInfo: el parámetro de Systems Manager (un parámetro SecureString) donde almacena su información del token de acceso GitHub, en el formato `{{ssm-secure:`*secure-string-token-name*`}}`.

  > **Note**
  >
  > Este campo `tokenInfo` es el único campo del complemento de documentos de SSM que admite un parámetro SecureString. Los parámetros SecureString no se admiten para ningún otro campo ni para ningún otro complemento de documentos de SSM.

```
{
    "owner":"TestUser",
    "repository":"GitHubTest",
    "path":"scripts/python/test-script",
    "getOptions":"branch:master",
    "tokenInfo":"{{ssm-secure:secure-string-token}}"
}
```

En sourceType **`Git`**, debe especificar lo siguiente:

- repositorio

  La URL del repositorio de Git al archivo o directorio que desea descargar.

  Tipo: cadena

Además, puede especificar los siguientes parámetros opcionales:

- getOptions

  Opciones adicionales para recuperar contenido de una rama que no sea maestra o de una confirmación específica en el repositorio. getOptions se pueden omitir si está utilizando la última confirmación en la rama maestra.

  Tipo: cadena

  Este parámetro utiliza el siguiente formato:

  - branch:refs/heads/*branch_name*

    El valor predeterminado es `master`.

    Solo se requiere `"branch"` si el documento SSM se almacena en una sucursal que no sea `master`. Por ejemplo:

    ```
    "getOptions": "branch:refs/heads/main"
    ```

  - commitID:*commitID*

    El valor predeterminado es `head`.

    Para utilizar la versión del documento SSM en una confirmación que no sea la última, especifique el ID de confirmación completo. Por ejemplo:

    ```
    "getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",
    ```

- privateSSHKey

  La clave SSH que se utilizará cuando se conecte al `repository` que especifique. Puede utilizar el siguiente formato para hacer referencia a un parámetro `SecureString` para el valor de su clave SSH: `{{ssm-secure:`*your-secure-string-parameter*`}}`.

  Tipo: cadena

- skipHostKeyChecking

  Determina el valor de la opción StrictHostKeyChecking cuando se conecta al `repository` que especifique. El valor predeterminado es `false`.

  Tipo: Booleano

- nombre de usuario

  El nombre de usuario que se utilizará cuando se conecte al `repository` que especifique mediante HTTP. Puede utilizar el siguiente formato para hacer referencia a un parámetro `SecureString` para el valor de su nombre de usuario: `{{ssm-secure:`*`your-secure-string-parameter`*`}}`.

  Tipo: cadena

- contraseña

  La contraseña que se utilizará cuando se conecte al `repository` que especifique mediante HTTP. Puede utilizar el siguiente formato para hacer referencia a un parámetro `SecureString` para el valor de su contraseña: `{{ssm-secure:`*`your-secure-string-parameter`*`}}`.

  Tipo: cadena

En sourceType **`HTTP`**, debe especificar lo siguiente:

- url

  La URL al archivo o directorio que desea descargar.

  Tipo: cadena

Además, puede especificar los siguientes parámetros opcionales:

- allowInsecureDownload

  Determina si una descarga se puede realizar a través de una conexión que no está cifrada con Capa de conexión segura (SSL) o Transport Layer Security (TLS). El valor predeterminado es `false`. No le recomendamos realizar descargas sin cifrado. Si decide hacerlo, deberá asumir todos los riesgos asociados. La seguridad es una responsabilidad compartida entre AWS y usted. Se describe como el modelo de responsabilidad compartida. Para obtener más información, consulte el Modelo de responsabilidad compartida.

Tipo: Booleano

- authMethod

  Determina si un nombre de usuario y una contraseña se utilizan para la autenticación cuando se conecta a la `url` que especifique. Si especifica `Basic` o `Digest`, debe proporcionar valores para los parámetros `username` y `password`. Para utilizar el método `Digest`, el SSM Agent versión 3.0.1181.0 o posterior debe estar instalada en la instancia. El método `Digest` admite el cifrado MD5 y SHA256.

  Tipo: cadena

  Valores válidos: `None` | `Basic` | `Digest`

- nombre de usuario

  El nombre de usuario que se utilizará cuando se conecte al `url` que especifique mediante autenticación `Basic`. Puede utilizar el siguiente formato para hacer referencia a un parámetro `SecureString` para el valor de su nombre de usuario: `{{ssm-secure:`*`your-secure-string-parameter`*`}}`.

  Tipo: cadena

- contraseña

  La contraseña que se utilizará cuando se conecte al `url` que especifique mediante autenticación `Basic`. Puede utilizar el siguiente formato para hacer referencia a un parámetro `SecureString` para el valor de su contraseña: `{{ssm-secure:`*`your-secure-string-parameter`*`}}`.

  Tipo: cadena

En sourceType **`S3`**, especifique lo siguiente:

- path: la URL al archivo o directorio que desea descargar de Amazon S3.

```
{
    "path": "https://s3.amazonaws.com/amzn-s3-demo-bucket/powershell/
helloPowershell.ps1"
}
```

Para sourceType **`SSMDocument`**, especifique uno de los siguientes:

- nombre: el nombre y la versión del documento en el siguiente formato: `name:version`. La versión es opcional.

```
{
    "name": "Example-RunPowerShellScript:3"
}
```

- name: el ARN del documento en el siguiente formato: `arn:aws:ssm:region:account_id:document/document_name`

```
{
   "name":"arn:aws:ssm:us-east-2:3344556677:document/MySharedDoc"
}
```

destinationPath

Una ruta local opcional en la instancia en la que se desea descargar el archivo. Si no se especifica una ruta, el contenido se descarga en una ruta relativa al ID de comando.

Tipo: cadena

Requerido: no

aws:psModule

Instalar módulos de PowerShell en una instancia Amazon EC2. Este complemento solo se ejecuta en los sistemas operativos Windows Server.

Sintaxis

Esquema 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 }}"
      }
    }
  ]
}
```

Esquema 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 }}"
            }
         ]
      }
   }
}
```

Propiedades

runCommand

El comando de PowerShell que se va a ejecutar tras la instalación del módulo.

Tipo: StringList

Obligatorio: no

origen

La dirección URL o ruta local en la instancia al archivo `.zip` de la aplicación.

Tipo: cadena

Obligatorio: sí

sourceHash

Hash SHA256 del archivo `.zip`.

Tipo: cadena

Requerido: no

timeoutSeconds

El tiempo en segundos para que un comando se complete antes de considerar que se ha producido un error.

Tipo: cadena

Requerido: no

workingDirectory

La ruta al directorio de trabajo en la instancia.

Tipo: cadena

Requerido: no

`aws:refreshAssociation`

(Versión de esquema 2.0 o posterior) Actualice (fuerce aplicación) una asociación bajo demanda. Esta acción cambiará el estado del sistema en función de lo que se define en la asociación seleccionada o todas las asociaciones vinculadas a los destinos. Este complemento se ejecuta en los sistemas operativos Linux y Microsoft Windows Server.

Sintaxis

Esquema 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 }}"
        ]
      }
    }
  ]
}
```

Entradas

associationIds

Lista de ID de asociación. Si está vacía, se aplican todas las asociaciones vinculadas al destino especificado.

Tipo: StringList

Obligatorio: no

`aws:runDockerAction`

(Versión de esquema 2.0 o posterior) Ejecute acciones de Docker en contenedores. Este complemento se ejecuta en los sistemas operativos Linux y Microsoft Windows Server.

Sintaxis

Esquema 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 }}"
    workingDirectory: "{{ workingDirectory }}"
    timeoutSeconds: "{{ timeoutSeconds }}"
```

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 }}",
            "workingDirectory": "{{ workingDirectory }}",
            "timeoutSeconds": "{{ timeoutSeconds }}"
         }
      }
   ]
}
```

Entradas

acción

El tipo de acción que se va a realizar.

Tipo: cadena

Obligatorio: sí

contenedor

El ID del contenedor de Docker.

Tipo: cadena

Requerido: no

imagen

El nombre de la imagen de Docker.

Tipo: cadena

Requerido: no

cmd

El comando del contenedor.

Tipo: cadena

Requerido: no

memoria

El límite de memoria del contenedor.

Tipo: cadena

Requerido: no

cpuShares

El contenedor que comparte la CPU (peso relativo).

Tipo: cadena

Requerido: no

volume

El volumen que el contenedor monta.

Tipo: StringList

Obligatorio: no

env

Las variables del entorno del contenedor.

Tipo: cadena

Requerido: no

usuario

El nombre de usuario de un contenedor.

Tipo: cadena

Requerido: no

publish

Los puertos publicados del contenedor.

Tipo: cadena

Requerido: no

workingDirectory

La ruta al directorio de trabajo en el nodo administrado.

Tipo: cadena

Requerido: no

timeoutSeconds

El tiempo en segundos para que un comando se complete antes de considerar que se ha producido un error.

Tipo: cadena

Requerido: no

`aws:runDocument`

(Versión de esquema 2.0 o posterior) Ejecuta documentos de SSM almacenados en Systems Manager o en un recurso compartido local. Puede utilizar este complemento con el complemento `aws:downloadContent` para descargar un documento de SSM a un recurso compartido local y, a continuación, ejecutarlo. Este complemento es compatible con los sistemas operativos Linux y Microsoft Windows Server. Este complemento no admite la ejecución del documento `AWS-UpdateSSMAgent` o cualquier documento que utilice el complemento `aws:updateSsmAgent`.

Sintaxis

Esquema 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 }}"
      }
    }
  ]
}
```

Entradas

documentType

El tipo de documento que se va a ejecutar. Puede ejecutar documentos locales (`LocalPath`) o documentos almacenados en Systems Manager (`SSMDocument`).

Tipo: cadena

Obligatorio: sí

documentPath

La ruta al documento. Si `documentType` es `LocalPath`, entonces especifique la ruta al documento en el recurso compartido local. Si `documentType` es `SSMDocument`, entonces especifique el nombre del documento.

Tipo: cadena

Requerido: no

documentParameters

Parámetros para el documento.

Tipo: StringMap

Obligatorio: no

`aws:runPowerShellScript`

Ejecute scripts de PowerShell o especifique la ruta a un script para su ejecución. Este complemento se ejecuta en los sistemas operativos Microsoft Windows Server y Linux.

Sintaxis

Esquema 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 }}"
        ]
      }
    }
  ]
}
```

Esquema 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 }}"
            }
         ]
      }
   }
}
```

Propiedades

runCommand

Especifique los comandos que se deben ejecutar o la ruta a un script existente en la instancia.

Tipo: StringList

Obligatorio: sí

timeoutSeconds

El tiempo en segundos para que un comando se complete antes de considerar que se ha producido un error. Cuando se alcanza el tiempo de espera, Systems Manager detiene la ejecución del comando.

Tipo: cadena

Requerido: no

workingDirectory

La ruta al directorio de trabajo en la instancia.

Tipo: cadena

Requerido: no

`aws:runShellScript`

Ejecute scripts de shell de Linux o especifique la ruta a un script para su ejecución. Este complemento solo se ejecuta en los sistemas operativos Linux.

Sintaxis

Esquema 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 }}"
        ]
      }
    }
  ]
}
```

Esquema 1.2

YAML

```
---
runtimeConfig:
  aws:runShellScript:
    properties:
    - runCommand: "{{ commands }}"
      workingDirectory: "{{ workingDirectory }}"
      timeoutSeconds: "{{ executionTimeout }}"
```

JSON

```
{
   "runtimeConfig":{
      "aws:runShellScript":{
         "properties":[
            {
               "runCommand":"{{ commands }}",
               "workingDirectory":"{{ workingDirectory }}",
               "timeoutSeconds":"{{ executionTimeout }}"
            }
         ]
      }
```

```
    }
}
```

Propiedades

runCommand

Especifique los comandos que se deben ejecutar o la ruta a un script existente en la instancia.

Tipo: StringList

Obligatorio: sí

timeoutSeconds

El tiempo en segundos para que un comando se complete antes de considerar que se ha producido un error. Cuando se alcanza el tiempo de espera, Systems Manager detiene la ejecución del comando.

Tipo: cadena

Requerido: no

workingDirectory

La ruta al directorio de trabajo en la instancia.

Tipo: cadena

Requerido: no

`aws:softwareInventory`

(Versión de esquema 2.0 o posterior) Recopilar metadatos de aplicaciones, archivos y configuraciones de sus instancias administradas. Este complemento se ejecuta en los sistemas operativos Linux y Microsoft Windows Server. Cuando configure la recopilación de inventario, comience por crear una asociación de AWS Systems Manager State Manager. Systems Manager recopila los datos de inventario cuando se ejecuta la asociación. Si no crea la asociación en primer lugar e intenta invocar el complemento `aws:softwareInventory`, el sistema devolverá el siguiente error:

```
The aws:softwareInventory plugin can only be invoked via ssm-associate.
```

Una instancia solo puede tener una única asociación a inventario configurada a la vez. Si configura una instancia con dos o más asociaciones, la asociación de inventario no funciona y no se recopilan los datos de inventario. Para obtener más información acerca de la recopilación de inventario, consulte Inventario de AWS Systems Manager.

Sintaxis

Esquema 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 }}"
         }
      }
   ]
}
```

Entradas

aplicaciones

(Opcional) Recopilar metadatos para las aplicaciones instaladas.

Tipo: cadena

Requerido: no

awsComponents

(Opcional) Recopilar metadatos para componentes de AWS como amazon-ssm-agent.

Tipo: cadena

Requerido: no

files

(Opcional, requiere SSM Agent versión 2.2.64.0 o posterior) Recopilar metadatos de archivos, incluidos los nombres de archivos, la hora en que se crearon los archivos, la hora en que se modificaron y se accedió por última vez a los archivos y los tamaños de los archivos, por citar algunos. Para obtener más información acerca de la recopilación de inventario de archivos, consulte Uso del inventario de archivos y del registro de Windows.

Tipo: cadena

Requerido: no

networkConfig

(Opcional) Recopilar metadatos para las configuraciones de red.

Tipo: cadena

Requerido: no

billingInfo

(Opcional) Recopile metadatos para los detalles de la plataforma asociados con el código de facturación de la AMI.

Tipo: cadena

Requerido: no

windowsUpdates

(Opcional) Recopilar metadatos de todas las actualizaciones de Windows.

Tipo: cadena

Requerido: no

InstanceDetailedInformation

(Opcional) Recopilar más información de instancia de la que proporciona el complemento de inventario predeterminado (`aws:instanceInformation`), incluido el modelo de CPU, la velocidad y el número de núcleos, entre otros.

Tipo: cadena

Requerido: no

servicios

(Opcional, solo para SO Windows, requiere SSM Agent versión 2.2.64.0 o posterior) Recopilar metadatos para configuraciones de servicio.

Tipo: cadena

Requerido: no

windowsRegistry

(Opcional, solo para SO Windows, requiere SSM Agent versión 2.2.64.0 o posterior) Recopilar claves y valores de Windows Registry. Puede elegir una ruta de clave y recopilar todas las claves y valores recursivamente. También puede recopilar una clave de registro específica y su valor

para una ruta específica. El inventario recopila la ruta de clave, el nombre, el tipo y el valor. Para obtener más información acerca de cómo recopilar el inventario de Windows Registry, consulte Uso del inventario de archivos y del registro de Windows.

Tipo: cadena

Requerido: no

windowsRoles

(Opcional, solo para SO Windows, requiere SSM Agent versión 2.2.64.0 o posterior) Recopilar metadatos para configuraciones de roles de Microsoft Windows.

Tipo: cadena

Requerido: no

customInventory

(Opcional) Recopilar datos de inventario personalizados. Para obtener más información acerca del inventario personalizado, consulte Uso del inventario personalizado.

Tipo: cadena

Requerido: no

customInventoryDirectory

(Opcional) Recopile datos de inventario personalizados desde el directorio especificado. Para obtener más información acerca del inventario personalizado, consulte Uso del inventario personalizado.

Tipo: cadena

Requerido: no

`aws:updateAgent`

Actualice el servicio EC2Config a la versión más reciente o especifique una versión más antigua. Este complemento solo se ejecuta en los sistemas operativos Microsoft Windows Server. Para obtener más información sobre el servicio EC2Config, consulte Configuración de una instancia de Windows mediante el servicio EC2Config (heredado) en la Guía del usuario de Amazon EC2.

Sintaxis

Esquema 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"
      }
    }
  ]
}
```

Esquema 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 }}"
         }
      }
   }
}
```

Propiedades

agentName

EC2Config. Este es el nombre del agente que ejecuta el servicio EC2Config.

Tipo: cadena

Obligatorio: sí

allowDowngrade

Permita el cambio del servicio EC2Config a una versión anterior. Si se establece en "false", el servicio solo se puede actualizar a versiones más recientes (predeterminado). Si se establece en "true", especifique la versión anterior.

Tipo: Booleano

Obligatorio: no

origen

La ubicación donde Systems Manager copia la versión de EC2Config que se va a instalar. No se puede cambiar esta ubicación.

Tipo: cadena

Obligatorio: sí

targetVersion

Una versión específica del servicio EC2Config que se va a instalar. Si no se especifica, el servicio se actualizará a la versión más reciente.

Tipo: cadena

Requerido: no

`aws:updateSsmAgent`

Actualice el SSM Agent a la versión más reciente o especifique una versión más antigua. Este complemento se ejecuta en los sistemas operativos Linux y Windows Server. Para obtener más información, consulte Uso de SSM Agent.

Sintaxis

Esquema 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 }}"
      }
    }
  ]
}
```

Esquema 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 }}"
            }
         ]
      }
   }
}
```

Propiedades

agentName

amazon-ssm-agent. Este es el nombre del agente de Systems Manager que procesa las solicitudes y ejecuta los comandos en la instancia.

Tipo: cadena

Obligatorio: sí

allowDowngrade

Permita el cambio del SSM Agent a una versión anterior. Si se establece en "false", el agente solo se puede actualizar a versiones más recientes (predeterminado). Si se establece en "true", especifique la versión anterior.

Tipo: Booleano

Obligatorio: sí

origen

La ubicación donde Systems Manager copia la versión del SSM Agent que se va a instalar. No se puede cambiar esta ubicación.

Tipo: cadena

Obligatorio: sí

targetVersion

Versión concreta del SSM Agent que se va a instalar. Si no se especifica, el agente se actualizará a la versión más reciente.

Tipo: cadena

Obligatorio: no

Crear contenido en el documento de SSM

Si los documentos públicos de AWS Systems Manager no llevan a cabo todas las acciones que desea realizar en sus recursos de AWS, puede crear sus propios documentos de SSM. También puede clonar documentos de SSM mediante la consola. La clonación de documentos copia el contenido de un documento existente a un documento nuevo que se puede modificar. Al crear o clonar un documento, el contenido del documento no debe superar los 64 KB. Esta cuota también incluye el contenido especificado para los parámetros de entrada en tiempo de ejecución. Cuando cree un nuevo documento `Policy` o `Command`, es recomendable que utilice la versión de esquema 2.2 o posterior para poder aprovechar las características más recientes, como la edición de documentos, el control automático de versiones, la secuenciación, etc.

Escribir contenido en el documento de SSM

Para crear su propio contenido del documento de SSM, es importante que conozca los diferentes esquemas, características, complementos y sintaxis disponibles para los documentos de SSM. Le recomendamos que se familiarice con los siguientes recursos.

- Escribir sus propios documentos de AWS Systems Manager
- Elementos y parámetros de datos
- Esquemas, características y ejemplos

- Referencia de complementos del documento de comandos
- Referencia de acciones de Automatización de Systems Manager
- Variables del sistema de Automation
- Ejemplos adicionales de manuales de procedimientos
- Trabajar con manuales de procedimientos de Automatización de Systems Manager mediante AWS Toolkit for Visual Studio Code
- Experiencia de diseño visual para manuales de procedimientos de automatización
- Uso de scripts en manuales de procedimientos

Los documentos de SSM predefinidos de AWS pueden realizar algunas de las acciones que necesita. Puede llamar a estos documentos utilizando los complementos , `aws:runDocument`, `aws:runCommand` o `aws:executeAutomation` dentro del documento de SSM personalizado, en función del tipo de documento. También puede copiar partes de esos documentos en un documento de SSM personalizado y editar el contenido de acuerdo con sus necesidades.

Tip

Cuando crea contenido del documento de SSM, puede cambiar el contenido y actualizar el documento de SSM varias veces mientras lo prueba. Los siguientes comandos actualizan el documento de SSM con el contenido más reciente y actualizan la versión predeterminada del documento a la versión más reciente del documento.

Note

Los comandos de Linux y Windows utilizan la herramienta de línea de comandos `jq` para filtrar los datos de respuesta 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
```

Prácticas recomendadas de seguridad para documentos de SSM

Al crear documentos SSM, siga estas prácticas recomendadas de seguridad para evitar la inyección de comandos y garantizar una gestión segura de los parámetros:

- Utilice la interpolación de variables de entorno para los parámetros de cadena que se utilizarán en comandos o scripts. Agregue la propiedad `interpolationType` con el valor `ENV_VAR` a los parámetros de cadena:

```
{
    "command": {
        "type": "String",
        "description": "Command to execute",
        "interpolationType": "ENV_VAR"
    }
}
```

Puede mejorar aún más la seguridad de sus documentos SSM si especifica que no se acepten comillas dobles en los valores proporcionados por interpolación:

```
{
    "command": {
        "type": "String",
        "description": "Command to execute",
        "interpolationType": "ENV_VAR",
            "allowedPattern": "^[^"]*$"
    }
}
```

- Cuando utilice lenguajes interpretados como Python, Ruby o Node.js, haga referencia a los parámetros mediante la sintaxis de variables de entorno adecuada:

```
# Python example
import os
command = os.environ['SSM_Message']
```

- A fin de garantizar la compatibilidad con versiones anteriores de SSM Agent (anteriores a la versión 3.3.2746.0), incluya una lógica alternativa para las variables de entorno:

```
if [ -z "${SSM_command+x}" ]; then
    export SSM_command="{{command}}"
fi
```

- Combine la interpolación de variables de entorno con `allowedPattern` para una validación de entrada adicional. En el siguiente ejemplo, el valor `allowedPattern` `^[^"]*$` evita específicamente las comillas dobles en el valor de la cadena:

```
{
    "command": {
```

```
        "type": "String",
        "interpolationType": "ENV_VAR",
        "allowedPattern": "^[a-zA-Z0-9_-]+$"
    }
}
```

- Antes de implementar el documento SSM, compruebe las siguientes consideraciones de seguridad:
  - Todos los parámetros de cadena que aceptan la entrada del usuario utilizan la interpolación de variables de entorno cuando es necesario.
  - La validación de las entradas se implementa mediante el uso de `allowedPattern` siempre que sea posible.
  - El documento incluye una gestión de errores adecuada para el procesamiento de los parámetros.
  - Se mantiene la compatibilidad con versiones anteriores para los entornos que utilizan versiones de SSM Agent más antiguas.

Clonar un documento de SSM

Puede clonar documentos AWS Systems Manager mediante la consola de documentos de Systems Manager para crear documentos de SSM. La clonación de documentos de SSM copia el contenido de un documento existente a un documento nuevo que se puede modificar. No puede clonar un documento que supere los 64 KB.

Para clonar un documento de SSM

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. En el cuadro de búsqueda, ingrese el nombre del documento que desea clonar.
4. Elija el nombre del documento que desea clonar y, a continuación, elija Clone document (Clonar documento) en el menú desplegable Actions (Acciones).
5. Modifique el documento como prefiera y, a continuación, elija Create document (Crear documento) para guardar el documento.

Después de escribir el contenido del documento de SSM, puede utilizarlo para crear un documento de SSM mediante uno de los métodos siguientes.

Crear documentos de SSM

- Creación de documentos compuestos

Creación de documentos compuestos

Un documento de AWS Systems Manager (SSM) compuesto es un documento personalizado que realiza una serie de acciones mediante la ejecución de uno o varios documentos de SSM secundarios. Los documentos compuestos promueven la infraestructura como código permitiéndole crear un conjunto estándar de documentos de SSM para tareas comunes como software de proceso de arranque o instancias que se unen a dominios. A continuación, puede compartir estos documentos en Cuentas de AWS en la misma Región de AWS para reducir el mantenimiento de documentos de SSM y garantizar la coherencia.

Por ejemplo, puede crear un documento compuesto que realiza las siguientes acciones:

1. Instala todos los parches de la lista de permisos.
2. Instala software antivirus.
3. Descarga scripts de GitHub y los ejecuta.

En este ejemplo, el documento de SSM personalizado incluye los siguientes complementos para llevar a cabo estas acciones:

1. El complemento `aws:runDocument` para ejecutar el documento `AWS-RunPatchBaseline`, que instala todas las revisiones permitidas que se enumeran.
2. El complemento `aws:runDocument` para ejecutar el documento `AWS-InstallApplication`, que instala el software antivirus.
3. El complemento `aws:downloadContent` para descargar scripts de GitHub y ejecutarlos.

Los documentos compuestos y secundarios se pueden almacenar en Systems Manager, GitHub (repositorios públicos y privados) o Amazon S3. Los documentos compuestos y secundarios se pueden crear en JSON o YAML.

> **Note**
>
> Los documentos compuestos solo se pueden ejecutar hasta una profundidad máxima de tres documentos. Esto significa que un documento compuesto puede llamar a un documento secundario; y que un documento secundario puede llamar a un último documento.

Para crear un documento compuesto, agregue el complemento `aws:runDocument` en un documento de SSM personalizado y especifique la información de entrada necesaria. Se muestra a continuación un ejemplo de un documento compuesto que realiza las siguientes acciones:

1. Ejecuta el complemento `aws:downloadContent` para descargar un documento de SSM de un repositorio público GitHub a un directorio local llamado bootstrap. El documento de SSM se denomina StateManagerBootstrap.yml (un documento YAML).
2. Ejecuta el complemento `aws:runDocument` para ejecutar el documento StateManagerBootstrap.yml. No se especifican parámetros.
3. Ejecuta el complemento `aws:runDocument` para ejecutar el documento de SSM `AWS-ConfigureDocker pre-defined`. Los parámetros especificados instalan Docker en la instancia.

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

Más información

- Para obtener información acerca de cómo reiniciar los servidores y las instancias cuando se utiliza Run Command para llamar a scripts, consulte Gestión de reinicios al ejecutar comandos.
- Para obtener más información acerca de los complementos que puede agregar a un documento de SSM personalizado, consulte Referencia de complementos del documento de comandos.
- Si solo desea ejecutar un documento desde una ubicación remota (sin crear un documento compuesto), consulte Ejecución de documentos de desde ubicaciones remotas.

Trabajo con documentos

Esta sección incluye información sobre cómo usar y trabajar con documentos de SSM.

Temas

- Comparar versiones de documentos de SSM
- Crear un documento de SSM
- Eliminación de documentos de SSM personalizados
- Ejecución de documentos de desde ubicaciones remotas
- Uso compartido de documentos de SSM

- Búsqueda de documentos de SSM

Comparar versiones de documentos de SSM

Puede comparar las diferencias de contenido entre las versiones de documentos AWS Systems Manager (SSM) en la consola de documentos de Systems Manager. Cuando compara versiones de un documento de SSM, se resaltan las diferencias entre el contenido de las versiones.

Para comparar el contenido del documento de SSM (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. En la lista de documentos, elija el documento cuyo contenido desea comparar.
4. En la pestaña Content (Contenido), seleccione Compare versions (Comparar versiones) y elija la versión del documento con la que desee comparar el contenido.

Crear un documento de SSM

Después de crear el contenido del documento de SSM personalizado, como se describe en Escribir contenido en el documento de SSM, puede utilizar la consola de Systems Manager para crear un documento de SSM con el contenido.

Para crear un documento de SSM

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. Elija Create command or session (Crear comando o sesión).
4. Ingrese un nombre descriptivo para asignar al documento
5. (Opcional) En Tipo de destino, especifique el tipo de recursos en los que se puede ejecutar el documento.
6. En la lista Document Type, seleccione el tipo de documento que desee crear.
7. Elimine los corchetes del campo Content (Contenido) y, a continuación, pegue el contenido del documento creado previamente.

8. (Opcional) En la sección Etiquetas de documento, aplique uno o más pares de nombre y valor de clave de etiqueta al documento.

   Las etiquetas son metadatos opcionales que usted asigna a un recurso. Las etiquetas permiten clasificar los recursos de diversas maneras, por ejemplo, según la finalidad, el propietario o el entorno. Por ejemplo, es posible que desee etiquetar un documento para identificar el tipo de tareas que ejecuta, el tipo de sistemas operativos al que se dirige y el entorno en el que se ejecuta. En este caso, puede especificar los siguientes pares de claves nombre-valor:

   - `Key=TaskType,Value=MyConfigurationUpdate`
   - `Key=OS,Value=AMAZON_LINUX_2`
   - `Key=Environment,Value=Production`
9. Elija Create document para guardar el documento.

Eliminación de documentos de SSM personalizados

Si ya no desea utilizar un documento personalizado de SSM, puede utilizar la consola de AWS Systems Manager para eliminarlo.

Para eliminar un documento de SSM

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. Seleccione el documento que desea eliminar.
4. Seleccione Delete (Eliminar). Cuando se le pida confirmación para eliminar el documento, seleccione Delete (Eliminar).

Para consultar ejemplos de cómo utilizar herramientas de línea de comandos o SDK para eliminar documentos de SSM, consulte Uso de `DeleteDocument` con un SDK de AWS o la CLI.

Ejecución de documentos de desde ubicaciones remotas

Puede ejecutar documentos de AWS Systems Manager (SSM) desde ubicaciones remotas mediante el documento `AWS-RunDocument` de SSM predefinido. Este documento admite la ejecución de documentos de SSM almacenados en las siguientes ubicaciones:

- Repositorios GitHub públicos y privados (no se admite GitHub Enterprise)

- Buckets de Amazon S3
- Systems Manager

Si bien también puede ejecutar documentos remotos con State Manager o Automatización, herramientas de AWS Systems Manager, el siguiente procedimiento describe solo cómo ejecutar documentos de SSM remotos con AWS Systems Manager Run Command en la consola de Systems Manager.

> **Note**
>
> `AWS-RunDocument` se puede emplear para ejecutar solo documentos de SSM de tipo comando, no otros tipos, como manuales de procedimientos de Automation. `AWS-RunDocument` utiliza el complemento `aws:downloadContent`. Para obtener más información acerca del complemento `aws:downloadContent`, consulte `aws:downloadContent`.

Antes de empezar

Antes de ejecutar un documento remoto, debe realizar las siguientes tareas.

- Crear un documento de SSM Command y guardarlo en una ubicación remota. Para obtener más información, consulte Crear contenido en el documento de SSM
- Si tiene previsto ejecutar un documento remoto almacenado en un repositorio GitHub privado, debe crear un parámetro `SecureString` de Systems Manager para su token de acceso de seguridad GitHub. No puede obtener acceso a un documento remoto en un repositorio GitHub privado pasando manualmente su token por SSH. El token de acceso debe pasarse como parámetro `SecureString` de Systems Manager. Para obtener más información acerca de cómo crear un parámetro `SecureString`, consulte Creación de parámetros de Parameter Store en Systems Manager.

Ejecutar un documento remoto (consola)

Para ejecutar un documento remoto

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.

2. En el panel de navegación, elija Run Command.
3. Elija Run command (Ejecutar comando).
4. En la lista Document (Documento), elija **`AWS-RunDocument`**.
5. En Command parameters (Parámetros de comando), para Source type (Tipo de origen), elija una opción.

   - Si elige GitHub, especifique la información correspondiente a Información del origen en el siguiente formato:

     ```
     {
         "owner": "owner_name",
         "repository": "repository_name",
         "path": "path_to_document",
         "getOptions":"branch:branch_name",
         "tokenInfo": "{{ssm-secure:secure-string-token}}"
     }
     ```

     Por ejemplo:

     ```
     {
         "owner":"TestUser",
         "repository":"GitHubTestExamples",
         "path":"scripts/python/test-script",
         "getOptions":"branch:exampleBranch",
         "tokenInfo":"{{ssm-secure:my-secure-string-token}}"
     }
     ```

     > **Note**
     >
     > `getOptions` son opciones adicionales para recuperar contenido de una plataforma que no sea maestra o de una confirmación específica en el repositorio. `getOptions` se pueden omitir si está utilizando la última confirmación en la rama maestra. Solo se requiere el parámetro `branch` si el documento de SSM se almacena en una sucursal que no sea `master`.
     > Para utilizar la versión de su documento SSM en una confirmación concreta del repositorio, use `commitID` con `getOptions` en lugar de `branch`. Por ejemplo:

```
"getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",
```

- Si elige S3, especifique la información correspondiente a Source Info en el siguiente formato:

```
{"path":"URL_to_document_in_S3"}
```

Por ejemplo:

```
{"path":"https://s3.amazonaws.com/amzn-s3-demo-bucket/scripts/ruby/
mySSMdoc.json"}
```

- Si elige SSMDocument, especifique la información correspondiente a Source Info en el siguiente formato:

```
{"name": "document_name"}
```

Por ejemplo:

```
{"name": "mySSMdoc"}
```

6. En el campo Document parameters (Parámetros del documento), ingrese parámetros para el documento de SSM remoto. Por ejemplo, si ejecuta el documento `AWS-RunPowerShell`, podría especificar:

```
{"commands": ["date", "echo \"Hello World\""]}
```

Si ejecuta el documento `AWS-ConfigureAWSPack`, podría especificar:

```
{
   "action":"Install",
   "name":"AWSPVDriver"
}
```

7. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

> **Tip**
>
> Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

8. En Otros parámetros:

   - En Comentario, ingrese la información acerca de este comando.
   - En Tiempo de espera (segundos), especifique el número de segundos que tiene que esperar el sistema antes de indicar que se ha producido un error en la ejecución del comando general.

9. En Rate control (Control de velocidad):

   - En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

     > **Note**
     >
     > Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

   - En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

10. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario

de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

11. En la sección Notificaciones de SNS, seleccione la casilla de verificación Habilitar notificaciones de SNS si desea recibir notificaciones sobre el estado de ejecución de los comandos.

    Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.

12. Seleccione Ejecutar.

> **Note**
>
> Para obtener información acerca de cómo reiniciar los servidores y las instancias cuando se utiliza Run Command para llamar a scripts, consulte Gestión de reinicios al ejecutar comandos.

Uso compartido de documentos de SSM

Puede compartir documentos de AWS Systems Manager (SSM) de forma privada o pública con cuentas en la misma Región de AWS. Para compartir un documento de forma privada, modifique los permisos del documento y permita que personas específicas obtengan acceso a él conforme a sus ID de Cuenta de AWS. Para compartir un documento de SSM de forma pública, modifique los permisos del documento y especifique `All`. Los documentos no se pueden compartir simultáneamente de forma pública y privada.

> **Warning**
>
> Utilice solo documentos de SSM compartidos procedentes de fuentes de confianza. Cuando utilice cualquier documento compartido, analice detenidamente el contenido del documento antes de utilizarlo para poder comprender la forma en que este cambiará la configuración de la instancia. Para obtener más información sobre las prácticas recomendadas para

documentos compartidos, consulte Prácticas recomendadas para documentos de SSM compartidos.

Limitaciones

Cuando empiece a trabajar con documentos de SSM, debe tener en cuenta las siguientes limitaciones.

- Solo el propietario puede compartir un documento.
- Debe dejar de compartir un documento para poder eliminarlo. Para obtener más información, consulte Modificar los permisos para un documento de SSM compartido.
- Puede compartir un documento con un máximo de 1000 Cuentas de AWS. Puede solicitar un aumento de este límite en el Centro de Soporte. En Tipo de límite, elija EC2 Systems Manager y describa el motivo de la solicitud.
- Puede compartir públicamente un máximo de cinco documentos de SSM. Puede solicitar un aumento de este límite en el Centro de Soporte. En Tipo de límite, elija EC2 Systems Manager y describa el motivo de la solicitud.
- Los documentos solo se pueden compartir con otras cuentas en la misma Región de AWS. No se admite el uso compartido entre regiones.

Important

En Systems Manager, un documento de SSM propiedad de Amazon es un documento creado y administrado por Amazon Web Services. Los documentos que son propiedad de Amazon incluyen un prefijo como `AWS-*` en el nombre del documento. Se considera que el propietario del documento es Amazon, no una cuenta de usuario específica de AWS. Estos documentos están a disposición del público para que los utilicen todos los usuarios.

Para obtener más información acerca de las Service Quotas de Systems Manager, consulte Service Quotas de AWS Systems Manager.

Contenido

- Prácticas recomendadas para documentos de SSM compartidos
- Bloquear el uso compartido público de documentos de SSM

- Compartir un documento de SSM
- Modificar los permisos para un documento de SSM compartido
- Uso de documentos de SSM compartidos

Prácticas recomendadas para documentos de SSM compartidos

Revise las siguientes directrices antes de compartir o utilizar un documento compartido.

Eliminar información confidencial

Revise el documento de AWS Systems Manager (SSM) detenidamente y elimine cualquier información confidencial. Por ejemplo, compruebe que el documento no incluya sus credenciales de AWS. Si comparte un documento con personas específicas, esos usuarios podrán ver la información del documento. Si comparte un documento públicamente, cualquiera podrá ver la información del documento.

Bloquear el uso compartido público de documentos

Revise todos los documentos SSM compartidos públicamente en su cuenta y confirme si quiere seguir compartiéndolos. Para dejar de compartir un documento con el público, debe modificar la configuración de permisos del documento tal y como se describe en la sección Modificar los permisos para un documento de SSM compartido de este tópico. Si activa la opción de bloquear el uso compartido con el público, no se verá afectado ningún documento que esté compartiendo con el público en ese momento. A menos que su caso de uso requiera que se permita el uso compartido público, le recomendamos que active la configuración de bloqueo de uso compartido público para sus documentos de SSM en la sección Preferencias en la consola de documentos de Systems Manager. Si se activa esta configuración, se evita el acceso no deseado a los documentos de SSM. La configuración de bloqueo de uso compartido público es una configuración a nivel de cuenta que puede diferir para cada Región de AWS.

Restringir las acciones de Run Command mediante una política de confianza de IAM

Cree una política de AWS Identity and Access Management (IAM) restrictiva para los usuarios que tendrán acceso al documento. La política de IAM determina los documentos de SSM que el usuario puede ver en la consola de Amazon Elastic Compute Cloud (Amazon EC2) o llamando a `ListDocuments` mediante la AWS Command Line Interface (AWS CLI) o AWS Tools for Windows PowerShell. La política también restringe las acciones que puede realizar el usuario con el documento de SSM. Puede crear una política restrictiva, de modo que el usuario solo pueda utilizar documentos específicos. Para obtener más información, consulte Ejemplos de políticas administradas por el cliente.

Actuar con precaución a la hora de utilizar documentos de SSM compartidos

Revise el contenido de cada documento compartido, en particular los documentos públicos, para comprender los comandos que se ejecutarán en sus instancias. Un documento podría tener repercusiones negativas después de haberlo ejecutado, ya sea de forma intencionada o no. Si el documento hace referencia a una red externa, revise la fuente externa antes de utilizar el documento.

Enviar comandos mediante el hash del documento

Al compartir un documento, el sistema crea el hash Sha-256 y lo asigna al documento. El sistema también guarda una instantánea del contenido del documento. Al enviar un comando utilizando un documento compartido, puede especificar el hash en el comando para asegurarse de que las siguientes condiciones sean verdaderas:

- Está ejecutando un comando desde el documento de Systems Manager correcto
- El contenido del documento no ha cambiado desde que se compartió con usted.

Si el hash no coincide con el documento especificado o si el contenido del documento compartido ha cambiado, el comando devuelve la excepción `InvalidDocument`. El hash no puede verificar el contenido de documentos de ubicaciones externas.

Uso del parámetro de interpolación para mejorar la seguridad

Para los parámetros de tipo `String` en sus documentos SSM, utilice el parámetro y el valor `interpolationType": "ENV_VAR` para mejorar la seguridad frente a los ataques de inyección de comandos. Esto se logra al tratar las entradas de los parámetros como cadenas literales en lugar de comandos potencialmente ejecutables. En este caso, el agente crea una variable de entorno denominada `SSM_`*`parameter-name`* con el valor del parámetro. Le recomendamos que actualice todos sus documentos SSM existentes que incluyan parámetros de tipo `String` para que incluyan `"interpolationType": "ENV_VAR"`. Para obtener más información, consulte the section called “Escribir contenido en el documento de SSM”.

Bloquear el uso compartido público de documentos de SSM

Antes de empezar, revise todos los documentos SSM compartidos públicamente que tenga en su Cuenta de AWS y confirme si quiere seguir compartiéndolos. Para dejar de compartir un documento SSM con el público, debe modificar la configuración de permisos del documento tal y como se describe en la sección Modificar los permisos para un documento de SSM compartido de este tópico. Si activa la opción de bloquear el uso compartido con el público, no se verá afectado ningún documento SSM que esté compartiendo con el público en ese momento. Si habilita la opción para

bloquear el uso compartido con el público de, no podrá compartir ningún documento SSM adicional con el público.

A menos que su caso de uso requiera que el uso compartido público esté activado, le recomendamos que active la configuración de bloqueo de uso compartido con el público para sus documentos SSM. Si se activa esta configuración, se evita el acceso no deseado a los documentos de SSM. La configuración de bloqueo de uso compartido público es una configuración a nivel de cuenta que puede diferir para cada Región de AWS. Complete las siguientes tareas para bloquear el uso compartido con el público de cualquier documento de SSM que no esté compartiendo en este momento.

Bloquear el uso compartido público (consola)

Para bloquear el uso compartido público de sus documentos de SSM

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. Elija Preferencias, luego elija Edit (Editar) en la sección Bloquear el uso compartido público.
4. Seleccione la casilla de verificación Bloquear el uso compartido público y, a continuación, elija Guardar.

Bloquear el uso compartido público (línea de comandos)

Abra el AWS Command Line Interface (AWS CLI) o AWS Tools for Windows PowerShell en el ordenador local y ejecute el siguiente comando para bloquear el uso compartido público de sus documentos de SSM.

Linux & macOS

```
aws ssm update-service-setting  \
    --setting-id /ssm/documents/console/public-sharing-permission \
    --setting-value Disable \
    --region 'The Región de 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 Región de AWS you want to block public sharing in"
```

PowerShell

```
Update-SSMServiceSetting `
    -SettingId /ssm/documents/console/public-sharing-permission `
    -SettingValue Disable `
    -Region The Región de AWS you want to block public sharing in
```

Confirme que el valor de configuración se ha actualizado mediante el siguiente comando.

Linux & macOS

```
aws ssm get-service-setting   \
    --setting-id /ssm/documents/console/public-sharing-permission \
    --region The Región de AWS you blocked public sharing in
```

Windows

```
aws ssm get-service-setting  ^
    --setting-id /ssm/documents/console/public-sharing-permission ^
    --region "The Región de AWS you blocked public sharing in"
```

PowerShell

```
Get-SSMServiceSetting `
    -SettingId /ssm/documents/console/public-sharing-permission `
    -Region The Región de AWS you blocked public sharing in
```

Restringir el acceso para bloquear el uso compartido público con IAM

Puede crear políticas AWS Identity and Access Management (IAM) que restringen a los usuarios de modificar la configuración de bloqueo de uso compartido público. Esto evita que los usuarios permitan el acceso no deseado a los documentos de SSM.

A continuación, se muestra un ejemplo de una política de IAM que impide que los usuarios actualicen la configuración de bloqueo de uso compartido público. Para utilizar este ejemplo, debe reemplazar el ejemplo ID de cuenta de Amazon Web Services por su propio ID de cuenta.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "ssm:UpdateServiceSetting",
            "Resource": "arn:aws:ssm:*:444455556666:servicesetting/ssm/documents/
console/public-sharing-permission"
        }
    ]
}
```

Compartir un documento de SSM

Puede compartir documentos AWS Systems Manager (SSM) mediante la consola de Systems Manager. Cuando se comparten documentos desde la consola, solo se puede compartir la versión predeterminada del documento. También puede compartir documentos de SSM mediante programación llamando a la operación de la API `ModifyDocumentPermission` con la AWS Command Line Interface (AWS CLI), AWS Tools for Windows PowerShell o el AWS SDK. Antes de compartir un documento, obtenga los ID de Cuenta de AWS de las personas con las que desea compartir. Especificará esos ID de cuentas cuando comparta el documento.

Compartir un documento (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. En la lista de documentos, elija el documento que desea compartir y, a continuación, elija Ver detalles. En la pestaña Permisos, verifique que es usted el propietario del documento. Solo el propietario del documento puede compartirlo.
4. Seleccione Editar.
5. Para compartir el comando públicamente, seleccione Public y, a continuación, Guardar. Para compartir el comando de forma privada, elija Privado, ingrese el ID de Cuenta de AWS, elija Añadir permiso y, a continuación, elija Guardar.

Compartir un documento (línea de comandos)

El siguiente procedimiento requiere que especifique una Región de AWS para su sesión de línea de comandos.

1. Abra la AWS CLI o AWS Tools for Windows PowerShell en el equipo local y ejecute el siguiente comando para especificar sus credenciales.

   En el siguiente comando, reemplace *region* con su propia información. Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de 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. Utilice el siguiente comando para enumerar todos los documentos de SSM que tiene disponibles. La lista incluye documentos que ha creado y documentos que han sido compartidos con usted.

Linux & macOS

```
aws ssm list-documents
```

Windows

```
aws ssm list-documents
```

PowerShell

```
Get-SSMDocumentList
```

3. Utilice el siguiente comando para obtener un documento específico.

Linux & macOS

```
aws ssm get-document \
    --name document name
```

Windows

```
aws ssm get-document ^
    --name document name
```

PowerShell

```
Get-SSMDocument `
    -Name document name
```

4. Utilice el siguiente comando para obtener una descripción del documento.

Linux & macOS

```
aws ssm describe-document \
    --name document name
```

Windows

```
aws ssm describe-document ^
```

```
    --name document name
```

PowerShell

```
Get-SSMDocumentDescription `
    -Name document name
```

5. Utilice el siguiente comando para ver los permisos para el documento.

   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. Utilice el siguiente comando para modificar los permisos para el documento y compartirlo. Debe ser el propietario del documento para editar los permisos. Si lo desea, en el caso de los documentos compartidos con ID de la Cuenta de AWS específicos, puede especificar la versión del documento que quiere compartir con el parámetro `--shared-document-version`. Si no especifica una versión, el sistema comparte la versión `Default` del documento. Si comparte un documento de manera pública (con `all`), todas las versiones del documento especificado se comparten de forma predeterminada. El siguiente comando de ejemplo comparte el documento de forma privada con una persona específica en función de su ID de Cuenta de AWS.

   Linux & macOS

   ```
   aws ssm modify-document-permission \
       --name document name \
   ```

```
    --permission-type Share \
    --account-ids-to-add Cuenta de AWS ID
```

Windows

```
aws ssm modify-document-permission ^
    --name document name ^
    --permission-type Share ^
    --account-ids-to-add Cuenta de AWS ID
```

PowerShell

```
Edit-SSMDocumentPermission `
    -Name document name `
    -PermissionType Share `
    -AccountIdsToAdd Cuenta de AWS ID
```

7. Utilice el siguiente comando para compartir un documento públicamente.

> **Note**
>
> Si comparte un documento de manera pública (con `all`), todas las versiones del documento especificado se comparten de forma predeterminada.

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')
```

Modificar los permisos para un documento de SSM compartido

Si comparte un comando, los usuarios pueden ver y utilizar el comando hasta que usted quite el acceso al documento de AWS Systems Manager (SSM) o elimine el documento de SSM. Sin embargo, no puede eliminar un documento mientras esté compartido. Primero debe dejar de compartirlo y, a continuación, eliminarlo.

Detener el uso compartido de un documento (consola)

Detener el uso compartido de un documento

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. En la lista de documentos, elija el documento que desea dejar de compartir y, a continuación, elija Detalles. En la sección Permisos, verifique que es usted el propietario del documento. Solo el propietario del documento puede dejar de compartirlo.
4. Seleccione Editar.
5. Elija X para eliminar el ID de Cuenta de AWS que ya no deberá tener acceso al comando y, a continuación, elija Guardar.

Dejar de compartir un documento (línea de comandos)

Abra la AWS CLI o AWS Tools for Windows PowerShell en el equipo local y ejecute el siguiente comando para dejar de compartir un comando.

Linux & macOS

```
aws ssm modify-document-permission \
    --name document name \
```

```
    --permission-type Share \
    --account-ids-to-remove 'Cuenta de AWS ID'
```

Windows

```
aws ssm modify-document-permission ^
    --name document name ^
    --permission-type Share ^
    --account-ids-to-remove "Cuenta de AWS ID"
```

PowerShell

```
Edit-SSMDocumentPermission `
    -Name document name `
    -PermissionType Share `
    –AccountIdsToRemove Cuenta de AWS ID
```

Uso de documentos de SSM compartidos

Cuando comparte un documento de AWS Systems Manager (SSM), el sistema genera un nombre de recurso de Amazon (ARN) y lo asigna al comando. Si selecciona y ejecuta un documento compartido desde la consola de Systems Manager, no verá el ARN. Sin embargo, si desea ejecutar un documento de SSM compartido utilizando un método distinto de la consola de Systems Manager, debe especificar el ARN completo del documento para el parámetro de solicitud de `DocumentName`. Cuando ejecuta el comando para enumerar los documentos verá el ARN completo de un documento de SSM.

> **Note**
>
> No es necesario especificar los ARN para documentos públicos de AWS (documentos que comienzan por `AWS-*`) o para documentos de su propiedad.

Uso de un documento de SSM compartido (línea de comandos)

Para enumerar todos los documentos de SSM públicos

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)
```

Para enumerar los documentos de SSM privados que se han compartido con usted

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)
```

Para enumerar todos los documentos de SSM que tiene disponibles

Linux & macOS

```
aws ssm list-documents
```

Windows

```
aws ssm list-documents
```

PowerShell

```
Get-SSMDocumentList
```

Para obtener información acerca de un documento de SSM que se ha compartido con usted

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
```

Para ejecutar un documento de SSM compartido

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
```

Búsqueda de documentos de SSM

Puede buscar documentos de SSM en el almacén de documentos AWS Systems Manager (SSM) mediante la búsqueda de texto libre o una búsqueda basada en filtros. También puede marcar documentos como favoritos para ayudarlo a encontrar documentos de SSM usados con frecuencia. En las siguientes secciones, se describe cómo utilizar estas características.

Uso de la búsqueda de texto libre

El cuadro de búsqueda en la página Documents (Documentos) de Systems Manager admite las búsquedas de texto libre. La búsqueda de texto libre compara el término o los términos de búsqueda que introduce con el nombre del documento en cada documento de SSM. Si introduce un solo término de búsqueda, por ejemplo **`ansible`**, Systems Manager devuelve todos los documentos de SSM donde se encontró este término. Si introduce varios términos de búsqueda, Systems Manager buscará mediante una instrucción `OR`. Por ejemplo, si especifica **`ansible`** y **`linux`**, a continuación, la búsqueda devuelve todos los documentos con cualquier palabra clave en su nombre.

Si ingresa un término de búsqueda de texto libre y elige una opción de búsqueda, como Platform type (Tipo de plataforma), a continuación, la búsqueda utiliza una instrucción `AND` y devuelve todos los documentos con la palabra clave en su nombre y el tipo de plataforma especificado.

> **Note**
>
> Tenga en cuenta los siguientes detalles sobre la búsqueda de texto libre.
>
> - La búsqueda de texto libre no distingue entre mayúsculas y minúsculas.

- Los términos de búsqueda deben tener 8 caracteres como mínimo y 20 caracteres como máximo.
- La búsqueda de texto libre acepta hasta cinco términos de búsqueda.
- Si ingresa un espacio entre los términos de búsqueda, el sistema incluye el espacio al realizar la búsqueda.
- Puede combinar la búsqueda de texto libre con otras opciones de búsqueda, como Document type (Tipo de documento) o Platform type (Tipo de plataforma).
- El filtro Document name prefix (Prefijo de nombre del documento) y la búsqueda de texto libre no se pueden utilizar juntos. Son mutuamente excluyentes.

Para buscar un documento de SSM

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. Escriba los términos de búsqueda en el cuadro de búsqueda y presione “Enter” (Ingresar).

Realizar la búsqueda de documentos de texto libre mediante la AWS CLI

Para realizar la búsqueda de documentos de texto libre mediante la CLI

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.
2. Para realizar búsquedas de documentos de texto libre con un solo término, ejecute el siguiente comando. En este comando, sustituya *search_term* con su propia información.

   ```
   aws ssm list-documents --filters Key="SearchKeyword",Values="search_term"
   ```

   A continuación se muestra un ejemplo.

   ```
   aws ssm list-documents --filters Key="SearchKeyword",Values="aws-asg" --region us-east-2
   ```

Para realizar una búsqueda con varios términos que crean una instrucción AND ejecute el siguiente comando. En este comando, sustituya *search_term_1* y *search_term_2* con su propia información.

```
aws ssm list-documents --filters
 Key="SearchKeyword",Values="search_term_1","search_term_2","search_term_3" --
region us-east-2
```

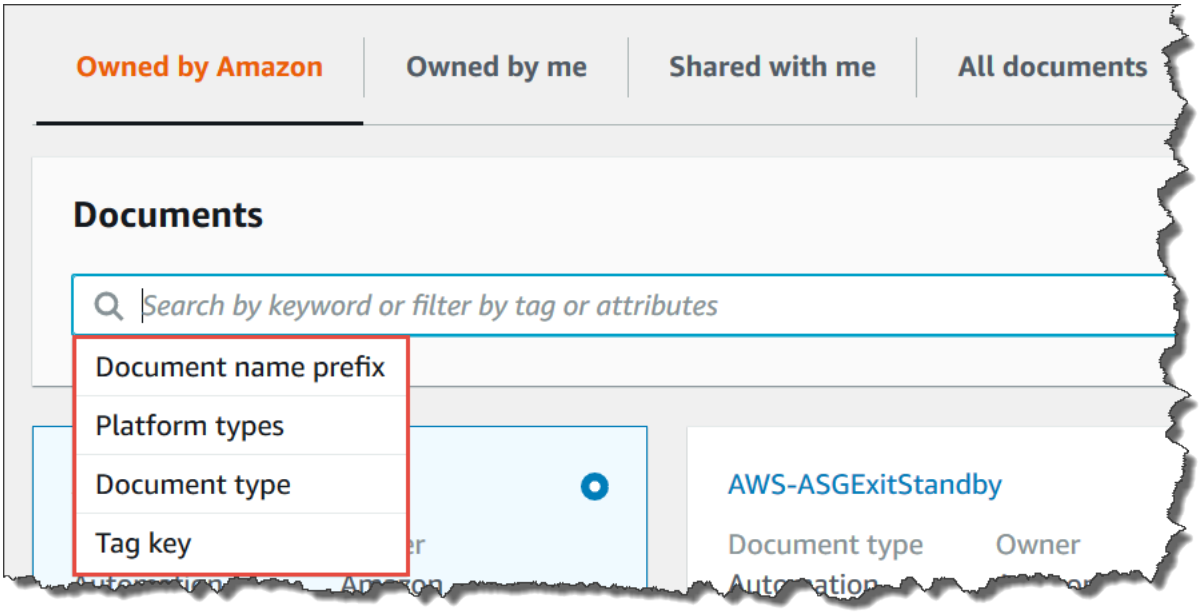
A continuación se muestra un ejemplo.

```
aws ssm list-documents --filters Key="SearchKeyword",Values="aws-asg","aws-
ec2","restart" --region us-east-2
```

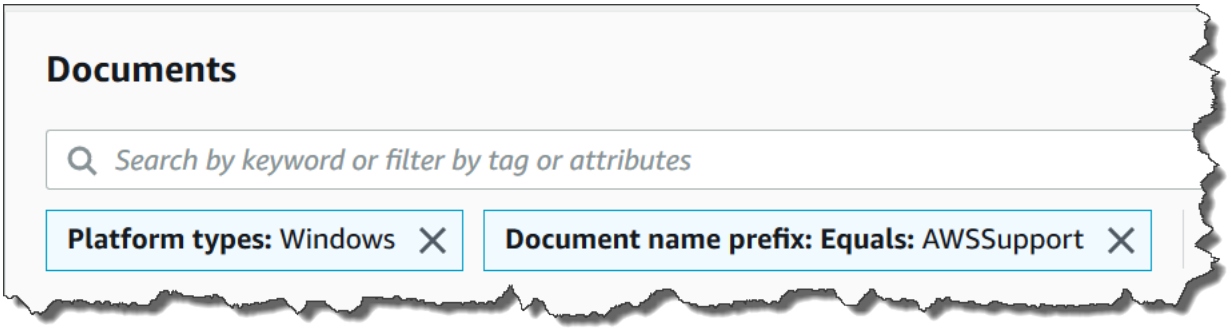
Uso de filtros

La página Documents (Documentos) de Systems Manager muestra automáticamente los siguientes filtros cuando elije el cuadro de búsqueda.

- Prefijo de nombre del documento
- Tipos de plataformas
- Tipo de documento
- Clave de etiqueta



Puede buscar documentos de SSM mediante un único filtro. Si desea volver un conjunto más específico de documentos de SSM, puede aplicar varios filtros. A continuación, se muestra un ejemplo de una búsqueda que utiliza los filtros Platform types (Tipos de plataforma) y Document name prefix (Prefijo de nombre del documento).



Si aplica varios filtros, Systems Manager crea instrucciones de búsqueda diferentes basadas en los filtros que elija:

- Si aplica el mismo filtro varias veces, por ejemplo Document name prefix (Prefijo de nombre del documento), a continuación, Systems Manager busca mediante una instrucción `OR`. Por ejemplo, si especifica un filtro Prefijo de nombre del documento=**`AWS`** y un segundo filtro Prefijo de nombre del documento=**`Lambda`**, a continuación, la búsqueda devuelve todos los documentos con el prefijo “`AWS`” y todos los documentos con el prefijo “`Lambda`”.
- Si aplica filtros diferentes, por ejemplo Document name prefix (Prefijo de nombre del documento) y Platform types (Tipos de plataformas), Systems Manager realiza la búsqueda utilizando una

instrucción AND. Por ejemplo, si especifica un filtro Document name prefix (Prefijo de nombre del documento) = **AWS** y un filtro Platform types (Tipos de plataformas) = **Linux**, la búsqueda devuelve todos los documentos con el prefijo "AWS" específicos de la plataforma Linux.

> **Note**
>
> Las búsquedas que utilizan filtros distinguen entre mayúsculas y minúsculas.

Adición de documentos a favoritos

Para ayudarlo a encontrar los documentos de SSM que usa con frecuencia, agregue documentos a sus favoritos. Puede marcar como favoritos hasta 20 documentos por tipo de documento, por Cuenta de AWS y Región de AWS. Puede elegir, modificar y ver sus favoritos desde la AWS Management Console de documentos. En los siguientes procedimientos, se describe cómo seleccionar, modificar y ver los favoritos.

Para agregar un documento de SSM a favoritos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. Seleccione el icono de estrella situado junto al nombre del documento que desea marcar como favorito.

Para eliminar un documento de SSM de favoritos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Documentos.
3. Anule la selección del icono de estrella junto al nombre del documento que desea eliminar de sus favoritos.

Para ver los favoritos desde la AWS Management Console de documentos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.

2. En el panel de navegación, elija Documentos.
3. Seleccione la pestaña Favoritos.

Solución de problemas de gestión de parámetros

Problemas comunes de gestión de parámetros

Las variables de entorno no están disponibles durante la ejecución

Problema: los comandos fallan porque no se encuentran las variables de entorno (`SSM_parameter-name`).

Causas posibles:

- La versión de SSM Agent no admite la interpolación de variables de entorno
- `interpolationType` no está configurado como `ENV_VAR`
- El nombre del parámetro no coincide con el nombre de la variable de entorno esperada

Solución:

- Compruebe que la versión de SSM Agent sea 3.3.2746.0 o posterior
- Agregue una lógica alternativa para las versiones anteriores del agente:

```
if [ -z "${SSM_parameterName+x}" ]; then
    export SSM_parameterName="{{parameterName}}"
fi
```

Valores de parámetro que contienen caracteres especiales

Problema: los comandos fallan cuando los valores de los parámetros contienen espacios, comillas u otros caracteres especiales.

Solución:

- Utilice las comillas adecuadas al hacer referencia a las variables de entorno:

```
# Correct
echo "$SSM_parameter-name"

# Incorrect
```

```
echo $SSM_parameter-name
```

- Agregue la validación de entrada mediante `allowedPattern` para restringir los caracteres especiales

Comportamiento incoherente en todas las plataformas

Problema: la gestión de parámetros funciona de manera diferente en Linux y en los sistemas Windows Server.

Solución:

- Utilice la sintaxis de variables de entorno específica de la plataforma:

```
# PowerShell
$env:SSM_parameter-name

# Bash
$SSM_parameter-name
```

- Utilice en su documento comprobaciones de condiciones previas específicas de la plataforma

Los valores de los parámetros no se han filtrado correctamente

Problema: hay vulnerabilidades en la inyección de comandos a pesar de utilizar la interpolación de variables de entorno.

Solución:

- Utilice siempre el método de escape adecuado al incluir valores de parámetros en los comandos:

```
# Correct
mysql_command="mysql -u \"$SSM_username\" -p\"$SSM_password\""

# Incorrect
mysql_command="mysql -u $SSM_username -p$SSM_password"
```

Consejos de validación de parámetros

Utilice estas técnicas para validar su gestión de parámetros:

1. Pruebe la disponibilidad de las variables del entorno:

```
#!/bin/bash
# Print all SSM_ environment variables
env | grep ^SSM_

# Test specific parameter
if [ -n "$SSM_parameter" ]; then
    echo "Parameter is available"
else
    echo "Parameter is not available"
fi
```

2. Verifique los patrones de los parámetros:

```
parameters:
  myParameter:
    type: String
    allowedPattern: "^[a-zA-Z0-9_-]+$"
    description: "Test this pattern with sample inputs"
```

3. Incluya la gestión de errores:

```
if [[ ! "$SSM_parameter" =~ ^[a-zA-Z0-9_-]+$ ]]; then
    echo "Parameter validation failed"
    exit 1
fi
```

AWS Systems Manager Maintenance Windows

Maintenance Windows, una herramienta de AWS Systems Manager, permite definir una programación en el momento de llevar a cabo acciones potencialmente disruptivas en los nodos, como la aplicación de revisiones en un sistema operativo, la actualización de controladores o la instalación de software o revisiones.

> **Note**
>
> State Manager y Maintenance Windows pueden realizar algunos tipos similares de actualizaciones en los nodos administrados. La opción que elija dependerá de si necesita automatizar la conformidad del sistema o realizar tareas de alta prioridad y urgencia durante los periodos que especifique.

Para obtener más información, consulte Elección entre State Manager y Maintenance Windows.

Gracias a Maintenance Windows, puede programar acciones en muchos otros recursos de AWS, como los buckets de Amazon Simple Storage Service (Amazon S3), las colas de Amazon Simple Queue Service (Amazon SQS), las claves de AWS Key Management Service (AWS KMS) y muchos más.

Para obtener una lista completa de tipos de recursos admitidos que puede incluir en un destino de periodo de mantenimiento, consulte Recursos que puede utilizar con AWS Resource Groups y Tag Editor en la Guía del usuario de AWS Resource Groups. Para comenzar a utilizar Maintenance Windows, abra la consola de Systems Manager. En el panel de navegación, elija Maintenance Windows.

Cada periodo de mantenimiento tiene una programación, una duración máxima, un conjunto de destinos registrados (los nodos administrados u otros recursos de AWS sobre los que se actúa) y un conjunto de tareas registradas. Puede agregar etiquetas a sus periodos de mantenimiento en el momento de crearlos o actualizarlos. (Las etiquetas son claves que ayudan a identificar y ordenar los recursos de la organización). También puede especificar fechas en las que un periodo de mantenimiento no debe ejecutarse antes ni después, y puede especificar la zona horaria internacional en la que basar la programación del periodo de mantenimiento.

Para ver una explicación de cómo se relacionan entre sí las distintas opciones relacionadas con la programación de los periodos de mantenimiento, consulte Programación de la ventana de mantenimiento y opciones de periodo activo.

Para obtener más información acerca del uso de la opción `--schedule`, consulte Referencia: expresiones cron y rate para Systems Manager.

Tipos de tareas admitidas

Con los periodos de mantenimiento puede ejecutar cuatro tipos de tareas:

- Comandos en Run Command, una herramienta de Systems Manager

  Para obtener más información acerca de Run Command, consulte AWS Systems Manager Run Command.
- Flujos de trabajo de Automatización, una herramienta de Systems Manager

Para obtener más información acerca de los flujos de trabajo de Automation, consulte AWS Systems Manager Automation.

- Funciones de AWS Lambda

  Para obtener más información acerca de las funciones de Lambda, consulte Introducción a Lambda en la Guía para desarrolladores de AWS Lambda.

- Tareas de AWS Step Functions

  > **Note**
  >
  > Las tareas de los periodos de mantenimiento solo admiten los flujos de trabajo de máquinas de estado estándar de Step Functions. No son compatibles con los flujos de trabajo de máquinas de estado rápidas. Para obtener información sobre los tipos de flujos de trabajo de máquinas de estado, consulte Flujos de trabajo estándar en comparación con flujos de trabajo rápidos en la Guía para desarrolladores de AWS Step Functions.

  Para obtener más información acerca de Step Functions, consulta la AWS Step FunctionsGuía para desarrolladores de .

Esto significa que puede utilizar los periodos de mantenimiento para realizar tareas tales como las siguientes en los destinos seleccionados.

- Instalar o actualizar aplicaciones.
- Aplicar revisiones.
- Instalar o actualizar SSM Agent.
- Ejecutar comandos de PowerShell y scripts de shell de Linux mediante una tarea de Systems Manager Run Command.
- Crear Amazon Machine Images (AMIs), arrancar software y configurar instancias mediante una tarea de Automatización de Systems Manager.
- Ejecutar funciones de AWS Lambda que invocan acciones adicionales, como el análisis de nodos para actualizar las revisiones.
- Ejecutar máquinas de estado de AWS Step Functions para realizar ciertas tareas, como quitar un nodo de un entorno de Elastic Load Balancing, aplicar revisiones a ese nodo y agregarlo de nuevo al entorno de Elastic Load Balancing.

- Dirigirse a los nodos sin conexión mediante la especificación de un grupo de recursos de AWS como destino.

> **Note**
>
> Se deben especificar uno o más destinos para las tareas de tipo Run Command del periodo de mantenimiento. Según la tarea, los destinos son opcionales para otros tipos de tarea de periodo de mantenimiento (Automation, AWS Lambda y AWS Step Functions). Para obtener más información acerca de la ejecución de tareas que no especifican destinos, consulte Registro de tareas del periodo de mantenimiento sin destinos.

Compatibilidad con EventBridge

Esta herramienta de Systems Manager se admite como un tipo de evento en las reglas de Amazon EventBridge. Para obtener más información, consulte Monitoreo de eventos de Systems Manager con Amazon EventBridge y Referencia: patrones y tipos de eventos de Amazon EventBridge para Systems Manager.

Contenido

- Configuración de Maintenance Windows
- Crear y administrar periodos de mantenimiento mediante la consola
- Tutoriales
- Utilización de pseudoparámetros en el registro de las tareas del periodo de mantenimiento
- Programación de la ventana de mantenimiento y opciones de periodo activo
- Registro de tareas del periodo de mantenimiento sin destinos
- Solución de problemas de periodos de mantenimiento

Configuración de Maintenance Windows

Para que los usuarios de su Cuenta de AWS puedan crear y programar tareas de periodo de mantenimiento mediante Maintenance Windows, una herramienta de AWS Systems Manager, es preciso que antes se les concedan los permisos necesarios. Además, debe crear un rol de servicio de IAM para los periodos de mantenimiento y la política de IAM que se le asocie.

Antes de empezar

Además de los permisos que configure en esta sección, las entidades de IAM (usuarios, roles o grupos que funcionarán con los periodos de mantenimiento ya deberían tener permisos de periodos de mantenimiento generales. Puede conceder estos permisos al asignar la política de IAM `AmazonSSMFullAccess` a las entidades, o asignar una política de IAM personalizada que proporcione un conjunto más pequeño de permisos de acceso a Systems Manager que abarque las tareas de periodos de mantenimiento.

Temas

- Cómo controlar el acceso a periodos de mantenimiento mediante la consola
- Control del acceso a los periodos de mantenimiento mediante la AWS CLI

Cómo controlar el acceso a periodos de mantenimiento mediante la consola

Los siguientes procedimientos describen cómo usar la consola de AWS Systems Manager para crear los permisos y los roles necesarios para periodos de mantenimiento.

Temas

- Tarea 1: cree una política personalizada para el rol de servicio de periodo de mantenimiento mediante la consola
- Tarea 2: cree un rol de servicio personalizado para periodos de mantenimiento mediante la consola
- Tarea 3: conceda permisos a usuarios específicos para registrar tareas de periodos de mantenimiento mediante la consola
- Tarea 4: impida que usuarios específicos registren las tareas de periodos de mantenimiento mediante la consola

Tarea 1: cree una política personalizada para el rol de servicio de periodo de mantenimiento mediante la consola

Las tareas de periodo de mantenimiento requieren un rol de IAM para proporcionar los permisos necesarios para su ejecución en los recursos de destino. Los permisos se conceden mediante una política de IAM asociada al rol. El contenido de esta política lo determinan los tipos de tareas que se ejecutan y el resto de requisitos operativos. Proporcionamos una política básica que se puede adaptar a sus necesidades. En función de las tareas y los tipos de tareas que ejecuten los periodos de mantenimiento, es posible que no necesite todos los permisos de esta política, o que tenga que

incluir permisos adicionales. Más tarde adjuntará esta política al rol que cree en Tarea 2: cree un rol de servicio personalizado para periodos de mantenimiento mediante la consola.

Cómo crear una política personalizada mediante la consola

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, seleccione Policies (Políticas) y, a continuación, seleccione Create policy (Crear política).
3. En el área del Editor de políticas, elija JSON.
4. Reemplace el contenido predeterminado con lo siguiente:

   JSON

```
{
    "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:DescribeInstanceInformation",
                "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": "arn:aws:iam::111122223333:role/maintenance-window-
role-name",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
                        "ssm.amazonaws.com"
                    ]
                }
            }
        }
    ]
```

```
}
```

5. Modifique el contenido JSON de acuerdo con lo que necesiten las tareas de mantenimiento que ejecute en la cuenta. Los cambios que realice son específicos de las operaciones planificadas.

   Por ejemplo:

   - Puede proporcionar nombres de recursos de Amazon (ARN) para funciones y máquinas de estado específicas en lugar de utilizar calificadores comodín (*).
   - Si no tiene previsto ejecutar tareas de AWS Step Functions, puede quitar los permisos y ARN de `states`.
   - Si no tiene previsto ejecutar tareas de AWS Lambda, puede quitar los permisos y ARN de `lambda`.
   - Si no tiene previsto ejecutar tareas de automatización, puede quitar los permisos `ssm:GetAutomationExecution` y `ssm:StartAutomationExecution`.
   - Agregue permisos adicionales que puedan ser necesarios para que se ejecuten las tareas. Por ejemplo, algunas acciones de Automation trabajan con pilas de AWS CloudFormation. Por lo tanto, los permisos `cloudformation:CreateStack`, `cloudformation:DescribeStacks` y `cloudformation:DeleteStack` son obligatorios.

     Otro ejemplo es el manual de procedimientos de Automation `AWS-CopySnapshot`, que requiere permisos para crear una instantánea de Amazon Elastic Block Store (Amazon EBS). Por lo tanto, el rol de servicio necesita los permisos `ec2:CreateSnapshot`.

     Para obtener información acerca de los permisos de rol que necesitan los manuales de procedimientos de automatización, consulte las descripciones de documentos en la Referencia de manuales de procedimientos de Automatización de AWS Systems Manager.

6. Cuando haya completado las revisiones de la política, elija Siguiente.
7. En Nombre de la política, ingrese un nombre que identifique esta como la política asociada al rol de servicio de que cree. Por ejemplo: **`my-maintenance-window-role-policy`**.
8. (Opcional) En el área de Agregar etiquetas, agregue uno o varios pares de valor etiqueta-clave para organizar, realizar un seguimiento o controlar el acceso a esta política.
9. Seleccione Crear política.

   Anote el nombre que haya especificado para la política. Hará referencia a él en el siguiente procedimiento, Tarea 2: cree un rol de servicio personalizado para periodos de mantenimiento mediante la consola.

Tarea 2: cree un rol de servicio personalizado para periodos de mantenimiento mediante la consola

La política que se crea en la tarea anterior se asocia al rol de servicio del periodo de mantenimiento que se crea en esta tarea. Cuando los usuarios registran una tarea de periodo de mantenimiento, especifican este rol de IAM como parte de la configuración de la tarea. Los permisos de este rol autorizan a Systems Manager a ejecutar tareas en periodos de mantenimiento en su nombre.

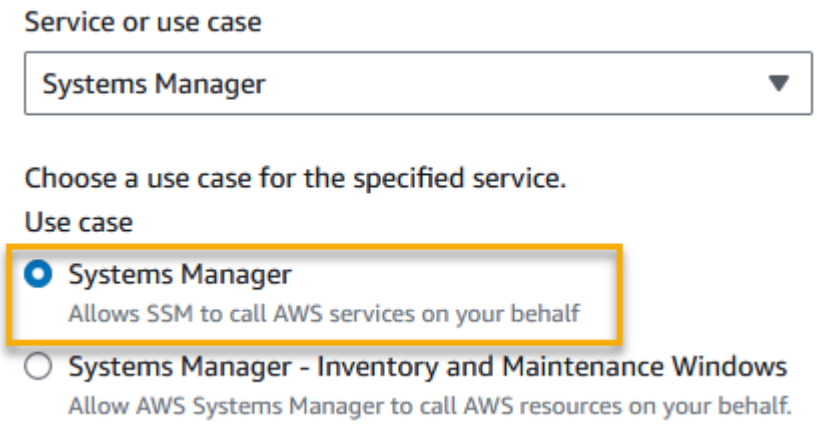
> **Important**
>
> Antes, la consola de Systems Manager ofrecía la posibilidad de elegir el rol vinculado a servicio de IAM `AWSServiceRoleForAmazonSSM` administrado de AWS que utilizar como rol de mantenimiento para las tareas. Ya no se recomienda utilizar este rol y su política asociada, `AmazonSSMServiceRolePolicy`, para tareas de periodo de mantenimiento. Si está utilizando actualmente este rol para tareas de periodo de mantenimiento, le recomendamos que deje de hacerlo. En su lugar, cree su propio rol de IAM que permita la comunicación entre Systems Manager y otros Servicios de AWS cuando se ejecuten las tareas de periodo de mantenimiento.

Utilice el siguiente procedimiento para crear un rol de servicio personalizado para Maintenance Windows, de modo que Systems Manager pueda ejecutar tareas de Maintenance Windows en su nombre. Asocie la política que ha creado en la tarea anterior al rol de servicio personalizado que cree.

Cómo crear un rol de servicio personalizado para periodos de mantenimiento mediante la consola

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, seleccione Roles y luego seleccione Crear rol.
3. En Select trusted entity (Seleccionar entidad de confianza), realice las siguientes elecciones:
    1. En Tipo de entidad de confianza, elija Servicio de AWS.
    2. En Caso de uso, elija Systems Manager
    3. Elija Configurar Systems Manager.

       La siguiente imagen resalta la ubicación de la opción Systems Manager.



4. Elija Siguiente.
5. En el área de Políticas de permisos, en el cuadro de búsqueda, ingrese el nombre de la política que ha creado en Tarea 1: cree una política personalizada para el rol de servicio de periodo de mantenimiento mediante la consola, seleccione la casilla situada junto a su nombre y, a continuación, elija Siguiente.
6. En el campo Role name (Nombre del rol), ingrese un nombre que identifique a este rol como un rol de Maintenance Windows. Por ejemplo: **`my-maintenance-window-role`**.
7. (Opcional) Puede cambiar la descripción predeterminada del rol para que refleje su finalidad. Por ejemplo: **`Performs maintenance window tasks on your behalf`**.
8. Para el Paso 1: selección de entidades de confianza, compruebe que la siguiente política aparezca en el cuadro Política de confianza.

   JSON

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

9. En el Paso 2: agregado de permisos, compruebe que la política que creó en Tarea 1: cree una política personalizada para el rol de servicio de periodo de mantenimiento mediante la consola esté presente.
10. (Opcional) En el Paso 3: agregado de etiquetas, agregue uno o varios pares de valor etiqueta-clave para organizar, realizar un seguimiento o controlar el acceso a este rol.
11. Elija Create role. El sistema le devuelve a la página Roles.
12. Elija el nombre del rol de IAM que acaba de crear.
13. Copie o anote el nombre del rol y el valor del ARN del área Summary (Resumen). Los usuarios de la cuenta tendrán que especificar esta información cuando creen periodos de mantenimiento.

Tarea 3: conceda permisos a usuarios específicos para registrar tareas de periodos de mantenimiento mediante la consola

Conceder a los usuarios permisos para acceder al rol de servicio del periodo de mantenimiento personalizado les permite utilizarlo con sus tareas de periodos de mantenimiento. Esto se suma a los permisos que ya se les haya concedido para utilizar los comandos de la API de Systems Manager para la herramienta de Maintenance Windows. Este rol de IAM transmite los permisos necesarios para ejecutar una tarea de periodo de mantenimiento. Como resultado, un usuario no puede registrar tareas con un periodo de mantenimiento mediante el rol de servicio personalizado sin la posibilidad de transferir esos permisos de IAM.

Cuando registra una tarea en un periodo de mantenimiento, debe especificar un rol de servicio para ejecutar las operaciones de la tarea real. Este es el rol que el servicio asume cuando ejecuta las tareas en su nombre. Antes de eso, para registrar la tarea en sí, asigne la política de IAM `PassRole` a una entidad de IAM (como un usuario o un grupo). Esto permite que la entidad de IAM especifique, como parte del registro de esas tareas con el periodo de mantenimiento, el rol que se debe usar al ejecutar tareas. Para obtener más información, consulte Concesión de permisos a un usuario para transferir un rol a un Servicio de AWS en la Guía del usuario de IAM.

Cómo configurar permisos para que los usuarios puedan registrar tareas de periodos de mantenimiento

Si una entidad de IAM (usuario, rol o grupo) está configurada con permisos de administrador, el usuario de IAM o el rol tendrá acceso a los periodos de mantenimiento. Para las entidades de IAM sin permisos de administrador, el administrador debe conceder los siguientes permisos a la entidad de IAM. Estos son los permisos mínimos necesarios para registrar tareas con un periodo de mantenimiento:

- La política administrada `AmazonSSMFullAccess`, o una política que proporcione permisos comparables.
- Los siguientes permisos `iam:PassRole` y `iam:ListRoles`.

  JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::111122223333:role/my-maintenance-window-
role"
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::111122223333:role/"
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::111122223333:role/aws-service-role/
ssm.amazonaws.com/"
        }
    ]
}
```

my-maintenance-window-role representa el nombre del rol de servicio del periodo de mantenimiento personalizado que ha creado antes.

account-id representa el ID de su Cuenta de AWS. Al agregar este permiso para el recurso `arn:aws:iam::`*account-id*`:role/` se permite al usuario ver y elegir entre los roles de cliente en la consola cuando crea una tarea del periodo de mantenimiento. La adición de este permiso para `arn:aws:iam::`*account-id*`:role/aws-service-role/ssm.amazonaws.com/` permite al usuario elegir el rol vinculado al servicio de Systems Manager en la consola cuando se crea una tarea del periodo de mantenimiento.

Para dar acceso, agregue permisos a los usuarios, grupos o roles:

- Usuarios y grupos en AWS IAM Identity Center:

Cree un conjunto de permisos. Siga las instrucciones de Creación de un conjunto de permisos en la Guía del usuario de AWS IAM Identity Center.

- Usuarios gestionados en IAM a través de un proveedor de identidades:

  Cree un rol para la federación de identidades. Siga las instrucciones descritas en Creación de un rol para un proveedor de identidad de terceros (federación) en la Guía del usuario de IAM.

- Usuarios de IAM:
  - Cree un rol que el usuario pueda aceptar. Siga las instrucciones descritas en Creación de un rol para un usuario de IAM en la Guía del usuario de IAM.
  - (No recomendado) Adjunte una política directamente a un usuario o añada un usuario a un grupo de usuarios. Siga las instrucciones descritas en Adición de permisos a un usuario (consola) de la Guía del usuario de IAM.

Cómo configurar permisos para grupos con autorización para registrar tareas de periodos de mantenimiento mediante la consola

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, elija User groups (Grupos de usuarios).
3. En la lista de grupos, seleccione el nombre del grupo al que desea asignar el permiso `iam:PassRole` o primero cree un nuevo grupo si es necesario
4. En la pestaña Permissions (Permisos), elija Add permissions, Create Inline Policy (Agregar permisos, Crear política insertada).
5. En el área del Editor de la política, elija JSON y reemplace los contenidos predeterminados del cuadro con lo siguiente.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::111122223333:role/my-maintenance-
window-role"
        },
```

```
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::111122223333:role/"
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::111122223333:role/aws-service-role/
ssm.amazonaws.com/"
        }
    ]
}
```

my-maintenance-window-role representa el nombre del rol de la ventana de mantenimiento personalizada que ha creado antes.

account-id representa el ID de su Cuenta de AWS. Al agregar este permiso para el recurso `arn:aws:iam::`*account-id*`:role/` se permite al usuario ver y elegir entre los roles de cliente en la consola cuando crea una tarea del periodo de mantenimiento. La adición de este permiso para `arn:aws:iam::`*account-id*`:role/aws-service-role/ssm.amazonaws.com/` permite al usuario elegir el rol vinculado al servicio de Systems Manager en la consola cuando se crea una tarea del periodo de mantenimiento.

6. Elija Siguiente.
7. En la página Revisar política, ingrese un nombre en el cuadro Nombre de la política para identificar esta política `PassRole`, como por ejemplo **`my-group-iam-passrole-policy`**, y luego, elija Crear política.

Tarea 4: impida que usuarios específicos registren las tareas de periodos de mantenimiento mediante la consola

Puede denegar el permiso `ssm:RegisterTaskWithMaintenanceWindow` a los usuarios de la Cuenta de AWS que no desee que registren tareas en periodos de mantenimiento. Esto proporciona una capa adicional de protección frente a usuarios que no deban registrar tareas de periodo de mantenimiento.

Cómo configurar permisos para grupos que no pueden asignar tareas de periodos de mantenimiento mediante la consola

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, elija User groups (Grupos de usuarios).
3. En la lista de grupos, seleccione el nombre del grupo al que desea denegar el formulario de permiso `ssm:RegisterTaskWithMaintenanceWindow`, o primero cree un nuevo grupo si es necesario.
4. En la pestaña Permissions (Permisos), elija Add permissions, Create Inline Policy (Agregar permisos, Crear política insertada).
5. En el área del Editor de la política, elija JSON y, luego, reemplace los contenidos predeterminados del cuadro con lo siguiente.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "ssm:RegisterTaskWithMaintenanceWindow",
            "Resource": "*"
        }
    ]
}
```

6. Elija Siguiente.
7. En la página Revisar y crear, ingrese un nombre en Nombre de la política para identificar esta política, como por ejemplo **`my-groups-deny-mw-tasks-policy`** y, luego, elija Crear política.

Control del acceso a los periodos de mantenimiento mediante la AWS CLI

En los procedimientos siguientes, se explica cómo se usa la AWS Command Line Interface (AWS CLI) para crear los roles y los permisos necesarios para Maintenance Windows, una herramienta de AWS Systems Manager.

Temas

- Tarea 1: crear archivos de política de confianza y política administrada por el cliente en formato JSON
- Tarea 2: crear y verificar un rol de servicio personalizado para el periodo de mantenimiento mediante la AWS CLI
- Tarea 3: conceder permisos a usuarios específicos para que registren tareas del periodo de mantenimiento mediante la AWS CLI
- Tarea 4: impida que usuarios específicos registren las tareas de periodos de mantenimiento mediante la AWS CLI

Tarea 1: crear archivos de política de confianza y política administrada por el cliente en formato JSON

Las tareas de periodo de mantenimiento requieren un rol de IAM para proporcionar los permisos necesarios para su ejecución en los recursos de destino. Los permisos se conceden mediante una política de IAM asociada al rol. El contenido de esta política lo determinan los tipos de tareas que se ejecutan y el resto de requisitos operativos. Proporcionamos una política básica que se puede adaptar a sus necesidades. En función de las tareas y los tipos de tareas que ejecuten los periodos de mantenimiento, es posible que no necesite todos los permisos de esta política, o que tenga que incluir permisos adicionales.

En esta tarea, debe especificar los permisos necesarios para el rol del periodo de mantenimiento personalizado en un par de archivos JSON. Más tarde adjuntará esta política al rol que cree en Tarea 2: crear y verificar un rol de servicio personalizado para el periodo de mantenimiento mediante la AWS CLI.

Para crear archivos de política de confianza y política administrada por el cliente

1. Copie y pegue la siguiente política de confianza en un archivo de texto. Guarde este archivo con el siguiente nombre y extensión de archivo: **`mw-role-trust-policy.json`**.

   JSON

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

```
                "Service": "ssm.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

2. Copie y pegue la siguiente política de JSON en un archivo de texto diferente. En el mismo directorio en el que creó el primer archivo, guárdelo con el siguiente nombre y extensión de archivo: **`mw-role-custom-policy.json`**.

   JSON

```
{
    "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": "arn:aws:iam::111122223333:role/maintenance-window-
role-name",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
                        "ssm.amazonaws.com"
                    ]
                }
            }
        }
    ]
}
```

3. Modifique el contenido de `mw-role-custom-policy.json` de acuerdo con las necesidades de las tareas de mantenimiento que ejecute en la cuenta. Los cambios que realice son específicos de las operaciones planificadas.

   Por ejemplo:

   - Puede proporcionar nombres de recursos de Amazon (ARN) para funciones y máquinas de estado específicas en lugar de utilizar calificadores comodín (*).
   - Si no tiene previsto ejecutar tareas de AWS Step Functions, puede quitar los permisos y ARN de `states`.
   - Si no tiene previsto ejecutar tareas de AWS Lambda, puede quitar los permisos y ARN de `lambda`.
   - Si no tiene previsto ejecutar tareas de automatización, puede quitar los permisos `ssm:GetAutomationExecution` y `ssm:StartAutomationExecution`.
   - Agregue permisos adicionales que puedan ser necesarios para que se ejecuten las tareas. Por ejemplo, algunas acciones de Automation trabajan con pilas de AWS CloudFormation. Por lo tanto, los permisos `cloudformation:CreateStack`, `cloudformation:DescribeStacks` y `cloudformation:DeleteStack` son obligatorios.

     Otro ejemplo es el manual de procedimientos de Automation `AWS-CopySnapshot`, que requiere permisos para crear una instantánea de Amazon Elastic Block Store (Amazon EBS). Por lo tanto, el rol de servicio necesita los permisos `ec2:CreateSnapshot`.

     Para obtener información acerca de los permisos de rol que necesitan los manuales de procedimientos de automatización, consulte las descripciones de documentos en la Referencia de manuales de procedimientos de Automatización de AWS Systems Manager.

   Guarde el archivo de nuevo después de realizar los cambios necesarios.

Tarea 2: crear y verificar un rol de servicio personalizado para el periodo de mantenimiento mediante la AWS CLI

La política que se crea en la tarea anterior se asocia al rol de servicio del periodo de mantenimiento que se crea en esta tarea. Cuando los usuarios registran una tarea de periodo de mantenimiento, especifican este rol de IAM como parte de la configuración de la tarea. Los permisos de este rol autorizan a Systems Manager a ejecutar tareas en periodos de mantenimiento en su nombre.

> **Important**
>
> Antes, la consola de Systems Manager ofrecía la posibilidad de elegir el rol vinculado a servicio de IAM `AWSServiceRoleForAmazonSSM` administrado de AWS que utilizar como rol de mantenimiento para las tareas. Ya no se recomienda utilizar este rol y su política asociada, `AmazonSSMServiceRolePolicy`, para tareas de periodo de mantenimiento. Si está utilizando actualmente este rol para tareas de periodo de mantenimiento, le recomendamos que deje de hacerlo. En su lugar, cree su propio rol de IAM que permita la comunicación entre Systems Manager y otros Servicios de AWS cuando se ejecuten las tareas de periodo de mantenimiento.

En esta tarea, ejecute comandos de la CLI para crear su rol de servicio del periodo de mantenimiento mediante la adición del contenido de la política de los archivos JSON que creó.

Cree un rol de servicio personalizado para el periodo de mantenimiento mediante la AWS CLI

1. Abra la AWS CLI y ejecute el siguiente comando en el directorio en el que guardó `mw-role-custom-policy.json` y `mw-role-trust-policy.json`. El comando crea un rol de servicio del periodo de mantenimiento llamado `my-maintenance-window-role` y le asocia la política de confianza.

   Linux & macOS

   ```
   aws iam create-role \
       --role-name "my-maintenance-window-role" \
       --assume-role-policy-document file://mw-role-trust-policy.json
   ```

   Windows

   ```
   aws iam create-role ^
       --role-name "my-maintenance-window-role" ^
       --assume-role-policy-document file://mw-role-trust-policy.json
   ```

   El sistema devuelve información similar a la siguiente.

   ```
   {
       "Role": {
   ```

```
        "AssumeRolePolicyDocument": {
            "Version": "2012-10-17",
            "Statement": [
                {
                    "Action": "sts:AssumeRole",
                    "Effect": "Allow",
                    "Principal": {
                        "Service": "ssm.amazonaws.com"
                    }
                }
            ]
        },
        "RoleId": "AROAIIZKPBKS2LEXAMPLE",
        "CreateDate": "2024-08-19T03:40:17.373Z",
        "RoleName": "my-maintenance-window-role",
        "Path": "/",
        "Arn": "arn:aws:iam::123456789012:role/my-maintenance-window-role"
    }
}
```

> **Note**
>
> Anote los valores de `RoleName` y `Arn`. Los incluirá en el siguiente comando.

2. Ejecute el siguiente comando para asociar la política administrada por el cliente al rol. Reemplace el marcador de posición *`account-id`* con su propio ID de la Cuenta de AWS

   Linux & macOS

   ```
   aws iam attach-role-policy \
       --role-name "my-maintenance-window-role" \
       --policy-arn "arn:aws:iam::account-id:policy/mw-role-custom-policy.json"
   ```

   Windows

   ```
   aws iam attach-role-policy ^
       --role-name "my-maintenance-window-role" ^
       --policy-arn "arn:aws:iam::account-id:policy/mw-role-custom-policy.json"
   ```

3. Ejecute el siguiente comando para comprobar que el rol se ha creado y que la política de confianza se ha asociado.

```
aws iam get-role --role-name my-maintenance-window-role
```

El comando devuelve información similar a la siguiente:

```
{
    "Role": {
        "Path": "/",
        "RoleName": "my-maintenance-window-role",
        "RoleId": "AROA123456789EXAMPLE",
        "Arn": "arn:aws:iam::123456789012:role/my-maintenance-window-role",
        "CreateDate": "2024-08-19T14:13:32+00:00",
        "AssumeRolePolicyDocument": {
            "Version": "2012-10-17",
            "Statement": [
                {
                    "Effect": "Allow",
                    "Principal": {
                        "Service": "ssm.amazonaws.com"
                    },
                    "Action": "sts:AssumeRole"
                }
            ]
        },
        "MaxSessionDuration": 3600,
        "RoleLastUsed": {
            "LastUsedDate": "2024-08-19T14:30:44+00:00",
            "Region": "us-east-2"
        }
    }
}
```

4. Ejecute el siguiente comando para comprobar que la política administrada por el cliente se ha asociado al rol.

```
aws iam list-attached-role-policies --role-name my-maintenance-window-role
```

El comando devuelve información similar a la siguiente:

```
{
    "AttachedPolicies": [
        {
```

```
            "PolicyName": "mw-role-custom-policy",
            "PolicyArn": "arn:aws:iam::123456789012:policy/mw-role-custom-policy"
        }
    ]
}
```

Tarea 3: conceder permisos a usuarios específicos para que registren tareas del periodo de mantenimiento mediante la AWS CLI

Conceder a los usuarios permisos para acceder al rol de servicio del periodo de mantenimiento personalizado les permite utilizarlo con sus tareas de periodos de mantenimiento. Esto se suma a los permisos que ya se les haya concedido para utilizar los comandos de la API de Systems Manager para la herramienta de Maintenance Windows. Este rol de IAM transmite los permisos necesarios para ejecutar una tarea de periodo de mantenimiento. Como resultado, un usuario no puede registrar tareas con un periodo de mantenimiento mediante el rol de servicio personalizado sin la posibilidad de transferir esos permisos de IAM.

Cuando registra una tarea en un periodo de mantenimiento, debe especificar un rol de servicio para ejecutar las operaciones de la tarea real. Este es el rol que el servicio asume cuando ejecuta las tareas en su nombre. Antes de eso, para registrar la tarea en sí, asigne la política de IAM `PassRole` a una entidad de IAM (como un usuario o un grupo). Esto permite que la entidad de IAM especifique, como parte del registro de esas tareas con el periodo de mantenimiento, el rol que se debe usar al ejecutar tareas. Para obtener más información, consulte Concesión de permisos a un usuario para transferir un rol a un Servicio de AWS en la Guía del usuario de IAM.

Para configurar los permisos para usuarios con autorización para registrar tareas de periodos de mantenimiento mediante la AWS CLI

1. Copie y pegue la siguiente política de AWS Identity and Access Management (IAM) en un editor de texto y guárdela con el nombre y la extensión de archivo siguientes: `mw-passrole-policy.json`.

   JSON

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

```
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::111122223333:role/my-maintenance-
window-role"
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::111122223333:role/"
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::111122223333:role/aws-service-role/
ssm.amazonaws.com/"
        }
    ]
}
```

Reemplace *my-maintenance-window-role* con el nombre del rol de la ventana de mantenimiento personalizada que ha creado antes.

Reemplace *account-id* por el ID de su Cuenta de AWS. Al agregar este permiso para el recurso `arn:aws:iam::account-id:role/` se permite a los usuarios del grupo ver y elegir entre los roles de cliente en la consola cuando crean una tarea del periodo de mantenimiento. La adición de este permiso para `arn:aws:iam::account-id:role/aws-service-role/ssm.amazonaws.com/` permite a los usuarios del grupo elegir el rol vinculado al servicio de Systems Manager en la consola cuando crean una tarea del periodo de mantenimiento.

2. Abra la AWS CLI.
3. En función de si el permiso se está asignando a una entidad de IAM (usuario o grupo), ejecute uno de los siguientes comandos.

   - Para una entidad de IAM:

     Linux & macOS

     ```
     aws iam put-user-policy \
         --user-name "user-name" \
         --policy-name "policy-name" \
         --policy-document file://path-to-document
     ```

Windows

```
aws iam put-user-policy ^
    --user-name "user-name" ^
    --policy-name "policy-name" ^
    --policy-document file://path-to-document
```

En *user-name*, especifique el usuario que asigna tareas a periodos de mantenimiento. En *policy-name*, especifique el nombre que desea utilizar para identificar la política, por ejemplo **my-iam-passrole-policy**. En *path-to-document* (ruta-del-documento), especifique la ruta del archivo que guardó en el paso 1. Por ejemplo: `file://C:\Temp\mw-passrole-policy.json`

> **Note**
>
> Para conceder acceso a un usuario para que registre tareas en periodos de mantenimiento mediante la consola de Systems Manager, también debe asignar la política `AmazonSSMFullAccess` a su usuario (o una política de IAM que proporcione un pequeño conjunto de permisos de acceso para Systems Manager que cubra las tareas de periodos de mantenimiento). Ejecute el siguiente comando para asignar la política `AmazonSSMFullAccess` a su usuario.
>
> Linux & macOS
>
> ```
> aws iam attach-user-policy \
>     --policy-arn "arn:aws:iam::aws:policy/AmazonSSMFullAccess" \
>     --user-name "user-name"
> ```
>
> Windows
>
> ```
> aws iam attach-user-policy ^
>     --policy-arn "arn:aws:iam::aws:policy/AmazonSSMFullAccess" ^
>     --user-name "user-name"
> ```

- Para un grupo de IAM:

  Linux & macOS

  ```
  aws iam put-group-policy \
  ```

```
    --group-name "group-name" \
    --policy-name "policy-name" \
    --policy-document file://path-to-document
```

Windows

```
aws iam put-group-policy ^
    --group-name "group-name" ^
    --policy-name "policy-name" ^
    --policy-document file://path-to-document
```

En *group-name*, especifique el grupo cuyos miembros asignarán tareas de periodos de mantenimiento. En *policy-name*, especifique el nombre que desea utilizar para identificar la política, por ejemplo **my-iam-passrole-policy**. En *path-to-document* (ruta-del-documento), especifique la ruta del archivo que guardó en el paso 1. Por ejemplo: `file://C:\Temp\mw-passrole-policy.json`

> **Note**
>
> Para conceder acceso a los miembros de un grupo de modo que puedan registrar tareas de periodos de mantenimiento mediante la consola de Systems Manager, también debe asignar la política `AmazonSSMFullAccess` al grupo. Ejecute el siguiente comando para asignar esta política al grupo.
>
> Linux & macOS
>
> ```
> aws iam attach-group-policy \
>     --policy-arn "arn:aws:iam::aws:policy/AmazonSSMFullAccess" \
>     --group-name "group-name"
> ```
>
> Windows
>
> ```
> aws iam attach-group-policy ^
>     --policy-arn "arn:aws:iam::aws:policy/AmazonSSMFullAccess" ^
>     --group-name "group-name"
> ```

4. Ejecute el siguiente comando para verificar que la política se haya asignado al grupo.

Linux & macOS

```
aws iam list-group-policies \
    --group-name "group-name"
```

Windows

```
aws iam list-group-policies ^
    --group-name "group-name"
```

Tarea 4: impida que usuarios específicos registren las tareas de periodos de mantenimiento mediante la AWS CLI

Puede denegar el permiso `ssm:RegisterTaskWithMaintenanceWindow` a los usuarios de la Cuenta de AWS que no desee que registren tareas en periodos de mantenimiento. Esto proporciona una capa adicional de protección frente a usuarios que no deban registrar tareas de periodo de mantenimiento.

En función de si el permiso `ssm:RegisterTaskWithMaintenanceWindow` se deniega a un usuario individual o a un grupo, utilice uno de los siguientes procedimientos para evitar que los usuarios registren tareas con un periodo de mantenimiento.

Para configurar permisos para usuarios que no pueden registrar tareas de periodos de mantenimiento mediante la AWS CLI

1. Copie y pegue la siguiente política de IAM en un editor de texto y guárdela con el nombre y la extensión de archivo **`deny-mw-tasks-policy.json`**.

   JSON

   ```
   {
       "Version": "2012-10-17",
       "Statement": [
           {
               "Effect": "Deny",
               "Action": "ssm:RegisterTaskWithMaintenanceWindow",
               "Resource": "*"
           }
   ```

```
    ]
}
```

2. Abra la AWS CLI.
3. En función de si el permiso se está asignando a una entidad de IAM (usuario o grupo), ejecute uno de los siguientes comandos.

   - Para un usuario:

     Linux & macOS

     ```
     aws iam put-user-policy \
         --user-name "user-name" \
         --policy-name "policy-name" \
         --policy-document file://path-to-document
     ```

     Windows

     ```
     aws iam put-user-policy ^
         --user-name "user-name" ^
         --policy-name "policy-name" ^
         --policy-document file://path-to-document
     ```

     En *user-name*, especifique el usuario para que no pueda asignar tareas de periodos de mantenimiento. En *policy-name*, especifique el nombre que desea utilizar para identificar la política, por ejemplo **my-deny-mw-tasks-policy**. En *path-to-document* (ruta-del-documento), especifique la ruta del archivo que guardó en el paso 1. Por ejemplo: `file://C:\Temp\deny-mw-tasks-policy.json`

   - Para un grupo:

     Linux & macOS

     ```
     aws iam put-group-policy \
         --group-name "group-name" \
         --policy-name "policy-name" \
         --policy-document file://path-to-document
     ```

     Windows

     ```
     aws iam put-group-policy ^
         --group-name "group-name" ^
     ```

```
    --policy-name "policy-name" ^
    --policy-document file://path-to-document
```

En *group-name*, especifique el grupo para que sus miembros no puedan asignar tareas de periodos de mantenimiento. En *policy-name*, especifique el nombre que desea utilizar para identificar la política, por ejemplo **my-deny-mw-tasks-policy**. En *path-to-document* (ruta-del-documento), especifique la ruta del archivo que guardó en el paso 1. Por ejemplo: `file://C:\Temp\deny-mw-tasks-policy.json`

4. Ejecute el siguiente comando para verificar que la política se haya asignado al grupo.

   Linux & macOS

   ```
   aws iam list-group-policies \
       --group-name "group-name"
   ```

   Windows

   ```
   aws iam list-group-policies ^
       --group-name "group-name"
   ```

Crear y administrar periodos de mantenimiento mediante la consola

En esta sección se describe cómo crear, configurar, actualizar y eliminar periodos de mantenimiento con la consola de AWS Systems Manager. En esta sección también se ofrece información sobre la administración de los destinos y las tareas de un periodo de mantenimiento.

> **Important**
>
> Le recomendamos que inicialmente cree y configure períodos de mantenimiento en un entorno de pruebas.

Antes de empezar

Antes de crear un periodo de mantenimiento, debe configurar el acceso a Maintenance Windows, una herramienta de AWS Systems Manager. Para obtener más información, consulte Configuración de Maintenance Windows.

Temas

- Crear y configurar un periodo de mantenimiento mediante la consola
- Asignar destinos a un periodo de mantenimiento mediante la consola
- Asignar tareas a un periodo de mantenimiento mediante la consola
- Activar o desactivar un periodo de mantenimiento mediante la consola
- Actualizar o eliminar recursos de un periodo de mantenimiento mediante la consola

Crear y configurar un periodo de mantenimiento mediante la consola

En este procedimiento, creará un periodo de mantenimiento en Maintenance Windows, una herramienta de AWS Systems Manager. Puede especificar las opciones básicas, como el nombre, la programación y la duración. Más adelante, puede elegir los destinos, o los recursos, que se actualizan y las tareas que se llevan a cabo cuando se ejecuta el periodo de mantenimiento.

> **Note**
>
> Para ver una explicación de cómo se relacionan entre sí las distintas opciones relacionadas con la programación de los periodos de mantenimiento, consulte Programación de la ventana de mantenimiento y opciones de periodo activo.
> Para obtener más información acerca del uso de la opción `--schedule`, consulte Referencia: expresiones cron y rate para Systems Manager.

Para crear un periodo de mantenimiento mediante la consola

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Maintenance Windows.
3. Elija Create maintenance window (Crear periodo de mantenimiento).
4. En el campo Name (Nombre), ingrese un nombre descriptivo que lo ayude a identificar este periodo de mantenimiento.
5. (Opcional) En Description (Descripción), ingrese una descripción para identificar cómo se utilizará este periodo de mantenimiento.
6. (Opcional) Si desea permitir que se ejecute una tarea del periodo de mantenimiento en los nodos administrados, incluso si no ha registrado esos nodos como destinos, elija Allow unregistered targets (Permitir destinos no registrados).

Si elige esta opción, podrá elegir los nodos no registrados (por ID de nodo) al registrar una tarea con el periodo de mantenimiento.

Si no elige esta opción, deberá elegir los destinos registrados anteriormente cuando registre una tarea con el periodo de mantenimiento.

7. Especifique una programación para el período de mantenimiento usando una de las tres opciones de programación.

   Para obtener información acerca de cómo crear expresiones Cron y Rate, consulte Referencia: expresiones cron y rate para Systems Manager.

8. En Duration (Duración), escriba el número de horas que se ejecutará el periodo de mantenimiento. El valor que especifique determina la hora de finalización específica del periodo de mantenimiento en función de la hora de inicio. No se permite que las tareas del período de mantenimiento comiencen después de la hora de enlace resultante menos el número de horas que especifique para Stop initiating tasks (Dejar de iniciar tareas) en el siguiente paso.

   Por ejemplo, si el período de mantenimiento comienza a las 15:00 h, la duración es de tres horas y el valor de Stop initiating tasks (Dejar de iniciar tareas) es de una hora, no se pueden iniciar tareas del período de mantenimiento después de las 17:00 h.

9. En el campo Stop initiating tasks (Dejar de iniciar tareas), escriba el número de horas que el sistema debe considerar antes de que finalice el período de mantenimiento para dejar de programar nuevas tareas por ejecutar.

10. (Opcional) En Window start date (Fecha de inicio del periodo), especifique una fecha y un horario en formato extendido ISO-8601 para cuando desee que se active el periodo de mantenimiento. Esto le permite retrasar la activación del periodo de mantenimiento hasta la fecha futura especificada.

    > **Note**
    >
    > No puede especificar una fecha y hora de inicio que se produzcan en el pasado.

11. (Opcional) En Fecha de finalización del periodo, especifique una fecha y un horario en formato extendido ISO-8601 para cuando desee que se desactive el periodo de mantenimiento. Esto le permite establecer una fecha y hora en el futuro después de la cual el periodo de mantenimiento dejará de ejecutarse.

12. (Opcional) En Schedule timezone (Programar zona horaria), especifique la zona horaria que se utilizará como base para ejecutar los periodos de mantenimiento programados, en formato Internet Assigned Numbers Authority (IANA). Por ejemplo: "America/Los_Angeles", "etc/UTC" o "Asia/Seoul".

    Para obtener más información sobre los formatos válidos, consulte Time Zone Database en el sitio web de IANA.

13. (Opcional) En Schedule offset (Programar offset), ingrese la cantidad de días que se debe esperar después de la fecha y hora especificadas por una expresión cron o rate antes de ejecutar el periodo de mantenimiento. Puede especificar entre uno y seis días.

    > **Note**
    >
    > Esta opción solo está disponible si ha especificado una programación mediante el ingreso manual de una expresión cron o rate.

14. (Opcional) En el área Manage tags (Administrar etiquetas), aplique uno o varios pares de claves nombre/valor al periodo de mantenimiento.

    Las etiquetas son metadatos opcionales que usted asigna a un recurso. Las etiquetas permiten clasificar los recursos de diversas maneras, por ejemplo, según la finalidad, el propietario o el entorno. Por ejemplo, es posible que desee etiquetar un periodo de mantenimiento para identificar el tipo de tareas que ejecuta, los tipos de destinos y el entorno en el que se ejecuta. En este caso, puede especificar los siguientes pares de claves nombre-valor:

    - `Key=TaskType,Value=AgentUpdate`
    - `Key=OS,Value=Windows`
    - `Key=Environment,Value=Production`

15. Elija Create maintenance window (Crear periodo de mantenimiento). El sistema le devuelve a la página de periodo de mantenimiento. El estado del período de mantenimiento que acaba de crear es Enabled (Habilitado).

Asignar destinos a un periodo de mantenimiento mediante la consola

En este procedimiento, se registra un destino con un período de mantenimiento. En otras palabras, debe especificar los recursos en los que el periodo de mantenimiento realiza acciones.

> **Note**
>
> Si se registra una sola tarea del periodo de mantenimiento con varios destinos, las invocaciones de la tarea se producen de forma secuencial y no en paralelo. Si la tarea debe ejecutarse en varios destinos al mismo tiempo, registre una tarea para cada destino de forma individual y asigne a cada tarea el mismo nivel de prioridad.

Para asignar destinos a un periodo de mantenimiento mediante la consola

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Maintenance Windows.
3. En la lista de periodos de mantenimiento, elija el período de mantenimiento al que desee añadir destinos.
4. Elija Actions (Acciones) y, a continuación, elija Register targets (Registrar destinos).
5. (Opcional) Para Target Name (Nombre de destino), escriba un nombre para los destinos.
6. (Opcional) En Description (Descripción), introduzca una descripción.
7. (Opcional) En Owner information (Información del propietario), especifique información para incluir en cualquier evento de Amazon EventBridge que se genere mientras se ejecutan las tareas para estos destinos en este periodo de mantenimiento.

   Para obtener información acerca de cómo utilizar EventBridge para monitorear los eventos de Systems Manager, consulte Monitoreo de eventos de Systems Manager con Amazon EventBridge.
8. En el área de Destinos , elija una de las opciones que se describen en la siguiente tabla.

Opción	Descripción
Especifique las etiquetas de las instancias	En los cuadros Specify instance tags (Especificar etiquetas de instancia), especifiq ue una o más claves de etiquetas y valores (opcional) que se hayan agregado o que se agregarán a los nodos administrados en la cuenta. Cuando el periodo de mantenimiento se ejecuta, intenta realizar tareas en todos

Opción	Descripción
	los nodos administrados a los que estas etiquetas se han agregado. Si especifica más de una clave de etiqueta, será necesario etiquetar un nodo con todos los valores y las claves de etiqueta especific ados para que se incluyan en el grupo de destino.
Elegir las instancias manualmente	En la lista, seleccione la casilla para cada nodo que desea incluir en el destino del periodo de mantenimiento. La lista incluye todos los nodos en la cuenta que están configurados para su uso con Systems Manager. Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas. En el caso de los dispositivos de borde, servidores locales y máquinas virtuales, consulte Administrador de nodos en entornos híbridos y multinube con Systems Manager

Opción	Descripción
Elegir un grupo de recursos	En Grupo de recursos, elija el nombre de un grupo de recursos existente en su cuenta de la lista. Para obtener más información acerca de cómo crear y trabajar con grupos de recursos, consulte los siguientes temas: • ¿Qué son los grupos de recursos? en la Guía del usuario de AWS Resource Groups • Grupos de recursos y etiquetado para AWS en el Blog de noticias de AWS (Opcional) En Resource types (Tipos de recursos), seleccione hasta cinco tipos de recursos disponibles, o seleccione All resource types (Todos los tipos de recursos). Si las tareas que se asignan al periodo de mantenimiento no actúan en uno de los tipos de recursos que agregó al destino, el sistema podría informar un error. Las tareas para las que se encuentra un tipo de recurso compatible siguen ejecutándose a pesar de estos errores. Por ejemplo, suponga que añadir los siguientes tipos de recurso a este objetivo: • `AWS::S3::Bucket` • `AWS::DynamoDB::Table` • `AWS::EC2::Instance`

Opción	Descripción
	Sin embargo, más tarde, al agregar tareas para el periodo de mantenimiento, debe incluir solo tareas que realizan acciones en los nodos, como, por ejemplo, la aplicació n de una base de referencia de revisione s o reiniciar un nodo. En el registro del periodo de mantenimiento, es posible que se notifique un error si no se encuentra n buckets de Amazon Simple Storage Service (Amazon S3) o tablas de Amazon DynamoDB. Sin embargo, el periodo de mantenimiento sigue administrando tareas en los nodos del grupo de recursos.

9. Elija Register target (Registrar destino).

Si desea asignar más destinos a este período, elija la pestaña Destinos y, a continuación, elija Registrar nuevos destinos. Con esta opción, puede elegir una manera diferente de dirigirse a los destinos. Por ejemplo, si anteriormente se dirigía a los nodos por el ID de nodo, puede registrar nuevos destinos y dirigirse a los nodos especificando etiquetas aplicadas a los nodos administrados o la elección de tipos de recursos a partir de un grupo de recursos.

Asignar tareas a un periodo de mantenimiento mediante la consola

En este procedimiento, añada una tarea a un período de mantenimiento. Las tareas son las acciones que se realizan cuando se ejecuta un periodo de mantenimiento.

Se pueden añadir a un periodo de mantenimiento los siguientes cuatro tipos de tareas:

- Comandos Run Command de AWS Systems Manager
- Flujos de trabajo de Systems Manager Automation
- Tareas de AWS Step Functions
- Funciones de AWS Lambda

> **Important**
>
> La política de IAM para Maintenance Windows requiere que se agregue el prefijo `SSM` a los nombres de la función (o alias) de Lambda. Antes de continuar con el registro de este tipo de tareas, actualice el nombre en AWS Lambda para incluir `SSM`. Por ejemplo, si el nombre de la función de Lambda es `MyLambdaFunction`, cámbielo a `SSMMyLambdaFunction`.

Para asignar tareas a un período de mantenimiento

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Maintenance Windows.
3. En la lista de periodos de mantenimiento, elija un periodo de mantenimiento.
4. Elija Actions (Acciones) y, luego, elija la opción para el tipo de tarea que desea registrar con el periodo de mantenimiento.
   - Registrar tarea de Run Command
   - Registrar tarea de Automation.
   - Registrar tarea de Lambda
   - Registrar tarea de Step Functions

   > **Note**
   >
   > Las tareas de los periodos de mantenimiento solo admiten los flujos de trabajo de máquinas de estado estándar de Step Functions. No son compatibles con los flujos de trabajo de máquinas de estado rápidas. Para obtener información sobre los tipos de flujos de trabajo de máquinas de estado, consulte Flujos de trabajo estándar en comparación con flujos de trabajo rápidos en la Guía para desarrolladores de AWS Step Functions.
5. (Opcional) En Name (Nombre), ingrese el nombre de la tarea.
6. (Opcional) En Description (Descripción), introduzca una descripción.

7. Para New task invocation cutoff (Nuevo límite de invocación de tareas), si no desea que se inicie ninguna invocación de tareas nueva después de que se haya alcanzado el tiempo límite de la ventana de mantenimiento, elija Enabled (Habilitado).

   Cuando esta opción está des activada, la tarea continúa ejecutándose cuando se alcanza el tiempo límite e inicia invocaciones de tareas nuevas hasta que se completan.

   > **Note**
   >
   > El estado de las tareas que no se han completado al habilitar esta opción es `TIMED_OUT`.

8. Para este paso, sigue los subpasos del tipo de tarea seleccionado.

   Run Command

   1. En la lista de Documento de comando, elija el documento de comandos de Systems Manager (documento de SSM) que define las tareas que se ejecutan.
   2. En Document version (Versión de documento), seleccione la versión de documento que se utilizará.
   3. En Task priority (Prioridad de tarea), especifique una prioridad para esta tarea. Cero (`0`) es la prioridad más alta. Las tareas de un período de mantenimiento se programan por orden de prioridad; las tareas que tengan la misma prioridad se programan en paralelo.

   Automation

   1. En la lista Documento de Automatización, elija el manual de procedimientos de Automatización que defina las tareas que se ejecutarán.
   2. En Document version (Versión de documento), elija la versión del manual de procedimientos que se utilizará.
   3. En Task priority (Prioridad de tarea), especifique una prioridad para esta tarea. Cero (`0`) es la prioridad más alta. Las tareas de un período de mantenimiento se programan por orden de prioridad; las tareas que tengan la misma prioridad se programan en paralelo.

   Lambda

   1. En el área Parámetros de Lambda, elija una función de Lambda de la lista.

2. (Opcional) En Payload (Carga), Client Context (Contexto de cliente) o Qualifier (Calificador) proporcione el contenido que desee incluir.

> **Note**
>
> En algunos casos, puede usar un pseudoparámetro como parte de su valor `Payload`. Luego, al ejecutarse la tarea del periodo de mantenimiento, esta pasa los valores correctos en lugar de los marcadores de posición del pseudoparámetro. Para obtener más información, consulta Utilización de pseudoparámetros en el registro de las tareas del periodo de mantenimiento.

3. En Task priority (Prioridad de tarea), especifique una prioridad para esta tarea. Cero (`0`) es la prioridad más alta. Las tareas de un período de mantenimiento se programan por orden de prioridad; las tareas que tengan la misma prioridad se programan en paralelo.

Step Functions

1. En el área Parámetros de Step Functions, elija una máquina de estado de la lista.
2. (Opcional) Proporcione un nombre para la ejecución de la máquina de estado y cualquier contenido que desee incluir para Input (Entrada).

> **Note**
>
> En algunos casos, puede usar un pseudoparámetro como parte de su valor `Input`. Luego, al ejecutarse la tarea del periodo de mantenimiento, esta pasa los valores correctos en lugar de los marcadores de posición del pseudoparámetro. Para obtener más información, consulta Utilización de pseudoparámetros en el registro de las tareas del periodo de mantenimiento.

3. En Task priority (Prioridad de tarea), especifique una prioridad para esta tarea. Cero (`0`) es la prioridad más alta. Las tareas de un período de mantenimiento se programan por orden de prioridad; las tareas que tengan la misma prioridad se programan en paralelo.

9. En el área Targets (Destinos), elija una de las siguientes opciones:

- Selecting registered target groups (Selección de grupos de destino registrados): seleccione uno o más destinos del periodo de mantenimiento que haya registrado con el periodo de mantenimiento actual.

- Selecting unregistered targets (Selección de destinos no registrados): elija los recursos disponibles uno por uno como destinos para la tarea.

  Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

- Task target not required (Destino de tarea no requerido): los destinos para la tarea pueden ya estar especificados en otras funciones para todas las tareas, excepto para las tareas de tipo Run Command.

  Especifique uno o más destinos para las tareas de tipo Run Command del periodo de mantenimiento. Según la tarea, los destinos son opcionales para otros tipos de tarea de periodo de mantenimiento (Automation, AWS Lambda y AWS Step Functions). Para obtener más información acerca de la ejecución de tareas que no especifican destinos, consulte Registro de tareas del periodo de mantenimiento sin destinos.

  > **Note**
  >
  > En muchos casos, no es necesario especificar de forma explícita un destino para una tarea de automatización. Por ejemplo, suponga que crea una tarea de tipo Automation para actualizar una Amazon Machine Image (AMI) para Linux mediante el manual de procedimientos `AWS-UpdateLinuxAmi`. Cuando se ejecuta la tarea, la AMI se actualiza con los paquetes de distribución de Linux y el software de Amazon disponibles más recientes. Las instancias nuevas que se crearon a partir de la AMI ya tienen estas actualizaciones instaladas. Como el ID de la AMI que se actualizará se especifica en los parámetros de entrada del manual de procedimientos, no es necesario volver a especificar un destino en la tarea del periodo de mantenimiento.

10. Solo tareas de Automatización:

    En el área Input parameters (Parámetros de entrada), proporcione valores para cualquier parámetro obligatorio u opcional necesario para ejecutar la tarea.

    > **Note**
    >
    > En algunos casos, puede utilizar un pseudoparámetro para determinados valores de parámetros de entrada. Luego, al ejecutarse la tarea del periodo de mantenimiento, esta pasa los valores correctos en lugar de los marcadores de posición del pseudoparámetro.

Para obtener más información, consulta Utilización de pseudoparámetros en el registro de las tareas del periodo de mantenimiento.

11. En Rate control (Control de velocidad):

    - En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

      Note

      Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

    - En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

12. (Opcional) En Rol de servicio de IAM, seleccione un rol que proporcione permisos para que los asuma Systems Manager al ejecutar una tarea del periodo de mantenimiento.

    Si no especifica el ARN de un rol de servicio, Systems Manager usa un rol vinculado al servicio de su cuenta. Este rol no aparece en el menú desplegable. Si en su cuenta no existe ningún rol vinculado al servicio adecuado para Systems Manager, se crea cuando la tarea se registra correctamente.

    Note

    Para mejorar la seguridad, le recomendamos encarecidamente que cree una política y un rol de servicio personalizados para ejecutar las tareas del periodo de mantenimiento. La política se puede diseñar para proporcionar solo los permisos necesarios para las tareas específicas del periodo de mantenimiento. Para obtener más información, consulte Configuración de Maintenance Windows.

13. Solo tareas de Run Command:

(Opcional) En Output options (Opciones de salida), realice las siguientes acciones:

- Seleccione la casilla de verificación Enable writing to S3 (Habilitar escritura en S3) para guardar la salida del comando en un archivo. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.
- Seleccione la casilla de verificación CloudWatch output (Salida de CloudWatch) para escribir la salida completa en los Registros de Amazon CloudWatch. Ingrese el nombre de un grupo de registro de los Registros de CloudWatch.

> **Note**
>
> Los permisos que conceden la capacidad de escribir datos en un bucket de S3 o en Registros de CloudWatch son los del perfil de instancia asignado al nodo, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager. Además, si el bucket de S3 o el grupo de registro especificado se encuentra en una Cuenta de AWS diferente, verifique que el perfil de instancia asociado al nodo tenga los permisos necesarios para escribir en ese bucket.

14. Solo tareas de Run Command:

    En la sección Notificaciones de SNS, seleccione la casilla de verificación Habilitar notificaciones de SNS si desea recibir notificaciones sobre el estado de ejecución de los comandos.

    Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.

15. Solo tareas de Run Command:

    En el área Parameters (Parámetros), especifique los parámetros para el documento.

> **Note**
>
> En algunos casos, puede utilizar un pseudoparámetro para determinados valores de parámetros de entrada. Luego, al ejecutarse la tarea del periodo de mantenimiento, esta pasa los valores correctos en lugar de los marcadores de posición del pseudoparámetro.

Para obtener más información, consulta Utilización de pseudoparámetros en el registro de las tareas del periodo de mantenimiento.

16. Solo tareas de Automatización y Run Command:

    (Opcional) En el área de alarma de CloudWatch, en Nombre de alarma, elija una alarma de CloudWatch existente para aplicarla a su tarea de monitoreo.

    Tenga en cuenta que si la alarma se activa, la tarea se detiene.

    > **Note**
    >
    > Para adjuntar una alarma de CloudWatch a la tarea, la entidad principal de IAM que ejecuta esta última debe tener permiso para la acción `iam:createServiceLinkedRole`. Para obtener más información sobre las alarmas de CloudWatch, consulte Uso de alarmas de Amazon CloudWatch.

17. Elija uno de los siguientes tipos de trabajo según su tipo de tarea:

    - Registrar tarea de Run Command
    - Registrar tarea de Automation.
    - Registrar tarea de Lambda
    - Registrar tarea de Step Functions

Activar o desactivar un periodo de mantenimiento mediante la consola

Puede habilitar o deshabilitar un periodo de mantenimiento en Maintenance Windows, una herramienta de AWS Systems Manager. Puede seleccionar un periodo de mantenimiento a la vez para desactivar o activar el período de mantenimiento para que no se ejecute. También puede seleccionar varios o todos los periodos de mantenimiento para activarlos o desactivarlos.

Esta sección describe cómo actualizar o eliminar un periodo de mantenimiento, los destinos y las tareas con la consola de Systems Manager. Para obtener ejemplos de cómo hacerlo mediante la AWS Command Line Interface (AWS CLI), consulte Tutorial actualización de un periodo de mantenimiento mediante la AWS CLI.

Temas

- Desactivar un periodo de mantenimiento mediante la consola

- Activar un periodo de mantenimiento mediante la consola

Desactivar un periodo de mantenimiento mediante la consola

Puede desactivar un periodo de mantenimiento para pausar una tarea durante un período específico, y permanecerá disponible para volver a activarse más adelante.

Desactivación de un periodo de mantenimiento

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Maintenance Windows.
3. Seleccione uno o más periodos de mantenimiento mediante la casilla de verificación situada junto al periodo de mantenimiento que desea desactivar.
4. En el menú Acciones, elija Desactivar un periodo de mantenimiento. El sistema le pedirá que confirme sus acciones.

Activar un periodo de mantenimiento mediante la consola

Puede activar un periodo de mantenimiento para reanudar una tarea.

> **Note**
>
> Si el periodo de mantenimiento utiliza un programa de frecuencias y la fecha de inicio está establecida actualmente en una fecha y hora anteriores, se utilizan la fecha y hora actuales como fecha de inicio del periodo de mantenimiento. Puede cambiar la fecha de inicio del periodo de mantenimiento antes o después de activarla. Para obtener más información, consulta Actualizar o eliminar recursos de un periodo de mantenimiento mediante la consola.

Activación de un periodo de mantenimiento

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Maintenance Windows.
3. Marque la casilla de verificación situada junto al periodo de mantenimiento para activarla.

4. Seleccione Acciones, Habilitar periodo de mantenimiento. El sistema le pedirá que confirme sus acciones.

Actualizar o eliminar recursos de un periodo de mantenimiento mediante la consola

Puede actualizar o eliminar un periodo de mantenimiento Maintenance Windows, una herramienta de AWS Systems Manager. También puede actualizar o eliminar los destinos o las tareas de un período de mantenimiento. Si edita los detalles de un período de mantenimiento, podrá cambiar la programación, los destinos y las tareas. También puede especificar los nombres y las descripciones de los periodos, los destinos y las tareas, lo que le ayuda a entender mejor sus propósitos y hace que sea más sencillo administrar la cola de períodos.

Esta sección describe cómo actualizar o eliminar un periodo de mantenimiento, los destinos y las tareas con la consola de Systems Manager. Para obtener ejemplos de cómo hacerlo mediante la AWS Command Line Interface (AWS CLI), consulte Tutorial actualización de un periodo de mantenimiento mediante la AWS CLI.

Temas

- Actualizar o eliminar un periodo de mantenimiento mediante la consola
- Actualizar o eliminar destinos de un periodo de mantenimiento mediante la consola
- Actualizar o eliminar tareas de un periodo de mantenimiento mediante la consola

Actualizar o eliminar un periodo de mantenimiento mediante la consola

Puede actualizar un periodo de mantenimiento para cambiar su nombre, la descripción y la programación y si el periodo de mantenimiento debe permitir destinos no registrados.

Para actualizar o eliminar un período de mantenimiento

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Maintenance Windows.
3. Seleccione el botón junto al periodo de mantenimiento que desea actualizar o eliminar y, luego, lleve a cabo alguna de las siguientes operaciones:
   - Elija Eliminar. El sistema le pedirá que confirme sus acciones.

- Elija Editar. En la página Edit maintenance window (Editar periodo de mantenimiento), cambie los valores y las opciones que desee y, a continuación, elija Save changes (Guardar cambios).

  Para obtener más información acerca de las opciones de configuración que puede realizar, consulte Crear y configurar un periodo de mantenimiento mediante la consola.

Actualizar o eliminar destinos de un periodo de mantenimiento mediante la consola

Puede actualizar o eliminar los destinos de un periodo de mantenimiento. Si decide actualizar un destino del período de mantenimiento, podrá especificar un nombre de destino, una descripción y un propietario nuevos. También puede elegir destinos diferentes.

Para actualizar o eliminar los destinos de un período de mantenimiento

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Maintenance Windows.
3. Elija el nombre del periodo de mantenimiento que desea actualizar, elija la pestaña Targets (Destinos) y, a continuación, realice las siguientes acciones:

   - Para actualizar los destinos, seleccione el botón que aparece junto al destino que desea actualizar y, a continuación, elija Edit (Editar).
   - Para anular el registro de destinos, seleccione el botón junto al destino que desea eliminar y, luego, elija Deregister target (Anular registro de destinos). En el cuadro de diálogo Deregister maintenance windows target (Anular registro de destinos de periodos de mantenimiento), elija Deregister (Anular registro).

Actualizar o eliminar tareas de un periodo de mantenimiento mediante la consola

Puede actualizar o eliminar los destinos de un periodo de mantenimiento. Si decide actualizar, podrá especificar un nombre de tarea, una descripción y un propietario nuevos. En las tareas de Run Command y de Automation, puede elegir un documento de SSM diferente para las tareas. No puede, sin embargo, editar una tarea para cambiar su tipo. Por ejemplo, si creó una tarea de Automation, no puede editar dicha tarea y cambiarla a una tarea de Run Command.

Para actualizar o eliminar tareas de un periodo de mantenimiento mediante la consola

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Maintenance Windows.
3. Elija el nombre del periodo de mantenimiento que desea actualizar.
4. Elija la pestaña Tasks (Tareas) y, a continuación, seleccione el botón que aparece junto a la tarea que desea actualizar.
5. Realice una de las siguientes acciones:
   - Para anular el registro de una tarea, elija Deregister task (Anular registro de tarea).
   - Elija Edit (Editar) para editar la tarea. Cambie los valores y las opciones que desee y, a continuación, elija Edit task (Editar tarea).

Tutoriales

En los tutoriales en esta sección aprenderá a realizar tareas comunes al trabajar con periodos de mantenimiento.

Completar los requisitos previos

Antes de intentar realizar estos tutoriales, complete los siguientes requisitos previos:

- Configure la AWS CLI en su equipo local: para poder ejecutar comandos de la AWS CLI, antes debe instalar y configurar la CLI en su equipo local. Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para PowerShell.
- Verifique los roles y los permisos del periodo de mantenimiento: un administrador de AWS de su cuenta debe concederle los permisos de AWS Identity and Access Management (IAM) que necesita para administrar los periodos de mantenimiento con la CLI. Para obtener más información, consulta Configuración de Maintenance Windows.
- Cree o configure una instancia compatible con Systems Manager: para completar los tutoriales, necesita al menos una instancia de Amazon Elastic Compute Cloud (Amazon EC2) que esté configurada para utilizarse con Systems Manager. Esto significa que el SSM Agent está instalado en la instancia y que hay un perfil de instancias de IAM para Systems Manager adjuntado a la instancia.

Se recomienda lanzar una instancia desde una Amazon Machine Image (AMI) administrada de AWS con el agente preinstalado. Para obtener más información, consulte Búsqueda de AMIs con SSM Agent preinstalado.

Para obtener más información sobre cómo instalar el SSM Agent en una instancia, consulte los siguientes temas:

- Cómo instalar y desinstalar de forma manual SSM Agent en instancias de EC2 para Windows Server
- Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux

Para obtener información sobre cómo configurar permisos de IAM para Systems Manager en su instancia, consulte Configuración de permisos de instancia requeridos para Systems Manager.

- Cree recursos adicionales según sea necesario: Run Command, una herramienta de Systems Manager, incluye muchas tareas que no requieren la creación de otros recursos además de los enumerados en este tema de requisitos previos. Por este motivo, proporcionamos una tarea Run Command sencilla para que pueda utilizarla la primera vez en los tutoriales. También necesita una instancia de EC2 que esté configurada para utilizarse con Systems Manager, como se describió antes en este tema. Después de configurar dicha instancia, puede registrar una tarea de Run Command sencilla.

  La herramienta Maintenance Windows de Systems Manager admite la ejecución de los siguientes cuatro tipos de tareas:

  - Comandos de la Run Command
  - Flujos de trabajo de Systems Manager Automation
  - Funciones de AWS Lambda
  - Tareas de AWS Step Functions

  En general, si una tarea de periodo de mantenimiento que desee ejecutar requiere recursos adicionales, debe crearlos antes. Por ejemplo, si desea un periodo de mantenimiento que ejecute una función AWS Lambda, cree la función de Lambda antes de comenzar; para una tarea de Run Command, cree el bucket de S3 en el que puede guardar la salida del comando (si planea hacerlo); y así sucesivamente.

Tutoriales

- Tutoriales: creación y administración de periodos de mantenimiento mediante la AWS CLI

- Tutorial: creación de un periodo de mantenimiento para la aplicación de revisiones mediante la consola

Tutoriales: creación y administración de periodos de mantenimiento mediante la AWS CLI

En esta sección se incluyen varios tutoriales para aprender a utilizar la AWS Command Line Interface (AWS CLI) para hacer lo siguiente:

- Crear y configurar un periodo de mantenimiento
- Ver información sobre un periodo de mantenimiento
- Ver información sobre las tareas y las ejecuciones de tareas de periodos de mantenimiento
- Actualizar un periodo de mantenimiento
- Eliminar un periodo de mantenimiento

Realizar un seguimiento de los ID de los recursos

A medida que complete las tareas en estos tutoriales de la AWS CLI, realice un seguimiento de los ID de recurso generados por los comandos que ejecuta. Puede utilizar muchos de estos como entrada de comandos posteriores. Por ejemplo, al crear el periodo de mantenimiento, el sistema le proporciona un ID de periodo de mantenimiento en el siguiente formato:

```
{
   "WindowId":"mw-0c50858d01EXAMPLE"
}
```

Anote los siguientes ID generados por el sistema, ya que los tutoriales de esta sección los utilizan:

- `WindowId`
- `WindowTargetId`
- `WindowTaskId`
- `WindowExecutionId`
- `TaskExecutionId`
- `InvocationId`
- `ExecutionId`

También necesita el ID de la instancia de EC2 que planea usar en los tutoriales. Por ejemplo: `i-02573cafcfEXAMPLE`.

Tutoriales

- Tutorial: cree y configure un periodo de mantenimiento mediante la AWS CLI
- Tutorial: vea información sobre periodos de mantenimiento mediante la AWS CLI
- Tutorial: visualización de información sobre tareas y ejecuciones de tareas mediante la AWS CLI
- Tutorial actualización de un periodo de mantenimiento mediante la AWS CLI
- Tutorial: eliminación de un periodo de mantenimiento mediante la AWS CLI

Tutorial: cree y configure un periodo de mantenimiento mediante la AWS CLI

En este tutorial se muestra cómo utilizar la AWS Command Line Interface (AWS CLI) para crear y configurar un periodo de mantenimiento, sus destinos y sus tareas. La ruta principal del tutorial se compone de pasos sencillos. Cree un único periodo de mantenimiento, identifique un solo destino y configure una tarea simple para ejecutar en el periodo de mantenimiento. Durante el proceso, proporcionamos información que puede utilizar para probar situaciones más complicadas.

A medida que siga los pasos que se indican en este tutorial, reemplace los valores en *`rojo`* y cursiva por sus propias opciones y sus ID. Por ejemplo, reemplace el ID del periodo de mantenimiento *`mw-0c50858d01EJEMPLO`* y el ID de la instancia *`i-02573cafcfEJEMPLO`* por los ID de los recursos que usted cree.

Contenido

- Paso 1: cree el periodo de mantenimiento mediante la AWS CLI
- Paso 2: registre un nodo de destino con el periodo de mantenimiento mediante la AWS CLI
- Paso 3: registre una tarea con el periodo de mantenimiento mediante la AWS CLI

Paso 1: cree el periodo de mantenimiento mediante la AWS CLI

En este paso va a crear un periodo de mantenimiento y va a especificar sus opciones básicas, como, por ejemplo, el nombre, la programación y la duración. Más adelante puede seleccionar la instancia que actualizará y la tarea que ejecutará.

En el ejemplo, se crea un periodo de mantenimiento que se ejecuta cada cinco minutos. Normalmente, un periodo de mantenimiento no se ejecutaría con esta frecuencia. Sin embargo, con

esta frecuencia puede ver los resultados del tutorial rápidamente. Mostraremos cómo cambiar a una frecuencia inferior una vez que la tarea se haya ejecutado de forma correcta.

> **Note**
>
> Para ver una explicación de cómo se relacionan entre sí las distintas opciones relacionadas con la programación de los periodos de mantenimiento, consulte Programación de la ventana de mantenimiento y opciones de periodo activo.
> Para obtener más información acerca del uso de la opción `--schedule`, consulte Referencia: expresiones cron y rate para Systems Manager.

Creación de un periodo de mantenimiento mediante la AWS CLI

1. Abra la AWS Command Line Interface (AWS CLI) y ejecute el siguiente comando en su máquina local para crear un periodo de mantenimiento que haga lo siguiente:

   - Se ejecuta cada cinco minutos durante un máximo de dos horas (según sea necesario).
   - Impide que se inicien nuevas tareas en el plazo de una hora desde la finalización de la operación del periodo de mantenimiento.
   - Permite destinos no asociados (instancias que no ha registrado en el periodo de mantenimiento).
   - Indica mediante el uso de etiquetas personalizadas que su creador piensa utilizarlo en un tutorial.

   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"
```

El sistema devuelve información similar a la siguiente.

```
{
   "WindowId":"mw-0c50858d01EXAMPLE"
}
```

2. A continuación, ejecute el siguiente comando para ver los detalles de esto y cualquier otro periodo de mantenimiento ya en su cuenta.

```
aws ssm describe-maintenance-windows
```

El sistema devuelve información similar a la siguiente.

```
{
   "WindowIdentities":[
      {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "My-First-Maintenance-Window",
            "Enabled": true,
            "Duration": 2,
            "Cutoff": 1,
            "NextExecutionTime": "2019-05-11T16:46:16.991Z"
      }
   ]
}
```

Siga en Paso 2: registre un nodo de destino con el periodo de mantenimiento mediante la AWS CLI.

Paso 2: registre un nodo de destino con el periodo de mantenimiento mediante la AWS CLI

En este paso, debe registrar un destino con el nuevo periodo de mantenimiento. En este caso, debe especificar qué nodo actualizar cuando se ejecuta el periodo de mantenimiento.

Para ver un ejemplo de registrar más de un nodo al mismo tiempo con ID de nodos, ejemplos de uso de etiquetas para identificar varios nodos, y ejemplos de especificar los grupos de recursos como destinos, consulte Ejemplos: registrar destinos con un periodo de mantenimiento.

> **Note**
>
> Ya debería haber creado una instancia de Amazon Elastic Compute Cloud (Amazon EC2) para utilizarla en este paso, como se describe en los requisitos previos para el tutorial de Maintenance Windows.

Para registrar un nodo de destino con el periodo de mantenimiento mediante la AWS CLI

1. Ejecute el siguiente comando en el equipo local. Reemplace cada *example resource placeholder* con su propia información.

   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"
   ```

   El sistema devuelve información similar a la siguiente.

   ```
   {
      "WindowTargetId":"e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
   }
   ```

2. Ahora, ejecute el siguiente comando en su máquina local para ver información detallada sobre el periodo de mantenimiento de destino.

Linux & macOS

```
aws ssm describe-maintenance-window-targets \
    --window-id "mw-0c50858d01EXAMPLE"
```

Windows

```
aws ssm describe-maintenance-window-targets ^
    --window-id "mw-0c50858d01EXAMPLE"
```

El sistema devuelve información similar a la siguiente.

```
{
    "Targets": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE",
            "ResourceType": "INSTANCE",
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-02573cafcfEXAMPLE"
                    ]
                }
            ]
        }
    ]
}
```

Siga en Paso 3: registre una tarea con el periodo de mantenimiento mediante la AWS CLI.

Ejemplos: registrar destinos con un periodo de mantenimiento

Puede registrar un solo nodo como destino utilizando su ID de nodo, tal y como se indica en Paso 2: registre un nodo de destino con el periodo de mantenimiento mediante la AWS CLI. También puede registrar uno o varios nodos como destinos utilizando los formatos de comandos de esta página.

En general, hay dos métodos para identificar los nodos que desea utilizar como destinos de periodo de mantenimiento: especificar nodos individuales y utilizar etiquetas de recursos. El método de etiquetas de recursos proporciona más opciones, tal y como se muestra en ejemplos 2-3.

También puede especificar uno o varios grupos de recursos como el destino de un periodo de mantenimiento. Un grupo de recursos puede incluir nodos y muchos otros tipos de recursos de AWS compatibles. Ejemplos 4 y 5, a continuación, muestran cómo agregar los grupos de recursos para el periodo de mantenimiento de los destinos.

> **Note**
>
> Si se registra una sola tarea del periodo de mantenimiento con varios destinos, las invocaciones de la tarea se producen de forma secuencial y no en paralelo. Si la tarea debe ejecutarse en varios destinos al mismo tiempo, registre una tarea para cada destino de forma individual y asigne a cada tarea el mismo nivel de prioridad.

Para obtener más información acerca de cómo se crean y administran los grupos de recursos, consulte ¿Qué son los grupos de recurso? en la Guía del usuario de AWS Resource Groups y Resource Groups y etiquetado para AWS en el Blog de noticias de AWS.

Para obtener información sobre las cuotas de Maintenance Windows, una herramienta de AWS Systems Manager, además de las especificadas en los siguientes ejemplos, consulte Systems Manager service quotas en la Referencia general de Amazon Web Services.

Ejemplo 1: registrar varios destinos utilizando ID de nodos

Ejecute el siguiente comando en el formato de su equipo local para registrar varios nodos como destinos mediante sus ID de nodos. Reemplace cada *example resource placeholder* con su propia información.

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
```

Uso recomendado: de máxima utilidad al registrar un grupo de nodos único con cualquier periodo de mantenimiento por primera vez y que no comparten una etiqueta de nodo común.

Cuotas: puede especificar hasta 50 nodos en total para el destino de cada periodo de mantenimiento.

Ejemplo 2: registrar destinos con etiquetas de recursos aplicadas a nodos

Ejecute el siguiente comando en su equipo local para registrar nodos que ya están etiquetados con un par de valor de clave que usted adjuntó. Reemplace cada *example resource placeholder* con su propia información.

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"
```

Uso recomendado: de máxima utilidad al registrar un grupo de nodos único con cualquier periodo de mantenimiento por primera vez y que sí comparten una etiqueta de nodo común.

Cuotas: puede especificar un total de hasta cinco pares clave-valor en cada destino. Si especifica varios pares de valor de clave, será necesario etiquetar un nodo con todos los valores y las claves de etiqueta especificados para que se incluyan en el grupo de destino.

> **Note**
>
> Puede etiquetar un grupo de instancias con la etiqueta-clave `Patch Group` o `PatchGroup` y asignar a las instancias un valor de clave común, como `my-patch-group`. (Debe utilizar `PatchGroup`, sin espacio, si ha permitido las etiquetas en los metadatos de las instancias de EC2). Patch Manager, una herramienta de Systems Manager, evalúa la clave `Patch Group` o `PatchGroup` en los nodos para ayudar a determinar qué línea de base de revisiones se aplica a ellos. Si su tarea va a ejecutar el documento de SSM `AWS-RunPatchBaseline` (o el documento de SSM antiguo `AWS-ApplyPatchBaseline`), puede especificar el mismo par de clave-valor `Patch Group` o `PatchGroup` al registrar destinos con un periodo de mantenimiento. Por ejemplo: `--target "Key=tag:PatchGroup,Values=`*`my-patch-group`*. Esto permite utilizar un periodo de mantenimiento para actualizar revisiones en un grupo de nodos que ya están asociados a la misma base de referencia de revisiones. Para obtener más información, consulte Grupos de revisiones.

Ejemplo 3: registrar destinos usando un grupo de claves de etiquetas (sin valores de etiqueta)

Ejecute el siguiente comando en su máquina local para registrar nodos que tienen una o más claves asignadas, independientemente de los valores de la clave. Reemplace cada *`example resource placeholder`* con su propia información.

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"
```

Uso recomendado: de máxima utilidad cuando se desea registrar nodos de destino especificando varias claves de etiqueta (sin sus valores) en lugar de solo un par etiqueta-clave o un par clave-valor.

Cuotas: puede especificar un total de hasta cinco pares etiqueta-clave para cada destino. Si especifica más de una clave de etiqueta, será necesario etiquetar un nodo con todas las claves de etiqueta especificadas para que se incluyan en el grupo de destino.

Ejemplo 4: registro de destinos con un nombre de grupo de recursos

Ejecute el siguiente comando en su máquina local para registrar un grupo de recursos especificado, independientemente del tipo de recursos que contiene. Reemplace *mw-0c50858d01EXAMPLE* con su propia información. Si las tareas que se asignan al periodo de mantenimiento no actúan en un tipo de recurso incluido en este grupo de recursos, el sistema podría informar un error. Las tareas para las que se encuentra un tipo de recurso compatible siguen ejecutándose a pesar de estos errores.

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"
```

Uso recomendado: Útil cuando desea especificar rápidamente un grupo de recursos como destino sin evaluar si todos sus tipos de recursos se destinará por un periodo de mantenimiento, o cuando se sabe que el grupo de recursos contiene únicamente los tipos de recursos que realizar acciones en sus tareas.

Cuotas: puede especificar un único grupo de recursos como destino.

Ejemplo 5: registro de destinos mediante el filtrado de tipos de recursos en un grupo de recursos

Ejecute el siguiente comando en su máquina local para registrar únicamente determinados tipos de recursos que pertenecen a un grupo de recursos que especifique. Reemplace *mw-0c50858d01EXAMPLE* con su propia información. Con esta opción, incluso si añade una tarea para un tipo de recurso que pertenece al grupo de recursos, la tarea no se ejecutará si no lo ha añadido explícitamente el tipo de recurso para el filtro.

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"
```

Uso recomendado: útil cuando desea mantener un control estricto sobre los tipos de recursos de AWS en los que el periodo de mantenimiento puede ejecutar acciones, o cuando el grupo de recursos contiene un gran número de tipos de recursos y desea evitar los informes de error innecesarios en los registros del periodo de mantenimiento.

Cuotas: puede especificar un único grupo de recursos como destino.

Paso 3: registre una tarea con el periodo de mantenimiento mediante la AWS CLI

En este paso del tutorial, registrará una tarea de AWS Systems Manager Run Command que ejecuta el comando `df` en su instancia de Amazon Elastic Compute Cloud (Amazon EC2) para Linux. Los resultados de este comando de Linux estándar muestran la cantidad de espacio libre y de espacio que se utiliza en el sistema de archivos de disco de la instancia.

-o bien-

Si tiene una instancia de Amazon EC2 para Windows Server en lugar de Linux, reemplace df en el siguiente comando por ipconfig. La salida de este comando muestra los detalles de la dirección IP, la máscara de subred y la gateway predeterminada para los adaptadores en la instancia de destino.

Cuando esté listo para registrar otros tipos de tareas o utilizar otras opciones disponibles de Systems Manager Run Command, consulte Ejemplos: registrar tareas en un periodo de mantenimiento. En este apartado se proporciona más información sobre los cuatro tipos de tarea

y algunas de sus opciones más importantes, con el fin de ayudarle a planificar escenarios más extensos y reales.

Para registrar una tarea en un periodo de mantenimiento

1. Ejecute el siguiente comando en el equipo local. Reemplace cada *example resource placeholder* con su propia información. La versión para ejecutar desde un equipo local con Windows incluye los caracteres de escape ("/") necesarios para ejecutar el comando desde la herramienta de línea de comandos.

   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\"]}}}
   ```

   El sistema devuelve información similar a la siguiente:

   ```
   {
       "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
   }
   ```

2. Ahora ejecute el siguiente comando para ver información detallada sobre la tarea del periodo de mantenimiento que ha creado.

   Linux & macOS

   ```
   aws ssm describe-maintenance-window-tasks \
       --window-id mw-0c50858d01EXAMPLE
   ```

   Windows

   ```
   aws ssm describe-maintenance-window-tasks ^
       --window-id mw-0c50858d01EXAMPLE
   ```

3. El sistema devuelve información similar a la siguiente.

   ```
   {
       "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. Espere hasta que la tarea haya tenido tiempo de ejecutarse, según la programación que especificó en Paso 1: cree el periodo de mantenimiento mediante la AWS CLI. Por ejemplo, si

especificó **`--schedule "rate(5 minutes)"`**, espere cinco minutos. A continuación, ejecute el siguiente comando para ver información sobre las ejecuciones que han tenido lugar para esta tarea.

Linux & macOS

```
aws ssm describe-maintenance-window-executions \
    --window-id mw-0c50858d01EXAMPLE
```

Windows

```
aws ssm describe-maintenance-window-executions ^
    --window-id mw-0c50858d01EXAMPLE
```

El sistema devuelve información similar a la siguiente.

```
{
    "WindowExecutions": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowExecutionId": "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE",
            "Status": "SUCCESS",
            "StartTime": 1557593493.096,
            "EndTime": 1557593498.611
        }
    ]
}
```

Tip

Una vez que la tarea se ejecuta correctamente, puede reducir la frecuencia de ejecución del periodo de mantenimiento. Por ejemplo, ejecute el siguiente comando para reducir la frecuencia a una vez a la semana. Reemplace *mw-0c50858d01EXAMPLE* con su propia información.

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)"
```

Para obtener información sobre cómo administrar las programaciones de los periodos de mantenimiento, consulte Referencia: expresiones cron y rate para Systems Manager y Programación de la ventana de mantenimiento y opciones de periodo activo.
Para obtener información acerca de cómo utilizar la AWS Command Line Interface (AWS CLI) para modificar un periodo de mantenimiento, consulte Tutorial actualización de un periodo de mantenimiento mediante la AWS CLI.

Para practicar la ejecución de comandos de la AWS CLI a fin de ver más detalles sobre la tarea del periodo de mantenimiento y sus ejecuciones, vaya a Tutorial: visualización de información sobre tareas y ejecuciones de tareas mediante la AWS CLI.

Acceder al resultado del comando del tutorial

Está fuera del alcance de este tutorial utilizar la AWS CLI para ver el resultado del comando de Run Command asociado a las ejecuciones de la tarea del periodo de mantenimiento.

No obstante, puede ver estos datos mediante la AWS CLI. (También puede ver la salida en la consola de Systems Manager o en un archivo de registros almacenado en un bucket de Amazon Simple Storage Service (Amazon S3), en caso de haber configurado el periodo de mantenimiento para almacenar las salidas de los comandos allí). Descubrirá que la salida del comando df en una instancia de EC2 para Linux es similar a lo siguiente.

```
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% /
```

El resultado del comando ipconfig en una instancia de EC2 para Windows Server es similar al siguiente:

```
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  . :
```

Ejemplos: registrar tareas en un periodo de mantenimiento

Puede registrar una tarea en Run Command, una herramienta de AWS Systems Manager, con un periodo de mantenimiento mediante la AWS Command Line Interface (AWS CLI), como se muestra en Registrar tareas en un periodo de mantenimiento. También puede registrar tareas para flujos de trabajo de Systems Manager Automation, funciones de AWS Lambda y tareas de AWS Step Functions, como se muestra más adelante en este tema.

> **Note**
>
> Especifique uno o más destinos para las tareas de tipo Run Command del periodo de mantenimiento. Según la tarea, los destinos son opcionales para otros tipos de tarea de periodo de mantenimiento (Automation, AWS Lambda y AWS Step Functions). Para obtener más información acerca de la ejecución de tareas que no especifican destinos, consulte Registro de tareas del periodo de mantenimiento sin destinos.

En este tema, se proporcionan ejemplos de cómo utilizar el comando `register-task-with-maintenance-window` de la AWS Command Line Interface (AWS CLI) para registrar cada uno de los cuatro tipos de tareas compatibles en un periodo de mantenimiento. Los ejemplos solo tienen fines de demostración, pero puede modificarlos para crear los comandos de registro de las tareas en uso.

Uso de la opción --cli-input-json

Para administrar mejor las opciones de la tarea, puede utilizar la opción de comando `--cli-input-json`, con valores de opción a los que se hace referencia en un archivo JSON.

Para utilizar el contenido del archivo JSON de ejemplo que proporcionamos en los siguientes ejemplos, haga lo siguiente en su equipo local:

1. Cree un archivo con un nombre como, por ejemplo, `MyRunCommandTask.json`, `MyAutomationTask.json` o cualquier otro nombre de su preferencia.
2. Copie el contenido de la muestra de JSON en el archivo.
3. Modifique el contenido del archivo para el registro de la tarea y, a continuación, guarde el archivo.
4. En el mismo directorio en el que almacenó el archivo, ejecute el comando siguiente. Sustituya el nombre de archivo por *`MiArchivo.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
```

Pseudoparámetros en las tareas de periodos de mantenimiento

En algunos ejemplos, utilizamos los pseudoparámetros como método para transferir información del ID a sus tareas. Por ejemplo, `{{TARGET_ID}}` y `{{RESOURCE_ID}}` se pueden utilizar para pasar los ID de los recursos de AWS a tareas de Automation, Lambda y Step Functions. Para obtener más información sobre los pseudoparámetros en el contenido `--task-invocation-parameters`, consulte Utilización de pseudoparámetros en el registro de las tareas del periodo de mantenimiento.

Más información

- Opciones de parámetros para el comando register-task-with-maintenance-windows.
- register-task-with-maintenance-window en la Referencia de comandos de la AWS CLI
- RegisterTaskWithMaintenanceWindow en la Referencia de la API de AWS Systems Manager

Ejemplos de registro de tareas

En las secciones siguientes se proporciona un comando de la AWS CLI de ejemplo para registrar un tipo de tarea admitido y una muestra de JSON que se puede utilizar con la opción `--cli-input-json`.

Registrar una tarea de Run Command de Systems Manager

Los siguientes ejemplos muestran cómo registrar tareas de Systems Manager Run Command con un periodo de mantenimiento mediante la 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\"]}}}"
```

Contenido JSON para usar con la opción de archivo **`--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": "amzn-s3-demo-bucket",
            "OutputS3KeyPrefix": "S3-PREFIX",
            "TimeoutSeconds": 3600
        }
    }
}
```

Registrar una tarea de Systems Manager Automation

Los siguientes ejemplos muestran cómo registrar tareas de Systems Manager Automation con un periodo de mantenimiento mediante la AWS CLI:

Comando de la AWS CLI:

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"
```

Contenido JSON para usar con la opción de archivo **`--cli-input-json`**:

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
        "TaskArn": "AWS-PatchInstanceWithRollback",
    "TaskType": "AUTOMATION","TaskInvocationParameters": {
        "Automation": {
            "DocumentVersion": "1",
            "Parameters": {
                "instanceId": [
                    "{{RESOURCE_ID}}"
                ]
            }
        }
    }
}
```

Registrar una tarea de AWS Lambda

Los siguientes ejemplos muestran cómo registrar las tareas de la función de Lambda con un periodo de mantenimiento mediante la AWS CLI.

En estos ejemplos, el usuario que creó la función de Lambda la nombró `SSMrestart-my-instances` y creó dos parámetros denominados `instanceId` y `targetType`.

> **Important**
>
> La política de IAM para Maintenance Windows requiere que se agregue el prefijo `SSM` a los nombres de la función (o alias) de Lambda. Antes de continuar con el registro de este tipo de tareas, actualice el nombre en AWS Lambda para incluir `SSM`. Por ejemplo, si el nombre de la función de Lambda es `MyLambdaFunction`, cámbielo a `SSMMyLambdaFunction`.

Comando de la AWS CLI:

Linux & macOS

> **Important**
>
> Si está utilizando la versión 2 de la AWS CLI, debe incluir la opción `--cli-binary-format raw-in-base64-out` en el siguiente comando si la carga de Lambda no

está codificada en base64. La opción `cli_binary_format` solo está disponible en la versión 2. Para obtener información acerca de esta y otras configuraciones del archivo `config` de la AWS CLI, consulte Configuraciones del archivo `config` admitidas en la Guía del usuario de AWS Command Line Interface.

```
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 está utilizando la versión 2 de la AWS CLI, debe incluir la opción `--cli-binary-format raw-in-base64-out` en el siguiente comando si la carga de Lambda no está codificada en base64. La opción `cli_binary_format` solo está disponible en la versión 2. Para obtener información acerca de esta y otras configuraciones del archivo `config` de la AWS CLI, consulte Configuraciones del archivo `config` admitidas en la Guía del usuario de AWS Command Line Interface.

```
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\"}}'
```

Contenido JSON para usar con la opción de archivo **`--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
}
```

Registrar una tarea de Step Functions

Los siguientes ejemplos muestran cómo registrar tareas de máquina de estado de Step Functions con un periodo de mantenimiento mediante la AWS CLI.

> **Note**
>
> Las tareas de los periodos de mantenimiento solo admiten los flujos de trabajo de máquinas de estado estándar de Step Functions. No son compatibles con los flujos de trabajo de máquinas de estado rápidas. Para obtener información sobre los tipos de flujos de trabajo de máquinas de estado, consulte Flujos de trabajo estándar en comparación con flujos de trabajo rápidos en la Guía para desarrolladores de AWS Step Functions.

En estos ejemplos, el usuario que creó la máquina de estado de Step Functions, creó una máquina de estado llamada `SSMMyStateMachine` con un parámetro denominado `instanceId`.

> **Important**
>
> La política de AWS Identity and Access Management (IAM) para Maintenance Windows requiere que se agregue el prefijo `SSM` a los nombres de máquina de estado de Step Functions. Antes de proceder al registro de este tipo de tarea, debe actualizar su nombre en AWS Step Functions para que incluya `SSM`. Por ejemplo, si el nombre de la máquina de estado es `MyStateMachine`, cámbielo a `SSMMyStateMachine`.

Comando de la AWS CLI:

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"
```

Contenido JSON para usar con la opción de archivo **`--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
}
```

Opciones de parámetros para el comando register-task-with-maintenance-windows

El comando register-task-with-maintenance-window proporciona varias opciones para configurar una tarea en función de sus necesidades. Algunas son obligatorias, otras opcionales y otras solo se aplican a un único tipo de tarea de ventana de mantenimiento.

En este tema se proporciona información sobre algunas de estas opciones para ayudarlo a trabajar con los ejemplos de esta sección del tutorial. Para obtener información acerca de todas las opciones de comandos, consulte register-task-with-maintenance-window en la Referencia de comando de la AWS CLI.

Opción de comando: **`--task-arn`**

La opción `--task-arn` se usa para especificar el recurso en el que opera la tarea. El valor que se especifica depende del tipo de tarea que se registra, como se describe en la siguiente tabla.

Formatos de TaskArn para tareas del periodo de mantenimiento

Tipo de tarea de la ventana de mantenimiento	Valor TaskArn
RUN_COMMAND y **AUTOMATION**	`TaskArn` es el nombre del documento de SSM o el nombre de recurso de Amazon (ARN). Por ejemplo: `AWS-RunBatchShellScript` -o bien- `arn:aws:ssm:` *`region`*`:11112222 3333:document/My-Document` .
LAMBDA	`TaskArn` es el nombre de la función o ARN. Por ejemplo: `SSMMy-Lambda-Function` -o bien- `arn:aws:lambda:` *`region`*`:11112222 3333:function:SSMMyLambdaFu nction` . **Important** La política de IAM para Maintenan ce Windows requiere que se agregue el prefijo `SSM` a los nombres de la función (o alias) de Lambda. Antes de continuar con el registro de este tipo de tareas, actualice el nombre en AWS Lambda para incluir `SSM`. Por ejemplo, si el nombre de la función de Lambda es `MyLambdaFunction` , cámbielo a `SSMMyLambdaFunction` .

Tipo de tarea de la ventana de mantenimiento	Valor TaskArn
STEP_FUNCTIONS	`TaskArn` es el ARN de la máquina de estado. Por ejemplo: `arn:aws:states:us-east-2:111122223333:stateMachine:SSMMyStateMachine` . **Important** La política de IAM para periodos de mantenimiento requiere que se agregue el prefijo `SSM` a los nombres de máquina de estado de Step Functions. Antes de registrar este tipo de tarea, debe actualizar su nombre en AWS Step Functions para que incluya `SSM`. Por ejemplo, si el nombre de la máquina de estado es `MyStateMachine` , cámbielo a `SSMMyStateMachine` .

Opción de comando: **`--service-role-arn`**

El rol que AWS Systems Manager debe asumir cuando se ejecuta la tarea del periodo de mantenimiento.

Para obtener más información, consulte Configuración de Maintenance Windows

Opción de comando: **`--task-invocation-parameters`**

La opción `--task-invocation-parameters` se utiliza para especificar los parámetros que son exclusivos de cada uno de los cuatro tipos de tarea. Los parámetros admitidos para cada uno de estos cuatro tipos se describen en la tabla siguiente.

> **Note**
>
> Para obtener información acerca de cómo utilizar los pseudoparámetros en contenido `--task-invocation-parameters`, como {{TARGET_ID}}, consulte Utilización de pseudoparámetros en el registro de las tareas del periodo de mantenimiento.

Opciones de los parámetros de invocación de tareas de periodos de mantenimiento

Tipo de tarea de la ventana de mantenimiento	Parámetros disponibles	Ejemplo

RUN_COMMAND	`Comment`
	`DocumentHash`
	`DocumentHashType`
	`NotificationConfig`
	`OutputS3BucketName`
	`OutPutS3KeyPrefix`
	`Parameters`
	`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" },`

Tipo de tarea de la ventana de mantenimiento	Parámetros disponibles	Ejemplo

```
            "OutputS3
BucketName": "amzn-s3-
demo-bucket",
            "OutputS3
KeyPrefix": "  S3-PREFIX
 ",
            "Paramete
rs": {

 "commands": [

 "Get-ChildItem$env:
 temp-Recurse|Remove-
Item-Recurse-force"
                ]
            },
            "ServiceR
oleArn": "arn:aws:
iam::123456789012:
role/MyMaintenance
WindowServiceRole",
            "TimeoutS
econds": 3600
        }
    }
```

Tipo de tarea de la ventana de mantenimiento	Parámetros disponibles	Ejemplo

AUTOMATION	DocumentVersion
	
	Parameters
		<pre>"TaskInvocationParameters": {
		  "Automation": {
		  "DocumentVersion": "3",
		  "Parameters": {
		
		  "instanceid": [
		
		  "{{TARGET_ID}}"
		  ]
		  }
		  }
		  }</pre>

LAMBDA	ClientContext
	
	Payload
	
	Qualifier
		<pre>"TaskInvocationParameters": {
		  "Lambda": {
		  "ClientContext": "ew0KICAi--truncated--0KIEXAMPLE",
		  "Payload": "{ \"targetId\": \"{{TARGET_ID}}\", \"targetType\": \"{{TARGET_TYPE}}\" }",
		  "Qualifier": "$LATEST"
		  }
		  }</pre>

Tipo de tarea de la ventana de mantenimiento	Parámetros disponibles	Ejemplo

STEP_FUNCTIONS	`Input`
	
	`Name`
		`"TaskInvocationParameters": { "StepFunctions": { "Input": "{ \"targetId\": \"{{TARGET_ID}}\" }", "Name": "{{INVOCATION_ID}}" } }`

Tutorial: vea información sobre periodos de mantenimiento mediante la AWS CLI

Este tutorial incluye comandos para ayudarle a actualizar u obtener información sobre sus periodos de mantenimiento, tareas, ejecuciones e invocaciones. Los ejemplos están organizados por comando para mostrar cómo utilizar las opciones de comando para filtrar el tipo de detalle que desea ver.

A medida que siga los pasos que se indican en este tutorial, reemplace los valores en *rojo* y cursiva por sus propias opciones y sus ID. Por ejemplo, reemplace el ID del periodo de mantenimiento *mw-0c50858d01EJEMPLO* y el ID de la instancia *i-02573cafcfEJEMPLO* por los ID de los recursos que usted cree.

Para obtener información acerca de la instalación y configuración de AWS Command Line Interface (AWS CLI), consulte Instalación, actualización y desinstalación de AWS CLI y Configuración de AWS CLI.

Ejemplos de comando

- Ejemplos de “describe-maintenance-windows”
- Ejemplos de “describe-maintenance-window-targets”
- Ejemplos de “describe-maintenance-window-tasks”
- Ejemplos de “describe-maintenance-windows-for-target”
- Ejemplos de “describe-maintenance-window-executions”

- Ejemplos de “describe-maintenance-window-schedule”

Ejemplos de “describe-maintenance-windows”

Enumerar todos los periodos de mantenimiento de su Cuenta de AWS

Ejecute el siguiente comando.

```
aws ssm describe-maintenance-windows
```

El sistema devuelve información similar a la siguiente.

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

Enumerar todos los períodos de mantenimiento habilitados

Ejecute el siguiente comando.

```
aws ssm describe-maintenance-windows --filters "Key=Enabled,Values=true"
```

El sistema devuelve información similar a la siguiente.

```
{
```

```
   "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"
      },
   ]
}
```

Enumerar todos los períodos de mantenimiento deshabilitados

Ejecute el siguiente comando.

```
aws ssm describe-maintenance-windows --filters "Key=Enabled,Values=false"
```

El sistema devuelve información similar a la siguiente.

```
{
    "WindowIdentities": [
        {
            "WindowId": "mw-6e5c9d4b7cEXAMPLE",
            "Name": "My-Disabled-Maintenance-Window",
            "Enabled": false,
            "Duration": 2,
            "Cutoff": 1
        }
    ]
}
```

Enumerar todos los periodos de mantenimiento que tienen nombres que comienzan por un prefijo determinado

Ejecute el siguiente comando.

```
aws ssm describe-maintenance-windows --filters "Key=Name,Values=My"
```

El sistema devuelve información similar a la siguiente.

```
{
    "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
        }
    ]
}
```

Ejemplos de “describe-maintenance-window-targets”

Mostrar los destinos de un período de mantenimiento que coincida con un valor específico de información del propietario

Ejecute el siguiente comando.

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**
>
> Las claves de filtro admitidas son `Type`, `WindowTargetId` y `OwnerInformation`.

El sistema devuelve información similar a la siguiente.

```
{
    "Targets": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE",
            "ResourceType": "INSTANCE",
            "Targets": [
                {
                    "Key": "tag:Name",
                    "Values": [
                        "Production"
                    ]
                }
            ],
            "OwnerInformation": "CostCenter1",
            "Name": "Target1"
        }
    ]
}
```

Ejemplos de “describe-maintenance-window-tasks”

Mostrar todas las tareas registradas que invoquen el documento de Command de SSM **AWS-RunPowerShellScript**

Ejecute el siguiente comando.

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"
```

El sistema devuelve información similar a la siguiente.

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

Visualización de todas las tareas registradas que tengan una prioridad de "3"

Ejecute el siguiente comando.

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"
```

El sistema devuelve información similar a la siguiente.

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

Mostrar todas las tareas registradas que tengan una prioridad de "1" y usar Run Command

Ejecute el siguiente comando.

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"
```

El sistema devuelve información similar a la siguiente.

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

Ejemplos de “describe-maintenance-windows-for-target”

Mostrar información acerca de los destinos de periodo de mantenimiento o las tareas asociadas con un nodo específico

Ejecute el siguiente comando.

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
```

El sistema devuelve información similar a la siguiente.

```
{
    "WindowIdentities": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "My-First-Maintenance-Window"
        },
```

```
        {
            "WindowId": "mw-9a8b7c6d5eEXAMPLE",
            "Name": "My-Second-Maintenance-Window"
        }
    ]
}
```

Ejemplos de “describe-maintenance-window-executions”

Enumerar todas las tareas ejecutadas antes de una fecha determinada

Ejecute el siguiente comando.

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"
```

El sistema devuelve información similar a la siguiente.

```
{
    "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
        }
    ]
}
```

Enumerar todas las tareas ejecutadas después de una fecha determinada

Ejecute el siguiente comando.

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"
```

El sistema devuelve información similar a la siguiente.

```
{
    "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
        }
    ]
}
```

Ejemplos de “describe-maintenance-window-schedule”

Mostrar las próximas diez ejecuciones programadas de periodo de mantenimiento para un nodo determinado

Ejecute el siguiente comando.

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
```

El sistema devuelve información similar a la siguiente.

```
{
    "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"
}
```

Mostrar el programa de periodo de mantenimiento para los nodos etiquetados con un determinado par clave-valor

Ejecute el siguiente comando.

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"
```

El sistema devuelve información similar a la siguiente.

```
{
    "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"
}
```

Mostrar las horas de inicio para las cuatro siguientes ejecuciones de un periodo de mantenimiento

Ejecute el siguiente comando.

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"
```

El sistema devuelve información similar a la siguiente.

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

Tutorial: visualización de información sobre tareas y ejecuciones de tareas mediante la AWS CLI

En este tutorial, se muestra cómo utilizar la AWS Command Line Interface (AWS CLI) para ver los detalles de las tareas completadas del periodo de mantenimiento.

Si continua directamente desde Tutorial: cree y configure un periodo de mantenimiento mediante la AWS CLI, asegúrese de haber dejado suficiente tiempo para que se ejecute el periodo de mantenimiento al menos una vez para ver los resultados de la ejecución.

A medida que siga los pasos que se indican en este tutorial, reemplace los valores en *rojo* y cursiva por sus propias opciones y sus ID. Por ejemplo, reemplace el ID del periodo de mantenimiento *mw-0c50858d01EJEMPLO* y el ID de la instancia *i-02573cafcfEJEMPLO* por los ID de los recursos que usted cree.

Para visualizar información sobre tareas y ejecuciones de tareas mediante la AWS CLI

1. Ejecute el siguiente comando para ver una lista de las ejecuciones de las tareas de un periodo de mantenimiento determinado.

Linux & macOS

```
aws ssm describe-maintenance-window-executions \
    --window-id "mw-0c50858d01EXAMPLE"
```

Windows

```
aws ssm describe-maintenance-window-executions ^
    --window-id "mw-0c50858d01EXAMPLE"
```

El sistema devuelve información similar a la siguiente.

```
{
    "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. Ejecute el siguiente comando para obtener información sobre la ejecución de una tarea del periodo de mantenimiento.

   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"
   ```

   El sistema devuelve información similar a la siguiente.

   ```
   {
       "WindowExecutionId": "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE",
       "TaskIds": [
           "c9b05aba-197f-4d8d-be34-e73fbEXAMPLE"
       ],
       "Status": "SUCCESS",
       "StartTime": 1557593493.096,
       "EndTime": 1557593498.611
   }
   ```

3. Ejecute el siguiente comando para enumerar las tareas ejecutadas como parte de una ejecución del periodo de mantenimiento.

   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"
   ```

   El sistema devuelve información similar a la siguiente.

```
{
    "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. Ejecute el siguiente comando para obtener los detalles de una ejecución de tareas.

   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"
   ```

   El sistema devuelve información similar a la siguiente.

   ```
   {
       "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. Ejecute el siguiente comando para obtener las invocaciones de tareas concretas realizadas en una ejecución de tareas.

   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"
   ```

   El sistema devuelve información similar a la siguiente.

   ```
   {
       "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
        }
    ]
}
```


Tutorial actualización de un periodo de mantenimiento mediante la AWS CLI

En este tutorial, se muestra cómo utilizar la AWS Command Line Interface (AWS CLI) para actualizar un periodo de mantenimiento. También se muestra cómo actualizar diferentes tipos de tarea, incluidos aquellos para AWS Systems Manager Run Command y Automation, AWS Lambda y AWS Step Functions.

En los ejemplos que aparecen en esta sección, se utilizan las siguientes acciones de Systems Manager para actualizar un periodo de mantenimiento:

- UpdateMaintenanceWindow
- UpdateMaintenanceWindowTarget
- UpdateMaintenanceWindowTask
- DeregisterTargetFromMaintenanceWindow

Para obtener información acerca de cómo utilizar la consola de Systems Manager para actualizar un periodo de mantenimiento, consulte Actualizar o eliminar recursos de un periodo de mantenimiento mediante la consola.

A medida que siga los pasos que se indican en este tutorial, reemplace los valores en *rojo* y cursiva por sus propias opciones y sus ID. Por ejemplo, reemplace el ID del periodo de mantenimiento *mw-0c50858d01EJEMPLO* y el ID de la instancia *i-02573cafcfEJEMPLO* por los ID de los recursos que usted cree.

Para actualizar un periodo de mantenimiento mediante la AWS CLI

1. Abra el AWS CLI y ejecute el siguiente comando para actualizar un destino e incluir un nombre y una descripción.

   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"
   ```

   El sistema devuelve información similar a la siguiente.

   ```
   {
       "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. Ejecute el siguiente comando si desea utilizar la opción `replace` para eliminar el campo de descripción y agregar un destino adicional. El campo de descripción se elimina, ya que la

actualización no incluye el campo (un valor nulo). Asegúrese de especificar un nodo adicional que se haya configurado para utilizarse con 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
```

El sistema devuelve información similar a la siguiente.

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
    "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE",
    "Targets": [
        {
            "Key": "InstanceIds",
            "Values": [
                "i-02573cafcfEXAMPLE",
                "i-0471e04240EXAMPLE"
            ]
        }
    ],
    "Name": "My-Maintenance-Window-Target"
}
```

3. La opción `start-date` permite retrasar la activación de un periodo de mantenimiento hasta una fecha futura especificada. La opción `end-date` permite establecer una fecha y hora en el futuro

después de la cual el periodo de mantenimiento dejará de ejecutarse. Especifique las opciones de formato extendido ISO-8601.

Ejecute el siguiente comando para especificar un intervalo de fecha y hora para ejecuciones programadas de forma regular del periodo de mantenimiento.

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. Ejecute el siguiente comando para actualizar una tarea del Run Command.

> **Tip**
>
> Si su destino es una instancia de Amazon Elastic Compute Cloud (Amazon EC2) para Windows Server, cambie `df` a `ipconfig` y `AWS-RunShellScript` a `AWS-RunPowerShellScript` en el siguiente comando.

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"
```

El sistema devuelve información similar a la siguiente.

```
{
    "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. Adapte y ejecute el siguiente comando para actualizar una tarea de 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"
```

   El sistema devuelve información similar a la siguiente.

```
{
    "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 está actualizando una tarea de Step Functions, adapte y ejecute el siguiente comando para actualizar los parámetros de invocación de tareas.

   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"
```

El sistema devuelve información similar a la siguiente.

```
{
    "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. Ejecute el siguiente comando para anular el registro de un destino del período de mantenimiento. Este ejemplo utiliza el parámetro `safe` para determinar que el destino no tenga referencias de alguna tarea y, por lo tanto, se pueda anular el registro.

   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
   ```

   El sistema devuelve información similar a la siguiente.

   ```
   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. Ejecute el siguiente comando para anular el registro de un destino de un período de mantenimiento incluso si el destino tiene referencias de una tarea. Puede forzar la operación de anulación del registro mediante el parámetro `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
```

El sistema devuelve información similar a la siguiente.

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
    "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
}
```

9. Ejecute el siguiente comando para actualizar una tarea del Run Command. En este ejemplo, se utiliza un parámetro de Systems Manager Parameter Store denominado `UpdateLevel`, que tiene el siguiente formato: "`{{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}}'}}"
   ```

   El sistema devuelve información similar a la siguiente.

```
{
    "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. Ejecute el siguiente comando para actualizar una tarea de Automation y especificar los parámetros `WINDOW_ID` y `WINDOW_TASK_ID` para el parámetro `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
```

El sistema devuelve información similar a la siguiente.

```
{
    "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"
}
```

Tutorial: eliminación de un periodo de mantenimiento mediante la AWS CLI

Para eliminar un periodo de mantenimiento que creó en estos tutoriales, ejecute el siguiente comando.

```
aws ssm delete-maintenance-window --window-id "mw-0c50858d01EXAMPLE"
```

El sistema devuelve información similar a la siguiente.

```
{
   "WindowId":"mw-0c50858d01EXAMPLE"
}
```

Tutorial: creación de un periodo de mantenimiento para la aplicación de revisiones mediante la consola

> **Important**
>
> Puede seguir utilizando este tema heredado para crear un período de mantenimiento para la aplicación de revisiones. Sin embargo, le recomendamos que utilice una política de revisiones. Para obtener más información, consulte Configuraciones de políticas de revisiones en Quick Setup y Cómo configurar las revisiones para las instancias de una organización mediante una política de parches Quick Setup.

Para minimizar el impacto en la disponibilidad de los servidores, le recomendamos que configure un período de mantenimiento para ejecutar la aplicación de revisiones durante las horas en que no se interrumpan las operaciones de negocio.

Debe configurar los roles y los permisos de Maintenance Windows, una herramienta de AWS Systems Manager, antes de comenzar este procedimiento. Para obtener más información, consulte Configuración de Maintenance Windows.

Para crear un período de mantenimiento para la aplicación de revisiones

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Maintenance Windows.
3. Elija Create maintenance window (Crear periodo de mantenimiento).
4. En Name (Nombre) escriba un nombre que designe este período de mantenimiento para aplicar revisiones de actualizaciones críticas e importantes.
5. (Opcional) En Description (Descripción), introduzca una descripción.
6. Elija Allow unregistered targets (Permitir destinos no registrados) si desea permitir que se ejecute una tarea del periodo de mantenimiento en los nodos administrados, incluso si no ha registrado esos nodos como destinos.

   Si elige esta opción, podrá elegir los nodos no registrados (por ID de nodo) al registrar una tarea con el periodo de mantenimiento.

   Si no elige esta opción, tendrá elegir los destinos registrados anteriormente cuando registre una tarea con el periodo de mantenimiento.
7. En la parte superior de la sección Schedule (Programación) especifique un programa para el período de mantenimiento mediante una de las tres opciones de programación.

   Para obtener información acerca de cómo crear expresiones Cron y Rate, consulte Referencia: expresiones cron y rate para Systems Manager.
8. En Duration (Duración), escriba el número de horas que se ejecutará el periodo de mantenimiento. El valor que especifique determina la hora de finalización específica del periodo de mantenimiento en función de la hora de inicio. No se permite que las tareas del período de mantenimiento comiencen después de la hora de enlace resultante menos el número de horas que especifique para Stop initiating tasks (Dejar de iniciar tareas) en el siguiente paso.

Por ejemplo, si el período de mantenimiento comienza a las 15:00 h, la duración es de tres horas y el valor de Stop initiating tasks (Dejar de iniciar tareas) es de una hora, no se pueden iniciar tareas del período de mantenimiento después de las 17:00 h.

9. En el campo Stop initiating tasks (Dejar de iniciar tareas), escriba el número de horas que el sistema debe considerar antes de que finalice el período de mantenimiento para dejar de programar nuevas tareas por ejecutar.
10. (Opcional) En Window start date (Fecha de inicio del periodo), especifique una fecha y un horario en formato extendido ISO-8601 para cuando desee que se active el periodo de mantenimiento. Esto le permite retrasar la activación del periodo de mantenimiento hasta la fecha futura especificada.
11. (Opcional) En Window end date (Fecha de finalización del periodo), especifique una fecha y un horario en formato extendido ISO-8601 para cuando desee que se desactive el periodo de mantenimiento. Esto le permite establecer una fecha y hora en el futuro después de la cual el periodo de mantenimiento dejará de ejecutarse.
12. (Opcional) En Zona horaria de la programación, especifique la zona horaria en la que se basan las programaciones de las ejecuciones de periodos de mantenimiento, en formato Internet Assigned Numbers Authority (IANA). Por ejemplo: "America/Los_Angeles", "etc/UTC" o "Asia/Seoul".

    Para obtener más información sobre los formatos válidos, consulte Time Zone Database en el sitio web de IANA.
13. (Opcional) En el área Manage tags (Administrar etiquetas), aplique uno o varios pares de claves nombre/valor al periodo de mantenimiento.

    Las etiquetas son metadatos opcionales que usted asigna a un recurso. Las etiquetas permiten clasificar los recursos de diversas maneras, por ejemplo, según la finalidad, el propietario o el entorno. Por ejemplo, es posible que desee etiquetar este periodo de mantenimiento para identificar el tipo de tareas que ejecuta. En este caso, puede especificar el siguiente par de clave nombre-valor:

    - `Key=TaskType,Value=Patching`
14. Elija Create maintenance window (Crear periodo de mantenimiento).
15. En la lista de períodos de mantenimiento, elija el que acaba de crear y, a continuación, elija Actions (Acciones), Register targets (Registrar destinos).

16. (Opcional) En la sección Maintenance window target details, proporcione un nombre, una descripción y la información del propietario (su nombre o alias) para este destino.
17. En Selección de destino, elija Especificar etiquetas de instancia.
18. En Especificar etiquetas de instancia, ingrese una clave de etiqueta y un valor de etiqueta para identificar los nodos que se va a registrar en el periodo de mantenimiento y, a continuación, seleccione Agregar.
19. Elija Register target (Registrar destino). El sistema crea un destino del período de mantenimiento.
20. En la página de detalles del período de mantenimiento que ha creado, elija Actions (Acciones), Register run command task (Registrar tarea de Run Command).
21. (Opcional) En Maintenance window task details (Detalles de la tarea de período de mantenimiento), proporcione un nombre y la descripción de esta tarea.
22. En Command document (Documento Command), elija `AWS-RunPatchBaseline`.
23. En Task priority (Prioridad de tarea), elija una prioridad. Cero (`0`) es la prioridad más alta.
24. En Targets (Destinos), en Target by (Destino por), elija el destino del período de mantenimiento que ha creado anteriormente en este procedimiento.
25. En Rate control (Control de velocidad):

    - En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

      > **Note**
      >
      > Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

    - En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

26. (Opcional) En Rol de servicio de IAM, seleccione un rol que proporcione permisos para que los asuma Systems Manager al ejecutar una tarea del periodo de mantenimiento.

Si no especifica el ARN de un rol de servicio, Systems Manager usa un rol vinculado al servicio de su cuenta. Si en su cuenta no existe ningún rol vinculado al servicio adecuado para Systems Manager, se crea cuando la tarea se registra correctamente.

> **Note**
>
> Para mejorar la seguridad, le recomendamos encarecidamente que cree una política y un rol de servicio personalizados para ejecutar las tareas del periodo de mantenimiento. La política se puede diseñar para proporcionar solo los permisos necesarios para las tareas específicas del periodo de mantenimiento. Para obtener más información, consulte Configuración de Maintenance Windows.

27. (Opcional) En Output options (Opciones de salida), para guardar la salida del comando en un archivo, seleccione el cuadro Enable writing output to S3 (Permitir la escritura de salida en S3). Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancias asignado al nodo administrado, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, verifique que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

Para transmitir la salida a un grupo de registro de Amazon CloudWatch Logs, seleccione la casilla CloudWatch output (Salida de CloudWatch). Ingrese el nombre del grupo de registro en la casilla.

28. En la sección SNS notifications (Notificaciones de SNS), seleccione la casilla de verificación Enable SNS notifications (Habilitar notificaciones de SNS) si desea recibir notificaciones sobre el estado de la ejecución de los comandos.

Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.

29. En Parameters (Parámetros):

    - En Operation (Operación), elija Scan (Analizar) para buscar las revisiones que faltan o elija Install (Instalar) para buscar las revisiones que faltan e instalarlos.
    - No necesita especificar nada en el campo Snapshot Id (Id de instantánea). Este sistema genera y proporciona este parámetro automáticamente.
    - No es necesario ingresar nada en el campo Install Override List (Lista de anulación de instalación) a menos que desee que Patch Manager utilice un conjunto de revisiones diferente al especificado para la línea de base de revisiones. Para obtener más información, consulta Nombre del parámetro: `InstallOverrideList`.
    - En RebootOption, especifique si desea que los nodos se reinicien si se instalan revisiones durante la operación `Install` o si Patch Manager detecta otras revisiones instaladas a partir del último reinicio de nodo. Para obtener más información, consulta Nombre del parámetro: `RebootOption`.
    - (Opcional) Para Comment (Comentario), introduzca una nota de seguimiento o un recordatorio sobre este comando.
    - Para Timeout (seconds) Tiempo de espera (segundos), escriba el número de segundos que el sistema tiene que esperar a que finalice la operación antes de que se considere incorrecta.

30. Elija Register Run command task (Registrar tarea de Run Command).

Una vez que se complete la tarea del periodo de mantenimiento, puede ver los detalles de la conformidad de revisiones en la consola de Systems Manager en la herramienta de Fleet Manager.

También puede ver la información de cumplimiento en la herramienta Patch Manager, en la pestaña Informes de conformidad.

También puede usar las API DescribePatchGroupState y DescribeInstancePatchStatesForPatchGroup para ver los detalles de conformidad. Para obtener información sobre los datos de conformidad de revisiones, consulte Acerca de la conformidad de parches.

Programación de aplicación de parches mediante periodos de mantenimiento

Después de configurar una base de referencia de revisiones (y, si lo desea, un grupo de revisiones), puede aplicar las revisiones al nodo mediante un periodo de mantenimiento. Un período de mantenimiento puede reducir el impacto en la disponibilidad del servidor al permitir especificar una hora para realizar el proceso de aplicación de parches que no interrumpa las operaciones de negocio. Un período de mantenimiento funciona del siguiente modo:

1. Cree un período de mantenimiento con una programación para sus operaciones de aplicación de parches.
2. Elija los destinos del periodo de mantenimiento mediante la especificación de la etiqueta `Patch Group` o `PatchGroup` para el nombre de grupo y cualquier valor para el que haya definido etiquetas de Amazon Elastic Compute Cloud (Amazon EC2), por ejemplo, “servidores de producción” o “US-EAST-PROD”. (Tiene que usar `PatchGroup`, sin espacio, si ha permitido las etiquetas en los metadatos de las instancias de EC2).
3. Cree una nueva tarea de periodo de mantenimiento y especifique el documento `AWS-RunPatchBaseline`.

Al configurar la tarea, puede elegir entre analizar los nodos o analizar e instalar las revisiones en los nodos. Si decide analizar nodos, Patch Manager, una herramienta de AWS Systems Manager, analiza cada uno de ellos y genera una lista de las revisiones que faltan para que las revise.

Si elige analizar e instalar revisiones, Patch Manager analiza cada nodo y compara la lista de revisiones instaladas con la de revisiones aprobadas en la base de referencia. Patch Manager identifica las revisiones que faltan y, a continuación, descarga e instala todas las revisiones faltantes y aprobadas.

Si desea realizar un único análisis o instalación para corregir un problema, puede usar Run Command para llamar al documento `AWS-RunPatchBaseline` directamente.

> **Important**
>
> Después de instalar las revisiones, Systems Manager reinicia cada nodo. El reinicio es necesario para asegurarse de que las revisiones se han instalado correctamente y para garantizar que el sistema no haya podido dejar el nodo en un estado incorrecto. (Excepción: si el parámetro `RebootOption` se configura en `NoReboot` en el documento

`AWS-RunPatchBaseline`, el nodo administrado no se reinicia una vez que se ejecuta Patch Manager. Para obtener más información, consulte Nombre del parámetro: `RebootOption`.)

Utilización de pseudoparámetros en el registro de las tareas del periodo de mantenimiento

Cuando se registra una tarea en Maintenance Windows, una herramienta de AWS Systems Manager, puede especificar los parámetros únicos para cada uno de los cuatro tipos. (En los comandos de la CLI, estas se ofrecen al utilizar la opción `--task-invocation-parameters`).

También puede hacer referencia a determinados valores mediante la sintaxis de pseudoparámetros, como `{{RESOURCE_ID}}`, `{{TARGET_TYPE}}` y `{{WINDOW_TARGET_ID}}`. Al ejecutarse la tarea del periodo de mantenimiento, esta pasa los valores correctos en lugar de los marcadores de posición del pseudoparámetro. Más adelante, en la sección Pseudoparámetros admitidos, se ofrece una lista completa de los pseudoparámetros que se pueden utilizar.

Important

Para el tipo de destino `RESOURCE_GROUP`, en función del formato de ID necesario para la tarea, puede elegir entre usar `{{TARGET_ID}}` y `{{RESOURCE_ID}}` para hacer referencia al recurso cuando se ejecute la tarea. `{{TARGET_ID}}` devuelve el ARN completo del recurso. `{{RESOURCE_ID}}` devuelve solo un ID o nombre más corto del recurso, como se muestra en estos ejemplos.

- Formato de `{{TARGET_ID}}`: `arn:aws:ec2:us-east-1:123456789012:instance/i-02573cafcfEXAMPLE`
- Formato de `{{RESOURCE_ID}}`: `i-02573cafcfEXAMPLE`

Para el tipo de destino `INSTANCE`, los parámetros `{{RESOURCE_ID}}` y `{{TARGET_ID}}` solo generan el ID de instancia. Para obtener más información, consulte Pseudoparámetros admitidos.
`{{TARGET_ID}}` y `{{RESOURCE_ID}}` se pueden utilizar para pasar los ID de los recursos de AWS solo a tareas de Automation, Lambda y Step Functions. Estos dos pseudoparámetros no se pueden utilizar con tareas de Run Command.

Ejemplos de pseudoparámetros

Suponga que la carga para una tarea de AWS Lambda necesita referenciar una instancia por su ID.

Si utiliza un destino de periodo de mantenimiento `INSTANCE` o `RESOURCE_GROUP`, esto se puede lograr con el pseudoparámetro `{{RESOURCE_ID}}`. Por ejemplo:

```
"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 su tarea de Lambda está diseñada para ejecutarse en otro tipo de destino compatible además de las instancias Amazon Elastic Compute Cloud (Amazon EC2), como una tabla de Amazon DynamoDB, se puede utilizar la misma sintaxis y `{{RESOURCE_ID}}` solo genera el nombre de la tabla. Sin embargo, si necesita el ARN completo de la tabla, utilice `{{TARGET_ID}}`, como se muestra en el siguiente ejemplo.

```
"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 misma sintaxis funciona para la definición de destinos de instancias u otros tipos de recursos. Cuando se han agregado varios tipos de recursos a un grupo de recursos, la tarea se ejecuta en cada uno de los recursos correspondientes.

> **Important**
>
> No todos los tipos de recursos que se pueden incluir en un grupo de recursos generan un valor para el parámetro `{{RESOURCE_ID}}`. Para obtener una lista de los tipos de recursos admitidos, consulte Pseudoparámetros admitidos.

Otro ejemplo es que, para ejecutar una tarea de Automation que detenga las instancias EC2, deberá especificar el documento de Systems Manager (documento de SSM) `AWS-StopEC2Instance` como el valor de `TaskArn` y utilizar el pseudoparámetro `{{RESOURCE_ID}}`:

```
"TaskArn": "AWS-StopEC2Instance",
    "TaskType": "AUTOMATION"
    "TaskInvocationParameters": {
        "Automation": {
            "DocumentVersion": "1",
            "Parameters": {
                "instanceId": [
                    "{{RESOURCE_ID}}"
                ]
            }
        }
    }
```

Para ejecutar una tarea de Automation que copie una instantánea de un volumen de Amazon Elastic Block Store (Amazon EBS), deberá especificar el documento de SSM `AWS-CopySnapshot` como el valor de `TaskArn` y utilizar el pseudoparámetro `{{RESOURCE_ID}}`.

```
"TaskArn": "AWS-CopySnapshot",
    "TaskType": "AUTOMATION"
    "TaskInvocationParameters": {
        "Automation": {
            "DocumentVersion": "1",
            "Parameters": {
                "SourceRegion": "us-east-2",
                "targetType":"RESOURCE_GROUP",
                "SnapshotId": [
                    "{{RESOURCE_ID}}"
                ]
            }
        }
```

```
    }
```

Pseudoparámetros admitidos

En la siguiente lista se describen los pseudoparámetros que puede especificar mediante la sintaxis {{*PSEUDO_PARAMETER*}} en la opción `--task-invocation-parameters`.

- **`WINDOW_ID`**: el ID del período de mantenimiento de destino.
- **`WINDOW_TASK_ID`**: el ID de la tarea del periodo que se está ejecutando.
- **`WINDOW_TARGET_ID`**: el ID del destino de la ventana que incluye el destino (ID de destino).
- **`WINDOW_EXECUTION_ID`**: el ID de ejecución de la ventana actual.
- **`TASK_EXECUTION_ID`**: el ID de la ejecución de tarea actual.
- **`INVOCATION_ID`**: el ID de la invocación actual.
- **`TARGET_TYPE`**: el tipo de destino. Los tipos admitidos son `RESOURCE_GROUP` e `INSTANCE`.
- **`TARGET_ID`**:

  Si el tipo de destino especificado es `INSTANCE`, el pseudoparámetro `TARGET_ID` se reemplaza por el ID de la instancia. Por ejemplo, `i-078a280217EXAMPLE`.

  Si el tipo de destino especificado es `RESOURCE_GROUP`, el valor que se referencia para la ejecución de la tarea es el ARN completo del recurso. Por ejemplo: `arn:aws:ec2:us-east-1:123456789012:instance/i-078a280217EXAMPLE`. La tabla siguiente proporciona valores de `TARGET_ID` de ejemplo para determinados tipos de recursos de un grupo de recursos.

  > **Note**
  >
  > `TARGET_ID` no es compatible con tareas de Run Command.

Tipo de recurso	TARGET_ID de ejemplo	
`AWS::CloudWatch::Alarm`	`arn:aws:cloudwatch:us-east-1:123456789012:alarm:MyCloudWatchAlarm i-078a280217EXAMPLE`	

Tipo de recurso	TARGET_ID de ejemplo	
AWS::DynamoDB::Table	arn:aws:dynamodb:us-east-1:123456789012:table/MyTable	
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::Security Group	arn:aws:ec2:us-east-1:123456789012:security-group/sg-cEXAMPLE	
AWS::EC2::Snapshot	arn:aws:ec2:us-east-1:123456789012:snapshot/snap-03866bf003EXAMPLE	
AWS::EC2::Volume	arn:aws:ec2:us-east-1:123456789012:volume/vol-0912e04d78EXAMPLE	
AWS::ECS::Service	arn:aws:ecs:us-east-1:123456789012:service/my-ecs-service	

Tipo de recurso	TARGET_ID de ejemplo	
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:::amzn-s3-demo-bucket	
AWS::SSM::ManagedInstance	arn:aws:ssm:us-east-1:123456789012:managed-instance/mi-0feadcf2d9EXAMPLE	

- **RESOURCE_ID**: el ID abreviado de un tipo de recurso incluido en un grupo de recursos. La tabla siguiente proporciona valores de RESOURCE_ID de ejemplo para determinados tipos de recursos de un grupo de recursos.

> **Note**
>
> RESOURCE_ID no es compatible con tareas de Run Command.

Tipo de recurso	RESOURCE_ID de ejemplo	
AWS::CloudWatch::Alarm	MyCloudWatchAlarm	
AWS::DynamoDB::Table	MyTable	
AWS::EC2::Instance	i-078a280217EXAMPLE	
AWS::EC2::Image	ami-02250b3732EXAMPLE	

Tipo de recurso	RESOURCE_ID de ejemplo	
AWS::EC2::Security Group	sg-cEXAMPLE	
AWS::EC2::Snapshot	snap-03866bf003EXAMPLE	
AWS::EC2::Volume	vol-0912e04d78EXAMPLE	
AWS::ECS::Service	my-ecs-service	
AWS::RDS::DBCluster	My-Cluster	
AWS::RDS::DBInstance	My-SQL-Instance	
AWS::S3::Bucket	amzn-s3-demo-bucket	
AWS::SSM::ManagedInstance	mi-0feadcf2d9EXAMPLE	

> **Note**
>
> Si el grupo de recursos de AWS que usted especifica incluye tipos de recursos que no generan un valor de `RESOURCE_ID` y no aparecen en la tabla anterior, el parámetro `RESOURCE_ID` no se rellena. Se seguirá produciendo una invocación de ejecución para ese recurso. En estos casos, utilice el pseudoparámetro `TARGET_ID` en su lugar, que se reemplazará por el ARN completo del recurso.

Programación de la ventana de mantenimiento y opciones de periodo activo

Al crear un período de mantenimiento, debe especificar la frecuencia con la que se ejecuta el período de mantenimiento mediante una expresión cron o rate. Si lo prefiere, puede especificar un intervalo de fechas durante el cual se puede ejecutar el periodo de mantenimiento en su programación habitual, así como una zona horaria en la que basar esa programación habitual.

Tenga en cuenta, sin embargo, que la opción de zona horaria y las opciones de fecha de inicio y fecha de finalización no se afectan entre sí. Cualquier fecha de inicio y de finalización que especifique (con o sin corrección para su zona horaria) solo determinan el período válido durante el cual se puede ejecutar el período de mantenimiento en su programación. Una opción de zona horaria determina la zona horaria internacional en la que se basa la programación del período de mantenimiento durante su período válido.

> **Note**
>
> Especifique las fechas de inicio y de finalización en formato de marca temporal ISO-8601. Por ejemplo: `2021-04-07T14:29:00-08:00`.
> Especifique las zonas horarias en formato IANA (del inglés, Internet Assigned Numbers Authority). Por ejemplo, `America/Chicago`, `Europe/Berlin` o `Asia/Tokyo`.

Ejemplos

- Ejemplo 1: especificar una fecha de inicio del período de mantenimiento
- Ejemplo 2: especificar una fecha de inicio y finalización del período de mantenimiento
- Ejemplo 3: crear un período de mantenimiento que se ejecuta solo una vez
- Ejemplo 4: especifique el número de días de desplazamiento de la programación de un período de mantenimiento

Ejemplo 1: especificar una fecha de inicio del período de mantenimiento

Suponga que utiliza la AWS Command Line Interface (AWS CLI) para crear un periodo de mantenimiento con las siguientes opciones:

- `--start-date 2021-01-01T00:00:00-08:00`
- `--schedule-timezone "America/Los_Angeles"`
- `--schedule "cron(0 09 ? * WED *)"`

Por ejemplo:

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 *)"
```

Esto significa que la primera ejecución del periodo de mantenimiento no ocurrirá hasta después de la fecha y la hora de inicio especificadas, que es a las 00.00 h en la zona horaria del Pacífico de EE. UU. del viernes 1 de enero de 2021. (Esta zona horaria está ocho horas por detrás de la hora UTC). En este caso, la fecha y la hora de inicio del periodo no representan cuándo se ejecutan por primera vez los periodos de mantenimiento. En conjunto, los valores `--schedule-timezone` y `--schedule` significan que el periodo de mantenimiento se ejecutará todos los miércoles a las 9.00 h en la zona horaria del Pacífico de EE. UU. (representados por “América/Los Ángeles” en formato IANA). La primera ejecución en el periodo permitido será el miércoles 4 de enero de 2021 a las 9.00 h en la zona horaria del Pacífico de EE. UU.

Ejemplo 2: especificar una fecha de inicio y finalización del período de mantenimiento

Supongamos que, a continuación, desea crear un período de mantenimiento con estas opciones:

- `--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)"`

Por ejemplo:

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)"
```

El periodo permitido para este periodo de mantenimiento comienza a las 3.15 h, hora estándar de Japón, el 1 de enero de 2019. El período válido para este período de mantenimiento termina a las 6:15, hora estándar de Japón, el domingo, 30 de junio de 2019. (Esta zona horaria está nueve horas por delante de la hora UTC). En conjunto, los valores `--schedule-timezone` y `--schedule` significan que el período de mantenimiento se ejecutará todos los martes a las 3:15 en la zona horaria estándar de Japón (representados por "Asia/Tokyo" en formato IANA). Esto se debe a que el periodo de mantenimiento se ejecuta cada siete días y se activa a las 3.15 h del martes 1 de enero. La última ejecución será a las 3:15, hora estándar de Japón, el martes, 25 de junio de 2019. Este es el último martes antes de que el periodo de mantenimiento permitido finalice cinco días más tarde.

Ejemplo 3: crear un período de mantenimiento que se ejecuta solo una vez

Ahora, cree un periodo de mantenimiento con esta opción:

- `--schedule "at(2020-07-07T15:55:00)"`

Por ejemplo:

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
```

Este período de mantenimiento se ejecuta una única vez, a las 15.55 (UTC) del 7 de julio de 2020. El periodo de mantenimiento se puede ejecutar un máximo de cinco horas, según sea necesario, pero se impide que se inicien tareas nuevas dos horas antes de que finalice el periodo de mantenimiento.

Ejemplo 4: especifique el número de días de desplazamiento de la programación de un período de mantenimiento

Ahora, cree un periodo de mantenimiento con esta opción:

```
--schedule-offset 2
```

Por ejemplo:

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 desplazamiento de la programación es el número de días que se debe esperar después de la fecha y hora especificadas por una expresión CRON antes de ejecutar el período de mantenimiento.

En el ejemplo anterior, la expresión CRON programa un periodo de mantenimiento para que se ejecute el tercer martes de cada mes a las 23.30 h:

```
--schedule "cron(0 30 23 ? * TUE#3 *)
```

Sin embargo, si incluye `--schedule-offset 2` significará que el período de mantenimiento no se ejecutará hasta las 23.30 h dos días después del tercer martes de cada mes.

Los desplazamientos de la programación solo se admiten en las expresiones CRON.

Más información

- Referencia: expresiones cron y rate para Systems Manager
- Crear y configurar un periodo de mantenimiento mediante la consola
- Tutorial: cree y configure un periodo de mantenimiento mediante la AWS CLI
- CreateMaintenanceWindow en la Referencia de la API de AWS Systems Manager
- create-maintenance-window en la sección AWS Systems Manager de la Referencia de comandos de la AWS CLI
- Base de datos de zonas horarias en el sitio web de IANA

Registro de tareas del periodo de mantenimiento sin destinos

Para cada periodo de mantenimiento que crea, puede especificar una o más tareas que se deben realizar cuando se ejecuta el periodo de mantenimiento. En la mayoría de los casos, debe especificar

los recursos o los destinos en los que se ejecutará la tarea. En algunos casos, sin embargo, no es necesario que especifique destinos de forma explícita en la tarea.

Se deben especificar uno o más destinos para las tareas de tipo Systems Manager Run Command del periodo de mantenimiento. Según la naturaleza de la tarea, los destinos son opcionales para otros tipos de tarea de periodo de mantenimiento (Automatización de Systems Manager, AWS Lambda y AWS Step Functions).

Para los tipos de tareas de Lambda y Step Functions, la necesidad de un destino dependerá del contenido de la función o la máquina de estado que haya creado.

> **Note**
>
> Cuando una tarea tiene objetivos registrados, Automatización, AWS Lambda y las tareas de AWS Step Functions resuelven los objetivos de los grupos de recursos y las etiquetas y envían una invocación por recurso resuelto, lo que da como resultado varias invocaciones de tareas. Pero supongamos, por ejemplo, que solo desea una invocación para una tarea de Lambda que esté registrada en un grupo de recursos que contenga más de una instancia. En este caso, si está trabajando en AWS Management Console, elija la opción Destino de tarea no requerido en la página Registrar tarea de Lambda o Editar tarea de Lambda. Si utiliza el comando AWS CLI, no especifique los objetivos mediante el parámetro `--targets` al ejecutar el comando register-task-with-maintenance-window o el comando update-maintenance-window-task.

En muchos casos, no es necesario especificar de forma explícita un destino para una tarea de automatización. Por ejemplo, suponga que crea una tarea de tipo Automation para actualizar una Amazon Machine Image (AMI) para Linux mediante el manual de procedimientos `AWS-UpdateLinuxAmi`. Cuando se ejecuta la tarea, la AMI se actualiza con los paquetes de distribución de Linux y el software de Amazon disponibles más recientes. Las instancias nuevas que se crearon a partir de la AMI ya tienen estas actualizaciones instaladas. Como el ID de la AMI que se actualizará se especifica en los parámetros de entrada del manual de procedimientos, no es necesario volver a especificar un destino en la tarea del periodo de mantenimiento.

Del mismo modo, suponga que utiliza la AWS Command Line Interface (AWS CLI) para registrar un periodo de mantenimiento de una tarea de automatización que utiliza el manual de procedimientos `AWS-RestartEC2Instance`. Puesto que el nodo que se debe reiniciar se especifica en el

argumento `--task-invocation-parameters`, no es necesario especificar también una opción `--targets`.

> **Note**
>
> En el caso de las tareas del periodo de mantenimiento sin un destino especificado, no puede proporcionar valores para `--max-errors` ni `--max-concurrency`. En su lugar, el sistema inserta un valor de marcador `1`, el cual podría notificarse en la respuesta a los comandos, como describe-maintenance-window-tasks y get-maintenance-window-task. Estos valores no afectan la ejecución de la tarea y se pueden ignorar.

En el siguiente ejemplo, se muestra la omisión de las opciones `--targets`, `--max-errors` y `--max-concurrency` para una tarea de un periodo de mantenimiento sin destino.

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**
>
> Para tareas de periodo de mantenimiento registradas antes del 23 de diciembre de 2020: si especificó destinos para la tarea y ya no es necesario uno, puede actualizar esa tarea para eliminar los destinos mediante la consola de Systems Manager o el comando AWS CLI de la update-maintenance-window-task.

Más información

- Mensajes de error: “Maintenance window tasks without targets don't support MaxConcurrency values” (“Las tareas del periodo de mantenimiento sin destinos no admiten valores MaxConcurrency”) y “Maintenance window tasks without targets don't support MaxErrors values” (“Las tareas del periodo de mantenimiento sin destinos no admiten valores MaxErrors”).

Solución de problemas de periodos de mantenimiento

Utilice la siguiente información como ayuda para solucionar problemas con periodos de mantenimiento.

Temas

- Error de edición de tarea: en la página de edición de una tarea de periodo de mantenimiento, la lista de roles de IAM devuelve un mensaje de error: “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" (No se encontró el rol de IAM Maintenance Window especificada para esta tarea. Puede que se haya eliminado o que todavía no se haya creado).
- No todos los destinos del periodo de mantenimiento se actualizan
- La tarea falla con el estado de invocación de tarea: “El rol proporcionado no contiene los permisos SSM correctos”.
- La tarea falla con el mensaje de error: “Step fails when it is validating and resolving the step inputs” (El paso falla cuando está validando y resolviendo las entradas del paso).
- Mensajes de error: “Maintenance window tasks without targets don't support MaxConcurrency values” (“Las tareas del periodo de mantenimiento sin destinos no admiten valores MaxConcurrency”) y “Maintenance window tasks without targets don't support MaxErrors values” (“Las tareas del periodo de mantenimiento sin destinos no admiten valores MaxErrors”).

Error de edición de tarea: en la página de edición de una tarea de periodo de mantenimiento, la lista de roles de IAM devuelve un mensaje de error: “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" (No se encontró el rol de IAM Maintenance Window especificada para esta tarea. Puede que se haya eliminado o que todavía no se haya creado).

Problema 1: el rol de periodo de mantenimiento de AWS Identity and Access Management (IAM) que especificó originalmente se eliminó después de que se creó la tarea.

Posible solución: 1) Seleccione un rol de periodo de mantenimiento de IAM diferente, si ya existe uno en su cuenta, o cree uno nuevo y selecciónelo para la tarea.

Problema 2: si la tarea se creó mediante la AWS Command Line Interface (AWS CLI), AWS Tools for Windows PowerShell o un AWS SDK, es posible que se haya especificado un nombre de rol de periodo de mantenimiento de IAM inexistente. Por ejemplo, es posible se haya eliminado el rol de periodo de mantenimiento de IAM antes de crear la tarea, o que el nombre del rol se haya escrito incorrectamente, como **`myrole`** en lugar de **`my-role`**.

Posible solución: Seleccione el nombre correcto del rol de periodo de mantenimiento de IAM que desea utilizar o cree uno nuevo con el fin de especificarlo para la tarea.

No todos los destinos del periodo de mantenimiento se actualizan

Problema: observa que las tareas del periodo de mantenimiento no se ejecutaron en todos los recursos indicados como destino por el periodo de mantenimiento. Por ejemplo, en los resultados de la ejecución del periodo de mantenimiento, la tarea de ese recurso se marca como error o tiempo de espera agotado.

Solución: la conectividad y la disponibilidad son las razones más comunes por las que una tarea del periodo de mantenimiento no se ejecuta en un recurso de destino. Por ejemplo:

- Systems Manager perdió la conexión con el recurso antes de la operación del periodo de mantenimiento o durante ella.
- El recurso se quedó sin conexión o se detuvo durante la operación del periodo de mantenimiento.

Puede esperar a la hora del próximo periodo de mantenimiento programado para ejecutar las tareas en los recursos. Puede ejecutar de forma manual las tareas del periodo de mantenimiento en los recursos que no estaban disponibles o que estaban sin conexión.

La tarea falla con el estado de invocación de tarea: “El rol proporcionado no contiene los permisos SSM correctos”.

Problema: especificó un rol de servicio de periodo de mantenimiento para una tarea, pero esta no se ejecuta de forma correcta y el estado de invocación de la tarea indica que “el rol proporcionado no contiene los permisos SSM correctos”.

- Solución: en Tarea 1: cree una política personalizada para el rol de servicio de periodo de mantenimiento mediante la consola, ofrecemos una política básica que puede adjuntar al rol de servicio de periodo de mantenimiento personalizado. La política incluye los permisos necesarios para muchos escenarios de tareas. Sin embargo, debido a la gran variedad de tareas que puede ejecutar, es posible que deba proporcionar permisos adicionales en la política para el rol de periodo de mantenimiento.

  Por ejemplo, algunas acciones de Automation trabajan con pilas de AWS CloudFormation. Por lo tanto, es posible que tenga que agregar permisos adicionales de `cloudformation:CreateStack`, `cloudformation:DescribeStacks`, y `cloudformation:DeleteStack` a la política para el rol de servicio del periodo de mantenimiento.

  Otro ejemplo es el manual de procedimientos de Automation `AWS-CopySnapshot`, que requiere permisos para crear una instantánea de Amazon Elastic Block Store (Amazon EBS). Por lo tanto, es posible que tenga que agregar el permiso `ec2:CreateSnapshot`.

  Para obtener información sobre los permisos de rol que necesita un manual de procedimientos de Automatización administrada de AWS, consulte las descripciones de manuales de procedimientos en la referencia de manuales de procedimientos de Automatización de AWS Systems Manager.

  Para obtener más información acerca de los permisos de rol que necesita un documento de SSM administrado de AWS, revise el contenido del mismo en la sección Documentos de la consola de Systems Manager.

  Para obtener información sobre los permisos de rol necesarios para las tareas de Step Functions, de Lambda y los manuales de procedimientos de automatización personalizados y los documentos de SSM, verifique los requisitos de permisos con el autor de esos recursos.

La tarea falla con el mensaje de error: “Step fails when it is validating and resolving the step inputs” (El paso falla cuando está validando y resolviendo las entradas del paso).

Problema: un manual de procedimientos de Automation o un documento de Command de Systems Manager que se utiliza en una tarea requiere que especifique entradas como `InstanceId` o `SnapshotId`, pero no se proporciona un valor o no se suministra correctamente.

- Solución 1: si la tarea tiene como destino un recurso único, como un nodo único o una instantánea única, ingrese su ID en los parámetros de entrada de la tarea.
- Solución 2: si la tarea tiene como destino varios recursos, como la creación de imágenes a partir de varios nodos cuando utiliza el manual de procedimientos `AWS-CreateImage`, puede utilizar uno de los pseudoparámetros admitidos para tareas del periodo de mantenimiento en los parámetros de entrada para representar los ID de nodo en el comando.

  Los siguientes comandos registran una tarea de Systems Manager Automation con un periodo de mantenimiento mediante la AWS CLI. El valor `--targets` indica un ID de destino del periodo de mantenimiento. Además, a pesar de que el parámetro `--targets` especifica un ID de destino del periodo, los parámetros del manual de procedimientos de Automation requieren que se proporcione un ID de nodo. En este caso, el comando utiliza el pseudoparámetro `{{RESOURCE_ID}}`como el valor de `InstanceId`.

  Comando de la AWS CLI:

  Linux & macOS

  El siguiente comando de ejemplo reinicia las instancias de Amazon Elastic Compute Cloud (Amazon EC2) que pertenecen al grupo de destino del periodo de mantenimiento con el ID e32eecb2-646c-4f4b-8ed1-205fbEJEMPLO.

```
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"
```

Para obtener más información acerca de cómo trabajar con pseudoparámetros para tareas de periodo de mantenimiento, consulte Utilización de pseudoparámetros en el registro de las tareas del periodo de mantenimiento y Ejemplos de registro de tareas.

Mensajes de error: “Maintenance window tasks without targets don't support MaxConcurrency values” (“Las tareas del periodo de mantenimiento sin destinos no admiten valores MaxConcurrency”) y “Maintenance window tasks without targets don't support MaxErrors values” (“Las tareas del periodo de mantenimiento sin destinos no admiten valores MaxErrors”).

Problema: cuando registra una tarea de tipo Run Command, debe especificar al menos un destino en el que se ejecutará la tarea. Para otros tipos de tarea (Automation, AWS Lambda y AWS Step Functions), en función de la naturaleza de la tarea, los destinos son opcionales. Las opciones `MaxConcurrency` (la cantidad de recursos en los que se ejecuta una tarea al mismo tiempo) y `MaxErrors` (la cantidad de errores para ejecutar la tarea en los recursos de destino antes de que se produzca un error en la tarea) no son necesarias ni compatibles con las tareas de periodo de mantenimiento que no especifican destinos. El sistema genera estos mensajes de error cuando se especifican valores para cualquiera de estas opciones y no se especifica ningún destino de tarea.

Solución: si recibe alguno de estos errores, elimine los valores de simultaneidad y del límite de error antes de continuar registrando o actualizando la tarea del periodo de mantenimiento.

Para obtener más información acerca de la ejecución de tareas que no especifican destinos, consulte Registro de tareas del periodo de mantenimiento sin destinos en la Guía del usuario de AWS Systems Manager.

AWS Systems Manager Quick Setup

Utilice Quick Setup, una herramienta de AWS Systems Manager, para configurar rápidamente los servicios y características de Amazon Web Services de uso frecuente con las prácticas recomendadas. Quick Setup simplifica la configuración de servicios, incluido Systems Manager, mediante la automatización de tareas comunes o recomendadas. Estas tareas incluyen, por ejemplo, la creación de roles de perfil de instancias de AWS Identity and Access Management (IAM) necesarios y la configuración de prácticas recomendadas operativas, como análisis periódicos de revisiones y recopilación de inventario. El uso de Quick Setup no supone costo alguno. Sin embargo, se puede incurrir en costos en función del tipo de servicios que configure y de los límites de uso, sin cargos por los servicios utilizados para configurar el servicio. Para comenzar a utilizar Quick Setup, abra la consola de Systems Manager. En el panel de navegación, elija Quick Setup.

> **Note**
>
> Si se le redirigió a Quick Setup a fin de configurar sus instancias para que Systems Manager las administre, complete el procedimiento en Instalar la administración de host de Amazon EC2 mediante Quick Setup.

¿Cuáles son los beneficios principales de Quick Setup?

Entre los beneficios de Quick Setup se incluyen los siguientes:

- Simplifique la configuración de servicios y funciones

  Quick Setup lo guía a través de la configuración de las prácticas recomendadas operativas y las implementa automáticamente. El panel de Quick Setup muestra una vista en tiempo real del estado de implementación de la configuración.
- Implementar configuraciones automáticamente en varias cuentas

  Puede utilizar Quick Setup en una Cuenta de AWS individual o en varias Cuentas de AWS y Regiones de AWS mediante la integración con AWS Organizations. El uso de Quick Setup en varias cuentas ayuda a garantizar que su organización mantenga configuraciones coherentes.

- Eliminar la desviación de la configuración

  La desviación de configuración se produce cada vez que un usuario realiza algún cambio en un servicio o característica que entra en conflicto con las selecciones realizadas a través de Quick Setup. Quick Setup verifica la desviación de la configuración de forma periódica e intenta corregirla.

¿Quién debe utilizar Quick Setup?

Quick Setup es más beneficioso para los clientes que ya tienen experiencia con los servicios y características que están configurando y desean simplificar su proceso de configuración. Si no está familiarizado con el Servicio de AWS que está configurando con Quick Setup, se recomienda que obtenga más información sobre el servicio. Revise el contenido de la Guía del usuario correspondiente antes de crear una configuración con Quick Setup.

Disponibilidad de Quick Setup en Regiones de AWS

En las Regiones de AWS siguientes, puede utilizar todos los tipos de configuración de Quick Setup para toda una organización, conforme a la configuración de AWS Organizations, o solo para las cuentas organizativas y las regiones que seleccione. También puede utilizar Quick Setup con una sola cuenta en estas regiones.

- Este de EE. UU. (Ohio)
- Este de EE. UU. (Norte de Virginia)
- Oeste de EE. UU. (Norte de California)
- Oeste de EE. UU. (Oregón)
- Asia-Pacífico (Bombay)
- Asia-Pacífico (Seúl)
- Asia-Pacífico (Singapur)
- Asia-Pacífico (Sídney)
- Asia-Pacífico (Tokio)
- Canadá (centro)
- Europa (Fráncfort)
- Europa (Estocolmo)

- Europa (Irlanda)
- Europa (Londres)
- Europa (París)
- América del Sur (São Paulo)

En las siguientes regiones, para cuentas individuales solo está disponible el tipo de configuración Administración de hosts:

- Europa (Milán)
- Asia-Pacífico (Hong Kong)
- Medio Oriente (Baréin)
- China (Pekín)
- China (Ningxia)
- AWS GovCloud (EE. UU. Este)
- AWS GovCloud (Oeste de EE. UU.)

Para ver una lista de todas las regiones admitidas para Systems Manager, consulte la columna Region de Systems Manager service endpoints en la Referencia general de Amazon Web Services.

Introducción a Quick Setup

Utilice la información de este tema como ayuda para prepararse para usar Quick Setup.

Temas

- Roles y permisos de IAM para la incorporación de Quick Setup
- Incorporación manual para trabajar con la API de Quick Setup de forma programática

Roles y permisos de IAM para la incorporación de Quick Setup

Quick Setup lanzó una nueva experiencia de consola y una API también nueva. Ahora puede interactuar con esta API mediante la consola, la AWS CLI, AWS CloudFormation y los SDK. Si opta por la nueva experiencia, las configuraciones existentes se recrearán con la nueva API. Este proceso puede tardar varios minutos, según el número de configuraciones existentes en la cuenta.

Para poder usar la nueva consola de Quick Setup, debe tener permisos para las acciones siguientes:

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm-quicksetup:*",
                "cloudformation:DescribeStackSetOperation",
                "cloudformation:ListStacks",
                "cloudformation:DescribeStacks",
                "cloudformation:DescribeStackResources",
                "cloudformation:ListStackSetOperations",
                "cloudformation:ListStackInstances",
                "cloudformation:DescribeStackSet",
                "cloudformation:ListStackSets",
                "cloudformation:DescribeStackInstance",
                "cloudformation:DescribeOrganizationsAccess",
                "cloudformation:ActivateOrganizationsAccess",
                "cloudformation:GetTemplate",
                "cloudformation:ListStackSetOperationResults",
                "cloudformation:DescribeStackEvents",
                "cloudformation:UntagResource",
                "ec2:DescribeInstances",
                "ssm:DescribeAutomationExecutions",
                "ssm:GetAutomationExecution",
                "ssm:ListAssociations",
                "ssm:DescribeAssociation",
                "ssm:GetDocument",
                "ssm:ListDocuments",
                "ssm:DescribeDocument",
                "ssm:ListResourceDataSync",
                "ssm:DescribePatchBaselines",
                "ssm:GetPatchBaseline",
                "ssm:DescribeMaintenanceWindows",
                "ssm:DescribeMaintenanceWindowTasks",
                "ssm:GetOpsSummary",
                "organizations:DeregisterDelegatedAdministrator",
                "organizations:DescribeAccount",
                "organizations:DescribeOrganization",
                "organizations:ListDelegatedAdministrators",
                "organizations:ListRoots",
```

```
                "organizations:ListParents",
                "organizations:ListOrganizationalUnitsForParent",
                "organizations:DescribeOrganizationalUnit",
                "organizations:ListAWSServiceAccessForOrganization",
                "s3:GetBucketLocation",
                "s3:ListAllMyBuckets",
                "s3:ListBucket",
                "resource-groups:ListGroups",
                "iam:ListRoles",
                "iam:ListRolePolicies",
                "iam:GetRole",
                "iam:CreatePolicy",
                "organizations:RegisterDelegatedAdministrator",
                "organizations:EnableAWSServiceAccess",
                "cloudformation:TagResource"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudformation:RollbackStack",
                "cloudformation:CreateStack",
                "cloudformation:UpdateStack",
                "cloudformation:DeleteStack"
            ],
            "Resource": [
                "arn:aws:cloudformation:*:*:stack/StackSet-AWS-QuickSetup-*",
                "arn:aws:cloudformation:*:*:stack/AWS-QuickSetup-*",
                "arn:aws:cloudformation:*:*:type/resource/*",
                "arn:aws:cloudformation:*:*:stack/StackSet-SSMQuickSetup"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudformation:CreateStackSet",
                "cloudformation:UpdateStackSet",
                "cloudformation:DeleteStackSet",
                "cloudformation:DeleteStackInstances",
                "cloudformation:CreateStackInstances",
                "cloudformation:StopStackSetOperation"
            ],
            "Resource": [
```

```
                "arn:aws:cloudformation:*:*:stackset/AWS-QuickSetup-*",
                "arn:aws:cloudformation:*:*:stackset/SSMQuickSetup",
                "arn:aws:cloudformation:*:*:type/resource/*",
                "arn:aws:cloudformation:*:*:stackset-target/AWS-QuickSetup-*:*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateRole",
                "iam:DeleteRole",
                "iam:AttachRolePolicy",
                "iam:DetachRolePolicy",
                "iam:GetRolePolicy",
                "iam:PutRolePolicy"
            ],
            "Resource": [
                "arn:aws:iam::*:role/AWS-QuickSetup-*",
                "arn:aws:iam::*:role/service-role/AWS-QuickSetup-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::111122223333:role/AWS-QuickSetup-*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "ssm-quicksetup.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DeleteAssociation",
                "ssm:CreateAssociation",
                "ssm:StartAssociationsOnce"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ssm:StartAutomationExecution",
```

```
            "Resource": "arn:aws:ssm:*:*:automation-definition/AWS-
EnableExplorer:*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetOpsSummary",
                "ssm:CreateResourceDataSync",
                "ssm:UpdateResourceDataSync"
            ],
            "Resource": "arn:aws:ssm:*:*:resource-data-sync/AWS-QuickSetup-*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole"
            ],
            "Condition": {
                "StringEquals": {
                    "iam:AWSServiceName": [
                        "accountdiscovery.ssm.amazonaws.com",
                        "ssm.amazonaws.com",
                        "ssm-quicksetup.amazonaws.com",
                        "stacksets.cloudformation.amazonaws.com"
                    ]
                }
            },
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole"
            ],
            "Resource": "arn:aws:iam::*:role/aws-
service-role/stacksets.cloudformation.amazonaws.com/
AWSServiceRoleForCloudFormationStackSetsOrgAdmin"
        }
    ]
}
```

Para restringir a los usuarios a permisos de solo lectura, solo permita operaciones de `ssm-quicksetup:List*` y `ssm-quicksetup:Get*` para la API de Quick Setup.

Durante la incorporación, Quick Setup crea los siguientes roles de AWS Identity and Access Management (IAM) en su nombre:

- `AWS-QuickSetup-LocalExecutionRole`: otorga permisos de AWS CloudFormation para usar cualquier plantilla, excepto la plantilla de política de revisiones, y para crear los recursos necesarios.
- `AWS-QuickSetup-LocalAdministrationRole`: Otorga a AWS CloudFormation el permiso para asumir `AWS-QuickSetup-LocalExecutionRole`.
- `AWS-QuickSetup-PatchPolicy-LocalExecutionRole`: otorga permisos a AWS CloudFormation para usar la plantilla de política de revisioens y crear los recursos necesarios.
- `AWS-QuickSetup-PatchPolicy-LocalAdministrationRole`: Otorga a AWS CloudFormation el permiso para asumir `AWS-QuickSetup-PatchPolicy-LocalExecutionRole`.

Si va a incorporar una cuenta de administración (la cuenta en la que crea una organización en AWS Organizations), Quick Setup también crea los siguientes roles en su nombre:

- `AWS-QuickSetup-SSM-RoleForEnablingExplorer`: Otorga los permisos al runbook de automatización `AWS-EnableExplorer`. El manual de procedimientos `AWS-EnableExplorer` configura Explorer, una herramienta de Systems Manager, para mostrar información de varias Cuentas de AWS y Regiones de AWS.
- `AWSServiceRoleForAmazonSSM`: Rol vinculado a un servicio que otorga el acceso a los recursos administrados de AWS y utilizados por Systems Manager.
- `AWSServiceRoleForAmazonSSM_AccountDiscovery`: un rol vinculado a servicio que otorga los permisos a Systems Manager para llamar a los Servicios de AWS para detectar información de las cuentas de Cuenta de AWS al sincronizar datos. Para obtener más información, consulte Uso de roles para recopilar información de la Cuenta de AWS para OpsCenter y Explorer.

Al incorporar una cuenta de administración, Quick Setup permite el acceso de confianza entre AWS Organizations y CloudFormation para implementar configuraciones de Quick Setup en toda la organización. Para habilitar el acceso de confianza, su cuenta de administración debe tener permisos de administrador. Tras la incorporación, ya no necesita los permisos de administrador. Para obtener más información, consulte Activar el acceso de confianza con Organizations.

Para obtener más información sobre los tipos de cuenta de AWS Organizations, consulte Conceptos y terminología de AWS Organizations en la Guía del usuario de AWS Organizations.

> **Note**
>
> Quick Setup utiliza StackSets de AWS CloudFormation para implementar las configuraciones en las Cuentas de AWS y regiones. Si el número de cuentas de destino multiplicado por el número de regiones supera las 10 000, la configuración no se implementará. Le recomendamos revisar su caso de uso y crear configuraciones que utilicen menos objetivos para adaptarse al crecimiento de su organización. Las instancias de pila no se implementan en la cuenta de administración de su organización. Para obtener mas información, consulte Consideraciones al crear un conjunto de pilas con permisos administrados por servicios.

Incorporación manual para trabajar con la API de Quick Setup de forma programática

Si utiliza la consola para trabajar con la Quick Setup, el servicio se encarga de los pasos de incorporación por usted. Si tiene pensado utilizar los SDK o la AWS CLI para trabajar con la API de Quick Setup, puede seguir utilizando la consola para que complete los pasos de incorporación por usted, de forma que no tenga que realizarlos manualmente. Sin embargo, algunos clientes tienen que completar los pasos de incorporación de la Quick Setup forma programática sin tener que interactuar con la consola. Si este método se ajusta a su caso de uso, debe realizar los siguientes pasos. Todos estos pasos deben completarse desde su cuenta de administración de AWS Organizations.

Para realizar la incorporación manual de Quick Setup

1. Active el acceso confiable para AWS CloudFormation con Organizations. Esto le proporciona a la cuenta de administración los permisos necesarios para crear y administrar StackSets para su organización. Puede usar la acción de la API de `ActivateOrganizationsAccess` de AWS CloudFormation para completar este paso. Para obtener más información, consulte ActivateOrganizationsAccess en la Referencia de la API de AWS CloudFormation.
2. Habilite la integración de Systems Manager con Organizations. Esto permite que Systems Manager cree un rol vinculado a servicios en todas las cuentas de su organización. También le permite realizar operaciones en su nombre en su organización y sus cuentas. Puede usar la acción de la API de `EnableAWSServiceAccess` de AWS Organizations para completar este paso. La entidad principal de servicio de Systems Manager es `ssm.amazonaws.com`. Para

obtener más información, consulte EnableAWSServiceAccess en la referencia de la API de AWS Organizations

3. Cree el rol de IAM requerido para Explorer. Esto permite que Quick Setup cree paneles de control para sus configuraciones, de modo que pueda ver los estados de implementación y asociación. Cree un rol de IAM y adjúntelo a la política administrada de `AWSSystemsManagerEnableExplorerExecutionPolicy`. Modifique la política de confianza del rol para que coincida con lo siguiente. Reemplace cada *ID de cuenta* con su propia información.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "ssm.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "account ID"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:*:ssm:*:111122223333:automation-
execution/*"
                }
            }
        }
    ]
}
```

4. Actualice la configuración del servicio de la Quick Setup para Explorer. Puede usar la acción de la API de `UpdateServiceSettings` de Quick Setup para completar este paso. Especifique el ARN para el rol de IAM que creó en el paso anterior para el parámetro de solicitud de `ExplorerEnablingRoleArn`. Para obtener más información, consulte UpdateServiceSettings en la referencia de la API de Quick Setup
5. Crea los roles de IAM necesarios para que los usen los StackSets de AWS CloudFormation. Debe crear un rol de ejecución y un rol de administración.

a. Creación del rol de ejecución. El rol de ejecución debe tener al menos una de las políticas administradas de `AWSQuickSetupDeploymentRolePolicy` o de `AWSQuickSetupPatchPolicyDeploymentRolePolicy` asociadas. Si solo está creando configuraciones de políticas de revisiones, puede usar la política administrada de `AWSQuickSetupPatchPolicyDeploymentRolePolicy`. Todas las demás configuraciones utilizan la política de `AWSQuickSetupDeploymentRolePolicy`. Modifique la política de confianza del rol para que coincida con lo siguiente. Reemplace cada *`ID de cuenta`* y *`nombre de rol de administración`* con su propia información.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/administration
 role name"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

b. Cree el rol de administración. La política de permisos debe coincidir con lo siguiente. Reemplace cada *`ID de cuenta`* y *`nombre de rol de ejecución`* con su propia información.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "sts:AssumeRole"
            ],
```

```
            "Resource": "arn:*:iam::111122223333:role/execution role
 name",
            "Effect": "Allow"
        }
    ]
}
```

Modifique la política de confianza del rol para que coincida con lo siguiente. Reemplace cada *ID de cuenta* con su propia información.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "cloudformation.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "111122223333"
                },
                "ArnLike": {
                    "aws:SourceArn":
 "arn:aws:cloudformation:*:111122223333:stackset/AWS-QuickSetup-*"
                }
            }
        }
    ]
}
```

Uso de un administrador delegado para Quick Setup

Tras registrar una cuenta de administrador delegado para Quick Setup, los usuarios con los permisos adecuados en esa cuenta pueden crear, actualizar, ver y eliminar administradores de configuración que tienen como destino las unidades organizativas de su estructura AWS Organizations. Esta

cuenta de administrador delegado también puede gestionar los administradores de configuración creados por la cuenta de administración de su organización.

En Organizations, la cuenta de administración puede designar una cuenta en su organización como administrador delegado. Cuando registra una cuenta como administrador delegado para Quick Setup, esta cuenta también se convierte automáticamente en administrador delegado para AWS CloudFormation StackSets y Systems Manager Explorer, ya que estos servicios son necesarios para implementar y supervisar las configuraciones de Quick Setup.

> **Note**
>
> En este momento, el administrador delegado para Quick Setup no admite el tipo de configuración de la política de revisiones. Las configuraciones de políticas de revisión para una organización deben crearse y mantenerse en la cuenta de administración de una organización. Para obtener más información, consulte the section called “Cómo crear una política de revisiones”.

En los temas siguientes se describe cómo registrar un administrador delegado para Quick Setup y cómo anular su registro.

Temas

- Registro de un administrador delegado para Quick Setup
- Anulación del registro de un administrador delegado para Quick Setup

Registro de un administrador delegado para Quick Setup

Utilice el procedimiento siguiente para registrar un administrador delegado para Quick Setup.

Registro de un administrador delegado de Quick Setup

1. Inicie sesión en la cuenta de administración de AWS Organizations.
2. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
3. En el panel de navegación, elija Quick Setup.
4. Elija Configuración.

5. En la sección Administrador delegado para Quick Setup, compruebe que ha configurado el rol vinculada al servicio y las opciones de acceso al servicio necesarias. Si es preciso, elija los botones Create rol (Crear rol) y Enable access (Habilitar acceso) para configurar estas opciones.
6. En Account ID (ID de cuenta), ingrese el ID de la Cuenta de AWS. Esta cuenta debe ser una cuenta de miembro en AWS Organizations.
7. Elija Registrar administrador delegado.

Anulación del registro de un administrador delegado para Quick Setup

Utilice el procedimiento siguiente para anular el registro de un administrador delegado para Quick Setup.

Anulación del registro de un administrador delegado de Quick Setup

1. Inicie sesión en la cuenta de administración de AWS Organizations.
2. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
3. En el panel de navegación, elija Quick Setup.
4. Elija Configuración.
5. En la sección Administrador delegado para Quick Setup, elija Anular registro en el menú desplegable Acciones.
6. Seleccione Confirmar.

Aprenda la terminología y detalles de Quick Setup

Quick Setup, una herramienta de AWS Systems Manager, muestra los resultados de todos los administradores de configuración que ha creado en todas las Regiones de AWS en la tabla Administradores de configuración de la página de inicio de Quick Setup. En esta página, puede seleccionar View details para ver los detalles de cada configuración, eliminar configuraciones del menú desplegable Actions (Acciones) o seleccionar Create para crear configuraciones. Esta tabla de Administradores de configuración contiene la siguiente información:

- Nombre: el nombre del administrador de la configuración, si se proporcionó al crearlo.
- Configuration type (Tipo de configuración): el tipo de configuración que se ha elegido al crear la configuración.
- Versión: la versión del tipo de configuración actualmente implementada.

- Organizational units (Unidades organizativas): muestra las unidades organizativas (OU) en las que se ha implementado la configuración si elige un conjunto de destinos Custom (Personalizados). Las unidades organizativas y los destinos personalizados solo están disponibles para la cuenta de administración de su organización. La cuenta de administración es la cuenta que usa para crear la organización en AWS Organizations.
- Tipo de implementación: indica si la implementación se aplica a toda la organización (`Organizational`) o solo a su cuenta (`Local`).
- Regions (Regiones): las regiones en las que se implementa la configuración si elige un conjunto de destinos Custom (Personalizados) o destinos dentro de su Current account (Cuenta actual).
- Deployment status (Estado de la implementación): el estado de implementación indica si AWS CloudFormation ha implementado correctamente la instancia de destino o pila. Las instancias de destino y pila contienen las opciones de configuración que eligió durante la creación de la configuración.
- Association status (Estado de asociación): el estado de asociación es el estado de todas las asociaciones creadas por la configuración que ha creado. Todas las asociaciones para todos los destinos deben ejecutarse correctamente; de lo contrario, el estado es Failed (Error).

  Quick Setup crea y ejecuta una asociación State Manager para cada destino de configuración. State Manager es una herramienta de AWS Systems Manager.

Para ver las configuraciones implementadas en la región por la que está navegando actualmente, seleccione la pestaña Configuraciones.

Detalles de configuración

La página Configuration details (Detalles de configuración) muestra información sobre la implementación de la configuración y las asociaciones relacionadas. Desde esta página puede editar opciones de configuración, actualizar destinos o eliminar la configuración. También puede ver los detalles de cada implementación de configuración para obtener más información sobre las asociaciones.

Según el tipo de configuración, se muestran uno o más de los siguientes gráficos de estado:

Estado de la implementación de configuración

Muestra la cantidad de implementaciones que se han realizado correctamente, han fallado, se están ejecutando o están pendientes. Las implementaciones se producen en las cuentas de destino especificadas y en las regiones que contienen nodos afectados por la configuración.

Estado de asociación de configuración

Muestra la cantidad de asociaciones de State Manager que se han realizado correctamente, han fallado o están pendientes. Quick Setup crea una asociación en cada implementación para las opciones de configuración seleccionadas.

Estado de configuración

Muestra el número de acciones realizadas por el tipo de configuración y sus estados actuales.

Conformidad de recursos

Muestra el número de recursos que cumplen con la política de configuración especificada.

La tabla Configuration details (Detalles de configuración) muestra información sobre la implementación de la configuración. Para ver más detalles sobre cada implementación, seleccione la implementación y, a continuación, elija View details (Ver detalles). En la página de detalles de cada implementación, se muestran las asociaciones implementadas en los nodos de esa implementación.

Edición y eliminación de la configuración

Puede editar las opciones de configuración de una configuración desde la página Configuration details (Detalles de configuración) al elegir Actions (Acciones) y, luego, Edit configuration options (Editar opciones de configuración). Al agregar nuevas opciones a la configuración, Quick Setup ejecuta las implementaciones y crea nuevas asociaciones. Al quitar opciones de una configuración, Quick Setup ejecuta las implementaciones y elimina cualquier asociación relacionada.

> **Note**
>
> Puede editar las configuraciones de Quick Setup para su cuenta en cualquier momento. Para editar la configuración de una Organization (Organización), el Configuration status (Estado de configuración) debe ser Success (Correcto) o Failed (Error).

También puede actualizar los destinos que se incluyen en las configuraciones si elige Actions (Acciones) y Add OUs (Agregar unidades organizativas), Add Regions (Agregar regiones), Remove OUs (Eliminar unidades organizativas), o Remove Regions (Eliminar regiones). Si su cuenta no está configurada como cuenta de administración o ha creado la configuración solo para la cuenta actual, no puede actualizar las unidades organizativas (OU) de destino. Al quitar una región o una unidad organizativa, se eliminan las asociaciones de esas regiones o unidades organizativas.

Quick Setup publica nuevas versiones de las configuraciones de forma periódica. Puede seleccionar la opción Actualizar la configuración para actualizar la configuración a la versión más reciente.

Puede eliminar una configuración de Quick Setup al elegir la configuración, Actions (Acciones) y, luego, Delete configuration (Eliminar la configuración). O bien puede eliminar la configuración de la página Configuration details (Detalles de configuración) en el menú desplegable Actions (Acciones), eligiendo Delete configuration (Eliminar la configuración). Quick Setup luego solicita Remove all OUs and Regions (Eliminar todas las unidades organizativas y regiones), que puede tardar un poco en llevarse a cabo. Al eliminar una configuración también se eliminan todas las asociaciones relacionadas. Este proceso de eliminación en dos pasos elimina todos los recursos implementados de todas las cuentas y regiones y, a continuación, elimina la configuración.

Conformidad de la configuración

Puede ver si sus instancias cumplen con las asociaciones creadas en las configuraciones en Explorer o Cumplimiento, que son herramientas de AWS Systems Manager. Para obtener más información sobre la conformidad, consulte Detalles sobre el cumplimiento. Para obtener más información sobre la conformidad en Explorer, consulte AWS Systems Manager Explorer.

Uso de la API Quick Setup para gestionar las configuraciones y las implementaciones

Se puede usar la API proporcionada por Quick Setup para crear y administrar configuraciones e implementaciones con la AWS CLI o con el SDK que se prefiera. También puede utilizar AWS CloudFormation para crear un recurso de administrador de configuración que implemente las configuraciones. Con la API, se crean administradores de configuración que implementan definiciones de configuración. Las definiciones de configuración contienen toda la información necesaria para implementar un tipo de configuración concreto. Para obtener más información sobre la API de Quick Setup consulte la Referencia de la API de Configuración Rápida.

Los siguientes ejemplos muestran cómo crear administradores de configuración mediante la AWS CLI y AWS CloudFormation.

AWS CLI

```
aws ssm-quicksetup create-configuration-manager \
--name configuration manager name \
--description Description of your configuration manager
--configuration-definitions JSON string containing configuration defintion
```

El siguiente es un ejemplo de cadena JSON que contiene una definición de configuración para la política de revisioens.

```
'{"Type":"AWSQuickSetupType-
PatchPolicy","LocalDeploymentAdministrationRoleArn":"arn:aws:iam::123456789012:role/
AWS-QuickSetup-StackSet-Local-
AdministrationRole","LocalDeploymentExecutionRoleName":"AWS-
QuickSetup-StackSet-Local-ExecutionRole","Parameters":
{"ConfigurationOptionsInstallNextInterval":"true","ConfigurationOptionsInstallValue":"cron(0
 2 ? * SAT#1
 *)","ConfigurationOptionsPatchOperation":"ScanAndInstall","ConfigurationOptionsScanNextInte
 1 * * ?
 *)","HasDeletedBaseline":"false","IsPolicyAttachAllowed":"true","OutputBucketRegion":"","Ou
east-1","PatchBaselineUseDefault":"custom","PatchPolicyName":"dev-patch-
policy","RateControlConcurrency":"5","RateControlErrorThreshold":"0%","RebootOption":"Reboot
\":{\"value\":\"arn:aws:ssm:us-east-1:123456789012:patchbaseline/
pb-0cb0c4966f86b059b\",\"label\":\"AWS-AlmaLinuxDefaultPatchBaseline\",\"description
\":\"Default Patch Baseline for Alma Linux Provided by AWS.\",\"disabled\":false},
\"AMAZON_LINUX\":{\"value\":\"arn:aws:ssm:us-east-1:123456789012:patchbaseline/
pb-0c10e657807c7a700\",\"label\":\"AWS-AmazonLinuxDefaultPatchBaseline\",
\"description\":\"Default Patch Baseline for Amazon Linux Provided by AWS.
\",\"disabled\":false},\"AMAZON_LINUX_2\":{\"value\":\"arn:aws:ssm:us-
east-1:123456789012:patchbaseline/pb-0be8c61cde3be63f3\",\"label\":\"AWS-
AmazonLinux2DefaultPatchBaseline\",\"description\":\"Baseline containing
 all Security and Bugfix updates approved for Amazon Linux 2 instances
\",\"disabled\":false},\"AMAZON_LINUX_2022\":{\"value\":\"arn:aws:ssm:us-
east-1:123456789012:patchbaseline/pb-0028ca011460d5eaf\",\"label\":\"AWS-
AmazonLinux2022DefaultPatchBaseline\",\"description\":\"Default Patch
 Baseline for Amazon Linux 2022 Provided by AWS.\",\"disabled\":false},
\"AMAZON_LINUX_2023\":{\"value\":\"arn:aws:ssm:us-east-1:123456789012:patchbaseline/
pb-05c9c9bf778d4c4d0\",\"label\":\"AWS-AmazonLinux2023DefaultPatchBaseline
\",\"description\":\"Default Patch Baseline for Amazon Linux 2023 Provided
 by AWS.\",\"disabled\":false},\"CENTOS\":{\"value\":\"arn:aws:ssm:us-
east-1:123456789012:patchbaseline/pb-03e3f588eec25344c\",\"label\":\"AWS-
CentOSDefaultPatchBaseline\",\"description\":\"Default Patch Baseline for CentOS
 Provided by AWS.\",\"disabled\":false},\"DEBIAN\":{\"value\":\"arn:aws:ssm:us-
east-1:123456789012:patchbaseline/pb-09a5f8eb62bde80b1\",\"label\":\"AWS-
DebianDefaultPatchBaseline\",\"description\":\"Default Patch Baseline for Debian
 Provided by AWS.\",\"disabled\":false},\"MACOS\":{\"value\":\"arn:aws:ssm:us-
east-1:123456789012:patchbaseline/pb-0ee4f94581368c0d4\",\"label\":\"AWS-
MacOSDefaultPatchBaseline\",\"description\":\"Default Patch Baseline for
 MacOS Provided by AWS.\",\"disabled\":false},\"ORACLE_LINUX\":{\"value\":
\"arn:aws:ssm:us-east-1:123456789012:patchbaseline/pb-06bff38e95fe85c02\",
```

```
\"label\":\"AWS-OracleLinuxDefaultPatchBaseline\",\"description\":\"Default
 Patch Baseline for Oracle Linux Server Provided by AWS.\",\"disabled\":false},
\"RASPBIAN\":{\"value\":\"arn:aws:ssm:us-east-1:123456789012:patchbaseline/
pb-0ec16280999c5c75e\",\"label\":\"AWS-RaspbianDefaultPatchBaseline\",
\"description\":\"Default Patch Baseline for Raspbian Provided by AWS.\",
\"disabled\":false},\"REDHAT_ENTERPRISE_LINUX\":{\"value\":\"arn:aws:ssm:us-
east-1:123456789012:patchbaseline/pb-0cbb3a633de00f07c\",\"label\":\"AWS-
RedHatDefaultPatchBaseline\",\"description\":\"Default Patch Baseline for Redhat
 Enterprise Linux Provided by AWS.\",\"disabled\":false},\"ROCKY_LINUX\":{\"value
\":\"arn:aws:ssm:us-east-1:123456789012:patchbaseline/pb-03ec98bc512aa3ac0\",\"label
\":\"AWS-RockyLinuxDefaultPatchBaseline\",\"description\":\"Default Patch Baseline
 for Rocky Linux Provided by AWS.\",\"disabled\":false},\"SUSE\":{\"value\":
\"arn:aws:ssm:us-east-1:123456789012:patchbaseline/pb-07d8884178197b66b\",\"label\":
\"AWS-SuseDefaultPatchBaseline\",\"description\":\"Default Patch Baseline for Suse
 Provided by AWS.\",\"disabled\":false},\"UBUNTU\":{\"value\":\"pb-06e3563bd35503f2b
\",\"label\":\"custom-UbuntuServer-Blog-Baseline\",\"description\":\"Default Patch
 Baseline for Ubuntu Provided by AWS.\",\"disabled\":false},\"WINDOWS\":{\"value
\":\"pb-016889927b2bb8542\",\"label\":\"custom-WindowsServer-Blog-Baseline\",
\"disabled\":false}}","TargetInstances":"","TargetOrganizationalUnits":"ou-9utf-
example","TargetRegions":"us-east-1,us-
east-2","TargetTagKey":"Patch","TargetTagValue":"true","TargetType":"Tags"}}' \
```

AWS CloudFormation

```
AWSTemplateFormatVersion: '2010-09-09'
Resources:
SSMQuickSetupTestConfigurationManager:
Type: "AWS::SSMQuickSetup::ConfigurationManager"
Properties:
    Name: "MyQuickSetup"
    Description: "Test configuration manager"
    ConfigurationDefinitions:
    - Type: "AWSQuickSetupType-CFGRecording"
      Parameters:
        TargetAccounts:
            Ref: AWS::AccountId
        TargetRegions:
            Ref: AWS::Region
        LocalDeploymentAdministrationRoleArn: !Sub "arn:aws:iam::
${AWS::AccountId}:role/AWS-QuickSetup-StackSet-ContractTest-AdministrationRole"
        LocalDeploymentExecutionRoleName: "AWS-QuickSetup-StackSet-ContractTest-
ExecutionRole"
    Tags:
```

```
        foo1: "bar1"
```

Tipos de configuración Quick Setup compatibles

Tipos de configuración compatibles

Quick Setup explica cómo configurar las prácticas recomendadas operativas para una serie de servicios de Systems Manager y otros Servicios de AWS, así como la forma de implementar automáticamente dichas configuraciones. El panel de Quick Setup muestra una vista en tiempo real del estado de implementación de la configuración.

Puede utilizar Quick Setup en una Cuenta de AWS individual o en varias regiones y AWS Organizations mediante la integración con Cuentas de AWS. El uso de Quick Setup en varias cuentas ayuda a garantizar que su organización mantenga configuraciones coherentes.

Quick Setup proporciona soporte para los siguientes tipos de configuración.

- Instalar la administración de host de Amazon EC2 mediante Quick Setup
- Instalar la Configuración de administración de hosts predeterminada de una organización mediante Quick Setup
- Cree un registrador de configuración de AWS Config mediante Quick Setup
- Implemente el paquete de conformidad de AWS Config mediante Quick Setup
- Cómo configurar las revisiones para las instancias de una organización mediante una política de parches Quick Setup
- Configuración de la organización de Change Manager
- Configurar DevOps Guru con Quick Setup
- Implemente paquetes de Distributor mediante Quick Setup
- Detenga e inicie las instancias EC2 automáticamente según una programación mediante Quick Setup
- Configuración de la organización de OpsCenter
- Configuración de Explorador de recursos de AWS mediante Quick Setup

Instalar la administración de host de Amazon EC2 mediante Quick Setup

Utilice Quick Setup, una herramienta de AWS Systems Manager, para configurar rápidamente los roles de seguridad necesarios y las herramientas de Systems Manager de uso común en

sus instancias de Amazon Elastic Compute Cloud (Amazon EC2). Puede utilizar Quick Setup en una cuenta individual o en varias cuentas y Regiones de AWS mediante la integración con AWS Organizations. Estas herramientas le ayudan a administrar y supervisar el estado de las instancias, a la vez que proporcionan los permisos mínimos necesarios para dar los primeros pasos.

Si no está familiarizado con los servicios y las características de Systems Manager, se recomienda que revise la Guía del usuario de AWS Systems Manager antes de crear una configuración con Quick Setup. Para obtener más información acerca de Systems Manager, consulte ¿Qué es AWS Systems Manager?.

Important

Quick Setup podría no ser la herramienta adecuada para la administración de EC2 si se da alguno de los siguientes casos:

- Intenta crear una instancia de EC2 por primera vez para probar las funciones de AWS.
- No sabe bien cómo funciona la administración de instancias de EC2.

Por ello, le recomendamos que consulte el siguiente contenido:

- Introducción a Amazon EC2
- Inicialización de una instancia mediante el nuevo asistente de inicialización de instancias en la Guía del usuario de Amazon EC2
- Tutorial: Introducción a las instancias de Linux de Amazon EC2 en la Guía del usuario de Amazon EC2

Si ya sabe cómo funciona la administración de instancias de EC2 y desea optimizar la configuración y la administración de varias instancias de EC2, utilice Quick Setup. Ya sea que su organización tenga docenas, miles o millones de instancias de EC2, puede utilizar el siguiente procedimiento de Quick Setup para configurar varias opciones a la vez.

Note

Este tipo de configuración permite establecer varias opciones para toda una organización definida en AWS Organizations, solo algunas cuentas y regiones de la organización, o una sola cuenta. Una de estas opciones consiste en comprobar cada dos semanas si

existen actualizaciones de SSM Agent y aplicarlas. Si es administrador de la organización, también puede optar por actualizar todas las instancias de EC2 de su organización con actualizaciones de los agentes cada dos semanas mediante el tipo Configuración de la administración de hosts predeterminada. Para obtener más información, consulta Instalar la Configuración de administración de hosts predeterminada de una organización mediante Quick Setup.

Configuración de las opciones de administración de hosts para instancias de EC2

Para configurar la administración de host, lleve a cabo las siguientes tareas en la consola de AWS Systems Manager Quick Setup.

Apertura de la página de configuración del Administrador de host

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Quick Setup.
3. En la tarjeta de Administración de host, elija Crear.

> **Tip**
>
> Si ya tiene una o más configuraciones en su cuenta, seleccione primero la pestaña Biblioteca o el botón Crear de la sección Configuraciones para ver las tarjetas.

Configuración de las opciones de la administración de hosts de Systems Manager

- Para configurar la funcionalidad de Systems Manager, en la sección Opciones de configuración, dentro del grupo Systems Manager, seleccione las opciones que quiera habilitar para la configuración:

  Actualizar el agente de Systems Manager (SSM) cada dos semanas

  Le permite a Systems Manager verificar cada dos semanas si hay una nueva versión disponible del agente. Si hay una nueva versión, Systems Manager actualiza automáticamente el agente en el nodo administrado a la versión más reciente publicada.

Quick Setup no instala el agente en instancias donde no está presente. Para obtener información sobre las AMIs que SSM Agent tiene preinstaladas, consulte Búsqueda de AMIs con SSM Agent preinstalado.

Le recomendamos que elija esta opción para asegurarse de que los nodos siempre ejecuten la versión más actualizada de SSM Agent. Para obtener más información acerca de SSM Agent, incluida información sobre cómo instalar manualmente el agente, consulte Uso de SSM Agent.

Recopilar el inventario de las instancias cara 30 minutos

Le permite a Quick Setup configurar la recopilación de los tipos de metadatos que se mencionan a continuación:

- Componentes de AWS: controlador de EC2, agentes, versiones y más.
- Aplicaciones: nombres de aplicaciones, editores, versiones y más.
- Detalles del nodo: nombre del sistema, nombre del sistema operativo (OS), versión del OS, último inicio, DNS, dominio, grupo de trabajo, arquitectura del OS y más.
- Configuración de red: dirección IP, dirección MAC, DNS, gateway, máscara de subred y más.
- Servicios: nombre, nombre de visualización, estado, servicios dependientes, tipo de servicio, tipo de inicio y más (solo en nodos de Windows Server).
- Roles de Windows: nombre, nombre de visualización, ruta, tipo de característica, estado instalado y más (solo en nodos de Windows Server).
- Actualizaciones de Windows: ID de hotfix, instalado por, fecha de instalación y más (solo en nodos de Windows Server).

Para obtener más información sobre Inventario, una herramienta de AWS Systems Manager, consulte Inventario de AWS Systems Manager.

> **Note**
>
> La opción Recopilación de inventario puede tardar hasta 10 minutos en completarse, aunque solo haya seleccionado algunos nodos.

Analizar a diario las instancias para identificar los parches que faltan

Permite a Patch Manager, una herramienta de Systems Manager, analizar los nodos a diario y generar un informe en la página Conformidad. El informe muestra cuántos nodos son conformes con las revisiones de acuerdo con la base de referencia de revisiones predeterminada. El informe incluye una lista de cada nodo y su estado de conformidad.

Para obtener información acerca de las operaciones de aplicación de revisiones y las líneas de base de revisiones, consulte AWS Systems Manager Patch Manager.

Para obtener información acerca de la conformidad de revisiones, consulte la página Conformidad de Systems Manager.

Para obtener información sobre cómo aplicar revisiones a los nodos administrados en varias cuentas y regiones en una configuración, consulte Configuraciones de políticas de revisiones en Quick Setup y Cómo configurar las revisiones para las instancias de una organización mediante una política de parches Quick Setup.

> **Important**
>
> Systems Manager admite varios métodos para analizar nodos administrados y así comprobar la conformidad de revisiones. Si implementa más de uno de estos métodos a la vez, la información de conformidad de las revisiones que ve siempre será el resultado del análisis más reciente. Los resultados de análisis anteriores se sobrescriben. Si los métodos de análisis utilizan diferentes líneas de base de revisiones con diferentes reglas de aprobación, la información de conformidad de revisiones puede cambiar inesperadamente. Para obtener más información, consulte Evitar sobrescrituras involuntarias de datos de conformidad de revisiones.

Configuración de las opciones de administración de hosts de Amazon CloudWatch

- Para configurar la funcionalidad de CloudWatch, dentro de la sección Opciones de configuración, en el grupo Amazon CloudWatch, seleccione las opciones que quiera habilitar para la configuración:

Instalación y configuración del agente de CloudWatch

Instala la configuración básica del agente unificado de CloudWatch en las instancias de Amazon EC2. El agente recopila métricas y archivos de registro de las instancias de Amazon CloudWatch. Esta información se consolida para que pueda determinar rápidamente el estado de las instancias. Para obtener más información sobre la configuración básica del agente de CloudWatch, consulte Conjuntos de métricas predefinidas del agente de CloudWatch. Es posible que haya un costo adicional. Para más información, consulte Precios de Amazon CloudWatch.

Actualizar el agente de CloudWatch una vez cada 30 días

Le permite a Systems Manager verificar cada 30 días si hay una nueva versión disponible del agente de CloudWatch. Si hay una nueva versión, Systems Manager actualiza automáticamente el agente en la instancia. Lo animamos a que elija esta opción para asegurarse de que sus instancias siempre ejecuten la versión más actualizada del agente de CloudWatch.

Configuración de las opciones de administración de hosts de Amazon EC2 Launch Agent

- Para configurar la funcionalidad de Amazon EC2 Launch Agent, dentro de la sección Opciones de configuración, en el grupo Amazon EC2 Launch Agent, seleccione las opciones que quiera habilitar para la configuración:

Actualizar el agente de ejecución de EC2 una vez cada 30 días

Le permite a Systems Manager verificar cada 30 días si hay una nueva versión del agente de ejecución instalada en la instancia. Si hay una nueva versión disponible, Systems Manager actualiza el agente en la instancia. Lo animamos a que elija esta opción para asegurarse de que sus instancias siempre ejecuten la versión más actualizada del agente de lanzamiento correspondiente. Para las instancias de Windows de Amazon EC2, esta opción es compatible con EC2Launch, EC2Launch v2 y EC2Config. Para las instancias de Linux de Amazon EC2, esta opción es compatible con `cloud-init`. Para las instancias Mac de Amazon EC2, esta opción es compatible con `ec2-macos-init`. Quick Setup no admite la actualización de los agentes de lanzamiento que estén instalados en sistemas operativos no compatibles con el agente de lanzamiento o en AL2023.

Para obtener más información sobre estos agentes de inicialización, consulte los siguientes temas:

- Configurar una instancia de Windows mediante EC2Launch v2
- Configurar una instancia de Windows mediante EC2Launch
- Configurar una instancia de Windows mediante el servicio de EC2Config
- Documentación de cloud-init
- ec2-macos-init

Selección de las instancias de EC2 que se actualizarán con la configuración de la administración de hosts

- En la sección Destinos, seleccione el método para determinar las cuentas y las regiones en las que se implementará la configuración:

> **Note**
>
> No puede crear varias configuraciones de Administración de host de Quick Setup que tengan como destino a la misma Región de AWS.

Entire organization

La configuración se implementará en todas las unidades organizativas (UO) y en las Regiones de AWS de su organización.

> **Note**
>
> La opción Organización completa solo está disponible si configura la administración de host desde la cuenta de administración de su organización.

Custom

1. En la sección UO de destino, seleccione las UO en las que quiere implementar esta configuración de la administración de hosts.

2. En la sección Regiones de destino, seleccione la región en la que quiere implementar la configuración de la administración de hosts.

Current account

Seleccione una de las opciones de región y siga los pasos de esa opción.

Región actual

Seleccione cómo elegir instancias como destino solo dentro de la región actual:

- Todas las instancias: la configuración de la administración de hosts selecciona como destino, de manera automática, cada EC2 dentro de la región actual.
- Etiqueta: seleccione Agregar e ingrese la clave y un valor opcional que se agregue a las instancias que serán el destino.
- Grupo de recursos: en Grupo de recursos, seleccione un grupo de recursos existente que contenga las instancias de EC2 que serán el destino.
- Manual: en la sección Instancias, seleccione la casilla de verificación para cada instancia de EC2 que será el destino.

Elección de regiones

Utilice una de las opciones que se enumeran a continuación para seleccionar cómo elegir instancias como destino en la región especificada:

- Todas las instancias: todas las instancias de las regiones especificadas se eligen como destino.
- Etiqueta: seleccione Agregar e ingrese la clave y un valor opcional que se haya agregado a las instancias que serán el destino.

En la sección Regiones de destino, seleccione la región en la que quiere implementar la configuración de la administración de hosts.

Especificación de una opción de perfil de instancia

- Solo para los destinos Toda la organización y Personalizado.

En la sección Instance profile options (Opciones de perfil de instancia), elija si desea agregar las políticas de IAM necesarias a los perfiles de instancias existentes asociados a sus instancias o permitir que Quick Setup cree las políticas de IAM y los perfiles de instancias con los permisos necesarios para la configuración que elija.

Luego de especificar todas las opciones de configuración, seleccione Crear.

Instalar la Configuración de administración de hosts predeterminada de una organización mediante Quick Setup

Con Quick Setup, una herramienta de AWS Systems Manager, puede activar la Configuración de la administración de hosts predeterminada para todas las cuentas y regiones que se hayan agregado a su organización en AWS Organizations. Esto garantiza que SSM Agent se mantenga actualizado en todas las instancias de Amazon Elastic Compute Cloud (EC2) de la organización, y que estas se puedan conectar a Systems Manager.

Antes de empezar

Antes de habilitar esta configuración, compruebe que se cumplen los siguientes requisitos.

- La última versión de SSM Agent ya está instalada en todas las instancias de EC2 que hay que administrar en su organización.
- Las instancias de ECS que hay que administrar están utilizando el Servicio de metadatos de instancia versión 2 (IMDSv2).
- Ha iniciado sesión en la cuenta de administración de su organización, tal como se especifica en AWS Organizations, mediante una identidad (usuario, rol o grupo) de AWS Identity and Access Management (IAM) con permisos de administrador.

Uso del rol de administración de instancias de EC2 predeterminado

La Configuración de la administración de hosts predeterminada utiliza la configuración del servicio `default-ec2-instance-management-role` para Systems Manager. Se trata de un rol con permisos que desea que estén disponibles en todas las cuentas de su organización para permitir la comunicación entre SSM Agent de la instancia y el servicio Systems Manager en la nube.

Si ya ha establecido este rol mediante el comando update-service-setting de la CLI, la Configuración de la administración de hosts predeterminada lo utiliza. Si aún no ha establecido este rol, Quick Setup se encargará de crearlo y aplicarlo.

Para comprobar si este rol ya se ha especificado para su organización, utilice el comando get-service-setting.

Habilitación de actualizaciones automáticas de SSM Agent cada dos semanas

Utilice el siguiente procedimiento para habilitar la opción de la Configuración de la administración de hosts predeterminada para toda la organización de AWS Organizations.

Para habilitar las actualizaciones automáticas de SSM Agent cada dos semanas

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Quick Setup.
3. En la tarjeta Configuración de la administración de hosts predeterminada, seleccione Crear.

> **Tip**
>
> Si ya tiene una o más configuraciones en su cuenta, seleccione primero la pestaña Biblioteca o el botón Crear de la sección Configuraciones para ver las tarjetas.

4. En la sección Opciones de configuración, seleccione Habilitar actualizaciones automáticas de SSM Agent cada dos semanas.
5. Elija Create (Creación de).

Cree un registrador de configuración de AWS Config mediante Quick Setup

Con Quick Setup, una herramienta de AWS Systems Manager, puede crear rápidamente un registro de configuración basado en AWS Config. Utilice el registro de configuración para detectar los cambios en las configuraciones del recurso y capturar los cambios como elementos de configuración. Si no está familiarizado con AWS Config, recomendamos obtener más información acerca del servicio. Para ello, consulte el contenido en la Guía para desarrolladores de AWS Config antes de crear una configuración con Quick Setup. Para obtener más información sobre AWS Config, consulte ¿Qué es AWS Config? en la Guía para desarrolladores de AWS Config.

De forma predeterminada, el registro de configuración registra todos los recursos admitidos en la Región de AWS donde se ejecuta AWS Config. Puede personalizar la configuración para que solo se registren los tipos de recursos que especifique. Para obtener más información, consulte Selección de los recursos que registra AWS Config en la Guía para desarrolladores de AWS Config.

Se le cobrarán las tarifas de uso del servicio cuando AWS Config comience a registrar configuraciones. Para obtener información sobre precios, consulte Precios de AWS Config.

> **Note**
>
> Si ya creó un registro de configuración, Quick Setup no detiene el registro ni realiza ningún cambio en los tipos de recursos que ya está registrando. Si decide registrar otros tipos de recursos con Quick Setup, el servicio los anexará a sus grupos de registradores existentes. Eliminar el tipo de configuración de Quick Setup Registro de configuración no detiene el registro de la configuración. Los cambios se siguen registrando y se aplican tarifas por uso del servicio hasta que detenga el registro de configuración. Para obtener más información acerca de la administración del registro de configuración, consulte Administración del registro de configuración en la Guía para desarrolladores de AWS Config.

Para configurar el registro de AWS Config, realice las siguientes tareas en la consola de AWS Systems Manager.

Para configurar el registro de AWS Config con Quick Setup

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Quick Setup.
3. En la tarjeta Registro de configuración, elija Crear.

   > **Tip**
   >
   > Si ya tiene una o más configuraciones en su cuenta, seleccione primero la pestaña Biblioteca o el botón Crear de la sección Configuraciones para ver las tarjetas.

4. En la sección Opciones de configuración, haga lo siguiente:

   a. En Elegir los tipos de recursos de AWS para registrar, especifique si desea registrar todos los recursos admitidos o solo los tipos de recursos que elija.

   b. En Configuración de entrega, especifique si desea crear un nuevo bucket de Amazon Simple Storage Service (Amazon S3) o elija un bucket existente al que enviar instantáneas de configuración.

c. En Opciones de notificación, elija la opción de notificación que prefiera. AWS Config utiliza Amazon Simple Notification Service (Amazon SNS) para enviar notificaciones sobre eventos de AWS Config importantes relacionados con los recursos. Si elige la opción Use existing SNS topics (Utilizar temas existentes de SNS), debe proporcionar el ID de la Cuenta de AWS y el nombre del tema de Amazon SNS existente en la cuenta que desea utilizar. Si apunta a varias Regiones de AWS, los nombres de los temas deben ser idénticos en cada región.

5. En la sección Schedule (Programación), elija la frecuencia con la que desea que Quick Setup corrija los cambios realizados en los recursos que difieren de la configuración. La opción Default (Predeterminado) se ejecuta una vez. Si no desea que Quick Setup corrija los cambios realizados en los recursos que difieren de la configuración, elija Disable remediation (Deshabilitar la corrección) en Custom (Personalizar).
6. En la sección Destinos, elija una de las siguientes opciones para identificar las cuentas y regiones para registrar.

> **Note**
>
> Si trabaja en una sola cuenta, las opciones para trabajar con organizaciones y unidades organizativas (OU) no están disponibles. Puede elegir si desea aplicar esta configuración a todas las Regiones de AWS de su cuenta o solo a las regiones que seleccione.

- Entire organization (Toda la organización): todas las cuentas y regiones de su organización.
- Custom (Personalizado): solo las unidades organizativas y las regiones que especifique.
  - En la sección Unidades organizativas de destino, seleccione las unidades organizativas donde desea permitir el registro.
  - En la sección Regiones de destino, seleccione las regiones donde desea permitir el registro.
- Current account (Cuenta actual): solo se seleccionan las regiones que especifica en la cuenta en la que ha iniciado sesión actualmente. Seleccione una de las siguientes opciones:
  - Current Region (Región actual): solo se dirige a los nodos administrados de la región seleccionada en la consola.
  - Elegir regiones: elija las regiones individuales a las que se aplicará la configuración de registro.

7. Seleccione Crear.

Implemente el paquete de conformidad de AWS Config mediante Quick Setup

Un paquete de conformidad es una colección de normas y medidas correctivas de AWS Config. Con Quick Setup, puede implementar un paquete de conformidad como una única entidad en una cuenta y una Región de AWS o en toda una organización en AWS Organizations. Esto le ayuda a administrar la conformidad de la configuración de los recursos de AWS a escala, desde la definición de políticas hasta la auditoría y la presentación de informes agregados, mediante un marco común y un modelo de empaquetado.

Para implementar paquetes de conformidad, realice las siguientes tareas en la consola de AWS Systems Manager Quick Setup.

> **Note**
>
> Debe habilitar el registro de AWS Config antes de implementar esta configuración. Para obtener más información, consulte Paquetes de conformidad en la Guía para desarrolladores de AWS Config.

Para implementar paquetes de conformidad con Quick Setup

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Quick Setup.
3. En la tarjeta Paquetes de conformidad, seleccione Crear.

   > **Tip**
   >
   > Si ya tiene una o más configuraciones en su cuenta, seleccione primero la pestaña Biblioteca o el botón Crear de la sección Configuraciones para ver las tarjetas.

4. En la sección Opciones de configuración, elija los paquetes de conformidad que desea implementar.
5. En la sección Schedule (Programación), elija la frecuencia con la que desea que Quick Setup corrija los cambios realizados en los recursos que difieren de la configuración. La opción Default (Predeterminado) se ejecuta una vez. Si no desea que Quick Setup corrija los cambios realizados en los recursos que difieren de la configuración, elija Disabled (Desactivado) en Custom (Personalizar).

6. En la sección Targets (Destinos), elija si desea implementar paquetes de conformidad en toda la organización, algunas Regiones de AWS, o la cuenta en la que ha iniciado sesión.

   Si elige Entire Organization (Toda la organización), continúe en el paso 8.

   Si elige Custom (Personalizado), continúe en el paso 7.
7. En la sección Target Regions (Regiones de destino), seleccione las casillas de verificación de las Regiones en las que desea implementar paquetes de conformidad.
8. Seleccione Crear.

Cómo configurar las revisiones para las instancias de una organización mediante una política de parches Quick Setup

Con Quick Setup, una regiones de AWS Systems Manager, puede crear políticas de revisiones con tecnología de Patch Manager. Una política de revisiones define la programación y la línea base que se utilizarán al aplicar revisiones automáticamente a las instancias de Amazon Elastic Compute Cloud (Amazon EC2) y a otros nodos administrados. Con una configuración de política de revisiones única, puede definir la aplicación de revisiones para todas las cuentas de varias Regiones de AWS de su organización, solo para las cuentas y regiones que elija o para un solo par de cuenta y región. Para obtener más información sobre las políticas de revisiones, consulte Configuraciones de políticas de revisiones en Quick Setup.

Requisito previo

Para definir una política de revisiones para un nodo que utilice Quick Setup, el nodo debe ser un nodo administrado. Para obtener más información sobre la administración de nodos, consulte Configuración de la consola unificada de Systems Manager para una organización.

> **Important**
>
> Métodos de escaneo de conformidad de las revisiones: Systems Manager admite varios métodos para analizar nodos administrados y así comprobar la conformidad de las revisiones. Si implementa más de uno de estos métodos a la vez, la información de conformidad de las revisiones que ve siempre será el resultado del análisis más reciente. Los resultados de análisis anteriores se sobrescriben. Si los métodos de análisis utilizan diferentes líneas de base de revisiones con diferentes reglas de aprobación, la información de conformidad de revisiones puede cambiar inesperadamente. Para obtener más

información, consulte Evitar sobrescrituras involuntarias de datos de conformidad de revisiones.

Estado de conformidad de la asociación y políticas de revisiones: el estado de las revisiones para un nodo administrado bajo una política de revisiones de Quick Setup coincide con el estado de la ejecución de la asociación de State Manager de ese nodo. Si el estado de ejecución de la asociación es `Compliant`, el estado de las revisiones del nodo administrado también se marca como `Compliant`. Si el estado de ejecución de la asociación es `Non-Compliant`, el estado de las revisiones del nodo administrado también se marca como `Non-Compliant`.

Regiones compatibles para configuraciones de políticas de revisiones

Las configuraciones de políticas de revisiones en Quick Setup se admiten actualmente en las siguientes regiones:

- Este de EE. UU. (Ohio) (us-east-2)
- Este de EE. UU. (Norte de Virginia) (us-east-1)
- EE. UU. Oeste (Norte de California) (us-west-1)
- Oeste de EE. UU. (Oregón) (us-west-2)
- Asia Pacífico (Bombay) (ap-south-1)
- Asia-Pacífico (Seúl) (ap-northeast-2)
- Asia-Pacífico (Singapur) (ap-southeast-1)
- Asia-Pacífico (Sídney) (ap-southeast-2)
- Asia-Pacífico (Tokio) (ap-northeast-1)
- Canadá (centro) (ca-central-1)
- Europa (Fráncfort) (eu-central-1)
- Europa (Irlanda) (eu-west-1)
- Europa (Londres) (eu-west-2)
- UE (París) (eu-west-3)
- Europa (Estocolmo) (eu-north-1)
- América del Sur (São Paulo) (sa-east-1)

Permisos para el bucket de S3 de la política de revisiones

Cuando crea una política de revisiones, Quick Setup crea un bucket de Amazon S3 que contiene un archivo denominado `baseline_overrides.json`. Este archivo almacena información sobre las líneas de base de revisiones que especificó para la política de revisiones.

El nombre del bucket de S3 tiene el siguiente formato `aws-quicksetup-patchpolicy-`*`account-id`*`-`*`quick-setup-configuration-id`*.

Por ejemplo: `aws-quicksetup-patchpolicy-123456789012-abcde`

Si va a crear una política de revisiones para una organización, el bucket se crea en la cuenta de administración de la organización.

Hay dos casos de uso en los que debe proporcionar permiso a otros recursos de AWS para acceder a este bucket de S3 mediante políticas AWS Identity and Access Management (IAM):

- Caso 1: uso de su propio perfil de instancia o rol de servicio con los nodos administrados en lugar de utilizar uno proporcionado por Quick Setup
- Caso 2: uso de puntos de conexión de VPC para conectarse a Systems Manager

La política de permisos que necesita en ambos casos se encuentra en la siguiente sección: Política de permisos para buckets de S3 de Quick Setup

Caso 1: uso de su propio perfil de instancia o rol de servicio con los nodos administrados en lugar de utilizar uno proporcionado por Quick Setup

Las configuraciones de políticas de revisiones incluyen una opción para Agregar las políticas de IAM necesarias a los perfiles de instancia existentes adjuntos a sus instancias.

Si no elige esta opción pero quiere que Quick Setup aplique revisiones a los nodos administrados mediante esta política de revisiones, debe asegurarse de implementar lo siguiente:

- La política gestionada de IAM `AmazonSSMManagedInstanceCore` debe adjuntarse al perfil de instancia de IAM o al rol de servicio de IAM que se utiliza para proporcionar permisos de Systems Manager a los nodos administrados.
- Debe agregar permisos para acceder a su bucket de políticas de revisiones como política integrada al perfil de instancia de IAM o rol de servicio de IAM. Puede proporcionar un acceso comodín a todos los buckets de `aws-quicksetup-patchpolicy` o solo al bucket específico creado para su organización o cuenta, como se muestra en los ejemplos de código anteriores.

- Debe etiquetar el perfil de instancia de IAM o rol de servicio de IAM con el siguiente par clave-valor.

  `Key: QSConfigId-`*`quick-setup-configuration-id`*`, Value: `*`quick-setup-configuration-id`*

  `quick-setup-configuration-id` representa el valor del parámetro aplicado a la pila AWS CloudFormation que se utiliza para crear la configuración de la política de revisiones. Para recuperar este identificador, haga lo siguiente:

  1. Abra la consola de AWS CloudFormation en https://console.aws.amazon.com/cloudformation.
  2. Seleccione el nombre de la pila que se utiliza para crear la política de revisiones. El nombre tiene un formato como `StackSet-AWS-QuickSetup-PatchPolicy-LA-q4bkg-52cd2f06-d0f9-499e-9818-d887cEXAMPLE`.
  3. Elija la pestaña Parámetros.
  4. En la lista de Parámetros, en la columna Clave, busque la clave QSConfigurationID. En la columna Valor de su fila, busque el ID de configuración, por ejemplo `abcde`.

     En este ejemplo, para que la etiqueta se aplique al perfil de instancia o al rol de servicio, la clave es `QSConfigId-abcde` y el valor es `abcde`.

Para obtener información sobre cómo agregar etiquetas a un rol de IAM, consulte Etiquetar los roles de IAM y Administrar las etiquetas en los perfiles de instancia (AWS CLIo AWS API) en la Guía del usuario de IAM.

Caso 2: uso de puntos de conexión de VPC para conectarse a Systems Manager

Si utiliza puntos de conexión de VPC para conectarse a Systems Manager, su política de puntos de conexión de VPC para S3 debe permitir el acceso al bucket de S3 de su política de revisiones de Quick Setup.

Para obtener información sobre cómo agregar permisos a una política de puntos de conexión de VPC para S3, consulte Control del acceso desde puntos de conexión de VPC con políticas de bucket en la Guía del usuario de Amazon S3.

Política de permisos para buckets de S3 de Quick Setup

Puede proporcionar acceso comodín a todos los buckets de `aws-quicksetup-patchpolicy` o solo al bucket específico creado para su organización o cuenta. Para proporcionar los permisos necesarios en los dos casos que se describen a continuación, utilice uno de los dos formatos.

All patch policy buckets

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AccessToAllPatchPolicyRelatedBuckets",
      "Effect": "Allow",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::aws-quicksetup-patchpolicy-*"
    }
  ]
}
```

Specific patch policy bucket

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AccessToMyPatchPolicyRelatedBucket",
      "Effect": "Allow",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::aws-quicksetup-patchpolicy-111122223333-quick-setup-configuration-id"
    }
  ]
}
```

[1] Una vez creada la configuración de la política de revisiones, puede buscar el nombre completo del bucket en la consola S3. Por ejemplo: `aws-quicksetup-patchpolicy-123456789012-abcde`

Cómo asignar al azar de identificadores de línea de base de revisiones en las operaciones de la política de revisiones

Las operaciones de aplicación de revisiones para las políticas de revisiones utilizan el parámetro `BaselineOverride` del documento de comandos SSM `AWS-RunPatchBaseline`.

Si se utiliza `AWS-RunPatchBaseline` para aplicar revisiones fuera de una política de revisiones, se puede utilizar `BaselineOverride` para especificar una lista de líneas de base de revisiones que se utilizarán durante la operación y que sean distintas de las predeterminadas especificadas. Esta lista se crea en un archivo denominado `baseline_overrides.json` y se agrega manualmente a un bucket de Amazon S3 de su propiedad, tal y como se explica en Uso del parámetro BaselineOverride.

Sin embargo, para las operaciones de aplicación de revisiones basadas en políticas de revisiones, Systems Manager crea automáticamente un bucket de S3 y le agrega un archivo `baseline_overrides.json`. A continuación, cada vez que Quick Setup ejecuta una operación de aplicación de revisiones (mediante la herramienta Run Command), el sistema genera un ID aleatorio para cada línea de base de revisiones. Este identificador es diferente para cada operación de aplicación de revisiones a la política de revisiones, y la línea de base de revisiones que representa no se almacena en su cuenta ni tiene acceso a ella.

Como resultado, no verá el ID de la línea de base de revisiones seleccionada en su configuración en los registros de aplicación de revisiones. Esto se aplica tanto a las líneas de base de revisiones de AWS administradas como a las líneas de base de revisiones personalizadas que haya seleccionado. El identificador de línea de base indicado en el registro es, en cambio, el que se generó para esa operación de aplicación de revisiones específica.

Además, si intenta ver los detalles en Patch Manager sobre la línea de base de revisiones que se generó con un identificador asignado al azar, el sistema indicará que la línea de base de revisiones no existe. Este es el comportamiento esperado y se puede omitir.

Cómo crear una política de revisiones

Para crear una política de revisiones, realice las siguientes tareas en la consola de Systems Manager.

Cómo crear una política de revisiones con Quick Setup

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.

Si va a configurar revisiones para una organización, asegúrese de haber iniciado sesión en la cuenta de administración de la organización. No puede configurar la política con la cuenta de administrador delegado ni con una cuenta de miembro.

2. En el panel de navegación, elija Quick Setup.
3. En la tarjeta de Patch Manager, elija Create (Crear).

> **Tip**
>
> Si ya tiene una o más configuraciones en su cuenta, seleccione primero la pestaña Biblioteca o el botón Crear de la sección Configuraciones para ver las tarjetas.

4. En Configuration name (Nombre de configuración), ingrese un nombre que ayude a identificar la política de revisiones.
5. En la sección Scanning and installation (Análisis e instalación), en Patch operation (Operación de revisiones), elija si la política de revisiones analizará los destinos especificados o analizará e instalará las revisiones en los destinos especificados.
6. En Scanning schedule (Programación de análisis), elija Use recommended defaults (Usar los valores predeterminados recomendados) o Custom scan schedule (Programación de análisis personalizado). La programación de análisis predeterminado analizará sus destinos todos los días a las 1:00 h UTC.

   - Si elige Custom scan schedule (Programación de análisis personalizado), seleccione la frecuencia de análisis (Scanning Frequency).
   - Si elige Daily (Diariamente), ingrese la hora, en UTC, en la que desea analizar sus destinos.
   - Si elige Custom CRON Expression (Expresión CRON personalizada), introduzca la programación como expresión CRON. Para obtener más información acerca de cómo dar formato a las expresiones CRON para Systems Manager, consulte Referencia: expresiones cron y rate para Systems Manager.

     Además, seleccione Wait to scan targets until first CRON interval (Esperar para analizar los destinos hasta el primer intervalo CRON). De forma predeterminada, Patch Manager analiza inmediatamente los nodos a medida que se convierten en destinos.
7. Si elige Scan and install (Analizar e instalar), elija la programación de instalación que se utilizará al instalar las revisiones en los destinos especificados. Si elige Use recommended defaults (Usar los valores predeterminados recomendados), Patch Manager instalará revisiones semanalmente a las 2:00 h UTC del domingo.

- Si elige Custom install schedule (Programación de instalación personalizada), seleccione la frecuencia de instalación (Installation Frequency).
- Si elige Daily (Diariamente), ingrese la hora, en UTC, en la que desea instalar las actualizaciones en sus destinos.
- Si elige Custom CRON expression (Expresión CRON personalizada), introduzca la programación como expresión CRON. Para obtener más información acerca de cómo dar formato a las expresiones CRON para Systems Manager, consulte Referencia: expresiones cron y rate para Systems Manager.

  Además, desactive Wait to install updates until first CRON interval (Esperar a instalar las actualizaciones hasta el primer intervalo CRON) para instalar inmediatamente las actualizaciones en los nodos a medida que se convierten en destinos. De forma predeterminada, Patch Manager espera hasta el primer intervalo CRON para instalar las actualizaciones.
- Elija Reboot if needed (Reiniciar si es necesario) para reiniciar los nodos después de la instalación de la revisión. Se recomienda reiniciar después de la instalación, pero puede causar problemas de disponibilidad.

8. En la sección Patch baseline (Línea de base de revisiones), elija las líneas de base de revisiones que se utilizarán al analizar y actualizar sus destinos.

   De forma predeterminada, Patch Manager utiliza las líneas de base de revisiones predefinidas. Para obtener más información, consulte Líneas de base predefinidas.

   Si elige Custom patch baseline (línea de base de revisiones personalizada), cambie la línea de base de revisiones seleccionada para los sistemas operativos que no desee utilizar una línea de base de revisiones de AWS predefinida.

   > **Note**
   >
   > Si usa puntos de conexión de VPC para conectarse a Systems Manager, asegúrese de que su política de puntos de conexión de VPC para S3 permita el acceso a este bucket de S3. Para obtener más información, consulte Permisos para el bucket de S3 de la política de revisiones.

> **Important**
>
> Si utiliza una configuración de política de revisiones en Quick Setup, las actualizaciones que realice en las líneas de base de revisiones personalizadas se sincronizan con Quick Setup cada hora.
> Si se elimina una línea de base de revisiones personalizada a la que se hacía referencia en una política de revisiones, aparece un banner en la página Configuration details (Detalles de configuración) de Quick Setup correspondiente a la política de revisiones. El banner le informa que la política de revisiones hace referencia a una línea de base de revisiones que ya no existe y que las operaciones de aplicación de revisiones posteriores fallarán. En este caso, vuelva a la página Configurations (Configuraciones) de Quick Setup, seleccione la configuración de Patch Manager y elija Actions (Acciones), Edit configuration (Editar configuración). El nombre de la línea de base de revisiones eliminado aparece resaltado y debe seleccionar una nueva línea de base de revisiones para el sistema operativo afectado.

9. (Opcional) En la sección Patching log storage (Almacenamiento de registros de revisiones), seleccione Write output to S3 bucket (Escribir salida en un bucket de S3) para almacenar los registros de operaciones de revisiones en un bucket de Amazon S3.

> **Note**
>
> Si está configurando una política de revisiones para una organización, la cuenta de administración de su organización debe tener al menos permisos de solo lectura para este bucket. Todas las unidades organizativas incluidas en la política deben tener acceso de escritura al bucket. Para obtener información sobre cómo conceder acceso a un bucket a diferentes cuentas, consulte el Ejemplo 2: el propietario del bucket concede permisos de bucket para varias cuentas en la Guía del usuario de Amazon Simple Storage Service.

10. Elija Examinar S3 para seleccionar el bucket en el que desea almacenar la salida del registro de revisiones. La cuenta de administración debe tener acceso de lectura a este bucket. Todos los destinos y las cuentas que no sean de administración configurados en la sección Targets (Destinos) deben tener acceso de escritura al bucket de S3 proporcionado para el registro.
11. En la sección Targets (Destinos), elija una de las siguientes opciones para identificar las cuentas y las regiones de esta operación de política de revisiones.

> **Note**
>
> Si trabaja en una sola cuenta, las opciones para trabajar con organizaciones y unidades organizativas (OU) no están disponibles. Puede elegir si desea aplicar esta configuración a todas las Regiones de AWS de su cuenta o solo a las regiones que seleccione.
> Si anteriormente especificó una región local para su cuenta y no se ha incorporado a la nueva experiencia de la consola Quick Setup, no puede excluir esa región de la configuración de Destinos.

- Entire organization (Toda la organización): todas las cuentas y regiones de su organización.
- Custom (Personalizado): solo las unidades organizativas y las regiones que especifique.
  - En la sección Target OUs (Unidades organizativas de destino), seleccione las unidades organizativas en las que desea configurar la política de revisiones.
  - En la sección Target Regions (Regiones de destino), seleccione las regiones en las que desea aplicar la política de revisiones.
- Current account (Cuenta actual): solo se seleccionan las regiones que especifica en la cuenta en la que ha iniciado sesión actualmente. Seleccione una de las siguientes opciones:
  - Current Region (Región actual): solo se dirige a los nodos administrados de la región seleccionada en la consola.
  - Choose Regions (Elegir regiones): elija las regiones individuales a las que se aplicará la política de revisiones.

12. En Choose how you want to target instances (Elegir cómo desea dirigir las instancias), elija una de las siguientes opciones para identificar los nodos en los que desea aplicar revisiones:

    - All managed nodes (Todos los nodos administrados): todos los nodos administrados de las unidades organizativas y regiones seleccionadas.
    - Specify the resource group (Especificar el grupo de recursos): elija el nombre de un grupo de recursos de la lista para dirigir sus recursos asociados.

> **Note**
>
> Actualmente, la selección de grupos de recursos solo se admite para configuraciones de cuentas individuales. Para aplicar revisiones a recursos de varias cuentas, elija una opción de segmentación diferente.

- Specify a node tag (Especificar una etiqueta de nodo): solo los nodos etiquetados con el par clave-valor que especifique tendrán revisiones aplicadas en todas las cuentas y regiones a las que se ha dirigido.
- Manual: elija manualmente los nodos administrados de todas las cuentas y regiones especificadas de una lista.

> **Note**
>
> Actualmente, esta opción solo admite instancias de Amazon EC2. Puede agregar un máximo de 25 instancias de forma manual en una configuración de política de revisiones.

13. En la sección Rate control (Control de frecuencia), haga lo siguiente:

    - En Concurrency (Simultaneidad), ingrese un número o un porcentaje de nodos en los cuales ejecutar la política de revisiones al mismo tiempo.
    - En Error threshold (Umbral de error), ingrese el número o el porcentaje de nodos que pueden experimentar un error antes de que falle la política de revisiones.

14. (Opcional) Seleccione la casilla Agregar políticas de IAM necesarias a los perfiles de instancia existentes asociados a sus instancias.

    Esta selección aplica las políticas de IAM creadas por esta configuración de Quick Setup a los nodos que ya tengan un perfil de instancia asociado (instancias de EC2) o un rol de servicio asociado (nodos activados de manera híbrida). Recomendamos que seleccione esta opción cuando los nodos administrados ya tengan un perfil de instancia o un rol de servicio asociado, pero no contengan todos los permisos necesarios para trabajar con Systems Manager.

    Su selección aquí se aplica a los nodos administrados que se creen más adelante en las cuentas y regiones a las que se aplica esta configuración de política de revisiones.

> **Important**
>
> Si no selecciona esta casilla pero quiere que Quick Setup aplique revisiones a los nodos administrados mediante esta política de revisiones, debe hacer lo siguiente:
> Agregue permisos a su perfil de instancia de IAM o rol de servicio de IAM para acceder al bucket de S3 creado para su política de revisiones
> Etiquete su perfil de instancia de IAM o rol de servicio de IAM con un par clave-valor específico.
> Para obtener más información, consulta Caso 1: uso de su propio perfil de instancia o rol de servicio con los nodos administrados en lugar de utilizar uno proporcionado por Quick Setup.

15. Seleccione Crear.

    Para revisar el estado de las revisiones una vez creada la política de revisiones, puede acceder a la configuración desde la página Quick Setup.

Configurar DevOps Guru con Quick Setup

Puede configurar rápidamente las opciones de DevOps Guru mediante Quick Setup. Amazon DevOps Guru es un servicio basado en Machine Learning (ML) que facilita la mejora del rendimiento operativo y la disponibilidad de la aplicación. DevOps Guru detecta comportamientos que se desvían de los patrones de funcionamiento normales para que pueda identificar problemas operativos mucho antes de que afecten a sus clientes. DevOps Guru captura automáticamente datos operativos de las aplicaciones de AWS y proporciona un único panel para visualizar problemas en los datos operativos. Puede comenzar con DevOps Guru para mejorar la disponibilidad y fiabilidad de las aplicaciones sin configuración manual ni experiencia en machine learning.

La configuración de DevOps Guru con Quick Setup está disponible en las siguientes Regiones de AWS:

- Este de EE. UU. (Norte de Virginia)
- Este de EE. UU. (Ohio)
- Oeste de EE. UU. (Oregón)
- Europa (Fráncfort)
- Europa (Irlanda)

- Europa (Estocolmo)
- Asia-Pacífico (Singapur)
- Asia-Pacífico (Sídney)
- Asia-Pacífico (Tokio)

Para obtener información acerca de los precios, consulte Precios de Amazon DevOps Guru.

Para configurar DevOps Guru, realice las siguientes tareas en la consola de AWS Systems Manager Quick Setup.

Para configurar DevOps Guru con Quick Setup

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Quick Setup.
3. En la tarjeta DevOps Guru, elija Crear.

   > **Tip**
   >
   > Si ya tiene una o más configuraciones en su cuenta, seleccione primero la pestaña Biblioteca o el botón Crear de la sección Configuraciones para ver las tarjetas.

4. En la sección Configuration options (Opciones de configuración), elija los tipos de recursos de AWS que desea analizar y las preferencias de notificación.

   Si no selecciona la opción Analyze all AWS resources in all the accounts in my organization (Analizar todos los recursos de AWS en todas las cuentas de la organización), puede elegir recursos de AWS para analizar más adelante en la consola de DevOps Guru. DevOps Guru analiza los distintos tipos de recursos de AWS (por ejemplo, buckets de Amazon Simple Storage Service [Amazon S3] e instancias de Amazon Elastic Compute Cloud [Amazon EC2]), que se clasifican en dos grupos de precios. Se paga por la cantidad de horas de recursos de AWS analizadas, por cada recurso activo. Un recurso solo está activo si produce métricas, eventos o entradas de registro en el plazo de una hora. La tarifa que se cobra por un tipo de recurso específico de AWS depende del grupo de precios.

   Si selecciona la opción Enable SNS notifications (Habilitar notificaciones de SNS), se crea un tema de Amazon Simple Notification Service (Amazon SNS) en cada Cuenta de AWS, en

las unidades organizativas de destino en su configuración. DevOps Guru utiliza el tema para enviarle notificaciones sobre eventos importantes de DevOps Guru, como la creación de nueva información. Si no habilita esta opción, puede agregar un tema más adelante en la consola de DevOps Guru.

Si selecciona la opción Enable AWS Systems Manager OpsItems (Habilitar elementos operativos de Systems Manager), se crearán elementos de trabajo operativos (OpsItems) para eventos relacionados de Amazon EventBridge y alarmas de Amazon CloudWatch.

5. En la sección Schedule (Programación), elija la frecuencia con la que desea que Quick Setup corrija los cambios realizados en los recursos que difieren de la configuración. La opción Default (Predeterminado) se ejecuta una vez. Si no desea que Quick Setup corrija los cambios realizados en los recursos que difieren de la configuración, elija Disabled (Desactivado) en Custom (Personalizar).
6. En la sección Targets (Destinos), elija si desea permitir que DevOps Guru analice los recursos de algunas de sus unidades organizativas o la cuenta en la que ha iniciado sesión.

   Si elige Custom (Personalizar), continúe al paso 8.

   Si elige Current account (Cuenta actual), continúe al paso 9.
7. En las secciones Target OUs (OU de destino) y Target Regions (Regiones de destino), seleccione las casillas de verificación de las OU y las Regiones en las que desea utilizar DevOps Guru.
8. Elija las Regiones en las que desea utilizar DevOps Guru en la cuenta actual.
9. Seleccione Crear.

Implemente paquetes de Distributor mediante Quick Setup

Distributor es una herramienta de AWS Systems Manager. El paquete Distributor es una colección de recursos de software instalable o activos que se pueden implementar como una sola entidad. Con Quick Setup, puede implementar un paquete Distributor en una Cuenta de AWS y una Región de AWS o en toda una organización en AWS Organizations. Actualmente, solo se puede implementar el agente de EC2Launch v2, el paquete de utilidades de Amazon Elastic File System (Amazon EFS) y el agente de Amazon CloudWatch con Quick Setup. Para obtener más información acerca de Distributor, consulte AWS Systems Manager Distributor.

Para implementar paquetes de Distributor realice las siguientes tareas en la consola de AWS Systems Manager Quick Setup.

Para implementar paquetes de Distributor con Quick Setup

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Quick Setup.
3. En la tarjeta de Distribuidor, seleccione Crear.

> **Tip**
>
> Si ya tiene una o más configuraciones en su cuenta, seleccione primero la pestaña Biblioteca o el botón Crear de la sección Configuraciones para ver las tarjetas.

4. En la sección Configuration options (Opciones de configuración), elija el paquete que desea implementar.
5. En la sección Targets (Destinos), elija si desea implementar el paquete para toda la organización, algunas de sus unidades organizativas o la cuenta en la que ha iniciado sesión.

   Si elige Entire Organization (Toda la organización), continúe en el paso 8.

   Si elige Custom (Personalizado), continúe en el paso 7.
6. En la sección Target OUs (Unidades organizativas de destino), seleccione las casillas de verificación de las unidades organizativas y las regiones en las que desea implementar el paquete.
7. Seleccione Crear.

Detenga e inicie las instancias EC2 automáticamente según una programación mediante Quick Setup

Con Quick Setup, una herramienta de AWS Systems Manager, puede configurar Programador de recursos para automatizar el inicio y la detención de las instancias de Amazon Elastic Compute Cloud (Amazon EC2).

Esta configuración de Quick Setup lo ayuda a reducir los costos operativos al iniciar y detener las instancias de acuerdo con la programación que especifique. Esta herramienta lo ayuda a evitar incurrir en costos innecesarios al ejecutar instancias cuando no son necesarias.

Por ejemplo, actualmente, puede que sus instancias estén en constante ejecución, aunque solo se usen 10 horas al día, 5 días a la semana. En su lugar, puede programar sus instancias para que se

detengan todos los días después del horario laboral. Como resultado, se ahorraría un 70 por ciento en esas instancias, ya que el tiempo de ejecución se reduce de 168 horas a 50 horas. El uso de Quick Setup no supone costo alguno. Sin embargo, se puede incurrir en costos en función de los recursos que configure y de los límites de uso, sin cargos por los servicios utilizados para configurar el servicio.

Con Programador de recursos, puede optar por detener e iniciar automáticamente las instancias en varias Regiones de AWS y Cuentas de AWS según una programación que usted defina. La configuración de Quick Setup se dirige a las instancias de Amazon EC2 mediante la clave de etiqueta y el valor que usted especifique. Programador de recursos solo detiene o inicia las instancias con una etiqueta que coincida con el valor que especifique en la configuración. Tenga en cuenta que si los volúmenes de Amazon EBS adjuntos a la instancia están cifrados, debe añadir los permisos necesarios para la clave AWS KMS al rol de IAM para que el Programador de recursos pueda iniciar la instancia.

Número máximo de instancias por configuración

Una configuración individual admite la programación de hasta 5000 instancias por región. Si su caso requiere programar más de 5000 instancias en una región determinada, debe crear varias configuraciones. Etiquete sus instancias en consecuencia para que cada configuración administre hasta 5000 instancias. Al crear varias configuraciones de Quick Setup de Programador de recursos, debe especificar diferentes valores de clave de etiqueta. Por ejemplo, una configuración puede usar la clave de etiqueta `Environment` con el valor `Production`, mientras que otra usa `Environment` y `Development`.

Comportamientos de programación

En los siguientes puntos se describen ciertos comportamientos de las configuraciones de programación:

- Programador de recursos inicia las instancias etiquetadas solo si están en estado `Stopped`. Del mismo modo, las instancias solo se detienen si están en estado `running`. Programador de recursos funciona con un modelo basado en eventos y solo inicia o detiene las instancias en los momentos que usted especifique. Por ejemplo, usted crea una programación que inicie las instancias a las 9:00 h. Programador de recursos inicia todas las instancias asociadas a la etiqueta que especifique que estén en estado `Stopped` a 9:00 h. Si las instancias se detienen manualmente más adelante, Programador de recursos no las volverá a iniciar para mantener el estado `Running`. Del mismo modo, si una instancia se inicia manualmente después de detenerla según lo programado, Programador de recursos no volverá a detenerla.

- Si crea una programación con una hora de inicio posterior a la hora de detención en una programación de 24 horas, Programador de recursos asume que las instancias se ejecutan durante la noche. Por ejemplo, usted crea una programación que inicie las instancias a las 21:00 h y las detenga a las 7:00 h. Programador de recursos inicia todas las instancias asociadas a la etiqueta que especifique que estén en estado `Stopped` a las 21:00 h y las detiene a las 7:00 h del día siguiente. En el caso de los horarios nocturnos, la hora de inicio aplica a los días que seleccione para su programación. Sin embargo, la hora de detención se aplica al día siguiente de su programación.
- Al crear una configuración de programación, es posible que el estado actual de las instancias cambie para que se ajuste a los requisitos de programación.

  Por ejemplo, supongamos que hoy es miércoles y especifica una programación para que las instancias administradas se inicien a las 9 h y se detengan a las 17 h solo los martes y los jueves. Como la hora actual está fuera del horario establecido para que las instancias se ejecuten, se detendrán una vez creada la configuración. Las instancias no volverán a ejecutarse hasta la siguiente hora establecida, es decir, el jueves a las 9 h.

  Si el estado actual de las instancias es `Stopped` y especifica una programación en la que se ejecutarían en la hora actual, Programador de recursos las inician después de que se cree la configuración.

Si elimina la configuración, las instancias ya no se detendrán ni se iniciarán de acuerdo con la programación definida anteriormente. En raras ocasiones, es posible que las instancias no se detengan o se inicien correctamente debido a errores en las operaciones de la API.

Para configurar la programación de las instancias de Amazon EC2, realice las siguientes tareas en la consola AWS Systems Manager Quick Setup.

Para configurar la programación de instancias con Quick Setup

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Quick Setup.
3. En la tarjeta del Programador de recursos, seleccione Crear.

> **Tip**
>
> Si ya tiene una o más configuraciones en su cuenta, seleccione primero la pestaña Biblioteca o el botón Crear de la sección Configuraciones para ver las tarjetas.

4. En la sección Instance tag (Etiqueta de instancia), especifique la clave de etiqueta y el valor aplicados a las instancias que desea asociar a su programación.
5. En la sección Schedule options (Opciones de programación), especifique la zona horaria, los días y las horas en que desea iniciar y detener las instancias.
6. En la sección Targets (Destinos), elija si desea configurar la programación para un grupo de unidades organizativas (OU) personalizadas (Custom) o la cuenta actual (Current account) en la que tenga iniciada la sesión:
   - Custom (Personalizadas): en la sección Target OUs (OU de destino), seleccione las OU en donde desea configurar la programación. Luego, en la sección Target Regions (Regiones de destino), seleccione las regiones en las que desea configurar la programación.
   - Cuenta actual: Seleccione Región actual o Elegir regiones. Si seleccionó Choose Regions (Elegir regiones), elija las Target Regions (Regiones de destino) en las que desea configurar la programación.
7. Verifique la información de la programación en la sección Summary (Resumen).
8. Seleccione Crear.

Configuración de Explorador de recursos de AWS mediante Quick Setup

Con Quick Setup, una herramienta de AWS Systems Manager, puede configurar Explorador de recursos de AWS rápidamente para buscar y descubrir recursos en su Cuenta de AWS o en toda una organización de AWS. Puede buscar sus recursos mediante metadatos, como nombres, etiquetas e ID. Explorador de recursos de AWS brinda respuestas rápidas a sus consultas de búsqueda mediante el uso de índices. Resource Explorer crea y mantiene índices con una variedad de orígenes de datos para recopilar información sobre los recursos de su Cuenta de AWS.

Quick Setup para Resource Explorer automatiza el proceso de configuración del índice. Para obtener más información sobre Explorador de recursos de AWS, consulte ¿Qué es Explorador de recursos de AWS? en la Guía del usuario de Explorador de recursos de AWS.

Con Quick Setup, Resource Explorer realiza lo siguiente:

- Crea un índice en cada una de la Región de AWS de su Cuenta de AWS.
- Actualiza el índice de la región que especifique como índice agregador de la cuenta.
- Crea una vista predeterminada en la región del índice de agregador. Esta vista no posee filtros, por lo que devuelve todos los recursos que se encuentran en el índice.

Permisos mínimos

Para realizar los siguientes pasos, debe tener los siguientes permisos:

- Acción: `resource-explorer-2:*` — Recurso: ningún recurso específico (`*`)
- Acción: `iam:CreateServiceLinkedRole` — Recurso: ningún recurso específico (`*`)

Para configurar Resource Explorer

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Quick Setup.
3. En la tarjeta de Resource Explorer, seleccione Crear.
4. En la sección Región del índice de agregadores, elija qué región desea que contenga el índice de agregadores. Debe seleccionar la región adecuada para la ubicación geográfica de los usuarios.
5. (Opcional) Seleccione la casilla de verificación Reemplazar los índices agregadores existentes en regiones distintas de la seleccionada anteriormente.
6. En la sección Destinos, elija la organización de destino o las unidades organizativas (UO) específicas que contengan los recursos que desea descubrir.
7. En la sección Regiones, elija qué regiones quiere incluir en la configuración.
8. Revise el resumen de la configuración y, a continuación, elija Crear.

En la página de Resource Explorer, puede supervisar el estado de la configuración.

Solución de problemas de los resultados de Quick Setup

Utilice la siguiente información como ayuda para solucionar problemas con Quick Setup, una herramienta de AWS Systems Manager. Este tema incluye tareas específicas para resolver problemas en función del tipo de incidencia de Quick Setup.

Problema: error de implementación

Se produce un error en la implementación si el conjunto de pilas de CloudFormation ha fallado durante la creación. Siga los siguientes pasos para investigar un error de implementación.

1. Vaya a la consola de AWS CloudFormation.
2. Elija la pila creada por su configuración de Quick Setup. El Stack name (Nombre de la pila) incluye `QuickSetup` seguida del tipo de configuración que eligió, como `SSMHostMgmt`.

   > **Note**
   >
   > CloudFormation, a veces, elimina implementaciones de pila con error. Si la pila no está disponible en la tabla Stacks (Pilas), elija Deleted (Eliminadas) de la lista de filtros.

3. Consulte Status (Estado) y Status reason (Motivo del estado). Para obtener más información sobre los estados de la pila, consulte Códigos de estado de pilas en la Guía del usuario de AWS CloudFormation.
4. Para entender el paso exacto que ha fallado, consulte la pestaña Events (Eventos) y revise el Status (Estado) de cada evento.
5. Revise la Solución de problemas en la Guía del usuario de AWS CloudFormation.
6. Si no puede resolver el error de implementación mediante los pasos de la solución de problemas de CloudFormation, elimine la configuración y vuelva a configurarla.

Problema: error de asociación

La tabla Configuration details (Detalles de configuración) de la página Configuration details (Detalles de configuración) de la configuración muestra Configuration status (Estado de configuración) con el valor Failed (Error) si alguna de las asociaciones ha fallado durante la configuración. Siga los pasos a continuación para solucionar un error de asociación.

1. En la tabla Configuration details (Detalles de configuración), elija la configuración con error y, a continuación, elija View details (Ver detalles).
2. Copie el Association name (Nombre de la asociación).
3. Vaya a State Manager y pegue el nombre de la asociación en el campo de búsqueda.
4. Elija la asociación y elija la pestaña Execution history (Historial de ejecución).

5. En Execution ID (ID de ejecución), elija la ejecución de la asociación que ha producido un error.
6. La página Association execution targets (Destinos de ejecución de asociación) muestra todos los nodos en los que se ejecutó la asociación. Elija el botón Output (Salida) para una ejecución que no se pudo ejecutar.
7. En la página Output (Salida) elija Step - Output (Paso: salida) para ver el mensaje de error de ese paso en la ejecución del comando. Cada paso puede mostrar un mensaje de error diferente. Revise los mensajes de error de todos los pasos para ayudar a solucionar el problema.

Si ver el resultado del paso no ayuda a solucionar el problema, puede volver a crear la asociación. Para volver a crear la asociación, elimine primero la asociación que falla en State Manager. Después de eliminar la asociación, edite la configuración, elija la opción que eliminó y elija Update (Actualizar).

> **Note**
>
> Para investigar las asociaciones que figuren con el estado Failed (Error) para una configuración de Organization (Organización), debe iniciar sesión en la cuenta con la asociación con error y seguir el procedimiento de la asociación errónea que antes se indicó. El ID de asociación no es un hipervínculo a la cuenta de destino cuando se visualizan los resultados de la cuenta de administración.

Problema: estado de deriva

Al ver la página de detalles de una configuración, puede ver el estado de la desviación de cada implementación. La desviación de la configuración se produce cada vez que un usuario realiza algún cambio en un servicio o característica que entra en conflicto con las selecciones realizadas a través de Quick Setup. Si una asociación ha cambiado después de la configuración inicial, la tabla muestra un icono de advertencia que indica la cantidad de elementos que se han desviado. Puede determinar qué causó la desviación al pasar el ratón sobre el icono.

Cuando se elimina una asociación en State Manager, las implementaciones relacionadas muestran una advertencia de desviación. Para solucionarlo, edite la configuración y elija la opción que se quitó cuando se eliminó la asociación. Elija Update (Actualizar) y espere a que se complete la implementación.

Herramientas de aplicaciones de AWS Systems Manager

Herramientas de aplicaciones es un conjunto de capacidades que lo ayudan a administrar las aplicaciones que se ejecutan en AWS.

Temas

- AWS AppConfig
- AWS Systems Manager Application Manager
- AWS Systems Manager Parameter Store

AWS AppConfig

La información sobre AWS AppConfig se ha trasladado a guías independientes. Para obtener más información, consulte los siguientes temas:

- Guía del usuario de AWS AppConfig
- Referencia de la API de AWS AppConfig

AWS Systems Manager Application Manager

Application Manager, una herramienta de AWS Systems Manager, ayuda a los ingenieros de DevOps a investigar y solucionar problemas con sus recursos de AWS en el contexto de sus aplicaciones y clústeres. Application Manager agrega información de operaciones de múltiples Servicios de AWS y las herramientas de Systems Manager a una única AWS Management Console.

En Application Manager, una aplicación es un grupo lógico de recursos de AWS que usted desea que opere como una unidad. Este grupo lógico puede representar diferentes versiones de una aplicación, límites de propiedad para operadores o entornos de desarrollador, por nombrar algunos. La compatibilidad de Application Manager con clústeres de contenedores incluye clústeres de Amazon Elastic Kubernetes Service (Amazon EKS) y Amazon Elastic Container Service (Amazon ECS).

La primera vez que abra Application Manager, aparecerá la página What Application Manager can do for you. Cuando elige la opción Empezar, Application Manager importa automáticamente metadatos sobre sus recursos que se han creado en otras herramientas de Servicios de AWS o Systems Manager. Luego, Application Manager muestra esos recursos en una lista agrupada por categorías predefinidas.

En Applications (Aplicaciones), la lista incluye lo siguiente:

- Pilas de AWS CloudFormation
- Aplicaciones personalizadas
- aplicaciones de AWS Launch Wizard
- Aplicaciones AppRegistry
- Aplicaciones de carga de trabajo empresarial de AWS para SAP
- Clústeres de Amazon ECS
- Clústeres de Amazon EKS

Después de configurar los Servicios de AWS y las herramientas de Systems Manager, Application Manager muestra los siguientes tipos de información acerca de los recursos:

- Información sobre el estado actual y el estado de Amazon EC2 Auto Scaling de las instancias de Amazon Elastic Compute Cloud (Amazon EC2) en su aplicación
- Alarmas proporcionadas por Amazon CloudWatch
- Información de conformidad proporcionada por AWS Config y State Manager (un componente de Systems Manager)
- Información del clúster de Kubernetes proporcionada por Amazon EKS
- Datos de registro proporcionados por AWS CloudTrail y los Registros de Amazon CloudWatch.
- OpsItems proporcionado por Systems Manager OpsCenter
- Detalles de recursos proporcionados por los Servicios de AWS que los alojan.
- Información del contenedor de clústeres proporcionada por Amazon ECS.

Para ayudarlo a solucionar problemas con componentes o recursos, Application Manager también proporciona manuales de procedimientos que puede asociar con sus aplicaciones. Para comenzar a utilizar Application Manager, abra la consola de Systems Manager. En el panel de navegación, elija Application Manager.

¿Cuáles son los beneficios de utilizar Application Manager?

Application Manager reduce el tiempo que tardan los ingenieros de DevOps en detectar e investigar problemas con recursos de AWS. Para ello, Application Manager muestra muchos tipos de información de operaciones en el contexto de una aplicación en una consola. Application

Manager también reduce el tiempo que se tarda en solucionar problemas a través de manuales de procedimientos que realizan tareas comunes de corrección en recursos de AWS.

¿Cuáles son las características de Application Manager?

Application Manager incluye las siguientes características:

- Importe sus recursos de AWS automáticamente

  Durante la configuración inicial, puede elegir que Application Manager importe y muestre automáticamente recursos en su Cuenta de AWS que se basan en pilas de CloudFormation, AWS Resource Groups, implementaciones de Launch Wizard, aplicaciones AppRegistry y clústeres de Amazon ECS y Amazon EKS. El sistema muestra estos recursos en categorías predefinidas de aplicaciones o clústeres. Posteriormente, siempre que se agreguen nuevos recursos de estos tipos a su Cuenta de AWS, Application Manager muestra automáticamente los nuevos recursos en las categorías predefinidas de aplicaciones y clústeres.

- Crear o editar pilas y plantillas de CloudFormation

  Application Manager lo ayuda a aprovisionar y administrar recursos para sus aplicaciones mediante la integración con CloudFormation. Puede crear, editar y eliminar plantillas y pilas de AWS CloudFormation en Application Manager. Application Manager también incluye una biblioteca de plantillas donde puede clonar, crear y almacenar plantillas. Application Manager y CloudFormation muestran la misma información sobre el estado actual de una pila. Las plantillas y las actualizaciones de plantillas se almacenan en Systems Manager hasta que aprovisione la pila, momento en el que los cambios también se muestran en CloudFormation.

- Ver información sobre sus instancias en el contexto de una aplicación

  Application Manager se integra con Amazon Elastic Compute Cloud (Amazon EC2) para mostrar información sobre sus instancias en el contexto de una aplicación. Application Manager muestra el estado de la instancia y el estado de Amazon EC2 Auto Scaling para una aplicación seleccionada en un formato gráfico. La pestaña Instances (Instancias) también incluye una tabla con la siguiente información para cada instancia de la aplicación.

  - Estado de la instancia (pendiente, deteniéndose, en ejecución, detenida)
  - Estado de ping para SSM Agent
  - Estado y nombre del último manual de procedimientos de Automatización de Systems Manager procesado en la instancia
  - Un recuento de las alarmas de Registros de Amazon CloudWatch por estado.

    - `ALARM`: la métrica o expresión está fuera del umbral definido.
    - `OK`: la métrica o expresión está dentro del umbral definido.
    - `INSUFFICIENT_DATA`: la alarma acaba de iniciarse, la métrica no está disponible o no hay suficientes datos disponibles en la métrica para determinar el estado de la alarma.
  - Estado de los grupos de escalado automático principal e individual
- Ver métricas y alarmas operativas para una aplicación o clúster

  Application Manager se integra con Amazon CloudWatch para proporcionar métricas y alarmas operativas en tiempo real para una aplicación o clúster. Puede profundizar en el árbol de aplicaciones para ver las alarmas en cada nivel de componente o ver las alarmas de un clúster individual.
- Ver los datos de registro de una aplicación

  Application Manager se integra con los Registros de Amazon CloudWatch para proporcionar datos de registro en el contexto de su aplicación sin tener que salir de Systems Manager.
- Ver y administrar OpsItems para una aplicación o clúster

  Application Manager se integra con AWS Systems Manager OpsCenter para proporcionar una lista de elementos de trabajo operacionales (OpsItems) para sus aplicaciones y clústeres. La lista refleja OpsItems automáticamente generados y creados de forma manual. Puede ver detalles sobre el recurso que creó un OpsItem y el estado, la fuente y la severidad de OpsItem.
- Ver datos de conformidad de recursos para una aplicación o clúster

  Application Manager se integra con AWS Config para proporcionar detalles sobre el cumplimiento y el historial de sus recursos de AWS según las reglas que especifique. Application Manager también se integra con AWS Systems Manager State Manager para proporcionar información de conformidad sobre el estado que desea mantener para sus instancias Amazon Elastic Compute Cloud (Amazon EC2).
- Ver información sobre la infraestructura de clúster de Amazon ECS y Amazon EKS

  Application Manager se integra con Amazon ECS y Amazon EKS para proporcionar información sobre el estado de las infraestructuras de clúster y una vista en tiempo de ejecución de componentes de los recursos informáticos, de redes y de almacenamiento de un clúster.

  Sin embargo, no puede administrar ni ver la información de operaciones sobre sus pods o contenedores de Amazon EKS en Application Manager. Solo puede administrar y ver información de operaciones sobre la infraestructura que aloja sus recursos de Amazon EKS.

- Ver detalles de costos de recursos de una aplicación

  Application Manager está integrado con AWS Cost Explorer, una característica de Administración de facturación y costos de AWS, a través del widget Cost (Costo). Después de habilitar Explorador de costos en la consola de administración de facturación y costo, el widget Cost en Application Manager muestra los datos de costos de una aplicación o componente de aplicación específicos que no están en contenedores. Puede utilizar filtros en el widget para ver los datos de costos según diferentes periodos de tiempo, detalles y tipos de costos en un gráfico de barras o líneas.

- Ver información detallada de recursos en una sola consola

  Elija un nombre de recurso enumerado en Application Manager y vea información contextual e información de operaciones sobre ese recurso sin tener que salir de Systems Manager.

- Recibir actualizaciones automáticas de recursos para aplicaciones

  Si realiza cambios en un recurso en una consola de servicio y ese recurso forma parte de una aplicación en Application Manager, a continuación, Systems Manager muestra automáticamente esos cambios. Por ejemplo, si actualiza una pila en la consola de AWS CloudFormation, y si esa pila es parte de una aplicación de Application Manager, las actualizaciones de la pila se reflejan automáticamente en Application Manager.

- Descubrir aplicaciones de Launch Wizard automáticamente

  Application Manager está integrado en AWS Launch Wizard. Si utilizó Launch Wizard para implementar recursos para una aplicación, Application Manager puede importarlos y mostrarlos automáticamente en una sección de Launch Wizard.

- Monitoreo de recursos de aplicaciones en Application Manager mediante CloudWatch Application Insights

  Application Manager se integra con Amazon CloudWatch Application Insights. Application Insights identifica y configura las métricas clave, los registros y las alarmas entre los recursos de aplicaciones y la pila de tecnología. Application Insights monitorea continuamente las métricas y los registros para detectar y relacionar anomalías y errores. Cuando el sistema detecta errores y anomalías, Application Insights genera CloudWatch Events que puede utilizar para configurar notificaciones o tomar medidas. Puede habilitar y ver Application Insights en las pestañas Overview (Información general) y Monitoring (Monitoreo) en Application Manager. Para obtener más información acerca de Application Insights, consulte What is Amazon CloudWatch Application Insights (¿Qué es Amazon CloudWatch Application Insights?) en la Guía del usuario de Amazon CloudWatch.

- Solucionar problemas con manuales de procedimientos

  Application Manager incluye manuales de procedimientos predefinidos de Systems Manager para solucionar problemas comunes con recursos de AWS. Puede ejecutar un manual de procedimientos en todos los recursos correspondientes de una aplicación sin tener que salir de Application Manager.

¿Se cobra por usar Application Manager?

Application Manager está disponible sin costo adicional.

¿Cuáles son las cuotas de recursos de Application Manager?

Puede ver las cuotas para todas las herramientas de Systems Manager en Systems Manager service quotas en la Referencia general de Amazon Web Services. A menos que se indique otra cosa, cada cuota es específica de la región.

Temas

- Configuración de servicios relacionados
- Configuración de permisos para Application Manager de Systems Manager
- Agregar aplicaciones y clústeres de contenedor a Application Manager
- Trabajo con aplicaciones

Configuración de servicios relacionados

Application Manager, una herramienta de AWS Systems Manager, muestra los recursos y la información de otros Servicios de AWS y herramientas de Systems Manager. Para maximizar la cantidad de información de operaciones que se muestra en Application Manager, le recomendamos que configure estos otros servicios o herramientas antes de utilizar Application Manager.

Temas

- Configurar tareas de importación de recursos
- Configurar tareas para ver información de operaciones acerca de los recursos

Configurar tareas de importación de recursos

Las siguientes tareas de configuración lo ayudan a ver recursos de AWS en Application Manager. Después de completar cada una de estas tareas, Systems Manager puede importar recursos automáticamente a Application Manager. Una vez importados los recursos, puede crear aplicaciones en Application Manager y mover sus recursos importados a ellas. Esto lo ayuda a ver información de operaciones en el contexto de una aplicación.

(Opcional) Organice sus recursos de AWS con etiquetas

Puede asignar metadatos a los recursos de AWS en forma de etiquetas. Cada etiqueta es una marca que consta de una clave y un valor definidos por el usuario. Las etiquetas pueden ayudarle a administrar, identificar, organizar, buscar y filtrar recursos. Puede crear etiquetas para clasificar los recursos según su finalidad, propietario, entorno u otro criterio.

(Opcional) Organice sus recursos de AWS con AWS Resource Groups

Puede utilizar grupos de recursos para organizar los recursos de AWS. Los grupos de recursos facilitan la administración, el monitoreo y la automatización de tareas en grandes cantidades de recursos al mismo tiempo.

Application Manager importa automáticamente todos los grupos de recursos y los enumera en la categoría Custom applications (Aplicaciones personalizadas).

(Opcional) Configure e implemente sus recursos de AWS con AWS CloudFormation

AWS CloudFormation le permite crear y aprovisionar implementaciones de infraestructura de AWS de forma predecible y uniforme. Lo ayuda a utilizar Servicios de AWS como Amazon EC2, Amazon Elastic Block Store (Amazon EBS), Amazon Simple Notification Service (Amazon SNS), Elastic Load Balancing y AWS Auto Scaling. Con CloudFormation, puede crear aplicaciones fiables, escalables y rentables en la nube sin preocuparse por la creación y configuración de la infraestructura de AWS subyacente.

Application Manager importa automáticamente todos los recursos de AWS CloudFormation y los enumera en la categoría pilas de AWS CloudFormation. Puede crear pilar y plantillas de CloudFormation en Application Manager. Los cambios de pila y plantilla se sincronizan automáticamente entre Application Manager y CloudFormation. También puede crear aplicaciones en Application Manager y mover pilas a ellas. Esto lo ayuda a ver información de operaciones de sus recursos en las pilas en el contexto de una aplicación. Para obtener información acerca de los precios, consulte Precios de AWS CloudFormation.

(Opcional) Configure e implemente sus aplicaciones con AWS Launch Wizard

Launch Wizard lo guía a lo largo del proceso de ajuste de tamaño, configuración e implementación de recursos de AWS para aplicaciones de terceros sin necesidad de identificar y aprovisionar manualmente recursos de AWS.

Application Manager importa automáticamente todos los recursos de Launch Wizard y los enumera en la categoría de Launch Wizard. Para obtener más información acerca de AWS Launch Wizard, consulte Introducción a AWS Launch Wizard para SQL Server. Launch Wizard está disponible sin costo adicional. Solo pagará por los recursos de AWS que aprovisiona para ejecutar la solución.

(Opcional) Configure e implemente sus aplicaciones en contenedores mediante Amazon ECS y Amazon EKS

Amazon Elastic Container Service (Amazon ECS) es un servicio de administración de contenedores altamente escalable y rápido que facilita la tarea de ejecutar, detener y administrar contenedores en un clúster. Los contenedores se determinan en una definición de tarea que se utiliza para ejecutar tareas individuales o tareas dentro de un servicio.

Amazon EKS es un servicio administrado que lo ayuda a ejecutar Kubernetes en AWS sin necesidad de instalar, operar ni mantener su propio plano de control o nodos de Kubernetes. Kubernetes es un sistema de código abierto para automatizar la implementación, escalado y administración de las aplicaciones en contenedores.

Application Manager importa automáticamente todos sus recursos de infraestructura de Amazon ECS y Amazon EKS y los enumera en la pestaña Container clusters (Clústeres de contenedores). Sin embargo, no puede administrar ni ver la información de operaciones sobre sus pods o contenedores de Amazon EKS en Application Manager. Solo puede administrar y ver información de operaciones sobre la infraestructura que aloja sus recursos de Amazon EKS. Para obtener información acerca de los precios, consulte Precios de Amazon ECS y Precios de Amazon EKS.

Configurar tareas para ver información de operaciones acerca de los recursos

Las siguientes tareas de configuración lo ayudan a ver la información de operaciones acerca de sus recursos de AWS en Application Manager.

(Recomendado) Verificar permisos de manuales de procedimientos

Puede solucionar los problemas con los recursos de AWS desde Application Manager empleando los manuales de procedimientos de Automatización de Systems Manager. Para utilizar esta

herramienta de corrección, debe configurar o verificar permisos. Para obtener información acerca de los precios, consulte Precios de AWS Systems Manager.

(Opcional) Habilitar el Explorador de costos

El AWS Cost Explorer es una característica de AWS Cost Management que puede utilizar para visualizar sus datos de costos para analizarlos más a fondo. Cuando habilita el Explorador de costos, puede ver la información, el historial y la optimización de los costos de los recursos de su aplicación en la consola de Application Manager.

(Opcional) Configuración de alarmas y Registros de Amazon CloudWatch

CloudWatch es un servicio de supervisión y administración que proporciona datos e información procesable para AWS, aplicaciones híbridas y multinube y recursos de infraestructura. Con CloudWatch, puede recopilar y acceder a todos sus datos operativos y de rendimiento en forma de registros y métricas desde una única plataforma. Para ver los registros y alarmas de CloudWatch para sus recursos en Application Manager, debe configurar CloudWatch. Para obtener información acerca de los precios, consulte Precios de CloudWatch.

> **Note**
>
> El soporte de los Registros de CloudWatch se aplica únicamente a las aplicaciones, no a los clústeres.

(Opcional) Configurar AWS Config

AWS Config proporciona una vista detallada de los recursos asociados a su Cuenta de AWS, incluido cómo se configuran, cómo están relacionados entre sí y cómo las configuraciones y sus relaciones han cambiado a lo largo del tiempo. Puede utilizar AWS Config para evaluar la configuración de sus recursos de AWS. Para ello, cree reglas de AWS Config que representen los ajustes de configuración ideal. Aunque AWS Config realiza un seguimiento continuo de los cambios de configuración que se producen entre los recursos, también comprueba si estos cambios infringen cualquiera de las condiciones de las reglas. Si un recurso infringe una regla, AWS Config marca el recurso y la regla como no conforme. Application Manager muestra información de cumplimiento acerca de las reglas de AWS Config. Para ver estos datos en Application Manager, debe configurar AWS Config. Para obtener información acerca de los precios, consulte Precios de AWS Config.

(Opcional) Crear asociaciones de State Manager

Puede utilizar State Manager de Systems Manager para crear una configuración que asigne a los nodos administrados. La configuración, denominada una asociación, define el estado que desea mantener en los nodos. Para ver los datos de conformidad de asociaciones en Application Manager debe configurar una o varias asociaciones de State Manager. State Manager se ofrece sin cargo adicional.

(Opcional) Configurar OpsCenter

Puede ver elementos de trabajo operativos (OpsItems) acerca de sus recursos en Application Managermediante OpsCenter. Puede configurar Amazon CloudWatch y Amazon EventBridge para enviar automáticamente OpsItems a OpsCenter basado en alarmas y eventos. También puede ingresar OpsItems manualmente. Para obtener información acerca de los precios, consulte AWS Systems Manager Pricing (Precios de Glue).

Configuración de permisos para Application Manager de Systems Manager

Puede utilizar todas las características de Application Manager, una herramienta de AWS Systems Manager, si la entidad (como el usuario, el grupo o el rol) de AWS Identity and Access Management (IAM) tiene acceso a las operaciones de la API que se enumeran en este tema. Las operaciones de API están separadas en dos tablas para ayudarlo a comprender las diferentes funciones que realizan.

La siguiente tabla contiene una lista de las operaciones de la API que Systems Manager llama si elige un recurso en Application Manager porque desea ver los detalles del recurso. Por ejemplo, si Application Manager muestra un grupo de Amazon EC2 Auto Scaling y, si elige ese grupo para ver sus detalles, Systems Manager llama las operaciones `autoscaling:DescribeAutoScalingGroups` de la API. Si no tiene ningún grupo de escalado automático en su cuenta, esta operación de API no se llama desde Application Manager.

Detalles del recurso únicamente

```
acm:DescribeCertificate
acm:ListTagsForCertificate
autoscaling:DescribeAutoScalingGroups
cloudfront:GetDistribution
cloudfront:ListTagsForResource
cloudtrail:DescribeTrails
```

Detalles del recurso únicamente

```
cloudtrail:ListTags
cloudtrail:LookupEvents
codebuild:BatchGetProjects
codepipeline:GetPipeline
codepipeline:ListTagsForResource
dynamodb:DescribeTable
dynamodb:ListTagsOfResource
ec2:DescribeAddresses
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
```

Detalles del recurso únicamente

```
redshift:DescribeClusters
s3:GetBucketTagging
```

La siguiente tabla contiene una lista de las operaciones de la API que Systems Manager utiliza para realizar cambios en las aplicaciones y los recursos que se indican en Application Manager o para ver la información de operaciones de una aplicación o recurso seleccionados.

Acciones y detalles de la aplicación

```
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
```

Acciones y detalles de la aplicación

```
config:GetComplianceDetailsByResource
config:GetResourceConfigHistory
config:ListDiscoveredResources
config:PutRemediationConfigurations
config:SelectResourceConfig
config:StartConfigRulesEvaluation
config:StartRemediationExecution
ec2:DescribeInstances
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
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
```

Acciones y detalles de la aplicación

```
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:UpdateOpsItem
ssm:UpdateOpsMetadata
ssm:UpdateServiceSetting
tag:GetTagKeys
tag:GetTagValues
tag:TagResources
tag:UntagResources
```

Ejemplo de política para todos los permisos de Application Manager

Para configurar permisos de Application Manager para una entidad (un usuario, un grupo o un rol) de IAM, cree una política de IAM mediante el siguiente ejemplo. Este ejemplo de política incluye todas las operaciones de la API utilizadas por Application Manager.

JSON

```
                {
"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

Puede restringir la capacidad de un usuario para realizar cambios en aplicaciones y recursos en Application Manager. Esto lo puede hacer si elimina las siguientes operaciones de la API de la política de permisos de IAM asociada a su usuario, grupo o rol. La eliminación de estas acciones crea una experiencia de solo lectura en Application Manager. Las siguientes son todas las API que permiten a los usuarios realizar cambios en la aplicación o en cualquier otro recurso relacionado.

```
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
```

Para obtener información acerca de la creación y la edición de políticas de IAM, consulte Creación de políticas de IAM en la Guía del usuario de IAM. Para obtener información acerca de cómo asignar esta política a una entidad (un usuario, un grupo o un rol) de IAM, consulte Adición y eliminación de permisos de identidad de IAM.

Agregar aplicaciones y clústeres de contenedor a Application Manager

Application Manager es un componente de AWS Systems Manager. En Application Manager, una aplicación es un grupo lógico de recursos de AWS que usted desea que opere como una unidad. Este grupo lógico puede representar diferentes versiones de una aplicación, límites de propiedad para operadores o entornos de desarrollador, por nombrar algunos.

La primera vez que abra Application Manager, aparecerá la página What Application Manager can do for you. Cuando elige la opción Empezar, Application Manager importa automáticamente metadatos sobre sus recursos que se han creado en otras herramientas de Servicios de AWS o Systems Manager. Luego, Application Manager muestra esos recursos en una lista agrupada por categorías predefinidas.

En Applications (Aplicaciones), la lista incluye lo siguiente:

- Pilas de AWS CloudFormation
- Aplicaciones personalizadas
- aplicaciones de AWS Launch Wizard
- Aplicaciones AppRegistry
- Aplicaciones de carga de trabajo empresarial de AWS para SAP
- Clústeres de Amazon ECS
- Clústeres de Amazon EKS

Una vez que finaliza la importación, puede ver la información de operaciones de una aplicación o un recurso específico en estas categorías predefinidas. O bien, si desea proporcionar más contexto acerca de una colección de recursos, puede crear manualmente una aplicación en Application Manager. A continuación, puede agregar recursos o grupos de recursos a esa aplicación. Después de crear una aplicación en Application Manager, puede ver información de operaciones sobre el recurso en el contexto de una aplicación.

Creación de una aplicación en Application Manager

Utilice el siguiente procedimiento para crear una aplicación en Application Manager y para agregar recursos a esa aplicación.

Para crear una aplicación en Application Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. Elija Crear aplicación.
4. Elija una opción en la lista desplegable y complete los campos en la página que aparece.

Trabajo con aplicaciones

Application Manager es un componente de AWS Systems Manager. En esta sección se incluyen temas que le ayudarán a trabajar con aplicaciones de Application Manager y ver la información de operaciones acerca de los recursos de AWS.

Contenido

- Descripción general de la aplicación en Application Manager
- Administración de las instancias EC2 de aplicaciones
- Visualización de los recursos asociados a la aplicación
- Administrar el cumplimiento de las aplicaciones
- Uso de CloudWatch Application Insights para supervisar una aplicación
- Visualización de OpsItems para una aplicación
- Administración de los registros de la aplicación
- Utilice los manuales de procedimientos de automatización para solucionar los problemas de las aplicaciones

- Etiquetar recursos en Application Manager.
- Trabajo con plantillas y pilas de AWS CloudFormation en Application Manager
- Trabajar con clústeres en Application Manager.

Descripción general de la aplicación en Application Manager

En Application Manager, un componente de AWS Systems Manager, la pestaña Overview (Información general) muestra un resumen de las alarmas de Amazon CloudWatch, los elementos de trabajo operativos (OpsItems), CloudWatch Application Insights y el historial de manuales de procedimiento. Elija View all (Ver todo) para que las tarjetas abran la pestaña correspondiente en donde pueda ver todas las aplicaciones, la información, las alarmas, los OpsItems o el historial de manuales de procedimiento.

Acerca de Application Insights

CloudWatch Application Insights identifica y configura las métricas clave, los registros y las alarmas entre los recursos de aplicaciones y la pila de tecnología. Application Insights monitorea continuamente las métricas y los registros para detectar y relacionar anomalías y errores. Cuando el sistema detecta errores y anomalías, Application Insights genera CloudWatch Events que puede utilizar para configurar notificaciones o tomar medidas. Si elige el botón Edit configuration (Editar configuración) en la pestaña Monitoring (Monitoreo), el sistema abre la consola de CloudWatch Application Insights. Para obtener más información acerca de Application Insights, consulte ¿Qué es Información de aplicaciones de Amazon CloudWatch? en la Guía del usuario de Amazon CloudWatch.

Acerca de Explorador de costos

Application Manager se integra con AWS Cost Explorer, una característica de la administración de costos de AWS, a través del widget Costos y la pestaña Costos. Después de habilitar el Explorador de costos en la consola de administración de costos, el widget Costos y la pestaña Costos en Application Manager muestran los datos de costos de una aplicación o un componente de aplicación específicos que no están en contenedores. Puede utilizar filtros en el widget o la pestaña para ver los datos de costos según diferentes periodos de tiempo, niveles de granularidad y tipos de costos en un gráfico de barras o de líneas.

Puede habilitar esta característica si selecciona el botón Ir a la consola de administración de costos de AWS. De forma predeterminada, los datos se filtran a los últimos tres meses. En el caso de una aplicación que no esté en un contenedor, si elige el botón View all (Ver todo), Application Manager

abre la pestaña Resources (Recursos). Para aplicaciones en contenedores, el botón View all (Ver todo) abre la consola de AWS Cost Explorer.

Acciones que puedes realizar en esta página

Puede activar los siguientes widgets y acceder a la información sobre ellos en la pestaña Overview (Descripción general) de esta página. Cuando un widget está habilitado, elija View all (Ver todo) para ver los detalles de la aplicación pertinentes para esa área.

- En la sección Insights and Alarms (Información y alarmas), elija el número de una gravedad para abrir la pestaña Monitoring (Supervisión), donde puede ver más detalles sobre las alarmas de la gravedad elegida.
- En la sección Cost (Costo), elija View all (Ver todo) para abrir la pestaña Resources (Recursos), donde puede ver los datos de costo de una aplicación o componente de aplicación específicos.
- En la sección Compliance (Conformidad), elija View all (Ver todo) para abrir la pestaña Compliance (Conformidad), donde puede ver la información de cumplimiento de AWS Config y asociaciones de State Manager.

> **Note**
>
> Para ver los detalles de conformidad de las revisiones, elija directamente la pestaña Compliance (Conformidad). Luego puede ver los detalles de conformidad de las revisiones para los nodos administrados que utiliza la aplicación seleccionada.

- En la sección Runbooks (Manuales de procedimientos), elija un manual de procedimientos para abrirlo en la página Documents (Documentos) de Systems Manager donde puede ver más detalles sobre el documento.
- En la sección OpsItems, elija la severidad para abrir la pestaña OpsItems donde puede ver todos los OpsItems de la severidad elegida.
- Elija el botón View all (Ver todo) para abrir la pestaña correspondiente. Puede ver todas las alarmas, OpsItems o entradas al historial de manuales de procedimientos para la aplicación.

Para abrir la pestaña Overview (Información general)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.

3. En la sección Applications (Aplicaciones), seleccione una categoría. Si desea abrir una aplicación que creó manualmente en Application Manager, elija Custom applications (Aplicaciones personalizadas).
4. Elija la aplicación en la lista. Application Manager abre la pestaña Overview (Información general).

Administración de las instancias EC2 de aplicaciones

Application Manager se integra con Amazon Elastic Compute Cloud (Amazon EC2) para mostrar información sobre sus instancias en el contexto de una aplicación. Application Manager muestra el estado de la instancia y el estado de Amazon EC2 Auto Scaling para una aplicación seleccionada en un formato gráfico. La pestaña Instances (Instancias) también incluye una tabla con la siguiente información para cada instancia de la aplicación:

- Estado de la instancia (pendiente, deteniéndose, en ejecución, detenida)
- Estado de ping para SSM Agent
- Estado y nombre del manual de procedimientos de Automatización de Systems Manager más reciente procesado en la instancia
- Un recuento de las alarmas de registros de Amazon CloudWatch por estado.
  - `ALARM`: la métrica o expresión está fuera del umbral definido.
  - `OK`: la métrica o expresión está dentro del umbral definido.
  - `INSUFFICIENT_DATA`: la alarma acaba de iniciarse, la métrica no está disponible o no hay suficientes datos disponibles en la métrica para determinar el estado de la alarma.
- Estado de los grupos de escalado automático principal e individual

Si elige una instancia en la tabla All instances (Todas las instancias), Application Manager muestra información sobre esa instancia en cuatro pestañas:

- Details (Detalles): todos los detalles de la instancia de Amazon EC2, incluida la imagen de máquina de Amazon (AMI), la información de DNS, la información de la dirección IP y más.
- Estado: el estado actual proporcionado por el sistema EC2 y las comprobaciones de estado de la instancia.
- Execution history (Historial de ejecución): registros de ejecución de los manuales de procedimientos de Automatización de Systems Manager y las llamadas a la API procesadas por la instancia.

- CloudWatch alarms (Alarmas de CloudWatch): el nombre, el estado y más información de cualquier alarma de CloudWatch generada por la instancia.

Acciones que puedes realizar en esta página

En esta página puede realizar las siguientes acciones:

- Iniciar, detener y terminar instancias.
- Aplique una receta Chef.
- Adjunte o desasocie instancias de un grupo de escalado automático.
- Habilite las actualizaciones automatizadas para SSM Agent.

Para abrir la pestaña Instances (Instancias)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. En la sección Applications (Aplicaciones), seleccione una categoría. Si desea abrir una aplicación que creó manualmente en Application Manager, elija Custom applications (Aplicaciones personalizadas).
4. Elija la aplicación en la lista. Application Manager abre la pestaña Overview (Información general).
5. Seleccione la pestaña Instances.

Para ver los detalles de sus instancias de aplicación

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. En la sección Applications (Aplicaciones), seleccione una categoría. Si desea abrir una aplicación que creó manualmente en Application Manager, elija Custom applications (Aplicaciones personalizadas).
4. Elija la aplicación en la lista. Application Manager abre la pestaña Overview (Información general).

5. Seleccione la pestaña Instances.
6. Elija el botón situado junto a la instancia cuyos detalles desea consultar.
7. Revise los detalles de la instancia en la parte inferior de la página.

Para actualizar automáticamente SSM Agent

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. En la sección Applications (Aplicaciones), seleccione una categoría. Si desea abrir una aplicación que creó manualmente en Application Manager, elija Custom applications (Aplicaciones personalizadas).
4. Elija la aplicación en la lista. Application Manager abre la pestaña Overview (Información general).
5. Seleccione la pestaña Instances.
6. En el menú desplegable Acciones del agente, elija Configurar actualización de SSM Agent.
7. Seleccione Todas las instancias para configurar las actualizaciones de SSM Agent automáticas para todas las instancias administradas. De forma alternativa, elija Instancia para configurar las actualizaciones de SSM Agent de automatización para una sola instancia de la aplicación.
8. Seleccione la opción Habilitar la actualización automática.
9. En el menú desplegable Especificar programación, elija la programación que desea utilizar para las actualizaciones de SSM Agent.
10. Seleccione Configure (Configurar).

Visualización de los recursos asociados a la aplicación

En Application Manager, un componente de AWS Systems Manager, la pestaña Resources (Recursos) muestra los recursos de AWS en su aplicación. Si elige un componente de nivel superior, esta página muestra todos los recursos de ese componente y de cualquier subcomponente. Si elige un subcomponente, esta página muestra sólo los recursos asignados a ese subcomponente.

Acciones que puedes realizar en esta página

En esta página puede realizar las siguientes acciones:

- Elija un nombre de recurso para ver información al respecto, incluidos los detalles proporcionados por la consola donde se creó, las etiquetas, las alarmas de Amazon CloudWatch, los detalles de AWS Config y la información de registro de AWS CloudTrail.
- Elija el botón de opción junto a un nombre de recurso. A continuación, elija el botón Resource timeline (Plazo del recursos) para abrir la consola de AWS Config, donde puede ver información de conformidad sobre un recurso seleccionado.
- Si habilitó AWS Cost Explorer, la sección Cost Explorer muestra los datos de costos de una aplicación o componente de aplicación que no está en un contenedor específicos. Puede habilitar esta característica si selecciona el botón Ir a la consola de administración de costos de AWS. Utilice los filtros de esta sección para ver la información de costos de la aplicación.

Para abrir la pestaña Resources (Recursos)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. En la sección Applications (Aplicaciones), seleccione una categoría. Si desea abrir una aplicación que creó manualmente en Application Manager, elija Custom applications (Aplicaciones personalizadas).
4. Elija la aplicación en la lista. Application Manager abre la pestaña Overview (Información general).
5. Elija la pestaña Recursos.

Administrar el cumplimiento de las aplicaciones

En Application Manager, un componente de AWS Systems Manager, la página Configurations (Configuraciones) muestra información de conformidad de reglas de configuración y recursos de AWS Config. Esta página también muestra información de conformidad de la asociación de AWS Systems Manager State Manager. Puede elegir un recurso, una regla o una asociación para abrir la consola correspondiente para obtener más información. Esta página muestra información de conformidad de los últimos 90 días.

Acciones que puedes realizar en esta página

En esta página puede realizar las siguientes acciones:

- Elija un nombre de recurso para abrir la consola AWS Config, donde puede ver información de conformidad sobre un recurso seleccionado.
- Elija el botón de opción junto a un nombre de recurso. A continuación, elija el botón Resource timeline (Plazo del recursos) para abrir la consola de AWS Config, donde puede ver información de conformidad sobre un recurso seleccionado.
- En la sección Config rules compliance (Conformidad con reglas de configuración), puede hacer lo siguiente:
  - elegir un nombre de regla para abrir la consola de AWS Config, donde puede ver información sobre esa regla
  - elegir Add rules (Agregar reglas) para abrir la consola de AWS Config donde se puede crear una regla
  - elegir el botón de opción junto a un nombre de regla, elegir Actions (Acciones) y luego Manage remediation (Administrar corrección) para cambiar la acción de corrección de una regla
  - elegir el botón de opción junto a un nombre de regla, elegir Actions (Acciones) y luego Re-evaluate (Reevaluar) para que AWS Config ejecute una verificación de conformidad en la regla seleccionada
- En la sección Association compliance (Conformidad con la asociación), puede hacer lo siguiente:
  - elegir un nombre de asociación para abrir la página Associations) en la que puede ver información sobre esa asociación
  - elegir Create association (Crear asociación) para abrir Systems Manager State Manager donde puede crear una asociación
  - elegir el botón de opción junto a un nombre de asociación y elegir Apply association (Aplicar asociación) para comenzar inmediatamente todas las acciones especificadas en la asociación

Para abrir la pestaña Compliance (Conformidad)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. En la sección Applications (Aplicaciones), seleccione una categoría. Si desea abrir una aplicación que creó manualmente en Application Manager, elija Custom applications (Aplicaciones personalizadas).
4. Elija la aplicación en la lista. Application Manager abre la pestaña Overview (Información general).

5. Elija la pestaña Compliance (Conformidad).

Uso de CloudWatch Application Insights para supervisar una aplicación

En Application Manager, un componente de AWS Systems Manager, la pestaña Monitoring (Monitoreo) muestra Amazon CloudWatch Application Insights y los detalles de la alarma para los recursos de una aplicación.

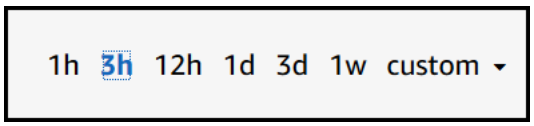
Acerca de Application Insights

CloudWatch Application Insights identifica y configura las métricas clave, los registros y las alarmas entre los recursos de aplicaciones y la pila de tecnología. Application Insights monitorea continuamente las métricas y los registros para detectar y relacionar anomalías y errores. Cuando el sistema detecta errores y anomalías, Application Insights genera CloudWatch Events que puede utilizar para configurar notificaciones o tomar medidas. Si elige el botón Edit configuration (Editar configuración) en la pestaña Monitoring (Monitoreo), el sistema abre la consola de CloudWatch Application Insights. Para obtener más información acerca de Application Insights, consulte ¿Qué es Información de aplicaciones de Amazon CloudWatch? en la Guía del usuario de Amazon CloudWatch.

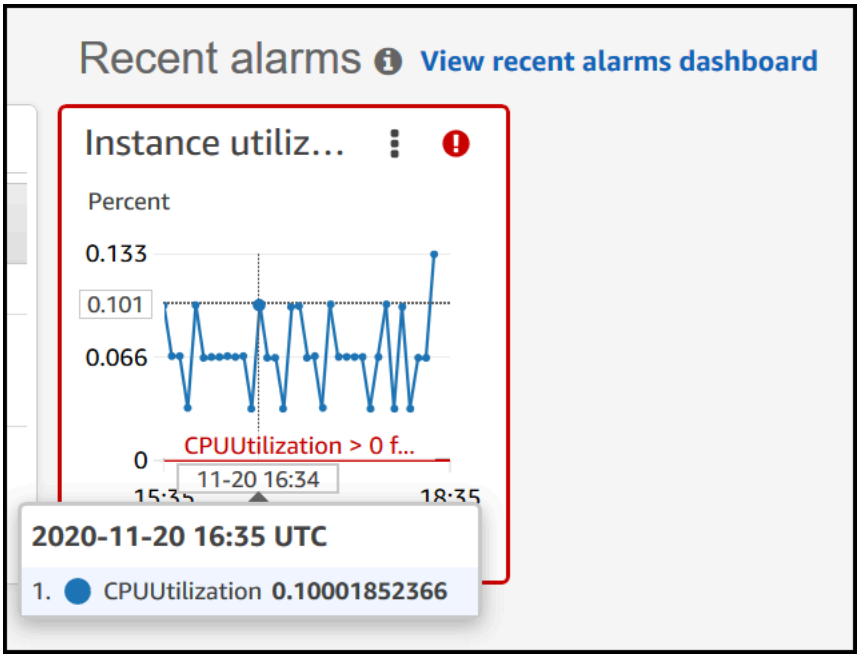
Acciones que puedes realizar en esta página

En esta página puede realizar las siguientes acciones:

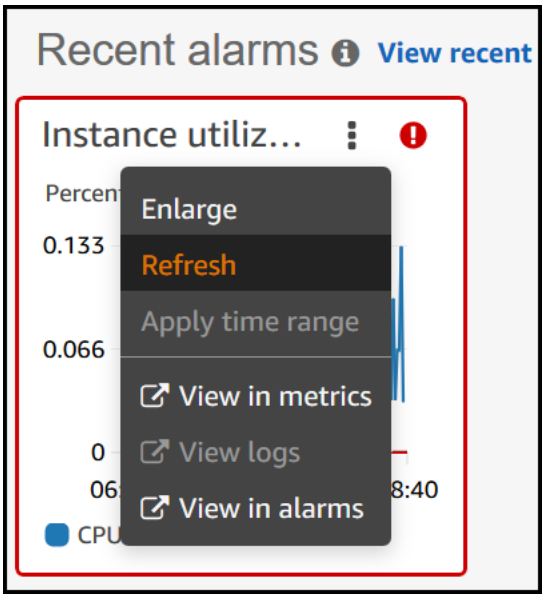
- Elegir un nombre de servicio en la sección Alarms by AWS service (Servicio de alarmas de) para abrir CloudWatch al servicio y la alarma seleccionados.
- Ajuste el periodo para los datos que se muestran en los widgets en la sección Recent alarms (Alarmas recientes) seleccionando uno de los valores predefinidos del periodo. Puede elegir la opción Custom (Personalizado) para definir su propio periodo.



- Pose el cursor sobre un widget en la sección Recent alarms (Alarmas recientes) para ver una ventana emergente de datos en una hora específica.



- Elija el menú de opciones de un widget para ver las opciones de visualización. Elija Enlarge (Ampliar) para expandir un widget. Elija Refresh (Actualizar) para actualizar los datos en un widget. Haga clic y arrastre el cursor en una pantalla de datos de un widget para seleccionar un rango específico. Luego, puede elegir Apply time range (Aplicar intervalo de tiempo).

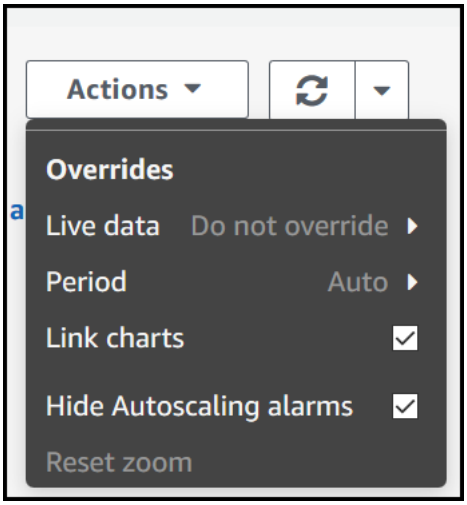


- Elija el menú Actions (Acciones) para ver pociones de Anulación de datos de alarmas, que incluyen lo siguiente:
  - Elegir si el widget muestra datos en directo. Los datos en directo son aquellos publicados en el último minuto que no se han agregado por completo. Si dichos datos están desactivados, solo se muestran los puntos de datos con un período de agregación de al menos un minuto en el

pasado. Por ejemplo, cuando se utilizan periodos de 5 minutos, el punto de datos para las 12:35 se agregaría de 12:35 a 12:40 y se mostraría a las 12:41.

Si los datos en directo están activados, se muestra el punto de datos más reciente tan pronto como se publiquen datos en el intervalo de agregación correspondiente. El punto de datos más reciente podría cambiar cada vez que actualice la pantalla según se publiquen nuevos datos en ese periodo de agregación.

- Especifique un periodo para los datos en directo.
- Vincule los gráficos en la sección Recent alarms (Alarmas recientes) para que cuando amplíe o reduzca la vista en un gráfico, el otro gráfico se amplíe o se reduzca al mismo tiempo. Puede desvincular los gráficos para limitar la ampliación a un gráfico.
- Ocultar alarmas de Auto Scaling.



Para abrir la pestaña Monitoring (Supervisión)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. En la sección Applications (Aplicaciones), seleccione una categoría. Si desea abrir una aplicación que creó manualmente en Application Manager, elija Custom applications (Aplicaciones personalizadas).
4. Elija la aplicación en la lista. Application Manager abre la pestaña Overview (Información general).
5. Elija la pestaña Monitorización.

Visualización de OpsItems para una aplicación

En Application Manager, un componente de AWS Systems Manager, la pestaña OpsItems muestra elementos de trabajo operativos (OpsItems) para los recursos de la aplicación seleccionada. Puede configurar OpsCenter de Systems Manager para crear OpsItems automáticamente desde alarmas de Amazon CloudWatch y eventos de Amazon EventBridge. También puede crear OpsItems manualmente.

Acciones que puede realizar en esta pestaña

En esta página puede realizar las siguientes acciones:

- Filtrar la lista de OpsItems utilizando el campo de búsqueda. Puede filtrar por nombre de OpsItem, ID, ID de origen o severidad. También puede filtrar la lista en función del estado. OpsItems es compatible con los siguientes estados: Abierto, En curso, Abierto y En curso, Resuelto o Todo.
- Cambiar el estado de un OpsItem a través del botón de opción junto a él y luego elegir una opción en el menú Set status (Establecer estado).
- Abrir OpsCenter de Systems Manager para crear un OpsItem por medio de la opción Create OpsItem (Crear OpsItem).

Para abrir la pestaña OpsItems (Información general)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. En la sección Applications (Aplicaciones), seleccione una categoría. Si desea abrir una aplicación que creó manualmente en Application Manager, elija Custom applications (Aplicaciones personalizadas).
4. Elija la aplicación en la lista. Application Manager abre la pestaña Overview (Información general).
5. Elija la pestaña OpsItems.

Administración de los registros de la aplicación

En Application Manager, un componente de AWS Systems Manager, la pestaña Logs (Registros) muestra una lista de grupos de registros de Amazon CloudWatch Logs.

Acciones que puede realizar en esta pestaña

En esta página puede realizar las siguientes acciones:

- Elegir un nombre de grupo de registros para abrirlo en CloudWatch Logs. A continuación, puede elegir un flujo de registro para ver los registros de un recurso en el contexto de una aplicación.
- Elija Create log groups (Crear grupos de registros) para crear un grupo de registros en CloudWatch Logs.

Para abrir la pestaña Logs (Registros)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. En la sección Applications (Aplicaciones), seleccione una categoría. Si desea abrir una aplicación que creó manualmente en Application Manager, elija Custom applications (Aplicaciones personalizadas).
4. Elija la aplicación en la lista. Application Manager abre la pestaña Overview (Información general).
5. Elija la pestaña Logs (Registros).

Utilice los manuales de procedimientos de automatización para solucionar los problemas de las aplicaciones

Puede solucionar los problemas con los recursos de AWS desde Application Manager, una herramienta de AWS Systems Manager, mediante los manuales de procedimientos de Automatización. Un manual de procedimientos de Automation define las acciones que Systems Manager realiza en las instancias administradas y en otros recursos de AWS cuando se ejecuta una automatización. Automatización es una herramienta de AWS Systems Manager. Un manual de procedimientos contiene uno o más pasos que se ejecutan en orden secuencial. Cada paso se construye en torno a una sola acción. La salida de un paso se puede utilizar como entrada en un paso posterior.

Cuando elige Start runbook (Comenzar manual de procedimientos) desde una aplicación de Application Manager, el sistema muestra una lista filtrada de manuales de procedimientos disponibles basada en el tipo de recursos de la aplicación o clúster. Cuando elija el manual de

procedimientos que desea comenzar, Systems Manager abre la página Execute automation document (Ejecutar documento de automatización).

Application Manager incluye las siguientes mejoras para trabajar con runbooks.

- Si elige el nombre de un recurso en Application Manager y luego elige Execute runbook (Ejecutar runbook), el sistema muestra una lista filtrada de runbooks para ese tipo de recurso.
- Puede iniciar una automatización en todos los recursos del mismo tipo mediante la elección de un runbook de la lista y, a continuación, elegir Run for resources of same type (Ejecutar para recursos del mismo tipo).

Antes de empezar

Antes de comenzar un manual de procedimientos desde Application Manager, realice una de las siguientes opciones:

- Compruebe que tiene los permisos correctos para iniciar manuales de procedimientos. Para obtener más información, consulte Configuración de Automation.
- Revise la documentación del procedimiento de automatización acerca del inicio de manuales de procedimientos. Para obtener más información, consulte Ejecución de una operación automatizada con la tecnología de automatización de Systems Manager.

Para comenzar un manual de procedimientos desde Application Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. En la sección Applications (Aplicaciones), seleccione una categoría. Si desea abrir una aplicación que creó manualmente en Application Manager, elija Custom applications (Aplicaciones personalizadas).
4. Elija la aplicación en la lista. Application Manager abre la pestaña Overview (Información general).
5. Seleccione Iniciar el manual de procedimientos. Application Manager abre la ventana emergente del widget de automatización. Para obtener información sobre las opciones en el widget de automatización, consulte Ejecución de una operación automatizada con la tecnología de automatización de Systems Manager.

Etiquetar recursos en Application Manager.

Puede agregar o eliminar etiquetas rápidamente en aplicaciones y recursos de AWS en Application Manager. Utilice el siguiente procedimiento para agregar o eliminar una etiqueta de una aplicación y de todos los recursos de AWS en esa aplicación.

Para agregar o eliminar una etiqueta de una aplicación y de todos los recursos en la aplicación

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. En la sección Applications (Aplicaciones), seleccione una categoría. Si desea abrir una aplicación que creó manualmente en Application Manager, elija Custom applications (Aplicaciones personalizadas).
4. Elija la aplicación en la lista. Application Manager abre la pestaña Overview (Información general).
5. En la sección Application information (Información de la aplicación), elija el número debajo de Application tags (Etiquetas de la aplicación). Si no se asignan etiquetas a la aplicación, el número es cero.
6. Para agregar una etiqueta, elija Add tag (Agregar etiqueta nueva). Especifique una clave y un valor opcional. Para eliminar una etiqueta, elija Remove (Eliminar).
7. Seleccione Guardar.

Utilice el siguiente procedimiento para agregar una etiqueta o eliminarla de un recurso específico en Application Manager.

Para agregar una etiqueta o eliminarla de un recurso

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. En la sección Applications (Aplicaciones), seleccione una categoría. Si desea abrir una aplicación que creó manualmente en Application Manager, elija Custom applications (Aplicaciones personalizadas).
4. Elija la aplicación en la lista. Application Manager abre la pestaña Overview (Información general).

5. Elija la pestaña Recursos.
6. Elija un nombre de recurso.
7. En la sección Tags (Etiquetas), elija Edit (Editar).
8. Para agregar una etiqueta, elija Add tag (Agregar etiqueta). Especifique una clave y un valor opcional. Para eliminar una etiqueta, elija Remove (Eliminar).
9. Seleccione Guardar.

Trabajo con plantillas y pilas de AWS CloudFormation en Application Manager

Application Manager, una herramienta de AWS Systems Manager, lo ayuda a aprovisionar y administrar recursos para sus aplicaciones mediante la integración con AWS CloudFormation. Puede crear, editar y eliminar platillas y pilas de AWS CloudFormation en Application Manager. Una pila es una colección de recursos de AWS, que puede administrar como una única unidad. Esto significa que puede crear, actualizar o eliminar una colección de recursos de AWS mediante el uso de pilas de CloudFormation. Una plantilla es un archivo de texto con formato en JSON o YAML que especifica los recursos que desea aprovisionar en sus pilas.

Application Manager también incluye una biblioteca de plantillas donde puede clonar, crear y almacenar plantillas. Application Manager y CloudFormation muestran la misma información sobre el estado actual de una pila. Las plantillas y las actualizaciones de plantillas se almacenan en Systems Manager hasta que aprovisione la pila, momento en el que los cambios también se muestran en CloudFormation.

Después de crear una pila en Application Manager, la página CloudFormation stacks (Pilas de CloudFormation) muestra información útil al respecto. Esto incluye la plantilla utilizada para crearla, un recuento de OpsItems para los recursos de su pila, el estado de pila, y el estado de desviación.

Acerca de Explorador de costos

Application Manager se integra con AWS Cost Explorer, una característica de Administración de costos de AWS, mediante el widget Cost. Después de habilitar Explorador de costos en la consola de administración de costo, el widget Cost en Application Manager muestra los datos de costos de una aplicación o componente de aplicación específicos que no están en contenedores. Puede utilizar filtros en el widget para ver los datos de costos según diferentes periodos de tiempo, detalles y tipos de costos en un gráfico de barras o líneas.

Puede habilitar esta característica si selecciona el botón Ir a la consola de administración de costos de AWS. De forma predeterminada, los datos se filtran a los últimos tres meses. En el caso de una

aplicación que no esté en un contenedor, si elige el botón View all (Ver todo), Application Manager abre la pestaña Resources (Recursos). Para aplicaciones en contenedores, el botón View all (Ver todo) abre la consola de AWS Cost Explorer.

> **Note**
>
> Explorador de costos usa etiquetas para hacer un seguimiento de los costos de la aplicación. Si la aplicación de AWS CloudFormation basada en pilas no está configurada con la clave de etiqueta `AppManagerCFNStackKey`, Explorador de costos no presenta datos de costos precisos en Application Manager. Si no se detecta la clave de la etiqueta `AppManagerCFNStackKey`, en la consola necesitará agregar la etiqueta a la pila de CloudFormation para permitir el seguimiento de los costos. Al agregarla, se asigna la clave de etiqueta al nombre de recurso de Amazon (ARN) de la pila y se habilita el widget Cost para mostrar datos de costos precisos.

> **Important**
>
> Cuando agregue la etiqueta `AppManagerCFNStackKey`, se desencadenará una actualización de la pila. Las configuraciones manuales que se hayan realizado después del despliegue original de la pila no se reflejarán después de agregar la etiqueta de usuario. Para obtener más información sobre los comportamientos de actualización de un recurso, consulte Comportamientos de actualización de los recursos de la pila en la Guía del usuario de AWS CloudFormation.

Antes de empezar

Utilice los siguientes enlaces para obtener información acerca de los conceptos de CloudFormation antes de crear, editar o eliminar plantillas y pilas de CloudFormation mediante Application Manager.

- ¿Qué es AWS CloudFormation?
- Prácticas recomendadas de AWS CloudFormation
- Más información sobre los aspectos básicos de las plantillas
- Trabajo con pilas de AWS CloudFormation
- Trabajo con plantillas de AWS CloudFormation
- Plantillas de ejemplo

Temas

- Utilización de Application Manager para gestionar plantillas de AWS CloudFormation
- Utilización de Application Manager para gestionar pilas de AWS CloudFormation

Utilización de Application Manager para gestionar plantillas de AWS CloudFormation

Application Manager, una herramienta de AWS Systems Manager, incluye una biblioteca de plantillas y otras herramientas para ayudarlo a administrar plantillas de AWS CloudFormation. Esta sección incluye la siguiente información.

Temas

- Trabajar con la biblioteca de plantillas
- Creación de una plantilla
- Edición de una plantilla

Trabajar con la biblioteca de plantillas

La biblioteca de plantillas de Application Manager proporciona herramientas para ayudarlo a ver, crear, editar, eliminar y clonar plantillas. También puede aprovisionar pilas directamente desde la biblioteca de plantillas. Las plantillas se almacenan como documentos de Systems Manager (SSM) de tipo `CloudFormation`. Cuando almacena plantillas como documentos de SSM, puede utilizar controles de versión para trabajar con diferentes versiones de una plantilla. También puede establecer permisos y compartir plantillas. Después de aprovisionar correctamente una pila, la pila y la plantilla están disponibles en Application Manager y en CloudFormation.

Antes de empezar

Le recomendamos que lea los siguientes temas para obtener más información acerca de los documentos de SSM antes de comenzar a trabajar con plantillas de CloudFormation en Application Manager.

- Documentos de AWS Systems Manager
- Uso compartido de documentos de SSM
- Prácticas recomendadas para documentos de SSM compartidos

Para ver la biblioteca de plantillas en Application Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. Elija Biblioteca de plantillas de CloudFormation.

Creación de una plantilla

El siguiente procedimiento describe cómo crear una plantilla de CloudFormation en Application Manager. Cuando crea una plantilla, ingresa los detalles de la pila de la plantilla en JSON o YAML. Si no está familiarizado con JSON ni YAML, puede utilizar AWS Infrastructure Composer, una herramienta para crear y modificar plantillas visualmente. Para obtener más información, consulte Creación visual de plantillas con Infrastructure Composer en la Guía del usuario de AWS CloudFormation. Para obtener más información sobre la estructura y la sintaxis de una plantilla, consulte Secciones de la plantilla de AWS CloudFormation en la Guía del usuario de AWS CloudFormation.

También puede construir una plantilla a partir de varios fragmentos de plantillas. Los fragmentos de plantillan son ejemplos que demuestran cómo escribir plantillas para un recurso en concreto. Por ejemplo, puede ver fragmentos de instancias de Amazon Elastic Compute Cloud (Amazon EC2), dominios de Amazon Simple Storage Service (Amazon S3), mapeos de AWS CloudFormation y mucho más. Los fragmentos se agrupan por recurso. Puede encontrar fragmentos de AWS CloudFormation de propósito general en la sección Fragmentos generales de plantillas en la Guía del usuario de AWS CloudFormation.

Creación de una plantilla de CloudFormation enApplication Manager (consola)

Utilice el siguiente procedimiento para crear una plantilla de CloudFormation en Application Manager mediante la AWS Management Console.

Para crear una plantilla de CloudFormation en Application Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. Seleccione Biblioteca de plantillas de CloudFormation y, a continuación, elija Crear plantilla o elija una plantilla existente y, luego, seleccione Acciones, Clonar.

4. En Name (Nombre), ingrese un nombre para la plantilla que lo ayude a identificar los recursos que crea o el propósito de la pila.
5. (Opcional) En Version name (Nombre de la versión), ingrese un nombre o un número para identificar la versión de la plantilla.
6. En la sección Code editor (Editor de código), elija YAML o JSON y, a continuación, ingrese o copie y pegue el código de la plantilla.
7. (Opcional) En la sección Tags (Etiquetas), aplique a la plantilla uno o más pares de nombre y valor de clave de etiqueta.
8. (Opcional) En la sección Permissions (Permisos), ingrese un ID de Cuenta de AWS y elija Add account (Agregar cuenta). Esta acción proporciona permiso de lectura a la plantilla. El propietario de la cuenta puede aprovisionar y clonar la plantilla, pero no puede editarla ni eliminarla.
9. Seleccione Crear. La plantilla se guarda en el servicio de documentos de Systems Manager (SSM).

Creación de una plantilla de CloudFormation en Application Manager (línea de comandos)

Después de crear el contenido de su plantilla de CloudFormation en JSON o YAML, puede utilizar AWS Command Line Interface (AWS CLI) o Herramientas de AWS para PowerShell para guardar la plantilla como documento de SSM. Reemplace cada *`example resource placeholder`* con su propia información.

Antes de empezar

Si aún no lo ha hecho, instale y configure la AWS CLI o Herramientas de AWS para PowerShell. Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para 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 se ejecuta correctamente, el comando devolverá una respuesta similar a la siguiente.

```
{
    "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"
            }
```

```
        ]
    }
```

Edición de una plantilla

Utilice el siguiente procedimiento para crear una plantilla de CloudFormation en Application Manager. Los cambios de plantilla están disponibles en CloudFormation después de aprovisionar una pila que utiliza la plantilla actualizada.

Para editar una plantilla de CloudFormation en Application Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. Elija Biblioteca de plantillas de CloudFormation.
4. Elija una plantilla y, a continuación, elija Actions (Acciones), Edit (Editar). No puede cambiar el nombre de una plantilla, pero puede cambiar todos los demás detalles.
5. Seleccione Save. La plantilla se guarda en el servicio de documentos de Systems Manager.

Utilización de Application Manager para gestionar pilas de AWS CloudFormation

Application Manager, una herramienta de AWS Systems Manager, lo ayuda a aprovisionar y administrar recursos para sus aplicaciones mediante la integración con AWS CloudFormation. Puede crear, editar y eliminar platillas y pilas de CloudFormation en Application Manager. Una pila es una colección de recursos de AWS, que puede administrar como una única unidad. Esto significa que puede crear, actualizar o eliminar una colección de recursos de AWS mediante el uso de pilas de CloudFormation. Una plantilla es un archivo de texto con formato en JSON o YAML que especifica los recursos que desea aprovisionar en sus pilas. Esta sección incluye la siguiente información.

Temas

- Creación de una pila
- Actualización de una pila

Creación de una pila

Los siguientes procedimientos describen cómo crear una pila de CloudFormation mediante Application Manager. Una pila se basa en una plantilla. Cuando crea una pila, puede elegir

una plantilla existente o crear una nueva. Después de crear la pila, el sistema intenta crear inmediatamente los recursos identificados en la pila. Después de que el sistema aprovisione correctamente los recursos, la plantilla y la pila están disponibles para ver y editar en Application Manager y en CloudFormation.

> **Note**
>
> Puede utilizar Application Manager para crear una pila sin costo, pero se le cobrará por los recursos de AWS creados en la pila.

Creación de una pila de CloudFormation mediante Application Manager (consola)

Utilice el siguiente procedimiento para crear una pila mediante Application Manager en la AWS Management Console.

Para crear una pila de CloudFormation

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. Elija Crear aplicación, Pila de CloudFormation.
4. En la sección Prepare a template (Preparar una plantilla), elija una opción. Si elige Use an existing template (Utilizar una plantilla existente) puede utilizar las pestañas de la sección Choose a template (Elegir una plantilla) para localizar la plantilla que desea. Si elige una de las otras opciones, complete el asistente para preparar una plantilla.
5. Seleccione el botón que está junto al nombre de una plantilla y, a continuación, elija Siguiente.
6. En la página Specify template details (Especificar detalles de la plantilla), compruebe los detalles de la plantilla para asegurarse de que el proceso crea los recursos que desea.
   - (Opcional) En la sección Tags (Etiquetas), aplique a la plantilla uno o más pares de nombre y valor de clave de etiqueta.
   - Las etiquetas son metadatos opcionales que usted asigna a un recurso. Con las etiquetas puede clasificar un recurso de diferentes maneras, por ejemplo, según su finalidad, propietario o entorno.
   - Elija Siguiente.

7. En la página Edit stack details (Editar detalles de la pila), para Stack name (Nombre de la pila), ingrese un nombre que lo ayude a identificar los recursos creados por la pila o su propósito.

   - La sección Parameters (Parámetros) incluye todos los parámetros opcionales y obligatorios especificados en la plantilla. Ingrese uno o varios parámetros en cada campo.
   - (Opcional) En la sección Tags (Etiquetas), aplique a la pila uno o varios pares de nombre-valor de claves de etiqueta.
   - (Opcional) En la sección Permissions (Permisos), especifique un rol de AWS Identity and Access Management (IAM) o un Nombre de recurso de Amazon (ARN) de IAM. El sistema utiliza el rol de servicio especificado para crear todos los recursos especificados en la pila. Si no especifica un rol de IAM, AWS CloudFormation utiliza una sesión temporal que el sistema genera a partir de sus credenciales de usuario. Para obtener más información acerca de este rol de IAM, consulte Rol de servicio de AWS CloudFormation en la Guía del usuario de AWS CloudFormation.
   - Elija Siguiente.
8. En la página Review and provision (Revisar y aprovisionar), revise todos los detalles de la pila. Elija el botón Edit (Editar) de esta página para realizar cambios.
9. Elija Provision stack (Aprovisionar pilas).

Application Manager abre la página CloudFormation stacks (Pilas de CloudFormation) y el estado de la creación e implementación de la pila. Si CloudFormation no puede crear ni aprovisionar la pila, consulte los siguientes temas en la Guía del usuario de AWS CloudFormation.

- Códigos de estado de pilas
- Solución de problemas de AWS CloudFormation

Una vez que los recursos de la pila están aprovisionados y en ejecución, los usuarios pueden editar los recursos directamente usando el servicio subyacente que ha creado el recurso. Por ejemplo, un usuario puede utilizar la consola de Amazon Elastic Compute Cloud (Amazon EC2) para actualizar una instancia de servidor que se creó como parte de una pila de CloudFormation. Algunos cambios pueden ser accidentales, mientras que otros pueden realizarse a propósito para dar respuesta a eventos operativos en los que el tiempo es crucial. Sea cual sea el motivo, los cambios realizados fuera de CloudFormation pueden complicar las operaciones de actualización de la pila o de eliminación. Puede utilizar la detección de desviaciones o estado de desviaciones para identificar los recursos de pila en los que se han introducido cambios de configuración fuera

de la administración de CloudFormation. Para obtener más información acerca del estado de las desviaciones, consulte Detección de cambios de configuración no administrados en pilas y recursos.

Creación de una pila de CloudFormation mediante Application Manager (línea de comandos)

Utilice el siguiente procedimiento de AWS Command Line Interface (AWS CLI) para aprovisionar una pila mediante una plantilla de CloudFormation que se almacena como un documento de SSM en Systems Manager. Reemplace cada *example resource placeholder* con su propia información. Para obtener información acerca de otros procedimientos de AWS CLI para la creación de pilas, consulte Creación de una pila en la Guía del usuario de AWS CloudFormation.

Antes de empezar

Si aún no lo ha hecho, instale y configure la AWS CLI o Herramientas de AWS para PowerShell. Para obtener información, consulte Instalación o actualización de la última versión de la AWS CLI e Instalación de Herramientas de AWS para 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" `
```

Actualización de una pila

Puede implementar actualizaciones en una pila de CloudFormation editando directamente la pila en Application Manager. Con una actualización directa, puede especificar actualizaciones de una

plantilla o parámetros de entrada. Después de guardar e implementar los cambios, CloudFormation actualiza los recursos de AWS según los cambios especificados.

Puede obtener una vista previa de los cambios que CloudFormation realizará en la pila antes de actualizarlo, mediante conjuntos de cambios. Para obtener más información, consulte Actualización de pilas con conjuntos de cambios en la Guía del usuario de AWS CloudFormation.

Para actualizar una pila de CloudFormation en Application Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. Seleccione el botón de opción que está junto al nombre de una aplicación y, a continuación, elija Acciones, Actualizar pila.
4. En la página Specify template source (Especificar origen de la plantilla), elija una de las siguientes opciones y, a continuación, elija Next (Siguiente).
   - Elija Use the template code currently provisioned in the stack (Utilizar el código de la plantilla aprovisionado actualmente en la pila) para ver una plantilla. Elija una versión de plantilla en la lista Versions (Versiones) y luego elija Next (Siguiente).
   - Elija Switch to a different template (Cambiar a una plantilla diferente) para elegir o crear una nueva plantilla para la pila.
5. Cuando termine de realizar los cambios en la plantilla, elija Next (Siguiente).
6. En la página Edit stack details (Editar detalles de la pila), puede editar parámetros, etiquetas y permisos. No puede modificar el nombre de la pila. Realice sus cambios y elija Next (Siguiente).
7. En la página Review and provision (Revisar y aprovisionar) revise todos los detalles de la pila y luego elija Provision stack (Aprovisionar pila).

Trabajar con clústeres en Application Manager.

Esta sección incluye temas que le ayudarán a trabajar con clústeres de contenedores de Amazon Elastic Container Service (Amazon ECS) y Amazon Elastic Kubernetes Service (Amazon EKS) en Application Manager, un componente de AWS Systems Manager.

Contenido

- Trabajo con Amazon ECS en Application Manager
- Trabajo con Amazon EKS en Application Manager

- Trabajar con manuales de procedimientos para clústeres

Trabajo con Amazon ECS en Application Manager

Con Application Manager, una herramienta de AWS Systems Manager, puede ver y administrar su infraestructura de clúster de Amazon Elastic Container Service (Amazon ECS). Application Manager aplica una etiqueta a su clúster de Amazon ECS con un Nombre de recurso de Amazon (ARN) del clúster como valor de la etiqueta. Application Manager proporciona una vista del tiempo de ejecución de los recursos de computación, redes y almacenamiento del clúster.

> **Note**
>
> No puede administrar ni ver la información de operaciones sobre sus contenedores en Application Manager. Solo puede administrar y ver información de operaciones sobre la infraestructura que aloja sus recursos de Amazon ECS.

Acciones que puedes realizar en esta página

En esta página puede realizar las siguientes acciones:

- Elija Manage cluster (Administrar clúster) para abrir el clúster en Amazon ECS.
- Elija View all (Ver todo) para ver una lista de recursos del clúster.
- Elija Ver en CloudWatchView in CloudWatch (Ver en CloudWatch) para ver las alarmas de recursos en Amazon CloudWatch.
- Elija Manage nodes (Administrar nodos) o Manage Fargate profiles (Administrar perfiles de Fargate) para ver estos recursos en Amazon ECS.
- Elija un ID de recurso para ver información detallada sobre él en la consola donde se creó.
- Ver una lista de OpsItems relacionados con sus clústeres.
- Vea un historial de manuales de procedimientos que se han ejecutado en sus clústeres.

Para abrir un clúster de ECS

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.

3. En la sección Container clusters (Clústeres de contenedores), elija ECS clusters (Clústeres de ECS).
4. Elija un clúster en la lista. Application Manager abre la pestaña Overview (Información general).

Trabajo con Amazon EKS en Application Manager

Application Manager, una herramienta de AWS Systems Manager, se integra con Amazon Elastic Kubernetes Service (Amazon EKS) para proporcionar información sobre el estado de las infraestructuras de clúster de Amazon EKS. Application Manager aplica una etiqueta a su clúster de Amazon EKS con un Nombre de recurso de Amazon (ARN) del clúster como valor de la etiqueta. Application Manager proporciona una vista en tiempo de ejecución de los recursos de computación, redes y de almacenamiento de un clúster.

> **Note**
>
> No puede administrar ni ver la información de operaciones sobre sus pods o contenedores de Amazon EKS en Application Manager. Solo puede administrar y ver información de operaciones sobre la infraestructura que aloja sus recursos de Amazon EKS.

Acciones que puedes realizar en esta página

En esta página puede realizar las siguientes acciones:

- Elija Manage cluster (Administrar clúster) para abrir el clúster en Amazon EKS.
- Elija View all (Ver todo) para ver una lista de recursos del clúster.
- Elija Ver en CloudWatchView in CloudWatch (Ver en CloudWatch) para ver las alarmas de recursos en Amazon CloudWatch.
- Elija Manage nodes (Administrar nodos) o Manage Fargate profiles (Administrar perfiles de Fargate) para ver estos recursos en Amazon EKS.
- Elija un ID de recurso para ver información detallada sobre él en la consola donde se creó.
- Ver una lista de OpsItems relacionados con sus clústeres.
- Vea un historial de manuales de procedimientos que se han ejecutado en sus clústeres.

Para abrir una aplicación Clústeres de EKS

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Application Manager.
3. En la sección Container clusters (Clústeres de contenedores), elija EKS clusters (Clústeres de EKS).
4. Elija un clúster en la lista. Application Manager abre la pestaña Overview (Información general).

Trabajar con manuales de procedimientos para clústeres

Puede solucionar los problemas con los recursos de AWS desde Application Manager, una herramienta de AWS Systems Manager, mediante los manuales de procedimientos de Automatización de Systems Manager. Cuando elige Start runbook (Comenzar manual de procedimientos) de un clúster de Application Manager, el sistema muestra una lista filtrada de manuales de procedimientos basada en el tipo de recursos del clúster. Cuando elija el manual de procedimientos que desea comenzar, Systems Manager abre la página Execute automation document (Ejecutar documento de automatización).

Antes de empezar

Antes de comenzar un manual de procedimientos desde Application Manager, realice una de las siguientes opciones:

- Compruebe que tiene los permisos correctos para iniciar manuales de procedimientos. Para obtener más información, consulte Configuración de Automation.
- Revise la documentación del procedimiento de automatización acerca del inicio de manuales de procedimientos. Para obtener más información, consulte Ejecución de una operación automatizada con la tecnología de automatización de Systems Manager.
- Si tiene intención de comenzar manuales de procedimientos en varios recursos a la vez, revise la documentación sobre el uso de destinos y controles de tasa. Para obtener más información, consulte Ejecución de operaciones automatizadas a escala.

Para comenzar un manual de procedimientos para clústeres desde Application Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.

2. En el panel de navegación, elija Application Manager.
3. En la sección Container clusters (Clústeres de contenedores) elija un tipo de contenedor.
4. Elija el clúster en la lista. Application Manager abre la pestaña Overview (Información general).
5. En la pestaña Runbooks (Manuales de procedimientos), elija Start runbook (Iniciar manual de procedimientos). Application Manager abre la página Execute automation document (Ejecutar documento de automatización) en una nueva pestaña. Para obtener información acerca de las opciones en la página Execute automation document (Ejecutar documento de automatización), consulte Ejecución de una operación automatizada con la tecnología de automatización de Systems Manager.

AWS Systems Manager Parameter Store

Parameter Store, una herramienta de AWS Systems Manager, proporciona un almacenamiento seguro y jerárquico para la administración de los datos de configuración y de los secretos. Puede almacenar datos como contraseñas, cadenas de base de datos, ID de Amazon Machine Image (AMI) y códigos de licencia como valores de parámetros. Puede almacenar valores como texto sin formato o como datos cifrados. Puede hacer referencia a parámetros de Systems Manager en los scripts, los comandos, los documentos de SSM y los flujos de trabajo de configuración y automatización utilizando el nombre único que especificó cuando creó el parámetro. Para comenzar a utilizar Parameter Store, abra la consola de Systems Manager. En el panel de navegación, elija Parameter Store.

Parameter Store también se integran con Secrets Manager. Puede recuperar secretos de Secrets Manager cuando utiliza otros Servicios de AWS que admiten las referencias a los parámetros de Parameter Store. Para obtener más información, consulte Referencia a los secretos de AWS Secrets Manager desde los parámetros de Parameter Store.

> **Note**
>
> Para implementar ciclos de vida de rotación de contraseñas, utilice AWS Secrets Manager. Puede rotar, administrar y recuperar credenciales de bases de datos, claves de API y otros datos confidenciales durante todo su ciclo de vida con Secrets Manager. Para obtener más información, consulte ¿Qué es AWS Secrets Manager? en la Guía del usuario de AWS Secrets Manager.

¿Cómo puede Parameter Store beneficiar a mi organización?

Parameter Store ofrece las ventajas siguientes:

- Utilice un servicio de administración de secretos alojado seguro y escalable, sin servidores que administrar.
- Mejore el nivel de seguridad separando los datos del código.
- Almacene datos de configuración y cadenas seguras en jerarquías y realice un seguimiento de las versiones.
- Controle y audite el acceso granular de forma detallada.
- Almacene los parámetros de forma fiable porque Parameter Store se aloja en varias zonas de disponibilidad en una Región de AWS.

¿Quién debe utilizar Parameter Store?

- Cualquier cliente de AWS que desea tener una forma centralizada de administrar los datos de configuración.
- Desarrolladores de software que desean almacenar diferentes inicios de sesión y flujos de referencia.
- Administradores que desean recibir notificaciones cuando sus secretos y contraseñas se cambian o no.

¿Cuáles son las características de Parameter Store?

- Notificación de cambio

  Puede configurar las notificaciones de cambios y active las acciones automatizadas tanto para los parámetros como para sus políticas correspondientes. Para obtener más información, consulte Configuración de notificaciones o activación de acciones en función de los eventos de Parameter Store.
- Organización de parámetros

  Puede etiquetar los parámetros de manera individual para facilitar la identificación de uno o varios parámetros en función de las etiquetas que les haya asignado. Por ejemplo, puede etiquetar los parámetros de etiqueta para departamentos o entornos específicos.
- Versiones de etiquetas

Puede asociar un alias para las versiones del parámetro mediante la creación de etiquetas. Las etiquetas pueden ayudarlo a recordar el propósito de una versión de un parámetro cuando hay varias versiones.

- Validación de datos

  Puede crear parámetros que apunten a una instancia de Amazon Elastic Compute Cloud (Amazon EC2) y Parameter Store valida estos parámetros para asegurarse de que hace referencia al tipo de recurso esperado, que el recurso existe y que el cliente tiene permiso para usar el recurso. Por ejemplo, puede crear un parámetro con el ID de una Amazon Machine Image (AMI) como un valor con tipo de datos `aws:ec2:image`, y Parameter Store realiza una operación de validación asíncrona para asegurarse de que el valor del parámetro cumple los requisitos de formato para un ID de AMI y que la AMI específica está disponible en su Cuenta de AWS.

- Secretos de referencia

  Parameter Store está integrado con AWS Secrets Manager, lo que permite recuperar secretos de Secrets Manager cuando utiliza otros Servicios de AWS que admiten las referencias a los parámetros de Parameter Store.

- Compartir parámetros con otras cuentas

  Si lo desea, puede centralizar los datos de configuración en una sola cuenta Cuenta de AWS y compartir los parámetros con otras cuentas que necesiten acceder a ellos.

- Accesible desde otros Servicios de AWS

  Puede utilizar parámetros de Parameter Store con otras herramientas de Systems Manager y otros Servicios de AWS para recuperar secretos y datos de configuración del almacén central. Los parámetros funcionan con las herramientas de Systems Manager como Run Command, Automatización y State Manager, herramientas de AWS Systems Manager. También puede hacer referencia a parámetros en otros Servicios de AWS, incluidos los siguientes:

  - 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
- Integración con otros Servicios de AWS

  Configure la integración con los siguientes Servicios de AWS para el cifrado, la notificación, la supervisión y la auditoría:

  - AWS Key Management Service (AWS KMS)
  - Amazon Simple Notification Service (Amazon SNS)
  - Amazon CloudWatch: para obtener más información, consulte Configuración de reglas de EventBridge para parámetros y políticas de parámetros.
  - Amazon EventBridge: para obtener más información, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS y Referencia: patrones y tipos de eventos de Amazon EventBridge para Systems Manager.
  - AWS CloudTrail: para obtener más información, consulte Registro de llamadas a la API de AWS Systems Manager con AWS CloudTrail.

¿Qué es un parámetro?

Un parámetro de Parameter Store es cualquier dato guardado en Parameter Store, como un bloque de texto, una lista de nombres, una contraseña, un ID de AMI, una clave de licencia, etc. Puede hacer referencia a estos datos de forma centralizada y segura en sus scripts, comandos y documentos SSM.

Cuando se hace referencia a un parámetro, se debe especificar el nombre del parámetro utilizando la siguiente convención.

{{ssm:*parameter-name*}}

> **Note**
>
> Los parámetros no se pueden referenciar ni anidar en los valores de otros parámetros. No puede incluir {{}} o {{ssm:*parameter-name*}} en el valor de un parámetro.

Parameter Store permite usar tres tipos de parámetros: `String`, `StringList` y `SecureString`.

Con una excepción, cuando cree o actualice un parámetro, deberá ingresar el valor del parámetro como texto sin formato y Parameter Store no realiza ninguna validación en el texto especificado. Sin

embargo, para los parámetros `String` puede especificar el tipo de datos como `aws:ec2:image`, y Parameter Store valida que el valor ingresado tenga el formato adecuado para una AMI de Amazon EC2; por ejemplo: `ami-12345abcdeEXAMPLE`.

Tipo de parámetro: String

De forma predeterminada, el valor de un parámetro `String` consta de cualquier bloque de texto especificado. Por ejemplo:

- `abc123`
- `Example Corp`
- `<img src="images/bannerImage1.png"/>`

Tipo de parámetro: StringList

Los valores de los parámetros `StringList` contienen una lista de valores separados por comas, como se muestra en el siguiente ejemplo.

`Monday,Wednesday,Friday`

`CSV,TSV,CLF,ELF,JSON`

Tipo de parámetro: SecuRestring

El valor de un parámetro `SecureString` es toda información confidencial que debe almacenarse o a la que se hace referencia de forma segura. Si tiene datos que no desea que los usuarios modifiquen o se remita a ellos como texto sin cifrar (por ejemplo, las contraseñas o las claves de licencia), cree esos parámetros utilizando el tipo de datos `SecureString`.

> **Important**
>
> No almacene información confidencial en un parámetro `String` ni `StringList`. Para toda la información confidencial que debe permanecer cifrada, utilice solo el tipo de parámetro `SecureString`.
> Para obtener más información, consulte Utilizar la AWS CLI para crear un parámetro SecureString.

Recomendamos utilizar los parámetros `SecureString` en las siguientes situaciones:

- Desea utilizar los datos/parámetros en los Servicios de AWS sin exponer los valores como texto sin cifrar en comandos, funciones, registros de agentes o registros de CloudTrail.
- Desea controlar quién tiene acceso a la información confidencial.
- Desea tener la posibilidad de auditar los accesos a la información confidencial (CloudTrail).
- Desea disponer de un cifrado para la información confidencial y desea utilizar sus propias claves de cifrado para administrar el acceso.

> **Important**
>
> Solo se cifra el valor de un parámetro `SecureString`. Los nombres de parámetros, las descripciones y otras propiedades no se cifran.

Puede utilizar el tipo de parámetro de `SecureString` para datos textuales que desea cifrar, como contraseñas, secretos de aplicaciones, datos de configuración confidenciales u otros tipos de datos que necesite proteger. Los datos de `SecureString` se cifran y descifran con una clave AWS KMS. Puede utilizar una clave de KMS predeterminada proporcionada por AWS o crear y utilizar su propia AWS KMS key. (Utilice su propia AWS KMS key si desea restringir el acceso de los usuarios a los parámetros de `SecureString`. Para obtener más información, consulte Los permisos de IAM para utilizar claves predeterminadas de AWS y claves administradas por el cliente).

También puede usar parámetros `SecureString` con otros Servicios de AWS. En el siguiente ejemplo, la función de Lambda recupera un parámetro `SecureString` con la API GetParameters.

```
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
```

Cifrado y precios de AWS KMS

Si elige el tipo de parámetro `SecureString` cuando crea el parámetro, Systems Manager utiliza AWS KMS para cifrar el valor del parámetro.

> **Important**
>
> Parameter Store solo es compatible con claves de cifrado de KMS simétricas. No se puede utilizar una clave de cifrado de KMS asimétrica para cifrar los parámetros. Para obtener ayuda para determinar si una clave de KMS es simétrica o asimétrica, consulte Identificación de claves KMS simétricas y asimétricas en la Guía para desarrolladores de AWS Key Management Service.

No se aplican cargos en Parameter Store por crear un parámetro `SecureString`, pero sí por utilizar el cifrado de AWS KMS. Para obtener información, consulte Precios de AWS Key Management Service.

Para obtener más información acerca de las Claves administradas por AWS y de las claves administradas por el cliente, consulte los conceptos de AWS Key Management Service en la Guía para desarrolladores de AWS Key Management Service. Para obtener más información acerca de Parameter Store y cifrado de AWS KMS, consulte Cómo Parameter Store de AWS Systems Manager usa AWS KMS.

> **Note**
>
> Para ver una Clave administrada de AWS, utilice la operación AWS KMS `DescribeKey`. En este ejemplo de la AWS Command Line Interface (AWS CLI) se utiliza `DescribeKey` para ver una Clave administrada de AWS.
>
> ```
> aws kms describe-key --key-id alias/aws/ssm
> ```

Más información

- Cómo crear un parámetro SecureString en Parameter Store y unir un nodo con un dominio (PowerShell)

- Use Parameter Store to Securely Access Secrets and Config Data in CodeDeploy (Utilizar Parameter Store para acceder de manera segura a secretos y datos de configuración en CodeDeploy)
- Interesting Articles on Amazon EC2 Systems Manager Parameter Store (Artículos interesantes en el almacén de parámetros de Amazon EC2 Systems Manager)

Límites de tamaño de parámetros

Parameter Store tiene diferentes límites de tamaño para los valores de los parámetros en función del nivel de parámetro que utilice:

- Parámetros estándar: tamaño máximo del valor de 4 KB
- Parámetros avanzados: tamaño máximo del valor de 8 KB

Si necesita almacenar valores de parámetros mayores de 4 KB, debe utilizar el nivel de parámetros avanzado. Los parámetros avanzados proporcionan capacidades adicionales, pero conllevan cargos en su cuenta de AWS. Para obtener más información acerca de los niveles de parámetros y sus características, consulte Administración de niveles de parámetros.

Para obtener una lista completa de cuotas y límites de Parameter Store, consulte los puntos de conexión y cuotas de AWS Systems Manager en la Referencia general de AWS.

Configuración de Parameter Store

Antes de configurar parámetros en Parameter Store, una herramienta de AWS Systems Manager, debe configurar antes las políticas de AWS Identity and Access Management (IAM) que proporcionan los usuarios de su cuenta con permiso para llevar a cabo las acciones que especifica.

En esta sección se incluye información acerca de cómo configurar manualmente estas políticas mediante la consola de IAM y cómo asignarlas a los usuarios y los grupos de usuarios. También puede crear y asignar políticas para controlar las acciones de parámetro que se pueden ejecutar en un nodo administrado.

En esta sección también se incluye información acerca de cómo crear reglas de Amazon EventBridge que le permitan recibir notificaciones sobre los cambios en los parámetros de Systems Manager. También puede utilizar las reglas de EventBridge para activar otras acciones en AWS en función de los cambios en Parameter Store.

Contenido

- Restringir el acceso a los parámetros de Parameter Store mediante políticas de IAM
- Administración de niveles de parámetros
- Aumentar o disminuir el rendimiento de Parameter Store
- Configuración de notificaciones o activación de acciones en función de los eventos de Parameter Store

Restringir el acceso a los parámetros de Parameter Store mediante políticas de IAM

Restrinja el acceso a parámetros de AWS Systems Manager mediante AWS Identity and Access Management (IAM). En concreto, debe crear políticas de IAM que restrinjan el acceso a las siguientes operaciones de las API:

- DeleteParameter
- DeleteParameters
- DescribeParameters
- GetParameter
- GetParameters
- GetParameterHistory
- GetParametersByPath
- PutParameter

Si utiliza políticas de IAM para restringir el acceso a parámetros de Systems Manager, le recomendamos que cree y utilice políticas de IAM restrictivas. Por ejemplo, la siguiente política permite a un usuario llamar a las operaciones de la API `DescribeParameters` y `GetParameters` para un conjunto limitado de recursos. Esto significa que el usuario puede obtener información sobre ellas y utilizar todos los parámetros que comiencen por `prod-*`.

JSON

```
{
    "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 usuario tiene acceso a una ruta, puede obtener acceso a todos los niveles de esa ruta. Por ejemplo, si un usuario tiene permiso para obtener acceso a la ruta `/a`, el usuario también puede obtener acceso a `/a/b`. Incluso si a un usuario se le ha denegado explícitamente el acceso en IAM al parámetro `/a/b`, aun así puede llamar a la operación de la API `GetParametersByPath` recursivamente para `/a` y ver `/a/b`.

En el caso de los administradores de confianza, puede proporcionar acceso a todas las operaciones de la API de parámetros de Systems Manager mediante el uso de una política similar a la del siguiente ejemplo. Esta política concede al usuario acceso completo a todos los parámetros de producción que empiecen por `dbserver-prod-*`.

JSON

```
{
    "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-2:111122223333:parameter/dbserver-
prod-*"
        },
        {
            "Effect": "Allow",
            "Action": "ssm:DescribeParameters",
            "Resource": "*"
        }
    ]
}
```

Denegar permisos

Cada API es única y tiene operaciones y permisos distintos que puede permitir o denegar individualmente. Una denegación explícita en cualquier política invalida el permiso concedido.

> **Note**
>
> La clave predeterminada AWS Key Management Service (AWS KMS) tiene permiso `Decrypt` para todos los principales de IAM dentro de la Cuenta de AWS. Si desea tener diferentes niveles de acceso a parámetros de `SecureString` en su cuenta, no es recomendable utilizar la clave predeterminada.

Si desea que todas las operaciones de API que recuperen valores de parámetros tengan el mismo comportamiento, puede utilizar un patrón como `GetParameter*` en una política. En el siguiente ejemplo se muestra cómo denegar `GetParameter`, `GetParameters`, `GetParameterHistory`, y `GetParametersByPath`para todos los parámetros que comiencen con `prod-*`.

JSON

```
{
    "Version": "2012-10-17",
```

```
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "ssm:GetParameter*"
            ],
            "Resource": "arn:aws:ssm:us-east-2:123456789012:parameter/prod-*"
        }
    ]
}
```

En el siguiente ejemplo se muestra cómo denegar algunos comandos mientras se permite al usuario realizar otros comandos en todos los parámetros que comienzan con `prod-*`.

JSON

```
{
    "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-2:111122223333:parameter/prod-*"
        }
    ]
}
```

> **Note**
>
> El historial de parámetros incluye todas las versiones de parámetros, incluida la actual. Por lo tanto, si se deniega a un usuario el permiso para `GetParameter`, `GetParameters`, y `GetParameterByPath` pero se le permite el permiso para `GetParameterHistory`, pueden ver el parámetro actual, incluyendo los parámetros `SecureString`, mediante `GetParameterHistory`.

Permitir que se ejecuten solo parámetros específicos en los nodos

Puede controlar el acceso de forma que los nodos administrados solo puedan ejecutar los parámetros que especifique.

Si elige el tipo de datos `SecureString` cuando cree el parámetro, Systems Manager utiliza AWS KMS para cifrar el valor del parámetro. AWS KMS cifra el valor mediante una clave de Clave administrada de AWS o una clave administrada por el cliente. Para obtener más información acerca de AWS KMS y AWS KMS key, consulte Guía para desarrolladores de AWS Key Management Service.

Puede ver la Clave administrada de AWS ejecutando el siguiente comando desde la AWS CLI.

```
aws kms describe-key --key-id alias/aws/ssm
```

El siguiente ejemplo permite que los nodos obtengan un valor de parámetro solo para los parámetros que comienzan con `prod-`. Si el parámetro es un parámetro de `SecureString`, el nodo descifra la cadena mediante AWS KMS.

> **Note**
>
> Las políticas de instancia, como las del ejemplo siguiente, se asignan a el rol de la instancia en IAM. Para obtener más información acerca de cómo configurar el acceso a las características de Systems Manager, incluido el modo de asignar políticas a los usuarios y las instancias, consulte Administrar instancias de EC2 con Systems Manager.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetParameters"
            ],
            "Resource": [
                "arn:aws:ssm:us-east-2:111122223333:parameter/prod-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:us-east-2:111122223333:key/4914ec06-e888-4ea5-
a371-5b88eEXAMPLE"
            ]
        }
    ]
}
```

Los permisos de IAM para utilizar claves predeterminadas de AWS y claves administradas por el cliente

Los parámetros Parameter Store de `SecureString` se cifran y descifran mediante claves AWS KMS. Puede elegir cifrar los parámetros `SecureString` mediante una AWS KMS key o la clave KMS predeterminada proporcionada por AWS.

Cuando se utiliza una clave administrada por el cliente, la política de IAM que concede a un usuario acceso a un parámetro o ruta de parámetro debe proporcionar permisos `kms:Encrypt` explícitos para la clave. Por ejemplo, la siguiente política permite a un usuario crear, actualizar y ver parámetros `SecureString` que comienzan por `prod-` en la Región de AWS y la Cuenta de AWS especificadas.

JSON

```
{
    "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"
            ],
            "Resource": [
                "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-12345EXAMPLE"
            ]
        }
    ]
}
```

[1]El permiso `kms:GenerateDataKey` es necesario para crear parámetros avanzados cifrados utilizando la clave administrada por el cliente especificada.

Por el contrario, todos los usuarios de la cuenta de cliente tienen acceso a la clave administrada de AWS predeterminada. Si utiliza esta clave predeterminada para cifrar parámetros `SecureString` y no desea que los usuarios trabajen con parámetros `SecureString`, sus políticas de IAM deben denegar explícitamente el acceso a la clave predeterminada, como se demuestra en el siguiente ejemplo de política.

> **Note**
>
> Puede localizar el Nombre de recurso de Amazon (ARN) de la clave predeterminada en la consola de AWS KMS de la página de claves administradas de AWS. La clave predeterminada es la que se identifica con `aws/ssm` en la columna Alias.

JSON

```
{
    "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 necesita un control de acceso detallado sobre los parámetros `SecureString` de su cuenta, debe utilizar una clave administrada por el cliente para proteger y restringir el acceso a estos parámetros. También se recomienda utilizar AWS CloudTrail para monitorear las actividades de los parámetros `SecureString`.

Para obtener más información, consulte los temas siguientes:

- Policy evaluation logic (Lógica de evaluación de políticas) en la Guía del usuario de IAM
- Uso de las políticas de claves en AWS KMS en la Guía para desarrolladores de AWS Key Management Service
- Visualización de eventos con el historial de eventos de CloudTrail en la Guía del usuario de AWS CloudTrail

Administración de niveles de parámetros

Parameter Store, una herramienta de AWS Systems Manager, incluye parámetros estándar y parámetros avanzados. Puede configurar individualmente parámetros para utilizar la capa de parámetros estándar (la capa predeterminada) o la capa de parámetros avanzados.

Puede cambiar un parámetro estándar a un parámetro avanzado en cualquier momento, pero no puede revertir un parámetro avanzado a un parámetro estándar. Esto se debe a que revertir un parámetro avanzado a un parámetro estándar haría que el sistema truncara el tamaño del parámetro de 8 KB a 4 KB, lo que provocaría una pérdida de datos. El proceso de reversión también eliminaría las políticas asociadas al parámetro. Además, los parámetros avanzados utilizan otra forma de cifrado diferente a la de los parámetros estándar. Para obtener más información, consulte ¿Cómo AWS Systems ManagerParameter Store utiliza AWS KMS en la Guía para Desarrolladores AWS Key Management Service.

Si ya no necesita un parámetro avanzado, o si ya no desea incurrir en gastos por un parámetro avanzado, elimínelo y vuelva a crearlo como un nuevo parámetro estándar.

En la siguiente tabla se describen las diferencias entre las capas.

	Estándar	Avanzado
Número total de parámetros permitidos (por Cuenta de AWS y Región de AWS)	10 000	100 000
Tamaño máximo del valor del parámetro	4 KB	8 KB
Políticas de parámetros disponibles	No	Sí Para obtener más informaci ón, consulte Asignación de políticas de parámetros en Parameter Store.
Costo	Sin cargo adicional.	Se aplican cargos

	Estándar	Avanzado
		Para obtener más información, consulte Precios de AWS Systems Manager para Parameter Store.

Temas

- Especificación de una capa de parámetros predeterminada
- Cambio de un parámetro estándar a un parámetro avanzado

Especificación de una capa de parámetros predeterminada

En las solicitudes para crear o actualizar un parámetro (es decir, la operación `PutParameter`), puede especificar la capa de parámetros que se va a utilizar en la solicitud. A continuación, se muestra un ejemplo de uso de 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"
```

Siempre que especifique una capa en la solicitud, Parameter Store crea o actualiza el parámetro de acuerdo con su solicitud. Sin embargo, si no especifica explícitamente una capa en una solicitud, la configuración predeterminada de Parameter Store establece en qué capa se crea el parámetro.

La capa predeterminada cuando comienza a utilizar Parameter Store es la capa de parámetros estándar. Si utiliza la capa de parámetros avanzados, puede especificar una de las siguientes opciones como predeterminada:

- Avanzado: con esta opción, Parameter Store evalúa todas las solicitudes como parámetros avanzados.
- Capas inteligentes: con esta opción, Parameter Store evalúa cada solicitud para determinar si el parámetro es estándar o avanzado.

  Si la solicitud no incluye ninguna opción que requiera un parámetro avanzado, el parámetro se crea en la capa de parámetros estándar. Si se incluyen en la solicitud una o varias opciones que requieren un parámetro avanzado, Parameter Store crea un parámetro en la capa de parámetros avanzados.

Beneficios de las capas inteligentes

A continuación se indican los motivos por los que podría elegir las capas inteligentes como capa predeterminada.

Control de costos: las capas inteligentes ayudan a controlar los costos relacionados con los parámetros al crear siempre parámetros estándar a menos que sea absolutamente necesario un parámetro avanzado.

Actualización automática a la capa de parámetros avanzados: cuando realiza un cambio en el código que requiere actualizar un parámetro estándar a un parámetro avanzado, las capas inteligentes se encargan de la conversión. No es necesario cambiar el código para gestionar la actualización.

A continuación, se muestran algunos ejemplos de actualización automática:

- Sus plantillas de AWS CloudFormation aprovisionan numerosos parámetros cuando se ejecutan. Cuando este proceso hace que alcance la cuota de 10 000 parámetros en el nivel de parámetros estándar, los niveles inteligentes actualizan automáticamente al nivel de parámetros avanzados y sus procesos de AWS CloudFormation no se ven interrumpidos.
- Puede almacenar un valor de certificado en un parámetro, rotar el valor del certificado con regularidad y el contenido es inferior a la cuota de 4 KB del nivel de parámetros estándar. Si el valor de un certificado de sustitución supera los 4 KB, las capas inteligentes actualizan automáticamente el parámetro a la capa de parámetros avanzados.

- Desea asociar numerosos parámetros estándar existentes a una política de parámetros, que requiere la capa de parámetros avanzados. En lugar de tener que incluir la opción `--tier Advanced` en todas las llamadas para actualizar los parámetros, las capas inteligentes actualizan automáticamente los parámetros a la capa de parámetros avanzados. La opción de capas inteligentes actualiza los parámetros de estándar a avanzado siempre que se introducen criterios para la capa de parámetros avanzados.

Entre las opciones que requieren un parámetro avanzado se incluyen las siguientes:

- El tamaño del contenido del parámetro es superior a 4 KB.
- El parámetro utiliza una política de parámetros.
- Ya existen más de 10 000 parámetros en su Cuenta de AWS en la Región de AWS actual.

Opciones de capa predeterminadas

Las opciones de capa que puede especificar como valor predeterminado son las siguientes.

- Estándar: la capa de parámetros estándar es la capa predeterminada cuando comienza a utilizar Parameter Store. Con la capa de parámetros estándar, puede crear 10 000 parámetros para cada Región de AWS en una Cuenta de AWS. El tamaño del contenido de cada parámetro puede ser igual a un máximo de 4 KB. Los parámetros estándar no admiten políticas de parámetros. El uso de la capa de parámetros estándar no conlleva ningún cargo adicional. Al elegir Estándar como capa predeterminada, Parameter Store siempre intenta crear un parámetro estándar para las solicitudes que no especifican una capa.
- Avanzado: utilice la capa de parámetros avanzados para crear un máximo de 100 000 parámetros para cada Región de AWS de una Cuenta de AWS. El tamaño del contenido de cada parámetro puede ser igual a un máximo de 8 KB. Los parámetros avanzados admiten políticas de parámetros. Para compartir un parámetro, debe estar en el nivel de parámetros avanzado. El uso de la capa de parámetros avanzados conlleva un cargo. Para obtener más información, consulte Precios de AWS Systems Manager para Parameter Store. Al elegir Advanced (Avanzado) como capa predeterminada, Parameter Store siempre intenta crear un parámetro avanzado para las solicitudes que no especifican una capa.

> **Note**
>
> Si elige la capa de parámetros avanzados, autorice explícitamente a AWS que cargue a su cuenta los parámetros avanzados que cree.

- Capas inteligentes: con la opción de capas inteligente, Parameter Store determina si debe utilizar la capa de parámetros estándar o la capa de parámetros avanzados en función del contenido de la solicitud. Por ejemplo, si ejecuta un comando para crear un parámetro con contenido inferior a 4 KB, hay menos de 10 000 parámetros en la Región de AWS actual de su Cuenta de AWS y no especifica una política de parámetros, se crea un parámetro estándar. Si ejecuta un comando para crear un parámetro con más de 4 KB de contenido, ya tiene más de 10 000 parámetros en la Región de AWS actual de su Cuenta de AWS o especifica una política de parámetros, se crea un parámetro avanzado.

> **Note**
>
> Si elige las capas inteligentes, autorice explícitamente a AWS que cargue a su cuenta los parámetros avanzados que cree.

Puede cambiar la configuración de capa predeterminada de Parameter Store en cualquier momento.

Configuración de permisos para especificar una capa predeterminada de Parameter Store

Compruebe que tiene permiso en AWS Identity and Access Management (IAM) para cambiar la capa de parámetros predeterminada en Parameter Store; para ello, realice una de las siguientes operaciones:

- Asegúrese de adjuntar la política de `AdministratorAccess` a su entidad de IAM (como usuario, grupo o rol).
- Asegúrese de que tiene permiso para cambiar la configuración de capa predeterminada mediante las siguientes acciones de la API:
  - GetServiceSetting
  - UpdateServiceSetting
  - ResetServiceSetting

Otorgue los siguientes permisos a la entidad de IAM a fin de permitir que un usuario vea y cambie la configuración de nivel predeterminada para los parámetros en una Región de AWS específica de una Cuenta de AWS.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "arn:aws:ssm:us-east-1:111122223333:servicesetting/ssm/
parameter-store/default-parameter-tier"
        }
    ]
}
```

Los administradores pueden especificar el permiso de solo lectura mediante la asignación de los siguientes permisos.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"
            ],
```

```
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "*"
        }
    ]
}
```

Para dar acceso, agregue permisos a los usuarios, grupos o roles:

- Usuarios y grupos en AWS IAM Identity Center:

  Cree un conjunto de permisos. Siga las instrucciones de Creación de un conjunto de permisos en la Guía del usuario de AWS IAM Identity Center.

- Usuarios gestionados en IAM a través de un proveedor de identidades:

  Cree un rol para la federación de identidades. Siga las instrucciones descritas en Creación de un rol para un proveedor de identidad de terceros (federación) en la Guía del usuario de IAM.

- Usuarios de IAM:
  - Cree un rol que el usuario pueda aceptar. Siga las instrucciones descritas en Creación de un rol para un usuario de IAM en la Guía del usuario de IAM.
  - (No recomendado) Adjunte una política directamente a un usuario o añada un usuario a un grupo de usuarios. Siga las instrucciones descritas en Adición de permisos a un usuario (consola) de la Guía del usuario de IAM.

Utilizar la consola para especificar o cambiar el nivel predeterminado de Parameter Store

El siguiente procedimiento muestra cómo utilizar la consola de Systems Manager para especificar o cambiar la capa de parámetros predeterminada para la Cuenta de AWS y la Región de AWS actuales.

> **Tip**
>
> Si aún no ha creado un parámetro, puede utilizar la AWS Command Line Interface (AWS CLI) o AWS Tools for Windows PowerShell para cambiar la capa de parámetro predeterminada. Para obtener más información, consulte Especificar o cambiar la capa predeterminada de Parameter Store mediante la AWS CLI y Especificación o cambio de la capa predeterminada de Parameter Store (PowerShell).

Para especificar o cambiar la capa predeterminada de Parameter Store

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Parameter Store.
3. Elija la pestaña Settings.
4. Elija Cambiar la capa predeterminada.
5. Elija una de las siguientes opciones.

   - Estándar
   - Advanced (Avanzado)
   - Intelligent-Tiering

   Para obtener información sobre estas opciones, consulte Especificación de una capa de parámetros predeterminada.
6. Revise el mensaje y seleccione Aceptar.

Si desea cambiar la configuración de capa predeterminada más adelante, repita este procedimiento y especifique una opción de capa predeterminada diferente.

Especificar o cambiar la capa predeterminada de Parameter Store mediante la AWS CLI

El siguiente procedimiento muestra cómo utilizar la AWS CLI para cambiar la configuración predeterminada de la capa de parámetros para la Cuenta de AWS y la Región de AWS actuales.

Para especificar o cambiar la capa predeterminada de Parameter Store mediante la AWS CLI

1. Abra la AWS CLI y ejecute el siguiente comando para cambiar la configuración predeterminada de la capa de parámetros para una Región de AWS específica de una Cuenta de 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
```

   region representa el identificador de Región de AWS compatible con AWS Systems Manager, como `us-east-2` para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

   Los valores de *opción-capa* incluyen `Standard`, `Advanced` y `Intelligent-Tiering`. Para obtener información sobre estas opciones, consulte Especificación de una capa de parámetros predeterminada.

   No se obtienen resultados si el comando se ejecuta correctamente.

2. Ejecute el siguiente comando para ver la configuración actual de nivel de parámetros predeterminados para Parameter Store en la Cuenta de AWS y la Región de AWS actuales.

```
aws ssm get-service-setting --setting-id arn:aws:ssm:region:account-
id:servicesetting/ssm/parameter-store/default-parameter-tier
```

   El sistema devuelve información similar a la siguiente.

```
{
    "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 desea cambiar de nuevo la configuración predeterminada de la capa, repita este procedimiento y especifique una opción de `SettingValue` diferente.

Especificación o cambio de la capa predeterminada de Parameter Store (PowerShell)

El siguiente procedimiento muestra cómo utilizar Tools for Windows PowerShell para cambiar la configuración predeterminada de la capa de parámetros para una Región de AWS específica en una cuenta de Amazon Web Services.

Para especificar o cambiar la capa predeterminada de Parameter Store mediante PowerShell

1. Cambie la capa predeterminada de Parameter Store en la Cuenta de AWS y la Región de AWS actuales mediante Herramientas de AWS para PowerShell (Tools for PowerShell).

   ```
   Update-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/
   ssm/parameter-store/default-parameter-tier" -SettingValue "tier-option" -
   Region region
   ```

   `region` representa el identificador de Región de AWS compatible con AWS Systems Manager, como `us-east-2` para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

   Los valores de *`opción-capa`* incluyen `Standard`, `Advanced` y `Intelligent-Tiering`. Para obtener información sobre estas opciones, consulte Especificación de una capa de parámetros predeterminada.

   No se obtienen resultados si el comando se ejecuta correctamente.

2. Ejecute el siguiente comando para ver la configuración actual de nivel de parámetros predeterminados para Parameter Store en la Cuenta de AWS y la Región de AWS actuales.

   ```
   Get-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/
   parameter-store/default-parameter-tier" -Region region
   ```

   `region` representa el identificador de Región de AWS compatible con AWS Systems Manager, como `us-east-2` para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

El sistema devuelve información similar a la siguiente.

```
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 desea cambiar de nuevo la configuración predeterminada de la capa, repita este procedimiento y especifique una opción de `SettingValue` diferente.

Cambio de un parámetro estándar a un parámetro avanzado

Utilice el siguiente procedimiento para cambiar un parámetro estándar existente a un parámetro avanzado. Para obtener información acerca de cómo crear un nuevo parámetro avanzado, consulte Creación de parámetros de Parameter Store en Systems Manager.

Para cambiar un parámetro estándar a un parámetro avanzado

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Parameter Store.
3. Elija un parámetro y, a continuación, elija Editar.
4. En Descripción, escriba información acerca del parámetro.
5. Seleccione Avanzado.
6. En Valor, introduzca el valor del parámetro. Los parámetros avanzados tienen un valor máximo de 8 KB.
7. Seleccione Save changes (Guardar cambios).

Aumentar o disminuir el rendimiento de Parameter Store

Si aumenta el rendimiento de Parameter Store, se incrementa el número máximo de transacciones por segundo (TPS) que Parameter Store, una herramienta de AWS Systems Manager, puede procesar. Un mayor rendimiento le permite operar con Parameter Store con mayores volúmenes

para admitir aplicaciones y cargas de trabajo que necesitan acceso simultáneo a varios parámetros. Puede aumentar la cuota hasta el rendimiento máximo en la pestaña Settings (Configuración).

La configuración de rendimiento de Parameter Store se aplica a todas las transacciones creadas por todos los usuarios de IAM en las Cuenta de AWS y Región de AWS actuales. El ajuste del rendimiento se aplica a los parámetros estándar y avanzados.

> **Note**
>
> Por lo general, las actualizaciones se muestran inmediatamente en Service Quotas. En casos excepcionales, una actualización puede tardar hasta 24 horas en reflejarse.

Para obtener más información sobre el rendimiento máximo predeterminado y los límites máximos, consulte Cuotas y puntos de conexión de AWS Systems Manager.

Aumentar la cuota de rendimiento genera un cargo en su Cuenta de AWS. Para más información, consulte Precios de AWS Systems Manager.

Temas

- Configuración de los permisos para cambiar el rendimiento de Parameter Store
- Utilizar la consola para aumentar o restablecer el rendimiento
- Utilizar la AWS CLI para aumentar o restablecer el rendimiento
- Aumentar o restablecer el rendimiento (PowerShell)

Configuración de los permisos para cambiar el rendimiento de Parameter Store

Compruebe que tiene permiso en IAM para cambiar el rendimiento de Parameter Store de la siguiente manera:

- Asegúrese de que la política de `AdministratorAccess` esté adjunta a su entidad de IAM (usuario, grupo o rol).
- Asegúrese de que tiene permiso para cambiar el ajuste de servicio de rendimiento mediante las siguientes operaciones de la API:
  - GetServiceSetting
  - UpdateServiceSetting
  - ResetServiceSetting

Otorgue los siguientes permisos a la entidad de IAM a fin de permitir que un usuario vea y cambie la configuración de rendimiento de parámetros para los parámetros en una Región de AWS específica de una Cuenta de AWS.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "arn:aws:ssm:us-east-1:111122223333:servicesetting/ssm/
parameter-store/high-throughput-enabled"
        }
    ]
}
```

Los administradores pueden especificar el permiso de solo lectura mediante la asignación de los siguientes permisos.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"
            ],
```

```
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "*"
        }
    ]
}
```

Para dar acceso, agregue permisos a los usuarios, grupos o roles:

- Usuarios y grupos en AWS IAM Identity Center:

  Cree un conjunto de permisos. Siga las instrucciones de Creación de un conjunto de permisos en la Guía del usuario de AWS IAM Identity Center.
- Usuarios gestionados en IAM a través de un proveedor de identidades:

  Cree un rol para la federación de identidades. Siga las instrucciones descritas en Creación de un rol para un proveedor de identidad de terceros (federación) en la Guía del usuario de IAM.
- Usuarios de IAM:
  - Cree un rol que el usuario pueda aceptar. Siga las instrucciones descritas en Creación de un rol para un usuario de IAM en la Guía del usuario de IAM.
  - (No recomendado) Adjunte una política directamente a un usuario o añada un usuario a un grupo de usuarios. Siga las instrucciones descritas en Adición de permisos a un usuario (consola) de la Guía del usuario de IAM.

Utilizar la consola para aumentar o restablecer el rendimiento

El siguiente procedimiento muestra cómo utilizar la consola de Systems Manager para aumentar el número de transacciones por segundo que Cuenta de AWS puede procesar para el Parameter Store y la Región de AWS actuales. También muestra cómo restablecer la configuración estándar si ya no necesita aumentar el rendimiento o ya no quiere incurrir en gastos.

Aumentar o restablecer el Parameter Store rendimiento con la consola

> **Tip**
>
> Si aún no ha creado un parámetro, puede utilizar la AWS Command Line Interface (AWS CLI) o AWS Tools for Windows PowerShell para aumentar el rendimiento. Para obtener más información, consulte Utilizar la AWS CLI para aumentar o restablecer el rendimiento y Aumentar o restablecer el rendimiento (PowerShell).

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Parameter Store.
3. Elija la pestaña Settings.
4. Para aumentar el rendimiento, seleccione Establecer límite.

   -o bien-

   Para restablecer el límite predeterminado, seleccione Reestablecer límite.
5. Si va a aumentar el límite, haga lo siguiente:

   - Seleccione la casilla de verificación Acepto que el cambio de esta configuración suponga cargos en mi Cuenta de AWS.
   - Elija Set limit (Establecer límite).

   -o bien-

   Si va a restablecer el límite a los valores predeterminados, haga lo siguiente:

   - Active la casilla de verificación Acepto que al restablecer el límite de rendimiento al valor predeterminado Parameter Store procesará menos transacciones por segundo.
   - Seleccione Establecer límite.

Utilizar la AWS CLI para aumentar o restablecer el rendimiento

En el siguiente procedimiento se muestra cómo utilizar AWS CLI para aumentar el número de transacciones por segundo que Parameter Store puede procesar para la Cuenta de AWS y la Región de AWS actuales. También puede restablecer al límite predeterminado.

Para aumentar el rendimiento de Parameter Store mediante la AWS CLI

1. Abra la AWS CLI y ejecute el siguiente comando para aumentar las transacciones por segundo que Parameter Store puede procesar en la Cuenta de AWS y la Región de AWS actuales.

   ```
   aws ssm update-service-setting --setting-id arn:aws:ssm:region:account-
   id:servicesetting/ssm/parameter-store/high-throughput-enabled --setting-value true
   ```

   No se obtienen resultados si el comando se ejecuta correctamente.

2. Ejecute el siguiente comando para ver la configuración del servicio de rendimiento actual para Parameter Store en la Cuenta de AWS y la Región de AWS actuales.

   ```
   aws ssm get-service-setting --setting-id arn:aws:ssm:region:account-
   id:servicesetting/ssm/parameter-store/high-throughput-enabled
   ```

   El sistema devuelve información similar a la siguiente:

   ```
   {
       "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 ya no necesita una mejora del rendimiento o si ya no desea incurrir en ningún gasto adicional, puede volver a la configuración estándar. Para revertir la configuración, ejecute el siguiente comando.

```
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"
    }
}
```

Aumentar o restablecer el rendimiento (PowerShell)

En el siguiente procedimiento se muestra cómo utilizar Tools for Windows PowerShell para aumentar el número de transacciones por segundo que Cuenta de AWS puede procesar para el Parameter Store y la Región de AWS actuales. También puede restablecer al límite predeterminado.

Para aumentar el rendimiento de Parameter Store con PowerShell

1. Aumente el rendimiento de Parameter Store en la Cuenta de AWS y la Región de AWS actuales mediante Herramientas de AWS para PowerShell (Tools for PowerShell).

   ```
   Update-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/
   ssm/parameter-store/high-throughput-enabled" -SettingValue "true" -Region region
   ```

   No se obtienen resultados si el comando se ejecuta correctamente.

2. Ejecute el siguiente comando para ver la configuración del servicio de rendimiento actual para Parameter Store en la Cuenta de AWS y la Región de AWS actuales.

   ```
   Get-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/
   parameter-store/high-throughput-enabled" -Region region
   ```

   Los sistemas devuelven información similar a la siguiente:

```
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 ya no necesita una mejora del rendimiento o si ya no desea incurrir en ningún gasto adicional, puede volver a la configuración estándar. Para revertir la configuración, ejecute el siguiente comando.

```
Reset-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/
parameter-store/high-throughput-enabled" -Region region
```

El sistema devuelve información similar a la siguiente:

```
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
```

Configuración de notificaciones o activación de acciones en función de los eventos de Parameter Store

En los temas de esta sección se explica cómo utilizar Amazon EventBridge y Amazon Simple Notification Service (Amazon SNS) para recibir notificaciones sobre los cambios en parámetros de AWS Systems Manager. Puede crear una regla de EventBridge que le informe sobre el momento en el que se crea, actualiza o elimina un parámetro o una versión de etiqueta de parámetro. Los eventos se emiten en la medida de lo posible. Puede recibir información sobre los cambios o el estado con relación a las políticas de parámetros, como, por ejemplo, cuándo vence un parámetro, cuándo va a vencer o cuándo no ha cambiado durante un periodo de tiempo especificado.

> **Note**
>
> Las políticas de parámetros están disponibles para los parámetros que utilizan la capa de parámetros avanzados. Se aplican cargos. Para obtener más información, consulte Asignación de políticas de parámetros en Parameter Store y Administración de niveles de parámetros.

En los temas de esta sección también se explica cómo activar otras acciones en un destino para eventos de parámetro específicos. Por ejemplo, puede ejecutar una función de AWS Lambda para volver a crear un parámetro automáticamente al vencer o eliminarse este. Puede configurar una notificación que active una función de Lambda cuando se actualiza la contraseña de la base de datos. La función de Lambda puede forzar que se restablezcan las conexiones de la base de datos o que vuelvan a conectarse con la contraseña nueva. EventBridge también permite ejecutar comandos de Run Command y ejecuciones de automatización, y acciones en muchos otros Servicios de AWS. Run Command y Automatización son herramientas de AWS Systems Manager. Para más información, consulte la Guía del usuario de Amazon EventBridge.

Antes de empezar

Cree los recursos que necesite para especificar la acción de destino para la regla que cree. Por ejemplo, si la regla que crea es para enviar una notificación, primero debe crear un tema de Amazon SNS. Para obtener más información, consulte Introducción a Amazon SNS en la Guía para desarrolladores de Amazon Simple Notification Service.

Configuración de reglas de EventBridge para parámetros y políticas de parámetros

En este tema se explica lo siguiente:

- Cómo crear una regla de EventBridge que invoque un destino en función de los eventos les ocurran a uno o varios parámetros de la Cuenta de AWS.
- Cómo crear reglas de EventBridge que invoquen destinos en función de los eventos que les ocurran a una o varias políticas de parámetros de la Cuenta de AWS. Cuando cree un parámetro avanzado, especifique cuándo vence un parámetro, cuándo desea recibir una notificación antes del vencimiento de un parámetro y cuánto tiempo hay que esperar antes de que deba enviarse una notificación de que no ha cambiado un parámetro. Puede configurar la notificación para estos eventos mediante el siguiente procedimiento. Para obtener más información, consulte Asignación de políticas de parámetros en Parameter Store y Administración de niveles de parámetros.

Para configurar una regla de EventBridge para un parámetro o una política de parámetros de Systems Manager

1. Abra la consola de Amazon EventBridge en https://console.aws.amazon.com/events/.
2. En el panel de navegación, elija Rules (Reglas) y, a continuación, elija Create rule (Crear regla).

   -o bien-

   Si la página de inicio de EventBridge se abre primero, elija Create rule (Crear regla).
3. Escriba un nombre y una descripción para la regla.

   Una regla no puede tener el mismo nombre que otra regla de la misma región y del mismo bus de eventos.
4. En Event bus (Bus de eventos), elija el bus de eventos que desee asociar a esta regla. Si desea que esta regla se inicie con eventos coincidentes procedentes de su propia Cuenta de AWS, seleccione default (predeterminado). Cuando un Servicio de AWS en su cuenta emite un evento, siempre va al bus de eventos predeterminado de su cuenta.
5. En Rule type (Tipo de regla), deje seleccionada la opción predeterminada Rule with an event pattern (Regla con un patrón de evento).
6. Elija Siguiente.
7. En Origen del evento, deje seleccionada la opción predeterminada Eventos de AWS o eventos de socios de EventBridge. Puede omitir la sección Sample event (Ejemplo de evento).
8. En Event pattern (Patrón de evento), realice una de las siguientes acciones:

   - Elija Custom patterns (JSON editor) (Patrones personalizados [editor de JSON]).
   - En Patrón de evento, pegue uno de los siguientes contenidos en el cuadro, en función de si va a crear una regla para un parámetro o una política de parámetros:

     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"
        ]
    }
}
```

- Modifique el contenido de los parámetros y las operaciones en los que desee actuar, tal y como se muestra en los siguientes ejemplos.

  Parameter

  En este ejemplo, se realizará una acción cuando se actualice cualquiera de los parámetros denominados `/Oncall` y `/Project/Teamlead`:

```
{
    "source": [
        "aws.ssm"
    ],
    "detail-type": [
        "Parameter Store Change"
    ],
    "detail": {
        "name": [
            "/Oncall",
            "/Project/Teamlead"
        ],
        "operation": [
            "Update"
        ]
    }
}
```

Parameter policy

En este ejemplo, se realizará una acción siempre que el parámetro denominado /`OncallDuties` caduque y se elimine:

```
{
    "source": [
        "aws.ssm"
    ],
    "detail-type": [
        "Parameter Store Policy Action"
    ],
    "detail": {
        "parameter-name": [
            "/OncallDuties"
        ],
        "policy-type": [
            "Expiration"
        ]
    }
}
```

9. Elija Siguiente.

10. En Target 1 (Destino 1), elija un tipo de destino y un recurso compatible. Por ejemplo, si elige SNS topic (Tema de SNS), seleccione Topic (Tema). Si elige CodePipeline, introduzca un ARN de canalización en Pipeline ARN (ARN de canalización). Proporcione los valores de configuración adicionales que sean necesarios.

    > **Tip**
    >
    > Elija Add another target (Agregar otro destino) si necesita destinos adicionales para la regla.

11. Elija Siguiente.
12. (Opcional) Introduzca una o varias etiquetas para la regla. Para obtener más información, consulte Etiquetas de Amazon EventBridge en la Guía del usuario de Amazon EventBridge.
13. Elija Siguiente.
14. Seleccione Crear regla.

Más información

- Uso de etiquetas de parámetros para una actualización de configuración sencilla en entornos
- Tutorial: Use EventBridge para transmitir eventos a AWS Systems Manager Run Command en la Guía del usuario de Amazon EventBridge
- Tutorial: configure AWS Systems Manager Automation como destino de EventBridge en la Guía del usuario de Amazon EventBridge

Uso de Parameter Store

En esta sección se describe cómo organizar, crear y etiquetar parámetros, además de cómo crear distintas versiones de ellos.

Puede utilizar la consola de AWS Systems Manager, la AWS Command Line Interface (AWS CLI), Herramientas de AWS para PowerShell y los SDK de AWS para crear parámetros y utilizarlos. Para obtener más información acerca de los parámetros, consulte ¿Qué es un parámetro?.

Temas

- Creación de parámetros de Parameter Store en Systems Manager
- Búsqueda de parámetros de Parameter Store en Systems Manager

- Asignación de políticas de parámetros en Parameter Store
- Trabajo con jerarquías de parámetros en Parameter Store
- Impedir el acceso a las operaciones de la API de Parameter Store
- Trabajo con etiquetas de parámetros en Parameter Store
- Trabajo con versiones de parámetros en Parameter Store
- Trabajo con parámetros compartidos en Parameter Store
- Trabajo con parámetros en Parameter Store con el uso de comandos de Run Command
- Uso de compatibilidad con parámetros nativos en Parameter Store para los ID de Amazon Machine Image
- Eliminación de parámetros de Parameter Store

Creación de parámetros de Parameter Store en Systems Manager

Utilice la información de los siguientes temas para ayudarlo a crear parámetros de Systems Manager con la consola de AWS Systems Manager, la AWS Command Line Interface (AWS CLI), o AWS Tools for Windows PowerShell(Tools for Windows PowerShell).

Esta sección muestra cómo crear, almacenar y ejecutar parámetros con Parameter Store en un entorno de pruebas. También muestra cómo utilizar Parameter Store con otras herramientas de Systems Manager en Servicios de AWS. Para obtener más información, consulte ¿Qué es un parámetro?

Conozca los requisitos y restricciones de los nombres de los parámetros

Utilice la información de este tema como ayuda para especificar los valores válidos de los nombres de los parámetros al crear un parámetro.

Esta información complementa los detalles del tema PutParameter de la Referencia de la API de AWS Systems Manager, que también proporciona información sobre los valores de AllowedPattern, Description, KeyId, Overwrite, Type y Value.

Los nombres de los parámetros deben cumplir los requisitos y restricciones siguientes:

- Diferenciación entre mayúsculas y minúsculas: los nombres de los parámetros distinguen entre mayúsculas y minúsculas.
- Espacios: los nombres de los parámetros no pueden incluir espacios.

- Caracteres válidos: los nombres de los parámetros solo pueden incluir los siguientes símbolos y letras: `a-zA-Z0-9_.-`

  Además, el carácter de barra inclinada (/) se utiliza para delinear jerarquías en los nombres de parámetros. Por ejemplo: `/Dev/Production/East/Project-ABC/MyParameter`
- Formato de AMI válido: cuando elija `aws:ec2:image` como tipo de datos para un parámetro `String`, el ID que especifica debe validarse para el formato de ID de AMI `ami-12345abcdeEXAMPLE`.
- Totalmente cualificado: cuando crea o hace referencia a un parámetro en una jerarquía, debe incluir un carácter de barra diagonal (/). Cuando hace referencia a un parámetro que forma parte de una jerarquía, especifique toda la ruta de la jerarquía, incluida la barra diagonal inicial (/).
  - Nombres de parámetros completos: `MyParameter1`, `/MyParameter2`, `/Dev/Production/East/Project-ABC/MyParameter`
  - Nombre de parámetro no completo: `MyParameter3/L1`
- Longitud: la longitud máxima del nombre de parámetro que especifique es de 1011 caracteres. Esto incluye los caracteres del ARN que preceden al nombre que especifique, como los 45 caracteres en `arn:aws:ssm:us-east-2:111122223333:parameter/`.
- Prefijos: un nombre de parámetro no puede tener como prefijo “`aws`” ni “`ssm`” (no se distingue entre mayúsculas y minúsculas). Por ejemplo, se produce un error con una excepción si se intenta crear parámetros con los siguientes nombres:
  - `awsTestParameter`
  - `SSM-testparameter`
  - `/aws/testparam1`

  > **Note**
  >
  > Cuando se especifica un parámetro en un documento, un comando o un script de SSM, se debe incluir `ssm` como parte de la sintaxis. Por ejemplo, {{ssm:*parameter_name*}} y {{ ssm:*parameter_name* }}, como `{{ssm:MyParameter}}`, y `{{ ssm:MyParameter }}`.
- Unicidad: los nombres de los parámetros deben ser únicos dentro de una Región de AWS. Por ejemplo, Systems Manager trata los siguientes parámetros como parámetros diferentes si existen en la misma región:
  - `/Test/TestParam1`

- `/TestParam1`

  Los siguientes ejemplos también son únicos:

  - `/Test/TestParam1/Logpath1`
  - `/Test/TestParam1`

  Sin embargo, los siguientes ejemplos, si están en la misma región, no son únicos:

  - `/TestParam1`
  - `TestParam1`

- Profundidad de la jerarquía: si especifica una jerarquía de parámetros, esta puede tener una profundidad máxima de quince niveles. Puede definir un parámetro en cualquier nivel de la jerarquía. Los dos ejemplos siguientes son válidos estructuralmente:

  - `/Level-1/L2/L3/L4/L5/L6/L7/L8/L9/L10/L11/L12/L13/L14/parameter-name`
  - `parameter-name`

  El intento de crear los siguientes parámetros tendría un error con la excepción `HierarchyLevelLimitExceededException`:

  - `/Level-1/L2/L3/L4/L5/L6/L7/L8/L9/L10/L11/L12/L13/L14/L15/L16/parameter-name`

> **Important**
>
> Si un usuario tiene acceso a una ruta, puede obtener acceso a todos los niveles de esa ruta. Por ejemplo, si un usuario tiene permiso para obtener acceso a la ruta `/a`, el usuario también puede obtener acceso a `/a/b`. Incluso si a un usuario se le ha denegado explícitamente el acceso en AWS Identity and Access Management (IAM) al parámetro `/a/b`, aun así puede llamar a la operación de la API GetParametersByPath recursivamente para `/a` y ver `/a/b`.

Temas

- Utilizar la consola para crear un parámetro Parameter Store
- Utilizar la AWS CLI para crear un parámetro de Parameter Store
- Utilizar las Herramientas para Windows PowerShell para crear un parámetro Parameter Store

Utilizar la consola para crear un parámetro Parameter Store

Puede utilizar la consola de AWS Systems Manager para crear y ejecutar `String`, `StringList`, y tipos de parámetros de `SecureString`. Después de eliminar un parámetro, espere al menos 30 segundos para crear un parámetro con el mismo nombre.

> **Note**
>
> Los parámetros solo están disponibles en la Región de AWS donde se crearon.

El siguiente procedimiento presenta el proceso de creación de un parámetro en la consola de Parameter Store. Puede crear `String`, `StringList` y tipos de parámetro de `SecureString` desde la consola.

Para crear un parámetro

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Parameter Store.
3. Elija Create parameter.
4. En el cuadro Name (Nombre), escriba una jerarquía y un nombre. Por ejemplo, escriba **`/Test/helloWorld`**.

   Para obtener más información acerca de las jerarquías de parámetros, consulte Trabajo con jerarquías de parámetros en Parameter Store.
5. En el cuadro Description (Descripción), escriba una descripción que identifique este parámetro como un parámetro de prueba.
6. En Parameter (Parámetro) elija Standard (Estándar) o Advanced (Avanzado). Para obtener más información sobre parámetros avanzados, consulte Administración de niveles de parámetros.
7. En Type, seleccione String, StringList o SecureString.

   - Si elige String (Cadena), aparecerá el campo Data type (Tipo de datos). Si va a crear un parámetro que contenga el ID de recurso de una Amazon Machine Image (AMI), seleccione `aws:ec2:image`. De lo contrario, mantenga el valor predeterminado `text` seleccionado.
   - Si elige SecureString, aparece el campo KMS Key ID (ID de clave de KMS). Si no proporciona un ID de AWS Key Management Service AWS KMS key, un AWS KMS key de nombre de

recurso de Amazon (ARN), un nombre de alias o un ARN de alias, el sistema utiliza `alias/aws/ssm`, que es la Clave administrada de AWS para Systems Manager. Si no desea utilizar esta clave, puede utilizar una clave administrada por el cliente. Para obtener más información acerca de las Claves administradas por AWS y de las claves administradas por el cliente, consulte los conceptos de AWS Key Management Service en la Guía para desarrolladores de AWS Key Management Service. Para obtener más información acerca de Parameter Store y cifrado de AWS KMS, consulte Cómo Parameter Store de AWS Systems Manager usa AWS KMS.

> **Important**
>
> Parameter Store solo es compatible con claves de cifrado de KMS simétricas. No se puede utilizar una clave de cifrado de KMS asimétrica para cifrar los parámetros. Para obtener ayuda para determinar si una clave de KMS es simétrica o asimétrica, consulte Identificación de claves KMS simétricas y asimétricas en la Guía para desarrolladores de AWS Key Management Service.

- Cuando crea un parámetro `SecureString` en la consola mediante el parámetro `key-id` con un nombre de alias de clave administrado por el cliente o un ARN de alias, especifique el prefijo `alias/` delante del alias. A continuación, se muestra un ejemplo de ARN.

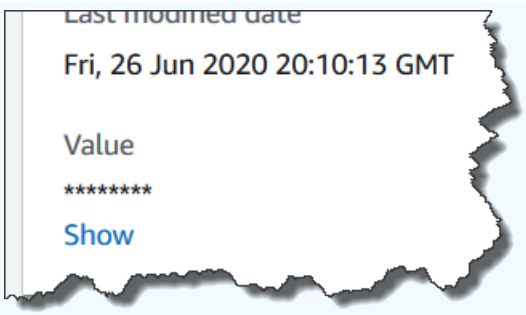
```
arn:aws:kms:us-east-2:123456789012:alias/abcd1234-ab12-cd34-ef56-abcdeEXAMPLE
```

A continuación, se muestra un ejemplo de un nombre de alias.

```
alias/MyAliasName
```

8. En el cuadro Value, escriba un valor. Por ejemplo, escriba **`This is my first parameter`** o **`ami-0dbf5ea29aEXAMPLE`**.

> **Note**
>
> Los parámetros no se pueden referenciar ni anidar en los valores de otros parámetros. No puede incluir `{{}}` o `{{ssm:parameter-name}}` en el valor de un parámetro. Si elije SecureString, el valor del parámetro se enmascara de forma predeterminada (“******”) cuando lo vea más adelante en la pestaña Información general del parámetro, como se muestra en la ilustración siguiente. Elija Show (Mostrar) para mostrar el valor del parámetro.



9. (Opcional) En el área Tags (Etiquetas), aplique al parámetro uno o varios pares de clave-valor de etiqueta.

   Las etiquetas son metadatos opcionales que usted asigna a un recurso. Las etiquetas permiten clasificar los recursos de diversas maneras, por ejemplo, según la finalidad, el propietario o el entorno. Por ejemplo, es posible que desee etiquetar un parámetro de Systems Manager para identificar el tipo de recurso al que se aplica, el entorno o el tipo de datos de configuración al que se hace referencia con el parámetro. En este caso, puede especificar los siguientes pares de clave-valor:

   - `Key=Resource,Value=S3bucket`
   - `Key=OS,Value=Windows`
   - `Key=ParameterType,Value=LicenseKey`

10. Elija Create parameter.
11. En la lista de parámetros, elija el nombre del parámetro que acaba de crear. Verifique los detalles en la pestaña Información general. Si ha creado un parámetro `SecureString`, elija Show para ver el valor sin cifrar.

> **Note**
>
> No se puede cambiar un parámetro avanzado a un parámetro estándar. Si ya no necesita un parámetro avanzado, o si ya no desea incurrir en gastos por un parámetro avanzado, elimínelo y vuelva a crearlo como un nuevo parámetro estándar.

Utilizar la AWS CLI para crear un parámetro de Parameter Store

Puede utilizar la AWS Command Line Interface (AWS CLI) para crear tipos de parámetros `String`, `StringList`, y `SecureString`. Después de eliminar un parámetro, espere al menos 30 segundos para crear un parámetro con el mismo nombre.

Los parámetros no se pueden referenciar ni anidar en los valores de otros parámetros. No puede incluir `{{}}` o `{{ssm:parameter-name}}` en el valor de un parámetro.

> **Note**
>
> Los parámetros solo están disponibles en la Región de AWS donde se crearon.

Temas

- Utilizar la AWS CLI para crear un parámetro de String
- Utilizar la AWS CLI para crear un parámetro StringList
- Utilizar la AWS CLI para crear un parámetro SecureString
- Utilizar la AWS CLI para crear un parámetro de varias líneas

Utilizar la AWS CLI para crear un parámetro de **`String`**

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Para crear un parámetro de tipo `String`, ejecute el siguiente comando. Reemplace cada *`example resource placeholder`* con su propia información.

   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"
```

-o bien-

Ejecute el siguiente comando para crear un parámetro que contenga un ID de Amazon Machine Image (AMI) como valor del parámetro.

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"
```

La opción `--name` admite jerarquías. Para obtener información acerca de las jerarquías, consulte Trabajo con jerarquías de parámetros en Parameter Store.

La opción `--data-type` solo se debe especificar si va a crear un parámetro que contiene un ID de AMI. Este valida que el valor del parámetro que ingresa es un ID de AMI de Amazon Elastic Compute Cloud (Amazon EC2) con formato adecuado. Para todos los demás parámetros, el tipo de datos predeterminado es `text` y es opcional especificar un valor. Para obtener más

información, consulte Uso de compatibilidad con parámetros nativos en Parameter Store para los ID de Amazon Machine Image.

> **Important**
>
> Si se ejecuta correctamente, el comando devuelve el número de la versión del parámetro. Excepción: si ha especificado `aws:ec2:image` como tipo de datos, un nuevo número de versión en la respuesta no significa que el valor del parámetro se haya validado todavía. Para obtener más información, consulte Uso de compatibilidad con parámetros nativos en Parameter Store para los ID de Amazon Machine Image.

En este ejemplo se agregan dos etiquetas de pares de clave-valor a un parámetro.

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\"}]
```

En el siguiente ejemplo se utiliza una jerarquía de parámetros en el nombre para crear un parámetro de `String` de texto sin formato. Este devuelve el número de la versión del parámetro. Para obtener más información acerca de las jerarquías de parámetros, consulte Trabajo con jerarquías de parámetros en Parameter Store.

Linux & macOS

Parámetro no en una jerarquía

```
aws ssm put-parameter \
    --name "golden-ami" \
    --type "String" \
    --value "ami-12345abcdeEXAMPLE"
```

Parámetro en una jerarquía

```
aws ssm put-parameter \
    --name "/amis/linux/golden-ami" \
    --type "String" \
    --value "ami-12345abcdeEXAMPLE"
```

Windows

Parámetro no en una jerarquía

```
aws ssm put-parameter ^
    --name "golden-ami" ^
    --type "String" ^
    --value "ami-12345abcdeEXAMPLE"
```

Parámetro en una jerarquía

```
aws ssm put-parameter ^
    --name "/amis/windows/golden-ami" ^
    --type "String" ^
    --value "ami-12345abcdeEXAMPLE"
```

3. Ejecute el siguiente comando para ver el valor más reciente del parámetro y comprobar los detalles del nuevo parámetro.

```
aws ssm get-parameters --names "/Test/IAD/helloWorld"
```

El sistema devuelve información similar a la siguiente.

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

Ejecute el siguiente comando para cambiar los valores del parámetro. Este devuelve el número de la versión del parámetro.

```
aws ssm put-parameter --name "/Test/IAD/helloWorld" --value "My updated 1st parameter"
 --type String --overwrite
```

Ejecute el siguiente comando para ver el historial del parámetro.

```
aws ssm get-parameter-history --name "/Test/IAD/helloWorld"
```

Ejecute el siguiente comando para utilizar este parámetro en un comando.

```
aws ssm send-command --document-name "AWS-RunShellScript" --parameters '{"commands":
["echo {{ssm:/Test/IAD/helloWorld}}"]}' --targets "Key=instanceids,Values=instance-ids"
```

Ejecute el siguiente comando si solo desea recuperar el valor del parámetro.

```
aws ssm get-parameter --name testDataTypeParameter --query "Parameter.Value"
```

Ejecute el siguiente comando si solo desea recuperar el valor del parámetro mediante `get-parameters`.

```
aws ssm get-parameters --names "testDataTypeParameter" --query "Parameters[*].Value"
```

Ejecute el siguiente comando para ver los metadatos del parámetro.

```
aws ssm describe-parameters --filters "Key=Name,Values=/Test/IAD/helloWorld"
```

> **Note**
>
> El nombre debe estar en mayúsculas.

El sistema devuelve información similar a la siguiente.

```
{
    "Parameters": [
        {
            "Name": "helloworld",
            "Type": "String",
            "LastModifiedUser": "arn:aws:iam::123456789012:user/JohnDoe",
            "LastModifiedDate": 1494529763.156,
            "Version": 1,
            "Tier": "Standard",
            "Policies": []
        }
    ]
}
```

Utilizar la AWS CLI para crear un parámetro StringList

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Para crear un parámetro, ejecute el siguiente comando. Reemplace cada *example resource placeholder* con su propia información.

   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**
>
> Si se ejecuta correctamente, el comando devuelve el número de la versión del parámetro.

En este ejemplo se añaden dos etiquetas de pares clave-valor a un parámetro. (En función del tipo de sistema operativo de su máquina local, ejecute uno de los siguientes comandos. La versión para ejecutar desde un equipo local de Windows incluye los caracteres de escape [“\”] que necesita para ejecutar el comando desde la herramienta de línea de comandos).

Aquí se incluye un ejemplo de `StringList` que utiliza una jerarquía de parámetros.

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**
>
> Los elementos de `StringList` deben ir separados por comas (,). No se puede utilizar otra puntuación ni carácter especial para exceptuar los elementos de la lista. Si tiene un valor de parámetro que requiera una coma, utilice el tipo de datos `String`.

3. Ejecute el comando `get-parameters` para verificar los detalles del parámetro. Por ejemplo:

```
aws ssm get-parameters --name "/IAD/ERP/Oracle/addUsers"
```

Utilizar la AWS CLI para crear un parámetro SecureString

Utilice el siguiente procedimiento para crear un parámetro `SecureString`. Reemplace cada *`example resource placeholder`* con su propia información.

> **Important**
>
> Solo se cifra el valor de un parámetro `SecureString`. Los nombres de parámetros, las descripciones y otras propiedades no se cifran.

> **Important**
>
> Parameter Store solo es compatible con claves de cifrado de KMS simétricas. No se puede utilizar una clave de cifrado de KMS asimétrica para cifrar los parámetros. Para obtener ayuda para determinar si una clave de KMS es simétrica o asimétrica, consulte Identificación de claves KMS simétricas y asimétricas en la Guía para desarrolladores de AWS Key Management Service.

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.
2. Ejecute uno de los siguientes comandos para crear un parámetro que utilice el tipo de datos `SecureString`.

Linux & macOS

Creación de un parámetro **SecureString** con la Clave administrada de AWS predeterminada

```
aws ssm put-parameter \
    --name "parameter-name" \
    --value "parameter-value" \
    --type "SecureString"
```

Creación de un parámetro **SecureString** que utilice una clave administrada por el cliente

```
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"
```

Creación de un **SecureString** parámetro que use una AWS KMS clave personalizada

```
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

Creación de un parámetro **SecureString** con la Clave administrada de AWS predeterminada

```
aws ssm put-parameter ^
    --name "parameter-name" ^
    --value "parameter-value" ^
    --type "SecureString"
```

Creación de un parámetro **SecureString** que utilice una clave administrada por el cliente

```
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"
```

Creación de un **`SecureString`** parámetro que use una AWS KMS clave personalizada

```
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 crea un parámetro `SecureString` usando la clave de Clave administrada de AWS en su cuenta y región, no tendrá que proporcionar un valor para el parámetro `--key-id`.

> **Note**
>
> Para utilizar la AWS KMS key asignada a su Cuenta de AWS y Región de AWS, quite el parámetro `key-id` del comando. Para obtener más información acerca de AWS KMS keys, consulte Conceptos de AWS Key Management Service en la Guía para desarrolladores de AWS Key Management Service.

Para utilizar una clave administrada por el cliente en lugar de la Clave administrada de AWS asignada a su cuenta, especifique la clave mediante el parámetro `--key-id`. El parámetro es compatible con los siguientes formatos de parámetros de KMS.

- Ejemplo de clave de nombre de recurso de Amazon (ARN):

  `arn:aws:kms:us-east-2:123456789012:key/key-id`

- Ejemplo de ARN de alias:

  `arn:aws:kms:us-east-2:123456789012:alias/alias-name`

- Ejemplo de ID de clave:

```
12345678-1234-1234-1234-123456789012
```

- Ejemplo de nombre de alias:

```
alias/MyAliasName
```

Puede crear una clave administrada por el cliente a través de la AWS Management Console o la API de AWS KMS. Los siguientes comandos de la AWS CLI crean una clave administrada por el cliente en la Región de AWS actual de su Cuenta de AWS.

```
aws kms create-key
```

Utilice un comando con el siguiente formato para crear un parámetro `SecureString` que use la clave que acaba de crear.

En el siguiente ejemplo se utiliza un nombre encubierto (`3l3vat3131`) para un parámetro de contraseña y una 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. Ejecute el siguiente comando para verificar los detalles del parámetro.

Si no especifica el parámetro `with-decryption` o si especifica el parámetro `no-with-decryption`, el comando devuelve un GUID cifrado.

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. Ejecute el siguiente comando para ver los metadatos del parámetro.

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. Ejecute el siguiente comando para cambiar el valor del parámetro si no está utilizando una AWS KMS key administrada por el cliente.

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
```

-o bien-

Ejecute el siguiente comando para cambiar el valor del parámetro si está utilizando una AWS KMS key administrada por el cliente.

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. Ejecute el siguiente comando para ver el valor más reciente del parámetro.

   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. Ejecute el siguiente comando para ver el historial del parámetro.

   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**
>
> Puede crear manualmente un parámetro con un valor cifrado. En ese caso, puesto que el valor ya está cifrado, no es necesario que elija el tipo de parámetro `SecureString`. Si elige `SecureString`, el parámetro se cifrará dos veces.

De forma predeterminada, todos `SecureString` los valores se muestran como texto cifrado. Para descifrar un valor `SecureString`, un usuario debe tener permisos para llamar a la operación Decrypt de la API de AWS KMS. Para obtener más información acerca de cómo configurar el control de acceso de AWS KMS, consulte Autenticación y control de acceso de AWS KMS en la Guía para desarrolladores de AWS Key Management Service.

> **Important**
>
> Si cambia el alias de clave de KMS para la clave KMS que se utiliza para cifrar un parámetro, también actualice el alias de clave que utiliza el parámetro para hacer referencia a AWS KMS. Esto solo se aplica al alias de clave de KMS; el ID de clave al que se adjunta un alias permanece igual a menos que elimine toda la clave.

Utilizar la AWS CLI para crear un parámetro de varias líneas

Puede utilizar la AWS CLI para crear un parámetro con saltos de línea. Utilice saltos de línea para dividir el texto en valores de parámetros más largos para una mejor legibilidad o, por ejemplo, para actualizar el contenido de parámetros de varios párrafos para una página web. Puede incluir el contenido en un archivo JSON y utilizar la opción `--cli-input-json`, mediante caracteres de salto de línea como `\n`, como se muestra en el ejemplo siguiente.

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Ejecute el siguiente comando para crear un parámetro de varias líneas.

   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
   ```

   El siguiente ejemplo muestra el contenido del archivo `MultiLineParameter.json`.

```
{
    "Value": "<para>Paragraph One</para>\n<para>Paragraph Two</para>
\n<para>Paragraph Three</para>"
}
```

El valor del parámetro guardado se almacena de la siguiente manera.

```
<para>Paragraph One</para>
<para>Paragraph Two</para>
<para>Paragraph Three</para>
```

Utilizar las Herramientas para Windows PowerShell para crear un parámetro Parameter Store

Puede utilizar AWS Tools for Windows PowerShell para crear tipos de parámetros `String`, `StringList`, y `SecureString`. Después de eliminar un parámetro, espere al menos 30 segundos para crear un parámetro con el mismo nombre.

Los parámetros no se pueden referenciar ni anidar en los valores de otros parámetros. No puede incluir `{{}}` o `{{ssm:parameter-name}}` en el valor de un parámetro.

> **Note**
>
> Los parámetros solo están disponibles en la Región de AWS donde se crearon.

Temas

- Crear un parámetro String (Herramientas para Windows PowerShell)
- Crear un parámetro StringList (Herramientas para Windows PowerShell)
- Crear un parámetro SecureString (Herramientas para Windows PowerShell)

Crear un parámetro String (Herramientas para Windows PowerShell)

1. Instale y configure Herramientas de AWS para PowerShell (Herramientas para Windows PowerShell), si aún no lo ha hecho.

   Para obtener más información, consulte Instalación de Herramientas de AWS para PowerShell.

2. Ejecute el siguiente comando para crear un parámetro que contenga un valor de texto sin formato. Reemplace cada *example resource placeholder* con su propia información.

```
Write-SSMParameter `
    -Name "parameter-name" `
    -Value "parameter-value" `
    -Type "String"
```

-o bien-

Ejecute el siguiente comando para crear un parámetro que contenga un ID de Amazon Machine Image (AMI) como valor del parámetro.

> **Note**
>
> Para crear un parámetro con una etiqueta, cree service.model.tag antes como una variable. A continuación se muestra un ejemplo.
>
> ```
> $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
```

La opción `-DataType` solo se debe especificar si va a crear un parámetro que contiene un ID de AMI. Para todos los demás parámetros, el tipo de datos predeterminado es `text`. Para obtener más información, consulte Uso de compatibilidad con parámetros nativos en Parameter Store para los ID de Amazon Machine Image.

Aquí se incluye un ejemplo que utiliza una jerarquía de parámetros.

```
Write-SSMParameter `
    -Name "/IAD/Web/SQL/IPaddress" `
```

```
    -Value "99.99.99.999" `
    -Type "String" `
    -Tags $tag
```

3. Ejecute el siguiente comando para verificar los detalles del parámetro.

```
(Get-SSMParameterValue -Name "the-parameter-name-you-specified").Parameters
```

Crear un parámetro StringList (Herramientas para Windows PowerShell)

1. Instale y configure Herramientas de AWS para PowerShell (Herramientas para Windows PowerShell), si aún no lo ha hecho.

   Para obtener más información, consulte Instalación de Herramientas de AWS para PowerShell.

2. Ejecute el siguiente comando para crear un parámetro StringList. Reemplace cada *example resource placeholder* con su propia información.

   > **Note**
   >
   > Para crear un parámetro con una etiqueta, cree service.model.tag antes como una variable. A continuación se muestra un ejemplo.
   >
   > ```
   > $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
   ```

   Si se ejecuta correctamente, el comando devuelve el número de la versión del parámetro.

   A continuación se muestra un ejemplo.

   ```
   Write-SSMParameter `
   ```

```
    -Name "stringlist-parameter" `
    -Value "Milana,Mariana,Mark,Miguel" `
    -Type "StringList" `
    -Tags $tag
```

> **Note**
>
> Los elementos de `StringList` deben ir separados por comas (,). No se puede utilizar otra puntuación ni carácter especial para exceptuar los elementos de la lista. Si tiene un valor de parámetro que requiera una coma, utilice el tipo de datos `String`.

3. Ejecute el siguiente comando para verificar los detalles del parámetro.

```
(Get-SSMParameterValue -Name "the-parameter-name-you-specified").Parameters
```

Crear un parámetro SecureString (Herramientas para Windows PowerShell)

Antes de crear un parámetro `SecureString`, obtenga información sobre los requisitos de este tipo de parámetro. Para obtener más información, consulte Utilizar la AWS CLI para crear un parámetro SecureString.

> **Important**
>
> Solo se cifra el valor de un parámetro `SecureString`. Los nombres de parámetros, las descripciones y otras propiedades no se cifran.

> **Important**
>
> Parameter Store solo es compatible con claves de cifrado de KMS simétricas. No se puede utilizar una clave de cifrado de KMS asimétrica para cifrar los parámetros. Para obtener ayuda para determinar si una clave de KMS es simétrica o asimétrica, consulte Identificación de claves KMS simétricas y asimétricas en la Guía para desarrolladores de AWS Key Management Service.

1. Instale y configure Herramientas de AWS para PowerShell (Herramientas para Windows PowerShell), si aún no lo ha hecho.

Para obtener más información, consulte Instalación de Herramientas de AWS para PowerShell.

2. Para crear un parámetro, ejecute el siguiente comando. Reemplace cada *example resource placeholder* con su propia información.

> **Note**
>
> Para crear un parámetro con una etiqueta, primero cree service.model.tag como una variable. A continuación se muestra un ejemplo.
>
> ```
> $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
```

Si se ejecuta correctamente, el comando devuelve el número de la versión del parámetro.

> **Note**
>
> Para utilizar la Clave administrada de AWS asignada a su cuenta, quite el parámetro `-KeyId` del comando.

A continuación, se muestra un ejemplo que utiliza un nombre encubierto (3l3vat3131) para un parámetro de contraseña y una Clave administrada de AWS.

```
Write-SSMParameter `
    -Name "/Finance/Payroll/3l3vat3131" `
    -Value "P@sSwW)rd" `
    -Type "SecureString"`
```

```
    -Tags $tag
```

3. Ejecute el siguiente comando para verificar los detalles del parámetro.

```
(Get-SSMParameterValue -Name "the-parameter-name-you-specified" -WithDecryption
 $true).Parameters
```

De forma predeterminada, todos `SecureString` los valores se muestran como texto cifrado. Para descifrar un valor `SecureString`, un usuario debe tener permisos para llamar a la operación Decrypt de la API de AWS KMS. Para obtener más información acerca de cómo configurar el control de acceso de AWS KMS, consulte Autenticación y control de acceso de AWS KMS en la Guía para desarrolladores de AWS Key Management Service.

> **Important**
>
> Si cambia el alias de clave de KMS para la clave KMS que se utiliza para cifrar un parámetro, también actualice el alias de clave que utiliza el parámetro para hacer referencia a AWS KMS. Esto solo se aplica al alias de clave de KMS; el ID de clave al que se adjunta un alias permanece igual a menos que elimine toda la clave.

Búsqueda de parámetros de Parameter Store en Systems Manager

Cuando tiene muchos parámetros en su cuenta, puede ser difícil encontrar información sobre solo uno o varios parámetros a la vez. En este caso, puede utilizar herramientas de filtro para buscar aquellos sobre los que necesita información, de acuerdo con los criterios de búsqueda que especifique. Puede utilizar la consola de AWS Systems Manager, la AWS Command Line Interface (AWS CLI), la Herramientas de AWS para PowerShell o la API DescribeParameters para buscar parámetros.

Temas

- Utilizar la consola para buscar un parámetro
- Utilizar la AWS CLI para buscar un parámetro

Utilizar la consola para buscar un parámetro

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Parameter Store.
3. Seleccione en el cuadro de búsqueda y elija cómo desea buscar. Por ejemplo, `Type` o `Name`.
4. Proporcione información para el tipo de búsqueda seleccionado. Por ejemplo:
   - Si está buscando por `Type`, elija entre `String`, `StringList` o `SecureString`.
   - Si está buscando por `Name`, elija `contains`, `equals` o `begins-with` y, a continuación, ingrese todo o parte de un nombre de parámetro.

   > **Note**
   >
   > En la consola, el tipo de búsqueda predeterminado para `Name` es `contains`.

5. Pulse Intro.

La lista de parámetros se actualiza con los resultados de su búsqueda.

> **Note**
>
> Es posible que su búsqueda contenga más resultados de los que se muestran en la primera página de resultados. Utilice la flecha derecha (>) en la parte superior de la lista de parámetros (si está disponible) para ver el siguiente conjunto de resultados.

Utilizar la AWS CLI para buscar un parámetro

Utilice el comando `describe-parameters` para ver información sobre uno o más parámetros en la AWS CLI.

En los ejemplos siguientes se muestran varias opciones que puede utilizar para ver información sobre los parámetros de su Cuenta de AWS Para obtener más información sobre estas opciones, consulte describe-parameters en la Guía del usuario de AWS Command Line Interface.

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Reemplace los valores de ejemplo de los comandos siguientes por valores que reflejen los parámetros que se han creado en su cuenta.

Linux & macOS

```
aws ssm describe-parameters \
    --parameter-filters "Key=Name,Values=MyParameterName"
```

Windows

```
aws ssm describe-parameters ^
    --parameter-filters "Key=Name,Values=MyParameterName"
```

> **Note**
>
> Para `describe-parameters`, el tipo de búsqueda predeterminado para `Name` es `Equals`. En los filtros de parámetros, especificar `"Key=Name,Values=MyParameterName"` es lo mismo que especificar `"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**
>
> En el último ejemplo, *key-id* representa el ID de una clave AWS Key Management Service (AWS KMS) utilizada para cifrar un parámetro `SecureString` creado en su cuenta. Alternativamente, puede introducir **alias/aws/ssm** para usar la clave AWS KMS predeterminada para su cuenta. Para obtener más información, consulte Utilizar la AWS CLI para crear un parámetro SecureString.

Si se ejecuta correctamente, el comando devolverá información similar a la siguiente.

```
{
    "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": []
        }
    ]
}
```

Asignación de políticas de parámetros en Parameter Store

Las políticas de parámetros lo ayudan a administrar un conjunto creciente de parámetros, por lo que le permite asignar criterios específicos a un parámetro como, por ejemplo, una fecha de vencimiento o un período de vida. Las políticas de parámetros son especialmente útiles para actualizar o eliminar contraseñas y datos de configuración almacenados en Parameter Store, una herramienta de AWS Systems Manager. Parameter Store ofrece los siguientes tipos de políticas: `Expiration`, `ExpirationNotification` y `NoChangeNotification`.

> **Note**
>
> Para implementar ciclos de vida de rotación de contraseñas, utilice AWS Secrets Manager. Puede rotar, administrar y recuperar credenciales de bases de datos, claves de API y otros datos confidenciales durante todo su ciclo de vida con Secrets Manager. Para obtener más información, consulte ¿Qué es AWS Secrets Manager? en la Guía del usuario de AWS Secrets Manager.

Parameter Store aplica las políticas de parámetros mediante análisis asíncronos periódicos. Después de crear una política no tiene que realizar acciones adicionales para aplicarla. Parameter Store realiza de forma independiente la acción definida por la política de acuerdo con los criterios especificados.

> **Note**
>
> Las políticas de parámetros están disponibles para los parámetros que utilizan la capa de parámetros avanzados. Para obtener más información, consulte Administración de niveles de parámetros.

Una política de parámetro es una matriz JSON, tal y como se muestra en la siguiente tabla. Puede asignar una política al crear un nuevo parámetro avanzado, o bien puede aplicar una política actualizando un parámetro. Parameter Store admite los siguientes tipos de políticas de parámetros.

Política	Detalles	Ejemplos

Expiration	Esta política elimina el parámetro. Puede especificar una fecha y hora concretas mediante con el formato `ISO_INSTANT` o `ISO_OFFSET_DATE_TIME` . Para cambiar cuándo desea que se elimine el parámetro, actualice la política. Actualizar un parámetro no afecta a la fecha de vencimiento o la hora de la política asociada a él. Cuando se llega a la fecha y hora de vencimiento, Parameter Store elimina el parámetro.
	
	Note
	En este ejemplo se utiliza el formato `ISO_INSTANT` . También puede especificar una fecha y una hora con el formato `ISO_OFFSET_DATE_TIME` . A continuación se muestra un ejemplo: `2019-11-0`
		<pre>{
		    "Type": "Expiration",
		    "Version": "1.0",
		    "Attributes": {
		        "Timestamp": "2018-12-02T21:34:33.000Z"
		    }
		}</pre>

Política	Detalles	Ejemplos
	`1T22:13:4 8.87+10:30:00` .	
ExpirationNotification	Esta política inicia un evento en Amazon EventBridge (EventBridge) que le notifica acerca del vencimiento. Con esta política podrá recibir notificaciones antes de la fecha de vencimiento, en unidades de días u horas.	`{ "Type": "ExpirationNotification", "Version": "1.0", "Attributes": { "Before": "15", "Unit": "Days" } }`
NoChangeNotification	Esta política inicia un evento en EventBridge si un parámetro no se ha modificado durante un periodo de tiempo especificado. Este tipo de política es útil cuando, por ejemplo, debe cambiarse una contraseña en un periodo de tiempo. Esta política determina cuándo enviar una notificación leyendo el atributo `LastModifiedTime` del parámetro. Si cambia o edita un parámetro, el sistema restablece el periodo de tiempo de notificación en función del nuevo valor de `LastModifiedTime` .	`{ "Type": "NoChangeNotification", "Version": "1.0", "Attributes": { "After": "20", "Unit": "Days" } }`

Puede asignar varias políticas a un parámetro. Por ejemplo, puede asignar las políticas `Expiration` y `ExpirationNotification` para que el sistema inicie un evento de EventBridge para informarle sobre la eliminación inminente de un parámetro. Puede asignar a un parámetro un máximo de diez (10) políticas.

En el siguiente ejemplo se muestra la sintaxis de una solicitud de API PutParameter que asigna cuatro políticas a un nuevo parámetro `SecureString` denominado `ProdDB3`.

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

Adición de políticas a un parámetro existente

Esta sección contiene información sobre cómo agregar políticas a un parámetro existente usando la consola de AWS Systems Manager, la AWS Command Line Interface (AWS CLI), y AWS Tools for Windows PowerShell. Para obtener información acerca de cómo crear un nuevo parámetro que incluya políticas, consulte Creación de parámetros de Parameter Store en Systems Manager.

Temas

- Utilizar la consola para agregar políticas a un parámetro existente
- Utilizar la AWS CLI para agregar políticas a un parámetro existente
- Agregar políticas a un parámetro existente (Herramientas para Windows PowerShell)

Utilizar la consola para agregar políticas a un parámetro existente

Utilice el siguiente procedimiento para agregar políticas a un parámetro existente mediante la consola de Systems Manager.

Para añadir políticas a un parámetro existente

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Parameter Store.
3. Elija la opción situada junto al parámetro que desea actualizar para incluir políticas y, a continuación, elija Edit (Editar).
4. Seleccione Avanzado.
5. (Opcional) En la sección Parameter policies (Políticas de parámetros), elija Enabled (Habilitadas). Puede especificar una fecha de vencimiento y una o varias políticas de notificación para este parámetro.
6. Seleccione Save changes (Guardar cambios).

Important

- Parameter Store conserva las políticas en un parámetro hasta que sobrescriba las políticas con nuevas políticas o las elimine.
- Para eliminar todas las políticas de un parámetro existente, edite el parámetro y aplique una política vacía mediante corchetes y llaves, tal y como se indica a continuación: `[{}]`
- Si agrega una nueva política a un parámetro que ya tenga políticas, Systems Manager sobrescribe las políticas asociadas al parámetro. Se eliminan las políticas existentes. Si desea agregar una nueva política a un parámetro que ya tiene una o varias políticas, copie y pegue las políticas originales, escriba la nueva política y, a continuación, guarde los cambios.

Utilizar la AWS CLI para agregar políticas a un parámetro existente

Utilice el siguiente procedimiento para añadir políticas a un parámetro existente mediante la AWS CLI.

Para añadir políticas a un parámetro existente

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Ejecute el siguiente comando para añadir políticas a un parámetro existente. Reemplace cada *example resource placeholder* con su propia información.

   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}]"
```

A continuación se muestra un ejemplo que incluye una política de vencimiento que elimina el parámetro después de 15 días. El ejemplo también incluye una política de notificación que genera un evento de EventBridge cinco (5) días antes de que se elimine el parámetro. Por último, incluye una política `NoChangeNotification` si no se producen cambios en este parámetro después de 60 días. En el ejemplo se utiliza un nombre encubierto (`3l3vat3131`) para una contraseña y una AWS Key Management Service AWS KMS key. Para obtener más información acerca de AWS KMS keys, consulte Conceptos de AWS Key Management Service en la Guía para desarrolladores de AWS Key Management Service.

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. Ejecute el siguiente comando para verificar los detalles del parámetro. Reemplace el *nombre del parámetro* con su propia información.

   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 conserva políticas para un parámetro hasta que se sobrescriban con nuevas políticas o se eliminen.
- Para eliminar todas las políticas de un parámetro existente, edite el parámetro y aplique una política sin corchetes ni llaves. Reemplace cada *example resource placeholder* con su propia información. Por ejemplo:

  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 agrega una nueva política a un parámetro que ya tenga políticas, Systems Manager sobrescribe las políticas asociadas al parámetro. Se eliminan las políticas existentes. Si desea agregar una nueva política a un parámetro que ya tiene una o varias políticas, copie y pegue las políticas originales, escriba la nueva política y, a continuación, guarde los cambios.

Agregar políticas a un parámetro existente (Herramientas para Windows PowerShell)

Utilice el siguiente procedimiento para agregar políticas a un parámetro existente utilizando Tools for Windows PowerShell. Reemplace cada *example resource placeholder* con su propia información.

Para añadir políticas a un parámetro existente

1. Abra Tools for Windows PowerShell y ejecute el siguiente comando para especificar sus credenciales. Debe tener permisos de administrador en Amazon Elastic Compute Cloud (Amazon EC2) o se le deben haber concedido los permisos adecuados en AWS Identity and Access Management (IAM).

   ```
   Set-AWSCredentials `
       -AccessKey access-key-name `
       -SecretKey secret-key-name
   ```

2. Ejecute el siguiente comando para establecer la región de la sesión de PowerShell. En el ejemplo se utiliza la región EE. UU. Este (Ohio) (us-east-2).

   ```
   Set-DefaultAWSRegion `
       -Region us-east-2
   ```

3. Ejecute el siguiente comando para añadir políticas a un parámetro existente. Reemplace cada *example resource placeholder* con su propia información.

   ```
   Write-SSMParameter `
       -Name "parameter name" `
       -Value "parameter value" `
       -Type "parameter type" `
   ```

```
    -Policies "[{policies-enclosed-in-brackets-and-curly-braces}]" `
    -Overwrite
```

A continuación se muestra un ejemplo que incluye una política de vencimiento que elimina el parámetro a medianoche (GMT) el 13 de mayo de 2020. El ejemplo también incluye una política de notificación que genera un evento de EventBridge cinco (5) días antes de que se elimine el parámetro. Por último, incluye una política `NoChangeNotification` si no se producen cambios en este parámetro después de 60 días. En el ejemplo se utiliza un nombre encubierto (`3l3vat3131`) para una contraseña y una Clave administrada de 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. Ejecute el siguiente comando para verificar los detalles del parámetro. Reemplace el *nombre del parámetro* con su propia información.

```
(Get-SSMParameterValue -Name "parameter name").Parameters
```

Important

- Parameter Store conserva las políticas en un parámetro hasta que sobrescriba las políticas con nuevas políticas o las elimine.
- Para eliminar todas las políticas de un parámetro existente, edite el parámetro y aplique una política sin corchetes ni llaves. Por ejemplo:

```
Write-SSMParameter `
    -Name "parameter name" `
    -Value "parameter value" `
    -Type "parameter type" `
    -Policies "[{}]"
```

- Si agrega una nueva política a un parámetro que ya tenga políticas, Systems Manager sobrescribe las políticas asociadas al parámetro. Se eliminan las políticas existentes. Si desea agregar una nueva política a un parámetro que ya tiene una o varias políticas, copie y pegue las políticas originales, escriba la nueva política y, a continuación, guarde los cambios.

Trabajo con jerarquías de parámetros en Parameter Store

Administrar docenas o centenares de parámetros como una lista sin formato requiere mucho tiempo y es una labor propensa a errores. También puede ser difícil identificar el parámetro correcto para una tarea. Esto significa que puede utilizar accidentalmente el parámetro equivocado o puede crear varios parámetros que utilizan los mismos datos de configuración.

Puede utilizar las jerarquías de parámetros como ayuda para organizar y administrar los parámetros de . Una jerarquía es un nombre de parámetro que incluye una ruta definida mediante barras inclinadas (/).

Temas

- Conozca la jerarquía de los parámetros a través de ejemplos
- Realización de consultas de parámetros en una jerarquía
- Utilizar la AWS CLI para administrar las jerarquías de los parámetros

Conozca la jerarquía de los parámetros a través de ejemplos

El ejemplo siguiente utiliza tres niveles de jerarquía en el nombre para identificar lo siguiente:

```
/Environment/Type of computer/Application/Data
```

```
/Dev/DBServer/MySQL/db-string13
```

Puede crear una jerarquía con un máximo de 15 niveles. Le recomendamos crear jerarquías que reflejen una estructura jerárquica existente en su entorno, tal y como se muestra en los siguientes ejemplos:

- El entorno de integración continua (CI) y entrega continua (CD) (flujos de trabajo de CI/CD)

  ```
  /Dev/DBServer/MySQL/db-string
  ```

  ```
  /Staging/DBServer/MySQL/db-string
  ```

```
/Prod/DBServer/MySQL/db-string
```

- Las aplicaciones que utilizan contenedores

```
/MyApp/.NET/Libraries/my-password
```

- La organización empresarial

```
/Finance/Accountants/UserList
```

```
/Finance/Analysts/UserList
```

```
/HR/Employees/EU/UserList
```

Las jerarquías de parámetros normalizan la forma de crear parámetros y hacen que sea más sencillo administrar los parámetros a lo largo del tiempo. Una jerarquía de parámetros también puede ayudarle a identificar el parámetro correcto para una tarea de configuración. Esto le ayuda a evitar la creación de varios parámetros con los mismos datos de configuración.

Puede crear una jerarquía que le permita compartir parámetros en diferentes entornos, tal y como se muestra en los siguientes ejemplos, que utilizan contraseñas en el entorno de desarrollo y pruebas.

`/DevTest/MyApp/database/my-password`

A continuación, puede crear una contraseña única para el entorno de producción, tal y como se muestra en el ejemplo siguiente:

`/prod/MyApp/database/my-password`

No es necesario especificar una jerarquía de parámetros. Puede crear parámetros en el nivel uno. Estos se denominan parámetros raíz. En lo que se refiere a la compatibilidad con versiones anteriores, todos los parámetros creados en Parameter Store antes de publicarse las jerarquías son parámetros raíz. El sistema trata los dos parámetros siguientes como parámetros raíz.

`/parameter-name`

`parameter-name`

Realización de consultas de parámetros en una jerarquía

Otro beneficio del uso de jerarquías es la capacidad de consultar todos los parámetros de un nivel determinado de una jerarquía con la operación GetParametersByPath de la API. Por ejemplo, si

ejecuta el siguiente comando en la AWS Command Line Interface (AWS CLI), el sistema devuelve todos los parámetros del nivel `Oncall`:

```
aws ssm get-parameters-by-path --path /Dev/Web/Oncall
```

Resultado de ejemplo:

```
{
    "Parameters": [
        {
            "Name": "/Dev/Web/Oncall/Week1",
            "Type": "String",
            "Value": "John Doe",
            "Version": 1,
            "LastModifiedDate": "2024-11-22T07:18:53.510000-08:00",
            "ARN": "arn:aws:ssm:us-east-2:123456789012:parameter/Dev/Web/Oncall/Week1",
            "DataType": "text"
        },
        {
            "Name": "/Dev/Web/Oncall/Week2",
            "Type": "String",
            "Value": "Mary Major",
            "Version": 1,
            "LastModifiedDate": "2024-11-22T07:21:25.325000-08:00",
            "ARN": "arn:aws:ssm:us-east-2:123456789012:parameter/Dev/Web/Oncall/Week2",
            "DataType": "text"
        }
    ]
}
```

Para ver los parámetros `SecureString` descifrados en una jerarquía, debe especificar la ruta y el parámetro `--with-decryption`, tal y como se muestra en el siguiente ejemplo.

```
aws ssm get-parameters-by-path --path /Prod/ERP/SAP --with-decryption
```

Utilizar la AWS CLI para administrar las jerarquías de los parámetros

Este procedimiento le muestra cómo trabajar con parámetros y jerarquías de parámetros con la AWS CLI.

Para administrar parámetros mediante jerarquías

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Ejecute el siguiente comando para crear un parámetro que use el parámetro `allowedPattern` y el tipo de parámetro `String`. El patrón permitido en este ejemplo significa que el valor del parámetro debe tener entre 1 y 4 dígitos.

   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
   ```

   El comando devuelve el número de la versión del parámetro.

3. Ejecute el siguiente comando para intentar sobrescribir el parámetro que acaba de crear con un nuevo valor.

   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
```

El sistema devuelve el siguiente error porque el valor nuevo no cumple los requisitos del patrón permitido que especificó en el paso anterior.

```
An error occurred (ParameterPatternMismatchException) when calling the PutParameter
 operation: Parameter value, cannot be validated against allowedPattern: \d{1,4}
```

4. Ejecute uno de los siguientes comandos para crear un parámetro `SecureString` que utilice una Clave administrada de AWS. El patrón permitido en este ejemplo significa que el usuario puede especificar cualquier carácter y que el valor debe tener entre 8 y 20 caracteres.

   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. Ejecute los siguientes comandos para crear más parámetros que utilicen la estructura jerárquica del paso anterior.

   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. Ejecute el siguiente comando para obtener el valor de dos parámetros.

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. Ejecute el siguiente comando para consultar todos los parámetros de un mismo nivel.

   Linux & macOS

   ```
   aws ssm get-parameters-by-path \
       --path "/MyService/Test"
   ```

   Windows

   ```
   aws ssm get-parameters-by-path ^
       --path "/MyService/Test"
   ```

8. Ejecute el siguiente comando para eliminar dos parámetros.

   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"
   ```

Impedir el acceso a las operaciones de la API de Parameter Store

Al utilizar condiciones específicas del servicio compatibles con las políticas de Systems Manager para las políticas de AWS Identity and Access Management (IAM), puede permitir o denegar explícitamente el acceso a las operaciones y el contenido de la API de Parameter Store. Al utilizar estas condiciones, puede permitir que solo determinadas entidades de IAM (usuarios y roles) de su organización activen determinadas acciones de la API o impedir que determinadas entidades de IAM las ejecuten. Esto incluye las acciones que se ejecutan a través de la consola de Parameter Store, el AWS Command Line Interface (AWS CLI) y los SDK.

Systems Manager admite actualmente tres condiciones específicas de Parameter Store.

Temas

- Utilizar ssm:Overwrite para evitar cambios en los parámetros existentes

- Utilizar ssm:Policies para evitar la creación o actualización de parámetros que utilizan una política de parámetros
- Utilizar ssm:Recursive para evitar el acceso a niveles de un parámetro jerárquico

Utilizar **`ssm:Overwrite`** para evitar cambios en los parámetros existentes

Utilice la condición `ssm:Overwrite` para controlar si las entidades de IAM pueden actualizar los parámetros existentes.

En la siguiente política de ejemplo, la declaración `"Allow"` concede permiso para crear parámetros mediante la ejecución de la operación de la API de `PutParameter` en la Cuenta de AWS 123456789012 en la región Este de EE. UU. (Ohio) (us-east-2).

Sin embargo, la declaración `"Deny"` evita que las entidades cambien los valores de los parámetros existentes porque la opción `Overwrite` se niega explícitamente para la operación de `PutParameter`. En otras palabras, las entidades que tienen asignada esta política pueden crear parámetros, pero no realizar cambios en los parámetros existentes.

JSON

```
{
    "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/*"
        }
    ]
}
```

Utilizar **`ssm:Policies`** para evitar la creación o actualización de parámetros que utilizan una política de parámetros

Use la condición `ssm:Policies` para controlar si las entidades pueden crear parámetros que incluyan una política de parámetros y actualizar los parámetros existentes que incluyan una política de parámetros.

En el siguiente ejemplo de política, la instrucción `"Allow"` concede permisos generales para crear parámetros, pero la instrucción `"Deny"` impide que las entidades creen o actualicen parámetros que incluyan una política de parámetros en la Cuenta de AWS 123456789012 en la región Este de EE. UU. (Ohio) (us-east-2). Las entidades pueden seguir creando o actualizando parámetros a los que no se les haya asignado una política de parámetros.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:PutParameter"
            ],
            "Resource": "arn:aws:ssm:us-east-2:111122223333:parameter/*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "ssm:PutParameter"
            ],
            "Condition": {
                "StringEquals": {
                    "ssm:Policies": [
                        "true"
```

```
                    ]
                }
            },
            "Resource": "arn:aws:ssm:us-east-2:111122223333:parameter/*"
        }
    ]
}
```

Utilizar **`ssm:Recursive`** para evitar el acceso a niveles de un parámetro jerárquico

Utilice la condición `ssm:Recursive` para controlar si las entidades de IAM pueden ver o hacer referencia a los niveles de un parámetro jerárquico. Puede proporcionar o restringir el acceso a todos los parámetros más allá de un nivel específico de una jerarquía.

En la siguiente política de ejemplo, la declaración `"Allow"` concede acceso a las operaciones de Parameter Store en todos los parámetros en la ruta `/Code/Departments/Finance/*` para la Cuenta de AWS 123456789012 en la región Este de EE. UU. (Ohio) (us-east-2).

Después de esto, la declaración `"Deny"` impide que las entidades de IAM visualicen o recuperen datos de parámetros a un nivel igual o inferior al nivel de `/Code/Departments/*`. Sin embargo, las entidades pueden seguir creando o actualizando parámetros en esa ruta. El ejemplo se ha diseñado para ilustrar que denegar de forma recursiva el acceso por debajo de un nivel determinado en una jerarquía de parámetros tiene prioridad sobre el acceso más permisivo en la misma política.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:*"
            ],
            "Resource": "arn:aws:ssm:us-east-2:123456789012:parameter/*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "ssm:GetParametersByPath"
```

```
            ],
            "Condition": {
                "StringEquals": {
                    "ssm:Recursive": [
                        "true"
                    ]
                }
            },
            "Resource": "arn:aws:ssm:us-east-2:123456789012:parameter/Code/
Departments/*"
        }
    ]
}
```

> **Important**
>
> Si un usuario tiene acceso a una ruta, puede obtener acceso a todos los niveles de esa ruta. Por ejemplo, si un usuario tiene permiso para obtener acceso a la ruta `/a`, el usuario también puede obtener acceso a `/a/b`. Esto es cierto a menos que se haya denegado explícitamente al usuario el acceso a IAM por `/b` de parámetro, como se ha ilustrado anteriormente.

Trabajo con etiquetas de parámetros en Parameter Store

Una etiqueta de parámetro es un alias definido por el usuario que ayuda a administrar las distintas versiones de un parámetro. Cuando se modifica un parámetro, AWS Systems Manager guarda automáticamente una versión nueva e incrementa en uno el número de la versión. Un rótulo puede ayudarle a recordar el propósito de una versión de un parámetro cuando hay varias versiones.

Por ejemplo, supongamos que tiene un parámetro llamado `/MyApp/DB/ConnectionString`. El valor del parámetro es una cadena de conexión a un servidor MySQL de una base de datos local en un entorno de pruebas. Cuando termine de actualizar la aplicación, desea que el parámetro utilice una cadena de conexión para una base de datos de producción. y cambia el valor de `/MyApp/DB/ConnectionString`. Systems Manager crea automáticamente la versión dos con la nueva cadena de conexión. Para recordar mejor el propósito de cada versión, asocie un rótulo a cada parámetro. Para la versión uno, asocie el rótulo Pruebas y para la versión dos, asocie el rótulo Producción.

Puede mover rótulos de una versión de un parámetro a otra. Por ejemplo, si crea la versión tres del parámetro `/MyApp/DB/ConnectionString` con una cadena de conexión para una nueva base

de datos de producción, puede mover la etiqueta Producción de la versión dos a la versión tres del parámetro.

Los rótulos de los parámetros son una alternativa ligera a las etiquetas de los parámetros. Puede que su organización tenga unas directrices estrictas para las etiquetas que deben aplicarse a los distintos recursos de AWS. Por el contrario, un rótulo es simplemente una asociación de texto para una versión específica de un parámetro.

Al igual que ocurre con las etiquetas, puede consultar los parámetros mediante el uso de rótulos. Puede ver una lista de las versiones específicas de los parámetros que utilizan la misma etiqueta si consulta el conjunto de parámetros mediante la operación GetParametersByPath de la API, tal y como se describe más adelante en esta sección.

> **Note**
>
> Si ejecuta un comando que especifica una versión de un parámetro que no existe, el comando fallará, ya que no recurre al valor más reciente o predeterminado del parámetro.

Requisitos y restricciones de etiquetas

Los rótulos de los parámetros tienen los siguientes requisitos y restricciones:

- Una versión de un parámetro puede tener un máximo de 10 rótulos.
- No se puede asociar el mismo rótulo a distintas versiones del mismo parámetro. Por ejemplo, si la versión 1 del parámetro tiene la etiqueta Producción, no se podrá asociar Producción a la versión 2.
- Puede mover un rótulo de una versión de un parámetro a otra versión.
- No puede crear una etiqueta cuando crea un parámetro. Se debe asociar un rótulo a una versión específica de un parámetro.
- Si ya no desea utilizar una etiqueta de parámetro, puede transferirlo a otra versión de un parámetro o eliminarla.
- Un rótulo puede tener un máximo de 100 caracteres.
- Los rótulos pueden contener letras (se distingue entre mayúsculas y minúsculas), números, puntos (.), guiones (-) o guiones bajos (_).

- Las etiquetas no pueden comenzar por un número, por “aws” ni por “ssm” (no se distingue entre mayúsculas y minúsculas). Si una etiqueta no cumple estos requisitos, la etiqueta no se asocia a la versión del parámetro y el sistema la muestra en la lista `InvalidLabels`.

Temas

- Trabajo de etiquetas de parámetros mediante la consola
- Trabajo con etiquetas de parámetros mediante la AWS CLI

Trabajo de etiquetas de parámetros mediante la consola

En esta sección, se describe cómo realizar las siguientes tareas mediante la consola de Systems Manager.

- Utilizar la consola para crear una etiqueta de parámetro
- Utilizar la consola para ver etiquetas asociadas a un parámetro
- Utilizar la consola para mover una etiqueta de parámetro
- Utilizar la consola para eliminar etiquetas de parámetros

Utilizar la consola para crear una etiqueta de parámetro

El siguiente procedimiento describe cómo asociar una etiqueta a una versión específica de un parámetro existente mediante la consola de Systems Manager. No se puede asociar un rótulo al crear un parámetro.

Para adjuntar una etiqueta a una de un parámetro

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Parameter Store.
3. Elija el nombre de un parámetro para abrir la página de detalles de ese parámetro.
4. Elija la pestaña History (Historial).
5. Elija la versión del parámetro a la que desea asociar un rótulo.
6. Elija Manage labels (Administrar etiquetas).
7. Elija Add new label (Agregar nueva etiqueta).

8. En el cuadro de texto, ingrese el nombre de etiqueta. Para agregar más etiquetas, elija Add new label (Agregar nueva etiqueta). Puede asociar un máximo de diez rótulos.
9. Cuando haya finalizado, elija Save changes (Guardar cambios).

Utilizar la consola para ver etiquetas asociadas a un parámetro

Una versión de un parámetro puede tener un máximo de diez rótulos. El siguiente procedimiento describe cómo ver todas las etiquetas asociadas a una versión de un parámetro mediante la consola de Systems Manager.

Para ver las etiquetas asociadas a la versión de un parámetro

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Parameter Store.
3. Elija el nombre de un parámetro para abrir la página de detalles de ese parámetro.
4. Elija la pestaña History (Historial).
5. Busque la versión de parámetro para la que desea ver todos los rótulos asociados. La columna Labels (Rótulos) muestra todos los rótulos asociados a la versión del parámetro.

Utilizar la consola para mover una etiqueta de parámetro

El siguiente procedimiento describe cómo trasladar una etiqueta de parámetro a otra versión del mismo parámetro mediante la consola de Systems Manager.

Para trasladar una etiqueta a otra versión de un parámetro

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Parameter Store.
3. Elija el nombre de un parámetro para abrir la página de detalles de ese parámetro.
4. Elija la pestaña History (Historial).
5. Elija la versión del parámetro cuyo rótulo desea trasladar.
6. Elija Manage labels (Administrar etiquetas).
7. Elija Add new label (Agregar nueva etiqueta).

8. En el cuadro de texto, ingrese el nombre de etiqueta.
9. Cuando haya finalizado, elija Save changes (Guardar cambios).

Utilizar la consola para eliminar etiquetas de parámetros

En el siguiente procedimiento se describe cómo eliminar una o varias etiquetas de parámetros mediante la consola de Systems Manager.

Para eliminar etiquetas de un parámetro

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Parameter Store.
3. Elija el nombre de un parámetro para abrir la página de detalles de ese parámetro.
4. Elija la pestaña History (Historial).
5. Elija la versión del parámetro cuyas etiquetas desea eliminar.
6. Elija Manage labels (Administrar etiquetas).
7. Elija Remove (Eliminar). Al lado de cada etiqueta que desee eliminar.
8. Cuando haya finalizado, elija Save changes (Guardar cambios).
9. Confirme que los cambios son correctos, ingrese `Confirm` en el cuadro de texto y elija Confirm (Confirmar).

Trabajo con etiquetas de parámetros mediante la AWS CLI

En esta sección, se describe cómo realizar las siguientes tareas mediante la AWS Command Line Interface (AWS CLI).

- Utilizar la AWS CLI para crear una etiqueta de parámetro nueva
- Utilizar la AWS CLI para ver las etiquetas de un parámetro
- Utilizar la AWS CLI para ver una lista de parámetros asignados a una etiqueta
- Utilizar la AWS CLI para mover una etiqueta de parámetro
- Utilizar la AWS CLI para eliminar etiquetas de parámetros

Utilizar la AWS CLI para crear una etiqueta de parámetro nueva

El siguiente procedimiento describe cómo asociar un rótulo a una versión específica de un parámetro existente mediante la AWS CLI. No se puede asociar un rótulo al crear un parámetro.

Para crear una etiqueta de parámetro

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Ejecute el siguiente comando para ver una lista de parámetros para los que tiene permiso para asociarles una etiqueta.

   > **Note**
   >
   > Los parámetros solo están disponibles en la Región de AWS donde se crearon. Si no ve el parámetro al que desea asociar un rótulo, compruebe la región.

   ```
   aws ssm describe-parameters
   ```

   Anote el nombre del parámetro al que desea asociar un rótulo.

3. Ejecute el siguiente comando para ver todas las versiones del parámetro.

   ```
   aws ssm get-parameter-history --name "parameter-name"
   ```

   Anote la versión del parámetro a la que desea asociar un rótulo.

4. Ejecute el siguiente comando para recuperar información sobre un parámetro por número de versión.

   ```
   aws ssm get-parameters --names "parameter-name:version-number"
   ```

   A continuación se muestra un ejemplo.

   ```
   aws ssm get-parameters --names "/Production/SQLConnectionString:3"
   ```

5. Ejecute uno de los siguientes comandos para asociar una etiqueta a una versión de un parámetro. Si asocia varias etiquetas, separe los nombres mediante espacios.

   Asociación de un rótulo a la versión más reciente de un parámetro

   ```
   aws ssm label-parameter-version --name parameter-name  --labels label-name
   ```

   Asociación de un rótulo a una versión específica de un parámetro

   ```
   aws ssm label-parameter-version --name parameter-name --parameter-version version-number --labels label-name
   ```

   Estos son algunos ejemplos.

   ```
   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 salida muestra la etiqueta que creó en la lista `InvalidLabels`, significa que la etiqueta no cumple los requisitos descritos anteriormente en este tema. Examine los requisitos e inténtelo de nuevo. Si la lista `InvalidLabels` está vacía, significa que el rótulo se ha aplicado correctamente a la versión del parámetro.

6. Puede ver los detalles del parámetro mediante un número de versión o un nombre de rótulo. Ejecute el siguiente comando y especifique la etiqueta que ha creado en el paso anterior.

   ```
   aws ssm get-parameter --name parameter-name:label-name --with-decryption
   ```

   El comando devuelve información similar a la siguiente.

   ```
   {
       "Parameter": {
           "Version": version-number,
           "Type": "parameter-type",
   ```

```
        "Name": "parameter-name",
        "Value": "parameter-value",
        "Selector": ":label-name"
    }
}
```

> **Note**
>
> Selector en la salida es el número de versión o el rótulo que ha especificado en el campo de entrada Name.

Utilizar la AWS CLI para ver las etiquetas de un parámetro

Puede utilizar la operación GetParameterHistory de la API para ver el historial completo y todas las etiquetas asociadas a un parámetro determinado. También puede utilizar la operación GetParametersByPath de la API para ver una lista de todos los parámetros que tienen asignada una etiqueta determinada.

Para ver las etiquetas de un parámetro mediante la operación GetParameterHistory de la API

1. Ejecute el siguiente comando para ver la lista de los parámetros cuyas etiquetas puede ver.

   > **Note**
   >
   > Los parámetros solo están disponibles en la región donde se crearon. Si no ve el parámetro cuyos rótulos desea ver, compruebe la región.

   ```
   aws ssm describe-parameters
   ```

   Anote el nombre del parámetro del que desea ver las etiquetas.

2. Ejecute el siguiente comando para ver todas las versiones del parámetro.

   ```
   aws ssm get-parameter-history --name parameter-name --with-decryption
   ```

   El sistema devuelve información similar a la siguiente.

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

Utilizar la AWS CLI para ver una lista de parámetros asignados a una etiqueta

Puede utilizar la operación GetParametersByPath de la API para ver una lista de todos los parámetros de una ruta que tienen asignada una etiqueta determinada.

Ejecute el siguiente comando para ver una lista de los parámetros de una ruta que tienen asignada una etiqueta determinada. Reemplace cada *example resource placeholder* con su propia información.

```
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
```

El sistema devuelve información similar a la siguiente. En este ejemplo, el usuario buscó en la ruta `/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"
        }
    ]
}
```

Utilizar la AWS CLI para mover una etiqueta de parámetro

El siguiente procedimiento describe cómo trasladar un rótulo de parámetro a otra versión del mismo parámetro.

Para trasladar un rótulo de parámetro

1. Ejecute el siguiente comando para ver todas las versiones del parámetro. Reemplace el *nombre del parámetro* con su propia información.

   ```
   aws ssm get-parameter-history \
       --name "parameter name"
   ```

   Tenga en cuenta las versiones de parámetros a las que desea mover la etiqueta.

2. Ejecute el siguiente comando para asignar una etiqueta existente a otra versión de un parámetro. Reemplace cada *example resource placeholder* con su propia información.

```
aws ssm label-parameter-version \
    --name parameter name \
    --parameter-version version number \
    --labels name-of-existing-label
```

> **Note**
>
> Si desea mover un rótulo existente a la versión más reciente de un parámetro, elimine `--parameter-version` del comando.

Utilizar la AWS CLI para eliminar etiquetas de parámetros

En el siguiente procedimiento se describe cómo eliminar etiquetas de parámetros mediante la AWS CLI.

Para eliminar una etiqueta de parámetro

1. Ejecute el siguiente comando para ver todas las versiones del parámetro. Reemplace el *nombre del parámetro* con su propia información.

```
aws ssm get-parameter-history \
    --name "parameter name"
```

El sistema devuelve información similar a la siguiente.

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

Tenga en cuenta la versión del parámetro a la que desea eliminar una o unas etiquetas.

2. Ejecute el siguiente comando para eliminar las etiquetas que elija de ese parámetro. Reemplace cada *example resource placeholder* con su propia información.

```
aws ssm unlabel-parameter-version \
    --name parameter name \
    --parameter-version version \
    --labels label 1,label 2,label 3
```

El sistema devuelve información similar a la siguiente.

```
{
    "InvalidLabels": ["invalid"],
    "DeletedLabels" : ["Prod"]
 }
```

Trabajo con versiones de parámetros en Parameter Store

Cada vez que edita el valor de un parámetro, Parameter Store, una herramienta de AWS Systems Manager crea una nueva versión del parámetro y retiene las versiones anteriores. Al crear inicialmente un parámetro, Parameter Store asigna la versión 1 a dicho parámetro. Cuando cambia el valor del parámetro, incrementa Parameter Store automáticamente el número de versión por uno. Puede ver los detalles, incluidos los valores, de todas las versiones del historial de un parámetro.

También puede especificar la versión de un parámetro que se va a utilizar en comandos API y documentos SSM; por ejemplo: `ssm:MyParameter:3`. Puede especificar un nombre de parámetro y un número de versión específico en las llamadas a la API y los documentos de &SSM;. Si no especifica un número de versión, el sistema utiliza automáticamente la versión más reciente. Si especifica el número de una versión que no existe, el sistema devuelve un error en lugar de recurrir a la versión más reciente o predeterminada del parámetro.

También puede utilizar versiones de parámetros para ver la cantidad de veces que ha cambiado un parámetro durante un período de tiempo. Las versiones de parámetros también proporcionan una capa de protección si un valor de parámetro se cambia accidentalmente.

Puede crear y mantener un máximo de 100 versiones de un parámetro. Después de crear 100 versiones de un parámetro, cada vez que cree una nueva versión, la versión más antigua del parámetro se elimina del historial para dejar espacio a la nueva versión.

Una excepción a esto es cuando ya hay 100 versiones de parámetros en el historial y se asigna una etiqueta de parámetro a la versión más antigua de un parámetro. En este caso, esa versión no se elimina del historial y se produce un error en la solicitud de crear una nueva versión de parámetro. Esta protección es para evitar que se eliminen las versiones de parámetros con etiquetas esenciales asignadas a ellas. Para continuar creando nuevos parámetros, primero mueva la etiqueta de la versión más antigua del parámetro a una más nueva para utilizarla en sus operaciones. Para obtener información acerca de cómo mover etiquetas de parámetros, consulte Utilizar la consola para mover una etiqueta de parámetro y Utilizar la AWS CLI para mover una etiqueta de parámetro.

Los siguientes procedimientos muestran cómo editar un parámetro y, a continuación, verificar que ha creado una nueva versión. Puede utilizar los comandos `get-parameter` y `get-parameters` para ver las versiones de parámetros. Para obtener ejemplos sobre el uso de estos comandos, consulte GetParameter y GetParameters en la Referencia de la API AWS Systems Manager

Temas

- Utilizar la consola para crear una nueva versión de un parámetro

- Referencia a una versión del parámetro

Utilizar la consola para crear una nueva versión de un parámetro

Puede utilizar la consola de Systems Manager para crear una nueva versión de un parámetro y ver su historial de versiones.

Para crear una nueva versión de un parámetro

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Parameter Store.
3. Seleccione el nombre de un parámetro que creó anteriormente. Para obtener información acerca de cómo crear un nuevo parámetro, consulte Creación de parámetros de Parameter Store en Systems Manager.
4. Elija Editar.
5. En el cuadro Value (Valor), ingrese un nuevo valor y, a continuación, elija Save changes (Guardar cambios).
6. Elija el nombre del parámetro que acaba de actualizar. En la pestaña Información general, verifique que el número de versión se incrementa en 1 y verifique el nuevo valor.
7. Para ver el historial de todas las versiones de un parámetro, elija la ficha Historial .

Referencia a una versión del parámetro

Puede hacer referencia a versiones específicas de parámetros en comandos, llamadas a la API y documentos de SSM utilizando el siguiente formato: ssm:*parameter-name*:*version-number*.

En el ejemplo siguiente, Amazon Elastic Compute Cloud (Amazon EC2) `run-instances command` utiliza la versión 3 del parámetro `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**
>
> El uso de `resolve` y un valor de parámetro solo funciona con la opción `--image-id` y un parámetro que contenga una Amazon Machine Image (AMI) como su valor. Para obtener más información, consulte Uso de compatibilidad con parámetros nativos en Parameter Store para los ID de Amazon Machine Image.

A continuación se muestra un ejemplo para especificar la versión 2 de un parámetro denominado `MyRunCommandParameter` en un documento 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}}"
                ]
            }
        }
    ]
}
```

Trabajo con parámetros compartidos en Parameter Store

Compartir parámetros avanzados simplifica la administración de los datos de configuración en un entorno con varias cuentas. Puede almacenar y administrar de forma centralizada los parámetros y compartirlos con otras Cuentas de AWS que necesiten hacer referencia a ellos.

Parameter Store se integra con AWS Resource Access Manager (AWS RAM) para permitir el intercambio avanzado de parámetros. AWS RAM es un servicio que permite compartir recursos con otras Cuentas de AWS o a través de AWS Organizations.

Con AWS RAM, puede compartir recursos de su propiedad creando un uso compartido de recursos. Un uso compartido de recursos especifica los recursos que compartir, los permisos que dar y los consumidores con quienes compartir. Los consumidores pueden incluir lo siguiente:

- Cuentas de AWS específicas dentro o fuera de su organización en AWS Organizations

- Una unidad organizativa dentro de la organización en AWS Organizations
- Toda la organización en AWS Organizations

Para obtener más información sobre AWS RAM, consulte la Guía del usuario de AWS RAM.

En este tema se explica cómo compartir los parámetros que le pertenecen y cómo utilizar los parámetros que se comparten.

Contenido

- Requisitos previos para compartir parámetros
- Compartir un parámetro
- Detención del uso compartido de un parámetro
- Identificación de los parámetros compartidos
- Acceder a los parámetros compartidos
- Conjuntos de permisos para compartir parámetros
- Rendimiento máximo para los parámetros compartidos
- Precios de los parámetros compartidos
- Acceso entre cuentas para Cuentas de AWS cerradas

Requisitos previos para compartir parámetros

Se deben cumplir los siguientes requisitos previos para poder compartir los parámetros de una cuenta:

- Para compartir un parámetro, debe ser el propietario en su Cuenta de AWS. No puede compartir un parámetro que se ha compartido con usted.
- Para compartir un parámetro, debe estar en el nivel de parámetros avanzado. Para obtener información los niveles de los parámetros, consulte Administración de niveles de parámetros. Para obtener información sobre el cambio de un parámetro estándar existente a un parámetro avanzado, consulte Cambio de un parámetro estándar a un parámetro avanzado.
- Para compartir un parámetro de `SecureString`, debe estar cifrado con una clave administrada por el cliente y usted debe compartir la clave por separado a través de AWS Key Management Service. Claves administradas por AWS no se puede compartir. Los parámetros cifrados con el valor predeterminado Clave administrada de AWS se puede actualizar para utilizar una clave

administrada por el cliente en su lugar. Para ver las definiciones de la clave AWS KMS, consulte los Conceptos de AWS KMS en la Guía para desarrolladores de AWS Key Management Service.

- Para compartir un parámetro con la organización o con una unidad organizativa en AWS Organizations, debe habilitar el uso compartido con AWS Organizations. Para obtener más información, consulte Habilitar el uso compartido con AWS Organizations en la Guía del usuario de AWS RAM.

Compartir un parámetro

Para compartir un parámetro, debe agregarlo al recurso compartido. Un uso compartido de recursos es un recurso de AWS RAM que le permite compartir los recursos a través de Cuentas de AWS. Un uso compartido de recursos especifica los recursos que compartir y los consumidores con quienes se comparten.

Al compartir un parámetro propio con otras Cuentas de AWS, se puede elegir entre dos permisos administrados por AWS para concederlos a los consumidores. Para obtener más información, consulte Conjuntos de permisos para compartir parámetros.

Si forma parte de una organización de AWS Organizations y esta permite el uso compartido, puede conceder a los consumidores de la organización acceso desde la consola de AWS RAM al parámetro compartido. De lo contrario, los consumidores reciben una invitación para unirse al recurso compartido y se les concede acceso al parámetro compartido después de aceptar la invitación.

Puede compartir un parámetro proprio con la consola de AWS RAM o AWS CLI.

> **Note**
>
> Si bien puede compartir un parámetro mediante la operación de la API PutResourcePolicy de Systems Manager, se recomienda utilizar AWS Resource Access Manager (AWS RAM) en su lugar. Esto se debe a la utilización de `PutResourcePolicy` requiere el paso adicional de convertir el parámetro en un recurso compartido estándar mediante la operación de API de AWS RAM PromoteResourceShareCreatedFromPolicy. De lo contrario, la operación de la API DescribeParameters de Systems Manager no devolverá el parámetro con la opción `--shared`.

Para compartir un parámetro propio con la consola de AWS RAM

Consulte Creating a resource share in AWS RAM en la Guía del usuario de AWS RAM.

Haga las siguientes selecciones a medida que completa el procedimiento:

- En la página del paso 1, en Recursos, seleccione `Parameter Store Advanced Parameter` y, a continuación, seleccione la casilla de cada parámetro del nivel de parámetros avanzado que desee compartir.
- En la página del paso 2, en Permisos administrados, seleccione el permiso que se va a conceder a los consumidores, tal y como se describe en Conjuntos de permisos para compartir parámetros más adelante en este tema.

Seleccione otras opciones en función de sus objetivos de uso compartido de parámetros.

Para compartir un parámetro propio con AWS CLI

Utilice el comando create-resource-share para agregar parámetros a un recurso compartido nuevo.

Utilice el comando associate-resource-share para agregar parámetros a un recurso compartido existente.

En el siguiente ejemplo, se crea un nuevo recurso compartido para compartir los parámetros con los consumidores de una organización y de una cuenta individual.

```
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"
```

Detención del uso compartido de un parámetro

Cuando deja de compartir un parámetro, la cuenta del consumidor ya no podrá acceder al parámetro.

Para dejar de compartir un parámetro propio, debe quitarlo del recurso compartido. Para ello, puede utilizar la consola de Systems Manager, la consola de AWS RAM o la AWS CLI.

Para dejar de compartir un parámetro propio con la consola de AWS RAM

Consulte Actualización de un recurso compartido en AWS RAM en la Guía del usuario de AWS RAM.

Para dejar de compartir un parámetro propio con AWS CLI

Utilice el comando disassociate-resource-share.

Identificación de los parámetros compartidos

Los propietarios y los consumidores pueden identificar los parámetros compartidos con AWS CLI.

Para identificar los parámetros compartidos con AWS CLI

Para identificar los parámetros compartidos con la AWS CLI, puede elegir entre el comando `describe-parameters` de Systems Manager y el comando de AWS RAM `list-resources`.

Si utiliza la opción `--shared` con `describe-parameters`, el comando devuelve los parámetros que se comparten con usted.

A continuación, se muestra un ejemplo:

```
aws ssm describe-parameters --shared
```

Acceder a los parámetros compartidos

Los consumidores pueden acceder a los parámetros compartidos con las herramientas de línea de comandos de AWS y los SDK de AWS. En el caso de las cuentas de consumidores, los parámetros compartidos con esa cuenta no se incluyen en la página Mis parámetros.

Ejemplo de CLI: acceso a los detalles de los parámetros compartidos con AWS CLI

Para acceder a los detalles de los parámetros compartidos con AWS CLI, puede utilizar los comandos get-parameter o get-parameters. Debe especificar el ARN completo del parámetro como el `--name` para poder recuperar el parámetro de otra cuenta.

A continuación se muestra un ejemplo.

```
aws ssm get-parameter \
    --name arn:aws:ssm:us-east-2:123456789012:parameter/MySharedParameter
```

Integraciones compatibles y no compatibles para los parámetros compartidos

Actualmente, puede utilizar parámetros compartidos en los siguientes escenarios de integración:

- Parámetros de plantilla de AWS CloudFormation
- La extensión de Lambda de Parámetros y secretos de AWS
- Plantillas de lanzamiento de Amazon Elastic Compute Cloud (EC2)

- Valores para `ImageID` con el comando RunInstances de EC2 para crear instancias a partir de una Amazon Machine Image (AMI)
- Recuperación de los valores de los parámetros en manuales de procedimientos para Automatización, una herramienta de Systems Manager

Actualmente, los siguientes escenarios y servicios integrados no son compatibles con el uso de los parámetros compartidos:

- Parámetros en comandos en Run Command, una herramienta de Systems Manager
- Referencias dinámicas de AWS CloudFormation
- Las variables de los valores de entorno en AWS CodeBuild
- Las variables de los valores de entorno en AWS App Runner
- El valor de un secreto en Amazon Elastic Container Service

Conjuntos de permisos para compartir parámetros

Las cuentas de consumidor solo reciben acceso de lectura a los parámetros que compartas con ellas. El consumidor no puede actualizar ni eliminar el parámetro. El consumidor no puede compartir el parámetro con una cuenta de terceros.

Al crear un recurso compartido en AWS Resource Access Manager para compartir sus parámetros, puede elegir entre dos conjuntos de permisos administrados de AWS para concederle acceso de solo lectura:

AWSRAMDefaultPermissionSSMParameterReadOnly

Acciones permitidas: `DescribeParameters`, `GetParameter` y `GetParameters`

AWSRAMPermissionSSMParameterReadOnlyWithHistory

Acciones permitidas: `DescribeParameters`, `GetParameter`, `GetParameters` y `GetParameterHistory`

Cuando siga los pasos que se indican en Creating a resource share in AWS RAM en la Guía del usuario de AWS RAM, seleccione `Parameter Store Advanced Parameters` como tipo de recurso y uno de estos permisos administrados, en función de si desea que los usuarios vean el historial de parámetros o no.

> **Note**
>
> Si va a recuperar parámetros compartidos mediante programación (por ejemplo, mediante AWS Lambda), es posible que tenga que agregar los permisos `ssm:GetResourcePolicies` y `ssm:PutResourcePolicy` a cualquier rol de IAM que llame a acciones de la API de AWS Resource Access Manager.

Rendimiento máximo para los parámetros compartidos

Systems Manager limita el rendimiento máximo (las transacciones por segundo) para las operaciones GetParameter y GetParameters. El rendimiento se aplica a nivel de cuenta individual. Por lo tanto, cada cuenta que consuma un parámetro compartido puede utilizar el rendimiento máximo permitido sin verse afectada por otras cuentas. Para obtener más información sobre el rendimiento máximo de los parámetros, consulte los siguientes temas:

- Aumento del rendimiento de Parameter Store
- Cuotas de servicio de Systems Manager en Referencia general de Amazon Web Services.

Precios de los parámetros compartidos

El uso compartido entre cuentas solo está disponible en el nivel de parámetros avanzados. En el caso de los parámetros avanzados, se cobra al precio actual por el almacenamiento y la utilización de las API por cada parámetro avanzado. Los cargos de almacenamiento del parámetro avanzado se cobran a la cuenta propietaria. A cualquier cuenta consumidora que realice una llamada a la API a un parámetro avanzado compartido se le cobrará por el uso del parámetro.

Por ejemplo, si la cuenta A crea el parámetro avanzado `MyAdvancedParameter`, se le cobrarán 0,05 USD al mes por almacenar el parámetro.

A continuación, la cuenta A comparte el parámetro `MyAdvancedParameter` con la cuenta B y la cuenta C. Durante un mes, las tres cuentas realizan llamadas a `MyAdvancedParameter`. En la siguiente tabla se muestran los cargos en los que incurrirían según el número de llamadas que realice cada cuenta.

> **Note**
>
> Los cargos de la siguiente tabla son solo ilustrativos. Para verificar los precios actuales, consulte Precios de AWS Systems Manager para Parameter Store.

Cuenta	Número de llamadas	Cargos
Cuenta A (cuenta propietaria)	10 000 llamadas	• Almacenamiento avanzado de parámetros durante un mes: 0,05 USD • 10 000 llamadas a `MyAdvancedParameter` : 0,05 USD • Total: 0,10 USD
Cuenta B (cuenta consumidora)	20 000 llamadas	• 20 000 llamadas a `MyAdvancedParameter` : 0,10 USD • Total: 0,10 USD
Cuenta C (cuenta consumidora)	30 000 llamadas	• 30 000 llamadas a `MyAdvancedParameter` : 0,15 USD • Total: 0,15 USD

Acceso entre cuentas para Cuentas de AWS cerradas

Si se cierra la Cuenta de AWS propietaria de un parámetro compartido, todas las cuentas consumidoras pierden el acceso al parámetro compartido. Si la cuenta propietaria se vuelve a abrir dentro de los 90 días siguientes al cierre de la cuenta, las cuentas consumidoras recuperan el acceso a los parámetros compartidos anteriormente. Para obtener más información sobre la reapertura de una cuenta durante el periodo posterior al cierre, consulte Acceso a la Cuenta de AWS después de cerrarla en la Guía de referencia de AWS Account Management.

Trabajo con parámetros en Parameter Store con el uso de comandos de Run Command

Puede trabajar con parámetros en Run Command, una herramienta de AWS Systems Manager. Para obtener más información, consulte AWS Systems Manager Run Command.

Utilizar la consola para ejecutar un parámetro String

El siguiente procedimiento presenta el proceso de ejecución de un comando que utiliza un parámetro `String`.

Para ejecutar un parámetro de cadena mediante Parameter Store

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command.
3. Elija Run command (Ejecutar comando).
4. En la lista Command document (Documento de comandos), elija `AWS-RunPowerShellScript` (Windows) o `AWS-RunShellScript` (Linux).
5. En Command parameters (Parámetros de comando), introduzca **`echo {{ssm:`*`parameter-name`*`}}`**. Por ejemplo: **`echo {{ssm:/Test/helloWorld}}`**.
6. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

   > **Tip**
   >
   > Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

7. En Otros parámetros:
   - En Comentario, ingrese la información acerca de este comando.
   - En Tiempo de espera (segundos), especifique el número de segundos que tiene que esperar el sistema antes de indicar que se ha producido un error en la ejecución del comando general.
8. En Rate control (Control de velocidad):

- En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

> **Note**
>
> Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

- En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

9. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

10. En la sección Notificaciones de SNS, seleccione la casilla de verificación Habilitar notificaciones de SNS si desea recibir notificaciones sobre el estado de ejecución de los comandos.

    Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.

11. Seleccione Ejecutar.
12. En la página Command ID (ID de comando), en el área Targets and outputs (Destinos y salidas), seleccione el botón junto al ID de un nodo donde ejecutó el comando y, a continuación, elija View output (Ver salida). Compruebe que el resultado del comando sea el valor proporcionado para el parámetro, como **`This is my first parameter`**.

Utilizar la AWS CLI para ejecutar un parámetro

Ejemplo 1: Comando simple

El siguiente comando de ejemplo incluye un parámetro de Systems Manager denominado `DNS-IP`. El valor de este parámetro es simplemente la dirección IP de un nodo. En este ejemplo se utiliza un comando de AWS Command Line Interface (AWS CLI) para reflejar el valor del parámetro.

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
```

El comando devuelve información similar a la siguiente.

```
{
    "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": []
    }
}
```

Una vez finalizada la ejecución de un comando, podrá ver más información sobre él mediante los siguientes comandos:

- get-command-invocation: vea información detallada sobre la ejecución del comando.
- list-command-invocations: vea el estado de ejecución del comando en un nodo administrado específico.
- list-commands: vea el estado de ejecución de los comandos en los nodos administrados.

Ejemplo 2: Descifrar el valor del parámetro **`SecureString`**

El siguiente comando de ejemplo utiliza un `SecureString` parámetro denominado SecurePassword. El comando utilizado en el campo `parameters` recupera y descifra el valor del parámetro `SecureString` y, a continuación, restablece la contraseña de administrador local sin tener que transferir la contraseña en texto sin cifrar.

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
```

Ejemplo 3: Hacer referencia a un parámetro en un documento SSM

También puede hacer referencia a los parámetros de Systems Manager en la sección Parameters (Parámetros) de un documento de SSM, tal y como se muestra en el siguiente ejemplo.

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

No confunda la sintaxis similar para los parámetros locales utilizados en la sección `runtimeConfig` de documentos de SSM con parámetros de Parameter Store. Un parámetro local no es lo mismo que

un parámetro de Systems Manager. Puede diferenciar los parámetros locales de los parámetros de Systems Manager por la ausencia del prefijo `ssm:`.

```
"runtimeConfig":{
        "aws:runShellScript":{
            "properties":[
                {
                    "id":"0.aws:runShellScript",
                    "runCommand":"{{ commands }}",
                    "workingDirectory":"{{ workingDirectory }}",
                    "timeoutSeconds":"{{ executionTimeout }}"
```

Note

Los documentos de SSM no admiten referencias a los parámetros de `SecureString`. Esto significa que, para utilizar los parámetros `SecureString` con, por ejemplo, Run Command, tendrá que recuperar el valor de parámetro antes de pasarlo a Run Command, tal y como se muestra en los siguientes ejemplos.

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
```

Uso de compatibilidad con parámetros nativos en Parameter Store para los ID de Amazon Machine Image

Cuando cree un parámetro `String`, puede especificar un tipo de datos como `aws:ec2:image` para asegurarse de que el valor de parámetro que especifique tenga formato de ID de Amazon Machine Image (AMI) válido.

La compatibilidad con los formatos de ID de AMI le exime de tener que actualizar todos los scripts y las plantillas con un nuevo ID cada vez que cambie la AMI que desea utilizar en sus procesos. Puede crear un parámetro con el tipo de datos `aws:ec2:image`, y para su valor, especifique el ID de una AMI. Esta es la AMI desde la que desea crear nuevas instancias. A continuación, haga referencia a este parámetro en sus plantillas, comandos y scripts.

Por ejemplo, puede especificar el parámetro que contiene su ID de AMI preferido cuando ejecuta el comando `run-instances` de Amazon Elastic Compute Cloud (Amazon EC2).

> **Note**
>
> El usuario que ejecute este comando debe tener permisos de AWS Identity and Access Management (IAM) que incluyan la operación de la API `ssm:GetParameters` para validar el valor del parámetro. De lo contrario, el proceso de creación de parámetros producirá un error.

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
```

También puede elegir la AMI de su elección cuando cree una instancia mediante la consola de Amazon EC2. Para obtener más información, consulte Uso de un parámetro de Systems Manager para encontrar una AMI en la Guía del usuario de Amazon EC2.

Cuando llegue el momento de utilizar una AMI diferente en el flujo de trabajo de creación de instancias, solo necesita actualizar el parámetro con el nuevo valor de AMI y Parameter Store validará que ha especificado un ID en el formato adecuado.

Conceder permisos para crear un parámetro de tipo de datos **`aws:ec2:image`**

Mediante las políticas de AWS Identity and Access Management (IAM), puede proporcionar o restringir el acceso de los usuarios a las operaciones y el contenido de la API de Parameter Store.

Para crear un parámetro de tipo de datos `aws:ec2:image`, el usuario debe tener los permisos `ssm:PutParameter` y `ec2:DescribeImages`.

En la siguiente política de ejemplo, se concede permiso de usuario para llamar a la operación `PutParameter` de la API para `aws:ec2:image`. Esto significa que el usuario puede agregar un parámetro de tipo de datos `aws:ec2:image` al sistema.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ssm:PutParameter",
            "Resource": "*"
        },
        {
```

```
            "Effect": "Allow",
            "Action": "ec2:DescribeImages",
            "Resource": "*"
        }
    ]
}
```

Funcionamiento de la validación del formato de la AMI

Cuando se especifica `aws:ec2:image` como tipo de datos para un parámetro, Systems Manager no crea el parámetro inmediatamente. En su lugar, realiza una operación de validación asíncrona para garantizar que el valor del parámetro cumple los requisitos de formato para un ID de AMI y que la AMI especificada esté disponible en su Cuenta de AWS.

Se puede generar un número de versión de parámetro antes de que se complete la operación de validación. La operación puede no estar completa incluso si el número de versión de parámetro está generado.

Para monitorear si los parámetros se crearon correctamente, le recomendamos que utilice Amazon EventBridge para que le envíe notificaciones sobre las operaciones de parámetros `create` y `update`. Estas notificaciones indican si una operación de parámetro se ha realizado correctamente o no. Si una operación falla, la notificación incluye un mensaje de error que indica el motivo del error.

```
{
    "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"
    }
```

```
}
```

Para obtener información acerca de cómo suscribirse a estos eventos de Parameter Store en EventBridge, consulte Configuración de notificaciones o activación de acciones en función de los eventos de Parameter Store.

Eliminación de parámetros de Parameter Store

En este tema se describe cómo se eliminan los parámetros que ha creado en el Parameter Store, una herramienta de AWS Systems Manager.

> **Warning**
>
> Al eliminar un parámetro, se eliminan todas sus versiones. Después de eliminarlo, no se pueden restaurar ni el parámetro ni sus versiones.

Para eliminar un parámetro con la consola

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Parameter Store.
3. En la pestaña My parameters (Mis parámetros), seleccione la casilla de verificación situada junto a cada parámetro que desee eliminar.
4. Elija Eliminar.
5. En el cuadro de diálogo de confirmación, elija Delete parameters (Eliminar parámetros).

Para eliminar un parámetro con la AWS CLI

- Ejecuta el siguiente comando:

  ```
  aws ssm delete-parameter --name "my-parameter"
  ```

  Sustituya *my-parameter* por el nombre del parámetro que desea eliminar.

  Para obtener más información acerca de otras opciones que puede utilizar con el comando `delete-parameter`, consulte delete-parameter en la AWS Systems Manager sección de la AWS CLI Referencia de comandos.

Trabajo con parámetros públicos en Parameter Store

Algunos Servicios de AWS publican información sobre artefactos comunes como AWS Systems Manager públicos. Por ejemplo, el servicio Amazon Elastic Compute Cloud (Amazon EC2) publica información sobre Amazon Machine Images (AMIs) como parámetros públicos.

Temas de esta guía

- Descubrir parámetros públicos en Parameter Store
- Llamada a parámetros públicos de AMI en Parameter Store
- Llamada al parámetro público de la AMI optimizada de ECS en Parameter Store
- Llamada al parámetro público de la AMI optimizada de EKS en Parameter Store
- Llamar a parámetros públicos para Servicios de AWS, regiones, puntos de conexión, zonas de disponibilidad, zonas locales y zonas de Wavelength en Parameter Store

Entradas de blog de AWS relacionadas

- Query for Regiones de AWS, Endpoints, and More Using 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

Descubrir parámetros públicos en Parameter Store

Puede buscar parámetros públicos mediante la consola de Parameter Store o AWS Command Line Interface.

Un nombre de parámetro público comienza con `aws/service/list`. La siguiente parte del nombre corresponde al servicio que posee ese parámetro.

A continuación se muestra una lista parcial de Servicios de AWS y otros recursos que proporcionen parámetros públicos:

- `ami-amazon-linux-latest`
- `ami-windows-latest`
- `ec2-macos`
- `appmesh`
- `aws-for-fluent-bit`

- aws-sdk-pandas
- bottlerocket
- canonical
- cloud9
- datasync
- debian
- deeplearning
- ecs
- eks
- fis
- freebsd
- global-infrastructure
- marketplace
- neuron
- powertools
- sagemaker-distribution
- storagegateway

No todos los parámetros públicos se publican en cada Región de AWS.

Encontrar parámetros públicos mediante la consola de Parameter Store

Debe tener al menos un parámetro en su Cuenta de AWS y Región de AWS antes de que pueda buscar parámetros públicos mediante la consola.

Para encontrar parámetros públicos mediante la consola.

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Parameter Store.
3. Elija la pestaña Public parameters (Parámetros públicos).
4. Elija el menú desplegable Select a service (Seleccionar un servicio). Elija el servicio cuyos parámetros desea utilizar.

5. (Opcional) Para filtrar los parámetros que pertenecen al servicio seleccionado, ingrese más información en la barra de búsqueda.
6. Elija el parámetro público que desea utilizar.

Encontrar parámetros públicos mediante el comando AWS CLI

Utilizar `describe-parameters` para el descubrimiento de parámetros públicos.

Utilizar `get-parameters-by-path` para obtener la ruta real de un servicio enumerado en `/aws/service/list`. Para obtener la ruta del servicio, elimine `/list` de la ruta. Por ejemplo, `/aws/service/list/ecs` se convierte en `/aws/service/ecs`.

Para recuperar una lista de parámetros públicos propiedad de diferentes servicios en Parameter Store, ejecute el siguiente comando.

```
aws ssm get-parameters-by-path --path /aws/service/list
```

El comando devuelve información similar a la siguiente. Este ejemplo se ha truncado por falta de espacio.

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

Si desea ver los parámetros que pertenecen a un servicio específico, elija el servicio de la lista que se produjo después de ejecutar el comando anterior. A continuación, haga una llamada `get-parameters-by-path` utilizando el nombre del servicio deseado.

Por ejemplo, `/aws/service/global-infrastructure`. La ruta puede ser de un nivel (solo llama a parámetros que coincidan con los valores exactos dados) o recursivo (contiene elementos en la ruta más allá de lo que ha dado).

> **Note**
>
> La ruta `/aws/service/global-infrastructure` no se admite en las consultas en todas las regiones. Para obtener más información, consulta Llamar a parámetros públicos para Servicios de AWS, regiones, puntos de conexión, zonas de disponibilidad, zonas locales y zonas de Wavelength en Parameter Store.

Si no se devuelve ningún resultado para el servicio que especifique, agregue la marca `--recursive` y vuelva a ejecutar el comando.

```
aws ssm get-parameters-by-path --path /aws/service/global-infrastructure
```

Esto devuelve todos los parámetros propiedad de `global-infrastructure`. A continuación se muestra un ejemplo.

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

También puede ver los parámetros que pertenecen a un servicio específico mediante el filtro `Option:BeginsWith`.

```
aws ssm describe-parameters --parameter-filters "Key=Name, Option=BeginsWith, Values=/
aws/service/ami-amazon-linux-latest"
```

El comando devuelve información similar a la siguiente. Este ejemplo de resultado se ha truncado por falta de espacio.

```
{
    "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**
>
> Los parámetros devueltos pueden ser diferentes cuando se utiliza `Option=BeginsWith` porque utiliza un patrón de búsqueda diferente.

Llamada a parámetros públicos de AMI en Parameter Store

Los parámetros públicos de la Amazon Machine Image (AMI) de Amazon Elastic Compute Cloud (Amazon EC2) están disponibles para Amazon Linux 1, Amazon Linux 2, Amazon Linux 2023 (AL2023), macOS y Windows Server en las siguientes rutas:

- Amazon Linux 1, Amazon Linux 2 y Amazon Linux 2023: `/aws/service/ami-amazon-linux-latest`

- macOS: `/aws/service/ec2-macos`
- Windows Server: `/aws/service/ami-windows-latest`

Llamada a parámetros públicos de AMI para Amazon Linux 1, Amazon Linux 2 y Amazon Linux 2023

Puede ver una lista de todas las AMIs de Amazon Linux 1, Amazon Linux 2 y Amazon Linux 2023 (AL2023) en la versión actual de Región de AWS mediante el uso del siguiente comando en la 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
```

El comando devuelve información similar a la siguiente.

```
[
    "/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"
]
```

Puede ver información detallada acerca de estas AMIs, incluidos los ID de AMI y los nombres de recursos de Amazon (ARN), mediante el siguiente comando.

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
```

region representa el identificador de Región de AWS compatible con AWS Systems Manager, como `us-east-2` para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

El comando devuelve información similar a la siguiente. Este ejemplo de resultado se ha truncado por falta de espacio.

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

Puede ver los detalles de una AMI específica mediante la operación de la API GetParameters con el nombre completo de la AMI, incluida la ruta. A continuación se muestra un comando de ejemplo.

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
```

El comando devuelve la siguiente información.

```
{
    "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": []
}
```

Llamada a parámetros públicos de AMI para macOS

Puede ver una lista de todas las AMIs de macOS en la Región de AWS actual mediante el siguiente comando en la AWS CLI.

Linux & macOS

```
aws ssm get-parameters-by-path \
    --path /aws/service/ec2-macos\
    --query 'Parameters[].Name'
```

Windows

```
aws ssm get-parameters-by-path ^
    --path /aws/service/ec2-macos ^
    --query Parameters[].Name
```

El comando devuelve información similar a la siguiente.

```
[
    "/aws/service/ec2-macos/sonoma/x86_64_mac/latest/image_id",
    "/aws/service/ec2-macos/ventura/x86_64_mac/latest/image_id",
    "/aws/service/ec2-macos/monterey/x86_64_mac/latest/image_id",
```

```
    "/aws/service/ec2-macos/sonoma/arm64_mac/latest/image_id",
    "/aws/service/ec2-macos/ventura/arm64_mac/latest/image_id",
    "/aws/service/ec2-macos/monterey/arm64_mac/latest/image_id"
]
```

Puede ver información detallada acerca de estas AMIs, incluidos los ID de AMI y los nombres de recursos de Amazon (ARN), mediante el siguiente comando.

Linux & macOS

```
aws ssm get-parameters-by-path \
    --path "/aws/service/ec2-macos" \
    --region region
```

Windows

```
aws ssm get-parameters-by-path ^
    --path "/aws/service/ec2-macos" ^
    --region region
```

region representa el identificador de Región de AWS compatible con AWS Systems Manager, como `us-east-2` para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

El comando devuelve información similar a la siguiente. Este ejemplo de resultado se ha truncado por falta de espacio.

```
{
    "Parameters": [
        ...sample results pending...
    ]
}
```

Puede ver los detalles de una AMI específica mediante la operación de la API GetParameters con el nombre completo de la AMI, incluida la ruta. A continuación se muestra un comando de ejemplo.

Linux & macOS

```
aws ssm get-parameters \
```

```
    --names /aws/service/ec2-macos/...pending... \
    --region us-east-2
```

Windows

```
aws ssm get-parameters ^
     --names /aws/service/ec2-macos/...pending... ^
    --region us-east-2
```

El comando devuelve la siguiente información.

```
{
    "Parameters": [
         ...sample results pending...
    ],
    "InvalidParameters": []
}
```

Llamada a parámetros públicos de AMI para Windows Server

Puede ver una lista de todas las AMIs de Windows Server en la Región de AWS actual mediante el siguiente comando en la 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
```

El comando devuelve información similar a la siguiente. Este ejemplo de resultado se ha truncado por falta de espacio.

```
[
```

```
    "/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
]
```

Puede ver información detallada acerca de estas AMIs, incluidos los ID de AMI y los nombres de recursos de Amazon (ARN), mediante el siguiente comando.

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
```

region representa el identificador de Región de AWS compatible con AWS Systems Manager, como `us-east-2` para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

El comando devuelve información similar a la siguiente. Este ejemplo de resultado se ha truncado por falta de espacio.

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

Puede ver los detalles de una AMI específica mediante la operación de la API GetParameters con el nombre completo de la AMI, incluida la ruta. A continuación se muestra un comando de ejemplo.

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
```

El comando devuelve la siguiente información.

```
{
    "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": []
}
```

Llamada al parámetro público de la AMI optimizada de ECS en Parameter Store

El servicio de Amazon Elastic Container Service (Amazon ECS) publica el nombre de la última Amazon Machine Images (AMIs) optimizada de Amazon ECS como parámetro público. Se recomienda a los usuarios que utilicen esta AMI al crear un clúster nuevo de Amazon Elastic Compute Cloud (Amazon EC2) para Amazon ECS, ya que la AMIs optimizada incluye correcciones de errores y nuevas características.

Utilice el siguiente comando para ver el nombre de la última AMI optimizada de Amazon ECS para Amazon Linux 2. Para ver comandos de otros sistemas operativos, consulte Recuperación de metadatos de AMI de Amazon ECS Optimizado en la Guía para desarrolladores de 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
```

El comando devuelve información similar a la siguiente.

```
{
```

```
    "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": []
}
```

Llamada al parámetro público de la AMI optimizada de EKS en Parameter Store

El servicio de Amazon Elastic Kubernetes Service (Amazon EKS) publica el nombre de la última Amazon Machine Image (AMI) optimizada de Amazon EKS como parámetro público. Le recomendamos que utilice esta AMI cuando agregue nodos a un clúster de Amazon EKS, ya que las nuevas versiones incluyen parches de Kubernetes y actualizaciones de seguridad. Anteriormente, para garantizar de que estaba utilizando la última AMI debía verificar la documentación de Amazon EKS y actualizar manualmente cualquier plantilla o recurso de implementación con el nuevo ID de la AMI.

Utilice el siguiente comando para ver el nombre de la última AMI optimizada de Amazon EKS para 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
```

El comando devuelve información similar a la siguiente.

```
{
    "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": []
}
```

Llamar a parámetros públicos para Servicios de AWS, regiones, puntos de conexión, zonas de disponibilidad, zonas locales y zonas de Wavelength en Parameter Store

Puede llamar a la Región de AWS, punto de enlace de servicio y zonas de disponibilidad y de Wavelength de parámetros públicos mediante la siguiente ruta.

`/aws/service/global-infrastructure`

> **Note**
>
> Actualmente, la ruta `/aws/service/global-infrastructure` solo se admite para consultas en las siguientes Regiones de AWS:
>
> - Este de EE. UU. (Norte de Virginia) (us-east-1)
> - Este de EE. UU. (Ohio) (us-east-2)
> - EE. UU. Oeste (Norte de California) (us-west-1)
> - Oeste de EE. UU. (Oregón) (us-west-2)
> - Asia-Pacífico (Hong Kong) (ap-east-1)
> - Asia Pacífico (Bombay) (ap-south-1)
> - Asia-Pacífico (Seúl) (ap-northeast-2)

- Asia-Pacífico (Singapur) (ap-southeast-1)
- Asia-Pacífico (Sídney) (ap-southeast-2)
- Asia-Pacífico (Tokio) (ap-northeast-1)
- Canadá (centro) (ca-central-1)
- Europa (Fráncfort) (eu-central-1)
- Europa (Irlanda) (eu-west-1)
- Europa (Londres) (eu-west-2)
- UE (París) (eu-west-3)
- Europa (Estocolmo) (eu-north-1)
- América del Sur (São Paulo) (sa-east-1)

Si trabaja en una región comercial diferente, puede especificar una región admitida en la consulta para ver resultados. Por ejemplo, si trabaja en la región Oeste de Canadá (Calgary) (ca-west-1), puede especificar Canadá (centro) (ca-central-1) en la consulta:

```
aws ssm get-parameters-by-path \
    --path /aws/service/global-infrastructure/regions \
    --region ca-central-1
```

Ver Regiones de AWS activas

Puede ver una lista de todas las Regiones de AWS activas de mediante el siguiente comando en la 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
```

El comando devuelve información similar a la siguiente.

```
[
    "/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"
]
```

Ver Servicios de AWS disponibles

Puede ver una lista completa de todos los Servicios de AWS disponibles y ordenarlos en orden alfabético mediante el siguiente comando. Este ejemplo de resultado se ha truncado por falta de espacio.

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(@)"
```

El comando devuelve información similar a la siguiente. Este ejemplo se ha truncado por falta de espacio.

```
[
     "/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",
```

Ver las regiones admitidas para un Servicio de AWS

Puede ver una lista de Regiones de AWS donde un servicio está disponible. En este ejemplo se utiliza 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
```

El comando devuelve información similar a la siguiente.

```
[
    "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"
]
```

Ver el punto de enlace regional de un servicio

Puede ver un punto de enlace regional de un servicio mediante el siguiente comando. Este comando consulta la región Este de EE. UU. (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
```

El comando devuelve información similar a la siguiente.

```
"ssm.us-east-2.amazonaws.com"
```

Ver detalles completos de la zona de disponibilidad

Puede ver las zonas de disponibilidad mediante el siguiente comando.

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/
```

El comando devuelve información similar a la siguiente. Este ejemplo se ha truncado por falta de espacio.

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

Ver solo los nombres de las zonas de disponibilidad

Puede ver solo los nombres de las zonas de disponibilidad mediante el siguiente comando.

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(@)"
```

El comando devuelve información similar a la siguiente. Este ejemplo se ha truncado por falta de espacio.

```
[
    "/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"
```

Ver los nombres de las zonas de disponibilidad de una sola región

Puede ver los nombres de las zonas de disponibilidad de una región (`us-east-2`, en este ejemplo) mediante el siguiente comando.

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(@)"
```

El comando devuelve información similar a la siguiente.

```
[
    "/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"
```

Ver solo los ARN de zona de disponibilidad

Puede ver solo los Nombres de recurso de Amazon (ARN) de las zonas de disponibilidad mediante el siguiente comando.

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(@)"
```

El comando devuelve información similar a la siguiente. Este ejemplo se ha truncado por falta de espacio.

```
[
    "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",
```

Ver detalles de la zona local

Puede ver las zonas locales mediante el siguiente comando.

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
```

El comando devuelve información similar a la siguiente. Este ejemplo se ha truncado por falta de espacio.

```
{
```

```
    "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"
        }
    ]
}
```

Ver detalles de la zona Wavelength

Puede ver las zonas Wavelength mediante el siguiente comando.

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
```

El comando devuelve información similar a la siguiente. Este ejemplo se ha truncado por falta de espacio.


```
{
    "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"
```

```
        }
    ]
}
```

Ver todos los parámetros y valores de una zona local

Puede ver todos los datos de parámetros de una zona local mediante el siguiente comando.

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"
```

El comando devuelve información similar a la siguiente. Este ejemplo se ha truncado por falta de espacio.

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

Ver solo los nombres de parámetros de la zona local

Puede ver solo los nombres de los parámetros de la zona local mediante el siguiente comando.

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(@)"
```

El comando devuelve información similar a la siguiente.

```
[
    "/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"
]
```

Tutoriales de Parameter Store

La explicación de esta sección muestra cómo crear, almacenar y ejecutar parámetros con Parameter Store, una herramienta de AWS Systems Manager, en un entorno de pruebas. Esta explicación muestra cómo utilizar Parameter Store con otras herramientas de Systems Manager. También puede utilizar Parameter Store con otros Servicios de AWS. Para obtener más información, consulte ¿Qué es un parámetro?.

Contenido

- Cómo crear un parámetro SecureString en Parameter Store y unir un nodo con un dominio (PowerShell)

Cómo crear un parámetro SecureString en Parameter Store y unir un nodo con un dominio (PowerShell)

Esta explicación muestra cómo unir un nodo Windows Server a un dominio mediante parámetros AWS Systems Manager `SecureString` y Run Command. El tutorial utiliza parámetros de dominio típicos, como el nombre de dominio y un nombre de usuario del dominio. Estos valores se transfieren como valores de cadena no cifrados. La contraseña del dominio se cifra con una Clave administrada de AWS y se transfiere como una cadena segura.

Requisitos previos

En este tutorial se supone que ya ha especificado el nombre de dominio y la dirección IP del servidor DNS en el conjunto de opciones de DHCP asociadas a su Amazon VPC. Para obtener más información, consulte Trabajar con conjuntos de opciones DHCP en la Guía del usuario de Amazon VPC.

Cómo crear un parámetro **`SecureString`** y unir un nodo a un dominio

1. Escriba parámetros en el sistema utilizando AWS Tools for Windows PowerShell.

   En el siguiente ejemplo, reemplace cada *`marcador de posición del usuario`* con su propia información.

```
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**
>
> Solo se cifra el valor de un parámetro `SecureString`. Los nombres de parámetros, las descripciones y otras propiedades no se cifran.

2. Adjunte las siguientes políticas de AWS Identity and Access Management (IAM) a los permisos de rol de IAM para el nodo:

   - AmazonSSMManagedInstanceCore: obligatorio. Esta política administrada por AWS permite que un nodo administrado utilice la funcionalidad básica del servicio Systems Manager.
   - AmazonSSMDirectoryServiceAccess: obligatorio. Esta política administrada de AWS permite a SSM Agent acceder a AWS Directory Service en su nombre para las solicitudes de unión al dominio por parte del nodo administrado.
   - Una política personalizada para el acceso al bucket de S3: obligatorio. SSM Agent, ubicado en su nodo y que desempeña las tareas de Systems Manager, requiere acceso a buckets de Amazon específicos de Amazon Simple Storage Service (Amazon S3). En la política de bucket de S3 personalizada que crea, también ofrece acceso a los buckets de S3 de su propiedad que sean necesarios para las operaciones de Systems Manager.

     Ejemplos: puede escribir resultados para los comandos de Run Command o sesiones de Session Manager en un bucket de S3 y utilizar este resultado más tarde para tareas de auditoría o resolución de problemas. Puede almacenar scripts de acceso o listas de línea de base de revisiones personalizadas en un bucket de S3 y, a continuación, consultar el script o la lista cuando ejecute un comando o cuando aplique la línea de base de revisiones.

     Para obtener más información acerca de la creación de políticas personalizadas para el acceso al bucket de Amazon S3, consulte Crear una política de bucket de S3 personalizada para un perfil de instancia

     > **Note**
     >
     > Guardar los datos del registro de salida en un bucket de S3 es opcional, pero recomendamos que configure esta opción al principio de su proceso de configuración

de Systems Manager si ha decidido utilizarlo. Para obtener más información, consulte Create a Bucket (Creación de un bucket) en la Guía del usuario de Amazon Simple Storage Service.

- CloudWatchAgentServerPolicy: opcional. Esta política administrada de AWS le permite ejecutar el agente de CloudWatch en nodos administrados. Esta política permite la lectura de información en un nodo y su escritura en Amazon CloudWatch. Su perfil de instancias solo precisa de esta política si hace uso de servicios, como Amazon EventBridge o Registros de CloudWatch.

> **Note**
>
> El uso de características de CloudWatch y EventBridge es opcional, pero recomendamos que las configure al principio de su proceso de configuración de Systems Manager si ha decidido usarlas. Para obtener más información, consulte la Guía del usuario de Amazon EventBridge y la Guía del usuario de Registros de Amazon CloudWatch.

3. Edite el rol de IAM adjunto al nodo y agregue la siguiente política. Esta política concede al nodo los permisos para llamar a `kms:Decrypt` y a `ssm:CreateDocument` de la API.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "ssm:CreateDocument"
            ],
            "Resource": [
                "arn:aws:kms:us-east-1:111122223333:key/kms-key-id"
            ]
        }
    ]
}
```

4. Copie y pegue el siguiente texto de json en un editor de texto y guarde el archivo como `JoinInstanceToDomain.json` en la siguiente ubicación: `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. Ejecute el siguiente comando en Tools for Windows PowerShell para crear un nuevo documento de SSM.

```
$json = Get-Content C:\temp\JoinInstanceToDomain | Out-String
New-SSMDocument -Name JoinInstanceToDomain -Content $json -DocumentType Command
```

6. Ejecute el siguiente comando en Tools for Windows PowerShell para unir el nodo con el dominio.

```
Send-SSMCommand -InstanceId instance-id -DocumentName JoinInstanceToDomain
```

Si el comando es exitoso, este devuelve información similar a la siguiente.

```
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 el comando falla, este devuelve información similar a la siguiente.

```
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.
```

Auditoría y registro de la actividad de Parameter Store

AWS CloudTrail captura las llamadas a la API realizadas en la consola de AWS Systems Manager, la AWS Command Line Interface (AWS CLI) y el SDK de Systems Manager. Puede ver la información en la consola de CloudTrail o en un bucket de Amazon Simple Storage Service (Amazon S3). Todos los registros de CloudTrail de su cuenta utilizan un bucket. Para obtener más información acerca de cómo ver y utilizar los registros de CloudTrail de la actividad de Systems Manager, consulte Registro de llamadas a la API de AWS Systems Manager con AWS CloudTrail. Para obtener más información acerca de cómo auditar y registrar opciones de Systems Manager, consulte Registro y monitorización en AWS Systems Manager.

Solución de problemas de Parameter Store

Utilice la siguiente información como ayuda para solucionar problemas con Parameter Store, una herramienta de AWS Systems Manager.

Solución de problemas de creación de parámetros de **aws:ec2:image**

Utilice la siguiente información para ayudar a solucionar problemas con la creación de parámetros de tipo de datos `aws:ec2:image`.

Sin permiso para crear una instancia

Problema: intenta crear una instancia con un parámetro `aws:ec2:image`, pero recibe un error como “No está autorizado a realizar esta operación”.

- Solución: no tiene todos los permisos necesarios para crear una instancia de EC2 con un valor de parámetro, como los permisos para `ec2:RunInstances`, `ec2:DescribeImages` y `ssm:GetParameter`, entre otros. Contáctese con un usuario con permisos de administrador de la organización para solicitar los permisos necesarios.

EventBridge envía el mensaje de error “Unable to Describe Resource” (No se puede describir el recursos).

Problema: ha ejecutado un comando para crear un parámetro `aws:ec2:image`, pero la operación de creación del parámetro produjo un error. Recibe una notificación de Amazon EventBridge que le informa de la excepción “Unable to Describe Resource” (No se puede describir el recursos).

Solución: este mensaje puede indicar lo siguiente:

- No tiene todos los permisos necesarios para la operación de la API `ec2:DescribeImages` o no tiene permiso para acceder a la imagen específica a la que se hace referencia en el parámetro. Póngase en contacto con un usuario con permisos de administrador de su organización para solicitar los permisos necesarios.
- El ID de Amazon Machine Image (AMI) que ingresó como valor de parámetro no es válido. Asegúrese de que está ingresando el ID de una AMI disponible en la Región de AWS actual y cuenta en la que está trabajando.

El nuevo parámetro **`aws:ec2:image`** no está disponible

Problema: acaba de ejecutar un comando para crear un parámetro `aws:ec2:image` y se indica el número de versión, pero el parámetro no está disponible.

- Solución: cuando ejecuta el comando para crear un parámetro que utiliza el tipo de datos `aws:ec2:image`, se genera inmediatamente un número de versión para el parámetro, pero el formato del parámetro debe validarse antes de que el parámetro esté disponible. Este proceso puede tardar unos minutos. Para monitorizar el proceso de creación y validación de parámetros, puede hacer lo siguiente:
  - Utilice EventBridge para que le envíe notificaciones sobre las operaciones de `create` y `update` de parámetros. Estas notificaciones indican si una operación de parámetro se ha realizado correctamente o no. Para obtener información acerca de cómo suscribirse a estos eventos de Parameter Store en EventBridge, consulte Configuración de notificaciones o activación de acciones en función de los eventos de Parameter Store.
  - En la sección Parameter Store (Almacén de parámetros) de la consola de Systems Manager, actualice periódicamente la lista de parámetros para buscar los detalles de los parámetros nuevos o actualizados.
  - Utilice el comando GetParameter para comprobar el parámetro nuevo o actualizado. Por ejemplo, con la AWS Command Line Interface (AWS CLI):

    ```
    aws ssm get-parameter name MyParameter
    ```

    Para un parámetro nuevo, se devuelve un mensaje `ParameterNotFound` hasta que se valida el parámetro. Para un parámetro existente que está actualizando, la información sobre la nueva versión no se incluye hasta que se valida el parámetro.

  Si intenta crear o actualizar el parámetro de nuevo antes de que finalice el proceso de validación, el sistema informa de que la validación sigue en curso. Si el parámetro no se ha creado o actualizado, puede intentarlo de nuevo después de transcurridos 5 minutos desde el intento original.

Herramientas de operaciones de AWS Systems Manager

Herramientas de operaciones es un conjunto de capacidades que lo ayuda a administrar los recursos de AWS.

- Los paneles de Amazon CloudWatch alojados en Systems Manager son páginas de inicio personalizables en la consola de CloudWatch que puede utilizar para monitorear sus recursos en Regiones de AWS en una única vista.
- Explorer es un panel de operaciones personalizable que transmite información sobre sus recursos de AWS. Explorer muestra una vista agregada de los datos de operaciones (OpsData) de sus Cuentas de AWS y en todas las Regiones de AWS.
- Incident Manager le ayuda a mitigar incidentes que afecten a sus aplicaciones alojadas en AWS y a recuperarse de estos.
- OpsCenter, proporciona una ubicación central donde los ingenieros de operaciones y profesionales de TI pueden ver, investigar y resolver los elementos de trabajo operativos (OpsItems) relacionados con los recursos de AWS.

Temas

- Administrador de incidentes de AWS Systems Manager
- AWS Systems Manager Explorer
- AWS Systems Manager OpsCenter
- Uso de paneles de Amazon CloudWatch alojados por Systems Manager

Administrador de incidentes de AWS Systems Manager

Utilice Administrador de incidentes, una herramienta de AWS Systems Manager, para administrar los incidentes que se produzcan en las aplicaciones alojadas por AWS. Incident Manager combina interacciones de usuarios, escaladas, manuales de procedimientos, planes de respuesta, canales de conversación y análisis posteriores a incidentes para ayudar a su equipo a clasificar los incidentes con mayor rapidez y a regresar las aplicaciones a la normalidad. Para obtener más información acerca de Incident Manager, consulte la Guía del usuario de Incident Manager.

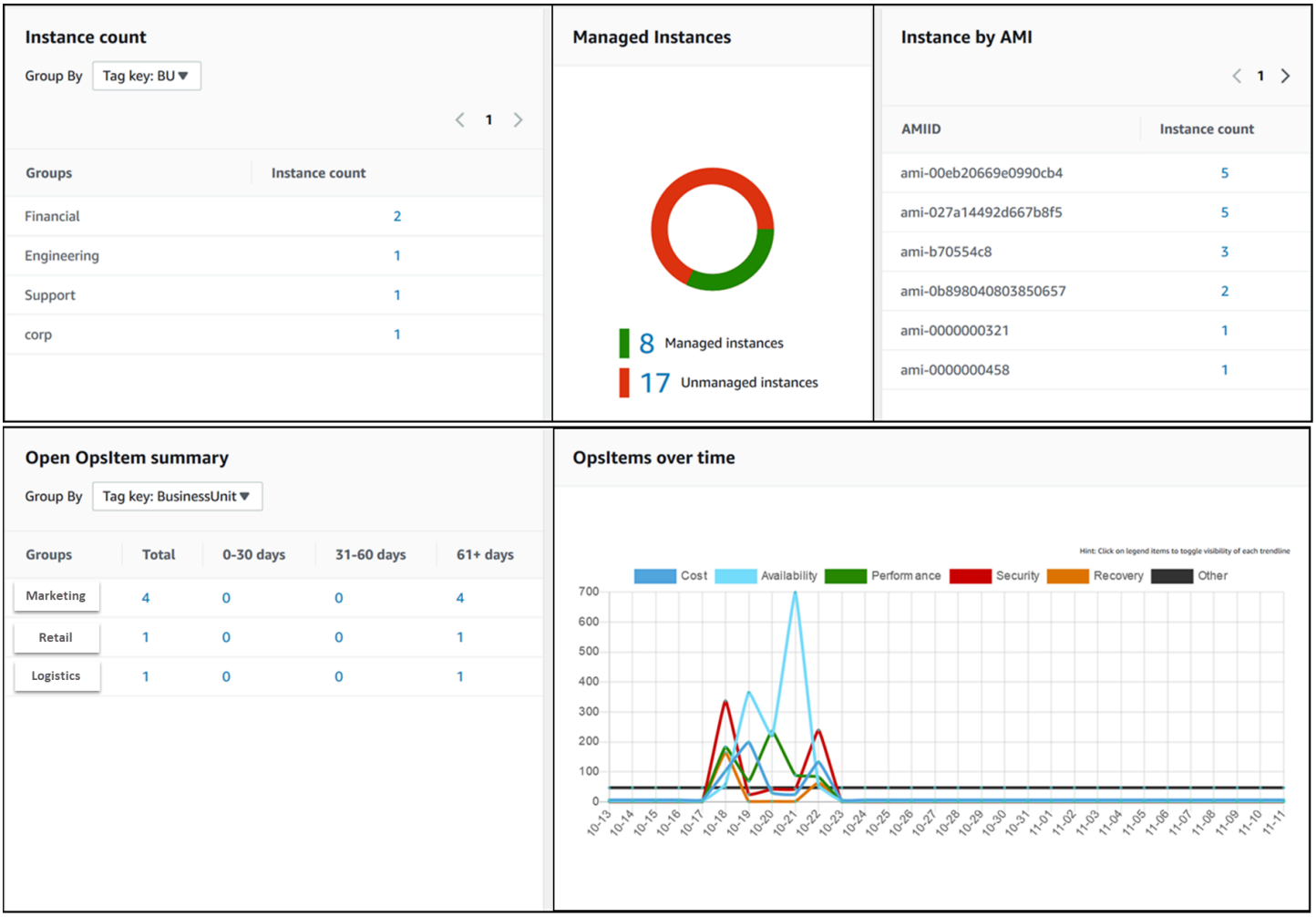
AWS Systems Manager Explorer

AWS Systems Manager Explorer es un panel de operaciones personalizable que transmite información sobre sus recursos de AWS. Explorer muestra una vista agregada de los datos de operaciones (OpsData) de sus Cuentas de AWS y en todas las Regiones de AWS. En Explorer, OpsData incluye metadatos sobre los nodos administrados del entorno híbrido y multinube. OpsData también incluye información proporcionada por otras herramientas de Systems Manager, incluidos

los detalles de cumplimiento de revisiones de Patch Manager y de cumplimiento de asociaciones de State Manager. Para simplificar aún más la forma en la que accede a OpsData, Explorer muestra información servicios de AWS de soporte como AWS Config, AWS Trusted Advisor, AWS Compute Optimizer y AWS Support (casos de soporte).

Para ampliar el conocimiento operacional, Explorer también muestra elementos de trabajo operativos (OpsItems). Explorer proporciona contexto acerca de cómo se distribuyen los OpsItems entre las aplicaciones o las unidades de empresas, cómo se presentan a lo largo del tiempo y cómo varían según la categoría. Puede agrupar y filtrar la información en Explorer para centrarse en los elementos que son relevantes para usted y que requieren que se tomen medidas. Cuando identifica problemas de alta prioridad, puede utilizar OpsCenter de Systems Manager para ejecutar manuales de procedimientos de Automation y resolver rápidamente esos problemas. Para comenzar a utilizar Explorer, abra la consola de Systems Manager. En el panel de navegación, elija Explorer.

La siguiente imagen muestra algunos de los cuadros de informe individuales, llamados widgets, que están disponibles en Explorer.



¿Cuáles son las características de Explorer?

Explorer incluye las siguientes características:

- Visualización personalizable de información procesable: Explorer incluye widgets de arrastrar y soltar que muestran automáticamente información procesable sobre los recursos de AWS. Explorer muestra información en dos tipos de widgets.
  - Widgets informativos: estos widgets resumen los datos de Amazon EC2, Patch Manager, State Manager y Servicios de AWS admitidos, como AWS Trusted Advisor, AWS Compute Optimizer y Soporte. Estos widgets proporcionan un contexto importante para ayudarlo a comprender el estado y los riesgos operativos de sus recursos de AWS. Algunos ejemplos de widgets informativos son recuento de instancias, instancias por AMI, instancias no conformes para revisiones, asociaciones no conformes y casos del centro de soporte.
  - Widgets de OpsItem: un OpsItem de Systems Manager es un elemento de trabajo operativo que está relacionado con uno o más recursos de AWS. Los OpsItems son una característica de Systems Manager OpsCenter. Los OpsItems podrían requerir que los ingenieros de DevOps investiguen y, de ser posible, corrijan un problema. Entre los ejemplos posibles de OpsItems se incluyen la utilización de CPU de instancias EC2 alta, volúmenes de Amazon Elastic Block Store (Amazon EBS) desasociados, error de implementación de AWS CodeDeploy o error de ejecución de Automatización de Systems Manager. Entre los ejemplos de widgets de OpsItem se incluyen Open OpsItem summary (Resumen de OpsItem abierto), OpsItem by status (OpsItem por estado), y OpsItems over time (OpsItem con el tiempo).
- Filtros: cada widget ofrece la capacidad de filtrar información en función de la Cuenta de AWS, la Región de AWS y la etiqueta. Los filtros le ayudan a refinar rápidamente la información mostrada en Explorer.
- Enlaces directos a pantallas de servicios: para ayudarlo a investigar problemas con los recursos de AWS, los widgets de Explorer contienen enlaces directos a pantallas de servicios relacionadas. Los filtros aplicados a un widget permanecen en vigor si se desplaza a una pantalla de servicio relacionada.
- Grupos: para ayudarlo a comprender los tipos de problemas operativos de la organización, algunos widgets le permiten agrupar datos en función de la cuenta, la región y la etiqueta.
- Teclas de etiqueta de informes: al configurar Explorer, puede especificar hasta cinco claves de etiqueta. Estas claves le ayudan a agrupar y filtrar datos en Explorer. Si una clave especificada coincide con una clave de un recurso que genera un OpsItem, la clave y el valor se incluyen en los OpsItems.

- Tres modos de visualización de la Cuenta de AWS y la Región de AWS: Explorer incluye los siguientes modos de visualización para OpsData y OpsItems en las Cuentas de AWS y las Regiones de AWS:
  - Una sola cuenta/una sola región: esta es la vista predeterminada. Este modo permite a los usuarios ver datos y OpsItems desde su propia cuenta y la región actual.
  - Cuenta única/varias regiones: este modo requiere que cree una o más sincronizaciones de datos de recursos mediante la página Settings (Configuración) de Explorer. Una sincronización de datos de recursos agrega OpsData de una o más regiones. Después de crear una sincronización de datos de recursos, puede determinar qué sincronización desea utilizar en el panel de Explorer. A continuación, puede filtrar y agrupar datos por región.
  - Varias cuentas/varias regiones: este modo requiere que su organización o empresa utilice AWS Organizations con All features (Todas las características) habilitadas. Después de configurar AWS Organizations en su entorno informático, puede agregar todos los datos de la cuenta a una cuenta de administración. A continuación, puede crear sincronizaciones de datos de recursos para poder filtrar y agrupar datos en función de la región. Para obtener más información acerca del modo All features (Todas las características) de Organizations, consulte Habilitar todas las características en la organización.
- Informes: puede exportar informes de Explorer como archivos de valores separados por comas (.csv) a un bucket de Amazon Simple Storage Service (Amazon S3). Cuando se complete la exportación, recibirá una alerta de Amazon Simple Notification Service (Amazon SNS).

¿Cómo se relaciona Explorer con OpsCenter?

Systems Manager OpsCenter proporciona una ubicación central en la que los ingenieros de operaciones y los profesionales de TI ven, investigan y resuelven OpsItems relacionados con recursos de AWS. Explorer es un centro de informes en el que los administradores de DevOps ven resúmenes agregados de sus datos de operaciones, incluidos los OpsItems, entre las Regiones de AWS y las cuentas. Explorer ayuda a los usuarios a descubrir tendencias y patrones y, si es necesario, resolver rápidamente problemas mediante manuales de procedimientos de Systems Manager Automation.

La instalación de OpsCenter ahora está integrada con la instalación de Explorer. Si ya ha configurado OpsCenter, Explorer muestra automáticamente los datos de operaciones, incluida la información agregada sobre OpsItems. Si no ha configurado OpsCenter, puede utilizar Configuración de Explorer para comenzar con ambas herramientas. Para obtener más información, consulte Introducción a Systems Manager Explorer y OpsCenter.

¿Qué es OpsData?

OpsData es cualquier dato de operaciones que se muestra en el panel de Systems Manager Explorer. Explorer recupera OpsData de los siguientes orígenes:

- Amazon Elastic Compute Cloud (Amazon EC2)

  Los datos que se muestran en Explorer incluyen: número total de nodos, número total de nodos administrados y no administrados y un recuento de nodos que utilizan una Amazon Machine Image (AMI) específica.

- Systems Manager OpsCenter

  Los datos que se muestran en Explorer incluyen: un recuento de OpsItems por estado, un recuento de OpsItems por gravedad, un recuento de OpsItems abiertos en los grupos y en períodos de tiempo de 30 días, y datos históricos de OpsItems a lo largo del tiempo.

- Systems Manager Patch Manager

  Los datos que se muestran en Explorer incluyen un recuento de nodos no conformes y nodos críticos no conformes.

- AWS Trusted Advisor

  Los datos que se muestran en Explorer incluyen: estado de las comprobaciones de prácticas recomendadas para instancias EC2 reservadas en las áreas de optimización de costos, seguridad, tolerancia a errores, rendimiento y límites de servicio.

- AWS Compute Optimizer

  Los datos que se muestran en Explorer incluyen: un recuento de instancias EC2 subaprovisionadas y sobreaprovisionadas conclusiones de optimización, información sobre precios bajo demanda y recomendaciones para el tipo de instancia y el precio.

- Casos de Soporte Center

  Los datos que se muestran en Explorer incluyen ID de caso, severidad, estado, hora de creación, asunto, servicio y categoría.

- AWS Config

  Los datos que se muestran en Explorer incluyen un resumen general de reglas conformes y no conformes de AWS Config, el número de recursos conformes y no conformes y los detalles específicos sobre cada uno (cuando se profundiza en una regla o un recurso no conforme).

- AWS Security Hub

  Los datos que se muestran en Explorer incluyen un resumen general de los resultados de Security Hub, el número de cada hallazgo agrupado según la severidad y los detalles específicos sobre este.

> **Note**
>
> Para ver casos de AWS Trusted Advisor y del Soporte Center en Explorer, debe tener una cuenta Enterprise o Business configurada con AWS Support.

Puede ver y administrar orígenes de OpsData desde la página Settings (Configuración) de Explorer. Para obtener información sobre cómo instalar y configurar servicios que rellenan widgets de Explorer con OpsData, consulte Configuración de servicios relacionados para Explorer.

¿Se cobra por usar Explorer?

Sí. Cuando activa las reglas predeterminadas para crear OpsItems durante la instalación integrada, se inicia un proceso que crea automáticamente OpsItems. Se cobra a la cuenta en función del número de OpsItems creados al mes. También se cobra a la cuenta en función del número de llamadas a la API `GetOpsItem`, `DescribeOpsItem`, `UpdateOpsItem` y `GetOpsSummary` realizadas al mes. Además, se le puede cobrar por las llamadas a la API públicas a otros servicios que expongan información de diagnóstico relevante. Para más información, consulte Precios de AWS Systems Manager.

Temas

- Introducción a Systems Manager Explorer y OpsCenter
- Uso de Explorer
- Exportación de OpsData desde Systems Manager Explorer
- Solución de problemas de Systems Manager Explorer

Introducción a Systems Manager Explorer y OpsCenter

AWS Systems Manager utiliza una experiencia de instalación integrada para ayudarlo a comenzar a utilizar Systems Manager Explorer y Systems ManagerOpsCenter. En esta documentación,

la instalación de Explorer y OpsCenter se denomina Instalación integrada. Si ya ha preparado OpsCenter, deberá completar la instalación integrada para verificar la configuración y las opciones. Si no ha configurado OpsCenter, puede utilizar la instalación integrada para comenzar con ambas herramientas.

> **Note**
>
> La instalación integrada solo está disponible en la consola de Systems Manager. No se puede configurar Explorer ni OpsCenter mediante programación.

La instalación integrada realiza las siguientes tareas:

- Configura roles y permisos: la instalación integrada crea un rol de AWS Identity and Access Management (IAM) que permite a Amazon EventBridge crear automáticamente OpsItems en función de reglas predeterminadas. Después de la configuración, debe establecer permisos de usuario, grupo o rol para OpsCenter, como se describe en esta sección.
- Habilita las reglas predeterminadas para la creación de OpsItem: la instalación integrada crea reglas predeterminadas en EventBridge. Estas reglas se crean automáticamente OpsItems en respuesta a eventos. Algunos ejemplos de estos eventos son: cambio de estado de un recurso de AWS, cambio en la configuración de seguridad o un servicio que deja de estar disponible.
- Activa fuentes OpsData: la instalación integrada activa los siguientes orígenes de datos que rellenan widgets de Explorer.
  - Soporte Center (Debe tener un plan Business o Enterprise Support para activar este origen).
  - AWS Compute Optimizer (Debe tener un plan Business o Enterprise Support para activar este origen).
  - Conformidad de asociación de Systems Manager State Manager
  - Conformidad de AWS Config
  - Systems Manager OpsCenter
  - Conformidad de parches de Systems Manager Patch Manager
  - Amazon Elastic Compute Cloud (Amazon EC2)
  - Systems Manager Inventory
  - AWS Trusted Advisor (Debe tener un plan Business o Enterprise Support para activar este origen).
  - AWS Security Hub

> **Note**
>
> Puede cambiar los ajustes de la configuración en cualquier momento en la página Settings (Configuración) .

Después de completar la instalación integrada, se recomienda Configurar Explorer para mostrar datos de varias cuentas y regiones. Explorer y OpsCenter sincronizan de forma automática OpsData y OpsItems para la Cuenta de AWS y la Región de AWS que se hayan utilizado al completar la instalación integrada. Puede agregar OpsData y OpsItems desde otras cuentas y regiones creando una sincronización de datos de recursos.

Configuración de servicios relacionados para Explorer

Explorador de AWS Systems Manager y OpsCenter de AWS Systems Manager recopilan información de otros Servicios de AWS y herramientas de Systems Manager o interactúan con ellos. Le recomendamos que ajuste y configure estos otros servicios o herramientas antes de utilizar el programa de instalación integrada.

La siguiente tabla incluye tareas que permiten que Explorer y OpsCenter recopilen información de otros Servicios de AWS y herramientas de Systems Manager o interactúen con ellos.

Tarea	Información
Verificar permisos en Automatización de Systems Manager	Explorer y OpsCenter permiten solucionar los problemas que surjan con los recursos de AWS mediante el uso de los manuales de procedimientos de Systems Manager Automation. Para utilizar esta herramienta de resolución de problemas, debe disponer de permiso para ejecutar manuales de procedimientos de Automatización de Systems Manager. Para obtener más información, consulte Configuración de Automation.
Ajustar y configurar Systems Manager Patch Manager	Explorer incluye un widget que proporciona información sobre la aplicación de revisiones. Para ver estos datos en Explorer, debe

Tarea	Información
	configurar la aplicación de revisiones. Para obtener más información, consulte AWS Systems Manager Patch Manager.
Ajustar y configurar Systems Manager State Manager	Explorer incluye un widget que proporciona información sobre la conformidad de asociación de Systems Manager State Manager. Para ver estos datos en Explorer, debe configurar State Manager. Para obtener más información, consulte AWS Systems Manager State Manager.
Activar el registro de configuración de AWS Config	Explorer utiliza los datos proporcionados por el registrador de configuración de AWS Config para rellenar los widgets con información sobre las instancias EC2. Para ver estos datos en Explorer, active el registro de configuración de AWS Config. Para obtener más información, consulte Administración del registrador de configuración. **Note** Después de habilitar el registro de configuración, Systems Manager puede tardar hasta seis horas en mostrar los datos en los widgets de Explorer que muestran información sobre las instancias EC2.

Tarea	Información
Activar AWS Trusted Advisor	Explorer utiliza los datos proporcionados por Trusted Advisor para mostrar el estado de las comprobaciones de prácticas recomendadas para instancias reservadas de Amazon EC2 en las áreas de optimización de costos, seguridad , tolerancia a errores, rendimiento y límites de servicio. Para ver estos datos en Explorer, debe contar con un plan de soporte empresari al o de negocios. Para obtener más informaci ón, consulte Soporte.
Activar AWS Compute Optimizer	Explorer utiliza los datos proporcionados por Compute Optimizer para mostrar los detalles de un recuento de instancias EC2 subaprovi sionadas y sobreaprovisionadas, conclusio nes de optimización, detalles de precios bajo demanda y recomendaciones para el tipo de instancia y el precio. Para ver estos datos en Explorer, active Compute Optimizer. Para obtener más información, consulta Introducción a AWS Compute Optimizer.
Activar AWS Security Hub	Explorer utiliza los datos proporcionados por Security Hub para rellenar los widgets con información sobre los resultados de seguridad . Para ver estos datos en Explorer, active la integración de Security Hub. Para obtener más información, consulte ¿Qué es AWS Security Hub?

Configuración de roles y permisos para Systems Manager Explorer

La instalación integrada crea y configura de forma automática roles de AWS Identity and Access Management (IAM) para AWS Systems Manager Explorer y AWS Systems Manager OpsCenter. Si completó la instalación integrada, no tendrá que realizar ninguna tarea adicional para configurar roles

y permisos de Explorer. Sin embargo, debe configurar el permiso para OpsCenter, como se describe más adelante en este tema.

El programa de instalación integrada crea y configura los siguientes roles para trabajar con Explorer y OpsCenter.

- `AWSServiceRoleForAmazonSSM`: proporciona acceso a recursos de AWS administrados o utilizados por Systems Manager.
- `OpsItem-CWE-Role`: permite que CloudWatch Events y EventBridge creen OpsItems en respuesta a eventos comunes.
- `AWSServiceRoleForAmazonSSM_AccountDiscovery`: permite que Systems Manager llame a otros Servicios de AWS para encontrar información de Cuenta de AWS cuando se sincronizan los datos. Para obtener más información acerca de este rol, consulte Uso de roles para recopilar información de la Cuenta de AWS para OpsCenter y Explorer.
- `AmazonSSMExplorerExport`: permite que Explorer exporte OpsData a un archivo de valores separados por comas (CSV).

Si configura Explorer para mostrar datos de varias cuentas y regiones mediante AWS Organizations y una sincronización de datos de recursos, Systems Manager creará el rol vinculado a servicios `AWSServiceRoleForAmazonSSM_AccountDiscovery`. Systems Manager utiliza este rol para obtener información acerca de su Cuentas de AWS en AWS Organizations. El rol utiliza la siguiente política de permisos.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "organizations:DescribeAccount",
            "organizations:DescribeOrganization",
            "organizations:ListAccounts",
            "organizations:ListAWSServiceAccessForOrganization",
            "organizations:ListChildren",
            "organizations:ListParents"
         ],
```

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

Para obtener más información acerca de los roles de `AWSServiceRoleForAmazonSSM_AccountDiscovery`, consulte Uso de roles para recopilar información de la Cuenta de AWS para OpsCenter y Explorer.

Configuración de permisos para Systems Manager OpsCenter

Después de completar la instalación integrada, debe configurar los permisos de usuario, grupo o rol para que los usuarios puedan realizar acciones en OpsCenter.

Antes de empezar

Puede configurar OpsCenter para crear y administrar OpsItems en una sola cuenta o en varias cuentas. Si configura OpsCenter para crear y administrar OpsItems en varias cuentas, puede usar la cuenta de administrador delegado de Systems Manager o la cuenta de administrador AWS Organizations para crear, ver o editar OpsItems en otras cuentas de forma manual. Para obtener más información acerca de la cuenta de administrador delegado de Systems Manager, consulte Configurar un administrador delegado para Explorer.

Si configura OpsCenter para una sola cuenta, solo podrá ver o editar OpsItems en la cuenta en la que se crearon los OpsItems. No se pueden compartir ni transferir OpsItems entre Cuentas de AWS. Por este motivo, le recomendamos que configure los permisos para OpsCenter en la Cuenta de AWS que se utiliza para ejecutar las cargas de trabajo de AWS. A continuación, puede crear usuarios o grupos de en dicha cuenta. De esta forma, varios ingenieros de operaciones o profesionales de TI pueden crear, ver y editar OpsItems en la misma Cuenta de AWS.

Explorer y OpsCenter utilizan las siguientes operaciones de la API. Puede utilizar todas las características de Explorer y OpsCenter si el usuario, grupo o rol tiene acceso a esas acciones. También puede crear un acceso más restrictivo, tal y como se describe más adelante en esta sección.

- CreateOpsItem
- CreateResourceDataSync
- DescribeOpsItems
- DeleteResourceDataSync

- GetOpsItem
- GetOpsSummary
- ListResourceDataSync
- UpdateOpsItem
- UpdateResourceDataSync

Si lo prefiere, puede especificar un permiso de solo lectura mediante la adición de la siguiente política insertada a su cuenta, grupo o rol.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:GetOpsItem",
        "ssm:GetOpsSummary",
        "ssm:DescribeOpsItems",
        "ssm:GetServiceSetting",
        "ssm:ListResourceDataSync"
      ],
      "Resource": "*"
    }
  ]
}
```

Para obtener más información acerca de la creación y edición de políticas de IAM, consulte Creación de políticas de IAM en la Guía del usuario de IAM. Para obtener información acerca de cómo asignar esta política a un grupo de IAM, consulte Asociación de una política a un grupo de IAM.

Cree un permiso con lo siguiente y agréguelo a sus usuarios, grupos o roles:

JSON

```
{
```

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

Según la aplicación de identidad que utilice en su organización,
puede seleccionar cualquiera de las siguientes opciones para configurar el acceso de los usuarios.

Para dar acceso, agregue permisos a los usuarios, grupos o roles:

- Usuarios y grupos en AWS IAM Identity Center:

  Cree un conjunto de permisos. Siga las instrucciones de Creación de un conjunto de permisos en la Guía del usuario de AWS IAM Identity Center.

- Usuarios gestionados en IAM a través de un proveedor de identidades:

  Cree un rol para la federación de identidades. Siga las instrucciones descritas en Creación de un rol para un proveedor de identidad de terceros (federación) en la Guía del usuario de IAM.

- Usuarios de IAM:
  - Cree un rol que el usuario pueda aceptar. Siga las instrucciones descritas en Creación de un rol para un usuario de IAM en la Guía del usuario de IAM.
  - (No recomendado) Adjunte una política directamente a un usuario o añada un usuario a un grupo de usuarios. Siga las instrucciones descritas en Adición de permisos a un usuario (consola) de la Guía del usuario de IAM.

Restricción del acceso a OpsItems mediante etiquetas

También puede restringir el acceso a OpsItems mediante la aplicación de una política de IAM en línea que especifique etiquetas. A continuación se muestra un ejemplo que especifica una clave de etiqueta de Department (Departamento) y un valor de etiqueta de Finance (Finanzas). Con esta política, el usuario solo puede llamar a la operación GetOpsItem de la API para ver OpsItems etiquetados anteriormente con Key=Department (Clave=Departamento) y Value=Finance (Valor=Finanzas). Los usuarios no pueden ver otros OpsItems.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:GetOpsItem"
             ],
      "Resource": "*"
      ,
      "Condition": { "StringEquals": { "ssm:resourceTag/Department":
 "Finance" } }
    }
  ]
}
```

A continuación se muestra un ejemplo que especifica las operaciones de la API para ver y actualizar OpsItems. Esta política también especifica dos conjuntos de pares de etiquetas clave-valor: Department-Finance (Departamento-Finanzas) y Project-Unity (Proyecto-Unidad).

JSON

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

```
            "ssm:GetOpsItem",
            "ssm:UpdateOpsItem"
         ],
         "Resource":"*",
         "Condition":{
            "StringEquals":{
               "ssm:resourceTag/Department":"Finance",
               "ssm:resourceTag/Project":"Unity"
            }
         }
      }
   ]
}
```

Para obtener más información acerca de cómo añadir etiquetas a un OpsItem, consulte Crear OpsItems manualmente.

Comprender las reglas de EventBridge predeterminadas creadas por la instalación integrada

Durante el proceso de configuración integrado de Explorer y OpsCenter, puede optar por activar una serie de reglas predeterminadas que se basan en los eventos detectados por Amazon EventBridge. Cuando se detectan estos eventos, el sistema crea OpsItems en AWS Systems Manager OpsCenter automáticamente.

Por ejemplo, la regla `SSMOpsItems-Autoscaling-instance-termination-failure` hace que se cree un OpsItem cuando se produce un error al finalizar una instancia de escalado automático de EC2.

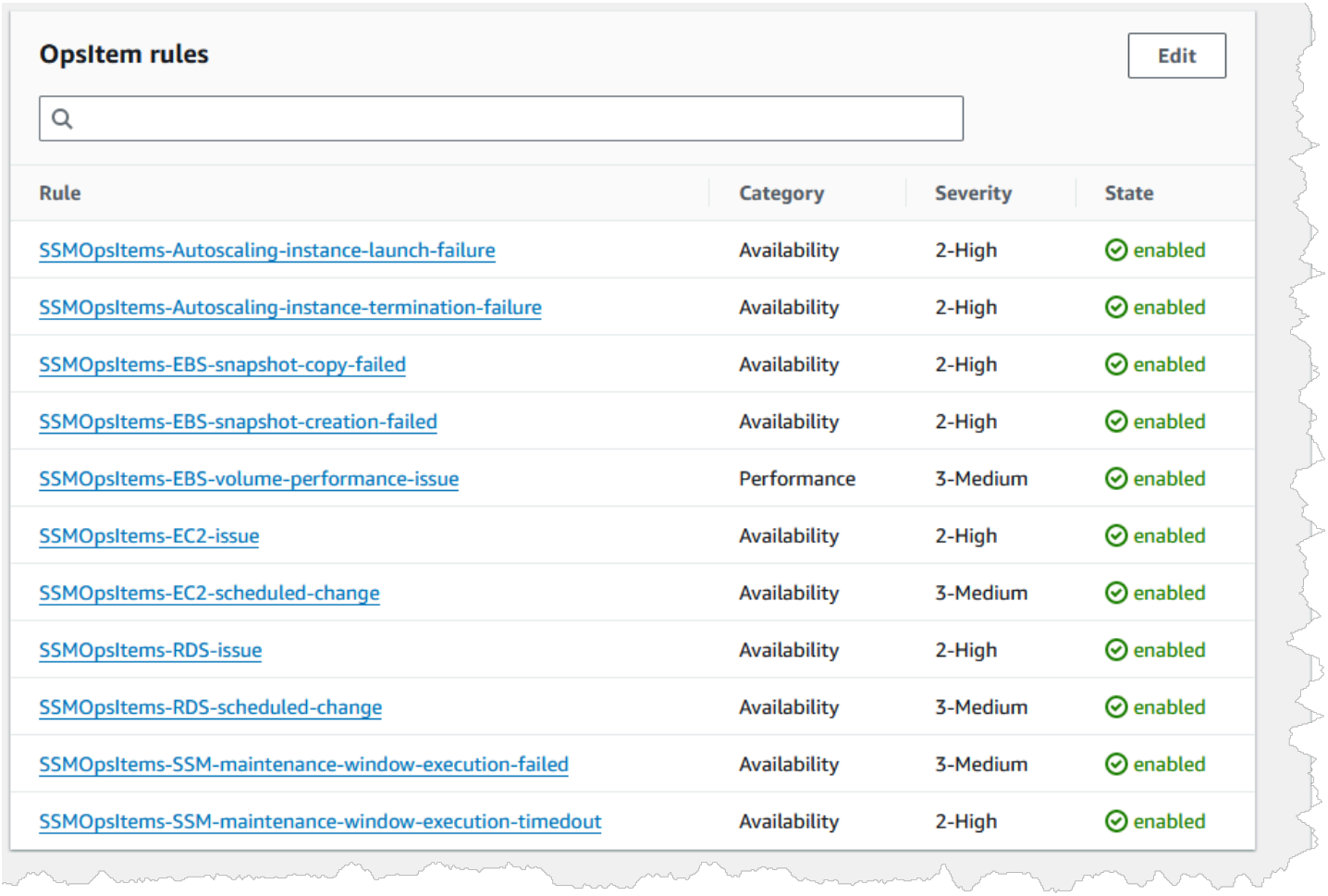
La regla `SSMOpsItems-SSM-maintenance-window-execution-failed` hace que se cree un OpsItem cuando un periodo de mantenimiento de Systems Manager no se completa correctamente.

Para obtener instrucciones de configuración y descripciones de todas las reglas de EventBridge que puede activar durante el proceso de configuración, consulte Configuración de OpsCenter.

Si no desea que EventBridge cree OpsItems para estos eventos, desactive esta opción en la instalación integrada. Si lo prefiere, puede especificar OpsCenter como destino de eventos específicos de EventBridge. Para obtener más información, consulte Configuración de las reglas de EventBridge para crear OpsItems.

Puede desactivar una regla predeterminada o cambiar su categoría y nivel de gravedad en la página de Configuraciones de OpsCenter. Para ello, seleccione OpsCenter, Configuraciones y, a continuación, seleccione Editar en el área de Reglas de OpsItem.

También puede editar la categoría o gravedad asignada a un solo OpsItem creado a partir de estas reglas en la consola de Systems Manager. Para obtener más información, consulte Edición de un OpsItem.



Configurar un administrador delegado para Explorer

Si agrega datos de Explorer de AWS Systems Manager de varias Regiones de AWS y cuentas mediante la sincronización de datos de recursos con AWS Organizations, sugerimos que configure un administrador delegado para Explorer. Un administrador delegado mejora la seguridad de Explorer de las siguientes maneras.

- Limita la cantidad de administradores Explorer que puede crear o eliminar varias cuentas y sincroniza los datos de recursos de la región a una sola persona Cuenta de AWS.

- Ya no es necesario iniciar sesión en la cuenta de administración de AWS Organizations para administrar las sincronizaciones de datos de recursos en Explorer.

Un administrador delegado puede usar las siguientes API de sincronización de datos de recursos de Explorer mediante la consola, el SDK, la AWS Command Line Interface (AWS CLI) o AWS Tools for Windows PowerShell:

- CreateResourceDataSync
- DeleteResourceDataSync
- ListResourceDataSync
- UpdateResourceDataSync

Un administrador delegado puede buscar, filtrar y agregar datos de Explorer de la consola o usar herramientas programáticas como el SDK, la AWS CLI o las AWS Tools for Windows PowerShell. Para buscar, filtrar y agregar datos se utiliza la operación GetOpsSummary de la API.

Un administrador delegado puede crear un máximo de cinco sincronizaciones de datos de recursos para toda una organización o para un subconjunto de unidades organizativas. Las sincronizaciones de datos de recursos creadas por un administrador delegado solo están disponibles en la cuenta del administrador delegado. No se pueden ver las sincronizaciones o los datos agregados en la cuenta de administración de AWS Organizations.

> **Note**
>
> No puede usar una cuenta de administrador delegado para crear una sincronización de datos de recursos en las Regiones de AWS registradas. Debe usar una cuenta de administración de AWS Organizations.

Para obtener más información acerca de la sincronización de datos de recursos, consulte Configuración de Systems Manager Explorer para mostrar datos de varias cuentas y regiones. Para obtener más información sobre AWS Organizations, consulte ¿Qué es AWS Organizations? en la Guía del usuario de AWS Organizations.

Temas

- Antes de empezar
- Configurar un administrador delegado de Explorer

- Anular el registro de un administrador delegado de Explorer

Antes de empezar

La siguiente lista incluye información importante sobre la administración delegada en Explorer.

- Solo puede delegar una cuenta para la administración en Explorer.
- El ID de la cuenta que especifique como administrador delegado de Explorer debe aparecer como cuenta de miembro en AWS Organizations. Para obtener más información, consulte Creación de una Cuenta de AWS en la organización en la Guía del usuario de AWS Organizations.
- Un administrador delegado puede utilizar todas las operaciones de la API de sincronización de datos de recursos de Explorer en la consola o usar herramientas programáticas como el SDK, la AWS Command Line Interface (AWS CLI) o las AWS Tools for Windows PowerShell. Entre las operaciones de la API de sincronización de datos de recursos se incluyen las siguientes: CreateResourceDataSync, DeleteResourceDataSync, ListResourceDataSync y UpdateResourceDataSync.
- Un administrador delegado puede buscar, filtrar y agregar datos de Explorer en la consola o usar herramientas programáticas como el SDK, la AWS CLI o las AWS Tools for Windows PowerShell. Para buscar, filtrar y agregar datos se utiliza la operación GetOpsSummary de la API.
- Las sincronizaciones de datos de recursos creadas por un administrador delegado solo están disponibles en la cuenta del administrador delegado. No se pueden ver las sincronizaciones o los datos agregados en la cuenta de administración de AWS Organizations.
- Un administrador delegado puede crear un máximo de cinco sincronizaciones de datos de recursos.
- Un administrador delegado puede crear una sincronización de datos de recursos para toda una organización de AWS Organizations o para un subconjunto de unidades organizativas.

Configurar un administrador delegado de Explorer

Utilice el procedimiento siguiente para registrar un administrador delegado de Explorer.

Para registrar un administrador delegado de Explorer

1. Inicie sesión en la cuenta de administración de AWS Organizations.
2. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.

3. En el panel de navegación, elija Explorer.
4. Elija Configuración.
5. En la sección Administrador delegado para Explorer, compruebe que ha configurado el rol vinculada al servicio y las opciones de acceso al servicio necesarias. Si es preciso, elija los botones Create rol (Crear rol) y Enable access (Habilitar acceso) para configurar estas opciones.
6. En Account ID (ID de cuenta), ingrese el ID de la Cuenta de AWS. Esta cuenta debe ser una cuenta de miembro en AWS Organizations.
7. Elija Registrar administrador delegado.

Ahora, el administrador delegado tiene acceso a las opciones Include all accounts from my AWS Organizations configuration (Incluir todas las cuentas de mi configuración de Amazon Organizations) y Select organization units in AWS Organizations (Seleccionar unidades organizativas en Amazon Organizations) en la página Crear sincronización de datos de recursos.

Anular el registro de un administrador delegado de Explorer

Utilice el procedimiento siguiente para anular el registro de un administrador delegado de Explorer. Solo se puede anular el registro de una cuenta de administrador delegado con la cuenta de administración de AWS Organizations. Cuando se anula el registro de una cuenta de administrador delegado, el sistema elimina todas las sincronizaciones de datos de recursos de AWS Organizations creadas por el administrador delegado.

Para anular el registro de un administrador delegado de Explorer

1. Inicie sesión en la cuenta de administración de AWS Organizations.
2. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
3. En el panel de navegación, elija Explorer.
4. Elija Configuración.
5. En la sección A delegado para Explorer, elija Anular el registro. El sistema muestra una advertencia.
6. Escriba el ID de la cuenta y elija Remove (Eliminar).

La cuenta ya no tiene acceso a las operaciones de la API de sincronización de datos de recursos de AWS Organizations. El sistema elimina todas las sincronizaciones de datos de recursos de AWS Organizations creadas por la cuenta.

Configuración de Systems Manager Explorer para mostrar datos de varias cuentas y regiones

AWS Systems Manager utiliza una experiencia de instalación integrada para ayudarle a empezar con AWS Systems Manager Explorer y AWS Systems Manager OpsCenter. Después de completar la instalación integrada, Explorer y OpsCenter sincronizar automáticamente los datos. Más específicamente, estas herramientas sincronizan OpsData y OpsItems para la Cuenta de AWS y la Región de AWS que utilizó cuando completó la instalación integrada. Si desea agregar OpsData y OpsItems de otras cuentas y regiones, debe crear una sincronización de datos de recursos, como se describe en este tema.

> **Note**
>
> Para obtener más información acerca de la instalación integrada, consulte Introducción a Systems Manager Explorer y OpsCenter.

Temas

- Comprender las sincronizaciones de datos de recursos para Explorer
- Comprender las sincronizaciones de datos de recursos de varias cuentas y regiones
- Creación de una sincronización de datos de recursos

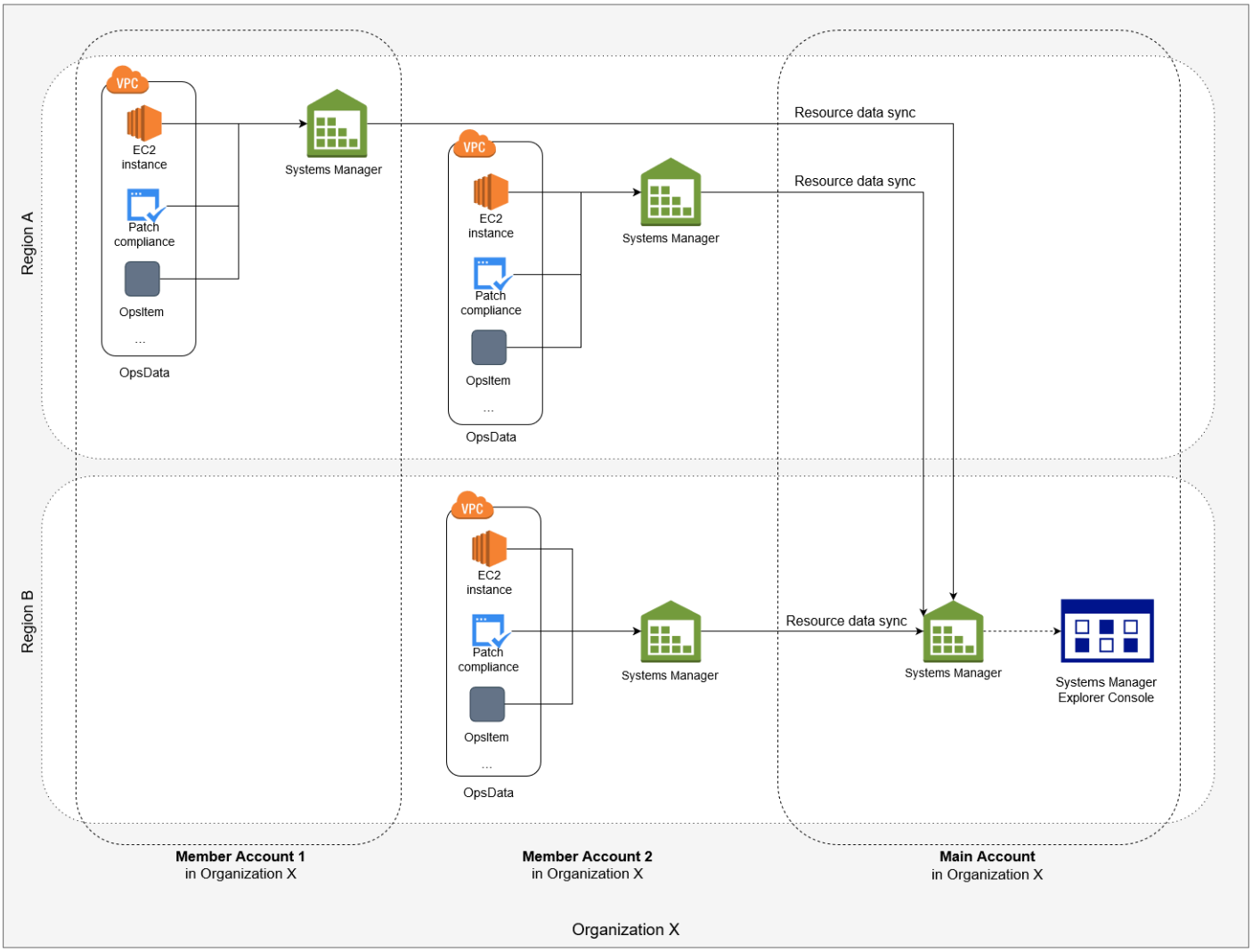
Comprender las sincronizaciones de datos de recursos para Explorer

La sincronización de datos de recursos para Explorer ofrece dos opciones de agregación:

- Cuenta única/varias regiones: puede configurar Explorer para agregar datos de OpsData y OpsItems de varias Regiones de AWS, pero el conjunto de datos está limitado a la Cuenta de AWS actual.
- Varias cuentas/varias regiones: puede configurar Explorer para agregar datos de varias Regiones de AWS y cuentas. Esta opción requiere que instale y configure AWS Organizations. Si ajusta y configura AWS Organizations, puede agregar datos en Explorer de una unidad organizativa o de toda una organización. Systems Manager agrega los datos en la cuenta de administración de AWS Organizations antes de mostrarlos en Explorer. Para obtener más información, consulte ¿Qué es AWS Organizations? en la Guía del usuario de AWS Organizations.

> **Warning**
>
> Si configura Explorer para agregar datos de una organización en AWS Organizations, el sistema habilita OpsData en todas las cuentas miembro de la organización. Habilitar orígenes de OpsData en todas las cuentas miembro aumenta el número de llamadas a API de OpsCenter tales como CreateOpsItem y GetOpsSummary. Se cobrarán las llamadas a estas acciones de API.

El siguiente diagrama muestra una sincronización de datos de recursos configurada para trabajar con AWS Organizations. En este escenario, el usuario tiene dos cuentas definidas en AWS Organizations. La sincronización de datos de recursos agrega datos de ambas cuentas y de varias Regiones de AWS en la cuenta de administración de AWS Organizations, en la que se muestran en Explorer.



Comprender las sincronizaciones de datos de recursos de varias cuentas y regiones

En esta sección se describen detalles importantes sobre sincronizaciones de datos de recursos de varias cuentas y regiones que utilizan AWS Organizations. En concreto, la información de esta sección se aplica si elige una de las siguientes opciones de la página Crear sincronización de datos de recursos:

- “Include all accounts from my AWS Organizations configuration” (Incluir todas las cuentas de mi configuración de Amazon Organizations)
- “Select organization units in AWS Organizations” (Seleccionar unidades organizativas en Amazon Organizations)

Si no tiene previsto utilizar una de estas opciones, puede omitir esta sección.

Cuando crea una sincronización de datos de recursos en la consola de SSM y elige una de las opciones de AWS Organizations, Systems Manager habilita de forma automática todos los orígenes de OpsData en las regiones seleccionadas para todas las Cuentas de AWS en su organización (o en las unidades organizativas seleccionadas). Por ejemplo, incluso si no ha activado Explorer en una región, si selecciona una opción de AWS Organizations para la sincronización de datos de recursos, entonces, Systems Manager recopilará de forma automática OpsData de esa región. Para crear una sincronización de datos de recursos sin permitir los orígenes de OpsData, especifique EnableAllOpsDataSources como falso cuando cree la sincronización de datos. Para obtener más información, consulte los detalles del parámetro `EnableAllOpsDataSources` del tipo de datos ResourceDataSyncSource en la Referencia de la API de Amazon EC2 Systems Manager.

Si no elige una de las opciones de AWS Organizations para una sincronización de datos de recursos, debe completar la instalación integrada en cada cuenta y región donde desee que Explorer acceda a los datos. Si no lo hace, Explorer no mostrará OpsData ni OpsItems para las cuentas y regiones en las que no haya completado la instalación integrada.

Si agrega una cuenta secundaria a la organización, Explorer habilita de forma automática todos los orígenes de OpsData para la cuenta. Si más adelante elimina la cuenta secundaria de la organización, Explorer continúa recopilando OpsData de la cuenta.

Si actualiza una sincronización de datos de recursos existente que utiliza una de las opciones de AWS Organizations, el sistema le solicitará que apruebe la recopilación de todos los orígenes de OpsData para todas las cuentas y regiones afectadas por el cambio.

Si agrega un nuevo servicio a su Cuenta de AWS y Explorer recopila OpsData para ese servicio, Systems Manager configura Explorer de forma automática para recopilar ese OpsData. Por ejemplo, si su organización no utilizó AWS Trusted Advisor cuando creó previamente una sincronización de datos de recursos, pero la organización se registra en este servicio, Explorer actualiza las sincronizaciones de datos de recursos de manera automática para recopilar este OpsData.

Important

Tenga en cuenta la siguiente información relevante acerca de las sincronizaciones de datos de recursos en varias cuentas y regiones:

- Cuando se elimina una sincronización de datos de recursos no se desactiva un origen de OpsData en Explorer.
- Para ver OpsData y OpsItems de varias cuentas, debe activar el modo de AWS Organizations All features (Todas las características) e iniciar sesión en la cuenta de administración de AWS Organizations.
- La mayoría de las regiones de Regiones de AWS están activas de forma predeterminada en la cuenta de Cuenta de AWS, pero algunas regiones solo se activan cuando las selecciona manualmente. Estas regiones se denominan regiones registradas. De forma predeterminada, las sincronizaciones de datos de recursos cross-account/cross-Region de Explorer entre cuentas o regiones no admiten la agregación de datos en las regiones registradas. El 30 de junio de 2025, se agregó compatibilidad para las siguientes regiones registradas.
  - Europa (Milán)
  - África (Ciudad del Cabo)
  - Medio Oriente (Baréin)
  - Asia-Pacífico (Hong Kong)

  Tenga en cuenta que no puede usar una cuenta de administrador delegado para crear una sincronización de datos de recursos en las regiones registradas. Debe usar una cuenta de administración de AWS Organizations.

Creación de una sincronización de datos de recursos

Antes de configurar la sincronización de datos de recursos para Explorer, tenga en cuenta los siguientes detalles.

- Explorer admite un máximo de cinco sincronizaciones de datos de recursos.
- Después de crear una sincronización de datos de recursos para una región, no puede cambiar las opciones de cuenta para esa sincronización. Por ejemplo, si crea una sincronización en la región us-east-2 (Ohio) y elige la opción Include only the current account (Incluir solo la cuenta actual), no puede editar esa sincronización más adelante y elegir la opción Include all accounts from my AWS Organizations configuration (Incluir todas las cuentas de mi configuración de Amazon Organizations) En cambio, debe eliminar la primera sincronización de datos de recursos y crear una nueva.
- La visualización de OpsData en Explorer es de solo lectura.

Utilice el procedimiento siguiente para crear una sincronización de datos de recursos para Explorer.

Cómo crear una sincronización de datos de recursos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Explorer.
3. Elija Configuración.
4. En la sección Configure resource data sync (Configurar sincronización de datos de recursos) elija Create resource data sync (Crear sincronización de datos de recursos).
5. En Resource data sync name (Nombre de sincronización de datos de recursos), escriba un nombre.
6. En la sección Add accounts (Agregar cuentas) elija una opción.

> **Note**
>
> Para utilizar cualquiera de las opciones de AWS Organizations, debe haber iniciado sesión en la cuenta de administración de AWS Organizations o haber iniciado sesión en una cuenta de administrador delegado de Explorer. Para obtener más información acerca de la cuenta de administrador delegado, consulte Configurar un administrador delegado para Explorer.

7. En la sección Regions to include (Regiones para incluir) elija una de las siguientes opciones.

- Elija All current and future regions (Todas las regiones actuales y futuras) para sincronizar automáticamente los datos de todas las Regiones de AWS actuales y de las regiones nuevas que se conecten en el futuro.
- Elija All regions (Todas las regiones) para sincronizar automáticamente los datos de todas las Regiones de AWS actuales.
- Seleccione individualmente las regiones que desee incluir.

8. Elija Crear sincronización de datos de recursos.

El sistema puede tardar varios minutos en completar Explorer con datos después de crear una sincronización de datos de recursos. Puede ver la sincronización seleccionándola en la lista Select a resource data sync (Seleccionar una sincronización de datos de recursos) en Explorer.

Uso de Explorer

Esta sección incluye información sobre cómo personalizar AWS Systems Manager Explorer si cambia el diseño del widget y cambiando los datos que se muestran en el panel.

Contenido

- Edición de las reglas de EventBridge creadas para Explorer
- Edición de orígenes de datos de Systems Manager Explorer
- Personalización de la pantalla de Explorer
- Recepción de resultados de AWS Security Hub en Explorer
- Eliminación de una sincronización de datos de recursos de Systems Manager Explorer

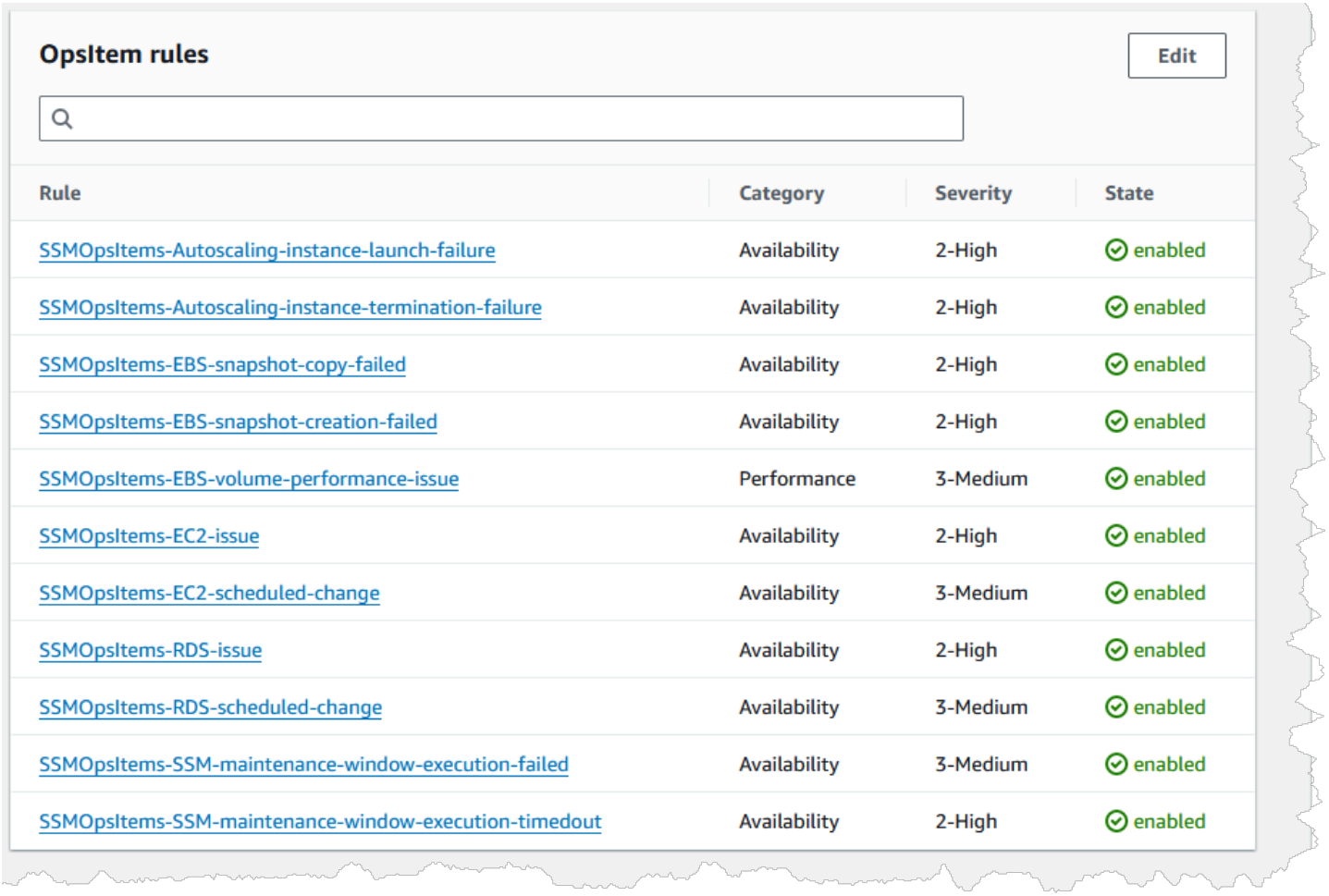
Edición de las reglas de EventBridge creadas para Explorer

Cuando se completa la instalación integrada, el sistema habilita más de una docena de reglas en Amazon EventBridge. Estas reglas crean automáticamente OpsItems en AWS Systems Manager OpsCenter. A continuación, AWS Systems Manager Explorer muestra información agregada sobre los OpsItems.

Cada regla incluye un valor predeterminado de Category (Categoría) y Severity (Gravedad). Cuando el sistema crea OpsItems a partir de un evento, asigna automáticamente la Category (Categoría) y Severity (Gravedad) predefinidas.

Important

No puede editar los valores Category (Categoría) y Severity (Severidad) de las reglas predeterminadas, pero puede editar estos valores en OpsItems creados a partir de las reglas predeterminadas.



Para editar las reglas predeterminadas para crear OpsItems

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Explorer.
3. Elija Configuración.
4. En la sección OpsItems rules (Reglas de OpsItems), elija Edit (Editar).
5. Expanda CWE rules (Reglas de CWE).

6. Desactive la casilla de verificación situada junto a las reglas que no desee utilizar.
7. Utilice las listas Category (Categoría) y Severity (Gravedad) para cambiar esta información de una regla.
8. Seleccione Guardar.

Los cambios surtirán efecto la próxima vez que el sistema cree un OpsItem.

Edición de orígenes de datos de Systems Manager Explorer

Para Regiones de AWS que estén disponibles de forma predeterminada, AWS Systems Manager Explorer muestra datos de los siguientes orígenes. Puede editar la configuración de Explorer para agregar o eliminar orígenes de datos:

- Amazon Elastic Compute Cloud (Amazon EC2)
- Inventario AWS Systems Manager de
- AWS Systems Manager OpsCenter OpsItems
- Conformidad de parches de AWS Systems Manager Patch Manager
- Conformidad de asociación de AWS Systems Manager State Manager
- AWS Trusted Advisor
- AWS Compute Optimizer
- Casos del AWS Support Center
- Conformidad de recursos y reglas de AWS Config
- Resultados de AWS Security Hub

> **Note**
>
> Para la región Asia-Pacífico (Osaka), Explorer no muestra datos ni resultados de AWS Compute Optimizer ni AWS Security Hub.

En las regiones registradas de AWS, Explorer muestra datos de las siguientes fuentes:

- Amazon Elastic Compute Cloud (Amazon EC2)
- Inventario AWS Systems Manager de
- AWS Systems Manager OpsCenter OpsItems

- Conformidad de parches de AWS Systems Manager Patch Manager
- Conformidad de asociación de AWS Systems Manager State Manager
- AWS Trusted Advisor
- Casos del AWS Support Center
- Conformidad de recursos y reglas de AWS Config

> **Note**
>
> - Para ver casos del Soporte Center en Explorer, debe tener una cuenta Enterprise o Business configurada con Soporte.
> - No es posible configurar Explorer para dejar de mostrar datos de OpsCenter OpsItem.

Antes de empezar

Compruebe que ha ajustado y configurado los servicios que rellenan widgets de Explorer con datos. Para obtener más información, consulte Configuración de servicios relacionados para Explorer.

Para editar orígenes de datos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Explorer.
3. Elija Configuración, y a continuación seleccione la pestaña Configurar panel.
4. En la sección Orígenes de OpsData, en la columna Estado, active o desactive los orígenes según los datos que desee ver.

Personalización de la pantalla de Explorer

Puede personalizar el diseño de widgets en AWS Systems Manager Explorer mediante una función de arrastrar y soltar. También puede personalizar OpsData y OpsItems mostrados en Explorer mediante filtros, como se describe en este tema.

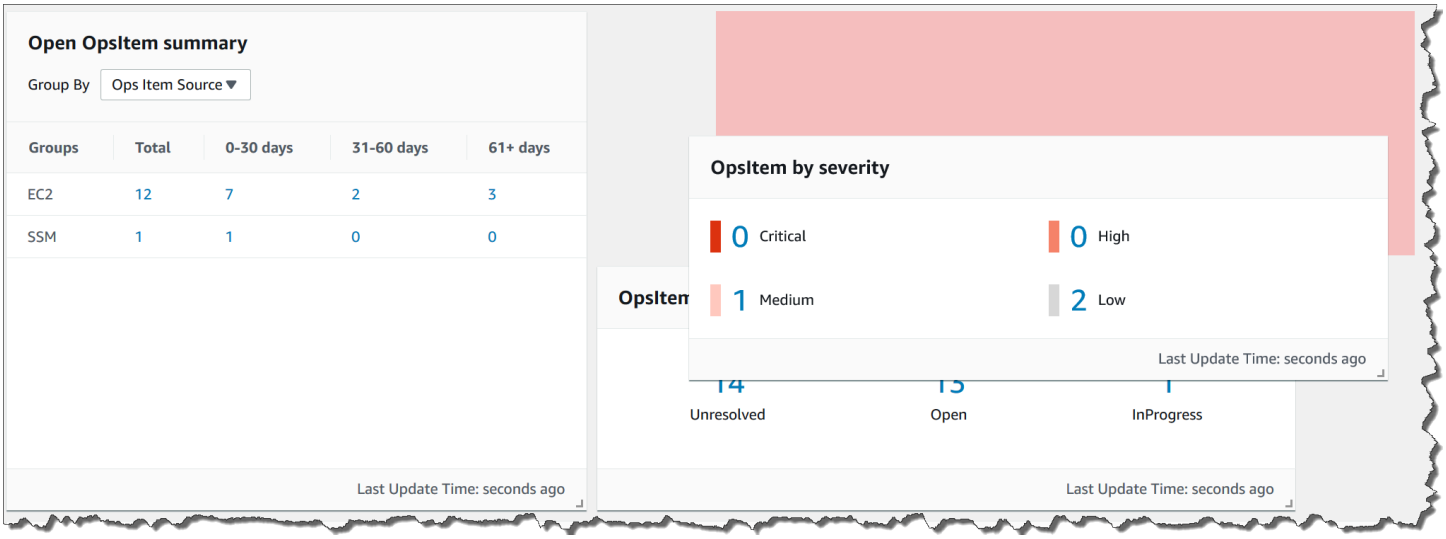
Antes de personalizar el diseño de los widgets, compruebe que los widgets que desea ver se muestran actualmente en Explorer. Para ver algunos widgets en Explorer (como el widget de

conformidad de AWS Config), debe habilitarlos en la página Configure dashboard (Configurar el panel).

Para permitir que los widgets se muestren en Explorer

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Explorer.
3. Seleccione Dashboard actions (Acciones del panel), Configure dashboard (Configurar el panel).
4. Seleccione la pestaña Configure Dashboard (Configurar panel).
5. Elija Enable all (Habilitar todo) o active un widget u origen de datos individual.
6. Elija Explorer para ver sus cambios.

Para personalizar el diseño del widget en Explorer, elija el widget que quiere mover. Haga clic y mantenga presionado el nombre del widget y, a continuación, arrástrelo a su nueva ubicación.



Repita este proceso para cada widget que desee cambiar de posición.

Si decide que no le gusta el nuevo diseño, elija Reset layout (Restablecer diseño) para volver a mover todos los widgets a su ubicación original.

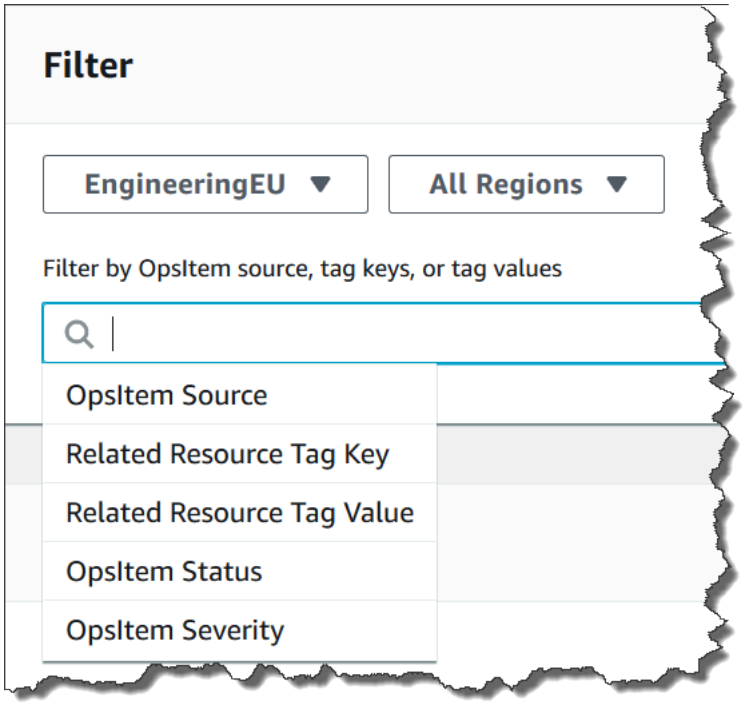
Uso de filtros para cambiar los datos mostrados en Explorer

De forma predeterminada, Explorer muestra los datos de la Cuenta de AWS y de la región actuales. Si crea una o varias sincronizaciones de datos de recursos, puede utilizar filtros para cambiar la sincronización que está activa. A continuación, puede elegir mostrar los datos de una región

específica o de todas las regiones. También puede utilizar la barra de búsqueda para filtrar según diferentes criterios de OpsItem y clave-etiqueta.

Para cambiar los datos mostrados en Explorer mediante filtros

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Explorer.
3. En la sección Filter (Filtro) utilice la lista Select a resource data sync (Seleccionar una sincronización de datos de recursos) para elegir una sincronización.
4. Utilice la lista Regions (Regiones) para elegir una Región de AWS específica o elija All Regions (Todas las regiones).
5. Elija la barra de búsqueda y, a continuación, elija los criterios por los que desea filtrar los datos.



6. Pulse Intro.

Explorer conserva las opciones de filtro seleccionadas si cierra y vuelve a abrir la página.

Recepción de resultados de AWS Security Hub en Explorer

AWS Security Hub proporciona una visión completa de su estado de seguridad en AWS. El servicio recopila datos de seguridad, denominados resultados, de todas las Cuentas de AWS, los servicios y

los productos de terceros compatibles. Los resultados de Security Hub permiten cotejar su entorno con los estándares y las prácticas recomendadas del sector de la seguridad, analizar sus tendencias de seguridad e identificar los problemas de seguridad más prioritarios.

Security Hub envía los resultados a Amazon EventBridge, el cual utiliza una regla de evento para enviar los resultados a Explorer. Después de habilitar la integración, tal y como se describe aquí, puede ver los resultados de Security Hub en un widget de Explorer y ver los detalles de los resultados en los OpsCenter OpsItems. El widget proporciona un resumen de todos los resultados de Security Hub en función de la gravedad. Los nuevos resultados en Security Hub suelen estar visibles en Explorer en cuestión de segundos después de su creación.

⚠ Warning

Tenga en cuenta la siguiente información importante:

- Explorer se integra con el OpsCenter, una herramienta de Systems Manager. Después de habilitar la integración de Explorer con Security Hub, el OpsCenter crea automáticamente los resultados del OpsItems para Security Hub. Según su entorno de AWS, habilitar la integración puede resultar en un gran número de OpsItems, con un coste adicional.

  Antes de continuar, lea sobre la integración de OpsCenter con Security Hub. El tema incluye detalles específicos sobre cómo se cargan a su cuenta los cambios y actualizaciones de los resultados y los OpsItems. Para obtener más información, consulte Comprensión de la integración de OpsCenter con AWS Security Hub. Para obtener más información sobre los precios de OpsCenter, consulte Precios de AWS Systems Manager.
- Si crea una sincronización de datos de recursos en Explorer mientras está conectado a la cuenta de administrador, la integración de Security Hub se habilita automáticamente para el administrador y todas las cuentas miembro de la sincronización. Una vez activado, el OpsCenter crea automáticamente los OpsItems para los resultados del Security Hub, con un coste adicional. Para obtener más información sobre cómo crear la sincronización de datos de recursos, consulte Configuración de Systems Manager Explorer para mostrar datos de varias cuentas y regiones.

Tipos de resultados que recibe Explorer

Explorer recibe todos los resultados de Security Hub. Puede ver todos los resultados en función de su gravedad en el widget de Explorer cuando active la configuración predeterminada de Security

Hub. Explorer crea OpsItems para los resultados de gravedad crítica y alta de forma predeterminada. Puede configurar Explorer manualmente y crear OpsItems para los resultados de gravedad media y baja.

Aunque Explorer no crea los OpsItems para fines informativos, puede ver los datos de operaciones informativas (OpsData) en el widget de resumen de resultados de Security Hub. Explorer crea OpsData para todos los resultados, independientemente de su gravedad. Para obtener más información sobre los niveles de gravedad, consulte la Gravedad en la Referencia de la API de AWS Security Hub.

Habilitación de la integración

En esta sección se describe cómo habilitar y configurar Explorer para comenzar a recibir los resultados de Security Hub.

Antes de empezar

Para comenzar a recibir los resultados de Security Hub, debe completar las siguientes tareas antes de configurar Explorer.

- Habilite y configure Security Hub. Para obtener más información, consulte la Configuración de Security Hub en la Guía del usuario de AWS Security Hub.
- Inicie sesión en la cuenta de administración de AWS Organizations. Systems Manager requiere acceso a AWS Organizations para crear OpsItems de los resultados de Security Hub. Después de iniciar sesión en la cuenta de administración, se le solicitará que seleccione el botón Enable access (Habilitar el acceso) en la pestaña Configure dashboard (Configurar panel) de Explorer, tal y como se describe en el siguiente procedimiento. Si no inicia sesión en la cuenta de administración de AWS Organizations, no se le permitirá el acceso y Explorer no podrá crear OpsItems de los resultados de Security Hub.

Para comenzar a recibir los resultados de Security Hub

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Explorer.
3. Seleccione Settings.
4. Seleccione la pestaña Configure dashboard (Configurar panel).

5. Seleccione AWS Security Hub.
6. Seleccione el control deslizante Disabled (Deshabilitado) para activar AWS Security Hub.

   De forma predeterminada, se muestran los resultados con gravedad alta y crítica. Para mostrar también los resultados de gravedad media y baja, seleccione el control deslizante Deshabilitado junto a Media, Baja.
7. En la sección OpsItems created by Security Hub findings (OpsItems creados por los resultados de Security Hub), elija Enable access (Habilitar el acceso). Si no ve este botón, inicie sesión en la cuenta de administración de AWS Organizations y regrese a esta página para seleccionar el botón.

Cómo ver los resultados de Security Hub

En el siguiente procedimiento se describe cómo ver los resultados de Security Hub.

Para ver los resultados de Security Hub

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Explorer.
3. Busque el widget AWS Security Hub findings summary (Resumen de resultados de Amazon Security Hub). De este modo, se muestran los resultados de Security Hub. Puede seleccionar un nivel de severidad para ver una descripción detallada del OpsItem correspondiente.

Cómo dejar de recibir resultados

En el siguiente procedimiento, se describe cómo dejar de recibir resultados de Security Hub.

Para dejar de recibir los resultados de Security Hub

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Explorer.
3. Seleccione Settings.
4. Seleccione la pestaña Configure dashboard (Configurar panel).
5. Seleccione el control deslizante Enabled (Habilitado) para desactivar AWS Security Hub.

Important

Si la opción para desactivar los resultados de Security Hub aparece atenuada en la consola, puede desactivar esta configuración mediante la ejecución del siguiente comando en la AWS CLI. Debe ejecutar el comando mientras esté conectado a la cuenta de administración de AWS Organizations o a la cuenta de administrador delegado de Systems Manager. Para el parámetro `region`, especifique la Región de AWS en la que desea dejar de recibir los resultados de Security Hub en Explorer.

```
aws ssm update-service-setting --setting-id /ssm/opsdata/SecurityHub --setting-
value Disabled --region Región de AWS
```

A continuación se muestra un ejemplo.

```
aws ssm update-service-setting --setting-id /ssm/opsdata/SecurityHub --setting-
value Disabled --region us-east-1
```

Eliminación de una sincronización de datos de recursos de Systems Manager Explorer

En AWS Systems Manager Explorer, puede agregar OpsData y OpsItems de otras cuentas y regiones creando una sincronización de datos de recursos.

No puede cambiar las opciones de cuenta para una sincronización de datos de recursos. Por ejemplo, si creó una sincronización en la región us-east-2 (Ohio) y eligió la opción Include only the current account (Incluir solo la cuenta actual), no puede editar esa sincronización más adelante y elegir la opción Include all accounts from my AWS Organizations configuration (Incluir todas las cuentas de mi configuración de Amazon Organizations). En cambio, debe eliminar la sincronización de datos de recursos y crear una nueva, como se describe en el procedimiento siguiente.

Para eliminar una sincronización de datos de recursos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Explorer.
3. Elija Configuración.

4. En la sección Configure resource data sync (Configurar sincronización de datos de recursos), elija la sincronización de datos de recursos que desea eliminar.
5. Elija Eliminar.

Exportación de OpsData desde Systems Manager Explorer

Puede exportar 5000 elementos OpsData como un archivo de valores separados por comas (.csv) a un bucket de Amazon Simple Storage Service (Amazon S3) desde el explorador de AWS Systems Manager. Explorer utiliza el manual de procedimientos de automatización AWS-ExportOpsDataToS3 para exportar OpsData. Cuando se exportan OpsData, el sistema muestra la página del manual de procedimientos de automatización donde se pueden especificar detalles, tales como assumeRole, nombre del bucket de Amazon S3, ARN del tema de SNS y campos de columnas que se deben exportar.

Para exportar OpsData:

- Paso 1: especificación de un tema de SNS
- Paso 2: (opcional) configuración de la exportación de datos
- Paso 3: exportación de OpsData

Paso 1: especificación de un tema de SNS

Cuando configure la exportación de datos, debe especificar un tema de Amazon Simple Notification Service (Amazon SNS) que exista en la misma Región de AWS donde desee exportar los datos. Systems Manager envía una notificación al tema de Amazon SNS cuando se completa una exportación. Para obtener información sobre la creación de un tema de Amazon SNS, consulte Creación de un tema de Amazon SNS.

Paso 2: (opcional) configuración de la exportación de datos

Puede configurar los ajustes de exportación de datos desde la página Configuración o Export Ops Data to S3 Bucket.

Configuración de la exportación de datos desde Explorer

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Explorer.

3. Elija Configuración.
4. En la sección Configure data export (Configurar exportación de datos), elija Edit (Editar).
5. Para cargar el archivo de exportación de datos a un bucket de Amazon S3 existente, elija Seleccionar un bucket de S3 existente y elija el bucket de la lista.

   Para cargar el archivo de exportación de datos a un bucket de Amazon S3 nuevo, elija Crear un bucket de S3 nuevo e ingrese el nombre que desea utilizar para el bucket nuevo.

   > **Note**
   >
   > Solo puede editar el nombre del bucket de Amazon S3 y el ARN del tema de Amazon SNS desde la página en la que configuró esos ajustes por primera vez en Explorer. Si configura el bucket de Amazon S3 y el ARN del tema de Amazon SNS desde la página Configuración, solo podrá modificar esa configuración desde la página Configuración.

6. En Seleccionar un ARN de tema de Amazon SNS, elija el tema que desea notificar cuando se complete la exportación.
7. Seleccione Crear.

Paso 3: exportación de OpsData

Cuando se exportan datos de Explorer, Systems Manager crea un rol de AWS Identity and Access Management (IAM) denominado `AmazonSSMExplorerExportRole`. Este rol utiliza la siguiente política de IAM.

JSON

```
{
      "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": [
                "arn:aws:sns:*:*:{{SNSTopicName}}"
            ]
        },
        {
            "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": [
                "*"
            ]
        }
    ]
}
```

El rol incluye la siguiente entidad de confianza.

JSON

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

Para exportar OpsData desde Explorer

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Explorer.

3. Elija un enlace de datos en un widget de Explorer. Por ejemplo, elija Sin resolver o Problemas abiertos en el widget OpsItems por estado. O bien, elija el enlace de datos Crítico o Alto en el widget OpsItem por gravedad. Explorer abre la página OpsData para el conjunto de datos seleccionado.
4. Elija Exportar tabla.

   > **Note**
   >
   > Cuando exporta OpsData por primera vez, el sistema crea un rol de asunción para la exportación. No se puede modificar el rol de asunción predeterminado.

5. En Nombre del bucket de S3, seleccione un bucket existente. Puede elegir Crear para crear un bucket de Amazon S3 si es necesario.

   Si no puede cambiar el nombre del bucket de S3, significa que lo configuró desde la página Configuración. Solo puede cambiar el nombre del bucket desde la página Configuración.

   > **Note**
   >
   > Solo puede editar el nombre del bucket de Amazon S3 y el ARN del tema de Amazon SNS desde la página en la que configuró esos ajustes por primera vez en Explorer.

6. En ARN de tema de SNS, elija un ARN de tema de Amazon SNS existente para notificar cuando se complete la descarga.

   Si no puede cambiar el ARN del tema de Amazon SNS, significa que lo configuró desde la página de Configuración. Solo puede cambiar el ARN del tema desde la página Configuración.
7. (Opcional) En Mensaje de éxito de SNS, especifique el mensaje de éxito que desea que aparezca cuando la exportación se complete correctamente.
8. Elija Enviar. El sistema navega a la página anterior y muestra el mensaje Haga clic para ver el estado del proceso de exportación. Ver detalles.

   Puede elegir Ver detalles para ver el estado del manual de procedimientos y el progreso en Automatización de Systems Manager.

Ahora puede exportar OpsData desde Explorer al bucket de Amazon S3 especificado.

Si no puede exportar datos mediante este procedimiento, compruebe que su usuario, grupo o rol incluya las acciones `iam:CreatePolicyVersion` y `iam:DeletePolicyVersion`. Para obtener más información acerca de cómo agregar estas acciones a su usuario, grupo o rol, consulte Edición de políticas de IAM en la Guía del usuario de IAM.

Solución de problemas de Systems Manager Explorer

Este tema contiene información acerca de cómo solucionar errores o problemas comunes de AWS Systems Manager Explorer.

La sincronización de datos de recursos dejó de funcionar

Si la sincronización de datos de recursos ha dejado de funcionar para una Región de AWS registrada , verifique que la región esté activada tanto para la cuenta de administración de AWS Organizations como para la cuenta de miembro para la que está configurada la sincronización.

La mayoría de las regiones de Regiones de AWS están activas de forma predeterminada en la cuenta de Cuenta de AWS, pero algunas regiones solo se activan cuando las selecciona manualmente. Estas regiones se denominan regiones registradas. De forma predeterminada, las sincronizaciones de datos de recursos cross-account/cross-Region de Explorer entre cuentas o regiones no admiten la agregación de datos en las regiones registradas. El 30 de junio de 2025, se agregó compatibilidad para las siguientes regiones registradas.

- Europa (Milán)
- África (Ciudad del Cabo)
- Medio Oriente (Baréin)
- Asia-Pacífico (Hong Kong)

Tenga en cuenta que no puede usar una cuenta de administrador delegado para crear una sincronización de datos de recursos en las regiones registradas. Debe usar una cuenta de administración de AWS Organizations.

No se puede filtrar recursos de AWS en Explorer después de actualizar las etiquetas en la página Settings (Configuración)

Si actualiza las claves de etiquetas u otra configuración de datos en Explorer, el sistema puede tardar hasta seis horas en sincronizar los datos en función de los cambios.

Las opciones de AWS Organizations en la página Crear sincronización de datos de recursos aparecen atenuadas

Las opciones Include all accounts from my AWS Organizations configuration (Incluir todas las cuentas de mi configuración de Amazon Organizations) y Select organization units in AWS Organizations (Seleccionar unidades organizativas en Amazon Organizations) en la página Crear sincronización de datos de recursos solo están disponibles si ha ajustado y configurado AWS Organizations. Si ajusta y configura AWS Organizations, la cuenta de administración de AWS Organizations o un administrador delegado de Explorer pueden crear sincronizaciones de datos de recursos que utilicen estas opciones.

Para obtener más información, consulte Configuración de Systems Manager Explorer para mostrar datos de varias cuentas y regiones y Configurar un administrador delegado para Explorer.

Explorerno muestra ningún dato

- Compruebe que ha completado la instalación integrada en cada cuenta y región a la que desea que Explorer pueda acceder y mostrar datos. Si no lo hace, Explorer no mostrará OpsData ni OpsItems para las cuentas y regiones en las que no haya completado la instalación integrada. Para obtener más información, consulte Introducción a Systems Manager Explorer y OpsCenter.
- Cuando utilice Explorer para ver datos de varias cuentas y regiones, compruebe que ha iniciado sesión en la cuenta de administración de AWS Organizations o en la cuenta del administrador delegado de Explorer. Para ver OpsData y OpsItems de varias cuentas y regiones, debe haber iniciado sesión en esta cuenta.

Los widgets sobre instancias de Amazon EC2 no muestran datos

Si los widgets sobre instancias de Amazon Elastic Compute Cloud (Amazon EC2), como los widgets Instance count (Recuento de instancias), Managed instances (Instancias administradas) e Instance by AMI (Instancia por AMI) no muestran datos, compruebe lo siguiente:

- Verifique que esperó varios minutos. OpsData puede tardar varios minutos en mostrarse en Explorer después de completar la instalación integrada.
- Compruebe que ha configurado el registrador de configuración de AWS Config. Explorer utiliza los datos proporcionados por el registrador de configuración de AWS Config para rellenar los widgets con información sobre las instancias EC2. Para obtener más información, consulte Administración del registrador de configuración.

- Compruebe que el origen de OpsData de Amazon EC2 está habilitado en la página Settings (Configuración). Además, compruebe que han pasado más de seis horas desde que activó el registro de configuración o desde que realizó cambios en las instancias. Systems Manager puede tardar hasta seis horas en mostrar los datos de AWS Config en los widgets de Explorer EC2 después de activar inicialmente el registro de configuración o realizar cambios en las instancias.
- Tenga en cuenta que si una instancia se detiene o termina, Explorer deja de mostrarlas al cabo de 24 horas.
- Compruebe que se encuentra en la Región de AWS correcta en la que configuró las instancias de Amazon EC2. Explorer no muestra datos sobre instancias locales.
- Si ha configurado una sincronización de datos de recursos para varias cuentas y regiones, compruebe que ha iniciado sesión en la cuenta de administración de Organizations o en la cuenta de administrador delegado de Explorer.

El widget de parches no muestra datos

El widget Non-compliant instances for patching (Instancias no conformes para parches) solo muestra datos sobre las instancias de parches que no son conformes. Este widget no muestra datos si las instancias son conformes. Si cree que tiene instancias no conformes, compruebe que ha ajustado y configurado la aplicación de parches de Systems Manager y utilice AWS Systems Manager Patch Manager para comprobar la conformidad de los parches. Para obtener más información, consulte AWS Systems Manager Patch Manager.

Cuestiones diversas

Explorer no le permite editar ni corregir OpsItems: la visualización de OpsItems en varias cuentas y regiones es de solo lectura. Solo se pueden actualizar y corregir desde su cuenta o región principal.

AWS Systems Manager OpsCenter

OpsCenter, una herramienta de AWS Systems Manager, proporciona una ubicación central en la que los ingenieros de operaciones y los profesionales de TI pueden ver, investigar y resolver elementos de trabajo operativos (OpsItems) relacionados con los recursos de AWS. OpsCenter está diseñado para reducir la media de tiempo de resolución de problemas que impactan en los recursos de AWS. OpsCenter agrega y estandariza OpsItems en todos los servicios, al tiempo que proporciona datos de investigación contextuales sobre cada OpsItem, OpsItems relacionados y recursos relacionados. OpsCenter también proporciona manuales de procedimientos de automatización de Systems Manager que puede utilizar para resolver problemas rápidamente. Puede especificar

datos que se pueden buscar y personalizar para cada OpsItem. También puede ver informes de resumen generados automáticamente sobre OpsItems por estado y origen. Para comenzar a utilizar OpsCenter, abra la consola de Systems Manager. En el panel de navegación, elija OpsCenter.

OpsCenter está integrado con Amazon EventBridge y Amazon CloudWatch. Esto significa que puede configurar estos servicios para crear automáticamente un OpsItem en OpsCenter cuando una alarma de CloudWatch ingresa en el estado `ALARM` o cuando EventBridge procesa un evento desde cualquier servicio de AWS que publica eventos. La configuración de alarmas de CloudWatch y eventos de EventBridge para crear automáticamente OpsItems le permite diagnosticar y remediar de manera rápida problemas con recursos de AWS desde una consola individual.

Para ayudarlo a diagnosticar problemas, cada OpsItem incluye información contextualmente relevante, como el nombre y el ID del recurso de AWS que generó el OpsItem, los detalles de alarmas o eventos, el historial de alarmas y un gráfico de línea de tiempo de alarma.

Para el recurso de AWS, OpsCenter agrega información de AWS Config, registros de AWS CloudTrail y Amazon CloudWatch Events para que no tenga que navegar por varias páginas de la consola durante la investigación.

En la siguiente lista se incluyen tipos de recursos de AWS y métricas para los cuales los clientes configuran alarmas de CloudWatch que crean OpsItems.

- Amazon DynamoDB: las acciones de lectura y escritura de bases de datos alcanzan un umbral.
- Amazon EC2: la utilización de la CPU alcanza un límite
- Facturación de AWS: los cargos estimados alcanzan un límite
- Amazon EC2: una instancia no logra hacer una verificación de estado
- Amazon Elastic Block Store (EBS): la utilización del espacio en disco alcanza un límite

En la siguiente lista se incluyen tipos de reglas de EventBridge configuradas por el cliente para crear OpsItems.

- AWS Security Hub: alerta de seguridad emitida
- DynamoDB: un evento de limitación controlada
- Amazon EC2 Auto Scaling: error al momento de lanzar una instancia
- Systems Manager: error al momento de ejecutar una automatización
- AWS Health: una alerta de mantenimiento programado
- Instancia EC2: cambio de estado de `Running` a `Stopped`

OpsCenter también está integrado con Amazon CloudWatch Application Insights para .NET y SQL Server. Esto significa que puede crear automáticamente OpsItems para problemas detectados en sus aplicaciones. También se pueden integrar OpsCenter con AWS Security Hub para agregar y tomar medidas en cuanto a sus problemas operativos, de seguridad y de rendimiento en Systems Manager.

Los ingenieros de operaciones y los profesionales de TI pueden crear, ver y editar OpsItems mediante la página de OpsCenter en la consola de AWS Systems Manager, las operaciones de la API pública, la AWS Command Line Interface (AWS CLI), AWS Tools for Windows PowerShell, o AWS SDK. Las operaciones de la API pública de OpsCenter también le permiten integrar OpsCenter con los sistemas de administración de casos y los paneles de estado.

¿Cómo puede OpsCenter beneficiar a mi organización?

OpsCenter proporciona una experiencia unificada y estándar para ver problemas relacionados con los recursos de AWS, trabajar con ellos y solucionarlos. Una experiencia estándar y unificada reduce el tiempo que se tarda en solucionar los problemas, en investigar los problemas relacionados y en formar a los nuevos ingenieros de operaciones y profesionales de TI. Una experiencia estándar y unificada también reduce el número de errores manuales en el sistema de administración y de resolución de problemas.

En concreto, OpsCenter ofrece las siguientes ventajas para los ingenieros de operaciones y para las organizaciones:

- Ya no tendrá que navegar por varias páginas de la consola para ver, investigar y resolver OpsItems relacionados con los recursos de AWS. Los OpsItems se recopilan en los servicios y se guardan en una ubicación central.
- Puede ver datos relevantes desde el punto de vista contextual para OpsItems que las alarmas de CloudWatch, los eventos de EventBridge y CloudWatch Application Insights para .NET y SQL Server generan automáticamente.
- Puede especificar el nombre de recurso de Amazon (ARN) de un recurso relacionado con un OpsItem. Al especificar los recursos relacionados, OpsCenter utiliza la lógica integrada para evitar que se cree un OpsItems duplicado.
- Puede ver los detalles y la información sobre la resolución de OpsItems similares.
- Puede ver rápidamente información sobre los manuales de procedimientos de Automatización de Systems Manager y ejecutarlos para resolver problemas.

¿Cuáles son las características de OpsCenter?

- Creación automática y manual de OpsItem

  OpsCenter está integrado con Amazon CloudWatch. Esto significa que puede configurar CloudWatch para crear un OpsItem en OpsCenter automáticamente cuando una alarma ingresa en el estado `ALARM` o cuando Amazon EventBridge procesa un evento desde cualquier servicio de AWS que publica eventos. También puede crear OpsItems manualmente.

  OpsCenter también está integrado con Amazon CloudWatch Application Insights para .NET y SQL Server. Esto significa que puede crear automáticamente OpsItems para problemas detectados en sus aplicaciones.

- OpsItems detallados y consultables

  Cada OpsItem incluye varios campos de información, incluido un título, un ID, una prioridad, una descripción, el origen del OpsItem, y la fecha y la hora en que se actualizó por última vez. Cada OpsItem también incluye las siguientes características configurables:

  - Status (Estado): Pendiente, En curso, Resuelto o Pendiente y En curso.
  - Related resources (Recursos relacionados): un recurso relacionado es el recurso afectado o el recurso que inició el evento de EventBridge que creó el OpsItem. Cada OpsItem incluye una sección Related resources (Recursos relacionados) donde OpsCenter enumera automáticamente el nombre de recurso de Amazon (ARN) del recurso relacionado. También puede especificar los ARN de los recursos relacionados manualmente. Para algunos tipos de ARN, OpsCenter crea automáticamente un enlace profundo que muestra detalles sobre el recurso por lo que ya no es necesario visitar otras páginas de la consola para ver dicha información. Por ejemplo, si especifica el ARN de una instancia de EC2, puede ver todos los detalles proporcionados por EC2 acerca de la instancia en OpsCenter. Puede añadir los ARN de recursos relacionados adicionales manualmente. Cada OpsItem puede enumerar un máximo de 100 ARN de recursos relacionados. Para obtener más información, consulte Adición de recursos relacionados a un OpsItem.
  - Related and Similar OpsItems (OpsItems relacionados y similares): la característica Related OpsItems (OpsItems relacionados) le permite especificar los ID de OpsItems relacionados de algún modo con el OpsItem actual. La característica Similar OpsItem (OpsItem similar) revisa de forma automática los títulos y las descripciones del OpsItem y, a continuación, enumera los OpsItems que pueden estar relacionados o que le pueden interesar.
  - Searchable and private operational data (Datos operativos privados y consultables): los datos operativos son datos personalizados que proporcionan información detallada útil de referencia

acerca del OpsItem. Por ejemplo, puede especificar archivos de registro, cadenas de error, claves de licencia, sugerencias para resolver problemas u otros datos pertinentes. Escriba los datos operativos como pares clave-valor. La clave tiene una longitud máxima de 128 caracteres. El valor tiene un tamaño máximo de 20 KB.

Los datos personalizados se pueden consultar, pero con restricciones. Para la característica Searchable operational data (Datos operativos consultables), todos los usuarios con acceso a la página de información general de OpsItem (tal como indica la operación de la API DescribeOpsItems) pueden ver y realizar búsquedas en los datos especificados. Para la característica Private operational data (Datos operativos privados), solo los usuarios con acceso a OpsItem pueden ver los datos (tal como indica la operación de la API GetOpsItem).

- Deduplication (Desduplicación): si especifica los recursos relacionados, OpsCenter utiliza la lógica integrada para evitar que se cree un OpsItems duplicado. OpsCenter también incluye una característica llamada Información de las operaciones, que muestra información sobre OpsItems duplicados. Para limitar aún más el número de OpsItems duplicados en su cuenta, puede especificar manualmente una cadena de desduplicación para una regla de evento de EventBridge. Para obtener más información, consulte Administración de OpsItems duplicados.

- Edición múltiple de OpsItems: Puede seleccionar varios OpsItems en OpsCenter y editar uno de los siguientes campos: Status (Estado), Priority (Prioridad), Severity (Severidad) y Category (Categoría).
- Soluciones sencillas usando manuales de procedimiento

  Cada OpsItem incluye una sección llamada Runbooks (Manuales de procedimientos) con una lista de manuales de procedimientos de Automatización de Systems Manager que se pueden utilizar para solucionar problemas comunes con los recursos de AWS de forma automática. Si abre un OpsItem, elija un recurso de AWS para ese OpsItem y, a continuación, elija el botón Run automation (Ejecutar automatización) de la consola y, luego, OpsCenter proporciona una lista de manuales de procedimientos de Automation que se pueden ejecutar en el recurso de AWS que generó el OpsItem. Después de ejecutar un manual de procedimientos de Automation desde un OpsItem, el manual de procedimientos se asocia automáticamente al recurso relacionado de OpsItem para futura referencia. Además, si configura reglas de OpsItem en EventBridge de forma automática mediante el uso de OpsCenter, EventBridge asocia automáticamente manuales de procedimientos para eventos comunes. OpsCenter mantiene un registro de 30 días de manuales de procedimientos de Automation ejecutados para un OpsItem. Para obtener más información, consulte Resolución de problemas de OpsItem.

- Change notification (Cambiar notificación): puede especificar el ARN de un tema de Amazon Simple Notification Service (SNS) y publicar notificaciones cada vez que se modifica o edita un OpsItem. El tema de SNS debe existir en la misma Región de AWS del OpsItem.
- Comprehensive OpsItem search capabilities (Numerosas opciones de búsqueda de): OpsCenter proporciona numerosas opciones de búsqueda para ayudarle a localizar los OpsItems rápidamente. A continuación, mostramos algunos ejemplos de cómo realizar búsquedas: ID de OpsItem, Título, Hora de última modificación, Valor de datos operativos, Origen e ID de automatización de una ejecución de manual de procedimientos, por citar solo algunos. Puede limitar aún más los resultados de búsqueda aplicando filtros de estado.
- Informes de resumen de OpsItem

  OpsCenter incluye una página de informes de resumen que muestra automáticamente las siguientes secciones:

  - Status summary (Resumen de estado): resumen de OpsItems por estado (Abierto, En curso, Resuelto, Abierto y En curso).
  - Sources with most open OpsItems (Orígenes con OpsItems más abiertos): desglose de los servicios principales de AWS con OpsItems abiertos.
  - OpsItems by source and age (por origen y antigüedad): recuento de los OpsItems agrupados por origen y días desde su creación.

  Para obtener más información sobre la visualización de informes de resumen de OpsCenter, consulte Visualización de informes de resumen de OpsCenter.
- Compatibilidad con la capacidad de registro y la auditoría

  Puede auditar y registrar las acciones de usuarios de OpsCenter en su Cuenta de AWS a través de la integración con otros servicios de AWS. Para obtener más información, consulte Visualización de registros e informes de OpsCenter.
- Acceso de la consola, CLI, PowerShell y SDK a la herramienta OpsCenter

  Puede trabajar con OpsCenter mediante la consola de AWS Systems Manager, AWS Command Line Interface (AWS CLI), Herramientas de AWS para PowerShell, o el AWS SDK de su elección.

¿Se integra OpsCenter con mi sistema de administración de casos existente?

OpsCenter está diseñado para complementar sus sistemas de administración de casos existentes. Puede integrar OpsItems en su sistema de administración de casos existente empleando

operaciones de la API pública. También puede mantener flujos de trabajo del ciclo de vida manuales en los sistemas actuales y utilizar OpsCenter como un centro de investigación y corrección de errores.

Para obtener más información acerca de las operaciones de la API pública de OpsCenter, consulte las siguientes operaciones de la API en la Referencia de la API de AWS Systems Manager.

- CreateOpsItem
- DescribeOpsItems
- GetOpsItem
- GetOpsSummary
- UpdateOpsItem

¿Se cobra por usar OpsCenter?

Sí. Para más información, consulte Precios de AWS Systems Manager.

¿Funciona OpsCenter con los nodos administrados híbridos y locales?

Sí. Puede utilizar OpsCenter para investigar y solucionar problemas con los nodos administrados locales configurados para Systems Manager. Para obtener más información acerca de la configuración de los servidores locales y las máquinas virtuales de Systems Manager, consulte Administrador de nodos en entornos híbridos y multinube con Systems Manager.

¿Cuáles son las cuotas de OpsCenter?

Puede ver las cuotas para todas las herramientas de Systems Manager en Systems Manager service quotas en la Referencia general de Amazon Web Services. A menos que se indique lo contrario, cada cuota es específica de la región.

Configuración de OpsCenter

AWS Systems Manager utiliza una experiencia de instalación integrada para ayudarlo a comenzar a utilizar OpsCenter y Explorer, herramientas de Systems Manager. Explorer es un panel de operaciones personalizable que ofrece información acerca de sus recursos de AWS. En esta documentación, la instalación de Explorer y OpsCenter se denomina Instalación integrada.

Debe utilizar Instalación integrada para instalar OpsCenter con Explorer. La configuración integrada solo está disponible en la consola de AWS Systems Manager. No se puede configurar Explorer

ni OpsCenter mediante programación. Para obtener más información, consulte Introducción a Systems Manager Explorer y OpsCenter.

Antes de empezar

Al configurar OpsCenter, se habilitan reglas predeterminadas en Amazon EventBridge que crean OpsItems automáticamente. En la siguiente tabla se describen las reglas predeterminadas de EventBridge que crea OpsItems automáticamente. Puede deshabilitar las reglas de EventBridge en la página Configuración de OpsCenter, bajo Reglas de OpsItem.

> **Important**
>
> Los OpsItems creados por las reglas predeterminadas se cargan a su cuenta. Para más información, consulte Precios de AWS Systems Manager.

Nombre de la regla	Descripción
SSMOpsItems-Autoscaling-instance-launch-failure	Esta regla crea OpsItems cuando se produce un error al lanzar una instancia de escalado automático de EC2.
SSMOpsItems-Autoscaling-instance-termination-failure	Esta regla crea OpsItems cuando se produce un error al finalizar una instancia de escalado automático de EC2.
SSMOpsItems-EBS-snapshot-copy-failed	Esta regla crea OpsItems cuando el sistema no ha podido copiar una instantánea de Amazon Elastic Block Store (Amazon EBS).
SSMOpsItems-EBS-snapshot-creation-failed	Esta regla crea OpsItems cuando el sistema no ha podido crear una instantánea de Amazon EBS.
SSMOpsItems-EBS-volume-performance-issue	Esta regla corresponde a una regla de seguimiento de AWS Health. Esta regla crea OpsItems cada vez que hay un problema de rendimiento con un volumen de Amazon EBS

Nombre de la regla	Descripción
	(evento de estado = `AWS_EBS_DEGRADED_EBS_VOLUME_PERFORMANCE`).
SSMOpsItems-EC2-issue	Esta regla corresponde a una regla de seguimiento de AWS Health de eventos inesperados que afectan a servicios o recursos de AWS. La regla crea OpsItems cuando, por ejemplo, un servicio envía comunicaciones sobre problemas operativos que están provocando una degradación del servicio o para poner de manifiesto problemas localizados en el nivel de los recursos. Por ejemplo, esta regla crea un OpsItem para el siguiente evento: `AWS_EC2_OPERATIONAL_ISSUE` .
SSMOpsItems-EC2-scheduled-change	Esta regla corresponde a una regla de seguimiento de AWS Health. AWS puede programar eventos para las instancias, tales como reinicios, detenciones o inicios de instancias. La regla crea OpsItems para eventos programados de EC2. Para obtener más información sobre los eventos programados, consulte Eventos programados para las instancias en la Guía del usuario de Amazon EC2.

Nombre de la regla	Descripción
SSMOpsItems-RDS-issue	Esta regla corresponde a una regla de seguimiento de AWS Health de eventos inesperados que afectan a servicios o recursos de AWS. La regla crea OpsItems cuando, por ejemplo, un servicio envía comunicac iones sobre problemas operativos que están provocando una degradación del servicio o para poner de manifiesto problemas localizados en el nivel de los recursos. Por ejemplo, esta regla crea un OpsItem para los siguientes eventos: `AWS_RDS_MYSQL_DATA BASE_CRASHING_REPEATEDLY` , `AWS_RDS_EXPORT_TASK_FAILED` y `AWS_RDS_CONNECTIVITY_ISSUE` .

Nombre de la regla	Descripción
SSMOpsItems-RDS-scheduled-change	Esta regla corresponde a una regla de seguimiento de AWS Health. La regla crea OpsItems para eventos programados de Amazon RDS. Los eventos programados proporcionan información sobre los próximos cambios en recursos de Amazon RDS. Es posible que algunos eventos recomienden tomar medidas para evitar interrupciones en el servicio. Otros eventos se producen automáticamente sin ninguna acción por su parte. Es posible que un recurso no esté disponible temporalmente durante la actividad de cambio programada. Por ejemplo, esta regla crea un OpsItem para los siguientes eventos: `AWS_RDS_SYSTEM_UPGRADE_SCHEDULED` y `AWS_RDS_MAINTENANCE_SCHEDULED` . Para obtener más información sobre los eventos programados, consulte Categorías de tipos de eventos en la Guía del usuario de AWS Health.
SSMOpsItems-SSM-maintenance-window-execution-failed	Esta regla crea OpsItems cuando se produce un error en el procesamiento del período de mantenimiento de Systems Manager.
SSMOpsItems-SSM-maintenance-window-execution-timedout	Esta regla crea OpsItems cuando se agota el tiempo de espera para iniciar la ventana de mantenimiento de Systems Manager.

Utilice el siguiente procedimiento para configurar OpsCenter.

Configuración de OpsCenter

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.

2. En el panel de navegación, elija OpsCenter.
3. En la página OpsCenter, elija Comenzar.
4. En la página de configuración de OpsCenter , seleccione Habilitar esta opción para que Explorer configure AWS Config y los eventos de Amazon CloudWatch se creen automáticamente en OpsItems en función de las reglas y eventos de uso común. Si no elige esta opción, OpsCenter permanece deshabilitado.

> **Note**
>
> Amazon EventBridge (anteriormente Eventos de Amazon CloudWatch) ofrece todas las funcionalidades de Eventos de CloudWatch y algunas características nuevas, como buses de eventos personalizados, orígenes de eventos de terceros y registro de esquemas.

5. Seleccione HabilitarOpsCenter.

Una vez habilitado OpsCenter, puede hacer lo siguiente desde Configuración:

- Cree alarmas de CloudWatch con el botón Abrir consola de CloudWatch. Para obtener más información, consulte Configuración de alarmas de CloudWatch para crear OpsItems.
- Habilite la información operativa. Para obtener más información, consulte Análisis de la información operativa para reducir OpsItems.
- Habilite las alarmas de resultados de AWS Security Hub. Para obtener más información, consulte Comprensión de la integración de OpsCenter con AWS Security Hub.

Contenido

- (Opcional) Configuración de OpsCenter para administrar OpsItems de forma centralizada entre cuentas
- (Opcional) Configuración de Amazon SNS para recibir notificaciones sobre OpsItems

(Opcional) Configuración de OpsCenter para administrar OpsItems de forma centralizada entre cuentas

Puede utilizar Systems Manager OpsCenter para administrar los OpsItems de forma centralizada, en varias Cuentas de AWS, en una Región de AWS seleccionada. Esta característica está disponible

después de que configura su organización en AWS Organizations. AWS Organizations un servicio de administración de cuentas que permite consolidar varias cuentas de AWS en una organización que cree y administre de forma centralizada. AWS Organizations incluye todas las prestaciones de facturación unificada y posibilidades de administración de cuentas para que pueda satisfacer mejor las necesidades presupuestarias, de seguridad y de conformidad de su negocio. Para obtener más información, consulte ¿Qué es AWS Organizations? en la Guía del usuario de AWS Organizations.

Los usuarios que pertenecen a la cuenta de administración de AWS Organizations pueden configurar una cuenta de administrador delegado para Systems Manager. En el contexto de OpsCenter, los administradores delegados pueden crear, editar y ver los OpsItems en las cuentas de miembros. El administrador delegado también puede usar los manuales de automatización de Systems Manager para resolver los OpsItems o corregir de forma masiva los problemas con los recursos de AWS que están generando los OpsItems.

> **Note**
>
> Solo puede asignar una cuenta como administrador delegado para Systems Manager. Para obtener más información, consulte Creación de un administrador delegado de AWS Organizations para Systems Manager.

Systems Manager ofrece los siguientes métodos de configuración de OpsCenter para administrar los OpsItems de forma centralizada en varias Cuentas de AWS.

- Configuración rápida: una herramienta de Systems Manager, simplifica las tareas de instalación y configuración de las herramientas de Systems Manager. Para obtener más información, consulte AWS Systems Manager Quick Setup.

  La Configuración Rápida para OpsCenter ayuda a completar las siguientes tareas de administración de los OpsItems entre cuentas:

  - Registrar una cuenta como administrador delegado (si aún no se designó el administrador delegado)
  - Creación de los roles y las políticas de AWS Identity and Access Management (IAM) requeridos
  - Especificar una organización o unidades organizativas (OU) de AWS Organizations en las que un administrador delegado pueda gestionar los OpsItems entre cuentas

  Para obtener más información, consulte (Opcional) Configuración de OpsCenter para administrar OpsItems en todas las cuentas mediante Quick Setup.

> **Note**
>
> La configuración rápida no está disponible en todas las Regiones de AWS donde Systems Manager está disponible actualmente. Si la configuración rápida no está disponible en una región en la que desee utilizarla para configurar OpsCenter y administrar los OpsItems de forma centralizada en varias cuentas, debe utilizar el método manual. Para ver una lista de las Regiones de AWS donde está disponible la configuración rápida, consulte Disponibilidad de Quick Setup en Regiones de AWS.

- Configuración manual: si la configuración rápida no está disponible en una región en la que desee utilizarla para configurar OpsCenter y administrar los OpsItems de forma centralizada en varias cuentas, debe utilizar el método manual. Para obtener más información, consulte (Opcional) Configuración manual de OpsCenter para administrar OpsItems de forma centralizada entre cuentas.

(Opcional) Configuración de OpsCenter para administrar OpsItems en todas las cuentas mediante Quick Setup

Quick Setup, una herramienta de AWS Systems Manager, simplifica las tareas de instalación y configuración de las herramientas de Systems Manager. Quick Setup para OpsCenter ayuda a completar las siguientes tareas de administración de los OpsItems entre cuentas:

- Especificación de la cuenta de administrador delegado
- Creación de los roles y las políticas de AWS Identity and Access Management (IAM) requeridos
- Especificación de una organización de AWS Organizations, o un subconjunto de cuentas de miembro, donde un administrador delegado pueda administrar OpsItems en las cuentas

Cuando se configura OpsCenter para administrar OpsItems en todas las cuentas mediante la Configuración Rápida, Quick Setup crea los siguientes recursos en las cuentas especificadas. Estos recursos conceden a las cuentas especificadas permiso para trabajar con OpsItems y utilizar manuales de procedimientos de automatización para solucionar problemas con los recursos de AWS que generan OpsItems.

Recursos	Cuentas
Rol vinculado al servicio de AWS Identity and Access Management (IAM) `AWSServiceRoleForAmazonSSM_AccountDiscovery` Para obtener más información acerca de este rol, consulte Uso de roles para recopilar información de la Cuenta de AWS para OpsCenter y Explorer.	Cuenta de administración y cuenta de administrador delegado de AWS Organizations
Rol de IAM `OpsItem-CrossAccountManagementRole` Rol de IAM `AWS-SystemsManager-AutomationAdministrationRole`	Cuenta de administrador delegado
Rol de IAM `OpsItem-CrossAccountExecutionRole` Rol de IAM `AWS-SystemsManager-AutomationExecutionRole` Política de recursos de Systems Manager `AWS::SSM::ResourcePolicy` para el grupo de OpsItem (`OpsItemGroup`) predeterminado	Todas las cuentas de miembro de AWS Organizations

> **Note**
>
> Si previamente configuró OpsCenter para administrar OpsItems en todas las cuentas mediante el método manual, debe eliminar las pilas de AWS CloudFormation o los conjuntos de pilas creados durante los pasos 4 y 5 de ese proceso. Si esos recursos existen en su cuenta cuando complete el siguiente procedimiento, Quick Setup no podrá configurar correctamente la administración de OpsItem entre cuentas.

Para configurar OpsCenter para administrar OpsItems en todas las cuentas mediante la Configuración Rápida

1. Inicie sesión en la AWS Management Console mediante la cuenta de administración de AWS Organizations.
2. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
3. En el panel de navegación, elija Quick Setup.
4. Elija la pestaña Biblioteca.
5. Desplácese hasta la parte inferior y localice el título de configuración de OpsCenter. Seleccione Crear.
6. En la página Quick Setup de OpsCenter, en la sección Administrador delegado, ingrese un ID de cuenta. Si no puede editar este campo, significa que ya se ha especificado una cuenta de administrador delegado para Systems Manager.
7. En la sección Destinos, elija una opción. Si elige Personalizado, seleccione las unidades organizativas (UO) en las que desea administrar OpsItems en todas las cuentas.
8. Seleccione Crear.

Quick Setup crea la configuración de OpsCenter e implementa los recursos de AWS necesarios en las UO designadas.

> **Note**
>
> Si no desea administrar OpsItems en todas las cuentas, puede eliminar la configuración desde Quick Setup. Cuando elimina la configuración, Quick Setup elimina los siguientes roles y políticas de IAM que se crearon cuando la configuración se implementó originalmente:
>
> - `OpsItem-CrossAccountManagementRole` de la cuenta de administrador delegado
> - `OpsItem-CrossAccountExecutionRole` y `SSM::ResourcePolicy` de todas las cuentas de miembro de Organizations
>
> Quick Setup elimina la configuración de todas las unidades organizativas y las Regiones de AWS donde se implementó la configuración originalmente.

Solución de problemas con una configuración de Quick Setup para OpsCenter

En esta sección se incluye información que lo ayudará a solucionar problemas cuando configure la administración de OpsItem entre cuentas mediante Quick Setup.

Temas

- Error en la implementación de estos StackSets: delegatedAdmin
- El estado de configuración de Quick Setup muestra error

Error en la implementación de estos StackSets: delegatedAdmin

Cuando crea una configuración de OpsCenter, Quick Setup implementa dos conjuntos de pilas de AWS CloudFormation en la cuenta de administración de Organizations. Los conjuntos de pilas utilizan el siguiente prefijo: `AWS-QuickSetup-SSMOpsCenter`. Si Quick Setup muestra el siguiente error: `Deployment to these StackSets failed: delegatedAdmin`, utilice el siguiente procedimiento para solucionar este problema.

Solución de problemas del error de StackSets failed:delegatedAdmin

1. Si recibió el error `Deployment to these StackSets failed: delegatedAdmin` en un cartel rojo en la consola de Quick Setup, inicie sesión en la cuenta de administrador delegado y en la Región de AWS designada como región de origen de Quick Setup.
2. Abra la consola de AWS CloudFormation en https://console.aws.amazon.com/cloudformation.
3. Elija la pila creada por su configuración de Quick Setup. El nombre de la pila incluye lo siguiente: AWS-QuickSetup-SSMOpsCenter.

   > **Note**
   >
   > A veces, CloudFormation elimina implementaciones de pilas que presentan errores. Si la pila no está disponible en la tabla Stacks (Pilas), elija Deleted (Eliminadas) de la lista de filtros.

4. Consulte Status (Estado) y Status reason (Motivo del estado). Para obtener más información sobre los estados de la pila, consulte Códigos de estado de pilas en la Guía del usuario de AWS CloudFormation.
5. Para entender el paso exacto que ha fallado, consulte la pestaña Events (Eventos) y revise el Status (Estado) de cada evento. Para obtener más información, consulte Solución de problemas en la Guía del usuario de AWS CloudFormation.

> **Note**
>
> Si no puede resolver el error de implementación mediante los pasos de la solución de problemas de CloudFormation, elimine la configuración y vuelva a intentarlo.

El estado de configuración de Quick Setup muestra error

Si la tabla Detalles de configuración de la página Detalles de configuración muestra un estado de configuración `Failed`, inicie sesión en la Cuenta de AWS y la región donde se produjo el error.

Solución de un error de Quick Setup para crear una configuración de OpsCenter

1. Inicie sesión en la Cuenta de AWS y la Región de AWS donde se produjo el error.
2. Abra la consola de AWS CloudFormation en https://console.aws.amazon.com/cloudformation.
3. Elija la pila creada por su configuración de Quick Setup. El nombre de la pila incluye lo siguiente: AWS-QuickSetup-SSMOpsCenter.

   > **Note**
   >
   > A veces, CloudFormation elimina implementaciones de pilas que presentan errores. Si la pila no está disponible en la tabla Stacks (Pilas), elija Deleted (Eliminadas) de la lista de filtros.

4. Consulte Status (Estado) y Status reason (Motivo del estado). Para obtener más información sobre los estados de la pila, consulte Códigos de estado de pilas en la Guía del usuario de AWS CloudFormation.
5. Para entender el paso exacto que ha fallado, consulte la pestaña Events (Eventos) y revise el Status (Estado) de cada evento. Para obtener más información, consulte Solución de problemas en la Guía del usuario de AWS CloudFormation.

La configuración de la cuenta de miembro muestra ResourcePolicyLimitExceededException

Si el estado de una pila muestra `ResourcePolicyLimitExceededException`, significa que la cuenta ya se incorporó previamente a la administración entre cuentas de OpsCenter mediante el método manual. Para resolver este problema, debe eliminar las pilas o los conjuntos de pilas de AWS CloudFormation creados durante los pasos 4 y 5 del proceso de incorporación manual. Para

obtener más información, consulte Eliminación de un conjunto de pilas y Eliminación de una pila en la consola de AWS CloudFormation en la Guía del usuario de AWS CloudFormation.

(Opcional) Configuración manual de OpsCenter para administrar OpsItems de forma centralizada entre cuentas

En esta sección se describe cómo configurar manualmente OpsCenter en la administración multicuenta de OpsItem. Si bien este proceso sigue siendo compatible, se ha sustituido por un proceso más reciente que utiliza Quick Setup de Systems Manager. Para obtener más información, consulte (Opcional) Configuración de OpsCenter para administrar OpsItems en todas las cuentas mediante Quick Setup.

Puede configurar una cuenta central para crear OpsItems manuales para las cuentas de miembros y administrar y corregir esos OpsItems. La cuenta central puede ser la cuenta de administración de AWS Organizations o tanto la cuenta de administración de AWS Organizations como la cuenta de administrador de Systems Manager. Le recomendamos que utilice la cuenta de administrador delegado de Systems Manager como cuenta central. Solo puede utilizar esta función después de configurar AWS Organizations.

Con AWS Organizations, puede consolidar varias Cuentas de AWS en una organización que puede crear y administrar de forma centralizada. El usuario de la cuenta central puede crear OpsItems simultáneamente en todas las cuentas de los miembros seleccionados y administrar los OpsItems.

Utilice el proceso de esta sección para habilitar la entidad principal del servicio System Manager en Organizations y configure los permisos de AWS Identity and Access Management (IAM) para trabajar con OpsItems entre cuentas.

Temas

- Antes de empezar
- Paso 1: creación de una sincronización de datos de recursos
- Paso 2: habilitación de la entidad principal del servicio de Systems Manager en AWS Organizations
- Paso 3: creación del rol vinculado al servicio AWSServiceRoleForAmazonSSM_AccountDiscovery
- Paso 4: configuración de permisos para trabajar con OpsItems entre cuentas
- Paso 5: configuración de permisos para trabajar con recursos relacionados entre cuentas

> **Note**
>
> Solo se admiten OpsItems de tipo `/aws/issue` cuando se trabaja con OpsCenter entre cuentas.

Antes de empezar

Antes de configurar OpsCenter para trabajar con OpsItems entre cuentas, asegúrese de haber configurado lo siguiente:

- Una cuenta de administrador delegado de Systems Manager. Para obtener más información, consulte Configurar un administrador delegado para Explorer.
- Una organización definida y configurada en Organizations. Para obtener más información, consulte Crear y administrar una organización en la Guía del usuario de AWS Organizations.
- Configuró automatización de Systems Manager para ejecutar manuales de procedimientos de automatización en varias Regiones de AWS y cuentas de AWS. Para obtener más información, consulte Ejecución de automatizaciones en varias cuentas y Regiones de AWS.

Paso 1: creación de una sincronización de datos de recursos

Después de instalar y configurar AWS Organizations, puede agregar OpsItems en OpsCenter para toda una organización si crea una sincronización de datos de recursos. Para obtener más información, consulte Creación de una sincronización de datos de recursos. Cuando se cree la sincronización, en la sección Agregar cuentas, seleccione la opción Incluir todas las cuentas de mi configuración de AWS Organizations.

Paso 2: habilitación de la entidad principal del servicio de Systems Manager en AWS Organizations

Para permitir que un usuario trabaje con OpsItems en varias cuentas, la entidad principal de servicio de Systems Manager debe estar habilitada en AWS Organizations. Si anteriormente configuró Systems Manager para escenarios de varias cuentas con otras herramientas, es posible que la entidad principal de servicio de Systems Manager ya esté configurada en Organizations. Ejecute los siguientes comandos desde la AWS Command Line Interface (AWS CLI) para verificarlo.
Si no ha configurado Systems Manager para otras situaciones de cuentas múltiples, pase al siguiente procedimiento, Para habilitar la entidad principal del servicio de Systems Manager en AWS Organizations.

Para comprobar que la entidad principal de servicio de Systems Manager está activada en AWS Organizations

1. Descargue la versión más reciente de la AWS CLI en su máquina local.
2. Abra la AWS CLI y ejecute el siguiente comando para especificar sus credenciales y una Región de AWS.

```
aws configure
```

El sistema le solicita que especifique lo siguiente. En el siguiente ejemplo, reemplace cada *marcador de posición del usuario* con su propia información.

```
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. Ejecute el siguiente comando para comprobar que la entidad principal de servicio de Systems Manager está habilitada para AWS Organizations.

```
aws organizations list-aws-service-access-for-organization
```

El comando devuelve información similar a la que se muestra en el siguiente ejemplo.

```
{
    "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"
        }
    ]
```

```
}
```

Para habilitar la entidad principal de servicio de Systems Manager en AWS Organizations

Si no ha configurado la entidad principal de servicio de Systems Manager para Organizations, utilice el siguiente procedimiento. Para obtener más información sobre este comando, consulte enable-aws-service-access en la Referencia de comandos de la AWS CLI.

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI). Para obtener más información, consulte Instalación de la CLI y Configuración de la CLI.
2. Descargue la versión más reciente de la AWS CLI en su máquina local.
3. Abra la AWS CLI y ejecute el siguiente comando para especificar sus credenciales y una Región de AWS.

   ```
   aws configure
   ```

   El sistema le solicita que especifique lo siguiente. En el siguiente ejemplo, reemplace cada *marcador de posición del usuario* con su propia información.

   ```
   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. Ejecute el siguiente comando para habilitar la entidad principal de servicio de Systems Manager para AWS Organizations.

   ```
   aws organizations enable-aws-service-access --service-principal "ssm.amazonaws.com"
   ```

Paso 3: creación del rol vinculado al servicio **AWSServiceRoleForAmazonSSM_AccountDiscovery**

Un rol vinculado a servicios, como el rol AWSServiceRoleForAmazonSSM_AccountDiscovery, es un tipo único de rol de IAM que está vinculado directamente a un Servicio de AWS, como Systems Manager. El servicio predefine los roles vinculados a servicios, que incluyen todos los permisos que el servicio requiere para llamar a otros Servicios de AWS en su nombre. Para obtener más información sobre el rol vinculado al servicio de

`AWSServiceRoleForAmazonSSM_AccountDiscovery`, consulte Permisos de roles vinculados al servicio de detección de cuentas de Systems Manager.

Utilice el siguiente procedimiento para crear el rol vinculado al servicio de `AWSServiceRoleForAmazonSSM_AccountDiscovery` mediante AWS CLI. Para obtener más información sobre el comando que se utilizó en este procedimiento, consulte create-service-linked-role en la AWS CLIreferencia de comandos.

Para crear el rol vinculado al servicio de **`AWSServiceRoleForAmazonSSM_AccountDiscovery`**

1. Inicie sesión en la consola de administración de AWS Organizations.
2. Mientras su sesión está iniciada en la cuenta de administración de Organizations, ejecute el siguiente comando.

```
aws iam create-service-linked-role \
    --aws-service-name accountdiscovery.ssm.amazonaws.com \
    --description "Systems Manager account discovery for AWS Organizations service-
linked role"
```

Paso 4: configuración de permisos para trabajar con OpsItems entre cuentas

Utilice conjuntos de pilas de AWS CloudFormation para crear una política de recursos de `OpsItemGroup` y un rol de ejecución de IAM que otorgue a los usuarios permisos para trabajar con OpsItems entre cuentas. Para comenzar, descargue y descomprima el archivo `OpsCenterCrossAccountMembers.zip`. Este archivo contiene el archivo de plantilla `OpsCenterCrossAccountMembers.yaml` de AWS CloudFormation. Cuando crea un conjunto de pilas con esta plantilla, CloudFormation genera automáticamente la política de recursos `OpsItemCrossAccountResourcePolicy` y el rol de ejecución `OpsItemCrossAccountExecutionRole` en la cuenta. Para obtener más información acerca de la creación de conjuntos de pilas, consulte crear un grupo de pilas en la guía del usuario de AWS CloudFormation.

> **Important**
>
> Tenga en cuenta la siguiente información importante sobre esta tarea:
>
> - Debe implementar el conjunto de pilas mientras tenga abierta la sesión de la cuenta de administración de AWS Organizations.

- Debe repetir este procedimiento mientras tenga abierta la sesión en todas las cuentas que desee utilizar para trabajar con OpsItems en todas ellas, incluida la cuenta de administrador delegada.
- Si desea habilitar la administración OpsItems entre cuentas en diferentes Regiones de AWS, elija agregar todas las regiones en la sección especificar regiones de la plantilla. La administración de OpsItem entre cuentas no es compatible con las regiones habilitadas.

Paso 5: configuración de permisos para trabajar con recursos relacionados entre cuentas

Un OpsItem puede incluir información detallada sobre recursos afectados, por ejemplo, instancias de Amazon Elastic Compute Cloud (Amazon EC2) o buckets de Amazon Simple Storage Service (Amazon S3). El rol de ejecución de `OpsItemCrossAccountExecutionRole` que creó en el Paso 4 anterior otorga a OpsCenter permisos de solo lectura para que las cuentas miembro vean los recursos relacionados. También debe crear un rol de IAM para proporcionar cuentas de administración con permiso para ver e interactuar con recursos relacionados, que completará en esta tarea.

Para comenzar, descargue y descomprima el archivo `OpsCenterCrossAccountManagementRole.zip`. Este archivo contiene el archivo de plantilla `OpsCenterCrossAccountManagementRole.yaml` de AWS CloudFormation. Al crear una pila con esta plantilla, CloudFormation crea automáticamente el rol de IAM `OpsCenterCrossAccountManagementRole` en la cuenta. Para obtener más información sobre la creación de pilas, consulte Crear pilas en la consola AWS CloudFormation en la Guía del usuario de AWS CloudFormation.

Important

Tenga en cuenta la siguiente información importante sobre esta tarea:

- Si planea especificar una cuenta como administrador delegado para OpsCenter, asegúrese de especificar esa Cuenta de AWS cuando cree la pila.
- Debe realizar este procedimiento mientras tenga abierta la sesión en la cuenta de administración de AWS Organizations y nuevamente mientras esté conectado a la cuenta de administrador delegada.

(Opcional) Configuración de Amazon SNS para recibir notificaciones sobre OpsItems

Puede configurar OpsCenter para enviar notificaciones a un tema de Amazon Simple Notification Service (Amazon SNS) cuando el sistema crea un OpsItem o actualiza un OpsItem existente.

Complete los siguientes pasos para recibir notificaciones de OpsItems.

- Paso 1: creación y suscripción a un tema de Amazon SNS
- Paso 2: actualización de la política de acceso de Amazon SNS
- Paso 3: actualización de la política de acceso de AWS KMS

> **Note**
>
> Si activa el cifrado de AWS Key Management Service (AWS KMS) del lado del servidor en el paso 2, debe completar el paso 3. De lo contrario, puede omitir el paso 3.

- Paso 4: activación de las reglas de OpsItems predeterminadas para enviar notificaciones para nuevos OpsItems

Paso 1: creación y suscripción a un tema de Amazon SNS

Para recibir notificaciones, debe crear un tema de Amazon SNS y suscribirse a él. Para obtener más información, consulte Creación de un tema de Amazon SNS y Suscripción a un tema de Amazon SNS en la Guía para desarrolladores de Amazon Simple Notification Service.

> **Note**
>
> Si utiliza OpsCenter en varias Regiones de AWS o cuentas, debe crear y suscribirse a un tema de Amazon SNS en cada región o cuenta donde desee recibir notificaciones de OpsItem.

Paso 2: actualización de la política de acceso de Amazon SNS

Debe asociar un tema de Amazon SNS a OpsItems. Utilice el siguiente procedimiento para actualizar la política de acceso de Amazon SNS de modo que Systems Manager pueda publicar las notificaciones de OpsItems en el tema de Amazon SNS que creó en el paso 1.

1. Inicie sesión en la AWS Management Console y abra la consola de Amazon SNS en https://console.aws.amazon.com/sns/v3/home.
2. En el panel de navegación, elija Temas.
3. Elija el tema que ha creado en el paso 1 y, a continuación, elija Editar.
4. Expanda Política de acceso.
5. Agregue el siguiente bloque `Sid` a la política existente. Reemplace cada *`example resource placeholder`* con su propia información.

```
{
      "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**
>
> Las claves de condición global `aws:SourceAccount` lo protegen contra el escenario de suplente confuso. Para usar esta clave de condición, establezca el valor al ID de cuenta del propietario de OpsItem. Para obtener más información, consulte El suplente confuso en la Guía del usuario de IAM.

6. Elija Guardar cambios.

El sistema envía notificaciones al tema de Amazon SNS cuando se crean o actualizan los OpsItems.

> **Important**
>
> Si configura el tema de Amazon SNS con una clave de cifrado de AWS Key Management Service (AWS KMS) del lado del servidor en el paso 2, luego debe completar el paso 3. De lo contrario, puede omitir el paso 3.

Paso 3: actualización de la política de acceso de AWS KMS

Si activó el cifrado de AWS KMS del lado del servidor para el tema de Amazon SNS, también debe actualizar la política de acceso de AWS KMS key que ha elegido al configurar el tema. Utilice el siguiente procedimiento para actualizar la política de acceso de modo que Systems Manager pueda publicar notificaciones de OpsItem en el tema de Amazon SNS que ha creado en el paso 1.

> **Note**
>
> OpsCenter no admite la publicación de OpsItems en un tema de Amazon SNS configurado con una Clave administrada de AWS.

1. Abra la consola de AWS KMS en https://console.aws.amazon.com/kms.
2. Para cambiar la Región de AWS, utilice el Selector de regiones ubicado en la esquina superior derecha de la página.
3. En el panel de navegación, elija Claves administradas por el cliente.
4. Elija el ID de la clave de KMS que ha elegido al crear el tema.
5. En la sección Key policy (Política de claves), elija Switch to policy view (Cambiar a la vista de política).
6. Elija Editar.
7. Agregue el siguiente bloque `Sid` a la política existente. Reemplace cada *`example resource placeholder`* con su propia información.

```
{
      "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"
  }
```

En el siguiente ejemplo, el nuevo bloque se escribe en la línea 14.



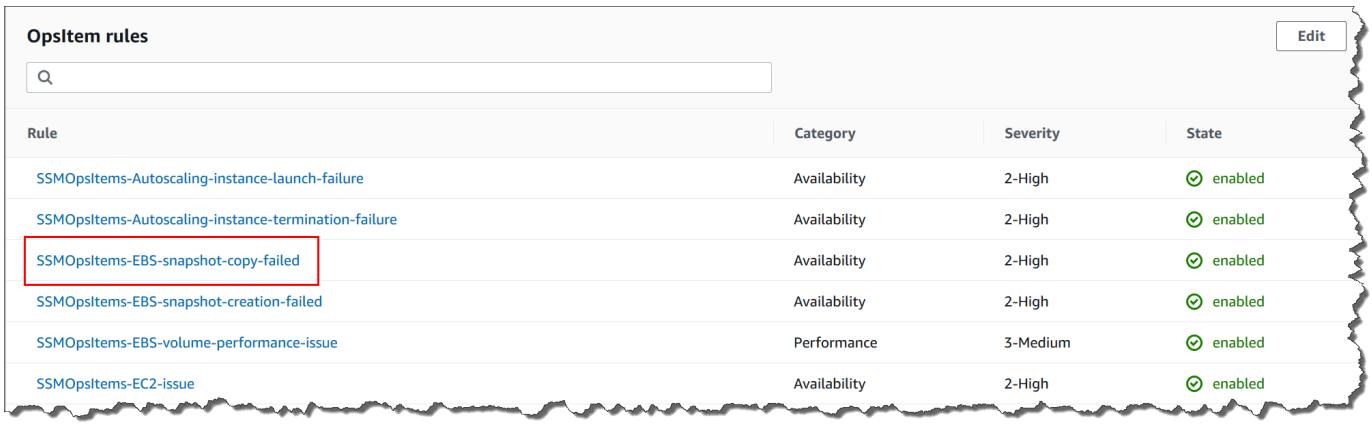
8. Elija Guardar cambios.

Paso 4: activación de las reglas de OpsItems predeterminadas para enviar notificaciones para nuevos OpsItems

Las reglas de OpsItems predeterminadas de Amazon EventBridge no están configuradas con un nombre de recurso de Amazon (ARN) para las notificaciones de Amazon SNS. Utilice el siguiente procedimiento para editar una regla en EventBridge e ingrese un bloque `notifications`.

Para agregar un bloque de notificaciones a una regla de OpsItem predeterminada

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija OpsCenter.
3. Elija la pestaña OpsItems y, a continuación, seleccione Configure sources (Configurar orígenes).
4. Elija el nombre de la regla de origen que desea configurar con un bloque `notifications`, tal y como se muestra en el siguiente ejemplo.



La regla se abre en Amazon EventBridge.

5. En la página Rule details (Detalles de la regla), dentro de la pestaña Targets (Destinos), elija Editar.
6. En la sección Additional settings (Ajustes adicionales), elija Configure input transformer (Configurar transformador de entrada).
7. En el cuadro Plantilla, agregue un bloque `notifications` en el siguiente formato.

```
"notifications":[{"arn":"arn:aws:sns:region:account ID:topic name"}],
```

A continuación se muestra un ejemplo.

```
"notifications":[{"arn":"arn:aws:sns:us-west-2:1234567890:MySNSTopic"}],
```

Introduzca el bloque de notificaciones antes del bloque de `resources`, como se muestra en el ejemplo siguiente para la región Oeste de EE. UU. (Oregón) (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. Seleccione Confirmar.
9. Elija Siguiente.
10. Elija Siguiente.
11. Elija Actualizar regla.

La próxima vez que el sistema cree un OpsItem para la regla predeterminada, publicará una notificación en el tema de Amazon SNS.

Integración de OpsCenter con otros Servicios de AWS

OpsCenter, una herramienta de AWS Systems Manager, se integra con varios Servicios de AWS para diagnosticar y corregir problemas con los recursos de AWS. Debe configurar el Servicio de AWS antes de integrarlo con OpsCenter.

De forma predeterminada, los siguientes Servicios de AWS se integran con OpsCenter y pueden crear OpsItems automáticamente:

- Amazon CloudWatch

- Información de aplicaciones de Amazon CloudWatch
- Amazon EventBridge
- AWS Config
- Administrador de incidentes de AWS Systems Manager

Debe integrar los siguientes servicios con OpsCenter para crear OpsItems automáticamente:

- Amazon DevOps Guru
- AWS Security Hub

Cuando alguno de estos servicios crea un OpsItem, puede administrar y corregir el OpsItem desde OpsCenter. Para obtener más información, consulte Administración de OpsItems y Resolución de problemas de OpsItem.

Para obtener más información acerca de cada Servicio de AWS y cómo se integran con OpsCenter, consulte los siguientes temas.

Temas

- Comprensión de la integración de OpsCenter con Amazon CloudWatch
- Comprensión de la integración de OpsCenter con Información de aplicaciones de Amazon CloudWatch
- Comprensión de la integración de OpsCenter con Amazon DevOps Guru
- Comprensión de la integración de OpsCenter con Amazon EventBridge
- Comprensión de la integración de OpsCenter con AWS Config
- Comprensión de la integración de OpsCenter con AWS Security Hub
- Comprensión de la integración de OpsCenter la integración con la administración de incidentes

Comprensión de la integración de OpsCenter con Amazon CloudWatch

Amazon CloudWatch monitorea sus recursos y servicios de AWS y muestra las métricas de cada Servicio de AWS que utiliza. Cuando la alarma ingresa en el estado de alarma, CloudWatch crea un OpsItem. Por ejemplo, puede configurar una alarma para crear un OpsItem automáticamente si hay un pico en los errores HTTP generados por el Application Load Balancer.

En la siguiente lista, se muestran algunas alarmas que puede configurar en CloudWatch para crear OpsItems:

- Amazon DynamoDB: las acciones de lectura y escritura de bases de datos alcanzan un umbral.
- Amazon EC2: la utilización de la CPU alcanza un límite
- Facturación de AWS: los cargos estimados alcanzan un límite
- Amazon EC2: una instancia no logra hacer una verificación de estado
- Amazon Elastic Block Store (EBS): la utilización del espacio en disco alcanza un límite

Puede crear una alarma o editar una alarma existente para crear un OpsItem. Para obtener más información, consulte Configuración de alarmas de CloudWatch para crear OpsItems.

Cuando habilita OpsCenter con Instalación integrada, CloudWatch se integra con OpsCenter.

Comprensión de la integración de OpsCenter con Información de aplicaciones de Amazon CloudWatch

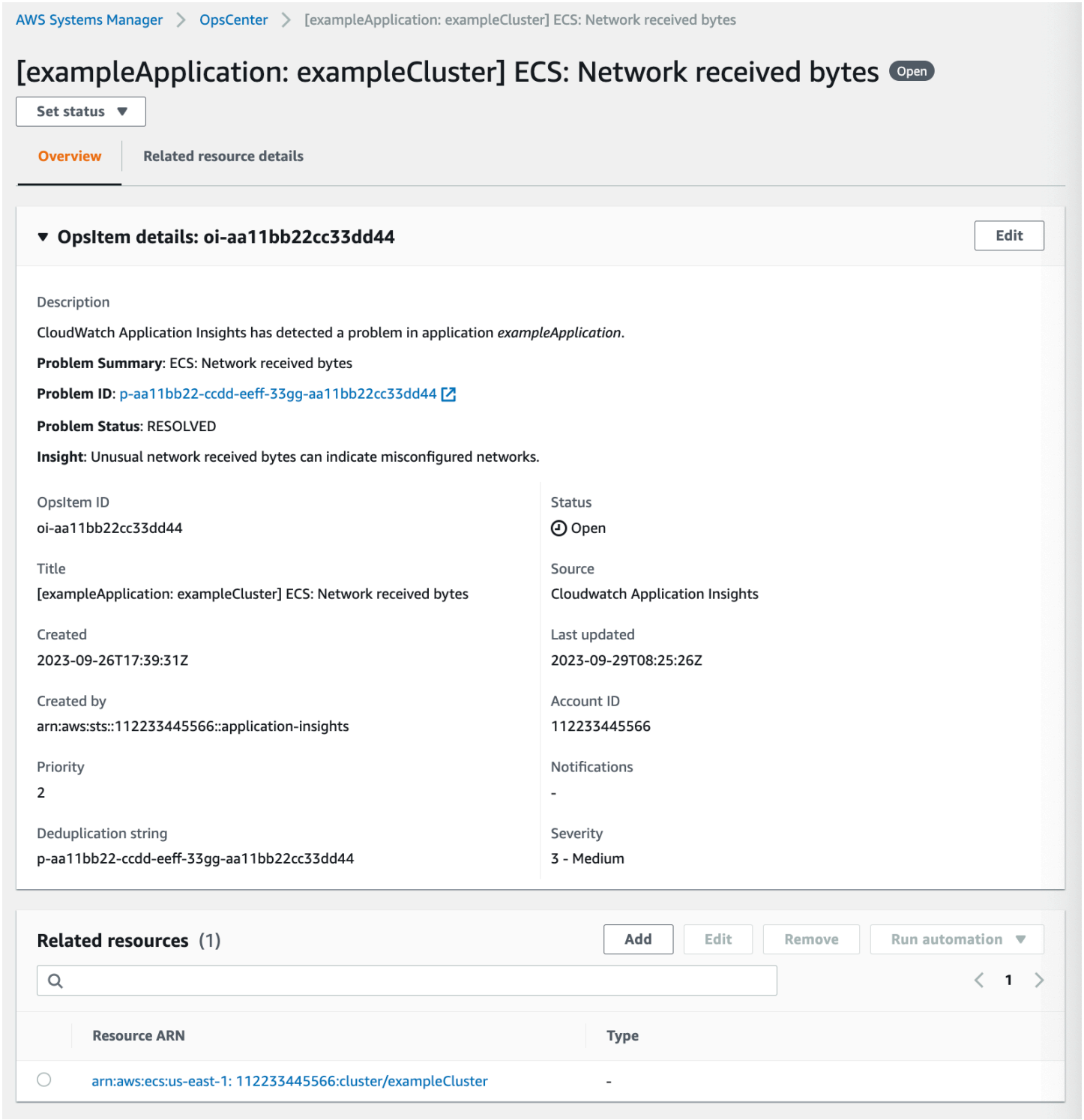
Con Información de aplicaciones de Amazon CloudWatch, puede configurar los monitores más apropiados para su aplicación con el fin de analizar de forma continua los datos en busca de señales que indiquen problemas con las aplicaciones. Cuando configura recursos de aplicaciones en Información de aplicaciones de CloudWatch, puede elegir que el sistema cree OpsItems en OpsCenter. OpsItem se crea un en la consola OpsCenter para cada problema detectado con la aplicación. Para obtener información, consulte Instalación, configuración y administración de la aplicación para monitoreo en la Guía del usuario de Amazon CloudWatch.

> **Note**
>
> A partir del 16 de octubre de 2023, el título y la descripción de OpsItems creados por CloudWatch utilizan ahora el siguiente formato mejorado:

```
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>
```

A continuación se muestra un ejemplo:



Comprensión de la integración de OpsCenter con Amazon DevOps Guru

Amazon DevOps Guru aplica el machine learning para analizar los datos operativos, las métricas y eventos de las aplicaciones para identificar comportamientos que se desvían de los patrones operativos normales. Si permite que DevOps Guru genere unOpsItem en OpsCenter, cada información genera un nuevo OpsItem. Puede utilizar OpsCenter para administrar sus OpsItems.

DevOps Guru crea OpsItems automáticamente. Puede permitir que Amazon DevOps Guru cree OpsItems mediante Quick Setup, una herramienta de Systems Manager. El sistema crea OpsItems mediante el rol vinculado al servicio de AWS Identity and Access Management (IAM) AWSServiceRoleForDevOpsGuru.

Para integrar OpsCenter con DevOps Guru

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Quick Setup.
3. En la página Opciones de configuración de Personalizar DevOps Guru, elija la pestaña Biblioteca.
4. En el panel DevOps Guru, elija Crear.
5. En Opciones de configuración, seleccione Habilitar AWS Systems Manager OpsItems.
6. Seleccione Crear después de completar la instalación.

Comprensión de la integración de OpsCenter con Amazon EventBridge

Amazon EventBridge proporciona una secuencia de eventos de sistema que describe cambios en los recursos de AWS. Cuando habilita OpsCenter con Instalación integrada, EventBridge se integra con OpsCenter y habilita las reglas predeterminadas de EventBridge. De acuerdo con estas reglas, EventBridge crea OpsItems. Con las reglas, puede filtrar eventos y dirigirlos a OpsCenter para investigarlos y corregirlos.

> **Note**
>
> Amazon EventBridge (anteriormente Eventos de Amazon CloudWatch) ofrece todas las funcionalidades de Eventos de CloudWatch y algunas características nuevas, como buses de eventos personalizados, orígenes de eventos de terceros y registro de esquemas.

Estas son algunas reglas que puede configurar en EventBridge para crear un OpsItem:

- Security Hub: alerta de seguridad emitida
- Amazon DynamoDB: un evento de limitación controlada
- Amazon Elastic Compute Cloud Auto Scaling: error al lanzar una instancia
- Systems Manager: error al momento de ejecutar una automatización
- AWS Health: una alerta de mantenimiento programado
- Amazon EC2: cambió el estado de la instancia de en ejecución a detenida

Según sus requisitos, puede crear una regla o editar una regla existente para crear OpsItems. Para obtener instrucciones acerca de cómo editar una regla para crear un OpsItem, consulte Configuración de las reglas de EventBridge para crear OpsItems.

Comprensión de la integración de OpsCenter con AWS Config

AWS Config proporciona una vista detallada de la configuración de los recursos de AWS de su Cuenta de AWS.

AWS Config no se integra directamente con OpsCenter. En su lugar, se crea una regla AWS Config que envía un evento a Amazon EventBridge, por ejemplo, cuando AWS Config detecta una instancia no conforme. A continuación, EventBridge evalúa ese evento con una regla de EventBridge que haya creado. Si la regla coincide, EventBridge transforma el evento en un OpsItem y lo transmite a OpsCenter como destino objetivo.

Con este OpsItem, puede realizar un seguimiento de los detalles del recurso que no cumple con los requisitos, registrar las acciones de investigación y proporcionar acceso a acciones de corrección coherentes.

Información relacionada

Configuración de las reglas de EventBridge para crear OpsItems

Uso de AWS Systems Manager OpsCenter y AWS Config para la supervisión del cumplimiento

Comprensión de la integración de OpsCenter con AWS Security Hub

AWS Security Hub recopila datos de seguridad, llamados resultados de las Cuentas de AWS y los servicios. Al utilizar un conjunto de reglas para detectar y generar resultados, Security Hub le ayuda

a identificar, priorizar y solucionar los problemas de seguridad de los recursos que administra. Tras configurar la integración, tal y como se describe en este tema, Systems Manager crea OpsItems para los resultados de Security Hub en OpsCenter.

> **Note**
>
> OpsCenter tiene integración bidireccional con Security Hub. Esto significa que si actualiza el campo Estado o Gravedad de un OpsItem relacionado con un resultado de seguridad, el sistema sincroniza los cambios con Security Hub. Del mismo modo, cualquier cambio en un resultado se actualiza automáticamente en el OpsItems correspondiente en OpsCenter. Cuando se crea un OpsItem a partir de un resultado de Security Hub, los metadatos del Security Hub se agregan automáticamente al campo de datos operativos delOpsItem. Si se eliminan estos metadatos, las actualizaciones bidireccionales dejan de funcionar.

De forma predeterminada, Systems Manager crea OpsItems para los resultados críticos y de alta gravedad. Puede configurar OpsCenter manualmente a fin de crear OpsItems para resultados de gravedad media y baja. OpsCenter no crea un OpsItems para resultados informativos porque estos no requieren corrección. Para obtener más información sobre los niveles de gravedad, consulte la Gravedad en la Referencia de la API de AWS Security Hub.

Antes de empezar

Antes de configurar OpsCenter para crear OpsItems en función de los resultados de Security Hub, compruebe que ha completado las tareas de configuración del Security Hub. Para obtener más información, consulte la Configuración de Security Hub en la Guía del usuario de AWS Security Hub.

Al integrar Security Hub con OpsCenter, el sistema crea OpsItems mediante el rol vinculado al servicio `AWSServiceRoleForSystemsManagerOpsDataSync` de IAM. Para obtener más información acerca de este rol, consulte Uso de roles para crear OpsData y OpsItems para Explorer.

> **Warning**
>
> Tenga en cuenta la siguiente información relevante acerca de los precios de la integración de OpsCenter con Security Hub:
>
> - Si ha iniciado sesión en la cuenta de administrador del Security Hub al configurar OpsCenter y una integración con el Security Hub, el sistema crea OpsItems para los resultados en las cuentas del administrador y de todos los miembros. Todos los OpsItems

se crean en la cuenta de administrador. En función de una variedad de factores, esto puede provocar una factura inesperadamente elevada de AWS.

Si ha iniciado sesión en una cuenta de miembro al configurar la integración, el sistema solo crea OpsItems para los resultados en esa cuenta individual. Para obtener más información sobre la cuenta de administrador de Security Hub, las cuentas de los miembros y su relación con la lista de eventos de EventBridge a fin de obtener información sobre los resultados, consulte Tipos de integración de Security Hub con EventBridge en la Guía del usuario de AWS Security Hub.

- Por cada resultados que cree un OpsItem, se le cobrará el precio normal de creación del OpsItem. También se le cobrará si edita el OpsItem o si el resultado correspondiente se actualiza en Security Hub (lo que desencadena una actualización de OpsItem).
- OpsItems que se crean mediante una integración con AWS Security Hub no se encuentran limitadas en este momento por la cuota máxima de 500.000 OpsItems por cuenta en una región. Por lo tanto, es posible que las alertas de Security Hub creen más de 500.000 OpsItems facturables en cada región de una cuenta.

  Por lo tanto, para los entornos de alta producción, recomendamos limitar el alcance de los resultados de Security Hub solo a los problemas de mayor gravedad.

Para configurar OpsCenter y crear OpsItems para los resultados de Security Hub

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija OpsCenter.
3. Elija Configuración.
4. En la sección Resultados de Security Hub, seleccione Editar.
5. Seleccione el control deslizante para cambiar de Desactivado a Activado.
6. Si desea que el sistema cree OpsItems para resultados de gravedad media o baja, active estas opciones.
7. Elija Save (Guardar) para guardar la configuración.

Utilice el siguiente procedimiento si no desea que el sistema siga creando OpsItems para los resultados de Security Hub.

Para dejar de recibir OpsItems para los resultados de Security Hub

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija OpsCenter.
3. Elija Configuración.
4. En la sección Resultados de Security Hub, seleccione Editar.
5. Seleccione el control deslizante para cambiar de Activado a Desactivado. Si no puede utilizar el control deslizante, significa que no se ha habilitado Security Hub para la Cuenta de AWS.
6. Seleccione Guardar para guardar la configuración. OpsCenter ya no crea OpsItems en función de los resultados de Security Hub.

> **Important**
>
> Un administrador delegado del Administrador de datos o la cuenta administrativa de AWS Organizations pueden habilitar los resultados de Security Hub en OpsCenter para varias cuentas y Regiones de AWS al crear una sincronización de datos de recursos en Explorer. Si el origen de Security Hub está habilitada en Explorer y existe una sincronización de datos de recursos destinada a la cuenta del miembro en la que deshabilitó la integración del Security Hub, prevalecerá la configuración seleccionada por el administrador. OpsCenter sigue creando OpsItems para los resultados de Security Hub. Para dejar de crear OpsItems para los resultados de Security Hub en una cuenta de miembro a la que se dirige una sincronización de datos de recursos, contáctese con el administrador y solicite que se elimine la cuenta de la sincronización de datos de recursos o que se desactive el origen de Security Hub en Explorer. Para obtener más información sobre cómo cambiar la configuración en Explorer, consulte Edición de orígenes de datos de Systems Manager Explorer.

Comprensión de la integración de OpsCenter la integración con la administración de incidentes

Administrador de incidentes, una herramienta de AWS Systems Manager, proporciona una consola de administración de incidentes que lo ayuda a mitigar y recuperarse de los incidentes que afectan a las aplicaciones alojadas de AWS. Un incidente es cualquier interrupción no planificada o reducción de la calidad de los servicios. Después de establecer y configurar Administrador de incidentes, el sistema crea OpsItems en OpsCenter automáticamente.

Cuando el sistema crea un incidente en Administrador de incidentes, también crea un OpsItem en OpsCenter y muestra el incidente como un elemento relacionado. Si el OpsItem ya existe, Administrador de incidentes no crea un OpsItem. El primer OpsItem se denomina OpsItem principal. Si un incidente crece en escala y alcance, puede agregar incidentes adicionales a un OpsItem existente. Si es necesario, puede crear un incidente de forma manual para un OpsItem. Después de que se cierra un incidente, puede crear un análisis en Incident Manager a fin de revisar y mejorar el proceso de corrección para problemas similares.

De forma predeterminada, OpsCenter se integra con Administrador de incidentes. Si Administrador de incidentes no está configurado, la página de OpsCenter muestra un mensaje para configurarlo. Cuando Administrador de incidentes crea un OpsItem, puede administrar y corregir el OpsItem desde OpsCenter. Para obtener instrucciones sobre cómo crear un incidente para un OpsItem, consulte Creación de un incidente para un OpsItem.

Create OpsItems

Después de configurar OpsCenter, una herramienta de AWS Systems Manager, e integrarlo con sus Servicios de AWS, los Servicios de AWS crean OpsItems automáticamente en función de reglas, eventos o alarmas predeterminados.

Puede ver los estados y los niveles de gravedad de las reglas predeterminadas de Amazon EventBridge. Si es necesario, puede crear o editar estas reglas desde Amazon EventBridge. También puede ver las alarmas desde Amazon CloudWatch y crear o editar alarmas. Mediante reglas y alarmas, puede configurar los eventos para los que desee generar OpsItems automáticamente.

Cuando el sistema crea un OpsItem, se encuentra en estado Abierto. Puede cambiar el estado a En curso al iniciar la investigación del OpsItem y a Resuelto después de corregir el OpsItem. Para obtener más información acerca de cómo configurar alarmas y reglas en los Servicios de AWS para crear OpsItems y cómo crear OpsItems de forma manual, consulte los siguientes temas.

Temas

- Configuración de las reglas de EventBridge para crear OpsItems
- Configuración de alarmas de CloudWatch para crear OpsItems
- Crear OpsItems manualmente

Configuración de las reglas de EventBridge para crear OpsItems

Cuando Amazon EventBridge recibe un evento, crea un OpsItem nuevo en función de reglas predeterminadas. Puede crear una regla o editar una regla existente para establecer OpsCenter como destino de un evento de EventBridge. Para obtener información acerca de cómo crear una regla de evento, consulte Creación de una regla para un Servicio de AWS en la Guía del usuario de Amazon EventBridge.

Para configurar una regla de EventBridge para crear OpsItems en OpsCenter

1. Abra la consola de Amazon EventBridge en https://console.aws.amazon.com/events/.
2. En el panel de navegación, seleccione Reglas.
3. En la página Rules (Reglas), para Event bus (Buses de eventos), elija default (predeterminado).
4. En Reglas, para elegir una regla, seleccione la casilla de verificación situada junto a su nombre.
5. Seleccione el nombre de la regla para abrir la página de detalles. En Detalles de regla, verifique que Estado tenga el valor Habilitado.

   > **Note**
   >
   > Si es necesario, puede actualizar el estado con Editar, en la esquina superior derecha de la página.

6. Elija la pestaña Destinos.
7. En la pestaña Targets, seleccione Edit.
8. Para los tipos de destino, seleccione Servicio de AWS.
9. En Select a target (Seleccione un destino), elija Systems Manager OpsItem.
10. Si hay muchos tipos de destino, EventBridge necesita permiso para enviar eventos al destino. En estos casos, EventBridge puede crear el rol de AWS Identity and Access Management (IAM) necesario para que se ejecute la regla:

    - Para crear un rol de IAM automáticamente, elija Creación de un nuevo rol para este recurso específico.
    - Para utilizar un rol de IAM creado con el objetivo de darle permiso a EventBridge para crear OpsItems en OpsCenter, elija Use existing role (Utilizar rol existente).

11. En Configuración adicional, en Configurar entrada de destino, elija Transformador de entrada.

Puede utilizar la opción Transformador de entrada para especificar una cadena de deduplicación y demás información importante para OpsItems, como título y gravedad.

12. Elija Configurar transformador de entrada.
13. En Transformador de entrada de destino, en Ruta de entrada, especifique los valores que se van a analizar del evento que se desencadena. Por ejemplo, para analizar la hora de inicio, de finalización y otros detalles del evento que desencadena la regla, utilice el siguiente JSON.

```
{
    "end-time": "$.detail.EndTime",
    "failure-cause": "$.detail.cause",
    "resources": "$.resources[0]",
    "source": "$.detail.source",
    "start-time": "$.detail.StartTime"
}
```

14. En Template (Plantilla), especifique la información que se va a enviar al destino. Por ejemplo, utilice el siguiente JSON para pasar información a OpsCenter. La información se utiliza para crear un OpsItem.

> **Note**
>
> Si la plantilla de entrada está en formato JSON, el valor del objeto de la plantilla no puede incluir comillas. Por ejemplo, los valores de los recursos, la causa del error, la fuente, la hora de inicio y la hora de finalización no pueden estar entre comillas.

```
{
    "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",
    "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>
        },
         },
        "resources": {
            "value": <resources>
        }
    }
}
```

Para obtener más información acerca de estos campos, consulte Transforming target input (Transformar la entrada de destino) en la Guía del usuario de Amazon EventBridge.

15. Seleccione Confirmar.
16. Elija Siguiente.
17. Elija Siguiente.
18. Elija Actualizar regla.

Después de crear un OpsItem a partir de un evento, puede ver los detalles del evento abriendo el OpsItem y deslizando el cursor hacia abajo hasta la sección Private operational data (Datos operativos privados). Para obtener información acerca de cómo configurar las opciones en un OpsItem, consulte Administración de OpsItems.

Configuración de alarmas de CloudWatch para crear OpsItems

Durante la instalación integrada de OpsCenter, una herramienta de AWS Systems Manager, permite que Amazon CloudWatch cree OpsItems automáticamente en función de alarmas comunes. Puede crear una alarma o editar una alarma existente para crear OpsItems en OpsCenter.

CloudWatch crea un nuevo rol vinculado a un servicio en AWS Identity and Access Management (IAM) cuando configura una alarma para crear OpsItems. El rol nuevo se denomina `AWSServiceRoleForCloudWatchAlarms_ActionSSM`. Para obtener más información acerca de los roles vinculados a servicios de CloudWatch, consulte Uso de roles vinculados a servicios de CloudWatch en la Guía del usuario de Amazon CloudWatch.

Cuando una alarma de CloudWatch genera un OpsItem, el OpsItem muestra Alarma de CloudWatch: "***alarm_name***" está en estado ALARM.

Para ver los detalles de un OpsItem determinado, elija el OpsItem y, a continuación, elija la pestaña Detalles de recursos relacionados. Puede editar OpsItems de forma manual para cambiar detalles como la gravedad o la categoría. Sin embargo, al editar la gravedad o la categoría de una alarma, Systems Manager no puede actualizar la gravedad o la categoría de los OpsItems que ya se han creado a partir de la alarma. Si una alarma crea un OpsItem y si especificó una cadena de desduplicación, la alarma no creará OpsItems adicionales incluso si edita la alarma en CloudWatch. Si el OpsItem se resuelve en OpsCenter, CloudWatch creará un nuevo OpsItem.

Para obtener más información sobre la configuración de alarmas de CloudWatch, consulte los siguientes temas.

Temas

- Configuración de una alarma de CloudWatch para crear OpsItems (consola)
- Configuración de una alarma de CloudWatch existente para crear OpsItems (mediante programación)

Configuración de una alarma de CloudWatch para crear OpsItems (consola)

Puede crear manualmente una alarma o actualizar una alarma existente para crear OpsItems desde Amazon CloudWatch.

Para crear una alarma de CloudWatch y configurar Systems Manager como destino de dicha alarma

1. Complete los pasos 1 a 9 tal como se especifica en Creación de una alarma de CloudWatch basada en un umbral estático en la Guía del usuario de Amazon CloudWatch.
2. En la sección Acción de Systems Manager, elija Agregar acción de Systems Manager OpsCenter.
3. Elija OpsItems.
4. En Gravedad, elija de 1 a 4.

5. (Opcional) En Categoría, elija una categoría para el OpsItem.
6. Complete los pasos 11 a 13 tal como se especifica en Creación de una alarma de CloudWatch basada en un umbral estático en la Guía del usuario de Amazon CloudWatch.
7. Elija Next (Siguiente) y complete el asistente.

Para editar una alarma existente y configurar Systems Manager como destino de dicha alarma

1. Abra la consola de CloudWatch en https://console.aws.amazon.com/cloudwatch/.
2. En el panel de navegación, elija Alarmas.
3. Seleccione la alarma y, a continuación, elija Actions (Acciones), Edit (Editar).
4. Cambie la configuración en las secciones Metrics (Métricas) y Conditions (Condiciones) y, a continuación, elija Next (Siguiente) (opcional).
5. En la sección Systems Manager, elija Add Systems Manager OpsCenter action (Agregar acción de Systems Manager OpsCenter).
6. En Severity (Severidad), elija un número.

> **Note**
>
> La severidad es un valor definido por el usuario. Usted o su organización determinan lo que significa cada valor de severidad y cualquier acuerdo de nivel de servicio asociado a cada severidad.

7. En Category (Categoría), elija una opción (opcional).
8. Elija Next (Siguiente) y complete el asistente.

Configuración de una alarma de CloudWatch existente para crear OpsItems (mediante programación)

Puede configurar alarmas de Amazon CloudWatch para crear OpsItems mediante la AWS Command Line Interface (AWS CLI), las plantillas de AWS CloudFormation o los fragmentos de código Java.

Temas

- Antes de empezar
- Configuración de alarmas de CloudWatch para crear OpsItems (AWS CLI)
- Configuración de alarmas de CloudWatch para crear o actualizar OpsItems (CloudFormation)

- Configuración de alarmas de CloudWatch para crear o actualizar OpsItems (Java)

Antes de empezar

Si edita mediante programación una alarma existente o crea una alarma que crea OpsItems, debe especificar un nombre de recurso de Amazon (ARN). Este ARN identifica Systems Manager OpsCenter como el objetivo de OpsItems creado a partir de la alarma. Puede personalizar el ARN para que los OpsItems creados a partir de la alarma incluyan información específica, como la severidad o la categoría. Cada ARN incluye la información que se describe en la tabla siguiente.

Parámetro	Detalles
`Region` (obligatorio)	La Región de AWS donde existe la alarma. Por ejemplo: `us-west-2` . Para obtener más información acerca de las Regiones de AWS donde puede utilizar OpsCenter, consulte Cuotas y puntos de enlace de AWS Systems Manager.
`account_ID` (obligatorio)	El mismo ID de Cuenta de AWS utilizado para crear la alarma. Por ejemplo: `123456789012` . El ID de cuenta debe ir seguido de dos puntos (`:`) y el parámetro `opsitem` como se muestra en los siguientes ejemplos.
`severity` (obligatorio)	Un nivel de severidad definido por el usuario para OpsItems creados a partir de la alarma. Valores válidos: `1`, `2`, `3`, `4`
`Category` (opcional)	Una categoría para los OpsItems creados a partir de la alarma. Valores válidos: `Availability` , `Cost`, `Performance` , `Recovery` y `Security`.

Cree el ARN utilizando la siguiente sintaxis. Este ARN no incluye el parámetro opcional `Category`.

```
arn:aws:ssm:Region:account_ID:opsitem:severity
```

A continuación se muestra un ejemplo.

```
arn:aws:ssm:us-west-2:123456789012:opsitem:3
```

Para crear un ARN que utilice el parámetro opcional `Category`, utilice la siguiente sintaxis.

```
arn:aws:ssm:Region:account_ID:opsitem:severity#CATEGORY=category_name
```

A continuación se muestra un ejemplo.

```
arn:aws:ssm:us-west-2:123456789012:opsitem:3#CATEGORY=Security
```

Configuración de alarmas de CloudWatch para crear OpsItems (AWS CLI)

Este comando requiere que especifique un ARN para el parámetro `alarm-actions`. Para obtener más información acerca de cómo crear el ARN, consulte Antes de empezar.

Para configurar CloudWatch a fin de crear OpsItems (AWS CLI)

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.
2. Ejecute el siguiente comando a fin de recopilar información sobre la alarma que desea configurar.

   ```
   aws cloudwatch describe-alarms --alarm-names "alarm name"
   ```
3. Ejecute el siguiente comando para actualizar una alarma. Reemplace cada *example resource placeholder* con su propia información.

   ```
   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
   ```

A continuación se muestra un ejemplo.

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
```

Configuración de alarmas de CloudWatch para crear o actualizar OpsItems (CloudFormation)

En esta sección, se incluyen plantillas de AWS CloudFormation que puede utilizar para configurar las alarmas de CloudWatch para crear o actualizar OpsItems de manera automática. Cada plantilla requiere que especifique un ARN para el parámetro `AlarmActions`. Para obtener más información acerca de cómo crear el ARN, consulte Antes de empezar.

Alarma métrica: utilice la siguiente plantilla de CloudFormation para crear o actualizar una alarma métrica de CloudWatch. La alarma especificada en esta plantilla monitorea las comprobaciones de estado de las instancias de Amazon Elastic Compute Cloud (Amazon EC2). Si la alarma ingresa en el estado `ALARM`, crea un OpsItem en 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"}}]
         }
       }
     }
   }
```

Alarma compuesta: utilice la siguiente plantilla de CloudFormation para crear o actualizar una alarma compuesta. Una alarma compuesta consta de múltiples alarmas métricas. Si la alarma ingresa en el estado ALARM, crea un OpsItem en 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"
         }
      }
   }
```

Configuración de alarmas de CloudWatch para crear o actualizar OpsItems (Java)

Esta sección incluye fragmentos de código Java que puede usar para configurar alarmas de CloudWatch para crear o actualizar OpsItems de manera automática. Cada fragmento requiere que especifique un ARN para el parámetro `validSsmActionStr`. Para obtener más información acerca de cómo crear el ARN, consulte Antes de empezar.

Una alarma específica: utilice el siguiente fragmento de código Java para crear o actualizar una alarma de CloudWatch. La alarma especificada en esta plantilla monitorea las verificaciones del estado de las instancias de Amazon EC2. Si la alarma ingresa en el estado `ALARM`, crea un OpsItem en 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);
    }
```

Actualizar todas las alarmas: utilice el siguiente fragmento de código Java para actualizar todas las alarmas de CloudWatch en su Cuenta de AWS para crear OpsItems cuando una alarma ingresa en el estado 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;
            }
        }
    }
```

Crear OpsItems manualmente

Cuando encuentre un problema operativo, puede crear manualmente un OpsItem desde OpsCenter, una herramienta de AWS Systems Manager, para administrar y resolver el problema.

Si configura OpsCenter para la administración entre cuentas, un administrador delegado de Systems Manager o una cuenta de administración de AWS Organizations puede crear OpsItems para cuentas de miembros. Para obtener más información, consulte (Opcional) Configuración manual de OpsCenter para administrar OpsItems de forma centralizada entre cuentas.

Puede crear OpsItems mediante la consola de AWS Systems Manager, la AWS Command Line Interface (AWS CLI) o AWS Tools for Windows PowerShell.

Temas

- Creación manual de OpsItems (consola)
- Creación de OpsItems manualmente (AWS CLI)
- Creación de OpsItems manualmente (PowerShell)

Creación manual de OpsItems (consola)

Puede crear OpsItems de forma manual mediante la consola de AWS Systems Manager. Cuando crea un OpsItem, se muestra en su cuenta de OpsCenter. Si configura OpsCenter para la administración entre cuentas, OpsCenter ofrece al administrador delegado o a la cuenta de administración la opción de crear OpsItems para las cuentas de miembros seleccionadas. Para obtener más información, consulte (Opcional) Configuración manual de OpsCenter para administrar OpsItems de forma centralizada entre cuentas.

Para crear un OpsItem con la consola de AWS Systems Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija OpsCenter.
3. Seleccione Crear OpsItem. Si este botón no aparece, elija la pestaña OpsItems y, a continuación, elija Crear OpsItem.
4. (Opcional) Elija Otra cuenta y, a continuación, elija la cuenta en la que desea crear el OpsItem.

   > **Note**
   >
   > Este paso es obligatorio si crea OpsItems para una cuenta de miembro.

5. En Título, escriba un nombre descriptivo que le ayude a comprender la finalidad del OpsItem.
6. En Source (Origen), ingrese el tipo de recurso de AWS afectado u otro tipo de información de origen para ayudar a los usuarios a comprender el origen del OpsItem.

   > **Note**
   >
   > No se puede editar el campo Origen una vez creado el OpsItem.

7. En Priority (Prioridad), elija el nivel de prioridad (opcional).
8. En Severity (Severidad), elija el nivel de severidad (opcional).
9. En Category (Categoría), elija una categoría (opcional).
10. En Descripción, escriba información acerca de este OpsItem, incluidos (si corresponden) los pasos para reproducir el problema.

    > **Note**
    >
    > La consola admite la mayoría de los formatos de Markdown del campo de descripción OpsItem. Para obtener más información, consulte Uso de Markdown en la consola en la Introducción a la AWS Management Console en la Guía de introducción.

11. En Cadena de deduplicación, ingrese las palabras que el sistema puede usar para comprobar si hay OpsItems duplicados. Para obtener más información sobre las cadenas de deduplicación, consulte Administración de OpsItems duplicados.

12. (Opcional) En Notificaciones, especifique el nombre de recurso de Amazon (ARN) del tema de Amazon SNS donde desea que se envíen notificaciones cuando se actualice este OpsItem. Debe especificar un ARN de Amazon SNS que se encuentre en la misma Región de AWS que el OpsItem.
13. (Opcional) En Recursos relacionados, elija Agregar para especificar el ID o ARN del recurso afectado y los recursos relacionados.
14. Seleccione Crear OpsItem.

Si se realiza correctamente, la página mostrará el OpsItem. Cuando una cuenta de administración o administrador delegado crea un OpsItem para cuentas de miembros seleccionadas, los nuevos OpsItems se muestran en el OpsCenter de las cuentas de administrador y miembros. Para obtener información acerca de cómo configurar las opciones en un OpsItem, consulte Administración de OpsItems.

Creación de OpsItems manualmente (AWS CLI)

En el siguiente procedimiento se describe cómo se crea un OpsItem con AWS Command Line Interface (AWS CLI).

Para crear un OpsItem mediante la AWS CLI

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.
2. Abra la AWS CLI y ejecute el siguiente comando para crear un OpsItem. Reemplace cada *example resource placeholder* con su propia información.

```
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"
```

   Especificar los datos operativos de un archivo

Al crear un OpsItem, puede especificar los datos operativos de un archivo. El archivo debe ser un archivo JSON y su contenido debe tener el siguiente formato.

```
{
  "key_name": {
    "Type": "SearchableString",
    "Value": "Up to 20 KB of data"
  }
}
```

A continuación se muestra un ejemplo.

```
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**
>
> Para obtener información acerca de cómo se ingresan los parámetros con formato JSON en la línea de comandos en diferentes sistemas operativos locales, consulte Uso de comillas con cadenas en AWS CLI en la Guía del usuario de AWS Command Line Interface.

El sistema devuelve información similar a la siguiente.

```
{
    "OpsItemId": "oi-1a2b3c4d5e6f"
}
```

3. Ahora ejecute el siguiente comando para ver detalles sobre el OpsItem que ha creado.

```
aws ssm get-ops-item --ops-item-id ID
```

El sistema devuelve información similar a la siguiente.

```
{
    "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. Ejecute el siguiente comando para actualizar el OpsItem. Este comando cambia el estado de `Open` (valor predeterminado) a `InProgress`.

```
aws ssm update-ops-item --ops-item-id ID --status InProgress
```

El comando no genera ningún resultado.

5. Ejecute el siguiente comando de nuevo para comprobar que el estado cambia a `InProgress`.

```
aws ssm get-ops-item --ops-item-id ID
```

Ejemplos de creación de un OpsItem

En los siguientes ejemplos de código se muestra cómo crear un OpsItem mediante el portal de administración de Linux, macOS o Windows.

Portal de administración de Linux o macOS

El siguiente comando crea un OpsItem cuando un disco de instancia de Amazon Elastic Compute Cloud (Amazon EC2) está lleno.

```
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"
```

El comando siguiente utiliza la clave `/aws/resources` en `OperationalData` para crear un OpsItem con un recurso relacionado de 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"
```

El siguiente comando utiliza la clave `/aws/automations` en `OperationalData` para crear un OpsItem que especifique el documento `AWS-ASGEnterStandby` como manual de procedimientos de Automatización asociado.

```
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

El siguiente comando crea un OpsItem cuando una instancia de Amazon Relational Database Service (Amazon RDS) no responde.

```
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"
```

El comando siguiente utiliza la clave `/aws/resources` en `OperationalData` para crear un OpsItem con un recurso relacionado con la instancia de Amazon 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\"}}
```

El siguiente comando utiliza la clave `/aws/automations` en `OperationalData` para crear un OpsItem que especifique el manual de procedimientos `AWS-RestartEC2Instance` como manual de procedimientos de Automatización asociado.

```
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\"}}
```

Creación de OpsItems manualmente (PowerShell)

En el siguiente procedimiento se describe cómo se crea un OpsItem con AWS Tools for Windows PowerShell.

Para crear un OpsItem con AWS Tools for Windows PowerShell

1. Abra AWS Tools for Windows PowerShell y ejecute el siguiente comando para especificar sus credenciales.

   ```
   Set-AWSCredentials –AccessKey key-name –SecretKey key-name
   ```

2. Ejecute el siguiente comando para establecer la Región de AWS para la sesión de PowerShell.

   ```
   Set-DefaultAWSRegion -Region Region
   ```

3. Ejecute el siguiente comando para crear un OpsItem nuevo. Reemplace cada *example resource placeholder* con su propia información. Este comando especifica un runbook de Automatización de Systems Manager para solucionar este 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
   ```

   Si se realiza correctamente, el comando genera el ID del OpsItem nuevo.

En el siguiente ejemplo, se especifica el nombre de recurso de Amazon (ARN) de una instancia de Amazon Elastic Compute Cloud (Amazon EC2) en mal estado.

```
$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
```

Administración de OpsItems

OpsCenter, una herramienta de AWS Systems Manager, hace un seguimiento de los OpsItems desde su creación hasta su resolución. Si configura OpsCenter para la administración entre cuentas, un administrador delegado o una cuenta de gestión puede administrar los OpsItems desde su cuenta. Para obtener más información, consulte (Opcional) Configuración manual de OpsCenter para administrar OpsItems de forma centralizada entre cuentas.

Puede ver y administrar OpsItems mediante las siguientes páginas de la consola de Systems Manager:

- Resumen: muestra un recuento de OpsItems abiertos y en curso, un recuento de OpsItems por origen y edad, e información operativa. Puede filtrar los OpsItems por origen y estado de OpsItems.

- OpsItems: muestra una lista de OpsItems con varios campos de información, como el título, el ID, la prioridad, la descripción, el origen del OpsItem y la fecha y hora de la última actualización. Con esta página, puede crear OpsItems, configurar orígenes, cambiar el estado de un OpsItem y filtrar OpsItems por nuevos incidentes de forma manual. Puede elegir un OpsItem para mostrar su página Detalles de OpsItems.

- Detalles de OpsItem: proporciona información y herramientas detalladas que puede utilizar para administrar un OpsItem. La página de detalles de OpsItems tiene las siguientes pestañas:
  - Información general: muestra los recursos relacionados, los manuales de procedimientos que se ejecutaron en los últimos 30 días y una lista de los manuales de procedimientos disponibles

que puede ejecutar. También puede ver OpsItems similares, agregar datos operativos y agregar OpsItems relacionados.

- Detalles de recursos relacionados: muestra información sobre el recurso de varios servicios de AWS. Expanda la sección Resource details (Detalles del recurso) para ver información sobre este recurso tal y como la proporciona el servicio de AWS que lo aloja. También puede activar y desactivar otros recursos relacionados asociados a este OpsItem utilizando la lista Related resources (Recursos relacionados).

Para obtener más información sobre cómo administrar OpsItems, consulte los siguientes temas.

Temas

- Visualización de los detalles de un OpsItem
- Edición de un OpsItem
- Adición de recursos relacionados a un OpsItem
- Adición de OpsItems relacionados a un OpsItem
- Adición de datos operativos a un OpsItem
- Creación de un incidente para un OpsItem
- Administración de OpsItems duplicados
- Análisis de la información operativa para reducir OpsItems
- Visualización de registros e informes de OpsCenter

Visualización de los detalles de un OpsItem

Para obtener una vista completa de un OpsItem, utilice la página Detalles de OpsItem de la consola de OpsCenter. La página Información general muestra la siguiente información:

- Detalles de OpsItems: muestra la información general del OpsItem seleccionado.
- Recursos relacionados: un recurso relacionado es el recurso afectado o el recurso que ha iniciado el evento que creó el OpsItem.
- Ejecuciones automatizadas en los últimos 30 días: una lista de los manuales de procedimientos que se ejecutaron en los últimos 30 días.
- Manuales de procedimientos: puede elegir un manual de procedimientos de una lista de manuales de procedimientos disponibles.

- OpsItems similares: esta es una lista de OpsItems generada por el sistema que puede estar relacionada o ser de su interés. Para generar la lista, el sistema examina los títulos y las descripciones de todos los OpsItems y devuelve OpsItems que utilizan palabras similares.
- Datos operativos: los datos operativos son datos personalizados que proporcionan información detallada acerca del OpsItem. Por ejemplo, puede especificar archivos de registro, cadenas de error, claves de licencia, sugerencias para resolver problemas u otros datos pertinentes.
- OpsItems relacionados: puede especificar los ID de los OpsItems que estén relacionados de alguna manera con el OpsItem actual.
- Detalles de los recursos relacionados: muestra proveedores de datos, incluidas métricas y alarmas de Amazon CloudWatch, registros de AWS CloudTrail y detalles de AWS Config.

Para ver los detalles de una OpsItem

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija OpsCenter.
3. Seleccione un OpsItem para ver sus detalles.

Edición de un OpsItem

En la sección Detalles de OpsItem, se incluye información sobre el OpsItem, que incluye la descripción, el título, el origen, el ID del OpsItem y el estado.
Puede editar el OpsItem de forma individual o puede seleccionar varios OpsItems y editar los siguientes campos: Estado, Prioridad, Gravedad, Categoría.

Cuando Amazon EventBridge crea un OpsItem, rellena los campos Título, Origen y Descripción. Puede editar los campos Título y Descripción, pero no puede editar el campo Origen.

> **Note**
>
> La consola admite la mayoría de los formatos de Markdown del campo de descripción de OpsItem. Para obtener más información, consulte Uso de Markdown en la consola en la Introducción a la AWS Management Console en la Guía de introducción.

Por lo general, puede editar los siguientes datos configurables para un OpsItem:

- Título: nombre del OpsItem. El origen crea el título del OpsItem.
- Descripción: información sobre este OpsItem, incluidos (si corresponde) los pasos para reproducir el problema.
- Estado: estado de un OpsItem. Para obtener una lista de valores de estado válidos, consulte el estado de OpsItem en la referencia de la API de AWS Systems Manager.
- Prioridad: la prioridad de un OpsItem puede estar entre 1 y 5. Recomendamos que su organización determine qué significa cada nivel de prioridad y elabore un acuerdo de nivel de servicio correspondiente para cada nivel.
- Gravedad: la gravedad de un OpsItem puede estar entre 1 y 4, donde 1 es crítica, 2 es alta, 3 es media y 4 es baja.
- Categoría: la categoría de un OpsItem puede ser la disponibilidad, el costo, el rendimiento, la recuperación o la seguridad.
- Notificaciones: cuando edita una OpsItem, puede especificar el nombre de recurso de Amazon (ARN) de un tema de Amazon Simple Notification Service en el campo Notificaciones. Al especificar un ARN, se asegura de que todas las partes interesadas reciban una notificación cuando se edite el OpsItem, incluido un cambio de estado. Para obtener más información, consulte la Guía para desarrolladores de Amazon Simple Notification Service.

> **Important**
>
> El tema de Amazon SNS debe existir en la misma Región de AWS del OpsItem. Si el tema y el OpsItem están en regiones diferentes, el sistema devuelve un error.

OpsCenter tiene integración bidireccional con AWS Security Hub. Cuando actualiza un estado de OpsItem y la gravedad relacionada con un hallazgo de seguridad, esos cambios se envían automáticamente a Security Hub para garantizar que siempre vea la información más reciente y correcta.

Cuando se crea un OpsItem a partir de un resultado de Security Hub, los metadatos del Security Hub se agregan automáticamente al campo de datos operativos delOpsItem. Si se eliminan estos metadatos, las actualizaciones bidireccionales dejan de funcionar.

Para editar los detalles de un OpsItem

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.

2. En el panel de navegación, elija OpsCenter.
3. Elija un ID de OpsItem para abrir la página de detalles o elija varios OpsItems. Si elige varios OpsItems, solo puedes editar el estado, la prioridad, la severidad o la categoría. Si edita varios OpsItems, OpsCenter actualiza y guarda los cambios tan pronto como elija el nuevo estado, la prioridad, la severidad o la categoría.
4. En la sección OpsItem details (Detalles del OpsItem), elija Edit (Editar).
5. Edite los detalles del OpsItem siguiendo los requisitos y las directrices especificados por su organización.
6. Cuando haya terminado, elija Save.

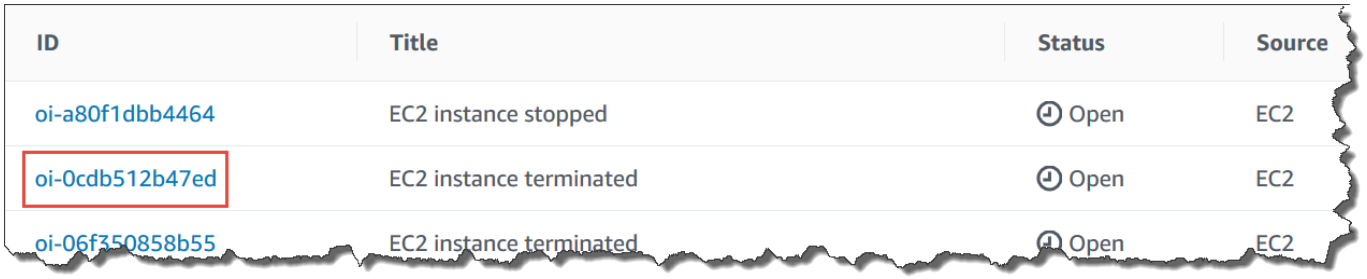
Adición de recursos relacionados a un OpsItem

Cada OpsItem incluye una sección Recursos relacionados que enumera el nombre de recurso de Amazon (ARN) del recurso relacionado. Un recurso relacionado es el recurso de AWS afectado que debe investigarse.

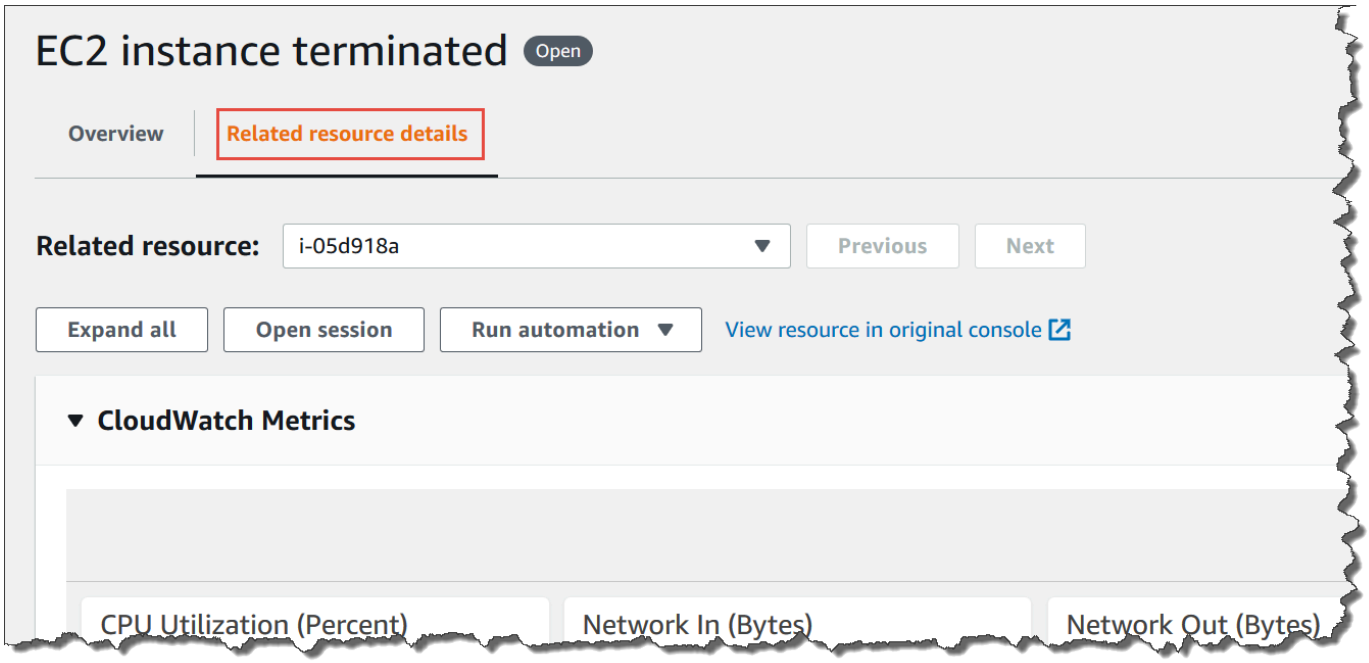
Si Amazon EventBridge crea el OpsItem, el sistema rellena automáticamente el OpsItem con el ARN del recurso. Puede especificar de forma manual los ARN de los recursos relacionados. Para algunos tipos de ARN, OpsCenter crea automáticamente un enlace profundo que muestra detalles sobre el recurso directamente en la consola OpsCenter. Por ejemplo, si especifica el ARN de una instancia de Amazon Elastic Compute Cloud (Amazon EC2) como un recurso relacionado, OpsCenter entonces obtiene los detalles de dicha instancia de EC2. Esto le permite ver información detallada sobre los recursos de AWS afectados sin tener que salir de OpsCenter.

Para ver y agregar recursos relacionados a un OpsItem

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija OpsCenter.
3. Elija la pestaña OpsItems.
4. Elija un ID de OpsItem.



5. Para ver información acerca de los recursos afectados, elija la pestaña Related resources details (Detalles de los recursos relacionados).



Esta pestaña muestra información sobre el recurso de varios Servicios de AWS. Expanda la sección Resource details (Detalles del recurso) para ver información sobre este recurso tal y como la proporciona el Servicio de AWS que lo aloja. También puede activar y desactivar otros recursos relacionados asociados a este OpsItem utilizando la lista Related resources (Recursos relacionados).

6. Para añadir más recursos relacionados, elija la pestaña Overview (Información general).
7. En la sección Related resources (Recursos relacionados), seleccione Add (Añadir).
8. En Resource type (Tipo de recurso), elija un recurso de la lista.
9. En Resource ID (ID de recurso), ingrese el ID o el nombre de recurso de Amazon (ARN). El tipo de información que elija depende del recurso que ha elegido en el paso anterior.

> **Note**
>
> Puede añadir los ARN de recursos relacionados adicionales manualmente. Cada OpsItem puede enumerar un máximo de 100 ARN de recursos relacionados.

En la siguiente tabla, se muestran los tipos de recursos que crean automáticamente enlaces profundos a recursos relacionados.

Tipos de recursos admitidos

Nombre del recurso	Formato de ARN
Certificado de AWS Certificate Manager	`arn:aws:acm: region:account-id :certificate/ certificate-id`
Grupo de Amazon EC2 Auto Scaling	`arn:aws:autoscaling: region:account-id :autoScalingGroup: groupid:autoScalingGroupName/ groupfriendlyname`
Distribución de Amazon CloudFront	`arn:aws:cloudfront:: account-id :*`
Pila de AWS CloudFormation	`arn:aws:cloudformation: region:account-id :stack/stackname /additionalidentifier`
Alarma de Amazon CloudWatch	`arn:aws:cloudwatch: region:account-id :alarm:alarm-name`
Registro de seguimiento de AWS CloudTrail	`arn:aws:cloudtrail: region:account-id :trail/trailname`
Proyecto de AWS CodeBuild	`arn:aws:codebuild: region:account-id :resourcetype /resource`

Nombre del recurso	Formato de ARN
AWS CodePipeline	`arn:aws:codepipeline:  region:account-id :resource-specifier`
Información de Amazon DevOps Guru	`arn:aws:devops-guru:  region:account-id :insight/ proactive or reactive/resource-id`
Tabla de Amazon DynamoDB	`arn:aws:dynamodb: region:account-id :table/tablename`
Gateway de cliente de Amazon Elastic Compute Cloud (Amazon EC2)	`arn:aws:ec2: region:account-id :customer-gateway/  cgw-id`
IP elástica de Amazon EC2	`arn:aws:ec2: region:account-id :eip/eipalloc-id`
Host dedicado de Amazon EC2	`arn:aws:ec2: region:account-id :dedicated-host/ host-id`
Instancia de Amazon EC2	`arn:aws:ec2: region:account-id :instance/ instance-id`
Gateway de Internet de Amazon EC2	`arn:aws:ec2: region:account-id :internet-gateway/  igw-id`
Lista de control de acceso de la red (ACL de la red) de Amazon EC2	`arn:aws:ec2: region:account-id :network-acl/ nacl-id`
Interfaz de red de Amazon EC2	`arn:aws:ec2: region:account-id :network-interface/  eni-id`

Nombre del recurso	Formato de ARN
Tabla de enrutamiento de Amazon EC2	`arn:aws:ec2: region:account-id :route-table/ route-table-id`
Grupo de seguridad de Amazon EC2	`arn:aws:ec2: region:account-id :security-group/ security-group-id`
Subred de Amazon EC2	`arn:aws:ec2: region:account-id :subnet/subnet-id`
Volumen de Amazon EC2	`arn:aws:ec2: region:account-id :volume/volume-id`
VPC de Amazon EC2	`arn:aws:ec2: region:account-id :vpc/vpc-id`
Conexión de VPN de Amazon EC2	`arn:aws:ec2: region:account-id :vpn-connection/ vpn-id`
Gateway de VPN de Amazon EC2	`arn:aws:ec2: region:account-id :vpn-gateway/ vgw-id`
Aplicación de 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`

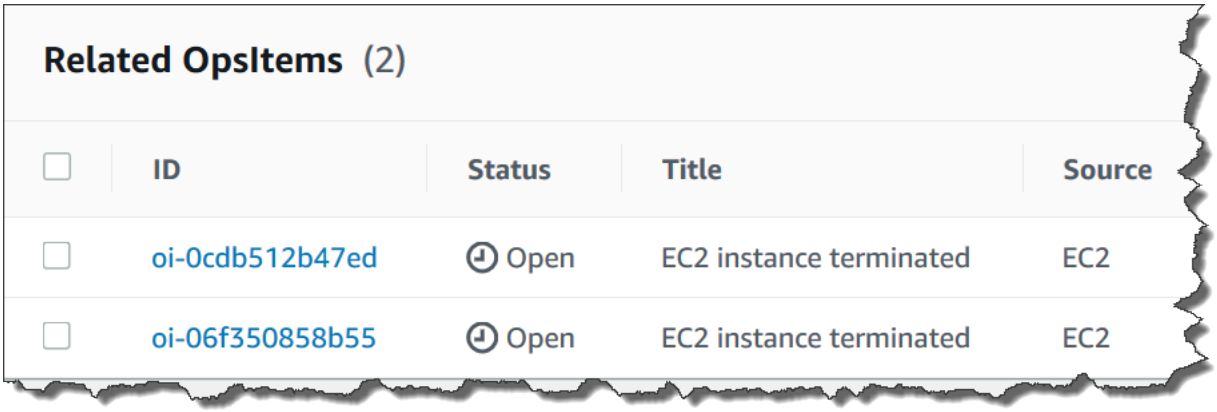
Nombre del recurso	Formato de ARN
Elastic Load Balancing (Application Load Balancer)	`arn:aws:elasticloadbalancing: region:account-id :loadbalancer/app/ load-balancer-name /load-balancer-id`
Elastic Load Balancing (Network Load Balancer)	`arn:aws:elasticloadbalancing: region:account-id :loadbalancer/net/ load-balancer-name /load-balancer-id`
Grupo de AWS Identity and Access Management (IAM)	`arn:aws:iam:: account-id :group/group-name`
Política de IAM	`arn:aws:iam:: account-id :policy/policy-name`
rol de IAM	`arn:aws:iam:: account-id :role/role-name`
Usuario de IAM	`arn:aws:iam:: account-id :user/user-name`
Función de AWS Lambda	`arn:aws:lambda: region:account-id :function: function-name`
Clúster de Amazon Relational Database Service (Amazon RDS)	`arn:aws:rds: region:account-id :cluster: db-cluster-name`
Instancia de base de datos de Amazon RDS	`arn:aws:rds: region:account-id :db:db-instance-name`

Nombre del recurso	Formato de ARN
Suscripción a Amazon RDS	`arn:aws:rds: region:account-id :es:subscription-name`
Grupo de seguridad de Amazon RDS	`arn:aws:rds: region:account-id :secgrp:security-group-name`
Instantánea de clúster de Amazon RDS	`arn:aws:rds: region:account-id :cluster-snapshot: cluster-snapshot-name`
Grupo de subredes de Amazon RDS	`arn:aws:rds: region:account-id :subgrp:subnet-group-name`
Clúster de Amazon Redshift	`arn:aws:redshift: region:account-id :cluster: cluster-name`
Grupo de parámetros de Amazon Redshift	`arn:aws:redshift: region:account-id :parametergroup: parameter-group-name`
Grupo de seguridad de Amazon Redshift	`arn:aws:redshift: region:account-id :securitygroup: security-group-name`
Instantánea de clúster de Amazon Redshift	`arn:aws:redshift: region:account-id :snapshot: cluster-name /snapshot-name`
Grupo de subredes de Amazon Redshift	`arn:aws:redshift: region:account-id :subnetgroup: subnet-group-name`
Bucket de Amazon Simple Storage Service (Amazon S3)	`arn:aws:s3::: bucket_name`

Nombre del recurso	Formato de ARN
Registro de AWS Config del inventario de nodos administrados de AWS Systems Manager	`arn:aws:ssm: region:account-id :managed-instance-inventory / node_id`
Asociación de State Manager de Systems Manager	`arn:aws:ssm: region:account-id :association/ association_ID`

Adición de OpsItems relacionados a un OpsItem

Al utilizar OpsItems relacionados de la página Detalles de OpsItems, puede investigar los problemas de las operaciones y proporcionar contexto para un problema. Los OpsItems pueden estar relacionados de diferentes maneras, incluida una relación principal-secundaria entre OpsItems, una causa raíz o un duplicado. Puede asociar un OpsItem con otro para mostrarlo en la sección OpsItem relacionados. Puede especificar un máximo de 10 ID para otros OpsItems que estén relacionados con el OpsItem actual.



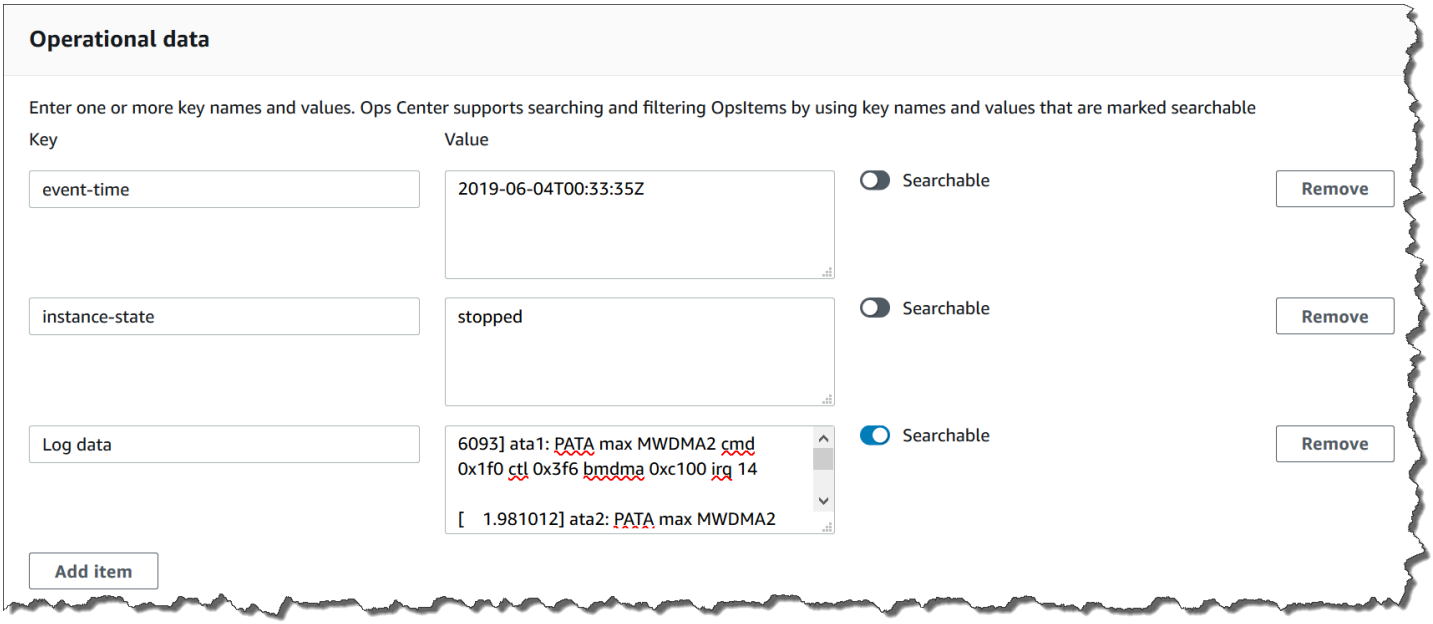
Para agregar un OpsItem

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija OpsCenter.
3. Elija un ID de OpsItem para abrir la página de detalles.
4. En la sección Related OpsItem (relacionado), seleccione Add (Añadir).
5. Para OpsItem ID (ID de), especifique un ID.

6. Seleccione Añadir.

Adición de datos operativos a un OpsItem

Los datos operativos son datos personalizados que proporcionan información detallada acerca de un OpsItem. Puede introducir varios pares clave-valor de datos operativos. Por ejemplo, puede especificar archivos de registro, cadenas de error, claves de licencia, sugerencias para resolver problemas u otros datos pertinentes. La longitud máxima de la clave puede ser de 128 caracteres y el tamaño máximo del valor puede ser de 20 KB.



Puede hacer que otros usuarios de la cuenta puedan realizar búsquedas en los datos o bien restringir el acceso de búsqueda. Poder hacer búsquedas en los datos significa que todos los usuarios con acceso a la página Información general de OpsItem (tal como indica la operación de la API DescribeOpsItems) pueden ver y realizar búsquedas en los datos especificados. Los datos operativos en los que no se pueden realizar búsquedas solo los pueden ver los usuarios con acceso al OpsItem (tal como indica la operación de la API GetOpsItem).

Para añadir datos operativos a un OpsItem

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija OpsCenter.
3. Elija un ID de OpsItem para abrir la página de detalles.
4. Expanda la sección Datos operativos.

5. Si no existen datos operativos para el OpsItem, seleccione Agregar. Si los datos operativos ya existen para el OpsItem, elija Manage (Administrar).

   Después de crear los datos operativos, puede editar la clave y el valor, eliminar los datos operativos o añadir más pares clave-valor. Para hacerlo, elija Manage (Administrar).

6. En Key (Clave), especifique una o varias palabras para ayudar a los usuarios a comprender la finalidad de los datos.

   > **Important**
   >
   > Las claves de datos operativos no pueden comenzar con lo siguiente:`amazon`, `aws`, `amzn`, `ssm`, `/amazon`, `/aws`, `/amzn`, `/ssm`.

7. En Value (Valor), especifique los datos.
8. Seleccione Save.

> **Note**
>
> Puede filtrar OpsItems utilizando el operador Operational data (Datos operativos) en la página OpsItems. En el cuadro Buscar, elija Datos operativos y, a continuación, escriba un par clave-valor en JSON. Para escribir el par clave-valor, utilice el siguiente formato: `{"key":"key_name","value":"a_value"}`

Creación de un incidente para un OpsItem

Utilice el siguiente procedimiento para crear manualmente un incidente de modo que un OpsItem lo rastree y administre en Administrador de incidentes de AWS Systems Manager, que es una herramienta de AWS Systems Manager. Un incidente es cualquier interrupción no planificada o reducción de la calidad de los servicios. Para obtener más información acerca de Administrador de incidentes, consulte the section called “Integración de OpsCenter con otros Servicios de AWS”.

Para crear un incidente manualmente para un OpsItem

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija OpsCenter.

3. Si Incident Manager creó un OpsItem para usted, elíjalo y vaya al paso 5. Si no, elija Create OpsItem (Crear OpsItem) y complete el formulario. Si este botón no aparece, elija la pestaña OpsItems y, a continuación, elija Create OpsItem (Crear OpsItem).
4. Si ha creado un OpsItem, ábralo.
5. Elija Start Incident (Iniciar Incident).
6. En Plan de respuesta, elija el plan de respuesta de Administrador de incidentes que desea asignar a este incidente.
7. En Title (Título), ingrese un nombre descriptivo para ayudar a otros miembros del equipo a comprender la naturaleza del incidente (opcional). Si no ingresa un nuevo título, OpsCenter crea el OpsItem y el incidente correspondiente en Incident Manager mediante el título del plan de respuesta.
8. En Incident impact (Impacto del incidente), elija un nivel de impacto para este incidente (opcional). Si no elige un nivel de impacto, OpsCenter crea el OpsItem y el incidente correspondiente en Incident Manager mediante el nivel de impacto en el plan de respuesta.
9. Elija Iniciar.

Administración de OpsItems duplicados

OpsCenter puede recibir varios OpsItems duplicados para un solo origen desde varios Servicios de AWS. OpsCenter utiliza una combinación de lógica integrada y cadenas de deduplicación configurables para evitar la creación de OpsItems duplicados. AWS Systems Manager aplica la lógica integrada de deduplicación cuando se llama a la operación de la API CreateOpsItem.

AWS Systems Manager utiliza la siguiente lógica de deduplicación:

1. Cuando crea el OpsItem, Systems Manager crea y almacena un hash basado en la cadena de desduplicación y en el recurso que desencadenó el OpsItem.
2. Al realizar otra solicitud para crear un OpsItem, el sistema comprueba la cadena de deduplicación de la nueva solicitud.
3. Si existe un hash coincidente con esta cadena de deduplicación, Systems Manager comprueba el estado del OpsItem existente. Si el estado del OpsItem existente es abierto o en curso, no se crea el OpsItem. Si se resuelve el OpsItem existente, Systems Manager crea un nuevo OpsItem.

Después de crear un OpsItem, no puede editar ni cambiar las cadenas de desduplicación de dicho OpsItem.

Para administrar OpsItems duplicados, puede realizar lo siguiente:

- Edite la cadena de deduplicación de una regla de Amazon EventBridge dirigida a OpsCenter. Para obtener más información, consulte Edición de una cadena de deduplicación en una regla de EventBridge predeterminada.
- Especifique una cadena de deduplicación al crear un nuevo OpsItem de forma manual. Para obtener más información, consulte Especificación de una cadena de deduplicación mediante la AWS CLI.
- Revise y resuelva los OpsItems duplicados con información operativa. Puede utilizar manuales de procedimientos para resolver OpsItems duplicados.

  Para ayudarlo a resolver OpsItems duplicados y reducir el número de OpsItems creados en un origen, Systems Manager proporciona manuales de procedimientos de automatización. Para obtener más información, consulte Resolución de OpsItems duplicados basados en información.

Edición de una cadena de deduplicación en una regla de EventBridge predeterminada

Utilice el siguiente procedimiento para especificar una cadena de desduplicación para una regla de EventBridge que tiene como destino OpsCenter.

Para editar una cadena de deduplicación para una regla de EventBridge

1. Inicie sesión en AWS Management Console y abra la consola de Amazon EventBridge en https://console.aws.amazon.com/events/.
2. En el panel de navegación, seleccione Reglas.
3. Elija una regla y, a continuación, elija Edit (Editar).
4. Vaya a la página Select target(s) (Seleccionar destinos).
5. En la sección Additional settings (Ajustes adicionales), elija Configure input transformer (Configurar transformador de entrada).
6. En la caja Template (Plantilla), busque la entrada JSON `"operationalData": { "/aws/dedup"` y las cadenas de desduplicación que desea editar.

   La entrada de la cadena de desduplicación en las reglas de EventBridge utiliza el siguiente formato JSON.

```
"operationalData": { "/aws/dedup": {"type": "SearchableString","value":
 "{\"dedupString\":\"Words the system should use to check for duplicate
 OpsItems\"}"}}
```

A continuación se muestra un ejemplo.

```
"operationalData": { "/aws/dedup": {"type": "SearchableString","value":
 "{\"dedupString\":\"SSMOpsCenter-EBS-volume-performance-issue\"}"}}
```

7. Edite las cadenas de deduplicación y, a continuación, elija Confirmar.
8. Elija Siguiente.
9. Elija Siguiente.
10. Elija Actualizar regla.

Especificación de una cadena de deduplicación mediante la AWS CLI

Puede especificar una cadena de deduplicación al crear manualmente un OpsItem nuevo mediante la consola de AWS Systems Manager o la AWS CLI. Para obtener información acerca de cómo especificar cadenas de desduplicación al crear manualmente un OpsItem en la consola, consulte Crear OpsItems manualmente. Si usa la AWS CLI, puede ingresar la cadena de deduplicación para el parámetro `OperationalData`. La sintaxis del parámetro utiliza JSON, tal como se muestra en el ejemplo siguiente.

```
--operational-data '{"/aws/dedup":{"Value":"{\"dedupString\": \"Words the system should
 use to check for duplicate OpsItems\"}","Type":"SearchableString"}}'
```

A continuación se muestra un comando de ejemplo que especifica una cadena de desduplicación 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"
```

Análisis de la información operativa para reducir OpsItems

La información operativa de OpsCenter muestra información sobre los OpsItems duplicados. OpsCenter analiza OpsItems de manera automática su cuenta y genera tres tipos de información. Puede ver esta información en la sección Información operativa de la pestaña Resumen de OpsCenter.

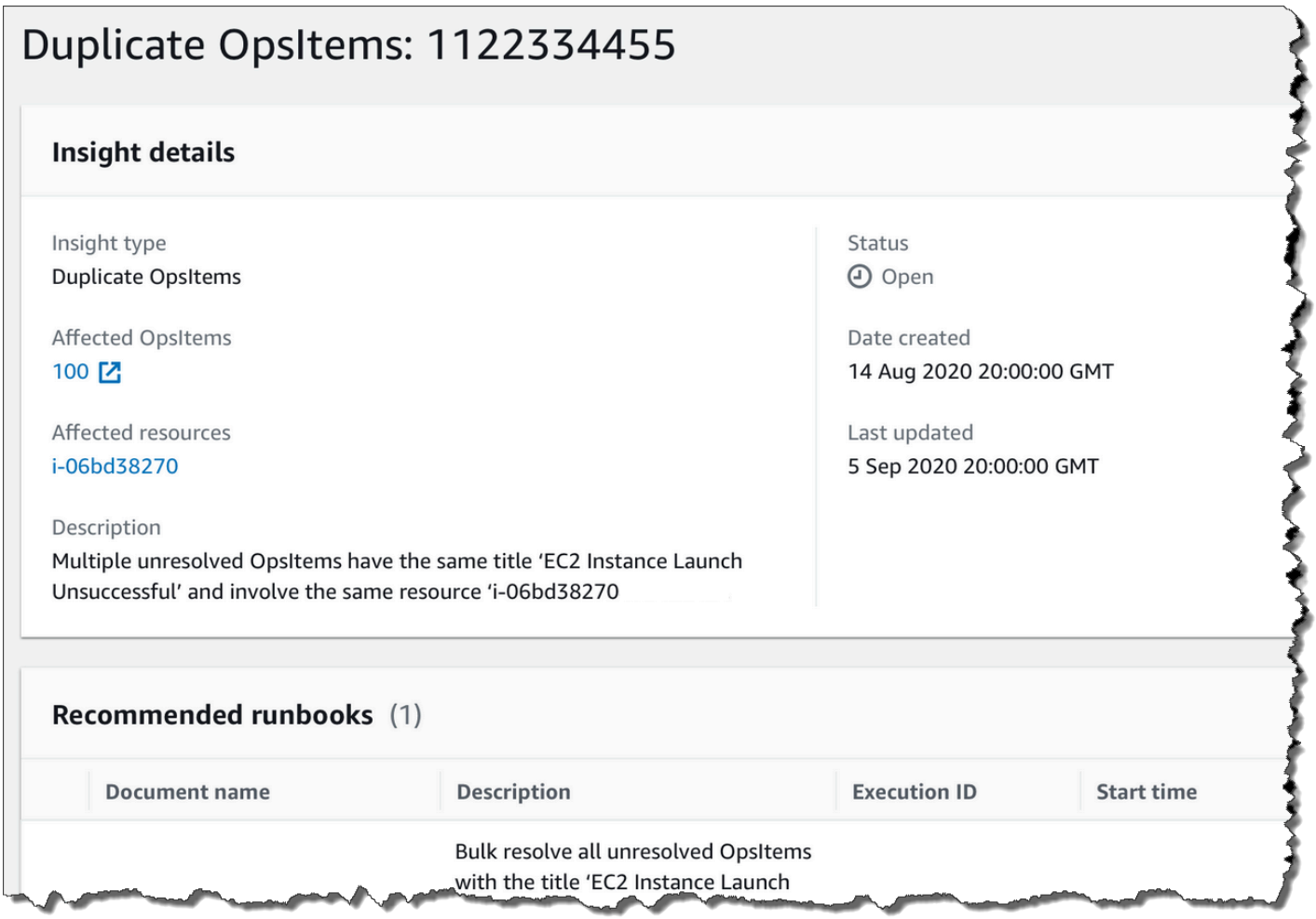
- OpsItems duplicados: se genera un elemento de información cuando ocho o más OpsItems tienen el mismo título para el mismo recurso.
- Títulos más comunes: se genera un elemento de información cuando más de 50 OpsItems tienen el mismo título.
- Recursos que generan la mayor cantidad de OpsItems: se genera un elemento de información cuando un recurso de AWS tiene más de 10 OpsItems abiertos. Esta información y los recursos correspondientes se muestran en la tabla Recursos que generan la mayor cantidad de OpsItems en la pestaña Resumen de OpsCenter. Los recursos se enumeran en orden decreciente de recuento de OpsItem.

> **Note**
>
> OpsCenter crea información de Recursos que generan la mayor cantidad de OpsItems para los siguientes tipos de recursos:
>
> - Instancias de Amazon Elastic Compute Cloud (Amazon EC2)

- Grupos de seguridad de Amazon EC2
- Grupo de Amazon EC2 Auto Scaling
- Base de datos de Amazon Relational Database Service (Amazon RDS)
- Clúster de Amazon RDS
- Función AWS Lambda
- Tabla de Amazon DynamoDB
- Equilibrador de carga de Elastic Load Balancing
- Clúster de Amazon Redshift
- Certificado de AWS Certificate Manager
- Volumen de Amazon Elastic Block Store

OpsCenter aplica un límite de 15 elementos de información por tipo. Si un tipo alcanza este límite, OpsCenter deja de mostrar información para ese tipo. Para ver información adicional, debe resolver todos los OpsItems asociados a un OpsInsight de ese tipo. Si un elemento de información pendiente no puede mostrarse en la consola debido al límite de 15 elementos de información, ese elemento se podrá ver después de que se cierre otro.

Cuando elige un elemento de información, OpsCenter muestra datos sobre los OpsItems y los recursos afectados. En la siguiente captura de pantalla se muestra un ejemplo con los detalles de información de OpsItem duplicada.



La información operativa está desactivada de forma predeterminada. Para obtener más información sobre cómo trabajar con información operativa, consulte los siguientes temas.

Temas

- Habilitación de la información operativa
- Resolución de OpsItems duplicados basados en información
- Desactivación de la información operativa

Habilitación de la información operativa

Puede habilitar la información operativa en la página de OpsCenter en la consola de Systems Manager. Al habilitar la información operativa, Systems Manager crea un rol de AWS Identity and Access Management (IAM) vinculado a un servicio denominado `AWSServiceRoleForAmazonSSM_OpsInsights`. Un rol vinculado a un servicio es un tipo único de rol de IAM que está vinculado directamente a Systems Manager. Los roles vinculados a

servicios están predefinidos e incluyen todos los permisos que el servicio requiere para llamar a otros Servicios de AWS en su nombre. Para obtener más información sobre el rol vinculado al servicio de `AWSServiceRoleForAmazonSSM_OpsInsights`, consulte Uso de roles para crear OpsItems con información operativa en Systems Manager OpsCenter.

> **Note**
>
> Tenga en cuenta la siguiente información importante:
>
> - Los costos de la información operativa se cargarán en su Cuenta de AWS. Para más información, consulte Precios de AWS Systems Manager.
> - OpsCenter actualiza información periódicamente mediante un proceso por lotes. Esto significa que la lista de información que se muestra en OpsCenter puede estar desincronizada.

Utilice el siguiente procedimiento para habilitar y ver la información operativa en OpsCenter.

Para habilitar y ver la información operativa

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija OpsCenter.
3. En el cuadro de mensaje La información operativa está disponible, elija Habilitar. Si no ve este mensaje, desplácese hacia abajo hasta la sección Información operativa y elija Habilitar.
4. Luego de habilitar esta característica, en la pestaña Resumen, desplácese hacia abajo hasta la sección Información operativa.
5. Para ver una lista filtrada de información, elija el enlace situado junto a Duplicar OpsItems, Títulos más comunes o Recursos que generan la mayoría de OpsItems. Para ver toda la información, elija View all operational insights (Ver toda la información operativa).
6. Elija un ID de información para ver más información.

Resolución de OpsItems duplicados basados en información

Para resolver la información, primero debe resolver todos los OpsItems asociados con la información. Puede utilizar el manual de procedimientos `AWS-BulkResolveOpsItemsForInsight` para resolver OpsItems asociados con la información.

Para ayudarlo a resolver OpsItems duplicados y reducir el número de OpsItems creados en un origen, Systems Manager proporciona los siguientes manuales de procedimientos de automatización:

- El manual de procedimientos `AWS-BulkResolveOpsItems` resuelve OpsItems que coincidan con un filtro especificado.
- El manual de procedimientos `AWS-AddOpsItemDedupStringToEventBridgeRule` agrega una cadena de deduplicación para todos los destinos de OpsItem asociados a una regla de Amazon EventBridge específica. Este manual de procedimientos no agrega una cadena de deduplicación si una regla ya tiene una.
- `AWS-DisableEventBridgeRule` desactiva una regla en EventBridge si la regla genera decenas o cientos de OpsItems.

Resolución de una información operativa

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija OpsCenter.
3. En la pestaña Overview (Información general), desplácese hacia abajo hasta Operational insights (Información operativa).
4. Elija Ver toda la información operativa.
5. Elija un ID de información para ver más información.
6. Elija un manual de procedimientos y, luego, Ejecutar.

Desactivación de la información operativa

Cuando desactiva la información operativa, el sistema deja de crear información nueva y deja de mostrar información en la consola. Cualquier información activa permanece sin cambios en el sistema, aunque no se mostrará en la consola. Si vuelve a habilitar esta característica, el sistema muestra información no resuelta previamente y comienza a crear nueva información. Utilice el siguiente procedimiento para desactivar la información operativa.

Para desactivar la información operativa

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.

2. En el panel de navegación, elija OpsCenter.
3. Elija Configuración.
4. En la sección Operational insights (Información operativa), elija Edit (Editar) y, a continuación, active la opción Disable (Desactivar).
5. Seleccione Guardar.

Visualización de registros e informes de OpsCenter

AWS CloudTrail registra las llamadas a la API de AWS Systems Manager OpsCenter a la consola, a la AWS Command Line Interface (AWS CLI) y al SDK. Puede ver la información en la consola de CloudTrail o en un bucket de Amazon Simple Storage Service (Amazon S3). Amazon S3 utiliza un bucket para almacenar todos los registros de CloudTrail de su cuenta.

Los registros de las acciones de OpsCenter muestran las actividades de creación, actualización, obtención y descripción de un OpsItem. Para obtener más información acerca de cómo ver y utilizar los registros de CloudTrail de la actividad de Systems Manager, consulte Registro de llamadas a la API de AWS Systems Manager con AWS CloudTrail.

AWS Systems Manager OpsCenter le proporciona la siguiente información sobre OpsItems:

- Resumen de estado de OpsItem: proporciona un resumen de OpsItems por estado (Abierto y en curso, Abierto o En curso).
- Orígenes con más OpsItems abiertos: proporciona un desglose de los Servicios de AWS principales con OpsItems abiertos.
- OpsItems por origen y antigüedad: proporciona un recuento de los OpsItems agrupados por origen y días desde su creación.

Para ver el resumen del informe de OpsCenter

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija OpsCenter.
3. En la página Información general de OpsItems, seleccione Resumen.
4. En OpsItems by source and age (OpsItems por origen y antigüedad), elija la barra de búsqueda para filtrar los OpsItems por Source (Origen). Utilice la lista para filtrar por Status (Estado).

Elimine OpsItems

Puede eliminar un OpsItem individual llamando a la operación de la API Delete OpsItem mediante la AWS Command Line Interface o el AWS SDK. No se puede eliminar un OpsItem en la AWS Management Console. Para eliminar un OpsItem, su usuario, grupo o rol de AWS Identity and Access Management (IAM) debe tener permiso de administrador, o bien hay que tener permiso para llamar a la operación de la API `DeleteOpsItem`.

Important

Tenga en cuenta la siguiente información importante sobre esta operación.

- Eliminar un OpsItem es irreversible. No se puede restaurar un OpsItem eliminado.
- Esta operación utiliza un modelo de coherencia eventual, lo que significa que el sistema puede tardar unos minutos en completarla. Si elimina un OpsItem y llama inmediatamente a Get OpsItem, por ejemplo, es posible que el OpsItem eliminado siga apareciendo en la respuesta.
- Esta operación es idempotente. El sistema no genera ninguna excepción si se llama repetidamente a esta operación para el mismo OpsItem. Si la primera llamada se realiza correctamente, todas las llamadas adicionales devuelven la misma respuesta correcta que la primera llamada.
- Esta operación no permite realizar llamadas entre cuentas. Una cuenta de administrador delegado o de administración no puede eliminar OpsItems de otras cuentas, aunque se haya configurado OpsCenter para la administración entre cuentas. Para obtener más información sobre la administración entre cuentas, consulte (Opcional) Configuración de OpsCenter para administrar OpsItems de forma centralizada entre cuentas.
- Si aparece `OpsItemLimitExceededException`, puede eliminar uno o varios OpsItems para que el número total de OpsItems quede por debajo de los límites de las cuotas. Para obtener más información sobre esta excepción, consulte Solución de problemas con OpsCenter.

Eliminación de un OpsItem

Utilice el siguiente procedimiento para eliminar un OpsItem.

Para eliminar un OpsItem

1. Si aún no lo ha hecho, instale y configure AWS CLI. Para obtener más información, consulte Instalación o actualización de la versión de AWS CLI más reciente.
2. Ejecute el siguiente comando. Reemplace *ID* con el ID del OpsItem que desea eliminar.

```
aws ssm delete-ops-item --ops-item-id ID
```

Si se ejecuta correctamente, el comando no devuelve ningún dato.

Resolución de problemas de OpsItem

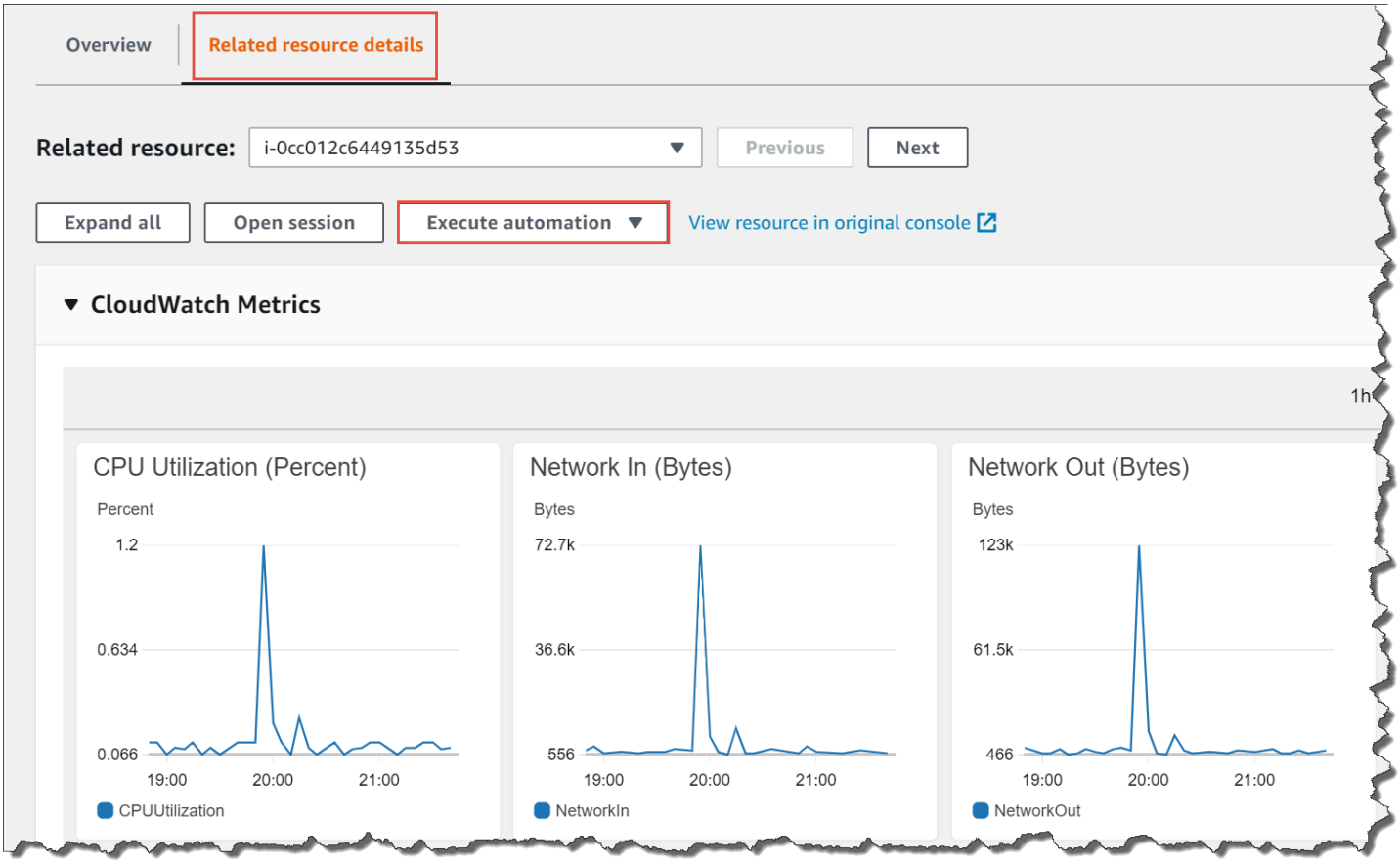
Con los manuales de procedimientos de Automatización de AWS Systems Manager, puede solucionar problemas con los recursos de AWS que se identifican en un OpsItem. Automatización utiliza manuales de procedimientos predefinidos para solucionar problemas comunes con los recursos de AWS.

Cada OpsItem incluye la sección Manual de procedimientos, la cual proporciona una lista de manuales de procedimientos que puede utilizar para la corrección. Cuando elige un manual de procedimientos de Automatización de la lista, OpsCenter muestra automáticamente algunos de los campos necesarios para ejecutar el documento. Al ejecutar un manual de procedimientos de Automatización, el sistema asocia el manual de procedimientos al recurso relacionado del OpsItem. Si Amazon EventBridge crea un OpsItem, asocia un manual de procedimientos al OpsItem. OpsCenter mantiene un registro de 30 días de manuales de procedimientos de Automatización para un OpsItem.

Puede elegir un estado para ver detalles importantes sobre el manual de procedimientos, como el motivo por el que falló una automatización y qué paso del manual de procedimientos de Automatización se estaba ejecutando cuando se produjo el error, como se muestra en el siguiente ejemplo.



La página Related resource details (Detalles de recursos relacionados) para un OpsItem seleccionado incluye la lista Run automation (Ejecutar automatización). Puede elegir manuales de procedimientos de Automatización recientes o específicos de recursos que puede ejecutar para solucionar problemas. Esta página también incluye proveedores de datos, como métricas y alarmas de Amazon CloudWatch, registros de AWS CloudTrail y detalles de AWS Config.



Puede ver información sobre un manual de procedimientos de Automation, ya sea eligiendo el nombre del manual de procedimientos en la consola o mediante Referencia del manual de procedimientos de Automatización de Systems Manager.

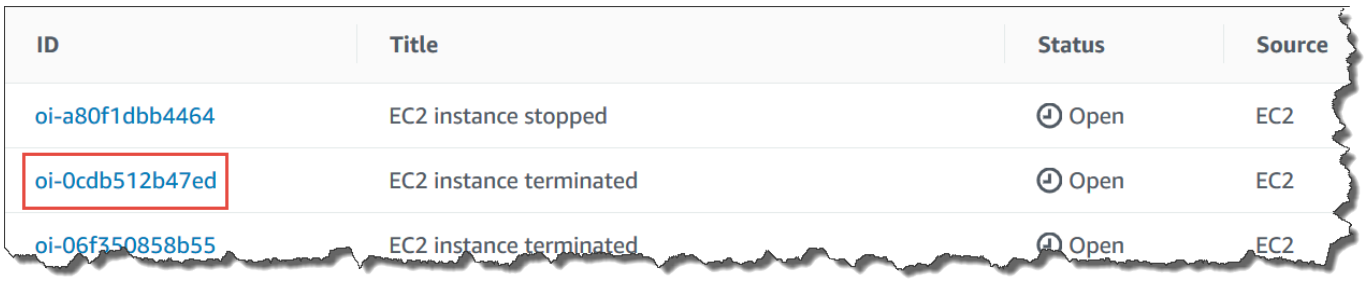
Corrección de un OpsItem con un manual de procedimientos

Antes de usar un manual de procedimientos de Automatización para solucionar un OpsItem, haga lo siguiente:

- Compruebe que tiene permiso para ejecutar manuales de procedimientos de Automatización de Systems Manager. Para obtener más información, consulte Configuración de Automation.
- Recopile información del ID específico del recurso para la automatización que desea ejecutar. Por ejemplo, si desea ejecutar una automatización que reinicia una instancia de EC2, debe especificar el ID de la instancia de EC2 que quiere reiniciar.

Para ejecutar un manual de procedimientos de Automation para corregir un problema de OpsItem

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija OpsCenter.
3. Elija el ID del OpsItem para abrir la página de detalles.



4. Desplácese hasta la sección Runbooks (Manuales de procedimientos).
5. Utilice la barra de búsqueda o los números en la esquina superior derecha de la pantalla para buscar el manual de procedimientos de Automatización que desea ejecutar.
6. Elija un manual de procedimientos y, a continuación, Execute (Ejecutar).
7. Escriba la información necesaria para el manual de procedimientos y, a continuación, elija Ejecutar.

   Una vez que inicie el manual de procedimientos, el sistema volverá a la pantalla anterior y mostrará el estado.
8. En la sección Ejecuciones automatizadas de los últimos 30 días, seleccione el enlace del ID de ejecución para ver los pasos y el estado de la ejecución.

Corrección de un OpsItem con un manual de procedimientos asociado

Después de ejecutar un manual de procedimientos de Automatización desde un OpsItem, OpsCenter asocia el manual de procedimientos al OpsItem. Un manual de procedimientos asociado tiene una clasificación superior a otros en la lista Manuales de procedimientos.

Utilice el siguiente procedimiento para ejecutar un manual de procedimientos de Automation que ya se ha asociado a un recurso relacionado en un OpsItem. Para obtener información sobre cómo añadir recursos relacionados, consulte Administración de OpsItems.

Para ejecutar un manual de procedimientos asociado a un recurso para solucionar un problema de OpsItem

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija OpsCenter.
3. Abra la OpsItem.
4. En la sección Related resources (Recursos relacionados), elija el recurso en el que desea ejecutar el manual de procedimientos de Automation.
5. Elija Run automation (Ejecutar automatización) y, a continuación, elija el manual de procedimientos de Automation asociado que desea ejecutar.
6. Escriba la información necesaria para el manual de procedimientos y, a continuación, elija Execute (Ejecutar).

   Una vez que inicie el manual de procedimientos, el sistema volverá a la pantalla anterior y mostrará el estado.
7. En la sección Ejecuciones automatizadas de los últimos 30 días, seleccione el enlace del ID de ejecución para ver los pasos y el estado de la ejecución.

Visualización de informes de resumen de OpsCenter

AWS Systems Manager OpsCenter incluye una página de resumen que muestra automáticamente la siguiente información:

- Resumen de estado de OpsItem: un resumen de OpsItems por estado, como `Open` y `In progress`.
- Orígenes con más OpsItems abiertos: desglose de los principales Servicios de AWS con OpsItems abiertos.
- OpsItems por origen y antigüedad: proporciona un recuento de los OpsItems agrupados por origen y días desde su creación.

Visualización de informes de OpsCenter resumidos

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.

2. En el panel de navegación, seleccione OpsCenter y luego haga clic en la pestaña Resumen.
3. En la sección OpsItems por origen y antigüedad, haga lo siguiente:
    1. (Opcional) En el campo de filtro, elija Origen, seleccione `Equal`, `Begin With` o `Not Equal`, y a continuación, introduzca un parámetro de búsqueda.
    2. En la lista adyacente, seleccione uno de los siguientes valores de estado:
        - `Open`
        - `In progress`
        - `Resolved`
        - `Open and in progress`
        - `All`

Solución de problemas con OpsCenter

Este tema incluye información que lo ayudará a solucionar errores y problemas comunes con OpsCenter.

Aparece la excepción OpsItemLimitExceededException

Si Cuenta de AWS ha alcanzado el número máximo permitido de OpsItems al llamar a la operación de la API CreateOpsItem, aparecerá una excepción `OpsItemLimitExceededException`. OpsCenter devuelve esta excepción si la llamada supondría superar el número máximo de OpsItems de cualquiera de las siguientes cuotas:

- Número total de OpsItems por Cuenta de AWS por región (incluyendo los OpsItems `Open` y `Resolved`): 500 000
- Número máximo de OpsItems por Cuenta de AWS al mes: 10 000

Estas cuotas se aplican a los OpsItems creados desde cualquier fuente, excepto las siguientes:

- OpsItems creados por resultados de AWS Security Hub
- OpsItems que se generan automáticamente cuando se abre un incidente de Incident Manager

Los OpsItems creados desde estas fuentes no cuentan para las cuotas de OpsItem, pero se le cobrará por cada OpsItem.

Si aparece una excepción `OpsItemLimitExceededException`, puede eliminar manualmente OpsItems hasta que quede por debajo de la cuota que impide crear un nuevo OpsItem. Una vez más, eliminar OpsItems creados para resultados de Security Hub o incidentes de Incident Manager no reducirá el número total de OpsItems aplicados por las cuotas. Debe eliminar OpsItems de otras fuentes. Para obtener información sobre cómo eliminar un OpsItem, consulte Elimine OpsItems.

Recibe una factura elevada de AWS por un gran número de OpsItems generados automáticamente

Si ha configurado la integración con AWS Security Hub, OpsCenter crea OpsItems para los resultados de Security Hub. Dependiendo del número de resultados que genere Security Hub y de la cuenta en la que inició sesión cuando configuró la integración, OpsCenter puede generar un gran número de OpsItems, con un coste adicional. Aquí tiene detalles más específicos relacionados con OpsItems generados por resultados de Security Hub:

- Si ha iniciado sesión en la cuenta de administrador del Security Hub al configurar OpsCenter y una integración con el Security Hub, el sistema crea OpsItems para los resultados en las cuentas del administrador y de todos los miembros. Todos los OpsItems se crean en la cuenta de administrador. En función de una variedad de factores, esto puede provocar una factura inesperadamente elevada de AWS.

  Si ha iniciado sesión en una cuenta de miembro al configurar la integración, el sistema solo crea OpsItems para los resultados en esa cuenta individual. Para obtener más información sobre la cuenta de administrador de Security Hub, las cuentas de los miembros y su relación con la lista de eventos de EventBridge a fin de obtener información sobre los resultados, consulte Tipos de integración de Security Hub con EventBridge en la Guía del usuario de AWS Security Hub.
- Por cada resultados que cree un OpsItem, se le cobrará el precio normal de creación del OpsItem. También se le cobrará si edita el OpsItem o si el resultado correspondiente se actualiza en Security Hub (lo que desencadena una actualización de OpsItem).
- OpsItems que se crean mediante una integración con AWS Security Hub no se encuentran limitadas en este momento por la cuota máxima de 500.000 OpsItems por cuenta en una región. Por lo tanto, es posible que las alertas de Security Hub creen más de 500.000 OpsItems facturables en cada región de una cuenta.

  Por lo tanto, para los entornos de alta producción, recomendamos limitar el alcance de los resultados de Security Hub solo a los problemas de mayor gravedad.

> **Important**
>
> Si cree que una gran número de OpsItems se han creado por error y que su factura de AWS no está justificada, contacte con Soporte.

Utilice el siguiente procedimiento si no desea que el sistema siga creando OpsItems para los resultados de Security Hub.

Para dejar de recibir OpsItems para los resultados de Security Hub

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija OpsCenter.
3. Elija Configuración.
4. En la sección Resultados de Security Hub, seleccione Editar.
5. Seleccione el control deslizante para cambiar de Activado a Desactivado. Si no puede utilizar el control deslizante, significa que no se ha habilitado Security Hub para la Cuenta de AWS.
6. Seleccione Guardar para guardar la configuración. OpsCenter ya no crea OpsItems en función de los resultados de Security Hub.

> **Important**
>
> Si OpsCenter vuelve a establecer el valor de la configuración como Activado y sigue creando OpsItems para resultados, inicie sesión en la cuenta de administrador delegado de Systems Manager o la cuenta de administración de AWS Organizations y repita este procedimiento. Si no tiene permiso para iniciar sesión en ninguna de esas cuentas, contacte con su administrador y pídale que repita este procedimiento para deshabilitar la integración en su cuenta.

Uso de paneles de Amazon CloudWatch alojados por Systems Manager

Los paneles de Amazon CloudWatch son páginas de inicio personalizables en la consola de CloudWatch que puede utilizar para monitorear sus recursos en una vista única, incluso aquellos que se reparten entre diferentes Regiones de AWS. Puede utilizar los paneles de CloudWatch para crear

vistas personalizadas de las métricas y las alarmas para sus recursos de AWS. Con los paneles, puede crear lo siguiente:

- Una vista única para las métricas y alarmas seleccionadas para ayudarlo a evaluar el estado de los recursos y las aplicaciones en una o más Regiones de AWS. Puede seleccionar el color utilizado para cada métrica en cada gráfico, de modo que pueda realizar un seguimiento de la misma métrica en varios gráficos.
- Un manual de estrategia operativo que ofrece asesoramiento a los miembros del equipo durante eventos operativos sobre cómo responder a determinados incidentes.
- Una vista común de las medidas de los recursos y las aplicaciones críticos que pueden compartir los miembros del equipo para un flujo de comunicación más rápido durante los eventos operativos.

Puede crear paneles mediante el uso de la consola, la AWS Command Line Interface (AWS CLI) o la API `PutDashboard` de CloudWatch. Para obtener más información, consulte Uso de los paneles de Amazon CloudWatch en la Guía del usuario de Amazon CloudWatch.

Uso de SSM Agent

AWS Systems Manager Agent (SSM Agent) es el software de Amazon que se ejecuta en instancias de Amazon Elastic Compute Cloud (Amazon EC2), dispositivos periféricos, servidores en las instalaciones o máquinas virtuales (VM). SSM Agent permite que Systems Manager actualice, administre y configure estos recursos. El agente procesa las solicitudes desde el servicio de Systems Manager en la Nube de AWS y, a continuación, las ejecuta tal y como se especifica en la solicitud. A continuación, SSM Agent devuelve información de estado y de ejecución al servicio de Systems Manager mediante el Amazon Message Gateway Service (`ssmmessages`). (Si Regiones de AWS se lanzó antes del 2024, la información de estado y ejecución también puede devolverse mediante el Amazon Message Delivery Service [prefijo de servicio: `ec2messages`]).

Si supervisa el tráfico, verá que los nodos administrados se comunican con los puntos de conexión de `ssmmessages.*` y, posiblemente, con los puntos de conexión de `ec2messages.*`. Para obtener más información, consulte Referencia: ec2messages, ssmmessages y otras operaciones de la API. Para obtener más información acerca de la transferencia de registros de SSM Agent a los Registros de Amazon CloudWatch, consulte Registro y monitorización en AWS Systems Manager.

Contenido

- Información sobre detalles técnicos acerca de SSM Agent
- Búsqueda de AMIs con SSM Agent preinstalado
- Uso de SSM Agent en instancias de EC2 para Linux
- Cómo utilizar SSM Agent en instancias de EC2 para macOS
- Cómo utilizar SSM Agent en instancias de EC2 para Windows Server
- Verificación del estado de SSM Agent e inicio del agente
- Verificación del número de versión de SSM Agent
- Visualización de registros de SSM Agent
- Restricción del acceso a los comandos de nivel raíz con SSM Agent
- Automatización de las actualizaciones de SSM Agent
- Suscripción a las notificaciones de SSM Agent
- Solución de problemas de SSM Agent

Información sobre detalles técnicos acerca de SSM Agent

Utilice la información de este tema para implementar AWS Systems Manager Agent (SSM Agent) y entender cómo funciona.

Temas

- Comportamiento de las credenciales de la versión 3.2.x.x de SSM Agent
- Prioridad de credenciales de SSM Agent
- Configuración de SSM Agent para su uso con el estándar federal de procesamiento de información (FIPS)
- Acerca de la cuenta ssm-user local
- SSM Agent y Instance Metadata Service (IMDS)
- Mantener el SSM Agent actualizado
- Ratificación de que el directorio de instalación de SSM Agent no se modifique, mueva o elimine
- Actualizaciones continuas de SSM Agent por Regiones de AWS
- Comunicaciones de SSM Agent con buckets de S3 administrados de AWS

Comportamiento de las credenciales de la versión 3.2.x.x de SSM Agent

SSM Agent almacena un conjunto de credenciales temporales en `/var/lib/amazon/ssm/credentials` (para Linux y macOS) o `%PROGRAMFILES%\Amazon\SSM\credentials` (para Windows Server) cuando se incorpora una instancia mediante la configuración de administración de host predeterminada en Quick Setup. Las credenciales temporales tienen los permisos que usted especifique para el rol de IAM que eligió para la configuración de administración de host predeterminada. En Linux, solo la cuenta raíz puede acceder a estas credenciales. En Windows Server, solo la cuenta SYSTEM y los administradores locales pueden acceder a estas credenciales.

Prioridad de credenciales de SSM Agent

En este tema se describe información importante acerca de cómo SSM Agent tiene permiso para realizar acciones en los recursos.

> **Note**
>
> La compatibilidad con los dispositivos periféricos difiere ligeramente. Debe configurar los dispositivos periféricos para que utilicen el software AWS IoT Greengrass Core, configurar

un rol de servicio de AWS Identity and Access Management (IAM) e implementar SSM Agent en los dispositivos mediante AWS IoT Greengrass. Para obtener más información, consulte Administración de dispositivos periféricos con Systems Manager.

Cuando SSM Agent está instalado en un equipo, requiere permisos para comunicarse con el servicio de Systems Manager. En las instancias de Amazon Elastic Compute Cloud (Amazon EC2), estos permisos se proporcionan en un perfil de instancias que está adjunto a la instancia. En un equipo que no es de EC2, por lo general SSM Agent obtiene los permisos necesarios del archivo de credenciales compartidas, ubicado en `/root/.aws/credentials` (Linux y macOS) o `%USERPROFILE%\.aws\credentials` (Windows Server). Los permisos necesarios se agregan a este archivo durante el proceso de activación híbrida.

Sin embargo, en raras ocasiones, es posible que un equipo tenga permisos agregados en más de una de las ubicaciones donde SSM Agent verifica los permisos para ejecutar sus tareas.

Por ejemplo, supongamos que configuró una instancia de EC2 para que la administre Systems Manager. Esa configuración incluye que se adjunte un perfil de instancia. Luego usted decide si usar también esa instancia para tareas de desarrollador o de usuario final e instalar la AWS Command Line Interface (AWS CLI) en ella. Como resultado de esta instalación, se agregan permisos adicionales a un archivo de credenciales en la instancia.

Cuando ejecuta un comando de Systems Manager en la instancia, SSM Agent podría intentar usar credenciales diferentes de las que se espera que use, como de un archivo de credenciales en lugar de un perfil de instancias. Esto se debe a que SSM Agent busca las credenciales en el orden prescrito para la cadena predeterminada de proveedores de credenciales.

> **Note**
>
> En Linux y macOS, SSM Agent se ejecuta como el usuario raíz. Por lo tanto, las variables de entorno y el archivo de credenciales que busca SSM Agent en este proceso son solo las del usuario raíz (`/root/.aws/credentials`). Durante la búsqueda de credenciales, SSM Agent no busca las variables de entorno o el archivo de credenciales de ningún otro usuario en la instancia.

La cadena predeterminada de proveedores busca las credenciales en el orden indicado a continuación:

1. variables de entorno, si se han configurado (`AWS_ACCESS_KEY_ID` y `AWS_SECRET_ACCESS_KEY`)
2. archivo de credenciales compartidas (`$HOME/.aws/credentials` para Linux y macOS o `%USERPROFILE%\.aws\credentials` para Windows Server) con permisos proporcionados, por ejemplo, por una activación híbrida o una instalación de la AWS CLI
3. un rol de AWS Identity and Access Management (IAM) para tareas en caso de que haya una aplicación que utilice una definición de tarea de Amazon Elastic Container Service (Amazon ECS) o una operación RunTask de la API
4. un perfil de instancias adjunto a una instancia de Amazon EC2
5. El rol de IAM elegido para la configuración de administración de host predeterminada.

Para obtener información relacionada, consulte los siguientes temas:

- Perfiles de instancia para instancias de EC2: Configuración de permisos de instancia requeridos para Systems Manager
- Activaciones híbridas: Creación de una activación híbrida para registrar nodos con Systems Manager
- Credenciales de la AWS CLI: ajustes de los archivos de credenciales y configuración en la Guía del usuario de la AWS Command Line Interface
- Cadena predeterminada de proveedores de credenciales: Especificación de credenciales en la Guía para desarrolladores de AWS SDK para Go

> **Note**
>
> Este tema de la Guía para desarrolladores de AWS SDK para Go describe la cadena predeterminada de proveedores en términos del SDK para Go. Sin embargo, los mismos principios se aplican a la evaluación de las credenciales de SSM Agent.

Configuración de SSM Agent para su uso con el estándar federal de procesamiento de información (FIPS)

Si necesita utilizar Systems Manager con módulos de cifrado validados por el Estándar Federal de Procesamiento de Información (FIPS) 140-3, puede configurar AWS Systems Manager Agent (SSM Agent) para que utilice los puntos de conexión FIPS en las regiones compatibles.

Para configurar la conexión de SSM Agent a los puntos de conexión del FIPS 140-3

1. Conéctese al nodo administrado
2. Vaya al directorio que contiene el archivo de `amazon-ssm-agent.json`.
   - Linux: `/etc/amazon/ssm/`
   - macOS: `/opt/aws/ssm/`
   - Windows Server: `C:\Program Files\Amazon\SSM\`
3. Abra el archivo con el nombre `amazon-ssm-agent.json` para editarlo.

> **Tip**
>
> Si aún no existe ningún archivo `amazon-ssm-agent.json`, copie el contenido de `amazon-ssm-agent.json.template` en un nuevo archivo denominado `amazon-ssm-agent.json`. Guarde `amazon-ssm-agent.json` en el mismo directorio donde se encuentra `amazon-ssm-agent.json.template`.

4. Añada el siguiente contenido al archivo. Sustituya los valores de los marcadores de posición de la *`región`* por el código de región correspondiente a su partición:

```
{
    ---Existing file content, if any---
    "Mds": {
        "Endpoint": "ec2messages-fips.region.amazonaws.com",
    },
    "Ssm": {
        "Endpoint": "ssm-fips.region.amazonaws.com",
    },
    "Mgs": {
        "Endpoint": "ssmmessages-fips.region.amazonaws.com",
        "Region": "region"
    },
    "S3": {
        "Endpoint": "s3-fips.dualstack.region.amazonaws.com",
        "Region": region"
    },
    "Kms": {
        "Endpoint": "kms-fips.region.amazonaws.com"
    }
}
```

Entre las regiones compatibles se incluyen:

- `us-east-1` para la región Este de EE. UU. (Norte de Virginia)
- `us-east-2` para la región Este de EE. UU. (Ohio)
- `us-west-1` para la región Oeste de EE. UU (Norte de California)
- `us-west-2` para la región Oeste de EE. UU. (Oregón)
- `ca-west-1` para la región Canadá Oeste (Calgary)

5. Guarde el archivo y reinicie el SSM Agent.

Cada vez que cambie la configuración, reinicie SSM Agent.

Puede personalizar otras características de SSM Agent mediante el mismo procedimiento. Para ver la lista actualizada de las propiedades de configuración disponibles y sus valores predeterminados, consulte Definiciones de propiedades de configuración en el repositorio de `amazon-ssm-agent` en GitHub.

Para obtener más información acerca de la compatibilidad de AWS para FIPS, consulte Estándar federal de procesamiento de la información (FIPS) 140-3.

Acerca de la cuenta ssm-user local

A partir de la versión 2.3.50.0 de SSM Agent, el agente crea una cuenta de usuario local denominada `ssm-user` y la agrega al directorio `/etc/sudoers.d` (Linux y macOS) o al grupo de administradores (Windows Server). En las versiones del agente anteriores a la 2.3.612.0, la cuenta se crea la primera vez que se inicia SSM Agent o al reiniciar después de la instalación. En la versión 2.3.612.0 y posteriores, la cuenta `ssm-user` se crea la primera vez que se inicia una sesión en una instancia. `ssm-user` es el usuario de SO predeterminado cuando se inicia una sesión en Session Manager, una herramienta de AWS Systems Manager. Puede cambiar los permisos al trasladar la cuenta `ssm-user` a un grupo con menos privilegios o al cambiar el archivo `sudoers`. La cuenta `ssm-user` no se elimina del sistema cuando se desinstala SSM Agent.

En Windows Server, SSM Agent controla la configuración de una nueva contraseña para la cuenta `ssm-user` cada vez que se inicia una sesión. No se establecen contraseñas para `ssm-user` en las instancias administradas de Linux.

A partir de la versión 2.3.612.0 de SSM Agent, la cuenta `ssm-user` no se crea de forma automática en equipos Windows Server que se utilizan como controladores de dominio. Para utilizar Session

Manager en un controlador de dominio de Windows Server, cree la cuenta `ssm-user` de forma manual si ella aún no está presente y asigne permisos de administrador de dominio al usuario.

> **Important**
>
> Para que se cree la cuenta `ssm-user`, el perfil de instancia asociado a la instancia debe proporcionar los permisos necesarios. Para obtener información, consulte el Paso 2: verificar o agregar permisos de instancia para Session Manager.

SSM Agent y Instance Metadata Service (IMDS)

Systems Manager utiliza metadatos de las instancias EC2 para funcionar de forma correcta. Systems Manager puede acceder a los metadatos de las instancias con la versión 1 o la versión 2 de Instance Metadata Service (IMDSv1 y IMDSv2). La instancia debe poder acceder a la dirección IPv4 del servicio de metadatos de la instancia: 169.254.169.254. Para obtener más información, consulte Metadatos de instancia y datos de usuario en la Guía del usuario de Amazon EC2.

Mantener el SSM Agent actualizado

Cada vez que se agregan herramientas nuevas a Systems Manager o se actualizan las herramientas existentes, se lanza una versión actualizada de SSM Agent. No utilizar la versión más reciente del agente puede impedir que el nodo administrado utilice diversas herramientas y características de Systems Manager. Por este motivo, se recomienda automatizar el proceso de mantener SSM Agent actualizado en los equipos. Para obtener más información, consulta Automatización de las actualizaciones de SSM Agent. Suscríbase a la página SSM Agent Release Notes en GitHub para recibir notificaciones sobre las actualizaciones de SSM Agent.

> **Note**
>
> Cada vez que se agregan herramientas nuevas a Systems Manager o se actualizan las herramientas existentes, se lanza una versión actualizada de SSM Agent. No utilizar la versión más reciente del agente puede impedir que el nodo administrado utilice diversas herramientas y características de Systems Manager. Por este motivo, se recomienda automatizar el proceso de mantener SSM Agent actualizado en los equipos. Para obtener más información, consulta Automatización de las actualizaciones de SSM Agent. Suscríbase a la página SSM Agent Release Notes en GitHub para recibir notificaciones sobre las actualizaciones de SSM Agent.

Las Amazon Machine Images (AMIs) que incluyen SSM Agent de forma predeterminada, pueden tardar hasta dos semanas en disponer de la versión más reciente de SSM Agent. Le recomendamos que configure las actualizaciones automáticas de SSM Agent con una mayor frecuencia.

Ratificación de que el directorio de instalación de SSM Agent no se modifique, mueva o elimine

SSM Agent está instalado en `/var/lib/amazon/ssm/` (Linux y macOS) y `%PROGRAMFILES%\Amazon\SSM\` (Windows Server). Estos directorios de instalación contienen archivos y carpetas críticos que se utilizan por SSM Agent, como un archivo de credenciales, recursos para la comunicación entre procesos (IPC) y carpetas de orquestación. No se debe modificar, mover ni eliminar nada dentro del directorio de instalación. De lo contrario, SSM Agent podría dejar de funcionar correctamente.

Actualizaciones continuas de SSM Agent por Regiones de AWS

Después de que una actualización de SSM Agent está disponible en su repositorio de GitHub, la implementación de la versión actualizada en todas las Regiones de AWS en diferentes momentos puede demorar hasta dos semanas. Por este motivo, es posible que reciba un error del tipo “No compatible en la plataforma actual” o “actualizando amazon-ssm-agent a una versión anterior; permita la opción para volver a una versión anterior para continuar” cuando intente implementar una nueva versión de SSM Agent en una región.

Para determinar la versión de SSM Agent a la que pueda acceder, puede ejecutar un comando `curl`.

Para ver la versión del agente disponible en el bucket de descarga global, ejecute el siguiente comando.

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

Para ver la versión del agente disponible en una región específica, ejecute el siguiente comando y sustituya la *`region`* con la región en la que está trabajando, como `us-east-2` para la región EE. UU. Este (Ohio).

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

También puede abrir el archivo `VERSION` directamente en su navegador sin un comando `curl`.

Comunicaciones de SSM Agent con buckets de S3 administrados de AWS

En el curso de la realización de diversas operaciones de Systems Manager, AWS Systems Manager Agent (SSM Agent) accede a varios buckets de Amazon Simple Storage Service (Amazon S3). Se puede acceder públicamente a estos buckets de S3 y, de forma predeterminada, SSM Agent se conecta a ellos a través de llamadas a `HTTP`.

Sin embargo, si utiliza un punto de conexión de nube privada virtual (VPC) en las operaciones de Systems Manager, debe conceder un permiso explícito en un perfil de instancia de Amazon Elastic Compute Cloud (Amazon EC2) para Systems Manager o en un rol de servicio para equipos que no sean de EC2 en un entorno híbrido y multinube. De lo contrario, sus recursos no puede acceder a estos buckets públicos.

Para otorgar a sus nodos administrados acceso a estos buckets cuando se utiliza un punto de conexión de VPC, cree una política de permisos de Amazon S3 personalizada y luego, adjúntela al perfil de instancia (para instancias de EC2) o al rol de servicio (para nodos administrados que no sean de EC2).

Para obtener información sobre el uso de un punto de conexión de nube privada virtual (VPC) en las operaciones de Systems Manager, consulte Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager.

> **Note**
>
> Estos permisos tan solo proporcionan acceso a los bukets administrados de AWS solicitados por SSM Agent. No conceden los permisos necesarios para otras operaciones de Amazon S3. Tampoco concede permiso para sus propios buckets de S3.

Para obtener más información, consulte los temas siguientes:

- Configuración de permisos de instancia requeridos para Systems Manager
- Creación del rol de servicio de IAM requerido para Systems Manager en entornos híbridos y multinube
- Referencia: buckets de Amazon S3 para operaciones de aplicación de revisiones

Contenido

- Permisos de bucket necesarios
- Ejemplo
- Validación de equipos activados de manera híbrida mediante una huella digital de hardware
- SSM Agent del GitHub

Permisos de bucket necesarios

En la siguiente tabla, se describe cada uno de los buckets de S3 a los que SSM Agent puede necesitar acceder para las operaciones de Systems Manager.

> **Note**
>
> *region* representa el identificador de una Región de AWS compatible con AWS Systems Manager, como us-east-2 para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

Los permisos de Amazon S3 que necesita SSM Agent

ARN del bucket de S3	Descripción
arn:aws:s3:::aws-windows-downloads- *region*/*	Necesario para algunos documentos SSM que solo admiten sistemas operativos Windows Server y para algunos documentos para la compatibilidad entre plataformas, como AWSEC2-ConfigureSTIG .
arn:aws:s3:::amazon-ssm- *region*/*	Se necesita para actualizar las instalaciones del SSM Agent. Estos buckets contienen los paquetes de instalación del SSM Agent y los manifiestos de instalación a los que se hace referencia en el complemento y documento AWS-UpdateSSMAgent . Si no se proporcionan estos permisos, SSM Agent realiza una llamada HTTP para descargar la actualización.

ARN del bucket de S3	Descripción
`arn:aws:s3:::aws-ssm-` *region*`/*`	Proporciona acceso al bucket de S3 que incluye los módulos que es necesario utilizar con ciertos documentos de Systems Manager (documentos de SSM) a los que no se aplican revisiones. Por ejemplo: `arn:aws:s3:::aws-ssm-us-east-2/*` . Estos son algunos documentos de SSM que suelen utilizarse y que se almacenan en estos buckets. • `AWS-ConfigureWindowsUpdate` • `AWS-FindWindowsUpdates` • `AWS-UpdateSSMAgent` • `AWS-UpdateEC2Config`

ARN del bucket de S3	Descripción

`arn:aws:s3:::patch-baseline-snapshot-` *region*`/*`
 	
-o bien-
 	
`arn:aws:s3:::patch-baseline-snapshot-` *region-unique-suffix* `/*`
	Proporciona acceso al bucket de S3 que incluye las instantáneas de las líneas de base de revisiones. Es necesario si utiliza cualquier a de los siguientes documentos de comando de SSM:
	
	• `AWS-RunPatchBaseline`
	• `AWS-RunPatchBaselineAssociation`
	• `AWS-RunPatchBaselineWithHooks`
	• `AWS-ApplyPatchBaseline` (un documento de SSM heredado)
	
	Los buckets más compatibles con las Regiones de AWS utilizan el siguiente formato:
	
	`arn:aws:s3:::patch-baseline-snapshot-` *region*
	
	En algunas regiones, se incluye un sufijo único adicional en el nombre del bucket. Por ejemplo, el nombre del bucket para la región Medio Oriente (Baréin) (me-south-1) es el siguiente:
	
	• `patch-baseline-snapshot-me-south-1-uduvl7q8`
	
	Para obtener una lista completa de los nombres de los buckets de instantáneas de línea de base de revisiones, consulte Buckets que contienen instantáneas de la línea de base de revisiones gestionadas de AWS.

ARN del bucket de S3	Descripción
	Note Si utiliza un firewall local y planea usar Patch Manager, ese firewall también debe permitir el acceso al punto de enlace de línea de base de revisiones correspondiente.

ARN del bucket de S3	Descripción
Para nodos administrados de Linux y Windows Server: `arn:aws:s3:::aws-patch-manager- region-unique-suffix /*` En instancias de Amazon EC2 para macOS: `arn:aws:s3:::aws-patchmanager-macos- region-unique-suffix /*`	Conceda acceso al bucket de S3 que contiene los documentos de comandos de SSM que se utilizan para las operaciones de aplicación de revisiones de Patch Manager. El nombre de cada bucket incluye un sufijo único, como en el caso de `552881074` para los buckets de la región Este de EE. UU. (Ohio) (us-east-2): • `arn:aws:s3:::aws-patch-manager-us-east-2-552881074/*` • `arn:aws:s3::: aws-patchmanager-macos-us-east-2-552881074/*` Documentos de SSM Estos son algunos documentos de SSM que suelen utilizarse y que se almacenan en estos buckets. • `AWS-RunPatchBaseline` • `AWS-RunPatchBaselineAssociation` • `AWS-RunPatchBaselineWithHooks` • `AWS-InstanceRebootWithHooks` • `AWS-PatchAsgInstance` • `AWS-PatchInstanceWithRollback` Para obtener listas completas de buckets de S3 administrados de AWS para las operaciones de aplicación de revisiones, consulte los siguientes temas: • Buckets que contienen documentos de comandos SSM para operaciones de

ARN del bucket de S3	Descripción
	aplicación de revisiones (Linux y Windows Server) • Buckets que contienen documentos de comandos SSM para operaciones de aplicación de revisiones (macOS)

Ejemplo

En el siguiente ejemplo se muestra cómo proporcionar acceso a los buckets de S3 necesarios para las operaciones de Systems Manager en la región EE. UU. Este (Ohio) (us-east-2). En la mayoría de los casos, debe proporcionar estos permisos de forma explícita en un perfil de instancias o un rol de servicio solo cuando se utiliza un punto de enlace de la VPC.

> **Important**
>
> Recomendamos que evite el uso de caracteres comodín (*) en lugar de regiones específicas en esta política. Por ejemplo, utilice `arn:aws:s3:::aws-ssm-us-east-2/*` y no `arn:aws:s3:::aws-ssm-*/*`. El uso de caracteres comodín podría conceder acceso a buckets de S3 a los que no quiere darlo. Si desea utilizar el perfil de instancia para más de una región, le recomendamos repetir el primer bloque de `Statement` de cada región.

JSON

```
{
        "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:::aws-ssm-us-east-2/*",
                    "arn:aws:s3:::patch-baseline-snapshot-us-east-2/*",
                    "arn:aws:s3:::aws-patch-manager-us-east-2-552881074/*",
```

```
                "arn:aws:s3:::aws-patchmanager-macos-us-east-2-552881074/*"
            ]
        }
    ]
}
```

Validación de equipos activados de manera híbrida mediante una huella digital de hardware

Cuando se trata de equipos que no son de EC2 en un entorno híbrido y multinube, SSM Agent recopila una serie de atributos del sistema (denominados como hash de hardware) y utiliza estos atributos para calcular una huella digital. La huella digital es una cadena opaca que el agente envía a determinadas API de Systems Manager. Esta huella digital única asocia a la persona que llama con un nodo administrado activado de manera híbrida en particular. El agente almacena la huella digital y el hash de hardware en el disco local en una ubicación denominada el Almacén.

El agente calcula el hash de hardware y la huella digital cuando el equipo se registra para su uso con Systems Manager. Luego, la huella digital se envía de nuevo al servicio de Systems Manager cuando el agente envía un comando `RegisterManagedInstance`.

Más tarde, al momento de enviar un comando `RequestManagedInstanceRoleToken`, el agente verifica la huella digital y el hash de hardware en el almacén para asegurarse de que los atributos de la máquina actual coincidan con el hash de hardware almacenado. Si los atributos de la máquina actual coinciden con el hash de hardware guardado en el almacén, el agente envía la huella digital del almacén a `RegisterManagedInstance`, lo que resulta en una llamada exitosa.

Si los atributos de la máquina actual no coinciden con el hash de hardware almacenado, SSM Agent calcula una nueva huella digital, almacena los nuevos hash de hardware y huella digital en el almacén, y envía la nueva huella digital a `RequestManagedInstanceRoleToken`. Esto provoca que se produzca un error con *`RequestManagedInstanceRoleToken`*, por lo que el agente no podrá obtener un token de rol para conectarse al servicio de Systems Manager.

Este error es intencional y se utiliza como paso de verificación para evitar que varios nodos administrados se comuniquen con el servicio de Systems Manager como el mismo nodo administrado.

Cuando compara los atributos de la máquina actual con el hash de hardware guardado en el almacén, el agente utiliza la siguiente lógica para determinar si los hash antiguos y los nuevos coinciden:

- Si el SID (ID del sistema/máquina) es diferente, no hay coincidencia.
- Pero si la dirección IP es la misma, entonces coinciden.
- De lo contrario, el porcentaje de atributos de las máquinas que coinciden se calcula y se compara con el límite de similitud que configuró el usuario para determinar si hay una coincidencia.

El límite de similitud se guarda en el almacenamiento, como parte del hash de hardware.

El límite de similitud se puede establecer después de registrar una instancia mediante un comando como el siguiente.

En equipos Linux:

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

En equipos Windows Server que utilizan PowerShell:

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

> **Important**
>
> El hecho de que uno de los componentes utilizados para calcular la huella digital cambie puede producir que el agente hiberne. Para evitar esta hibernación, establezca el límite de similitud en un valor bajo, como **1**.

SSM Agent del GitHub

El código fuente del SSM Agent está disponible en GitHub para que pueda adaptar el agente a sus necesidades. Le recomendamos enviar solicitudes de inserción para los cambios que le gustaría que incluyamos. No obstante, Amazon Web Services no admite la ejecución de copias modificadas de este software.

Búsqueda de AMIs con SSM Agent preinstalado

AWS Systems Manager Agent (SSM Agent) viene preinstalado en algunos Amazon Machine Images (AMIs) proporcionados por AWS y por terceros de confianza.

Por ejemplo, cuando lanza una instancia de Amazon Elastic Compute Cloud (Amazon EC2), creada desde una AMI con uno de los siguientes sistemas operativos, es probable que se dé cuenta de que SSM Agent ya está instalado:

- AlmaLinux
- Base AMI de Amazon Linux 1 con fecha de 09/2017 y posterior
- Amazon Linux 2
- AMIs básicas optimizadas para ECS de Amazon Linux 2
- Amazon Linux 2023 (AL2023)
- AMIs de Amazon Linux optimizada para Amazon EKS
- macOS 10.14.x (Mojave), 10.15.x (Catalina), 11.x (Big Sur), 12.x (Monterey), 13.x (Ventura), 14.x (Sonoma) y 15.x (Sequoia)
- SUSE Linux Enterprise Server (SLES) 12 y 15
- Ubuntu Server 16.04 LTS de 64 bits (Snap), 18.04, 20.04, 22.04 LTS, 23.10, 24.04 LTS, 24.0 y 25.04
- Windows Server AMIs 2008-2012 R2 publicadas en noviembre de 2016 o posteriormente
- Windows Server 2016, 2019, 2022 (no se incluyen las versiones Nano) y 2025

> **Note**
>
> Es posible que la versión del SSM Agent preinstalada en una AMI no sea la última disponible. Como práctica recomendada, recomendamos utilizar siempre la versión más reciente disponible del SSM Agent en los nodos administrados. Para obtener más información sobre la automatización de las actualizaciones del SSM Agent, consulte Automatización de las actualizaciones de SSM Agent.
>
> SSM Agent puede estar preinstalado en las AMIs administradas por AWS que no estén en esta lista. Esto generalmente indica que el sistema operativo (SO) no es totalmente compatible con todas las herramientas de Systems Manager.
>
> Puede que SSM Agent también esté preinstalado en AMIs encontradas en AWS Marketplace o en el repositorio de AMIs de la comunidad; sin embargo, AWS no admite esas AMIs.

Verificación del estado de SSM Agent

Dependiendo de cuándo se inicializó, es posible que una instancia creada a partir de una AMI de la lista anterior no tenga SSM Agent preinstalado. También es posible que una instancia tenga el agente preinstalado, pero el agente no se esté ejecutando. Por lo tanto, le recomendamos que verifique el estado de SSM Agent antes de intentar usar Systems Manager en una instancia por primera vez.

Siga este procedimiento para verificar que SSM Agent está instalado y en ejecución en una instancia. Si descubre que el agente no está instalado, puede instalarlo de manera manual en instancias de Linux, macOS y Windows Server.

Para verificar la instalación de SSM Agent en una instancia

1. Después de lanzar una instancia, espere unos minutos para que se inicie.
2. Conéctese a la instancia mediante el método que prefiera. Por ejemplo, puede usar SSH para conectarse a instancias de Linux o usar el escritorio remoto para conectarse a instancias de Windows Server.
3. Ejecute el comando correspondiente al tipo de sistema operativo de la instancia para comprobar el estado de SSM Agent.

Sistema operativo	Comando
Amazon Linux 1	`sudo status amazon-ssm-agent`
Amazon Linux 2 y Amazon Linux 2023	`sudo systemctl status amazon-ssm-agent`
macOS	No hay ningún comando para comprobar el estado del SSM Agent en macOS. Puede comprobar el estado si localiza y evalúa el archivo de registro del agente `/var/log/amazon/ssm/amazon-ssm-agent.log` .
SUSE Linux Enterprise Server	`sudo systemctl status amazon-ssm-agent`

Sistema operativo	Comando
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**
>
> Para ver los comandos con los que se puede comprobar el estado de SSM Agent en todos los tipos de sistemas operativos admitidos por Systems Manager, consulte Verificación del estado de SSM Agent e inicio del agente.

4. Evalúe el resultado del comando para conocer el estado de SSM Agent.

   Estado: instalado y en ejecución

   En la mayoría de los casos, el resultado del comando indica que el agente está instalado y se está ejecutando.

   En el siguiente ejemplo se muestra que SSM Agent está instalado y se está ejecutando en una instancia de 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--
```

   En el siguiente ejemplo se muestra que SSM Agent está instalado y se está ejecutando en una instancia de Windows Server.

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

Estado: instalado, pero no en ejecución

En algunos casos, el resultado del comando indica que el agente está instalado, pero que no se está ejecutando.

En el siguiente ejemplo se muestra que SSM Agent está instalado, pero que no se está ejecutando en una instancia de 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--
```

En el siguiente ejemplo se muestra que SSM Agent está instalado, pero que no se está ejecutando en una instancia de Windows Server.

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

Si el agente está instalado, pero no se está ejecutando, actívelo de manera manual mediante el comando correspondiente al tipo de sistema operativo de su instancia.

Sistema operativo	Comando
Amazon Linux 1	`sudo start amazon-ssm-agent`

Sistema operativo	Comando

Amazon Linux 2 y 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`
Windows Server	Ejecute el siguiente comando en PowerShell. `Start-Service AmazonSSMAgent`

Estado: No instalado

En algunos casos, el resultado del comando indica que el agente no está instalado.

En el siguiente ejemplo se muestra que SSM Agent no está instalado en una instancia de Amazon Linux 2.

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

En el siguiente ejemplo se muestra que SSM Agent no está instalado en una instancia de Windows Server.

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

Si el agente no está instalado, puede instalarlo manualmente mediante el procedimiento correspondiente al tipo de sistema operativo:

- Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux
- Instalación y desinstalación manual de SSM Agent en instancias de EC2 para macOS
- Cómo instalar y desinstalar de forma manual SSM Agent en instancias de EC2 para Windows Server

Uso de SSM Agent en instancias de EC2 para Linux

AWS Systems Manager Agent (SSM Agent) procesa solicitudes de Systems Manager y configura la máquina tal y como se especifica en la solicitud. Utilice los procedimientos de los siguientes temas para instalar, configurar o desinstalar SSM Agent en sistemas operativos Linux.

Temas

- Verificación de la firma de SSM Agent
- Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux
- Configuración de SSM Agent para utilizar un proxy en nodos de Linux

Verificación de la firma de SSM Agent

Los paquetes de instaladores deb y rpm de AWS Systems Manager Agent (SSM Agent) para instancias de Linux están firmados criptográficamente. Puede utilizar la clave pública para verificar que el paquete del agente sea original y que no se haya modificado. Si hay algún tipo de daño o

alteración en los archivos, se produce un error en la verificación. Puede verificar la firma del paquete del instalador con RPM o GPG. La siguiente información es para SSM Agent versión 3.1.1141.0 o posterior.

Para buscar el archivo de firma adecuado para la arquitectura y el sistema operativo de la instancia, consulte la siguiente tabla.

region representa el identificador de una Región de AWS compatible con AWS Systems Manager, como us-east-2 para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

Arquitectura	Sistema operativo	URL del archivo de firma	Nombre del archivo de descarga del agente
x86_64	AlmaLinux, Amazon Linux 1, Amazon Linux 2, Amazon Linux 2023, CentOS, CentOS Stream, RHEL, Oracle Linux, Rocky Linux, SLES	https://s3.*region*.amazonaws.com/amazon-ssm-*region*/latest/linux_amd64/amazon-ssm-agent.rpm.sig https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_amd64/amazon-ssm-agent.rpm.sig	amazon-ssm-agent.rpm
x86_64	Debian Server, Ubuntu Server	https://s3.*region*.amazonaws.com/amazon-	amazon-ssm-agent.deb

Arquitectura	Sistema operativo	URL del archivo de firma	Nombre del archivo de descarga del agente

		`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`	

Arquitectura	Sistema operativo	URL del archivo de firma	Nombre del archivo de descarga del agente

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`

Arquitectura	Sistema operativo	URL del archivo de firma	Nombre del archivo de descarga del agente

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`

Arquitectura	Sistema operativo	URL del archivo de firma	Nombre del archivo de descarga del agente

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`

Verificación del paquete de SSM Agent en un servidor Linux (v3.3.1802.0 y versiones posteriores)

Antes de empezar

Los procedimientos para GPG y RPM de esta sección se aplican a la versión 3.3.1802.0 y posteriores de SSM Agent. Antes de verificar la firma de SSM Agent mediante el siguiente procedimiento, asegúrese de haber descargado el paquete de agente más reciente para su sistema operativo. Por ejemplo, `https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_arm64/amazon-ssm-agent.rpm`. Para obtener más información sobre cómo descargar los paquetes de SSM Agent, consulte Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux.

Si tiene algún motivo para seguir usando la versión 3.3.1611.0 o una anterior del agente, siga las instrucciones que se indican en Verificación del paquete de SSM Agent en un servidor Linux (v3.3.1611.0 y versiones anteriores) en su lugar.

GPG

Para verificar el paquete de SSM Agent en un servidor Linux (v3.3.1802.0 y versiones posteriores)

1. Copie una de las siguientes claves públicas y guárdela en un archivo denominado `amazon-ssm-agent.gpg`.

> **Important**
>
> La siguiente clave pública caduca el 15/07/2026 (15 de julio de 2026). Systems Manager publicará una nueva clave pública en este tema antes de que caduque la antigua. Se recomienda suscribirse a la fuente RSS de este tema para recibir una notificación cuando la nueva clave esté disponible.

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

mQINBGeRNq4BEACrlf5h6Pz+k+M+QCJJ2LfK7d2Tn9J8iJ9qBK2Vwvuxco1rpSO+
KEI3nTeysPuheximps8WOCADX4VlbsKxMZQLjQM4mA26m1Tiw9nAI4kod4bKjiuM
BMUTCD1wfnjH3zQi4kDUdbpfAEMiPgNLVLH85Wf+lhK+Zm+V38DYzLyVj03kX4wK
iG6RMoxzOBZa5gNsVq+j+oCUITGz/URxH713Rgo8WeoEegI0+7iCBLKg+PM0b7GV
2nzkwWJz796HdkqSg8BwXsYaLTrHxa2P1IpwPCisAkyO7gZaMd6Uj69dtMFO+V8a
Qee6b57qGuFKZw7h1Vvc85PbF1Gy/wNIpary57kUHBFUg1vYep/roJuEbJCq97r5
I2liLl4NAyrWb9r/TAVxlXvqM4iZUhxm8GAp0FywMdBr9ZECClKa5HxuVmlm0Wgl
TXoYTOZKeDg6ZoCvyhNxWneCNip74fohXymeFF5L/budhBwy5wuwSniOgTGLo/4C
VgZHWCcN+d0Q3bx/sl2QNqPg5/xzsxEtymXLdVdwLIsLdEQUnIvy8KTs5jol3Dwi
nnEEyhly6wdaw+qDOhkSOT/VnErrSMkYF8VJfa5GjhCBWKw9JVSkaP2CI/VHOgHM
MKROnulq0hRQBR7RmLYt98xu38BHJWMmF8Ga/HJuIxzD1VmkZOPvDDESUwARAQAB
tCdTU00gQWdlbnQgPHNzbS1hZ2VudC1zaWduZXJAYW1hem9uLmNvbT6JAj8EEwEC
ACkFAmeRNq4CGy8FCQLGmIAHCwkIBwMCAQYVCAIJCgsEFgIDAQIeAQIXgAAKCRBR
qOBQ0AUuXTdND/9qldQ1E3dYjBVXOnbhiUQL594bkS5VoEX7D4fZ5UMVZa5pGiz+
husnoRUS9rH1cSeq7aHJu9hSCMuMdvRpuoo0CwLB+7HtzJvAO2M01hcEkUYa6Qdj
njTzP0ZjnoenJmqF9SYmVqAI/VPa9mNQ1OJ+HQ3qh5i6w+FoWlVqEdXjZGrWijub
TqyN33i1Y26t7Os/x8I9fUeNx37y/7Kama8LTdtv9GhWiMVBg2IuVf27HCMYofrQ
m2uCGe61IhtsnhsYaYupmljl+6qgdiuCiS9BAsoIGtqTnu8lnKcGyGz6YnRszN+U
```

```
1bNE4w+UFpXWJF8ogpYcghJ06aW/LhjZnQSx3VliLdW8eOJzou41yWmiuL3ZY8eW
KAlD+7eYKS6N6fEJCeNO2VX2lcKtDfaOX+lqGIVyexKayMfpi+0frNzt/92YCpF5
3jkeS77vMMVqKIUiIp1OCGv3XsFpIr6Bt2c2throYPDoQL3zvq6vvG40BKeRQ4tT
Y+5vTc8MeNn3LdzTl9pusxTcKifrJq7f5FIsL2CpAX8uQ+Qz+XWsYQQ5PvyUDtOz
nU/MRZaP6HnqY42bzI9ZlKgXi9IE3MXIwoET9YyzFjkIDvat7SlB4uJCpeIqp/KM
OIrTMb7paGLYmBU6YqxNBkDWItNG7NeZzyhh/R/Qqb4vJaf4S+ZqD1RZXokCHAQQ
AQIABgUCZ5E2rwAKCRB90Jej2tf1/CdnD/46It+RNoE00TesZK5n2bijH5Eljw0E
4/UpMi1SV6t2zY7lIm7TcKNn18tynJNFqB6YXXOwSbBG/fbN2E9RaoUCZw23TmAv
amuHwrfsDqsHb7zzPF0bISYjqEDLQJj/gtEugUc6XY1dEpFSlWJIOvgryG04cFXI
uD2KY87ya4s1R+sEVAJ14K4RlUCiMmzJdR0NJNYJOwBi1gkLEp6jG86ttiG2U7fY
pE2ibV+c0GeIFq8PIzqqENsn9KBuRH5EcbdBwfnsj2XfM4aR3ZtRIdWXkKkdP9Rs
yU5dTF/Y7XPId5h8/gp00+DMlXFBinQ1jE7A7eDYviEFd1ba8P7dIom3Q3gzKiWu
KTGpnykShs5NvpQmvGUF6JqDHI4RK9s3kLqsNyZkhenJfRBrJ/45fQAuP4CRedkF
7PSfX0Xp7kDnKuyK6wEUEfXXrqmuLGDmigTXblO5qgdyMwkOLjiY9znBZbHoKs76
VplOoNgGnN19i3nuMcPf2npFICJv7kTIyn5Fh7pjWDCahl3U/PwoLjrrlEzpyStU
oXSZrK3kiAADEdSODXJl8KYU0Pb27JbRr1ZbWnxb+O39TOhtssstulkR0v+IDGDQ
rQE1b12sKgcNFSzInzWrNGu4S06WN8DYzlrTZ9aSHj+37ZqpXAevi8WOFXKPV3PA
E6+O8RI2451Dcg==
=aDkv
-----END PGP PUBLIC KEY BLOCK-----
```

2. Importe la clave pública a su conjunto de claves y tenga en cuenta el valor de clave regresado.

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

3. Verifique la huella digital. Asegúrese de sustituir el *valor de clave* por el valor del paso anterior. Le recomendamos que utilice GPG para verificar la huella digital, incluso si utiliza RPM para verificar el paquete del instalador.

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

Este comando devuelve un resultado similar al siguiente:

```
pub   4096R/D0052E5D 2025-01-22 [expires: 2026-07-15]
      Key fingerprint = 4855 A9E6 8332 16D6 A77D  8FE4 51A8 E050 D005 2E5D
uid                  SSM Agent <ssm-agent-signer@amazon.com>
```

La huella digital debe coincidir con la siguiente.

```
4855 A9E6 8332 16D6 A77D 8FE4 51A8 E050 D005 2E5D
```

Si la huella digital no coincide, no instale el agente. Ponte en contacto con AWS Support.

4. Descargue el archivo de firma según la arquitectura y el sistema operativo de su instancia si aún no lo ha hecho.
5. Verifique la firma del paquete del instalador. Asegúrese de sustituir *signature-filename* y *agent-download-filename* por los valores que especificó cuando descargó el archivo de firma y el agente, como se muestra en la tabla que está más arriba en este tema.

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

Por ejemplo, para la arquitectura x86_64 en Amazon Linux 2:

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

Este comando devuelve un resultado similar al siguiente:

```
gpg: Signature made Sat 08 Feb 2025 12:05:08 AM UTC using RSA key ID D0052E5D
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: 4855 A9E6 8332 16D6 A77D  8FE4 51A8 E050 D005 2E5D
```

Si el resultado incluye la expresión `BAD signature`, compruebe si ha realizado el procedimiento correctamente. Si sigue recibiendo esta respuesta, contacte con Soporte y no instale el agente. El mensaje de advertencia sobre la confianza no significa que la firma no sea válida, sino que no se ha verificado la clave pública. Una clave solo es de confianza si la ha firmado usted o alguien en quien confíe. Si el resultado incluye la frase `Can't check signature: No public key`, verifique que ha descargado SSM Agent versión 3.1.1141.0 o posterior.

RPM

Para verificar el paquete de SSM Agent en un servidor Linux (v3.3.1802.0 y versiones posteriores)

1. Copie la siguiente clave pública y guárdela en un archivo denominado `amazon-ssm-agent.gpg`.

> **Important**
>
> La siguiente clave pública caduca el 15/07/2026 (15 de julio de 2026). Systems Manager publicará una nueva clave pública en este tema antes de que caduque la antigua. Se recomienda suscribirse a la fuente RSS de este tema para recibir una notificación cuando la nueva clave esté disponible.

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

mQINBGeRNq4BEACrlf5h6Pz+k+M+QCJJ2LfK7d2Tn9J8iJ9qBK2Vwvuxco1rpSO+
KEI3nTeysPuheximps8WOCADX4VlbsKxMZQLjQM4mA26m1Tiw9nAI4kod4bKjiuM
BMUTCD1wfnjH3zQi4kDUdbpfAEMiPgNLVLH85Wf+lhK+Zm+V38DYzLyVj03kX4wK
iG6RMoxzOBZa5gNsVq+j+oCUITGz/URxH713Rgo8WeoEegI0+7iCBLKg+PM0b7GV
2nzkwWJz796HdkqSg8BwXsYaLTrHxa2P1IpwPCisAkyO7gZaMd6Uj69dtMFO+V8a
Qee6b57qGuFKZw7h1Vvc85PbF1Gy/wNIpary57kUHBFUg1vYep/roJuEbJCq97r5
I2liLl4NAyrWb9r/TAVxlXvqM4iZUhxm8GAp0FywMdBr9ZECClKa5HxuVmlm0Wgl
TXoYTOZKeDg6ZoCvyhNxWneCNip74fohXymeFF5L/budhBwy5wuwSniOgTGLo/4C
VgZHWCcN+d0Q3bx/sl2QNqPg5/xzsxEtymXLdVdwLIsLdEQUnIvy8KTs5jol3Dwi
nnEEyhly6wdaw+qDOhkSOT/VnErrSMkYF8VJfa5GjhCBWKw9JVSkaP2CI/VHOgHM
MKROnulq0hRQBR7RmLYt98xu38BHJWMmF8Ga/HJuIxzD1VmkZOPvDDESUwARAQAB
tCdTU00gQWdlbnQgPHNzbS1hZ2VudC1zaWduZXJAYW1hem9uLmNvbT6JAj8EEwEC
ACkFAmeRNq4CGy8FCQLGmIAHCwkIBwMCAQYVCAIJCgsEFgIDAQIeAQIXgAAKCRBR
qOBQ0AUuXTdND/9qldQ1E3dYjBVXOnbhiUQL594bkS5VoEX7D4fZ5UMVZa5pGiz+
husnoRUS9rH1cSeq7aHJu9hSCMuMdvRpuoo0CwLB+7HtzJvAO2M01hcEkUYa6Qdj
njTzP0ZjnoenJmqF9SYmVqAI/VPa9mNQ1OJ+HQ3qh5i6w+FoWlVqEdXjZGrWijub
TqyN33i1Y26t7Os/x8I9fUeNx37y/7Kama8LTdtv9GhWiMVBg2IuVf27HCMYofrQ
m2uCGe61IhtsnhsYaYupmljl+6qgdiuCiS9BAsoIGtqTnu8lnKcGyGz6YnRszN+U
1bNE4w+UFpXWJF8ogpYcghJ06aW/LhjZnQSx3VliLdW8eOJzou41yWmiuL3ZY8eW
KAlD+7eYKS6N6fEJCeNO2VX2lcKtDfaOX+lqGIVyexKayMfpi+0frNzt/92YCpF5
3jkeS77vMMVqKIUiIp1OCGv3XsFpIr6Bt2c2throYPDoQL3zvq6vvG40BKeRQ4tT
Y+5vTc8MeNn3LdzTl9pusxTcKifrJq7f5FIsL2CpAX8uQ+Qz+XWsYQQ5PvyUDtOz
nU/MRZaP6HnqY42bzI9ZlKgXi9IE3MXIwoET9YyzFjkIDvat7SlB4uJCpeIqp/KM
OIrTMb7paGLYmBU6YqxNBkDWItNG7NeZzyhh/R/Qqb4vJaf4S+ZqD1RZXokCHAQQ
AQIABgUCZ5E2rwAKCRB90Jej2tf1/CdnD/46It+RNoE00TesZK5n2bijH5Eljw0E
4/UpMi1SV6t2zY7lIm7TcKNn18tynJNFqB6YXXOwSbBG/fbN2E9RaoUCZw23TmAv
amuHwrfsDqsHb7zzPF0bISYjqEDLQJj/gtEugUc6XY1dEpFSlWJIOvgryG04cFXI
uD2KY87ya4s1R+sEVAJ14K4RlUCiMmzJdR0NJNYJOwBi1gkLEp6jG86ttiG2U7fY
pE2ibV+c0GeIFq8PIzqqENsn9KBuRH5EcbdBwfnsj2XfM4aR3ZtRIdWXkKkdP9Rs
yU5dTF/Y7XPId5h8/gp00+DMlXFBinQ1jE7A7eDYviEFd1ba8P7dIom3Q3gzKiWu
```

```
KTGpnykShs5NvpQmvGUF6JqDHI4RK9s3kLqsNyZkhenJfRBrJ/45fQAuP4CRedkF
7PSfX0Xp7kDnKuyK6wEUEfXXrqmuLGDmigTXblO5qgdyMwkOLjiY9znBZbHoKs76
VplOoNgGnN19i3nuMcPf2npFICJv7kTIyn5Fh7pjWDCahl3U/PwoLjrrlEzpyStU
oXSZrK3kiAADEdSODXJl8KYU0Pb27JbRr1ZbWnxb+O39TOhtssstulkR0v+IDGDQ
rQE1b12sKgcNFSzInzWrNGu4S06WN8DYzlrTZ9aSHj+37ZqpXAevi8WOFXKPV3PA
E6+O8RI2451Dcg==
=aDkv
-----END PGP PUBLIC KEY BLOCK-----
```

2. Importe la clave pública a su conjunto de claves y tenga en cuenta el valor de clave regresado.

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

3. Verifique la huella digital. Le recomendamos que utilice GPG para verificar la huella digital, incluso si utiliza RPM para verificar el paquete del instalador.

```
rpm -qa gpg-pubkey --qf '%{Description}' | gpg --with-fingerprint | grep -A 1
 "ssm-agent-signer@amazon.com"
```

Este comando devuelve un resultado similar al siguiente:

```
pub  4096R/D0052E5D 2025-01-22 SSM Agent <ssm-agent-signer@amazon.com>
      Key fingerprint = 4855 A9E6 8332 16D6 A77D  8FE4 51A8 E050 D005 2E5D
```

La huella digital debe coincidir con la siguiente.

```
4855 A9E6 8332 16D6 A77D 8FE4 51A8 E050 D005 2E5D
```

Si la huella digital no coincide, no instale el agente. Ponte en contacto con AWS Support.

4. Verifique la firma del paquete del instalador. Asegúrese de reemplazar *agent-download-filename* por los valores que especificó cuando descargó el agente, como se muestra en la tabla que figura más arriba en este tema.

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

Por ejemplo, para la arquitectura x86_64 en Amazon Linux 2:

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

Este comando devuelve un resultado similar al siguiente.

```
amazon-ssm-agent.rpm: rsa sha1 md5 OK
```

Si en el resultado no se incluye pgp y ha importado la clave pública, entonces el agente no está firmado. Si el resultado contiene la frase `NOT OK (MISSING KEYS: (MD5) key-id)`, compruebe si ha realizado el procedimiento correctamente y verifique que ha descargado SSM Agent versión 3.1.1141.0 o posterior. Si sigue recibiendo esta respuesta, contacte con Soporte y no instale el agente.

Verificación del paquete de SSM Agent en un servidor Linux (v3.3.1611.0 y versiones anteriores)

Antes de empezar

Los procedimientos para GPG y RPM de esta sección se aplican a la versión 3.3.1611.0 y a las versiones anteriores de SSM Agent. Recomendamos usar siempre la versión más reciente del agente. Para obtener más información, consulta Verificación del paquete de SSM Agent en un servidor Linux (v3.3.1802.0 y versiones posteriores). Sin embargo, si tiene un motivo específico para seguir usando la versión 3.3.1611.0 o una anterior del agente, siga las instrucciones de uno de los siguientes procedimientos.

GPG

Para verificar el paquete de SSM Agent en un servidor Linux (v3.3.1611.0 y versiones anteriores)

1. Copie las siguientes claves públicas y guárdelas en un archivo denominado `amazon-ssm-agent.gpg`.

> **Important**
>
> La clave pública que se muestra a continuación caducó el 17/02/2025 (17 de febrero de 2025) y funciona para la versión 3.3.1611.0 y las versiones anteriores hasta la 3.2.1542.0, y solo si se usó anteriormente para verificar la firma del agente. Systems Manager publicará una nueva clave pública en este tema antes de que caduque la antigua. Se recomienda suscribirse a la fuente RSS de este tema para recibir una notificación cuando la nueva clave esté disponible.

```
-----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. Importe la clave pública a su conjunto de claves y tenga en cuenta el valor de clave regresado.

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

3. Verifique la huella digital. Asegúrese de sustituir el *valor de clave* por el valor del paso anterior. Le recomendamos que utilice GPG para verificar la huella digital, incluso si utiliza RPM para verificar el paquete del instalador.

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

Este comando regresa un resultado similar al siguiente.

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

La huella digital debe coincidir con la siguiente.

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

Si la huella digital no coincide, no instale el agente. Ponte en contacto con AWS Support.

4. Descargue el archivo de firma según la arquitectura y el sistema operativo de su instancia si aún no lo ha hecho.
5. Verifique la firma del paquete del instalador. Asegúrese de sustituir *signature-filename* y *agent-download-filename* por los valores que especificó cuando descargó el archivo de firma y el agente, como se muestra en la tabla que está más arriba en este tema.

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

Por ejemplo, para la arquitectura x86_64 en Amazon Linux 2:

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

Este comando devuelve un resultado similar al siguiente:

```
gpg: Signature made Fri 10 Jan 2025 01:54:18 AM UTC using RSA key ID 97DD04ED
gpg: Good signature from "SSM Agent <ssm-agent-signer@amazon.com>"
gpg: Note: This key has expired!
Primary key fingerprint: DE92 C7DA 3E56 E923 31D6  2A36 BC1F 495C 97DD 04ED
```

Si el resultado incluye la expresión `BAD signature`, compruebe si ha realizado el procedimiento correctamente. Si sigue recibiendo esta respuesta, contacte con Soporte y no instale el agente. El mensaje de advertencia sobre la confianza no significa que la firma no sea válida, sino que no se ha verificado la clave pública. Una clave solo es de confianza si la ha firmado usted o alguien en quien confíe. Si el resultado incluye la frase `Can't check`

`signature: No public key`, verifique que ha descargado SSM Agent versión 3.1.1141.0 o posterior.

RPM

Para verificar el paquete de SSM Agent en un servidor Linux (v3.3.1611.0 y versiones anteriores)

1. Copie la siguiente clave pública y guárdela en un archivo denominado `amazon-ssm-agent.gpg`.

> **Important**
>
> La clave pública que se muestra a continuación caducó el 17/02/2025 (17 de febrero de 2025) y funciona para la versión 3.3.1611.0 y las versiones anteriores hasta la 3.2.1542.0, y solo si se usó anteriormente para verificar la firma del agente. Systems Manager publicará una nueva clave pública en este tema antes de que caduque la antigua. Se recomienda suscribirse a la fuente RSS de este tema para recibir una notificación cuando la nueva clave esté disponible.

```
-----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. Importe la clave pública a su conjunto de claves y tenga en cuenta el valor de clave regresado.

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

3. Verifique la huella digital. Le recomendamos que utilice GPG para verificar la huella digital, incluso si utiliza RPM para verificar el paquete del instalador.

```
rpm -qa gpg-pubkey --qf '%{Description}' | gpg --with-fingerprint | grep -A 1
 "ssm-agent-signer@amazon.com"
```

Este comando devuelve un resultado similar al siguiente:

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

La huella digital debe coincidir con la siguiente.

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

Si la huella digital no coincide, no instale el agente. Ponte en contacto con AWS Support.

4. Verifique la firma del paquete del instalador. Asegúrese de reemplazar *agent-download-filename* por los valores que especificó cuando descargó el agente, como se muestra en la tabla que figura más arriba en este tema.

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

Por ejemplo, para la arquitectura x86_64 en Amazon Linux 2:

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

Este comando devuelve un resultado similar al siguiente.

```
amazon-ssm-agent.rpm: rsa sha1 md5 OK
```

Si en el resultado no se incluye pgp y ha importado la clave pública, entonces el agente no está firmado. Si el resultado contiene la frase `NOT OK (MISSING KEYS: (MD5) key-id)`, compruebe si ha realizado el procedimiento correctamente y verifique que ha descargado SSM Agent versión 3.1.1141.0 o posterior. Si sigue recibiendo esta respuesta, contacte con Soporte y no instale el agente.

Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux

Antes de instalar AWS Systems Manager Agent (SSM Agent) de forma manual en un sistema operativo Linux de Amazon Elastic Compute Cloud (Amazon EC2), consulte la siguiente información.

URL de los archivos de instalación de SSM Agent

Puede acceder a los archivos de instalación de SSM Agent que estén almacenados en cualquier Región de AWS comercial. También proporcionamos archivos de instalación en un bucket de Amazon Simple Storage Service (Amazon S3) disponible a nivel mundial que puede utilizar como fuente de archivos alternativa o de respaldo.

Si va a instalar manualmente el agente en una o dos instancias, puede utilizar los comandos de los procedimientos de Instalación rápida que proporcionamos para ahorrar tiempo. Los comandos proporcionados en estos procedimientos también se pueden pasar a las instancias de Amazon EC2 como scripts a través de los datos del usuario.

Si va a crear un script o una plantilla con objeto de emplearlos para instalar el agente en varias instancias, es recomendable utilizar los archivos de instalación de una Región de AWS donde se ubique geográficamente, o de una región próxima a esta. En el caso de instalaciones masivas, esto puede aumentar la velocidad de las descargas y reducir la latencia. En esos casos, se recomienda utilizar los procedimientos de Creación de comandos de instalación personalizados de los temas de instalación.

Amazon Machine Images con el agente preinstalado

SSM Agent está preinstalado en algunas Amazon Machine Images (AMIs) proporcionadas por AWS. Para obtener más información, consulta Búsqueda de AMIs con SSM Agent preinstalado.

Instalación en otros tipos de equipos

Si necesita instalar el agente en un servidor local o en una máquina virtual (VM) para que se pueda utilizar con Systems Manager, consulte Cómo instalar SSM Agent en nodos de Linux híbridos. Para obtener más información acerca de la instalación del agente en dispositivos de borde, consulte Administración de dispositivos periféricos con Systems Manager.

Actualización del agente

Cada vez que se agregan herramientas nuevas a Systems Manager o se actualizan las herramientas existentes, se lanza una versión actualizada de SSM Agent. No utilizar la versión más reciente del agente puede impedir que el nodo administrado utilice diversas herramientas y características de Systems Manager. Por este motivo, se recomienda automatizar el proceso de mantener SSM Agent actualizado en los equipos. Para obtener más información, consulta Automatización de las actualizaciones de SSM Agent. Suscríbase a la página SSM Agent Release Notes en GitHub para recibir notificaciones sobre las actualizaciones de SSM Agent.

Elija su sistema operativo

Para ver el procedimiento de instalación manual de SSM Agent en el sistema operativo especificado, elija un vínculo de la lista siguiente:

> **Note**
>
> Para obtener una lista de las versiones compatibles con cada uno de los siguientes sistemas operativos, consulte Sistemas operativos compatibles con Systems Manager.

- AlmaLinux
- Amazon Linux 2 y Amazon Linux 2023
- Amazon Linux 1
- CentOS
- CentOS Stream
- Debian Server

- Oracle Linux
- Red Hat Enterprise Linux
- Rocky Linux
- SUSE Linux Enterprise Server
- Ubuntu Server

Desinstalación de SSM Agent de instancias de Linux

Para desinstalar SSM Agent desde instancias de Linux, utilice el administrador de paquetes del sistema operativo. Según el sistema operativo, el comando de desinstalación será similar al siguiente ejemplo:

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

Instalación manual de SSM Agent en instancias de AlmaLinux

Utilice la información de esta sección como ayuda para instalar o reinstalar SSM Agent de manera manual en una instancia de AlmaLinux.

Antes de empezar

Antes de instalar SSM Agent en una instancia de AlmaLinux, tenga en cuenta lo siguiente:

- Asegúrese de que Python 3 esté instalado en la instancia de AlmaLinux. Esto es necesario para que SSM Agent funcione correctamente.
- Para obtener información importante que es aplicable a la instalación de SSM Agent en todos los sistemas operativos basados en Linux, consulte Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux.

Temas

- Comandos de instalación rápida para SSM Agent en AlmaLinux
- Creación de comandos de instalación del agente personalizados para AlmaLinux en una región

Comandos de instalación rápida para SSM Agent en AlmaLinux

Siga estos pasos para instalar manualmente SSM Agent en una sola instancia. En este procedimiento se utilizan archivos de instalación disponibles de manera global.

Antes de empezar

Antes de instalar SSM Agent en una instancia de AlmaLinux, tenga en cuenta lo siguiente:

- Asegúrese de que Python 3 esté instalado en la instancia de AlmaLinux. Esto es necesario para que SSM Agent funcione correctamente.

Para instalar SSM Agent en AlmaLinux

1. Conéctese a la instancia de AlmaLinux mediante el método que prefiera, como SSH.
2. Copie el comando correspondiente a la arquitectura de la instancia y ejecútelo en la instancia.

> **Note**
>
> Si bien las URL de los siguientes comandos incluyen un directorio `ec2-downloads-windows`, estos son los archivos de instalación globales correctos para AlmaLinux.

Instancias x86_64

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

Instancias ARM64

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

3. (Recomendado) Ejecute el siguiente comando para verificar que el agente está funcionando.

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

En la mayoría de los casos, el comando informa de que el agente se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

En casos excepcionales, el comando informa de que el agente está instalado pero no se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

Para activar el agente en estos casos, ejecute los siguientes comandos.

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

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

Creación de comandos de instalación del agente personalizados para AlmaLinux en una región

Cuando se instala SSM Agent en varias instancias mediante un script o una plantilla, se recomienda utilizar los archivos de instalación almacenados en la Región de AWS en la que se está trabajando.

Para los siguientes comandos, proporcionamos ejemplos en los que se utiliza un bucket de S3 de acceso público de la región Este de EE. UU. (Ohio) (`us-east-2`).

> **Tip**
>
> También puede crear una URL regional personalizada para reemplazar una URL global en el procedimiento Comandos de instalación rápida para SSM Agent en AlmaLinux que aparece antes en este tema.

En el siguiente comando, reemplace *region* con su propia información. Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

x86_64

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

Consulte el siguiente ejemplo.

```
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
```

Consulte el siguiente ejemplo.

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

Instalación manual de SSM Agent en instancias de Amazon Linux 2 y Amazon Linux 2023

Important

Este tema proporciona comandos para trabajar con SSM Agent en instancias de Amazon Linux 2 y Amazon Linux 2023. Algunos de estos comandos no son compatibles con las instancias de Amazon Linux 1. Antes de continuar, asegúrese de que esté viendo el tema correcto para su tipo de instancias. Para ver los comandos para ejecutar en las instancias de Amazon Linux 1, consulte Instalación manual de SSM Agent en instancias de Amazon Linux 1.

En la mayoría de los casos, las Amazon Machine Images (AMIs) para Amazon Linux 2 y Amazon Linux 2023 que proporciona AWS vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Para obtener más información, consulte Búsqueda de AMIs con SSM Agent preinstalado.

En caso de que SSM Agent no esté preinstalado en una nueva instancia de Amazon Linux 2 o Amazon Linux 2023, o si necesita volver a instalar de manera manual el agente, utilice la información de esta página como ayuda.

Antes de empezar

Antes de instalar SSM Agent en una instancia de Amazon Linux 2 o Amazon Linux 2023, tenga en cuenta lo siguiente:

- Para obtener información importante que es aplicable a la instalación de SSM Agent en todos los sistemas operativos basados en Linux, consulte Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux.
- Si usa un comando `yum` para actualizar el agente SSM SSM Agent en un nodo administrado después de instalar o actualizar el agente con el documento SSM `AWS-UpdateSSMAgent`, es posible que vea el siguiente mensaje: `"Warning: RPMDB altered outside of yum."` Este mensaje entra dentro de lo esperado y se puede ignorar de forma segura.

Temas

- Comandos de instalación rápida para SSM Agent en Amazon Linux 2 o Amazon Linux 2023
- Creación de comandos de instalación del agente personalizados para Amazon Linux 2 o Amazon Linux 2023 en una región

Comandos de instalación rápida para SSM Agent en Amazon Linux 2 o Amazon Linux 2023

Siga estos pasos para instalar manualmente SSM Agent en una sola instancia. En este procedimiento se utilizan archivos de instalación disponibles de manera global.

Para instalar SSM Agent en Amazon Linux 2 o Amazon Linux 2023 mediante comandos rápidos de copiar y pegar

1. Conéctese a la instancia de Amazon Linux 2 o Amazon Linux 2023 mediante el método que prefiera, como SSH.
2. Copie el comando correspondiente a la arquitectura de la instancia y ejecútelo en la instancia.

> **Note**
>
> Si bien las URL de los siguientes comandos incluyen un directorio `ec2-downloads-windows`, estos son los archivos de instalación globales correctos para Amazon Linux 2 y 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. (Recomendado) Ejecute el siguiente comando para verificar que el agente está funcionando.

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

En la mayoría de los casos, el comando informa de que el agente se está ejecutando, como se muestra en el siguiente ejemplo.

```
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--
```

En casos excepcionales, el comando informa de que el agente está instalado pero no se está ejecutando, como se muestra en el siguiente ejemplo.

```
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--
```

Para activar el agente en estos casos, ejecute el siguiente comando.

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

Creación de comandos de instalación del agente personalizados para Amazon Linux 2 o Amazon Linux 2023 en una región

Cuando se instala SSM Agent en varias instancias mediante un script o una plantilla, se recomienda utilizar los archivos de instalación almacenados en la Región de AWS en la que se está trabajando.

Para los siguientes comandos, proporcionamos ejemplos en los que se utiliza un bucket de S3 de acceso público de la región Este de EE. UU. (Ohio) (`us-east-2`).

> **Tip**
>
> También puede crear una URL regional personalizada para reemplazar una URL global en el procedimiento Comandos de instalación rápida para SSM Agent en Amazon Linux 1 que aparece antes en este tema.

En el siguiente comando, reemplace *`region`* con su propia información. Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

x86_64

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

Consulte el siguiente ejemplo.

```
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
```

Consulte el siguiente ejemplo.

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

Instalación manual de SSM Agent en instancias de Amazon Linux 1

> **Important**
>
> Amazon Linux 1 finalizó su soporte estándar el 31 de diciembre de 2020 y llegó al final de su vida útil el 31 de diciembre de 2023, según se anunció en la Actualización sobre el fin de la vida útil de la AMI de Amazon Linux en el blog de noticias de AWS. AWS ya no ofrece Amazon Machine Images (AMIs) para este sistema operativo. Sin embargo, AWS Systems Manager aún ofrece soporte para las instancias de Amazon Linux 1 existentes.
> Este tema, proporciona comandos para trabajar con SSM Agent en las instancias de Amazon Linux 1. Algunos de estos comandos no se admiten en Amazon Linux 2 y no son compatibles con las instancias de Amazon Linux 2023. Antes de continuar, compruebe que está visualizando el tema correcto para el tipo de instancias en cuestión. Para ver los comandos que se ejecutarán en las instancias de Amazon Linux 2 o Amazon Linux 2023, consulte Instalación manual de SSM Agent en instancias de Amazon Linux 2 y Amazon Linux 2023.

En la mayoría de los casos, las Amazon Machine Images (AMIs) para Amazon Linux 1 que proporciona AWS vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Para obtener más información, consulte Búsqueda de AMIs con SSM Agent preinstalado.

En caso de que necesite reinstalar manualmente el agente en Amazon Linux 1, utilice la información de esta página como ayuda.

Antes de empezar

Antes de instalar SSM Agent en una instancia de Amazon Linux 1, tenga en cuenta lo siguiente:

- Para obtener información importante que es aplicable a la instalación de SSM Agent en todos los sistemas operativos basados en Linux, consulte Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux.

- Si usa un comando `yum` para actualizar el agente SSM SSM Agent en un nodo administrado después de instalar o actualizar el agente con el documento SSM `AWS-UpdateSSMAgent`, es posible que vea el siguiente mensaje: `"Warning: RPMDB altered outside of yum."` Este mensaje entra dentro de lo esperado y se puede ignorar de forma segura.

Temas

- Comandos de instalación rápida para SSM Agent en Amazon Linux 1
- Crear comandos de instalación del agente personalizados para Amazon Linux 1 en una región

Comandos de instalación rápida para SSM Agent en Amazon Linux 1

Siga estos pasos para instalar manualmente SSM Agent en una sola instancia. En este procedimiento se utilizan archivos de instalación disponibles de manera global.

Para instalar SSM Agent en Amazon Linux 1 mediante comandos de copiar y pegar rápidos

1. Conéctese a la instancia de Amazon Linux 1 mediante el método que prefiera, como SSH.
2. Copie el comando correspondiente a la arquitectura de la instancia y ejecútelo en la instancia.

> **Note**
>
> Aunque las URL de los siguientes comandos incluyen un directorio `ec2-downloads-windows`, estos son los archivos de instalación globales correctos para 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. (Recomendado) Ejecute el comando correspondiente a la arquitectura de la instancia para verificar que el agente se está ejecutando.

   x86_64 y x86

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

   ARM64

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

   En la mayoría de los casos, el comando informa de que el agente se está ejecutando, como se muestra en los siguientes ejemplos.

   x86_64 y 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--
   ```

   En casos excepcionales, el comando informa de que el agente está instalado pero no se está ejecutando, como se muestra en los siguientes ejemplos.

   x86_64 y 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--
```

Para activar el agente en estos casos, ejecute el comando correspondiente a la arquitectura de la instancia.

x86_64 y x86

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

ARM64

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

Crear comandos de instalación del agente personalizados para Amazon Linux 1 en una región

Cuando se instala SSM Agent en varias instancias mediante un script o una plantilla, se recomienda utilizar los archivos de instalación almacenados en la Región de AWS en la que se está trabajando.

Para los siguientes comandos, proporcionamos ejemplos en los que se utiliza un bucket de S3 de acceso público de la región Este de EE. UU. (Ohio) (`us-east-2`).

> **Tip**
>
> También puede crear una URL regional personalizada para reemplazar una URL global en el procedimiento Comandos de instalación rápida para SSM Agent en Amazon Linux 1 que aparece antes en este tema.

En el siguiente comando, reemplace *`region`* con su propia información. Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

x86_64

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

Consulte el siguiente ejemplo.

```
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
```

Consulte el siguiente ejemplo.

```
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
```

Consulte el siguiente ejemplo.

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

Instalación manual de SSM Agent en instancias de CentOS

Las Amazon Machine Images (AMIs) para CentOS que proporciona AWS no vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Para obtener una lista de AMIs administradas de AWS en las que es posible que el agente esté preinstalado, consulte Búsqueda de AMIs con SSM Agent preinstalado.

Utilice la información de esta sección como ayuda para instalar o reinstalar manualmente SSM Agent en una instancia de CentOS.

Antes de empezar

Antes de instalar SSM Agent en una instancia de CentOS, tenga en cuenta lo siguiente:

- Para obtener información importante que es aplicable a la instalación de SSM Agent en todos los sistemas operativos basados en Linux, consulte Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux.
- Si usa un comando `yum` para actualizar SSM Agent en un nodo administrado después de instalar o actualizar el agente mediante el documento de SSM `AWS-UpdateSSMAgent`, es posible que aparezca el siguiente mensaje: `"Warning: RPMDB altered outside of yum."` Este mensaje está dentro de lo esperado y puede ser ignorado de forma segura.

Temas

- Instalar SSM Agent en CentOS 8.x
- Instalar SSM Agent en CentOS 7.x
- Instalar SSM Agent en CentOS 6.x

Instalar SSM Agent en CentOS 8.x

Las Amazon Machine Images (AMIs) para CentOS 8 que proporciona AWS no vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Utilice la información de esta página como ayuda para instalar o reinstalar el agente en instancias de CentOS 8.

Antes de empezar

Antes de instalar SSM Agent en una instancia de CentOS 8, tenga en cuenta lo siguiente:

- Asegúrese de que Python 2 o Python 3 esté instalado en su instancia de CentOS 8. Esto es necesario para que SSM Agent funcione correctamente.

Temas

- Comandos de instalación rápida para SSM Agent en CentOS 8
- Crear comandos de instalación del agente personalizados para CentOS 8 en una región

Comandos de instalación rápida para SSM Agent en CentOS 8

Siga estos pasos para instalar manualmente SSM Agent en una sola instancia. En este procedimiento se utilizan archivos de instalación disponibles de manera global.

Para instalar SSM Agent en CentOS 8.x

1. Conéctese a la instancia de CentOS 8 mediante el método que prefiera, como SSH.
2. Copie el comando correspondiente a la arquitectura de la instancia y ejecútelo en la instancia.

> **Note**
>
> Aunque las URL de los siguientes comandos incluyen un directorio `ec2-downloads-windows`, estos son los archivos de instalación globales correctos para CentOS 8.

x86_64Instancias

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

Instancias ARM64

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

3. (Recomendado) Ejecute el siguiente comando para verificar que el agente está funcionando.

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

En la mayoría de los casos, el comando informa de que el agente se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

En casos excepcionales, el comando informa de que el agente está instalado pero no se está ejecutando, como se muestra en el siguiente ejemplo.

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

Para activar el agente en estos casos, ejecute los siguientes comandos.

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

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

Crear comandos de instalación del agente personalizados para CentOS 8 en una región

Cuando se instala SSM Agent en varias instancias mediante un script o una plantilla, se recomienda utilizar los archivos de instalación almacenados en la Región de AWS en la que se está trabajando.

Para los siguientes comandos, proporcionamos ejemplos en los que se utiliza un bucket de S3 de acceso público de la región Este de EE. UU. (Ohio) (`us-east-2`).

> **Tip**
>
> También puede crear una URL regional personalizada para reemplazar una URL global en el procedimiento Comandos de instalación rápida para SSM Agent en CentOS 8 que aparece antes en este tema.

En el siguiente comando, reemplace *`region`* con su propia información. Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

x86_64

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

Consulte el siguiente ejemplo.

```
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
```

Consulte el siguiente ejemplo.

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

Instalar SSM Agent en CentOS 7.x

Las Amazon Machine Images (AMIs) para CentOS 7 que proporciona AWS no vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Utilice la información de esta página como ayuda para instalar o reinstalar el agente en instancias de CentOS 7.

Temas

- Comandos de instalación rápida para SSM Agent en CentOS 7
- Crear comandos de instalación del agente personalizados para CentOS 7 en una región

Comandos de instalación rápida para SSM Agent en CentOS 7

Siga estos pasos para instalar manualmente SSM Agent en una sola instancia. En este procedimiento se utilizan archivos de instalación disponibles de manera global.

Para instalar el SSM Agent en CentOS 7.x

1. Conéctese a la instancia de CentOS 7 mediante el método que prefiera, como SSH.
2. Copie el comando correspondiente a la arquitectura de la instancia y ejecútelo en la instancia.

> **Note**
>
> Aunque las URL de los siguientes comandos incluyen un directorio `ec2-downloads-windows`, estos son los archivos de instalación globales correctos para CentOS 7.

Instancias x86_64

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

Instancias ARM64

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

3. (Recomendado) Ejecute el siguiente comando para verificar que el agente está funcionando.

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

En la mayoría de los casos, el comando informa de que el agente se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

En casos excepcionales, el comando informa de que el agente está instalado pero no se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

Para activar el agente en estos casos, ejecute los siguientes comandos.

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

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

Crear comandos de instalación del agente personalizados para CentOS 7 en una región

Cuando se instala SSM Agent en varias instancias mediante un script o una plantilla, se recomienda utilizar los archivos de instalación almacenados en la Región de AWS en la que se está trabajando.

Para los siguientes comandos, proporcionamos ejemplos en los que se utiliza un bucket de S3 de acceso público de la región Este de EE. UU. (Ohio) (`us-east-2`).

> **Tip**
>
> También puede crear una URL regional personalizada para reemplazar una URL global en el procedimiento Comandos de instalación rápida para SSM Agent en CentOS 7 que aparece antes en este tema.

En el siguiente comando, reemplace *`region`* con su propia información. Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

x86_64

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

Consulte el siguiente ejemplo.

```
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
```

Consulte el siguiente ejemplo.

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

Instalar SSM Agent en CentOS 6.x

Las Amazon Machine Images (AMIs) para CentOS 6 que proporciona AWS no vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Utilice la información de esta página como ayuda para instalar o reinstalar el agente en instancias de CentOS 6.

Temas

- Comandos de instalación rápida para SSM Agent en CentOS 6
- Crear comandos de instalación del agente personalizados para CentOS 6 en una región

Comandos de instalación rápida para SSM Agent en CentOS 6

Siga estos pasos para instalar manualmente SSM Agent en una sola instancia. En este procedimiento se utilizan archivos de instalación disponibles de manera global.

Para instalar el SSM Agent en CentOS 6.x

1. Conéctese a la instancia de CentOS 6 mediante el método que prefiera, como SSH.
2. Copie el comando correspondiente a la arquitectura de la instancia y ejecútelo en la instancia.

> **Note**
>
> Aunque las URL de los siguientes comandos incluyen un directorio `ec2-downloads-windows`, estos son los archivos de instalación globales correctos para CentOS 6. Los siguientes comandos especifican el directorio de la versión `3.0.1479.0` en lugar del directorio `latest`. Esto se debe a que SSM Agent versión 3.1 y posteriores no son compatibles con CentOS 6.

Instancias x86_64

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

Instancias x86

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

3. (Recomendado) Ejecute el siguiente comando para verificar que el agente está funcionando.

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

En la mayoría de los casos, el comando informa de que el agente se está ejecutando, como se muestra en el siguiente ejemplo.

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

En casos excepcionales, el comando informa de que el agente está instalado pero no se está ejecutando, como se muestra en el siguiente ejemplo.

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

Para activar el agente en estos casos, ejecute el siguiente comando.

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

Crear comandos de instalación del agente personalizados para CentOS 6 en una región

Cuando se instala SSM Agent en varias instancias mediante un script o una plantilla, se recomienda utilizar los archivos de instalación almacenados en la Región de AWS en la que se está trabajando.

Para los siguientes comandos, proporcionamos ejemplos en los que se utiliza un bucket de S3 de acceso público de la región Este de EE. UU. (Ohio) (`us-east-2`).

> **Tip**
>
> También puede crear una URL regional personalizada para reemplazar una URL global en el procedimiento Comandos de instalación rápida para SSM Agent en CentOS 6 que aparece antes en este tema.

En el siguiente comando, reemplace *region* con su propia información. Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

> **Note**
>
> Los siguientes comandos especifican el directorio de la versión `3.0.1479.0` en lugar del directorio `latest`. Esto se debe a que SSM Agent versión 3.1 y posteriores no son compatibles con 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
```

Consulte el siguiente ejemplo.

```
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
```

Consulte el siguiente ejemplo.

```
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
```

Instalar manualmente SSM Agent en instancias de CentOS Stream

Las Amazon Machine Images (AMIs) para CentOS Stream que proporciona AWS no vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Para obtener una lista de AMIs administradas de AWS en las que es posible que el agente esté preinstalado, consulte Búsqueda de AMIs con SSM Agent preinstalado.

Utilice la información de esta sección como ayuda para instalar o reinstalar manualmente SSM Agent en una instancia de CentOS Stream.

Antes de empezar

Antes de instalar SSM Agent en una instancia de CentOS Stream, tenga en cuenta lo siguiente:

- Para obtener información importante que es aplicable a la instalación de SSM Agent en todos los sistemas operativos basados en Linux, consulte Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux.

Temas

- Comandos de instalación rápida para SSM Agent en CentOS Stream
- Crear comandos de instalación del agente personalizados para CentOS Stream en una región

Comandos de instalación rápida para SSM Agent en CentOS Stream

Siga estos pasos para instalar manualmente SSM Agent en una sola instancia. En este procedimiento se utilizan archivos de instalación disponibles de manera global.

Antes de empezar

Antes de instalar SSM Agent en una instancia de CentOS Stream, tenga en cuenta lo siguiente:

- Asegúrese de que Python 2 o Python 3 esté instalado en la instancia de CentOS Stream 8. Esto es necesario para que SSM Agent funcione correctamente.

Para instalar el SSM Agent en CentOS Stream

1. Conéctese a la instancia de CentOS Stream mediante el método que prefiera, como SSH.
2. Copie el comando correspondiente a la arquitectura de la instancia y ejecútelo en la instancia.

> **Note**
>
> Aunque las URL de los siguientes comandos incluyen un directorio `ec2-downloads-windows`, estos son los archivos de instalación globales correctos para CentOS Stream.

Instancias x86_64

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

Instancias ARM64

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

3. (Recomendado) Ejecute el siguiente comando para verificar que el agente está funcionando.

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

En la mayoría de los casos, el comando informa de que el agente se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

En casos excepcionales, el comando informa de que el agente está instalado pero no se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

Para activar el agente en estos casos, ejecute los siguientes comandos.

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

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

Crear comandos de instalación del agente personalizados para CentOS Stream en una región

Cuando se instala SSM Agent en varias instancias mediante un script o una plantilla, se recomienda utilizar los archivos de instalación almacenados en la Región de AWS en la que se está trabajando.

Para los siguientes comandos, proporcionamos ejemplos en los que se utiliza un bucket de S3 de acceso público de la región Este de EE. UU. (Ohio) (`us-east-2`).

> **Tip**
>
> También puede crear una URL regional personalizada para reemplazar una URL global en el procedimiento Comandos de instalación rápida para SSM Agent en CentOS Stream que aparece antes en este tema.

En el siguiente comando, reemplace *`region`* con su propia información. Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

x86_64

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

Consulte el siguiente ejemplo.

```
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
```

Consulte el siguiente ejemplo.

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

Instalación manual de SSM Agent en instancias de Debian Server

Las Amazon Machine Images (AMIs) para Debian Server que proporciona AWS no vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Para obtener una lista de AMIs administradas de AWS en las que es posible que el agente esté preinstalado, consulte Búsqueda de AMIs con SSM Agent preinstalado.

Utilice la información de esta sección como ayuda para instalar o reinstalar manualmente SSM Agent en una instancia de Debian Server.

Antes de empezar

Antes de instalar SSM Agent en una instancia de Debian Server, tenga en cuenta lo siguiente:

- Para obtener información importante que es aplicable a la instalación de SSM Agent en todos los sistemas operativos basados en Linux, consulte Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux.

Temas

- Comandos de instalación rápida para SSM Agent en Debian Server
- Crear comandos de instalación del agente personalizados para Debian Server en una región

Comandos de instalación rápida para SSM Agent en Debian Server

Siga estos pasos para instalar manualmente SSM Agent en una sola instancia. En este procedimiento se utilizan archivos de instalación disponibles de manera global.

Para instalar el SSM Agent en Debian Server

1. Conéctese a la instancia de Debian Server mediante el método que prefiera, como SSH.
2. Ejecute el siguiente comando para crear un directorio temporal en la instancia.

```
mkdir /tmp/ssm
```

3. Ejecute el siguiente comando para cambiar al directorio temporal.

```
cd /tmp/ssm
```

4. Copie el comando correspondiente a la arquitectura de la instancia y ejecútelo en la instancia.

> **Note**
>
> Aunque las URL de los siguientes comandos incluyen un directorio `ec2-downloads-windows`, estos son los archivos de instalación globales correctos para Debian Server. Para Debian Server 8, solo se admite la arquitectura x86_64.

Instancias x86_64

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

Instancias ARM64

```
wget https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/
debian_arm64/amazon-ssm-agent.deb
```

5. Ejecute el siguiente comando.

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

6. (Recomendado) Ejecute el siguiente comando para verificar que el agente está funcionando.

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

En la mayoría de los casos, el comando informa de que el agente se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

En casos excepcionales, el comando informa de que el agente está instalado pero no se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

Para activar el agente en estos casos, ejecute los siguientes comandos.

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

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

Crear comandos de instalación del agente personalizados para Debian Server en una región

Cuando se instala SSM Agent en varias instancias mediante un script o una plantilla, se recomienda utilizar los archivos de instalación almacenados en la Región de AWS en la que se está trabajando.

Para los siguientes comandos, proporcionamos ejemplos en los que se utiliza un bucket de S3 de acceso público de la región Este de EE. UU. (Ohio) (`us-east-2`).

> **Tip**
>
> También puede crear una URL regional personalizada para reemplazar una URL global en el procedimiento Comandos de instalación rápida para SSM Agent en Debian Server que aparece antes en este tema.

En el siguiente comando, reemplace *region* con su propia información. Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

> **Note**
>
> Para Debian Server 8, solo se admite la arquitectura x86_64.

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
```

Consulte el siguiente ejemplo.

```
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
```

Consulte el siguiente ejemplo.

```
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
```

Instalación manual de SSM Agent en instancias de Oracle Linux

Las Amazon Machine Images (AMIs) para Oracle Linux que proporciona AWS no vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Para obtener una lista de AMIs administradas de AWS en las que es posible que el agente esté preinstalado, consulte Búsqueda de AMIs con SSM Agent preinstalado.

Utilice la información de esta sección como ayuda para instalar o reinstalar manualmente SSM Agent en una instancia de Oracle Linux.

Antes de empezar

Antes de instalar SSM Agent en una instancia de Oracle Linux, tenga en cuenta lo siguiente:

- Para obtener información importante que es aplicable a la instalación de SSM Agent en todos los sistemas operativos basados en Linux, consulte Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux.
- Si usa un comando `yum` para actualizar SSM Agent en un nodo administrado después de instalar o actualizar el agente mediante el documento de SSM `AWS-UpdateSSMAgent`, es posible que aparezca el siguiente mensaje: `"Warning: RPMDB altered outside of yum."` Este mensaje está dentro de lo esperado y puede ser ignorado de forma segura.

Temas

- Comandos de instalación rápida para SSM Agent en Oracle Linux
- Crear comandos de instalación del agente personalizados para Oracle Linux en una región

Comandos de instalación rápida para SSM Agent en Oracle Linux

Siga estos pasos para instalar manualmente SSM Agent en una sola instancia. En este procedimiento se utilizan archivos de instalación disponibles de manera global.

Para instalar SSM Agent en Oracle Linux mediante comandos de copiar y pegar rápidos

1. Conéctese a la instancia de Oracle Linux mediante el método que prefiera, como SSH.
2. Copie el siguiente comando y ejecútelo en la instancia.

> **Note**
>
> Aunque la URL del siguiente comando incluye un directorio `ec2-downloads-windows`, estos son los archivos de instalación globales correctos para Oracle Linux.

x86_64

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

3. (Recomendado) Ejecute el siguiente comando para verificar que el agente está funcionando.

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

En la mayoría de los casos, el comando informa de que el agente se está ejecutando, como se muestra en el siguiente ejemplo.

```
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--
```

En casos excepcionales, el comando informa de que el agente está instalado pero no se está ejecutando, como se muestra en el siguiente ejemplo.

```
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--
```

Para activar el agente en estos casos, ejecute los siguientes comandos.

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

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

Crear comandos de instalación del agente personalizados para Oracle Linux en una región

Cuando se instala SSM Agent en varias instancias mediante un script o una plantilla, se recomienda utilizar los archivos de instalación almacenados en la Región de AWS en la que se está trabajando.

Para los siguientes comandos, proporcionamos ejemplos en los que se utiliza un bucket de S3 de acceso público de la región Este de EE. UU. (Ohio) (`us-east-2`).

> **Tip**
>
> También puede crear una URL regional personalizada para reemplazar una URL global en el procedimiento Comandos de instalación rápida para SSM Agent en Oracle Linux que aparece antes en este tema.

En el siguiente comando, reemplace *`region`* con su propia información. Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

x86_64

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

Consulte el siguiente ejemplo.

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

Instalación manual de SSM Agent en instancias de Red Hat Enterprise Linux

Las Amazon Machine Images (AMIs) para Red Hat Enterprise Linux (RHEL) que proporciona AWS no vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada.

Para obtener una lista de AMIs administradas de AWS en las que es posible que el agente esté preinstalado, consulte Búsqueda de AMIs con SSM Agent preinstalado.

Utilice la información de esta sección como ayuda para instalar o reinstalar manualmente SSM Agent en una instancia de RHEL.

Antes de empezar

Antes de instalar SSM Agent en una instancia de RHEL, tenga en cuenta lo siguiente:

- Para obtener información importante que es aplicable a la instalación de SSM Agent en todos los sistemas operativos basados en Linux, consulte Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux.
- Si usa un comando `yum` para actualizar SSM Agent en un nodo administrado después de instalar o actualizar el agente mediante el documento de SSM `AWS-UpdateSSMAgent`, es posible que aparezca el siguiente mensaje: `"Warning: RPMDB altered outside of yum."` Este mensaje está dentro de lo esperado y puede ser ignorado de forma segura.

Temas

- Instale el SSM Agent en RHEL 8.x, 9.x y 10.x
- Instalar SSM Agent en RHEL 7.x
- Instalar SSM Agent en RHEL 6.x

Instale el SSM Agent en RHEL 8.x, 9.x y 10.x

Las Amazon Machine Images (AMIs) para RHEL 8 y 9 que proporciona AWS no vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Utilice la información de esta página como ayuda para instalar o reinstalar el agente en instancias de RHEL 8 y 9.

Antes de empezar

Antes de instalar SSM Agent en una instancia de RHEL 8, 9 o 10, tenga en cuenta lo siguiente:

- Asegúrese de que Python 2 o Python 3 esté instalado en la instancia de RHEL 8, 9 o 10. Esto es necesario para que SSM Agent funcione correctamente.

Temas

- Comandos de instalación rápida para SSM Agent en RHEL 8, 9 y 10.

- Crear comandos de instalación del agente personalizados para RHEL 8, 9 y 10 en una región

Comandos de instalación rápida para SSM Agent en RHEL 8, 9 y 10.

Siga estos pasos para instalar manualmente SSM Agent en una sola instancia. En este procedimiento se utilizan archivos de instalación disponibles de manera global.

Para instalar SSM Agent en RHEL 8.x, 9.x, o 10.x

1. Conéctese a la instancia de RHEL 8, 9 o 10 mediante el método que prefiera, como SSH.
2. Copie el comando correspondiente a la arquitectura de la instancia y ejecútelo en la instancia.

> **Note**
>
> Aunque las URL de los siguientes comandos incluyen un directorio `ec2-downloads-windows`, estos son los archivos de instalación globales correctos para RHEL 8, 9 y 10.

Instancias x86_64

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

Instancias ARM64

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

3. (Recomendado) Ejecute el siguiente comando para verificar que el agente está funcionando.

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

En la mayoría de los casos, el comando informa de que el agente se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

En casos excepcionales, el comando informa de que el agente está instalado pero no se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

Para activar el agente en estos casos, ejecute los siguientes comandos.

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

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

Crear comandos de instalación del agente personalizados para RHEL 8, 9 y 10 en una región

Cuando se instala SSM Agent en varias instancias mediante un script o una plantilla, se recomienda utilizar los archivos de instalación almacenados en la Región de AWS en la que se está trabajando.

Para los siguientes comandos, proporcionamos ejemplos en los que se utiliza un bucket de S3 de acceso público de la región Este de EE. UU. (Ohio) (`us-east-2`).

> **Tip**
>
> También puede crear una URL regional personalizada para reemplazar una URL global en el procedimiento Comandos de instalación rápida para SSM Agent en RHEL 8, 9 y 10. que aparece antes en este tema.

En el siguiente comando, reemplace *region* con su propia información. Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

x86_64

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

Consulte el siguiente ejemplo.

```
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
```

Consulte el siguiente ejemplo.

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

Instalar SSM Agent en RHEL 7.x

Las Amazon Machine Images (AMIs) para RHEL 7 que proporciona AWS no vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Utilice la información de esta página como ayuda para instalar o reinstalar el agente en instancias de RHEL 7.

Temas

- Comandos de instalación rápida para SSM Agent en RHEL 7
- Crear comandos de instalación del agente personalizados para RHEL 7 en una región

Comandos de instalación rápida para SSM Agent en RHEL 7

Siga estos pasos para instalar manualmente SSM Agent en una sola instancia. En este procedimiento se utilizan archivos de instalación disponibles de manera global.

Para instalar el SSM Agent en RHEL 7.c

1. Conéctese a la instancia de RHEL 7 mediante el método que prefiera, como SSH.
2. Copie el comando correspondiente a la arquitectura de la instancia y ejecútelo en la instancia.

> **Note**
>
> Aunque las URL de los siguientes comandos incluyen un directorio `ec2-downloads-windows`, estos son los archivos de instalación globales correctos para RHEL 7.

Instancias x86_64

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

Instancias ARM64

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

3. (Recomendado) Ejecute el siguiente comando para verificar que el agente está funcionando.

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

En la mayoría de los casos, el comando informa de que el agente se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

En casos excepcionales, el comando informa de que el agente está instalado pero no se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

Para activar el agente en estos casos, ejecute los siguientes comandos.

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

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

Crear comandos de instalación del agente personalizados para RHEL 7 en una región

Cuando se instala SSM Agent en varias instancias mediante un script o una plantilla, se recomienda utilizar los archivos de instalación almacenados en la Región de AWS en la que se está trabajando.

Para los siguientes comandos, proporcionamos ejemplos en los que se utiliza un bucket de S3 de acceso público de la región Este de EE. UU. (Ohio) (`us-east-2`).

> **Tip**
>
> También puede crear una URL regional personalizada para reemplazar una URL global en el procedimiento Comandos de instalación rápida para SSM Agent en RHEL 7 que aparece antes en este tema.

En el siguiente comando, reemplace *`region`* con su propia información. Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

x86_64

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

Consulte el siguiente ejemplo.

```
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
```

Consulte el siguiente ejemplo.

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

Instalar SSM Agent en RHEL 6.x

Las Amazon Machine Images (AMIs) para RHEL 6 que proporciona AWS no vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Utilice la información de esta página como ayuda para instalar o reinstalar el agente en instancias de RHEL 6.

Temas

- Comandos de instalación rápida para SSM Agent en RHEL 6
- Crear comandos de instalación del agente personalizados para RHEL 6 en una región

Comandos de instalación rápida para SSM Agent en RHEL 6

Siga estos pasos para instalar manualmente SSM Agent en una sola instancia. En este procedimiento se utilizan archivos de instalación disponibles de manera global.

Para instalar el SSM Agent en RHEL 6.x

1. Conéctese a la instancia de RHEL 6 mediante el método que prefiera, como SSH.

2. Copie el comando correspondiente a la arquitectura de la instancia y ejecútelo en la instancia.

> **Note**
>
> Aunque las URL de los siguientes comandos incluyen un directorio `ec2-downloads-windows`, estos son los archivos de instalación globales correctos para RHEL 6.
> Los siguientes comandos especifican el directorio de la versión `3.0.1479.0` en lugar del directorio `latest`. Esto se debe a que SSM Agent versión 3.1 y posteriores no son compatibles con RHEL 6.

Instancias x86_64

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

Instancias x86

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

3. (Recomendado) Ejecute el siguiente comando para verificar que el agente está funcionando.

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

En la mayoría de los casos, el comando informa de que el agente se está ejecutando, como se muestra en el siguiente ejemplo.

```
amazon-ssm-agent start/running, process 1788
```

En casos excepcionales, el comando informa de que el agente está instalado pero no se está ejecutando, como se muestra en el siguiente ejemplo.

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

Para activar el agente en estos casos, ejecute el siguiente comando.

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

Crear comandos de instalación del agente personalizados para RHEL 6 en una región

Cuando se instala SSM Agent en varias instancias mediante un script o una plantilla, se recomienda utilizar los archivos de instalación almacenados en la Región de AWS en la que se está trabajando.

Para los siguientes comandos, proporcionamos ejemplos en los que se utiliza un bucket de S3 de acceso público de la región Este de EE. UU. (Ohio) (`us-east-2`).

> **Tip**
>
> También puede crear una URL regional personalizada para reemplazar una URL global en el procedimiento Comandos de instalación rápida para SSM Agent en RHEL 6 que aparece antes en este tema.

En el siguiente comando, reemplace *`region`* con su propia información. Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

> **Note**
>
> Los siguientes comandos especifican el directorio de la versión `3.0.1479.0` en lugar del directorio `latest`. Esto se debe a que SSM Agent versión 3.1 y posteriores no son compatibles con 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
```

Consulte el siguiente ejemplo.

```
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
```

Consulte el siguiente ejemplo.

```
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
```

Instalar manualmente SSM Agent en instancias de Rocky Linux

Las Amazon Machine Images (AMIs) para Rocky Linux que proporciona AWS no vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Para obtener una lista de AMIs administradas de AWS en las que es posible que el agente esté preinstalado, consulte Búsqueda de AMIs con SSM Agent preinstalado.

Utilice la información de esta sección como ayuda para instalar o reinstalar manualmente SSM Agent en una instancia de Rocky Linux.

Antes de empezar

Antes de instalar SSM Agent en una instancia de Rocky Linux, tenga en cuenta lo siguiente:

- Para obtener información importante que es aplicable a la instalación de SSM Agent en todos los sistemas operativos basados en Linux, consulte Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux.

Temas

- Comandos de instalación rápida para SSM Agent en Rocky Linux
- Crear comandos de instalación del agente personalizados para Rocky Linux en una región

Comandos de instalación rápida para SSM Agent en Rocky Linux

Siga estos pasos para instalar manualmente SSM Agent en una sola instancia. En este procedimiento se utilizan archivos de instalación disponibles de manera global.

Antes de empezar

Antes de instalar SSM Agent en una instancia de Rocky Linux, tenga en cuenta lo siguiente:

- Asegúrese de que Python 2 o Python 3 esté instalado en la instancia de Rocky Linux. Esto es necesario para que SSM Agent funcione correctamente.

Para instalar el SSM Agent en Rocky Linux

1. Conéctese a la instancia de Rocky Linux mediante el método que prefiera, como SSH.
2. Copie el comando correspondiente a la arquitectura de la instancia y ejecútelo en la instancia.

> **Note**
>
> Aunque las URL de los siguientes comandos incluyen un directorio `ec2-downloads-windows`, estos son los archivos de instalación globales correctos para Rocky Linux.

Instancias x86_64

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

Instancias ARM64

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

3. (Recomendado) Ejecute el siguiente comando para verificar que el agente está funcionando.

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

En la mayoría de los casos, el comando informa de que el agente se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

En casos excepcionales, el comando informa de que el agente está instalado pero no se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

Para activar el agente en estos casos, ejecute los siguientes comandos.

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

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

Crear comandos de instalación del agente personalizados para Rocky Linux en una región

Cuando se instala SSM Agent en varias instancias mediante un script o una plantilla, se recomienda utilizar los archivos de instalación almacenados en la Región de AWS en la que se está trabajando.

Para los siguientes comandos, proporcionamos ejemplos en los que se utiliza un bucket de S3 de acceso público de la región Este de EE. UU. (Ohio) (`us-east-2`).

> **Tip**
>
> También puede crear una URL regional personalizada para reemplazar una URL global en el procedimiento Comandos de instalación rápida para SSM Agent en Rocky Linux que aparece antes en este tema.

En el siguiente comando, reemplace *`region`* con su propia información. Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

x86_64

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

Consulte el siguiente ejemplo.

```
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
```

Consulte el siguiente ejemplo.

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

Instalar manualmente SSM Agent en instancias de SUSE Linux Enterprise Server

En la mayoría de los casos, las Amazon Machine Images (AMIs) para SUSE Linux Enterprise Server (SLES) que proporciona AWS vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Para obtener más información, consulte Búsqueda de AMIs con SSM Agent preinstalado.

En caso de que SSM Agent no esté preinstalado en una nueva instancia de SLES, o si necesita reinstalar manualmente el agente, utilice la información de esta página como ayuda.

Antes de empezar

Antes de instalar SSM Agent en una instancia de SLES, tenga en cuenta lo siguiente:

- Para obtener información importante que es aplicable a la instalación de SSM Agent en todos los sistemas operativos basados en Linux, consulte Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux.

Temas

- Comandos de instalación rápida para SSM Agent en SLES
- Crear comandos de instalación del agente personalizados para SLES en una región

Comandos de instalación rápida para SSM Agent en SLES

Siga estos pasos para instalar manualmente SSM Agent en una sola instancia. En este procedimiento se utilizan archivos de instalación disponibles de manera global.

Para instalar SSM Agent en SLES mediante comandos de copiar y pegar rápidos

1. Conéctese a la instancia de SLES mediante el método que prefiera, como SSH.
2. Opción 1: utilizar un comando `zypper`:

   - Ejecuta el siguiente comando:

     ```
     sudo zypper install amazon-ssm-agent
     ```

   - Ingrese y como respuesta a las posibles solicitudes de información.

   Opción 2: utilizar un comando `rpm`.

   - Cree un directorio temporal en la instancia.

     ```
     mkdir /tmp/ssm
     ```

   - Cambie al directorio temporal.

     ```
     cd /tmp/ssm
     ```

   - Ejecute los siguientes comandos de uno en uno para descargar y ejecutar el instalador de SSM Agent.

     > **Note**
     >
     > Aunque las URL de los siguientes comandos incluyen un directorio `ec2-downloads-windows`, estos son los archivos de instalación globales correctos para SLES.

     Instancias de x86_64:

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

Instancias de ARM64:

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

- Ejecute el siguiente comando.

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

- (Recomendado) Ejecute el siguiente comando para verificar que el agente está funcionando.

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

En la mayoría de los casos, el comando informa de que el agente se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

En casos excepcionales, el comando informa de que el agente está instalado pero no se está ejecutando, como se muestra en el siguiente ejemplo.

```
# 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--
```

Para activar el agente en estos casos, ejecute los siguientes comandos.

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

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

Crear comandos de instalación del agente personalizados para SLES en una región

Cuando se instala SSM Agent en varias instancias mediante un script o una plantilla, se recomienda utilizar los archivos de instalación almacenados en la Región de AWS en la que se está trabajando.

Para los siguientes comandos, proporcionamos ejemplos en los que se utiliza un bucket de S3 de acceso público de la región Este de EE. UU. (Ohio) (`us-east-2`).

> **Tip**
>
> También puede crear una URL regional personalizada para reemplazar una URL global en el procedimiento Comandos de instalación rápida para SSM Agent en Amazon Linux 1 que aparece antes en este tema.

En el siguiente comando, reemplace *`region`* con su propia información. Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de 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
```

Consulte el siguiente ejemplo.

```
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
```

Consulte el siguiente ejemplo.

```
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
```

Instalación manual de SSM Agent en instancias de Ubuntu Server

> **Important**
>
> Antes de instalar SSM Agent en una versión de 64 bits de Ubuntu Server, asegúrese de que está utilizando las herramientas de instalación correctas. A partir de las imágenes de máquina de Amazon (AMI) que están identificadas con 20180627, SSM Agent está preinstalado en la versión 16.04 mediante paquetes Snap. En las instancias creadas a partir de AMI anteriores, debe instalarse SSM Agent mediante paquetes de instalador deb. Para obtener más información, consulte Determinación de la versión correcta de SSM Agent que se debe instalar en instancias de Ubuntu Server 16.04 de 64 bits.

En la mayoría de los casos, las Amazon Machine Images (AMIs) para Ubuntu Server que proporciona AWS vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Para obtener más información, consulte Búsqueda de AMIs con SSM Agent preinstalado.

En caso de que SSM Agent no esté preinstalado en una nueva instancia de Ubuntu Server, o si necesita reinstalar manualmente el agente, utilice la información de esta sección como ayuda.

Antes de empezar

Antes de instalar SSM Agent en una instancia de Ubuntu Server, tenga en cuenta lo siguiente:

- Para obtener información importante que es aplicable a la instalación de SSM Agent en todos los sistemas operativos basados en Linux, consulte Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux.

Temas

- Instalar SSM Agent en Ubuntu Server 16.04 LTS de 64 bits (Snap), 18.04, 20.04, 20.10 STR, 22.04 LTS, 23.10, 24.04 LTS, 24.0 y 25.04
- Instalar SSM Agent en Ubuntu Server 16.04 y 14.04 de 64 bits (deb)
- Instalar SSM Agent en Ubuntu Server 16.04 y 14.04 de 32 bits
- Determinación de la versión correcta de SSM Agent que se debe instalar en instancias de Ubuntu Server 16.04 de 64 bits

Instalar SSM Agent en Ubuntu Server 16.04 LTS de 64 bits (Snap), 18.04, 20.04, 20.10 STR, 22.04 LTS, 23.10, 24.04 LTS, 24.0 y 25.04

Antes de empezar

Antes de instalar SSM Agent en un Ubuntu Server 16.04 LTS de 64 bit (Snap), 18.04, 20.04, 20.10 STR, 22.04 LTS, 23.10, 24.04 LTS, 24.0, y 25.04, tenga en cuenta lo siguiente:

Instalación de la versión 16.04 por parte de instaladores de Snaps o deb

En Ubuntu Server 16.04, SSM Agent se instala mediante Snaps o paquetes de instalación deb, en función de la versión de la AMI de la versión 16.04.

Ubicaciones de archivos del instalador de SSM Agent

En Ubuntu Server 16.04 LTS de 64 bits (Snap), 18.04, 20.04, 20.10 STR, 22.04 LTS, 23.10, 24.04 LTS, 24.0, y 25.04, los archivos del instalador de SSM Agent, incluidos los archivos binarios y de configuración del agente, se almacenan en el siguiente directorio: `/snap/amazon-ssm-agent/current/`. Si realiza cambios en cualquiera de los archivos de configuración de este directorio, debe copiar estos archivos desde el directorio `/snap` al directorio `/etc/amazon/ssm/`. Los archivos de registros y bibliotecas no han cambiado (`/var/lib/amazon/ssm`, `/var/log/amazon/ssm`).

Uso del canal `candidate` de Snap

El canal candidato en el almacén de Snap contiene la versión más reciente de SSM Agent (incluidas todas las correcciones de errores más recientes); no el canal estable. Para obtener más información acerca de las diferencias entre los canales candidato y estable, consulte Risk-levels en https://snapcraft.io/docs/channels.

Si desea realizar un seguimiento de información de la versión de SSM Agent en el canal candidato, ejecute el siguiente comando en las instancias de 64 bits de Ubuntu Server 20.10 STR y 20.04, 18.04 y 16.04 LTS.

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

Se recomienda Snaps en las versiones 18.04 y posteriores

En Ubuntu Server 18.04, 20.04, 20.10 STR, 22.04 LTS, 23.10, 24.04 LTS, 24.0, y 25.04, le recomendamos que solo utilice Snaps. Asegúrese también de que solo hay una instancia del agente instalada y en ejecución en las instancias. Si desea utilizar SSM Agent sin Snaps, desinstale SSM Agent. Luego, instale SSM Agent como paquete de Debian siguiendo las instrucciones de instalación de SSM Agent en Ubuntu Server 16.04 y 14.04 de 64 bits (deb). Antes de realizar la instalación, asegúrese de no tener instalado ningún Snaps que se solape con la lista de paquetes que desea que se administren como paquetes de Debian.

Mensaje de error `Maximum timeout exceeded`

Debido a un problema conocido de Snap, puede aparecer un error `Maximum timeout exceeded` con los comandos `snap`. Si recibe este error, ejecute los siguientes comandos por separado para iniciar el agente, detenerlo y comprobar su estado:

```
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
```

Para instalar SSM Agent en Ubuntu Server 16.04 LTS de 64 bits (Snap), 18.04, 20.04, 20.10 STR, 22.04 LTS, 23.10, 24.04 LTS, 24.0, y 25.04 (con un paquete Snap)

1. SSM Agent se instala de forma predeterminada en las AMIs de Ubuntu Server de 16.04 LTS de 64 bits (Snap), 18.04, 20.04, 20.10 STR, 22.04 LTS, 23.10, 24.04 LTS, 24.0, y 25.04 con el identificador `20180627` o uno posterior.

   Puede utilizar el siguiente script si necesita instalar el SSM Agent en un servidor local o si necesita volver a instalar el agente. No es necesario especificar una URL para la descarga, ya que el comando `snap` descarga automáticamente el agente en la tienda de aplicaciones de Snap en https://snapcraft.io.

   ```
   sudo snap install amazon-ssm-agent --classic
   ```

2. Ejecute el siguiente comando para determinar si el SSM Agent se está ejecutando.

   ```
   sudo snap list amazon-ssm-agent
   ```

3. Ejecute el siguiente comando para iniciar el servicio si el comando anterior devuelve `amazon-ssm-agent is stopped`, `inactive` o `disabled`.

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

4. Compruebe el estado del agente.

   ```
   sudo snap services amazon-ssm-agent
   ```

Instalar SSM Agent en Ubuntu Server 16.04 y 14.04 de 64 bits (deb)

> **Important**
>
> Antes de instalar SSM Agent en una versión de 64 bits de Ubuntu Server, asegúrese de que está utilizando las herramientas de instalación de correcciones. A partir de las imágenes de máquina de Amazon (AMI) que están identificadas con 20180627, SSM Agent está preinstalado en la versión 16.04 mediante paquetes Snap. En las instancias creadas a partir de AMI anteriores, debe instalarse SSM Agent mediante paquetes de instalador deb. Para obtener más información, consulte Determinación de la versión correcta de SSM Agent que se debe instalar en instancias de Ubuntu Server 16.04 de 64 bits. Si SSM Agent está

instalado en la instancia junto con un Snap y se instala o actualiza SSM Agent con un paquete de instalador deb, la instalación o las operaciones de SSM Agent pueden producir un error.

En la mayoría de los casos, las Amazon Machine Images (AMIs) para Ubuntu Server 16.04 que proporciona AWS vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Para obtener más información, consulte Búsqueda de AMIs con SSM Agent preinstalado.

En caso de que SSM Agent no esté preinstalado en una nueva instancia de Ubuntu Server 16.04 anterior a la versión 20180627, de que vaya a realizar la instalación en Ubuntu Server 14.04, o de que necesite reinstalar manualmente el agente, utilice la información de esta página como ayuda.

Comandos de instalación rápida para SSM Agent en Ubuntu Server 16.04 y 14.04 de 64 bits (deb)

Siga estos pasos para instalar manualmente SSM Agent en una sola instancia. En este procedimiento se utilizan archivos de instalación disponibles de manera global.

Para instalar SSM Agent enUbuntu Server 16.04 y 14.04 de 64 bits (deb) mediante comandos de copiar y pegar rápidos

1. Conéctese a la instancia de Ubuntu Server mediante el método que prefiera, como SSH.
2. Ejecute el siguiente comando para crear un directorio temporal en la instancia.

```
mkdir /tmp/ssm
```

3. Cambie al directorio temporal.

```
cd /tmp/ssm
```

4. Ejecute los siguientes comandos.

> **Note**
>
> Aunque las URL de los siguientes comandos incluyen un directorio `ec2-downloads-windows`, estos son los archivos de instalación globales correctos para Ubuntu Server 16.04 y 14.04 de 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. (Recomendado) Ejecute uno de los comandos siguientes para determinar si SSM Agent se está ejecutando.

   Ubuntu Server 16.04

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

   Ubuntu Server 14.04

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

   En la mayoría de los casos, el comando informa de que el agente se está ejecutando.

   En casos excepcionales, el comando informa de que el agente está instalado pero no se está ejecutando, como se muestra en el siguiente ejemplo.

6. Ejecute uno de los siguientes comando para iniciar el servicio si el comando anterior devuelve `amazon-ssm-agent is stopped`, `inactive` o `disabled`.

   Ubuntu Server 16.04:

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

   Ubuntu Server 14.04:

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

Crear comandos de instalación personalizados para SSM Agent en Ubuntu Server 16.04 y 14.04 de 64 bits (deb) en una región

Cuando se instala SSM Agent en varias instancias mediante un script o una plantilla, se recomienda utilizar los archivos de instalación almacenados en la Región de AWS en la que se está trabajando.

Para los siguientes comandos, proporcionamos ejemplos en los que se utiliza un bucket de S3 de acceso público de la región Este de EE. UU. (Ohio) (`us-east-2`).

> **Tip**
>
> También puede crear una URL regional personalizada para reemplazar una URL global en el procedimiento Comandos de instalación rápida para SSM Agent en Ubuntu Server 16.04 y 14.04 de 64 bits (deb) que aparece antes en este tema.

En el siguiente comando, reemplace *`region`* con su propia información. Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de 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
```

Consulte el siguiente ejemplo.

```
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
```

Instalar SSM Agent en Ubuntu Server 16.04 y 14.04 de 32 bits

En la mayoría de los casos, las Amazon Machine Images (AMIs) para Ubuntu Server 16.04 que proporciona AWS vienen con AWS Systems Manager Agent (SSM Agent) preinstalado de manera predeterminada. Para obtener más información, consulte Búsqueda de AMIs con SSM Agent preinstalado.

En caso de que SSM Agent no esté preinstalado en una nueva instancia de Ubuntu Server 16.04, de que vaya a realizar la instalación en Ubuntu Server 14.04, o de que necesite reinstalar manualmente el agente, utilice la información de esta página como ayuda.

Comandos de instalación rápida para SSM Agent en Ubuntu Server 16.04 y 14.04 de 32 bits (deb)

Siga estos pasos para instalar manualmente SSM Agent en una sola instancia. En este procedimiento se utilizan archivos de instalación disponibles de manera global.

Para instalar SSM Agent en Ubuntu Server 16.04 y 14.04 de 32 bits (deb) mediante comandos de copiar y pegar rápidos

1. Conéctese a la instancia de Ubuntu Server mediante el método que prefiera, como SSH.
2. Ejecute el siguiente comando para crear un directorio temporal en la instancia.

```
mkdir /tmp/ssm
```

3. Cambie al directorio temporal.

```
cd /tmp/ssm
```

4. Ejecute los siguientes comandos.

> **Note**
>
> Aunque las URL de los siguientes comandos incluyen un directorio `ec2-downloads-windows`, estos son los archivos de instalación globales correctos para Ubuntu Server 16.04 y 14.04 de 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. (Recomendado) Ejecute uno de los comandos siguientes para determinar si SSM Agent se está ejecutando.

Ubuntu Server 16.04

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

Ubuntu Server 14.04

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

En la mayoría de los casos, el comando informa de que el agente se está ejecutando.

En casos excepcionales, el comando informa de que el agente está instalado pero no se está ejecutando, como se muestra en el siguiente ejemplo.

6. Ejecute uno de los siguientes comando para iniciar el servicio si el comando anterior devuelve `amazon-ssm-agent is stopped`, `inactive` o `disabled`.

Ubuntu Server 16.04:

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

Ubuntu Server 14.04:

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

Crear comandos de instalación personalizados para SSM Agent en Ubuntu Server 16.04 y 14.04 32 bits (deb) en una región

Cuando se instala SSM Agent en varias instancias mediante un script o una plantilla, se recomienda utilizar los archivos de instalación almacenados en la Región de AWS en la que se está trabajando.

Para los siguientes comandos, proporcionamos ejemplos en los que se utiliza un bucket de S3 de acceso público de la región Este de EE. UU. (Ohio) (`us-east-2`).

> **Tip**
>
> También puede crear una URL regional personalizada para reemplazar una URL global en el procedimiento Comandos de instalación rápida para SSM Agent en Ubuntu Server 16.04 y 14.04 de 32 bits (deb) que aparece antes en este tema.

En el siguiente comando, reemplace *region* con su propia información. Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de 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
```

Consulte el siguiente ejemplo.

```
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
```

Determinación de la versión correcta de SSM Agent que se debe instalar en instancias de Ubuntu Server 16.04 de 64 bits

> **Important**
>
> Antes de instalar SSM Agent en una versión de 64 bits de Ubuntu Server, asegúrese de que está utilizando las herramientas de instalación de correcciones. A partir de las imágenes de máquina de Amazon (AMI) que están identificadas con 20180627, SSM Agent está preinstalado en la versión 16.04 mediante paquetes Snap. En las instancias creadas a partir de AMI anteriores, debe instalarse SSM Agent mediante paquetes de instalador deb. Para obtener más información, consulte Determinación de la versión correcta de SSM Agent que se debe instalar en instancias de Ubuntu Server 16.04 de 64 bits.

Tenga en cuenta que, si una instancia tiene más de una instalación de SSM Agent (por ejemplo, una instalada con un Snap y otra instalada con un instalador deb), las operaciones del agente no funcionarán de manera correcta.

Puede verificar la fecha de creación del ID de la AMI de origen de una instancia mediante cualquiera de los siguientes métodos. Estos procedimientos se aplican solo a las AMIs administradas de AWS.

Verificación de la fecha de creación del ID de la AMI de origen (consola)

1. Abra la consola de Amazon EC2 en https://console.aws.amazon.com/ec2/.
2. En el panel de navegación izquierdo, elija instancias.
3. Seleccione una instancia.
4. En la pestaña Details (Detalles), compruebe si hay un identificador YYYYMMDD en el valor para el campo AMI name (Nombre de la AMI). Por ejemplo: `ubuntu/images/hvm-ssd/ubuntu-xenial-16.04-amd64-server-20180627`.

Verificación de la fecha de creación del ID de la AMI de origen (AWS CLI)

- Ejecute el siguiente comando de la .

```
aws ec2 describe-images --image-ids ami-id
```

ami-id representa el ID de una AMI que proporciona AWS, como, por ejemplo, `ami-07c8bc5c1ce9598c3`.

Si la operación se realiza correctamente, este comando regresa información como la siguiente, en la que puede verificar los campos `CreationDate` y `Name` para obtener información.

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

Configuración de SSM Agent para utilizar un proxy en nodos de Linux

Puede configurar AWS Systems Manager Agent (SSM Agent) para que se comunique a través de un proxy HTTP, mediante la creación de un archivo de configuración de anulación y la inclusión de las configuraciones `http_proxy`, `https_proxy` y `no_proxy` en el archivo. Un archivo de anulación también guarda la configuración del proxy si se instalan las versiones nuevas o anteriores del SSM Agent. En esta sección, se incluyen procedimientos para crear un archivo de anulación en los entornos upstart y systemd. Si piensa utilizar Session Manager, tenga en cuenta que los servidores proxy HTTPS no son compatibles.

Temas

- Configurar el SSM Agent para utilizar un proxy (upstart)
- Configurar el SSM Agent para utilizar un proxy (systemd)

Configurar el SSM Agent para utilizar un proxy (upstart)

Utilice el siguiente procedimiento para crear un archivo de configuración de anulación para un entorno `upstart`.

Para configurar SSM Agent para utilizar un proxy (upstart)

1. Conéctese a la instancia administrada en la que ha instalado SSM Agent.
2. Abra un editor simple como VIM y, en función de si utiliza un servidor proxy HTTP o un servidor proxy HTTPS, agregue una de las siguientes configuraciones.

   Para un servidor proxy HTTP:

   ```
   env http_proxy=http://hostname:port
   env https_proxy=http://hostname:port
   env no_proxy=IP address for instance metadata services (IMDS)
   ```

Para un servidor 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**
>
> Agregue el valor `no_proxy` al archivo y especifique la dirección IP. La dirección IP de `no_proxy` es el punto de conexión de los servicios de metadatos de instancias (IMDS) de Systems Manager. Si no especifica `no_proxy`, las llamadas a Systems Manager adoptan la identidad del servicio del proxy (si la opción alternativa de IMDSv1 está habilitada) o se produce un error en las llamadas a Systems Manager (si se aplica IMDSv2).
>
> - Para IPv4, especifique `no_proxy=169.254.169.254`.
> - Para IPv6, especifique `no_proxy=[fd00:ec2::254]`. La dirección IPv6 del servicio de metadatos de instancia es compatible con los comandos IMDSv2. Solo se puede acceder a la dirección IPv6 en instancias integradas en AWS Nitro System. Para obtener más información, consulte Funcionamiento de Servicio de metadatos de instancia versión 2 en la Guía del usuario de Amazon EC2.

3. Guarde el archivo con el nombre `amazon-ssm-agent.override` en la siguiente ubicación: `/etc/init/`
4. Detenga y reinicie SSM Agent con los siguientes comandos.

```
sudo service stop amazon-ssm-agent
sudo service start amazon-ssm-agent
```

> **Note**
>
> Para obtener más información sobre el uso de archivos `.override` en entornos Upstart, consulte init: Upstart init daemon job configuration.

Configurar el SSM Agent para utilizar un proxy (systemd)

Utilice el siguiente procedimiento para configurar SSM Agent para utilizar un proxy en un entorno `systemd`.

> **Note**
>
> Algunos de los pasos de este procedimiento contienen instrucciones explícitas para las instancias de Ubuntu Server donde SSM Agent se instaló mediante Snap.

1. Conéctese a la instancia en la que ha instalado el SSM Agent.
2. Ejecute uno de los siguientes comandos, en función del tipo de sistema operativo.

   - En las instancias de Ubuntu Server donde SSM Agent se instala mediante un Snap:

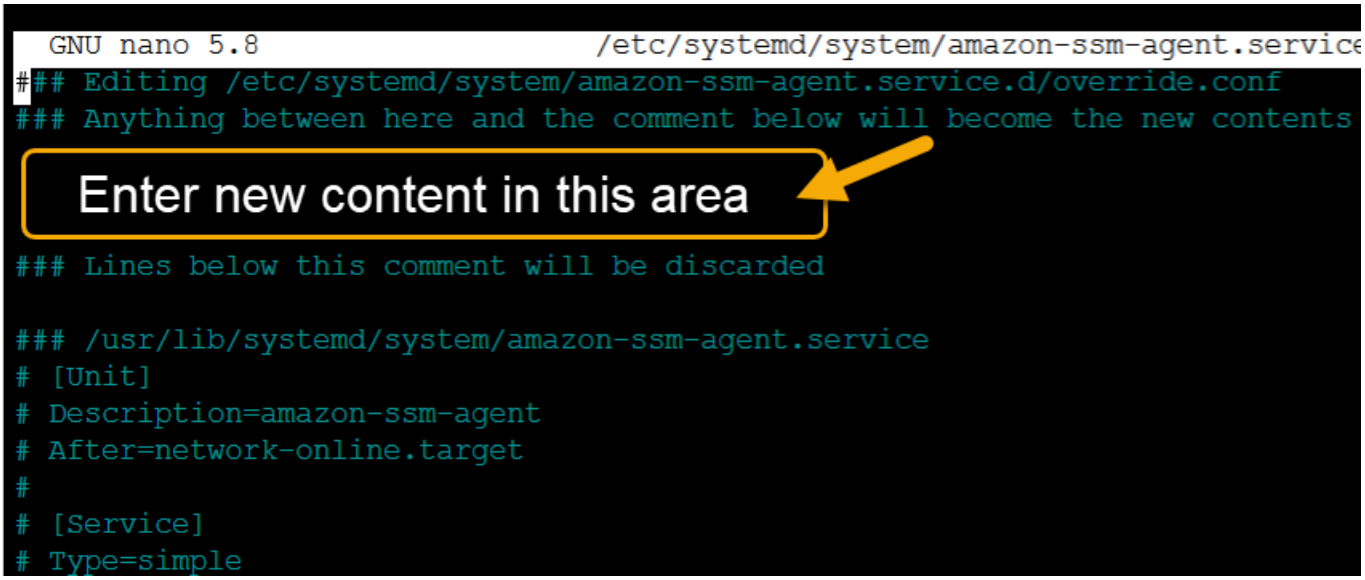
     ```
     sudo systemctl edit snap.amazon-ssm-agent.amazon-ssm-agent
     ```

     En otros sistemas operativos:

     ```
     sudo systemctl edit amazon-ssm-agent
     ```

3. Abra un editor simple como VIM y, en función de si utiliza un servidor proxy HTTP o un servidor proxy HTTPS, agregue una de las siguientes configuraciones.

   Asegúrese de introducir la información que aparece encima del comentario que dice “`### Lines below this comment will be discarded`”, como se indica en la siguiente imagen.



Para un servidor 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)"
```

Para un servidor 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

Agregue el valor `no_proxy` al archivo y especifique la dirección IP. La dirección IP de `no_proxy` es el punto de conexión de los servicios de metadatos de instancias (IMDS) de Systems Manager. Si no especifica `no_proxy`, las llamadas a Systems Manager adoptan la identidad del servicio del proxy (si la opción alternativa de IMDSv1 está habilitada) o se produce un error en las llamadas a Systems Manager (si se aplica IMDSv2).

- Para IPv4, especifique `no_proxy=169.254.169.254`.

- Para IPv6, especifique `no_proxy=[fd00:ec2::254]`. La dirección IPv6 del servicio de metadatos de instancia es compatible con los comandos IMDSv2. Solo se puede acceder a la dirección IPv6 en instancias integradas en AWS Nitro System. Para obtener más información, consulte Funcionamiento de Servicio de metadatos de instancia versión 2 en la Guía del usuario de Amazon EC2.

4. Guarde los cambios. El sistema crea de forma automática uno de los siguientes archivos, en función del tipo de sistema operativo.

   - En las instancias de Ubuntu Server donde SSM Agent se instala mediante un Snap:

     `/etc/systemd/system/snap.amazon-ssm-agent.amazon-ssm-agent.service.d/`**`override.conf`**

   - En instancias de Amazon Linux 2 y Amazon Linux 2023:

     `/etc/systemd/system/amazon-ssm-agent.service.d/`**`override.conf`**

   - En otros sistemas operativos:

     `/etc/systemd/system/amazon-ssm-agent.service.d/`**`amazon-ssm-agent.override`**

5. Reinicie SSM Agent con uno de los siguientes comandos, en función del tipo de sistema operativo.

   - En las instancias de Ubuntu Server donde se instala mediante un Snap:

     ```
     sudo systemctl daemon-reload && sudo systemctl restart snap.amazon-ssm-agent.amazon-ssm-agent
     ```

   - En otros sistemas operativos:

     ```
     sudo systemctl daemon-reload && sudo systemctl restart amazon-ssm-agent
     ```

> **Note**
>
> Para obtener más información sobre cómo trabajar con archivos `.override` en entornos systemd, consulte Modificación de archivos de unidades existentes en la Guía del administrador de sistemas de Red Hat Enterprise Linux 7.

Cómo utilizar SSM Agent en instancias de EC2 para macOS

AWS Systems Manager (SSM Agent) procesa solicitudes de Systems Manager y configura el equipo tal y como se especifica en la solicitud. Utilice los siguientes procedimientos para instalar, configurar o desinstalar SSM Agent para macOS.

> **Note**
>
> SSM Agent se instala de forma predeterminada en las Amazon Machine Images (AMIs) para macOS. No es necesario instalar SSM Agent en una instancia de Amazon Elastic Compute Cloud (Amazon EC2) para macOS a menos que lo haya desinstalado.

El código fuente del SSM Agent está disponible en GitHub para que pueda adaptar el agente a sus necesidades. Le recomendamos enviar solicitudes de inserción para los cambios que le gustaría que incluyamos. No obstante, AWS no admite la ejecución de copias modificadas de este software.

> **Note**
>
> Para ver los detalles de las diferentes versiones del SSM Agent, consulte las notas de la versión.

Antes de instalar SSM Agent de forma manual en un sistema operativo macOS, revise la siguiente información.

- SSM Agent se instala de forma predeterminada en las instancias EC2 y Amazon Machine Images es compatible con Systems Manager. Para obtener más información, consulte la lista de sistemas operativos compatibles para macOS.

  No es necesario instalar SSM Agent de forma manual en instancias EC2 de macOS a menos que se haya desinstalado.
- No todas las Regiones de AWS admiten instancias de EC2 para macOS. Para obtener listas de regiones en las que se admiten instancias EC2 basadas en x86 y M1 para macOS, consulte macOS cargas de trabajo en las preguntas frecuentes de Amazon EC2.
- Cada vez que se agregan herramientas nuevas a Systems Manager o se actualizan las herramientas existentes, se lanza una versión actualizada de SSM Agent. No utilizar la versión más reciente del agente puede impedir que el nodo administrado utilice diversas herramientas

y características de Systems Manager. Por este motivo, se recomienda automatizar el proceso de mantener SSM Agent actualizado en los equipos. Para obtener más información, consulta Automatización de las actualizaciones de SSM Agent. Suscríbase a la página SSM Agent Release Notes en GitHub para recibir notificaciones sobre las actualizaciones de SSM Agent.

Temas

- Instalación y desinstalación manual de SSM Agent en instancias de EC2 para macOS

Instalación y desinstalación manual de SSM Agent en instancias de EC2 para macOS

Conéctese a su instancia de macOS y realice los siguientes pasos para instalar AWS Systems Manager Agent (SSM Agent). Lleve a cabo estos pasos en cada instancia que ejecutará comandos con Systems Manager. Los comandos proporcionados en este procedimiento también se pueden pasar a las instancias de Amazon EC2 como scripts a través de los datos del usuario.

Antes de empezar

Instale `wget` con Homebrew.

Para instalar el SSM Agent en macOS

1. Descargue el archivo de instalación del agente para las instancias x86_64 con el siguiente comando.

   En el siguiente comando, reemplace *`region`* con su propia información. Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

   ```
   sudo wget https://s3.region.amazonaws.com/amazon-ssm-region/latest/darwin_amd64/
   amazon-ssm-agent.pkg
   ```

   Para las instancias Apple silicon, utilice el siguiente comando.

   ```
   sudo wget https://s3.region.amazonaws.com/amazon-ssm-region/latest/darwin_arm64/
   amazon-ssm-agent.pkg
   ```

   A continuación se muestra un ejemplo.

```
sudo wget https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
darwin_amd64/amazon-ssm-agent.pkg
```

2. Utilice el siguiente comando para ejecutar el instalador de SSM Agent.

   x86_64:

   ```
   sudo installer -pkg amazon-ssm-agent.pkg -target /
   ```

3. Compruebe el estado del agente.

   Para determinar si SSM Agent se está ejecutando, verifique el registro del agente en `/var/log/amazon/ssm/amazon-ssm-agent.log`.

4. Ejecute el siguiente comando para iniciar el servicio si el registro del agente indica que “amazon-ssm-agent is stopped” (amazon-ssm-agent se ha detenido).

   ```
   sudo launchctl load -w /Library/LaunchDaemons/com.amazon.aws.ssm.plist && sudo
    launchctl start com.amazon.aws.ssm
   ```

> **Important**
>
> Cada vez que se agregan herramientas nuevas a Systems Manager o se actualizan las herramientas existentes, se lanza una versión actualizada de SSM Agent. No utilizar la versión más reciente del agente puede impedir que el nodo administrado utilice diversas herramientas y características de Systems Manager. Por este motivo, se recomienda automatizar el proceso de mantener SSM Agent actualizado en los equipos. Para obtener más información, consulta Automatización de las actualizaciones de SSM Agent. Suscríbase a la página SSM Agent Release Notes en GitHub para recibir notificaciones sobre las actualizaciones de SSM Agent.

Desinstalación de SSM Agent de las instancias de macOS

macOS no admite de forma nativa la desinstalación de archivos `PKG`. Para desinstalar AWS Systems Manager Agent (SSM Agent) de una instancia de Amazon Elastic Compute Cloud (Amazon EC2) para macOS, puede utilizar el script administrado de AWS desde la siguiente ubicación.

https://github.com/aws/amazon-ssm-agent/blob/mainline/Tools/src/update/darwin/uninstall.sh

Cómo utilizar SSM Agent en instancias de EC2 para Windows Server

AWS Systems Manager Agent (SSM Agent) esta preinstalado, de forma predeterminada, en las Amazon Machine Images (AMIs) para Windows Server que proporciona AWS. Se proporciona soporte para las siguientes versiones de sistemas operativos (SO).

- Windows Server AMIs 2008-2012 R2 publicadas en noviembre de 2016 o posteriormente
- Windows Server 2016, 2019, 2022 (no se incluyen las versiones Nano) y 2025

Notas de soporte para versiones anteriores

Las AMIs de Windows Server publicadas antes de noviembre de 2016 utilizan el servicio EC2Config para procesar solicitudes y configurar instancias.

A menos que tenga un motivo específico para utilizar el servicio EC2Config, o una versión anterior de SSM Agent, para procesar las solicitudes de Systems Manager, le recomendamos que descargue e instale la versión más reciente de SSM Agent en cada instancia de Amazon Elastic Compute Cloud (Amazon EC2) o equipo que no sea de EC2 que esté configurado para Systems Manager en un entorno híbrido y multinube.

A partir del 14 de enero de 2020, Windows Server 2008 ya no será compatible para obtener actualizaciones de características o de seguridad de Microsoft. Amazon Machine Images heredadas (AMIs) para Windows Server 2008 y 2008 R2 aún incluyen la versión 2 de SSM Agent preinstalada, pero Systems Manager ya no admite oficialmente las versiones 2008 ni actualiza el agente para estas versiones de Windows Server. Además, es posible que la versión 3 de SSM Agent no sea compatible con todas las operaciones en Windows Server 2008 y 2008 R2. La versión final oficialmente admitida de SSM Agent para las versiones Windows Server 2008 es 2.3.1644.0.

Cómo mantener SSM Agent actualizado

Cada vez que se agregan herramientas nuevas a Systems Manager o se actualizan las herramientas existentes, se lanza una versión actualizada de SSM Agent. No utilizar la versión más reciente del agente puede impedir que el nodo administrado utilice diversas herramientas y características de Systems Manager. Por este motivo, se recomienda automatizar el proceso de mantener SSM Agent actualizado en los equipos. Para obtener más información, consulta Automatización de las

actualizaciones de SSM Agent. Suscríbase a la página SSM Agent Release Notes en GitHub para recibir notificaciones sobre las actualizaciones de SSM Agent.

Para ver los detalles de las diferentes versiones del SSM Agent, consulte las notas de la versión.

Temas

- Cómo instalar y desinstalar de forma manual SSM Agent en instancias de EC2 para Windows Server
- Configurar el SSM Agent para usar un proxy para las instancias de Windows Server

Cómo instalar y desinstalar de forma manual SSM Agent en instancias de EC2 para Windows Server

AWS Systems Manager Agent (SSM Agent) está preinstalado, de manera predeterminada, en las siguientes Amazon Machine Images (AMIs) para Windows Server proporcionadas por Amazon:

- Windows Server AMIs 2008-2012 R2 publicadas en noviembre de 2016 o posteriormente
- Windows Server 2016, 2019, 2022 (no se incluyen las versiones Nano) y 2025

Cómo instalar manualmente SSM Agent en instancias EC2 para Windows Server

Si fuera necesario, puede descargar e instalar de forma manual la versión más reciente de SSM Agent en la instancia de Amazon Elastic Compute Cloud (Amazon EC2) para Windows Server mediante el siguiente procedimiento. Los comandos proporcionados en este procedimiento también se pueden pasar a las instancias de Amazon EC2 como scripts a través de los datos del usuario.

SSM Agent requiere Windows PowerShell 3.0 o una versión posterior para ejecutar determinados documentos de AWS Systems Manager (documentos de SSM) en instancias de Windows Server (por ejemplo, el documento `AWS-ApplyPatchBaseline` heredado). Compruebe que sus instancias de Windows Server ejecutan Windows Management Framework 3.0 o posterior. Este marco contiene Windows PowerShell. Para obtener más información, consulte Windows Management Framework 3.0.

> **Note**
>
> Este procedimiento se aplica a la instalación o reinstalación del SSM Agent en una instancia de EC2 para Windows Server. Si necesita instalar el agente en un servidor local o en una

máquina virtual (VM) para que se pueda utilizar con Systems Manager, consulte Cómo instalar SSM Agent en nodos de Windows híbridos.

Cómo instalar de forma manual la versión más reciente de SSM Agent en las instancias EC2 para Windows Server

1. Conéctese a la instancia utilizando Escritorio remoto o Windows PowerShell. Para obtener más información, consulte Conexión a una instancia en la Guía del usuario de Amazon EC2.
2. Descargue en la instancia la versión más reciente del SSM Agent. Puede realizar la descarga con comandos de PowerShell o mediante un enlace de descarga directa.

> **Note**
>
> Las URL de este paso permiten descargar SSM Agent desde cualquier Región de AWS. Si desea descargar el agente desde una región específica, utilice la URL específica de la región: `https://amazon-ssm-region.s3.region.amazonaws.com/latest/windows_amd64/AmazonSSMAgentSetup.exe`
> *region* representa el identificador de una Región de AWS compatible con AWS Systems Manager, como `us-east-2` para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

PowerShell

Ejecute los siguientes tres comandos de PowerShell en orden. Estos comandos le permiten descargar SSM Agent sin ajustar la configuración de seguridad mejorada de Internet Explorer (IE) y, luego, instalar el agente y eliminar el archivo de instalación.

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" `
    -Wait
```

```
rm -Force $env:USERPROFILE\Desktop\SSMAgent_latest.exe
```

Descarga directa

Descargue la versión más reciente de SSM Agent en su instancia mediante el siguiente enlace. Si lo desea, actualice esta URL con una URL específica de la Región de AWS.

https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/windows_amd64/AmazonSSMAgentSetup.exe

Ejecute el archivo `AmazonSSMAgentSetup.exe` descargado para instalar el SSM Agent.

3. Para iniciar o reiniciar el SSM Agent, envíe el siguiente comando en PowerShell:

```
Restart-Service AmazonSSMAgent
```

> **Note**
>
> Para desinstalar el SSM Agent de una instancia de Windows Server, abra Panel de control, Programas. Elija la opción Desinstalar un programa. Abra el menú contextual de Amazon (haga clic con el botón derecho)SSM Agent y elija Uninstall (Desinstalar).

Configurar el SSM Agent para usar un proxy para las instancias de Windows Server

La información de este tema se aplica a las instancias de Windows Server creadas en noviembre de 2016 o a partir de esta fecha que no utilicen la opción de instalación Nano. Si piensa utilizar Session Manager, tenga en cuenta que los servidores proxy HTTPS no son compatibles.

> **Note**
>
> A partir del 14 de enero de 2020, Windows Server 2008 ya no será compatible para obtener actualizaciones de características o de seguridad de Microsoft. Amazon Machine Images heredadas (AMIs) para Windows Server 2008 y 2008 R2 aún incluyen la versión 2 de SSM Agent preinstalada, pero Systems Manager ya no admite oficialmente las versiones 2008 ni actualiza el agente para estas versiones de Windows Server. Además, es posible que la versión 3 de SSM Agent no sea compatible con todas las operaciones en Windows Server 2008 y 2008 R2. La versión final oficialmente admitida de SSM Agent para las versiones Windows Server 2008 es 2.3.1644.0.

Antes de empezar

Antes de configurar SSM Agent para usar un proxy, tenga en cuenta la siguiente información importante.

En el siguiente procedimiento, debe ejecutar un comando para configurar SSM Agent para usar un proxy. El comando incluye un valor `no_proxy` con una dirección IP. La dirección IP es el punto de conexión de los servicios de metadatos de instancias (IMDS) de Systems Manager. Si no especifica `no_proxy`, las llamadas a Systems Manager adoptan la identidad del servicio del proxy (si la opción alternativa de IMDSv1 está habilitada) o se produce un error en las llamadas a Systems Manager (si se aplica IMDSv2).

- Para IPv4, especifique `no_proxy=169.254.169.254`.
- Para IPv6, especifique `no_proxy=[fd00:ec2::254]`. La dirección IPv6 del servicio de metadatos de instancia es compatible con los comandos IMDSv2. Solo se puede acceder a la dirección IPv6 en instancias integradas en AWS Nitro System. Para obtener más información, consulte Funcionamiento de Servicio de metadatos de instancia versión 2 en la Guía del usuario de Amazon EC2.

Para configurar el SSM Agent para utilizar un proxy

1. Con Escritorio remoto o Windows PowerShell, conéctese a la instancia que desea configurar para utilizar un proxy.
2. Ejecute el siguiente bloque de comandos en PowerShell. Reemplace *hostname* y *port* por la información sobre su 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
```

Después de ejecutar el comando anterior, puede revisar los registros de SSM Agent para confirmar que se aplicó la configuración del proxy. Las entradas de los registros tienen un aspecto similar al siguiente. Para obtener más información acerca de los registros de SSM Agent, consulte Visualización de registros de 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
```

Para restablecer la configuración de proxy del SSM Agent

1. Utilizando el Escritorio remoto o Windows PowerShell, conéctese a la instancia que desee configurar.
2. Si se ha conectado con Escritorio remoto, lance PowerShell como administrador.
3. Ejecute el siguiente bloque de comandos en PowerShell.

```
Remove-ItemProperty -Path HKLM:\SYSTEM\CurrentControlSet\Services\AmazonSSMAgent -
Name Environment
Restart-Service AmazonSSMAgent
```

Prioridad de la configuración del proxy en SSM Agent

Al configurar los ajustes del proxy para SSM Agent en las instancias de Windows Server, es importante que conozca cómo estos ajustes se evalúan y aplican a la configuración del agente cuando se inicia SSM Agent. La forma en que establece la configuración del proxy para una instancia de Windows Server puede determinar si otras configuraciones pueden anular la configuración prevista. El agente utiliza la primera configuración de proxy que encuentra.

> **Important**
>
> SSM Agent se comunica mediante el protocolo HTTPS. Por este motivo, debe configurar el parámetro `HTTPS proxy` mediante una de las siguientes opciones de configuración.

La configuración del proxy de SSM Agent se evalúa en el siguiente orden.

1. Configuración del registro de `AmazonSSMAgent` (`HKLM:\SYSTEM\CurrentControlSet\Services\AmazonSSMAgent`)
2. Variables de entorno de `System` (`http_proxy`, `https_proxy`, `no_proxy`)
3. Variables de entorno de cuenta de usuario de `LocalSystem` (`http_proxy`, `https_proxy`, `no_proxy`)
4. Configuración del navegador (`HTTP`, `secure`, `exceptions`)
5. Configuración del proxy `WinHTTP` (`http=`, `https=`, `bypass-list=`)

Configuración del proxy de SSM Agent y servicios de Systems Manager

Si configuró el SSM Agent para usar un proxy y utiliza herramientas de AWS Systems Manager, como Run Command y Patch Manager, que utilizan PowerShell o el cliente de Windows Update durante su ejecución en instancias de Windows Server, configure los ajustes del proxy adicionales. De lo contrario, la operación podría producir un error porque PowerShell y el cliente de Windows Update utilizan la configuración del proxy y estos ajustes no se heredan de la configuración del proxy de SSM Agent.

Para Run Command, configure los ajustes del proxy `WinINet` en sus instancias de Windows Server. La comandos `[System.Net.WebRequest]` se proporcionan por sesión. Para aplicar estas configuraciones a los comandos de red posteriores que se ejecutan en Run Command, estos comandos deben preceder a otros comandos de PowerShell en la misma entrada de un complemento `aws:runPowershellScript`.

Los siguientes comandos de PowerShell devuelven la configuración del proxy de `WinINet` actual y aplican la configuración del proxy a `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
```

Para Patch Manager, debe configurar los ajustes del proxy de todo el sistema para que el cliente de Windows Update pueda buscar y descargar las actualizaciones. Se recomienda utilizar Run Command para ejecutar los siguientes comandos, ya que se ejecutan en la cuenta SYSTEM y la configuración se aplica a todo el sistema. Los siguientes comandos `netsh` regresan la configuración del proxy actual y aplican la configuración del proxy al sistema local.

```
netsh winhttp show proxy

netsh winhttp set proxy proxy-server="hostname:port" bypass-list="169.254.169.254"
```

Para obtener más información acerca del uso de Run Command, consulte AWS Systems Manager Run Command.

Verificación del estado de SSM Agent e inicio del agente

En este tema se enumeran los comandos para verificar si AWS Systems Manager Agent (SSM Agent) se ejecuta en cada sistema operativo compatible. También proporciona los comandos para iniciar el agente si no se está ejecutando.

Sistema operativo	Comando para verificar el estado de SSM Agent	Comando para iniciar SSM Agent
Amazon Linux 1	`sudo status amazon-ssm-agent`	`sudo start amazon-ssm-agent`
Amazon Linux 2, Amazon Linux 2022, y Amazon Linux 2023	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`
CentOS 6.x	`sudo status amazon-ssm-agent`	`sudo start amazon-ssm-agent`
CentOS 7.x y 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 y 10	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`
macOS	Verifique el archivo de registros del agente en `/var/log/amazon/ssm/amazon-ssm-agent.log`	`sudo launchctl load -w /Library/LaunchDaemons/com.amazon.aws.ssm.plist`

Sistema operativo	Comando para verificar el estado de SSM Agent	Comando para iniciar SSM Agent
		`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`
Red Hat Enterprise Linux (RHEL) 7.x, 8.x, 9.x, and 10.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 (todas las instancias) y 16.04 (instancias de 32 bits)	`sudo status amazon-ssm-agent`	`sudo start amazon-ssm-agent`
Instancias de 64 bits de Ubuntu Server 16.04 (instalación de paquetes deb)	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`

Sistema operativo	Comando para verificar el estado de SSM Agent	Comando para iniciar SSM Agent
Ubuntu Server 16.04 LTS, 18.04, 20.04, 20.10 STR, 22.04 LTS, 23.10, 24.04 LTS, 24.0, y 25.04	`sudo systemctl status snap.amazon-ssm-agent.amazon-ssm-agent.service`	`sudo snap start amazon-ssm-agent`
Windows Server	Ejecutar en PowerShell: `Get-Service AmazonSSMAgent`	Ejecutar en el modo de administrador de PowerShell: `Start-Service AmazonSSMAgent`

Más información

- Uso de SSM Agent en instancias de EC2 para Linux
- Cómo utilizar SSM Agent en instancias de EC2 para Windows Server
- Verificación del número de versión de SSM Agent

Verificación del número de versión de SSM Agent

Ciertas funcionalidades de AWS Systems Manager tienen requisitos previos que incluyen una versión de Systems Manager Agent (SSM Agent) mínima instalada en los nodos administrados. Puede obtener la versión de SSM Agent instalada en la actualidad en los nodos administrados mediante la consola de Systems Manager o iniciando sesión en los nodos administrados.

> **Note**
>
> Para ver los detalles sobre las versiones anteriores, consulte SSM Agent release notes en GitHub.

En los procedimientos siguientes se describe cómo obtener la versión de SSM Agent instalada actualmente en los nodos administrados.

Para verificar el número de versión de SSM Agent instalada en un nodo administrado

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. En la columna SSM Agent version (Versión de SSM Agent), anote el número de Agent version (Versión del agente).

Para obtener la versión de SSM Agent instalada actualmente desde el sistema operativo

Para obtener la versión de SSM Agent instalada en la actualidad desde el sistema operativo, elija entre las siguientes pestañas.

Amazon Linux 1, Amazon Linux 2, and Amazon Linux 2023

> **Note**
>
> Este comando varía en función del administrador de paquetes del sistema operativo.

1. Inicie sesión en el nodo administrado.
2. Ejecute el siguiente comando.

```
yum info amazon-ssm-agent
```

Este comando devuelve un resultado similar al siguiente.

```
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Installed Packages
Name        : amazon-ssm-agent
Arch        : x86_64
Version     : 3.0.655.0
```

CentOS

1. Inicie sesión en el nodo administrado.
2. Ejecute el siguiente comando para CentOS 6 y 7.

```
yum info amazon-ssm-agent
```

Este comando devuelve un resultado similar al siguiente.

```
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Installed Packages
Name        : amazon-ssm-agent
Arch        : x86_64
Version     : 3.0.655.0
```

Servidor Debian

1. Inicie sesión en el nodo administrado.
2. Ejecute el siguiente comando.

```
apt list amazon-ssm-agent
```

Este comando regresa un resultado similar al siguiente.

```
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. Inicie sesión en el nodo administrado.
2. Ejecute el siguiente comando.

```
pkgutil --pkg-info com.amazon.aws.ssm
```

RHEL

1. Inicie sesión en el nodo administrado.

2. Ejecute el siguiente comando para RHEL 6, 7, 8 y 9.

```
yum info amazon-ssm-agent
```

Este comando devuelve un resultado similar al siguiente.

```
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Installed Packages
Name        : amazon-ssm-agent
Arch        : x86_64
Version     : 3.0.655.0
```

Ejecute el siguiente comando para la utilidad de paquetes DNF.

```
dnf info amazon-ssm-agent
```

SLES

1. Inicie sesión en el nodo administrado.
2. Ejecute el siguiente comando para SLES 12 y 15.

```
zypper info amazon-ssm-agent
```

Este comando devuelve un resultado similar al siguiente.

```
Loading repository data...
Reading installed packages...
Information for package amazon-ssm-agent:
----------------------------------------
Repository : @System
Name : amazon-ssm-agent
Version : 3.0.655.0-1
```

Servidor Ubuntu

> **Note**
>
> Para comprobar si su instancia de Ubuntu Server 16.04 utiliza paquetes deb o Snap, consulte Instalación manual de SSM Agent en instancias de Ubuntu Server.

1. Inicie sesión en el nodo administrado.
2. Ejecute el siguiente comando para instancias de 64 bits de Ubuntu Server 16.04 y 14.04 (con un paquete de instalador deb).

   ```
   apt list amazon-ssm-agent
   ```

   Este comando devuelve un resultado similar al siguiente.

   ```
   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
   ```

   Ejecute el siguiente comando para Ubuntu Server 16.04 LTS de 64 bits (Snap), 18.04, 20.04, 20.10 STR, 22.04 LTS, 23.10, 24.04 LTS, 24.0 y 25.04.

   ```
   sudo snap list amazon-ssm-agent
   ```

   Este comando devuelve un resultado similar al siguiente.

   ```
   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. Inicie sesión en el nodo administrado.

2. Ejecute los comandos de PowerShell siguientes:

```
& "C:\Program Files\Amazon\SSM\amazon-ssm-agent.exe" -version
```

Este comando regresa un resultado similar al siguiente.

```
SSM Agent version: 3.1.804.0
```

Recomendamos utilizar la versión más reciente de SSM Agent para que pueda beneficiarse de capacidades nuevas o actualizadas. Para asegurarse de que las instancias administradas siempre ejecutan la versión más actualizada de SSM Agent, puede automatizar el proceso de actualización de SSM Agent. Para obtener más información, consulte Automatización de las actualizaciones de SSM Agent.

Visualización de registros de SSM Agent

AWS Systems Manager Agent (SSM Agent) escribe información acerca de ejecuciones, comandos, acciones programadas, errores y estados en los archivos de registros en cada nodo administrado. Puede ver los archivos de registros si se conecta a un nodo administrado de forma manual o puede enviar los registros a Amazon CloudWatch Logs de forma automática. Para obtener más información sobre el envío de registros a Registros de CloudWatch, consulte Registro y monitorización en AWS Systems Manager.

Puede ver los registros de SSM Agent de los nodos administrados en las siguientes ubicaciones.

Linux and macOS

```
/var/log/amazon/ssm/
```

Windows

```
%PROGRAMDATA%\Amazon\SSM\Logs\
```

Para los nodos administrados de Linux, los archivos de SSM Agent `stderr` y `stdout` se escriben en el siguiente directorio: `/var/lib/amazon/ssm/`.

Para los nodos administrados de Windows, los archivos de SSM Agent `stderr` y `stdout` se escriben en el siguiente directorio: `%PROGRAMDATA%\Amazon\SSM\InstanceData\`.

Para obtener más información acerca de cómo permitir el registro de depuración de SSM Agent, consulte Permiso del registro de depuración de SSM Agent.

Para obtener más información acerca de la configuración de `cihub/seelog`, consulte la Seelog Wiki en GitHub. Para ver ejemplos de configuraciones de `cihub/seelog`, consulte el repositorio cihub/seelog examples en GitHub.

Permiso del registro de depuración de SSM Agent

Utilice el siguiente procedimiento para permitir el registro de depuración de SSM Agent en los nodos administrados.

Linux and macOS

Para permitir el registro de depuración de SSM Agent en nodos administrados de Linux y macOS

1. Utilice Session Manager, una herramienta de AWS Systems Manager, para conectarse al nodo administrado en el que desea permitir el registro de depuración o inicie sesión en el nodo administrado. Para obtener más información, consulte Uso de Session Manager.
2. Localice el archivo seelog.xml.template.

   Linux:

   En la mayoría de los tipos de nodos administrados de Linux, el archivo se ubica en el directorio `/etc/amazon/ssm/seelog.xml.template`.

   En Ubuntu Server 20.10 STR y 20.04, 18.04 y 16.04 LTS, el archivo se ubica en el directorio `/snap/amazon-ssm-agent/current/seelog.xml.template`. Copie este archivo del directorio `/snap/amazon-ssm-agent/current/` en el directorio `/etc/amazon/ssm/` antes de realizar cualquier cambio.

   macOS:

   En los tipos de instancias de macOS, el archivo se ubica en el directorio `/opt/aws/ssm/seelog.xml.template`.
3. Cambie el nombre del archivo de `seelog.xml.template` a `seelog.xml`.

> **Note**
>
> En Ubuntu Server 20.10 STR y 20.04, 18.04 y 16.04 LTS, el archivo `seelog.xml` se debe crear en el directorio `/etc/amazon/ssm/`. Ejecute los siguientes comandos para crear este directorio y archivo.
>
> ```
> sudo mkdir -p /etc/amazon/ssm
> ```
>
> ```
> sudo cp -p /snap/amazon-ssm-agent/current/seelog.xml.template /etc/amazon/ssm/seelog.xml
> ```

4. Edite el archivo `seelog.xml` para cambiar el comportamiento de registro predeterminado. Cambie el valor de minlevel de info (información) a debug (depurar), tal y como se muestra en el siguiente ejemplo.

   ```
   <seelog type="adaptive" mininterval="2000000" maxinterval="100000000" critmsgcount="500" minlevel="debug">
   ```

5. (Opcional) Reinicie SSM Agent con el siguiente comando.

   Linux:

   ```
   sudo service amazon-ssm-agent restart
   ```

   macOS:

   ```
   sudo /opt/aws/ssm/bin/amazon-ssm-agent restart
   ```

Windows

Para permitir el registro de depuración de SSM Agent en nodos administrados de Windows Server

1. Utilice Session Manager para conectarse al nodo administrado en el que desea permitir el registro de depuración o inicie sesión en el nodo administrado. Para obtener más información, consulte Uso de Session Manager.

2. Realice una copia del archivo seelog.xml.template. Cambie el nombre de la copia por seelog.xml. El archivo se encuentra en el siguiente directorio:

   ```
   %PROGRAMFILES%\Amazon\SSM\seelog.xml.template
   ```

3. Edite el archivo `seelog.xml` para cambiar el comportamiento de registro predeterminado. Cambie el valor de minlevel de info (información) a debug (depurar), tal y como se muestra en el siguiente ejemplo.

   ```
   <seelog type="adaptive" mininterval="2000000" maxinterval="100000000"
   critmsgcount="500" minlevel="debug">
   ```

4. Localice la siguiente entrada.

   ```
   filename="{{LOCALAPPDATA}}\Amazon\SSM\Logs\{{EXECUTABLENAME}}.log"
   ```

   Cambie esta entrada para utilizar la siguiente ruta.

   ```
   filename="C:\ProgramData\Amazon\SSM\Logs\{{EXECUTABLENAME}}.log"
   ```

5. Localice la siguiente entrada.

   ```
   filename="{{LOCALAPPDATA}}\Amazon\SSM\Logs\errors.log"
   ```

   Cambie esta entrada para utilizar la siguiente ruta.

   ```
   filename="C:\ProgramData\Amazon\SSM\Logs\errors.log"
   ```

6. Reinicie SSM Agent con el siguiente comando de PowerShell en el modo de administrador.

   ```
   Restart-Service AmazonSSMAgent
   ```

Restricción del acceso a los comandos de nivel raíz con SSM Agent

AWS Systems Manager Agent (SSM Agent) se ejecuta en instancias de Amazon Elastic Compute Cloud (Amazon EC2) y en otros tipos de equipos en entornos híbridos y multinube mediante permisos raíz (Linux) o permisos SYSTEM (Windows Server). Dado que estos son los permisos de acceso al sistema con mayor nivel, toda entidad de confianza a la que se le haya concedido permiso para enviar comandos a SSM Agent tiene permisos raíz o SYSTEM. (En AWS, una entidad de confianza que puede realizar acciones y acceder a los recursos en AWS se

denomina entidad principal. Una entidad principal puede ser un Usuario raíz de la cuenta de AWS, un usuario, o un rol).

Este nivel de acceso es necesario para que una entidad principal pueda enviar comandos autorizados de Systems Manager a SSM Agent, pero también puede permitir que una entidad principal ejecute código malintencionado mediante la explotación de cualquier posible vulnerabilidad de SSM Agent.

En especial, los permisos para ejecutar los comandos SendCommand y StartSession deben restringirse cuidadosamente. Un buen primer paso consiste en conceder permisos para cada comando únicamente a determinadas entidades principales de la organización. Sin embargo, recomendamos aumentar aún más la posición de seguridad restringiendo los nodos administrados en los que una entidad principal puede ejecutar los comandos. Esto se puede hacer en la política de IAM asignada a la entidad principal. Es posible incluir en la política de IAM una condición que solo permita al usuario ejecutar comandos en los nodos administrados que tengan determinadas etiquetas o una combinación de etiquetas.

Por ejemplo, supongamos que tiene dos flotas de servidores, una para pruebas y otra para producción. En la política de IAM que se aplica a los ingenieros menos experimentados, se especifica que puedan ejecutar comandos solo en las instancias que tienen la etiqueta `ssm:resourceTag/testServer`. Sin embargo, a un pequeño grupo de ingenieros con mucha experiencia, que deben tener acceso a todas las instancias, les concede acceso a las instancias que tienen las etiquetas `ssm:resourceTag/testServer` y `ssm:resourceTag/productionServer`.

Con este enfoque, si los ingenieros menos experimentados intentan ejecutar un comando en una instancia de producción, se les denegará el acceso, porque la política de IAM que tienen asignada no proporciona acceso explícito a las instancias que tienen la etiqueta `ssm:resourceTag/productionServer`.

Para obtener más información y ejemplos, consulte los siguientes temas:

- Restricción de acceso de Run Command basado en etiquetas
- Restringir el acceso de sesión en función de las etiquetas de instancia

Automatización de las actualizaciones de SSM Agent

AWS lanza una versión nueva del Agente de AWS Systems Manager (SSM Agent) cuando agregamos o actualizamos las herramientas de Systems Manager. Si los nodos administrados

utilizan una versión anterior del agente, no puede utilizar las nuevas herramientas ni beneficiarse de las herramientas actualizadas. Por este motivo, le recomendamos que automatice el proceso de actualización de SSM Agent en los nodos administrados utilizando uno de los siguientes métodos.

Actualizaciones de agentes en el sistema operativo Bottlerocket

SSM Agent en el sistema operativo Bottlerocket no se puede actualizar mediante el documento de comandos `AWS-UpdateSSMAgent` de Systems Manager. Las actualizaciones se administran en el contenedor de control de Bottlerocket. Para obtener más información, consulte contenedor de control de Bottlerocket y operador de actualización de Bottlerocket en GitHub.

Requisito de versión de macOS

Si una instancia ejecuta la versión 11.0 (Big Sur) o una posterior de macOS, la instancia debe tener la versión 3.1.941.0 o una superior de SSM Agent para ejecutar el documento AWS-UpdateSSMAgent. Si la instancia ejecuta una versión de SSM Agent anterior a la 3.1.941.0, actualice SSM Agent para ejecutar AWS-UpdateSSMAgent al ejecutar los comandos `brew update` y `brew upgrade amazon-ssm-agent`.

Método	Detalles
Actualización automatizada con un solo clic en todos los nodos administrados (recomendado)	Puede configurar todos los nodos administr ados de su Cuenta de AWS para que verifique n y descarguen de forma automática las versiones nuevas de SSM Agent. Para ello, elija Actualización automática de SSM Agent en la pestaña Configuración de Fleet Manager, como se describe más adelante en este tema.
Actualización global o selectiva	Puede utilizar State Manager, una herramien ta de AWS Systems Manager, para crear una asociación que descargue e instale SSM Agent de forma automática en los nodos administr ados. Si desea minimizar la interrupción de las cargas de trabajo, puede crear un periodo de mantenimiento de Systems Manager para realizar la instalación durante los periodos de tiempo designados. Ambos métodos permiten crear una configuración de actualización

Método	Detalles
	global para todos los nodos administrados o elegir de forma selectiva qué instancias se actualizan. Para obtener información sobre la creación de una asociación de State Manager, consulte Tutorial: actualización automática de SSM Agent con AWS CLI. Para obtener más información acerca de cómo crear un periodo de mantenimiento, consulte Tutorial: creación de un periodo de mantenimiento para la aplicación de revisiones mediante la consola.
Actualización global o selectiva para nuevos entornos	Si acaba de empezar a utilizar Systems Manager, le recomendamos que utilice la opción Actualizar Systems Manager (SSM) Agent cada dos semanas en Quick Setup, una herramienta de AWS Systems Manager. Quick Setup le permite crear una configura ción de actualización global para todos los nodos administrados o elegir de forma selectiva qué nodos administrados se actualizan. Para obtener más información, consulte Instalar la administración de host de Amazon EC2 mediante Quick Setup.

Si prefiere actualizar SSM Agent en los nodos administrados de forma manual, puede suscribirse a las notificaciones que publica AWS cuando se lanza una nueva versión del agente. Para obtener más información, consulta Suscripción a las notificaciones de SSM Agent. Después de suscribirse a las notificaciones, puede utilizar Run Command para actualizar de forma manual uno o más nodos con la versión más reciente. Para obtener más información, consulte Actualización de SSM Agent mediante Run Command.

Actualización automática de SSM Agent

Puede configurar Systems Manager para que actualice de forma automática SSM Agent en todos los nodos administrados basados en Linux y en Windows en su Cuenta de AWS. Si habilita esta

opción, Systems Manager verifica de forma automática cada dos semanas si hay una nueva versión del agente. Si hay una nueva versión, Systems Manager actualiza de forma automática el agente a la versión más reciente publicada mediante el documento de SSM `AWS-UpdateSSMAgent`. Le recomendamos que elija esta opción para asegurarse de que los nodos administrados siempre ejecuten la versión más actualizada de SSM Agent.

> **Note**
>
> Si usa un comando `yum` para actualizar SSM Agent en un nodo administrado después de instalar o actualizar el agente mediante el documento de SSM `AWS-UpdateSSMAgent`, es posible que aparezca el siguiente mensaje: `"Warning: RPMDB altered outside of yum."` Este mensaje está dentro de lo esperado y puede ser ignorado de forma segura.

Para actualizar automáticamente SSM Agent

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Fleet Manager.
3. Elija la pestaña Settings.
4. En el área Actualización automática del agente, elija Actualización automática de SSM Agent.

Para cambiar la versión de SSM Agent a la que se actualiza la flota, elija Edit (Editar) en la opción Agent auto update (Actualización automática del agente) de la pestaña Settings (Configuración). A continuación, ingrese el número de versión de SSM Agent a la que desea actualizarse en la opción de Version (Versión) en Parameters (Parámetros). Si no se especifica, el agente se actualiza a la versión más reciente.

Para detener la implementación automática de versiones actualizadas de SSM Agent en todos los nodos administrados de la cuenta, elija Delete (Eliminar) en la opción Agent auto update (Actualización automática del agente) de la pestaña Settings (Configuración). Esta acción elimina la asociación de State Manager que actualiza automáticamente SSM Agent en los nodos administrados.

Suscripción a las notificaciones de SSM Agent

Amazon Simple Notification Service (Amazon SNS) puede notificarle cuando se lancen las versiones nuevas de AWS Systems Manager Agent (SSM Agent). Para suscribirse a estas notificaciones, utilice el siguiente procedimiento.

> **Tip**
>
> También puede suscribirse a las notificaciones mediante la página SSM Agent Release Notes en GitHub.

Para suscribirse a las notificaciones de SSM Agent

1. Abra la consola de Amazon SNS en https://console.aws.amazon.com/sns/v3/home.
2. En el selector de región de la barra de navegación, elija US East (N. Virginia) (EE. UU. Este [Norte de Virginia]), si todavía no está seleccionada. Debe seleccionar esta Región de AWS porque las notificaciones de Amazon SNS para SSM Agent a las que se va a suscribir solo se generan desde esa región.
3. En el panel de navegación, seleccione Subscriptions.
4. Seleccione Create subscription.
5. En Create subscription (Crear suscripción), haga lo siguiente:
   a. En Topic ARN, use el siguiente nombre de recurso de Amazon (ARN):

      ```
      arn:aws:sns:us-east-1:720620558202:SSM-Agent-Update
      ```

   b. En Protocolo, elija `Email` o `SMS`.
   c. En Endpoint (Punto de conexión), ingrese una dirección de email o bien un código de área y un número para recibir notificaciones, en función de si ha elegido `Email` o `SMS` en el paso anterior.
   d. Seleccione Crear una suscripción.
6. Si elige `Email`, recibirá un email que le pedirá que confirme la suscripción. Abra el mensaje y siga las instrucciones para completar la suscripción.

Cada vez que se publique una nueva versión del SSM Agent, enviaremos una notificación a los suscriptores. Si ya no desea recibir estas notificaciones, utilice el siguiente procedimiento para cancelar la suscripción.

Para cancelar la suscripción a las notificaciones de SSM Agent

1. Abra la consola de Amazon SNS.
2. En el panel de navegación, seleccione Subscriptions.
3. Seleccione la suscripción, y luego elija Delete (Eliminar). Cuando se le pida confirmación, elija Delete (Eliminar).

Solución de problemas de SSM Agent

Si tiene problemas a la hora de ejecutar operaciones en los nodos administrados, es posible que haya algún problema con AWS Systems Manager Agent (SSM Agent). Utilice la siguiente información como ayuda para ver los archivos de registro de SSM Agent y solucionar los problemas del agente.

Temas

- SSM Agent está desactualizado
- Solucionar problemas con los archivos de registro de SSM Agent
- Los archivos de registros del agente no rotan (Windows)
- No es posible conectarse a los puntos de enlace de SSM
- Verificación de la configuración VPC
- Compruebe los atributos relacionados con el DNS de la VPC
- Verificación de las reglas de entrada en los grupos de seguridad de punto de conexión
- Utilice ssm-cli para solucionar los problemas de disponibilidad de los nodos administrados

SSM Agent está desactualizado

Cada vez que se agregan herramientas nuevas a Systems Manager o se actualizan las herramientas existentes, se lanza una versión actualizada de SSM Agent. No utilizar la versión más reciente del agente puede impedir que el nodo administrado utilice diversas herramientas y características de Systems Manager. Por este motivo, se recomienda automatizar el proceso de mantener SSM Agent actualizado en los equipos. Para obtener más información, consulta Automatización de las

actualizaciones de SSM Agent. Suscríbase a la página SSM Agent Release Notes en GitHub para recibir notificaciones sobre las actualizaciones de SSM Agent.

Solucionar problemas con los archivos de registro de SSM Agent

El SSM Agent registra información en los siguientes archivos. La información en estos archivos también puede ayudarlo a solucionar los problemas. Para obtener más información acerca de los archivos de registros de SSM Agent, e incluso acerca de cómo activar el registro de depuración, consulte Visualización de registros de SSM Agent.

> **Note**
>
> Si elige ver estos registros mediante el explorador de archivos de Windows, asegúrese de permitir la visualización de archivos ocultos y archivos del sistema en “Folder Options” (Opciones de carpeta).

En Windows

- `%PROGRAMDATA%\Amazon\SSM\Logs\amazon-ssm-agent.log`
- `%PROGRAMDATA%\Amazon\SSM\Logs\errors.log`

En Linux y macOS

- `/var/log/amazon/ssm/amazon-ssm-agent.log`
- `/var/log/amazon/ssm/errors.log`

Para los nodos administrados de Linux, puede encontrar más información en el archivo `messages` escrito en el siguiente directorio: `/var/log`.

Para obtener información adicional sobre la solución de problemas con los registros de los agentes, consulte ¿Cómo puedo utilizar los registros de SSM Agent para solucionar los problemas con SSM Agent en mi instancia administrada? en el Centro de conocimientos re:Post de AWS.

Los archivos de registros del agente no rotan (Windows)

Si especifica la rotación del archivo de registro basada en la fecha en el archivo seelog.xml (en los nodos administrados de Windows Server) y los registros no rotan, especifique el parámetro

`fullname=true`. A continuación, se muestra un ejemplo de un archivo de configuración seelog.xml con el parámetro `fullname=true` especificado.

```
<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>
```

No es posible conectarse a los puntos de enlace de SSM

SSM Agent debe permitir el tráfico saliente HTTPS (puerto 443) a los siguientes puntos de conexión:

- `ssm.`*`region`*`.amazonaws.com`
- `ssmmessages.`*`region`*`.amazonaws.com`

`region` representa el identificador de Región de AWS compatible con AWS Systems Manager, como `us-east-2` para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *`regiones`* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

> **Note**
>
> Antes de 2024, `ec2messages.region.amazonaws.com` también era obligatorio.
> En el caso de las Regiones de AWS lanzadas antes de 2024, sigue siendo obligatorio permitir el tráfico a `ssmmessages.region.amazonaws.com`, pero a `ec2messages.region.amazonaws.com` es opcional.
> En el caso de las regiones lanzadas a partir de 2024, es obligatorio permitir el tráfico `ssmmessages.region.amazonaws.com`, pero estas regiones no admiten los puntos de conexión de `ec2messages.region.amazonaws.com`.

SSM Agent no funcionará si no puede comunicarse con los puntos de conexión anteriores, tal y como se ha descrito, incluso si usa las Amazon Machine Images (AMIs) proporcionadas por AWS, como Amazon Linux 2 o Amazon Linux 2023. La configuración de red debe tener acceso abierto a Internet o debe tener configurados los puntos de enlace personalizados de la nube privada virtual (VPC). Si no tiene previsto crear un punto de enlace de la VPC personalizado, verifique las gateways de Internet o las gateways NAT. Para obtener más información acerca de cómo administrar los puntos de enlace de la VPC, consulte Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager.

Verificación de la configuración VPC

Si está utilizando una nube privada virtual (VPC), para administrar las instancias de EC2 con Systems Manager los puntos de conexión de VPC deben estar configurados correctamente para `ssm.region.amazonaws.com`, `ssmmessages.region.amazonaws.com`, y en algunos casos explicados anteriormente en este tema en No es posible conectarse a los puntos de enlace de SSM, `ec2messages.region.amazonaws.com`

> **Note**
>
> La alternativa a usar un punto de enlace de la VPC es permitir el acceso a Internet saliente en las instancias administradas. En este caso, las instancias administradas también deben permitir el tráfico saliente HTTPS (puerto 443) a los siguientes puntos de enlace:
>
> - `ssm.region.amazonaws.com`
> - `ssmmessages.region.amazonaws.com`
> - `ec2messages.region.amazonaws.com`

SSM Agent inicia todas las conexiones al servicio de Systems Manager en la nube. Por este motivo, no es necesario configurar el firewall para permitir el tráfico entrante a las instancias de Systems Manager.
Para obtener más información acerca de los puntos de enlace, consulte Referencia: ec2messages, ssmmessages y otras operaciones de la API.

Para solucionar los problemas con los puntos de conexión de VPC, realice lo siguiente:

- Asegúrese de que los puntos de conexión de VPC estén incluidos en el nivel de VPC. Si el punto de conexión de VPC con un nombre de servicio específico no se encuentra en la VPC, compruebe primero que la compatibilidad con DNS esté habilitada en el nivel de la VPC. A continuación, cree un nuevo punto de conexión de VPC y asócielo con una subred en cada zona de disponibilidad.
- Asegúrese de que un nombre de DNS privado esté habilitado en el nivel de punto de conexión de VPC. Los nombres de DNS privados están habilitados de forma predeterminada, pero es posible que se hayan deshabilitado manualmente en algún punto.
- Asegúrese de que los puntos de conexión de VPC existentes estén asociados a la subred adecuada. Además, asegúrese de que VPCE ya esté asociado a una subred en esa zona de disponibilidad.

Para obtener más información, consulte los temas siguientes:

- Acceda a un Servicio de AWS mediante un punto de conexión de VPC de interfaz en la Guía AWS PrivateLink
- Asocie un nombre de DNS privado en la Guía de AWS PrivateLink
- Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager

Compruebe los atributos relacionados con el DNS de la VPC

Como parte de la verificación de la configuración de la nube privada virtual (VPC), asegúrese de que los atributos `enableDnsSupport` y `enableDnsHostnames` estén activados.

Puede habilitar estos atributos mediante la acción de la API ModifyVpcAttribute de Amazon EC2 o el comando AWS CLI modify-vpc-attribute.

Para obtener más información sobre la habilitación de estos atributos en la consola de Amazon VPC, consulte Ver y actualizar los atributos de DNS de su VPC en la Guía del usuario de Amazon VPC.

> **Note**
>
> La alternativa a usar un punto de enlace de la VPC es permitir el acceso a Internet saliente en las instancias administradas. En este caso, las instancias administradas también deben permitir el tráfico saliente HTTPS (puerto 443) a los siguientes puntos de enlace:
>
> - `ssm.region.amazonaws.com`
> - `ssmmessages.region.amazonaws.com`
> - `ec2messages.region.amazonaws.com`
>
> SSM Agent inicia todas las conexiones al servicio de Systems Manager en la nube. Por este motivo, no es necesario configurar el firewall para permitir el tráfico entrante a las instancias de Systems Manager.
> Para obtener más información acerca de los puntos de enlace, consulte Referencia: ec2messages, ssmmessages y otras operaciones de la API.

Verificación de las reglas de entrada en los grupos de seguridad de punto de conexión

Asegúrese de que todos los puntos de conexión de VPC que haya configurado (`ssm`, `ssmmessages` y `ec2messages`) incluyan una regla de entrada en sus grupos de seguridad para permitir que el tráfico entre por el puerto 443. Si es necesario, puede crear un nuevo grupo de seguridad en la VPC con una regla de entrada para permitir el tráfico en el puerto 443 para el bloque de enrutamiento entre dominios sin clases (CIDR) de la VPC. Después de crear el grupo de seguridad, adjúntelo a cada punto de conexión de VPC.

Para obtener más información, consulte los temas siguientes:

- ¿Cómo creo puntos de conexión de VPC para utilizar Systems Manager a fin de administrar instancias de EC2 privadas sin acceso a Internet? en AWS re:Post
- Bloques CIDR de VPC en la Guía del usuario de Amazon VPC

Utilice `ssm-cli` para solucionar los problemas de disponibilidad de los nodos administrados

A partir de la versión 3.1.501.0 de SSM Agent, se puede utilizar `ssm-cli` para determinar si un nodo administrado cumple los requisitos principales para que lo administre Systems Manager y para que aparezca en las listas de nodos administrados en Fleet Manager. La `ssm-cli` es una herramienta de la línea de comandos independiente incluida en la instalación de SSM Agent. Se encuentran incluidos comandos preconfigurados que recopilan información para ayudarlo a diagnosticar por qué una instancia de Amazon EC2 o una máquina que no es de EC2 que ha confirmado que se está ejecutando no se incluye en las listas de nodos administrados de Systems Manager. Estos comandos se ejecutan cuando se especifica la opción `get-diagnostics`.

Para obtener más información, consulte Solución de problemas de disponibilidad de nodos administrados mediante `ssm-cli`.

Seguridad en AWS Systems Manager

La seguridad en la nube en Amazon Web Services es la máxima prioridad. Como cliente de AWS, se beneficiará de una arquitectura de red y un centro de datos que están diseñados para satisfacer los requisitos de seguridad de las organizaciones más exigentes.

La seguridad es una responsabilidad compartida entre AWS y usted. El modelo de responsabilidad compartida la describe como seguridad de la nube y seguridad en la nube:

- Seguridad de la nube: AWS es responsable de proteger la infraestructura que ejecuta Servicios de AWS en Nube de AWS. Además, AWS proporciona servicios que puede utilizar de forma segura. Los auditores externos prueban y verifican periódicamente la eficacia de nuestra seguridad como parte de los Programas de conformidad de AWS. Para obtener información sobre los programas de conformidad que se aplican a AWS Systems Manager, consulte Servicios de AWS en el ámbito del programa de conformidad.
- Seguridad en la nube: su responsabilidad viene determinada por el Servicio de AWS que utilice. También es responsable de otros factores, incluida la confidencialidad de los datos, los requisitos de la empresa y la legislación y los reglamentos aplicables.

Esta documentación le ayuda a comprender cómo aplicar el modelo de responsabilidad compartida a la hora de utilizar AWS Systems Manager. Los siguientes temas le mostrarán cómo configurar Systems Manager para satisfacer sus objetivos de seguridad y de conformidad. También aprenderá a utilizar otros Servicios de AWS que le ayudarán a monitorizar y a proteger los recursos de Systems Manager.

Temas

- Protección de los datos en AWS Systems Manager
- Administración de identidades y accesos en AWS Systems Manager
- Cómo utilizar roles vinculados a servicios de Systems Manager
- Registro y monitorización en AWS Systems Manager
- Validación de conformidad en AWS Systems Manager
- Resiliencia en AWS Systems Manager
- Seguridad de la infraestructura en AWS Systems Manager
- Configuración y análisis de vulnerabilidades en AWS Systems Manager

- Prácticas recomendadas de seguridad para Systems Manager

Protección de los datos en AWS Systems Manager

La protección de datos consiste en proteger los datos mientras están en tránsito (cuando viajan a Systems Manager y desde este) y en reposo (mientras están almacenados en centros de datos de AWS).

El modelo de responsabilidad compartida, y de AWS se aplica a la protección de datos de AWS Systems Manager. Como se describe en este modelo, AWS es responsable de proteger la infraestructura global que ejecuta toda la Nube de AWS. Eres responsable de mantener el control sobre el contenido alojado en esta infraestructura. También eres responsable de las tareas de administración y configuración de seguridad para los Servicios de AWS que utiliza. Para obtener más información sobre la privacidad de los datos, consulta las Preguntas frecuentes sobre la privacidad de datos. Para obtener información sobre la protección de datos en Europa, consulta la publicación de blog sobre el Modelo de responsabilidad compartida de AWS y GDPR en el Blog de seguridad de AWS.

Con fines de protección de datos, recomendamos proteger las credenciales de la Cuenta de AWS y configurar cuentas de usuario individuales con AWS IAM Identity Center o AWS Identity and Access Management (IAM). De esta manera, solo se otorgan a cada usuario los permisos necesarios para cumplir sus obligaciones laborales. También recomendamos proteger sus datos de la siguiente manera:

- Utiliza la autenticación multifactor (MFA) en cada cuenta.
- Utiliza SSL/TLS para comunicarse con los recursos de AWS. Se recomienda el uso de TLS 1.2 y recomendamos TLS 1.3.
- Configure los registros de API y de actividad de los usuarios con AWS CloudTrail. Para obtener información sobre cómo utilizar registros de seguimiento de CloudTrail para capturar actividades de AWS, consulta Working with CloudTrail trails en la Guía del usuario de AWS CloudTrail.
- Utiliza las soluciones de cifrado de AWS, junto con todos los controles de seguridad predeterminados dentro de los servicios de Servicios de AWS.
- Utiliza servicios de seguridad administrados avanzados, como Amazon Macie, que lo ayuden a detectar y proteger los datos confidenciales almacenados en Amazon S3.
- Si necesita módulos criptográficos validados FIPS 140-3 al acceder a AWS a través de una interfaz de línea de comandos o una API, utiliza un punto de conexión de FIPS. Para obtener

más información sobre los puntos de conexión de FIPS disponibles, consulta Estándar de procesamiento de la información federal (FIPS) 140-3.

Se recomienda encarecidamente no introducir nunca información confidencial o sensible, como por ejemplo, direcciones de correo electrónico de clientes, en etiquetas o campos de formato libre, tales como el campo Nombre. Esto incluye las situaciones en las que debe trabajar con la Systems Manager u otros Servicios de AWS a través de la consola, la API, la AWS CLI o los SDK de AWS. Cualquier dato que ingrese en etiquetas o campos de texto de formato libre utilizados para nombres se puede emplear para los registros de facturación o diagnóstico. Si proporciona una URL a un servidor externo, recomendamos encarecidamente que no incluya información de credenciales en la URL a fin de validar la solicitud para ese servidor.

Cifrado de datos

Cifrado en reposo

Parámetros Parameter Store

Los tipos de parámetros que puede crear en Parameter Store, una herramienta de AWS Systems Manager, incluyen `String`, `StringList` y `SecureString`.

Todos los parámetros, independientemente de su tipo, se cifran en tránsito y en reposo. En tránsito, los parámetros se cifran mediante la seguridad de la capa de transporte (TLS) para crear una conexión HTTPS segura para las solicitudes de API. En reposo, se cifran con una Clave propiedad de AWS en AWS Key Management Service (AWS KMS). Para obtener más información sobre Clave propiedad de AWS el cifrado, consulte Claves propiedad de AWS en la AWS Key Management Service Guía para desarrolladores.

El tipo `SecureString` ofrece opciones de cifrado adicionales y se recomienda para todos los datos confidenciales. Puede elegir entre los siguientes tipos de claves AWS KMS para cifrar y descifrar el valor de un parámetro `SecureString`:

- La Clave administrada de AWS de su cuenta
- Una clave administrada por el cliente (CMK) que ha creado en su cuenta
- Una CMK en otra Cuenta de AWS que se ha compartido con usted

Para obtener más información sobre AWS KMS el cifrado, consulte AWS Key Management Service Guía para desarrolladores.

Contenido en buckets de S3

En las operaciones de Systems Manager, puede elegir cargar o almacenar datos en uno o más buckets de Amazon Simple Storage Service (Amazon S3).

Para obtener información acerca del cifrado de buckets de S3, consulte Protecting data using encryption y Protección de datos en Amazon S3 en la Guía del usuario para Amazon Simple Storage Service.

A continuación se indican los tipos de datos que puede cargar o almacenar en buckets de S3 en las actividades de Systems Manager:

- La salida de los comandos en Run Command, una herramienta de AWS Systems Manager
- Paquetes en Distributor, una herramienta de AWS Systems Manager
- Registros de operaciones de revisiones en Patch Manager, una herramienta de AWS Systems Manager
- listas de anulación de revisiones de Patch Manager
- Scripts o manuales de estrategias de Ansible para ejecutarlos en el flujo de trabajo de un manual de procedimientos de Automatización, una herramienta de AWS Systems Manager
- Perfiles de Chef InSpec para su uso con análisis de Cumplimiento, una herramienta de AWS Systems Manager
- Registros de AWS CloudTrail
- Registros de historial de sesiones en Session Manager, una herramienta de AWS Systems Manager
- Informes de Explorer, una herramienta de AWS Systems Manager
- OpsData de OpsCenter, una herramienta de AWS Systems Manager
- AWS CloudFormation plantillas para usar con flujos de trabajo de Automation
- datos de conformidad de análisis de sincronización de datos de recursos
- Resultado de solicitudes para crear o editar una asociación en State Manager, una herramienta de AWS Systems Manager, en nodos administrados
- Documentos personalizados de Systems Manager (documentos de SSM) que puede ejecutar con el documento de SSM administrado de AWS `AWS-RunDocument`

Grupos de registros de CloudWatch Logs

En las operaciones de Systems Manager, puede elegir transmitir datos a uno o más grupos de registros de Amazon CloudWatch.

Para obtener información acerca del cifrado de un grupo de registros de los Registros de CloudWatch, consulte Cifrado de datos de registro en CloudWatch Logs mediante AWS Key Management Service en la Guía del usuario de los Registros de Amazon CloudWatch.

Los siguientes son los tipos de datos que puede haber transmitido a un grupo de registros de CloudWatch Logs en sus actividades de Systems Manager:

- Resultado de los comandos de Run Command
- resultado de los scripts que se ejecutan mediante la acción `aws:executeScript` de un manual de procedimientos de automatización
- Registros del historial de sesiones de Session Manager
- Registros de SSM Agent en los nodos administrados

Cifrado en tránsito

Recomendamos utilizar un protocolo de cifrado como Transport Layer Security (TLS) para cifrar la información confidencial en tránsito entre los clientes y los nodos.

Systems Manager proporciona la siguiente compatibilidad para el cifrado de los datos en tránsito.

Conexiones a los puntos de enlace de API de Systems Manager

> Los puntos de enlace de API de Systems Manager solo admiten conexiones seguras a través de HTTPS. Cuando administra recursos de Systems Manager con la AWS Management Console, el AWS SDK o la API de Systems Manager, todas las comunicaciones se cifran con Transport Layer Security (TLS). Para obtener una lista completa de puntos de conexión de la API, consulte los Puntos de conexión de Servicio de AWS en la Referencia general de Amazon Web Services.

Instancias administradas

> AWS proporciona conectividad segura y privada entre instancias de Amazon Elastic Compute Cloud (Amazon EC2). Además, ciframos automáticamente el tráfico en tránsito entre instancias admitidas en la misma nube virtual privada (VPC) o en VPC interconectadas, mediante algoritmos AEAD con cifrado de 256 bits. Esta característica de cifrado utiliza las capacidades de descarga del hardware subyacente y no afecta al rendimiento de red. Las instancias soportadas son: c5n, G4, I3en, M5dn, M5n, P3dn, R5dn y R5n.

Sesiones de Session Manager

De forma predeterminada, Session Manager utiliza TLS 1.3 para cifrar los datos de sesión transmitidos entre las máquinas locales de los usuarios de la cuenta y las instancias EC2. También puede optar por cifrar aún más los datos en tránsito mediante una AWS KMS key que se ha creado en AWS KMS. El cifrado AWS KMS está disponible para tipos de sesiones `Standard_Stream`, `InteractiveCommands` y `NonInteractiveCommands`.

Acceso a Run Command

De forma predeterminada, el acceso remoto a los nodos mediante Run Command se cifra con TLS 1.3 y las solicitudes para crear una conexión se firman con SigV4.

Privacidad del tráfico entre redes

Puede utilizar Amazon Virtual Private Cloud (Amazon VPC) para crear límites entre los recursos en los nodos administrados y controlar el tráfico entre ellos, la red local e Internet. Para obtener detalles, consulte Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager.

Para obtener más información sobre la seguridad de Amazon Virtual Private Cloud, consulte Internetwork traffic privacy in Amazon VPC en la Guía del usuario de Amazon VPC.

Administración de identidades y accesos en AWS Systems Manager

AWS Identity and Access Management (IAM) es un Servicio de AWS que ayuda a los administradores a controlar de forma segura el acceso a los recursos de AWS. Los administradores de IAM controlan quién está autenticado (ha iniciado sesión) y autorizado (tiene permisos) para utilizar recursos de Systems Manager. IAM es un Servicio de AWS que se puede utilizar sin cargo adicional.

Temas

- Público
- Autenticación con identidades
- Administración de acceso mediante políticas
- Cómo funciona AWS Systems Manager con IAM

- AWS Systems Manager ejemplos de políticas basadas en identidad de
- Políticas administradas de AWS para AWS Systems Manager
- Solución de problemas de identidades de AWS Systems Manager y accesos

Público

La forma en que utilice AWS Identity and Access Management (IAM) difiere en función del trabajo que realice en Systems Manager.

Usuario de servicio: si utiliza el servicio de Systems Manager para realizar su trabajo, su administrador le proporciona las credenciales y los permisos que necesita. A medida que utilice más características de Systems Manager para realizar su trabajo, es posible que necesite permisos adicionales. Entender cómo se administra el acceso puede ayudarle a solicitar los permisos correctos al administrador. Si no puede acceder a una característica en Systems Manager, consulte Solución de problemas de identidades de AWS Systems Manager y accesos.

Administrador de servicio: si está a cargo de los recursos de Systems Manager en su empresa, probablemente tenga acceso completo a Systems Manager. Su trabajo consiste en determinar a qué características y recursos de Systems Manager deben acceder los usuarios del servicio. A continuación, debe enviar solicitudes a su administrador de IAM para cambiar los permisos de los usuarios de sus servicios. Revise la información de esta página para conocer los conceptos básicos de IAM. Para obtener más información sobre cómo su empresa puede utilizar IAM con Systems Manager, consulte Cómo funciona AWS Systems Manager con IAM.

Administrador de IAM: si es un administrador de IAM, es posible que quiera conocer más detalles sobre cómo escribir políticas para administrar el acceso a Systems Manager. Para consultar ejemplos de políticas basadas en identidad de Systems Manager que puede utilizar en IAM, consulte AWS Systems Manager ejemplos de políticas basadas en identidad de .

Autenticación con identidades

La autenticación es la manera de iniciar sesión en AWS mediante credenciales de identidad. Debe estar autenticado (haber iniciado sesión en AWS) como Usuario raíz de la cuenta de AWS, como un usuario de IAM o asumiendo un rol de IAM.

Puedes iniciar sesión en AWS como una identidad federada mediante las credenciales proporcionadas a través de una fuente de identidad. AWS IAM Identity Center Los usuarios (del

IAM Identity Center), la autenticación de inicio de sesión único de su empresa y sus credenciales de Google o Facebook son ejemplos de identidades federadas. Al iniciar sesión como una identidad federada, su gestionador habrá configurado previamente la federación de identidades mediante roles de IAM. Cuando accede a AWS mediante la federación, está asumiendo un rol de forma indirecta.

Según el tipo de usuario que sea, puedes iniciar sesión en AWS Management Console o en el portal de acceso AWS. Para obtener más información sobre el inicio de sesión en AWS, consulta Cómo iniciar sesión en su Cuenta de AWS en la Guía del usuario de AWS Sign-In.

Si accede a AWS mediante programación, AWS proporciona un kit de desarrollo de software (SDK) y una interfaz de la línea de comandos (CLI) para firmar criptográficamente las solicitudes mediante el uso de las credenciales. Si no usa las herramientas de AWS, debe firmar las solicitudes. Para obtener más información sobre la firma de solicitudes, consulte AWSSignature Versión 4 para solicitudes API en la Guía del usuario de IAM.

Independientemente del método de autenticación que use, es posible que deba proporcionar información de seguridad adicional. Por ejemplo, AWS le recomienda el uso de la autenticación multifactor (MFA) para aumentar la seguridad de su cuenta. Para obtener más información, consulte Autenticación multifactor en la Guía del usuario de AWS IAM Identity Center y Autenticación multifactor AWSen IAM en la Guía del usuario de IAM.

Usuario raíz de Cuenta de AWS

Cuando se crea una Cuenta de AWS, se comienza con una identidad de inicio de sesión que tiene acceso completo a todos los recursos y Servicios de AWS de la cuenta. Esta identidad recibe el nombre de usuario raíz de la Cuenta de AWS y se accede a ella iniciando sesión con el email y la contraseña que utilizó para crear la cuenta. Recomendamos encarecidamente que no utiliza el usuario raíz para sus tareas diarias. Proteja las credenciales del usuario raíz y utilícelas solo para las tareas que solo el usuario raíz pueda realizar. Para ver la lista completa de las tareas que requieren que inicie sesión como usuario raíz, consulte Tareas que requieren credenciales de usuario raíz en la Guía del usuario de IAM.

Usuarios y grupos de IAM

Un usuario de IAM es una identidad de la Cuenta de AWSque dispone de permisos específicos para una sola persona o aplicación. Siempre que sea posible, recomendamos emplear credenciales temporales, en lugar de crear usuarios de IAM que tengan credenciales de larga duración como contraseñas y claves de acceso. No obstante, si tiene casos de uso específicos que requieran credenciales de larga duración con usuarios de IAM, recomendamos rotar las claves de acceso.

Para más información, consulte Rotar las claves de acceso periódicamente para casos de uso que requieran credenciales de larga duración en la Guía del usuario de IAM.

Un grupo de IAM es una identidad que especifica un conjunto de usuarios de IAM. No puedes iniciar sesión como grupo. Puedes usar los grupos para especificar permisos para varios usuarios a la vez. Los grupos facilitan la administración de los permisos de grandes conjuntos de usuarios. Por ejemplo, podría tener un grupo cuyo nombre fuese IAMAdmins y conceder permisos a dicho grupo para administrar los recursos de IAM.

Los usuarios son diferentes de los roles. Un usuario se asocia exclusivamente a una persona o aplicación, pero la intención es que cualquier usuario pueda asumir un rol que necesite. Los usuarios tienen credenciales de larga duración permanentes; no obstante, los roles proporcionan credenciales temporales. Para obtener más información, consulte Casos de uso para usuarios de IAM en la Guía del usuario de IAM.

Roles de IAM

Un rol de IAM es una identidad de la Cuenta de AWS que dispone de permisos específicos. Es similar a un usuario de IAM, pero no está asociado a una persona determinada. Para asumir temporalmente un rol de IAM en la AWS Management Console, puede cambiar de una función de usuario a un rol de IAM (consola). Puedes asumir un rol llamando a una operación de la AWS CLI o de la API de AWS, o utilizando una URL personalizada. Para más información sobre los métodos para el uso de roles, consulta Métodos para asumir un rol en la Guía del usuario de IAM.

Los roles de IAM con credenciales temporales son útiles en las siguientes situaciones:

- Acceso de usuario federado: para asignar permisos a una identidad federada, puede crear un rol y definir sus permisos. Cuando se autentica una identidad federada, se asocia la identidad al rol y se le conceden los permisos define el rol. Para obtener información acerca de roles de federación, consulte Crear un rol para un proveedor de identidad de terceros (federación) en la Guía de usuario de IAM. Si utiliza el IAM Identity Center, debe configurar un conjunto de permisos. IAM Identity Center correlaciona el conjunto de permisos con un rol en IAM para controlar a qué puedes acceder las identidades después de autenticarse. Para obtener información acerca de los conjuntos de permisos, consulta Conjuntos de permisos en la Guía del usuario de AWS IAM Identity Center.
- Permisos de usuario de IAM temporales: un usuario de IAM puedes asumir un rol de IAM para recibir temporalmente permisos distintos que le permitan realizar una tarea concreta.
- Acceso entre cuentas: puede utilizar un rol de IAM para permitir que alguien (una entidad principal de confianza) de otra cuenta acceda a los recursos de la cuenta. Los roles son la forma principal

de conceder acceso entre cuentas. No obstante, con algunos Servicios de AWS se puedes adjuntar una política directamente a un recurso (en lugar de utilizar un rol como representante). Para obtener información acerca de la diferencia entre los roles y las políticas basadas en recursos para el acceso entre cuentas, consulta Acceso a recursos entre cuentas en IAM en la Guía del usuario de IAM.

- Acceso entre servicios: algunos Servicios de AWS utilizan características de otros Servicios de AWS. Por ejemplo, cuando realiza una llamada en un servicio, es común que ese servicio ejecute aplicaciones en Amazon EC2 o almacene objetos en Amazon S3. Es posible que un servicio haga esto usando los permisos de la entidad principal, usando un rol de servicio o usando un rol vinculado a servicios.
  - Reenviar sesiones de acceso (FAS): cuando utiliza un rol o un usuario de IAM para llevar a cabo las acciones en AWS, se le considera una entidad principal. Cuando utiliza algunos servicios, es posible que realice una acción que desencadene otra acción en un servicio diferente. FAS utiliza los permisos de la entidad principal para llamar a un Servicio de AWS, combinados con el Servicio de AWS solicitante para realizar solicitudes a servicios posteriores. Las solicitudes de FAS solo se realizan cuando un servicio recibe una solicitud que requiere interacciones con otros Servicios de AWS o recursos para completarse. En este caso, debe tener permisos para realizar ambas acciones. Para obtener información sobre las políticas a la hora de realizar solicitudes de FAS, consulte Reenviar sesiones de acceso.
  - Rol de servicio: un rol de servicio es un rol de IAM que adopta un servicio para realizar acciones en su nombre. Un administrador de IAM puede crear, modificar y eliminar un rol de servicio desde IAM. Para obtener más información, consulte Creación de un rol para delegar permisos a un Servicio de AWS en la Guía del usuario de IAM.
  - Rol vinculado a los servicios: un rol vinculado a servicios es un tipo de rol de servicio que está vinculado a un Servicio de AWS. El servicio puedes asumir el rol para realizar una acción en su nombre. Los roles vinculados a servicios aparecen en la Cuenta de AWS y son propiedad del servicio. Un administrador de IAM puedes ver, pero no editar, los permisos de los roles vinculados a servicios.
- Aplicaciones que se ejecutan en Amazon EC2: puedes utilizar un rol de IAM que le permita administrar credenciales temporales para las aplicaciones que se ejecutan en una instancia de EC2 y realizan solicitudes a la AWS CLI o a la API de AWS. Es preferible hacerlo de este modo a almacenar claves de acceso en la instancia de EC2. Para asignar un rol de AWS a una instancia de EC2 y ponerla a disposición de todas las aplicaciones, cree un perfil de instancia adjuntado a la instancia. Un perfil de instancia contiene el rol y permite a los programas que se ejecutan en la instancia de EC2 obtener credenciales temporales. Para obtener más información, consulte Uso de

un rol de IAM para conceder permisos a aplicaciones que se ejecutan en instancias Amazon EC2 en la Guía del usuario de IAM.

Administración de acceso mediante políticas

Para controlar el acceso en AWS, se crean políticas y se adjuntan a identidades o recursos de AWS. Una política es un objeto de AWS que, cuando se asocia a una identidad o un recurso, define sus permisos. AWS evalúa estas políticas cuando una entidad principal (sesión de rol, usuario o usuario raíz) realiza una solicitud. Los permisos en las políticas determinan si la solicitud se permite o se deniega. La mayoría de las políticas se almacenan en AWS como documentos JSON. Para obtener más información sobre la estructura y el contenido de los documentos de política JSON, consulte Información general de políticas JSON en la Guía del usuario de IAM.

Los administradores puede utilizar las políticas JSON de AWS para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

De forma predeterminada, los usuarios y los roles no tienen permisos. Un administrador de IAM puede crear políticas de IAM para conceder permisos a los usuarios para realizar acciones en los recursos que necesitan. A continuación, el administrador puede agregar las políticas de IAM a roles y los usuarios puedes asumirlos.

Las políticas de IAM definen permisos para una acción independientemente del método que se utiliza para realizar la operación. Por ejemplo, suponga que dispone de una política que permite la acción `iam:GetRole`. Un usuario con dicha política puedes obtener información del usuario de la AWS Management Console, la AWS CLI o la API de AWS.

Políticas basadas en identidades

Las políticas basadas en identidad son documentos de políticas de permisos JSON que puede asociar a una identidad, como un usuario de IAM, un grupo de usuarios o un rol. Estas políticas controlan qué acciones pueden realizar los usuarios y los roles, en qué recursos y en qué condiciones. Para obtener más información sobre cómo crear una política basada en identidad, consulte Creación de políticas de IAM en la Guía del usuario de IAM.

Las políticas basadas en identidades puedes clasificarse además como políticas insertadas o políticas administradas. Las políticas insertadas se integran directamente en un único usuario, grupo o rol. Las políticas administradas son políticas independientes que puedes adjuntar a varios usuarios, grupos y roles de su Cuenta de AWS. Las políticas gestionadas incluyen las políticas gestionadas de AWS y las políticas gestionadas por el cliente. Para obtener más información sobre cómo elegir una

política administrada o una política insertada, consulte Elegir entre políticas administradas y políticas insertadas en la Guía del usuario de IAM.

Para obtener información acerca de las políticas administradas de AWS para Systems Manager, consulte Políticas gestionadas de AWS Systems Manager.

Políticas basadas en recursos

Las políticas basadas en recursos son documentos de política JSON que se asocian a un recurso. Los ejemplos de políticas basadas en recursos son las políticas de confianza de roles de IAM y las políticas de bucket de Amazon S3. En los servicios que admiten políticas basadas en recursos, los administradores de servicios puede utilizarlos para controlar el acceso a un recurso específico. Para el recurso al que se asocia la política, la política define qué acciones puede realizar una entidad principal especificada en ese recurso y en qué condiciones. Debe especificar una entidad principal en una política en función de recursos. Las entidades principales puedes incluir cuentas, usuarios, roles, usuarios federados o Servicios de AWS.

Las políticas basadas en recursos son políticas insertadas que se encuentran en ese servicio. No se puedes utilizar políticas de IAM administradas de AWS en una política basada en recursos.

Claves de condición de políticas

Las acciones que los usuarios y los roles pueden llevar a cabo y los recursos con los que pueden efectuar esas acciones pueden restringirse aún más mediante condiciones específicas.

En documentos de políticas JSON, el elemento `Condition` (o bloque de `Condition`) permite especificar condiciones en las que entra en vigor una instrucción. El elemento `Condition` es opcional. Puede crear expresiones condicionales que utilizan operadores de condición, como `StringEquals` o `StringNotLike`, para que la condición de la política coincida con los valores de la solicitud.

Si especifica varios elementos de `Condition` en una instrucción o varias claves en un único elemento de `Condition`, AWS las evalúa mediante una operación `AND` lógica. Si especifica varios valores para una única clave de condición, AWS evalúa la condición con una operación lógica `OR`. Se deben cumplir todas las condiciones antes de que se concedan los permisos de la instrucción.

También puedes utilizar variables de marcador de posición al especificar condiciones. Por ejemplo, puedes conceder un permiso de usuario de IAM para acceder a un recurso solo si está etiquetado con su nombre de usuario de IAM. Para más información, consulta Elementos de la política de IAM: variables y etiquetas en la Guía del usuario de IAM.

AWS admite claves de condición globales y claves de condición específicas del servicio. Para obtener más información, consulte Claves de contexto de condición globales de AWS en la Guía del usuario de IAM.

> **Important**
>
> Si utiliza Automatización de Systems Manager, le recomendamos que no utilice la clave de condición aws:SourceIp en sus políticas. El comportamiento de esta clave de condición depende de varios factores, como si se proporciona un rol de IAM para la ejecución del manual de Automatización y las acciones de Automatización utilizadas en el manual de procedimientos. Como resultado, la clave de condición puede producir un comportamiento inesperado. Por este motivo, le recomendamos que no la use.

Systems Manager admite varias claves de condición propias. Para obtener más información, consulte Condition Keys for AWS Systems Manager en la Referencia de autorizaciones de servicio. Las acciones y los recursos con los que puede utilizar una clave de condición específica de Systems Manager se enumeran en Resource types defined by AWS Systems Manager en la Referencia de autorizaciones de servicio.

Si su política debe depender del nombre de entidad principal de servicio propiedad del servicio de Systems Manager, le recomendamos que compruebe su existencia mediante la clave de condición multivalor `aws:PrincipalServiceNamesList`, en lugar de la clave de condición `aws:PrincipalServiceName`. La clave de condición `aws:PrincipalServiceName` contiene solo una entrada de la lista de nombres de entidades principales de servicio y puede que no siempre sea el nombre de entidad principal de servicio esperado. En el siguiente bloque de `Condition` se muestra cómo comprobar la existencia de `ssm.amazonaws.com`.

```
{
    "Condition": {
        "ForAnyValue:StringEquals": {
            "aws:PrincipalServiceNamesList": "ssm.amazonaws.com"
        }
    }
}
```

Para ver ejemplos de políticas basadas en identidades de Systems Manager, consulte AWS Systems Manager ejemplos de políticas basadas en identidad de .

Listas de control de acceso (ACL)

Las listas de control de acceso (ACL) controlan qué entidades principales (miembros de cuentas, usuarios o roles) tienen permisos para acceder a un recurso. Las ACL son similares a las políticas basadas en recursos, aunque no utilizan el formato de documento de políticas JSON.

Amazon S3, AWS WAF y Amazon VPC son ejemplos de servicios que admiten las ACL. Para obtener más información sobre las ACL, consulta Información general de Lista de control de acceso (ACL) en la Guía para desarrolladores de Amazon Simple Storage Service.

Otros tipos de políticas

AWS admite otros tipos de políticas adicionales menos frecuentes. Estos tipos de políticas pueden establecer el máximo de permisos que los tipos de políticas más frecuentes le conceden.

- Límites de permisos: un límite de permisos es una característica avanzada que le permite establecer los permisos máximos que una política basada en identidad puede conceder a una entidad de IAM (usuario o rol de IAM). Puedes establecer un límite de permisos para una entidad. Los permisos resultantes son la intersección de las políticas basadas en la identidad de la entidad y los límites de permisos. Las políticas basadas en recursos que especifiquen el usuario o rol en el campo `Principal` no estarán restringidas por el límite de permisos. Una denegación explícita en cualquiera de estas políticas anulará el permiso. Para obtener más información sobre los límites de los permisos, consulta Límites de permisos para las entidades de IAM en la Guía del usuario de IAM.
- Políticas de control de servicio (SCP): las SCP son políticas de JSON que especifican los permisos máximos de una organización o una unidad organizativa en AWS Organizations. AWS Organizations es un servicio que le permite agrupar y administrar de manera centralizada varias Cuentas de AWS que posea su empresa. Si habilita todas las características en una empresa, entonces podrá aplicar políticas de control de servicio (SCP) a una o todas sus cuentas. Una SCP limita los permisos para las entidades de las cuentas de miembro, incluido cada Usuario raíz de la cuenta de AWS. Para obtener más información acerca de SCP y Organizations, consulta Políticas de control de servicios en la Guía del usuario de AWS Organizations.
- Políticas de control de recursos (RCP): las RCP son políticas JSON que permiten establecer los permisos máximos disponibles para los recursos de las cuentas sin actualizar las políticas de IAM asociadas a cada recurso que posea. La RCP limita los permisos de los recursos en las cuentas de miembros y puede afectar a los permisos efectivos de las identidades, incluidos los Usuario raíz de la cuenta de AWS, independientemente de si pertenecen a su organización. Para obtener más información sobre Organizations y RCP, incluida una lista de los Servicios de AWS

que admiten RCP, consulte Políticas de control de recursos (RCP) en la Guía del usuario de AWS Organizations.

- Políticas de sesión: las políticas de sesión son políticas avanzadas que se pasan como parámetro cuando se crea una sesión temporal mediante programación para un rol o un usuario federado. Los permisos de la sesión resultantes son la intersección de las políticas basadas en identidades del rol y las políticas de la sesión. Los permisos también puedes proceder de una política en función de recursos. Una denegación explícita en cualquiera de estas políticas anulará el permiso. Para más información, consulte Políticas de sesión en la Guía del usuario de IAM.

Varios tipos de políticas

Cuando se aplican varios tipos de políticas a una solicitud, los permisos resultantes son más complicados de entender. Para obtener información acerca de cómo AWS decide si permitir o no una solicitud cuando hay varios tipos de políticas implicados, consulte Lógica de evaluación de políticas en la Guía del usuario de IAM.

Cómo funciona AWS Systems Manager con IAM

Antes de utilizar AWS Identity and Access Management (IAM) para administrar el acceso a AWS Systems Manager, debe comprender qué características de IAM están disponibles para su uso con Systems Manager. Para obtener una perspectiva general sobre cómo funcionan Systems Manager y otros Servicios de AWS con IAM, consulte Servicios de AWS que funcionan con IAM en la Guía del usuario de IAM.

Temas

- Políticas de Systems Manager basadas en identidades
- Políticas de Systems Manager basadas en recursos
- Autorización basada en etiquetas de Systems Manager
- Roles de IAM de Systems Manager

Políticas de Systems Manager basadas en identidades

Con las políticas basadas en identidades de IAM, puede especificar las acciones y los recursos permitidos o denegados y las condiciones en las que se permiten o deniegan las acciones. Systems Manager admite acciones, claves de condiciones y recursos específicos. Para obtener más

información acerca de los elementos que utiliza en una política de JSON, consulte Referencia de los elementos de las políticas de JSON de IAM en la Guía del usuario de IAM.

Acciones

Los administradores pueden utilizar las políticas JSON de AWS para especificar quién tiene acceso a qué. Es decir, qué entidad principal puedes realizar acciones en qué recursos y en qué condiciones.

El elemento `Action` de una política JSON describe las acciones que puedes utilizar para conceder o denegar el acceso en una política. Las acciones de la política generalmente tienen el mismo nombre que la operación de API de AWS asociada. Hay algunas excepciones, como acciones de solo permiso que no tienen una operación de API coincidente. También hay algunas operaciones que requieren varias acciones en una política. Estas acciones adicionales se denominan acciones dependientes.

Incluya acciones en una política para conceder permisos y así llevar a cabo la operación asociada.

Las acciones de políticas de Systems Manager utilizan el siguiente prefijo antes de la acción: `ssm:`. Por ejemplo, para conceder a alguien permiso para crear un parámetro de Systems Manager (parámetro de SSM) con la operación de la API de Systems Manager `PutParameter`, incluya la acción `ssm:PutParameter` en su política. Las instrucciones de política deben incluir un elemento `Action` o `NotAction`. Systems Manager define su propio conjunto de acciones que describen las tareas que se pueden realizar con este servicio.

Para especificar varias acciones en una única instrucción, sepárelas con comas del siguiente modo:

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

> **Note**
>
> Las siguientes herramientas de AWS Systems Manager utilizan diferentes prefijos antes de las acciones.
>
> - AWS AppConfig usa el prefijo `appconfig:` antes de las acciones.
> - Administrador de incidentes usa el prefijo `ssm-incidents:` o `ssm-contacts:` antes de las acciones.

- La GUI Connect de Systems Manager usa el prefijo `ssm-guiconnect:` antes de las acciones.
- Quick Setup usa el prefijo `ssm-quicksetup:` antes de las acciones.

Puede utilizar caracteres comodín (*) para especificar varias acciones . Por ejemplo, para especificar todas las acciones que comiencen con la palabra `Describe`, incluya la siguiente acción:

```
"Action": "ssm:Describe*"
```

Para ver una lista de las acciones de Systems Manager, consulte Acciones definidas por AWS Systems Manager en la Referencia de autorizaciones de servicio.

Recursos

Los administradores pueden utilizar las políticas JSON de AWS para especificar quién tiene acceso a qué. Es decir, qué entidad principal puedes realizar acciones en qué recursos y en qué condiciones.

El elemento `Resource` de la política JSON especifica el objeto u objetos a los que se aplica la acción. Las instrucciones deben contener un elemento `Resource` o `NotResource`. Como práctica recomendada, especifique un recurso utilizando el Nombre de recurso de Amazon (ARN). Puedes hacerlo para acciones que admitan un tipo de recurso específico, conocido como permisos de nivel de recurso.

Para las acciones que no admiten permisos de nivel de recurso, como las operaciones de descripción, utiliza un carácter comodín (*) para indicar que la instrucción se aplica a todos los recursos.

```
"Resource": "*"
```

Por ejemplo, el recurso de periodo de mantenimiento de Systems Manager tiene el siguiente formato de ARN.

```
arn:aws:ssm:region:account-id:maintenancewindow/window-id
```

Para especificar los periodos de mantenimiento de mw-0c50858d01EXAMPLE en la instrucción en la región Este de EE. UU. (Ohio), debería utilizar un ARN similar al siguiente.

```
"Resource": "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-0c50858d01EXAMPLE"
```

Para especificar todos los periodos de mantenimiento que pertenecen a una cuenta específica, utilice el comodín (*).

```
"Resource": "arn:aws:ssm:region:123456789012:maintenancewindow/*"
```

Para las operaciones de la API `Parameter Store`, puede proporcionar o restringir el acceso a todos los parámetros en un nivel de una jerarquía utilizando nombres jerárquicos y políticas de AWS Identity and Access Management (IAM) tal como se indica a continuación.

```
"Resource": "arn:aws:ssm:region:123456789012:parameter/Dev/ERP/Oracle/*"
```

Algunas acciones de Systems Manager, como las empleadas para la creación de recursos, no se pueden llevar a cabo en un recurso específico. En dichos casos, debe utilizar el carácter comodín (*).

```
"Resource": "*"
```

Algunas operaciones de la API de Systems Manager aceptan varios recursos. Para especificar varios recursos en una única instrucción, separe sus ARN con comas, tal y como se indica a continuación.

```
"Resource": [
      "resource1",
      "resource2"
```

> **Note**
>
> La mayoría de los Servicios de AWS tratan el carácter de dos puntos (:) o la barra inclinada (/) como el mismo carácter en los ARN. Sin embargo, Systems Manager requiere una coincidencia exacta en las reglas y los patrones de los recursos. Al crear patrones de eventos, asegúrese de utilizar los caracteres de ARN correctos para que coincidan con el ARN del recurso.

En la siguiente tabla se describen los formatos de ARN para los tipos de recursos admitidos por Systems Manager.

Note

Tenga en cuenta las siguientes excepciones a los formatos ARN.

- Las siguientes herramientas de AWS Systems Manager utilizan diferentes prefijos antes de las acciones.
  - AWS AppConfig usa el prefijo `appconfig:` antes de las acciones.
  - Administrador de incidentes usa el prefijo `ssm-incidents:` o `ssm-contacts:` antes de las acciones.
  - La GUI Connect de Systems Manager usa el prefijo `ssm-guiconnect` antes de las acciones.
- Los documentos y los recursos de definición de automatización que son propiedad de Amazon, así como los parámetros públicos que proporcionan tanto Amazon como fuentes de terceros, no incluyen los ID de cuenta en sus formatos ARN. Por ejemplo:
  - Uso de documentos `AWS-RunPatchBaseline` de SSM:

    ```
    arn:aws:ssm:us-east-2::document/AWS-RunPatchBaseline
    ```
  - Manual de procedimientos de automatización `AWS-ConfigureMaintenanceWindows`:

    ```
    arn:aws:ssm:us-east-2::automation-definition/AWS-
    ConfigureMaintenanceWindows
    ```
  - Parámetros públicos `/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
    ```

  Para obtener más información sobre estas características, consulte los siguientes temas:

  - Trabajo con documentos
  - Ejecución de una operación automatizada con la tecnología de automatización de Systems Manager
  - Trabajo con parámetros públicos en Parameter Store
- Quick Setup usa el prefijo `ssm-quicksetup:` antes de las acciones.

Tipo de recurso	Formato de ARN
Aplicación (AWS AppConfig)	arn:aws:appconfig:*region*:*account-id* :application/*application-id*
Asociación	arn:aws:ssm:*region*:*account-id* :association/*association-id*
Ejecución de automatización	arn:aws:ssm:*región*:*id-cuenta* :automation-execution/*id-ejecución-automatización*
Definición de automatización (con subrecurso de versión)	arn:aws:ssm:*región*:*id-cuenta* :automation-definition/*id-definición-automatización* :*id-versión* [1]
Perfil de configuración (AWS AppConfig)	arn:aws:appconfig:*region*:*account-id* :application/*application-id* /configurationprofile/*configurationprofile-id*
Contacto (Incident Manager)	arn:aws:ssm-contacts:*region*:*account-id* :contact/*contact-alias*
Estrategia de implementación (AWS AppConfig)	arn:aws:appconfig:*region*:*account-id* :deploymentstrategy/*deploymentstrategy-id*
Documento	arn:aws:ssm:*región*:*id-cuenta* :document/*nombre-documento*
Entorno (AWS AppConfig)	arn:aws:appconfig:*region*:*account-id* :application/*application-id* /environment/*environment-id*
Incidente	arn:aws:ssm-incidents:*region*:*account-id* :incident-record/*response-plan-name* /*incident-id*
Periodo de mantenimiento	arn:aws:ssm:*region*:*account-id* :maintenancewindow/*window-id*
Nodo administrado	arn:aws:ssm:*region*:*account-id* :managed-instance/*managed-node-id*
Inventario de nodos administrados	arn:aws:ssm:*region*:*account-id* :managed-instance-inventory /*managed-node-id*

Tipo de recurso	Formato de ARN
OpsItem	arn:aws:ssm:*region*:*account-id* :opsitem/*OpsItem-id*
Parámetro	Parámetros de un nivel: • arn:aws:ssm:*región*:*id-cuenta* :parameter/*nombre-parámetro* / Un parámetro denominado con una construcción jerárquica: • arn:aws:ssm:*región*:*id-cuenta* :parameter/*raíz-nombre-parámetro* /*nivel-2*/*nivel-3*/*nivel-4*/*nivel-5* [2]
línea de base de revisiones	arn:aws:ssm:*región*:*id-cuenta* :patchbaseline/*id-base-referencia-revisiones*
Plan de respuesta	arn:aws:ssm-incidents:*region*:*account-id* :response-plan/*response-plan-name*
Sesión	arn:aws:ssm:*región*:*id-cuenta* :session/*id-sesión* [3]
Todos los recursos de Systems Manager	arn:aws:ssm:*
Todos los recursos de Systems Manager que pertenecen a la Cuenta de AWS especificada en la Región de AWS indicada	arn:aws:ssm:*región*:*id-cuenta* :*

[1]

Para las definiciones de automatización, Systems Manager admite un recurso de segundo nivel, el ID de versión. En AWS, estos recursos de segundo nivel se denominan subrecursos. Si especifica un subrecurso de versión para un recurso de definición de automatización, le permite proporcionar

acceso a determinadas versiones de una definición de automatización. Por ejemplo, es posible que desee garantizar que solo la versión más reciente de una definición de automatización se utilice en su administración de nodos.

2

Para organizar y administrar parámetros, puede crear nombres para parámetros con una estructura jerárquica. Con dicha estructura, un nombre de parámetro puede incluir una ruta que se define con barras inclinadas. Puede dar un nombre a un recurso de parámetro con un máximo de quince niveles. Le recomendamos que cree jerarquías que reflejen una estructura jerárquica existente en su entorno. Para obtener más información, consulte Creación de parámetros de Parameter Store en Systems Manager.

3

En la mayoría de los casos, el ID de sesión se construye con el ID del usuario de la cuenta que inició la sesión, además de un sufijo alfanumérico. Por ejemplo:

```
arn:aws:us-east-2:111122223333:session/JohnDoe-1a2b3c4sEXAMPLE
```

Sin embargo, si el ID de usuario no está disponible, el ARN se construye de esta manera:

```
arn:aws:us-east-2:111122223333:session/session-1a2b3c4sEXAMPLE
```

Para obtener más información acerca del formato de los ARN, consulte Nombres de recursos de Amazon (ARN) en la Referencia general de Amazon Web Services.

Para ver una lista de los tipos de recursos de Systems Manager y sus ARN, consulte Recursos definidos por AWS Systems Manager en la Referencia de autorizaciones de servicio. Para obtener información sobre las acciones con las que puede especificar el ARN de cada recurso, consulte Acciones definidas por AWS Systems Manager.

Claves de condición de Systems Manager

Los administradores puedes utilizar las políticas JSON de AWS para especificar quién tiene acceso a qué. Es decir, qué entidad principal puedes realizar acciones en qué recursos y en qué condiciones.

El elemento `Condition` (o bloque de `Condition`) permite especificar condiciones en las que entra en vigor una instrucción. El elemento `Condition` es opcional. Puedes crear expresiones condicionales que utilizan operadores de condición, tales como igual o menor que, para que la condición de la política coincida con los valores de la solicitud.

Si especifica varios elementos de `Condition` en una instrucción o varias claves en un único elemento de `Condition`, AWS las evalúa mediante una operación `AND` lógica. Si especifica varios valores para una única clave de condición, AWS evalúa la condición con una operación lógica `OR`. Se deben cumplir todas las condiciones antes de que se concedan los permisos de la instrucción.

También puedes utilizar variables de marcador de posición al especificar condiciones. Por ejemplo, puedes conceder un permiso de usuario de IAM para acceder a un recurso solo si está etiquetado con su nombre de usuario de IAM. Para más información, consulta Elementos de la política de IAM: variables y etiquetas en la Guía del usuario de IAM.

AWS admite claves de condición globales y claves de condición específicas del servicio. Para ver todas las claves de condición globales de AWS, consulte Claves de contexto de condición globales de AWS en la Guía del usuario de IAM.

Para ver una lista de las claves de condición de Systems Manager, consulte Claves de condición para AWS Systems Manager en la Referencia de autorizaciones de servicio. Para obtener más información sobre las acciones y los recursos con los que puede utilizar una clave de condición, consulte Acciones definidas por AWS Systems Manager.

Para obtener información sobre el uso de la clave de condición `ssm:resourceTag/*`, consulte los temas siguientes:

- Restricción del acceso a los comandos de nivel raíz con SSM Agent
- Restricción de acceso de Run Command basado en etiquetas
- Restringir el acceso de sesión en función de las etiquetas de instancia

Para obtener información sobre el uso de las claves de condición `ssm:Recursive`, `ssm:Policies` y `ssm:Overwrite`, consulte Impedir el acceso a las operaciones de la API de Parameter Store.

Ejemplos

Para ver ejemplos de políticas basadas en identidad de Systems Manager, consulte AWS Systems Manager ejemplos de políticas basadas en identidad de .

Políticas de Systems Manager basadas en recursos

Otros Servicios de AWS, como Amazon Simple Storage Service (Amazon S3), admiten políticas de permisos basadas en recursos. Por ejemplo, puede asociar una política de permisos a un bucket de S3 para administrar los permisos de acceso a dicho bucket.

Systems Manager no admite políticas basadas en recursos.

Autorización basada en etiquetas de Systems Manager

Puede asociar etiquetas a los recursos de Systems Manager o transferirlas en una solicitud a Systems Manager. Para controlar el acceso según las etiquetas, debe proporcionar información de las etiquetas en el elemento de condición de una política mediante las claves de condición `ssm:resourceTag/key-name`, `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` o `aws:TagKeys`. Puede agregar etiquetas a los siguientes tipos de recursos al crearlos o actualizarlos:

- Documento
- Nodo administrado
- Periodo de mantenimiento
- Parámetro
- Línea de base de revisiones
- OpsItem

Para consultar un ejemplo de política basada en la identidad para limitar el acceso a un recurso en función de las etiquetas de ese recurso, consulte Visualización de documentos de Systems Manager basados en etiquetas.

Roles de IAM de Systems Manager

Un rol de IAM es una entidad de la Cuenta de AWS que dispone de permisos específicos.

Uso de credenciales temporales con Systems Manager

Puede utilizar credenciales temporales para iniciar sesión con federación, asumir un rol de IAM o asumir un rol de acceso entre cuentas. Las credenciales de seguridad temporales se obtienen mediante una llamada a operaciones de la API de AWS Security Token Service (AWS STS), como AssumeRole o GetFederationToken.

Systems Manager admite el uso de credenciales temporales.

Roles vinculados a servicios

Los roles vinculados a servicios permiten a los Servicios de AWS obtener acceso a los recursos de otros servicios para completar una acción en su nombre. Los roles vinculados a servicios están

listados en la cuenta de IAM y son propiedad del servicio. Un administrador puede ver, pero no editar, los permisos de los roles vinculados a servicios.

Systems Manager admite roles vinculados a servicios. Para obtener más información acerca de cómo crear o administrar roles vinculados a servicios de Systems Manager, consulte Cómo utilizar roles vinculados a servicios de Systems Manager.

Roles de servicio

Esta característica permite que un servicio asuma un rol de servicio en su nombre. Este rol permite que el servicio obtenga acceso a los recursos de otros servicios para completar una acción en su nombre. Las funciones del servicio se muestran en su cuenta de IAM y son propiedad de la cuenta. Esto significa que un administrador puede cambiar los permisos de este rol. Sin embargo, hacerlo podría deteriorar la funcionalidad del servicio.

Systems Manager admite roles de servicio.

Elección de un rol de IAM en Systems Manager

Para que Systems Manager interactúe con los nodos administrados, debe elegir un rol que permita a Systems Manager acceder a los nodos en su nombre. Si ha creado previamente un rol de servicio o un rol vinculado a servicios, Systems Manager le proporciona una lista de roles para elegir. Es importante seleccionar un rol que le permita el acceso para iniciar y detener nodos administrados.

Para acceder a las instancias de EC2, debe configurar los permisos de instancia. Para obtener información, consulte Configuración de permisos de instancia requeridos para Systems Manager.

Para acceder a los nodos que no son de EC2 en una nube híbrida y multinube, el rol que su Cuenta de AWS necesita es un rol de servicio de IAM. Para obtener información, consulte Creación del rol de servicio de IAM requerido para Systems Manager en entornos híbridos y multinube.

Un flujo de trabajo de Automation se puede iniciar en el contexto de un rol de servicio (o rol de asunción). Esto permite al servicio realizar acciones en su nombre. Si no especifica un rol de asunción, Automation utiliza el contexto del usuario que invocó la ejecución. Sin embargo, algunas situaciones requieren especificar un rol de servicio para Automation. Para obtener más información, consulte Configuración del acceso de un rol de servicio (rol de asunción) para automatizaciones.

Políticas gestionadas de AWS Systems Manager

AWS aborda muchos casos de uso comunes dando políticas de IAM independientes creadas y administradas por AWS. Estas AWSpolíticas administradas de conceden los permisos necesarios

para casos de uso común, lo que le evita tener que investigar qué permisos se necesitan. (También puede crear sus propias políticas de IAM personalizadas para conceder permisos a las acciones y recursos de Systems Manager).

Para obtener más información sobre las políticas administradas de Systems Manager, consulte Políticas administradas de AWS para AWS Systems Manager

Para obtener más información sobre las políticas administradas, consulte Políticas administradas por AWS en la Guía del usuario de IAM.

AWS Systems Manager ejemplos de políticas basadas en identidad de

De forma predeterminada, las entidades de AWS Identity and Access Management (IAM) (usuarios y roles) no tienen permiso para crear o modificar recursos de AWS Systems Manager. Tampoco se pueden realizar tareas con la consola de Systems Manager, la AWS Command Line Interface (AWS CLI), o la API de AWS. Un administrador debe crear políticas de IAM que concedan permisos a los usuarios y a los roles para realizar operaciones de la API concretas en los recursos especificados que necesiten. El administrador debe adjuntar esas políticas a los usuarios o grupos que necesiten esos permisos.

A continuación se muestra un ejemplo de una política de permisos que permite a un usuario eliminar documentos con nombres que comienzan con **`MyDocument-`** en la Región de AWS Este de EE. UU. (Ohio) (us-east-2).

JSON

```
{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "ssm:DeleteDocument"
      ],
      "Resource" : [
        "arn:aws:ssm:us-east-2:111122223333:document/MyDocument-*"
      ]
    }
  ]
}
```

Para obtener información acerca de cómo se crea una política basada en identidades de IAM con estos documentos de políticas JSON de ejemplo, consulte Creating IAM policies (Creación de políticas de IAM) en la Guía del usuario de IAM.

Temas

- Prácticas recomendadas sobre las políticas
- Ejemplo: permiso para usar la consola de Systems Manager
- Ejemplo: permiso para que los usuarios vean sus propios permisos
- Ejemplo: permiso para leer y describir parámetros individuales
- Prevención de la sustitución confusa entre servicios
- Ejemplos de políticas administradas por el cliente
- Visualización de documentos de Systems Manager basados en etiquetas

Prácticas recomendadas sobre las políticas

Las políticas basadas en identidades determinan si alguien puede crear, acceder o eliminar los recursos de Systems Manager de la cuenta. Estas acciones pueden generar costes adicionales para su Cuenta de AWS. Siga estas directrices y recomendaciones al crear o editar políticas basadas en identidades:

- Comienza con las políticas administradas por AWSy continúa con los permisos de privilegio mínimo: a fin de comenzar a conceder permisos a los usuarios y las cargas de tarea, utiliza las políticas administradas por AWS, que conceden permisos para muchos casos de uso comunes. Están disponibles en su Cuenta de AWS. Se recomienda definir políticas gestionadas por el cliente de AWS específicas para sus casos de uso a fin de reducir aún más los permisos. Con el fin de obtener más información, consulta las políticas administradas por AWS o las políticas administradas por AWS para funciones de tarea en la Guía de usuario de IAM.
- Aplique permisos de privilegio mínimo: cuando establezca permisos con políticas de IAM, conceda solo los permisos necesarios para realizar una tarea. Para ello, debe definir las acciones que se pueden llevar a cabo en determinados recursos en condiciones específicas, también conocidos como permisos de privilegios mínimos. Con el fin de obtener más información sobre el uso de IAM para aplicar permisos, consulta Políticas y permisos en IAM en la Guía del usuario de IAM.
- Utilice condiciones en las políticas de IAM para restringir aún más el acceso: puede agregar una condición a sus políticas para limitar el acceso a las acciones y los recursos. Por ejemplo, puedes escribir una condición de políticas para especificar que todas las solicitudes deben

enviarse utilizando SSL. También puedes usar condiciones para conceder acceso a acciones de servicios si se emplean a través de un Servicio de AWS determinado como, por ejemplo, AWS CloudFormation. Para obtener más información, consulta Elementos de la política de JSON de IAM: Condición en la Guía del usuario de IAM.

- Utiliza el analizador de acceso de IAM para validar las políticas de IAM con el fin de garantizar la seguridad y funcionalidad de los permisos: el analizador de acceso de IAM valida políticas nuevas y existentes para que respeten el lenguaje (JSON) de las políticas de IAM y las prácticas recomendadas de IAM. El analizador de acceso de IAM proporciona más de 100 verificaciones de políticas y recomendaciones procesables para ayudar a crear políticas seguras y funcionales. Para más información, consulte Validación de políticas con el Analizador de acceso de IAM en la Guía del usuario de IAM.
- Solicite la autenticación multifactor (MFA): si se encuentra en una situación en la que necesite usuarios raíz o de IAM en su Cuenta de AWS, active la MFA para obtener una mayor seguridad. Para exigir la MFA cuando se invoquen las operaciones de la API, añada condiciones de MFA a sus políticas. Para más información, consulte Acceso seguro a la API con MFA en la Guía del usuario de IAM.

Para obtener más información sobre las prácticas recomendadas de IAM, consulte Prácticas recomendadas de seguridad en IAM en la Guía del usuario de IAM.

Ejemplo: permiso para usar la consola de Systems Manager

Para acceder a la consola de Systems Manager, debe tener un conjunto mínimo de permisos. Estos permisos deben permitirle registrar y consultar los detalles acerca de los recursos de Systems Manager y otros recursos en su Cuenta de AWS.

Si crea una política basada en identidades que es más restrictiva que los permisos mínimos requeridos, la consola no funcionará según lo previsto para las entidades de IAM (usuarios o roles) con esa política.

No es necesario que conceda permisos mínimos para la consola a los usuarios que solo realizan llamadas a la AWS CLI o a la API de AWS. En su lugar, permite acceso únicamente a las acciones que coincidan con la operación de API que intenta realizar.

Para asegurarse de que los usuarios y los roles puedan seguir utilizando la consola de Systems Manager, asocie también la política administrada de AWS AmazonSSMFullAccess o AmazonSSMReadOnlyAccess a las entidades. Para obtener más información, consulte Adición de permisos a un usuario en la Guía del usuario de IAM:

Ejemplo: permiso para que los usuarios vean sus propios permisos

En este ejemplo, se muestra cómo podría crear una política que permita a los usuarios de IAM ver las políticas gestionadas e insertadas que se asocian a la identidad de sus usuarios. Esta política incluye permisos para llevar a cabo esta acción en la consola o mediante programación con la AWS CLI o la API de 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": "*"
        }
    ]
}
```

Ejemplo: permiso para leer y describir parámetros individuales

Example Leer y describir un parámetro

Puede conceder acceso a un secreto si adjunta la siguiente política a una identidad.

JSON

```
{
"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "ssm:GetParameter",
      "ssm:DescribeParameters"
      ],
    "Resource": "arn:aws:ssm:us-east-1:111122223333:parameter/parameter-name"
  }
]
}
```

Prevención de la sustitución confusa entre servicios

El problema de la sustitución confusa es un problema de seguridad en el que una entidad que no tiene permiso para realizar una acción puede obligar a una entidad con más privilegios a realizar la acción. En AWS, la suplantación entre servicios puede dar lugar al problema de la sustitución confusa. La suplantación entre servicios puedes producirse cuando un servicio (el servicio que lleva a cabo las llamadas) llama a otro servicio (el servicio al que se llama). El servicio que lleva a cabo las llamadas se puedes manipular para utilizar sus permisos a fin de actuar en función de los recursos de otro cliente de una manera en la que no debe tener permiso para acceder. Para evitarlo, AWS proporciona herramientas que lo ayudan a proteger sus datos para todos los servicios con entidades principales de servicio a las que se les ha dado acceso a los recursos de su cuenta.

Se recomienda utilizar las claves de contexto de condición global `aws:SourceArn` y `aws:SourceAccount` en las políticas de recursos para limitar los permisos que AWS Systems Manager concede a otro servicio para el recurso. Si el valor de `aws:SourceArn` no contiene el ID de cuenta, como un nombre de recurso de Amazon (ARN) de bucket de S3, debe utilizar ambas claves de contexto de condición global para limitar los permisos. Si utiliza claves de

contexto de condición global y el valor de `aws:SourceArn` contiene el ID de cuenta, el valor de `aws:SourceAccount` y la cuenta en el valor de `aws:SourceArn` deben utilizar el mismo ID de cuenta cuando se utiliza en la misma instrucción de política. Utiliza `aws:SourceArn` si desea que solo se asocie un recurso al acceso entre servicios. Utiliza `aws:SourceAccount` si quiere permitir que cualquier recurso de esa cuenta se asocie al uso entre servicios.

En las siguientes secciones, se presentan ejemplos de políticas para herramientas de AWS Systems Manager.

Ejemplo de política de activación híbrida

Para los roles de servicio utilizados en una activación híbrida, el valor de `aws:SourceArn` debe ser el ARN de la Cuenta de AWS. Asegúrese de especificar la Región de AWS en el ARN donde creó la activación híbrida. Si no conoce el ARN completo del recurso o si especifica varios recursos, utilice la clave de condición de contexto global `aws:SourceArn` con comodines (*) para las partes desconocidas del ARN. Por ejemplo, `arn:aws:ssm:*:region:123456789012:*`.

En el siguiente ejemplo se muestra el uso de las claves de contexto de condición global `aws:SourceArn` y `aws:SourceAccount` para Automation para evitar el problema del suplente confuso en la región Este de EE. UU. (Ohio) (us-east-2).

JSON

```
{
   "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:*"
            }
         }
```

```
      }
   ]
}
```

Ejemplo de política de sincronización de datos de recursos

Systems Manager Inventory, Explorer y Compliance le permiten crear una sincronización de datos de recursos para centralizar el almacenamiento de los datos de operaciones (OpsData) en un bucket central de Amazon Simple Storage Service. Si desea cifrar la sincronización de datos de recursos mediante AWS Key Management Service (AWS KMS), debe crear una clave nueva que incluya la siguiente política o actualizar una clave existente y agregarle esta política. Las claves de condición `aws:SourceArn` y `aws:SourceAccount` de esta política evitan el problema del suplente confuso. A continuación, se muestra un ejemplo de política.

JSON

```
{
    "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**
>
> El ARN del ejemplo de política permite al sistema cifrar OpsData de todos los orígenes, excepto AWS Security Hub. Si necesita cifrar los datos de Security Hub, por ejemplo, si utiliza Explorer para recopilar datos de Security Hub, debe adjuntar una política adicional que especifique el siguiente ARN:
> `"aws:SourceArn": "arn:aws:ssm:*:account-id:role/aws-service-role/opsdatasync.ssm.amazonaws.com/AWSServiceRoleForSystemsManagerOpsDataSync"`

Ejemplos de políticas administradas por el cliente

Puede crear políticas independientes que puede administrar en su propia Cuenta de AWS. Las denominamos políticas administradas por el cliente. Puede adjuntar estas políticas a varias entidades principales de su Cuenta de AWS. Al asociar una política a una entidad principal, concederá a la entidad los permisos que están definidos en la política. Para obtener más información, consulte Customer managed policy examples (Ejemplos de políticas administradas por el cliente) en la IAM User Guide (Guía del usuario de IAM).

En los siguientes ejemplos de políticas de usuario, se concede permiso para diversas acciones de Systems Manager. Úselas para limitar el acceso a Systems Manager de sus entidades de IAM (usuarios y roles). Estas políticas funcionan cuando ejecuta acciones en la API de Systems Manager, los AWS SDK o la AWS CLI. En el caso de los usuarios que utilizan la consola, debe conceder permisos adicionales específicos a la consola. Para obtener más información, consulte Ejemplo: permiso para usar la consola de Systems Manager.

> **Note**
>
> Todos los ejemplos utilizan la región EE. UU. Oeste (Oregón) (us-west-2) y contienen identificadores de cuenta ficticios. No debe especificarse el ID de cuenta en el nombre de recurso de Amazon (ARN) para documentos públicos de AWS (documentos que comienzan con `AWS-*`).

Ejemplos

- Ejemplo 1: permitir a un usuario realizar operaciones de Systems Manager en una única región
- Ejemplo 2: permitir a un usuario generar una lista de documentos de una única región

Ejemplo 1: permitir a un usuario realizar operaciones de Systems Manager en una única región

En el siguiente ejemplo se conceden permisos para realizar operaciones de Systems Manager solo en la región Este de EE. UU. (Ohio) (us-east-2).

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:*"
            ],
            "Resource": [
                "arn:aws:ssm:us-east-2:111122223333:*"
            ]
        }
    ]
}
```

Ejemplo 2: permitir a un usuario generar una lista de documentos de una única región

En el siguiente ejemplo se conceden permisos para enumerar todos los nombres de documentos que comienzan con **`Update`** en la región Este de EE. UU. (Ohio) (us-east-2).

JSON

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

```
            "Action": [
                "ssm:ListDocuments"
            ],
            "Resource": [
                "arn:aws:ssm:us-east-2:111122223333:document/Update*"
            ]
        }
    ]
}
```

Ejemplo 3: permitir a un usuario utilizar un documento de SSM concreto para ejecutar comandos en nodos específicos

El siguiente ejemplo de política de IAM permite a un usuario hacer lo siguiente en la región Este de EE. UU.(Ohio) (us-east-2):

- Permite enumerar documentos de Systems Manager (documentos de SSM) y versiones de documentos.
- Ver detalles sobre los documentos.
- Enviar un comando utilizando el documento especificado en la política. El nombre del documento se determina con la siguiente entrada:

```
arn:aws:ssm:us-east-2:aws-account-ID:document/Systems-Manager-document-name
```

- Enviar un comando a tres nodos. Los nodos se determinan en función de las siguientes entradas de la segunda sección `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"
```

- Ver detalles sobre un comando después de que se haya enviado.
- Permite comenzar y detener flujos de trabajo de Automatización, una herramienta de AWS Systems Manager.
- Permite obtener información acerca de los flujos de trabajo de Automation.

Si desea otorgar un permiso a un usuario para usar este documento con el fin de enviar comandos en cualquier nodo al que el usuario tenga acceso, puede especificar una entrada similar a la

siguiente en la sección `Resource` y eliminar las otras entradas de nodo. En el siguiente ejemplo se utiliza la región Este de EE. UU. (Ohio) (us-east-2).

```
"arn:aws:ec2:us-east-2:*:instance/*"
```

JSON

```
{
    "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-2:111122223333:instance/
i-02573cafcfEXAMPLE",
                "arn:aws:ec2:us-east-2:111122223333:instance/
i-0471e04240EXAMPLE",
                "arn:aws:ec2:us-east-2:111122223333:instance/
i-07782c72faEXAMPLE",

                "arn:aws:ssm:us-east-2:111122223333: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-2:111122223333:automation-definition/*"
            ]
        },
        {
            "Action": "ssm:DescribeAutomationExecutions",
            "Effect": "Allow",
            "Resource": [
                "*"
            ]
        },
        {
            "Action": [
                "ssm:StopAutomationExecution",
                "ssm:GetAutomationExecution"
            ],
            "Effect": "Allow",
            "Resource": [
                "*"
            ]
        }
    ]
}
```

Visualización de documentos de Systems Manager basados en etiquetas

Puede utilizar las condiciones de su política basada en identidad para controlar el acceso a los recursos de Systems Manager basados en etiquetas. En este ejemplo, se muestra cómo crear una

política que permita visualizar un documento de SSM. Sin embargo, los permisos solo se conceden si la etiqueta de documento `0wner` tiene el valor del nombre de usuario de dicho usuario. Esta política también proporciona los permisos necesarios para llevar a cabo esta acción en la consola.

JSON

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

También puede adjuntar esta política al usuario de en su cuenta. Si un usuario llamado `richard-roe` intenta ver un documento de Systems Manager, el documento debe estar etiquetado como `0wner=richard-roe` o `owner=richard-roe`. De lo contrario, se le deniega el acceso. La clave de la etiqueta de condición `0wner` coincide con `0wner` y `owner` porque los nombres de clave de condición no distinguen entre mayúsculas y minúsculas. Para obtener más información, consulta Elementos de la política de JSON de IAM: Condición en la Guía del usuario de IAM.

Políticas administradas de AWS para AWS Systems Manager

Una política administrada de AWS es una política independiente que AWS crea y administra. Las políticas administradas de AWS se diseñan para ofrecer permisos para muchos casos de uso comunes, por lo que puede empezar a asignar permisos a los usuarios, grupos y roles.

Considere que es posible que las políticas administradas de AWS no concedan permisos de privilegio mínimo para los casos de uso concretos, ya que están disponibles para que las utilicen todos los clientes de AWS. Se recomienda definir políticas administradas por el cliente específicas para sus casos de uso a fin de reducir aún más los permisos.

No puede cambiar los permisos definidos en las políticas administradas de AWS. Si AWS actualiza los permisos definidos en una política administrada de AWS, la actualización afecta a todas las identidades de entidades principales (usuarios, grupos y roles) a las que está adjunta la política. Lo más probable es que AWS actualice una política administrada de AWS cuando se lance un nuevo Servicio de AWS o las operaciones de la API nuevas estén disponibles para los servicios existentes.

Para obtener más información, consulte Políticas administradas de AWS en la Guía del usuario de IAM.

Temas

- Política administrada de AWS: AmazonSSMServiceRolePolicy
- Política administrada de AWS: AmazonSSMAutomationRole
- Política administrada de AWS: AmazonSSMReadOnlyAccess
- Política administrada por AWS: AWSSystemsManagerOpsDataSyncServiceRolePolicy
- Política administrada por AWS: AmazonSSMManagedEC2InstanceDefaultPolicy
- Política administrada de AWS: SSMQuickSetupRolePolicy
- Política administrada de AWS: AWSQuickSetupDeploymentRolePolicy
- Política administrada de AWS: AWSQuickSetupPatchPolicyDeploymentRolePolicy
- Política adminsitrada de AWS: AWSQuickSetupPatchPolicyBaselineAccess
- Política administrada de AWS: AWSSystemsManagerEnableExplorerExecutionPolicy
- Política administrada de AWS: AWSSystemsManagerEnableConfigRecordingExecutionPolicy
- Política administrada de AWS: AWSQuickSetupDevOpsGuruPermissionsBoundary

- Política administrada de AWS: AWSQuickSetupDistributorPermissionsBoundary
- Política administrada de AWS: AWSQuickSetupSSMHostMgmtPermissionsBoundary
- Política administrada de AWS: AWSQuickSetupPatchPolicyPermissionsBoundary
- Política administrada de AWS: AWSQuickSetupSchedulerPermissionsBoundary
- Política administrada de AWS: AWSQuickSetupCFGCPacksPermissionsBoundary
- Política administrada de AWS: AWS-SSM-DiagnosisAutomation-AdministrationRolePolicy
- Política administrada de AWS: AWS-SSM-DiagnosisAutomation-ExecutionRolePolicy
- Política administrada de AWS: AWS-SSM-RemediationAutomation-AdministrationRolePolicy
- Política administrada de AWS: AWS-SSM-RemediationAutomation-ExecutionRolePolicy
- Política administrada de AWS: AWSQuickSetupSSMManageResourcesExecutionPolicy
- Política administrada de AWS: AWSQuickSetupSSMLifecycleManagementExecutionPolicy
- Política administrada de AWS: AWSQuickSetupSSMDeploymentRolePolicy
- Política administrada de AWS: AWSQuickSetupSSMDeploymentS3BucketRolePolicy
- Política administrada de AWS: AWSQuickSetupEnableDHMCExecutionPolicy
- Política administrada de AWS: AWSQuickSetupEnableAREXExecutionPolicy
- Política administrada de AWS: AWSQuickSetupManagedInstanceProfileExecutionPolicy
- Política administrada de AWS: AWSQuickSetupFullAccess
- Política administrada de AWS: AWSQuickSetupReadOnlyAccess
- Política administrada por AWS: AWSQuickSetupManageJITNAResourcesExecutionPolicy
- Política administrada por AWS: AWSQuickSetupJITNADeploymentRolePolicy
- Política administrada por AWS: AWSSystemsManagerJustInTimeAccessServicePolicy
- Política administrada por AWS: AWSSystemsManagerJustInTimeAccessTokenPolicy
- Política administrada por AWS: AWSSystemsManagerJustInTimeAccessTokenSessionPolicy
- Política administrada por AWS: AWSSystemsManagerJustInTimeNodeAccessRolePropagationPolicy
- Política administrada por AWS: AWSSystemsManagerNotificationsServicePolicy
- Política administrada de AWS: AWS-SSM-Automation-DiagnosisBucketPolicy
- Política administrada de AWS: AWS-SSM-RemediationAutomation-OperationalAccountAdministrationRolePolicy
- Política administrada de AWS: AWS-SSM-DiagnosisAutomation-OperationalAccountAdministrationRolePolicy

- Actualizaciones de Systems Manager para las políticas administradas de AWS
- Políticas administradas adicionales para Systems Manager

Política administrada de AWS: AmazonSSMServiceRolePolicy

Esta política proporciona acceso a una serie de recursos de AWS que son administrados por AWS Systems Manager o utilizados en las operaciones de Systems Manager.

No puede adjuntar `AmazonSSMServiceRolePolicy` a sus entidades de AWS Identity and Access Management (IAM). Esta política está adjunta a un rol vinculado a servicios que permite a AWS Systems Manager realizar acciones en su nombre. Para obtener más información, consulte Uso de roles para recopilar datos de inventario y ver OpsData.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite a las entidades principales iniciar y escalonar las ejecuciones de Run Command y Automatización; recuperar información sobre las operaciones de Run Command y Automatización; recuperar información sobre los parámetros del Parameter Store y calendarios de Change Calendar; actualizar y recuperar información sobre la configuración del servicio de Systems Manager para recursos de OpsCenter; y leer información sobre las etiquetas que se aplicaron a los recursos.
- `cloudformation`: permite a las entidades principales recuperar información sobre las operaciones y las instancias de los conjuntos de pilas, así como eliminarlos del recurso `arn:aws:cloudformation:*:*:stackset/AWS-QuickSetup-SSM*:*`. Permite a las entidades principales eliminar las instancias de pila asociadas a los siguientes recursos:

```
arn:aws:cloudformation:*:*:stackset/AWS-QuickSetup-SSM*:*
arn:aws:cloudformation:*:*:stackset-target/AWS-QuickSetup-SSM*:*
arn:aws:cloudformation:*:*:type/resource/*
```

- `cloudwatch`: permite a las entidades principales obtener información sobre alarmas de Amazon CloudWatch.
- `compute-optimizer`: permite a las entidades principales recuperar el estado de inscripción (suscripción) de una cuenta en el servicio de AWS Compute Optimizer y recuperar recomendaciones para las instancias de Amazon EC2 que cumplan con un conjunto específico de requisitos establecidos.

- `config`: permite a las entidades principales recuperar información, corregir las configuraciones y los registradores de configuración de AWS Config y determinar si las reglas de AWS Config y los recursos de AWS especificados cumplen con los requisitos.
- `events`. permite a las entidades principales recuperar información sobre las reglas de EventBridge; crear reglas y destino de EventBridge exclusivamente para el servicio de Systems Manager (`ssm.amazonaws.com`); y eliminar reglas y destinos del recurso `arn:aws:events:*:*:rule/SSMExplorerManagedRule`.
- `ec2`: permite a las entidades principales recuperar información sobre las instancias de Amazon EC2.
- `iam`: permite a las entidades principales transferir permisos de roles para el servicio de Systems Manager (`ssm.amazonaws.com`).
- `lambda`: permite a las entidades principales invocar funciones de Lambda configuradas específicamente para su uso por parte de Systems Manager.
- `resource-explorer-2`: permite a las entidades principales recuperar datos sobre las instancias de EC2 para determinar si Systems Manager administra actualmente cada instancia.

  La acción `resource-explorer-2:CreateManagedView` está permitida para el recurso `arn:aws:resource-explorer-2:*:*:managed-view/AWSManagedViewForSSM*`.
- `resource-groups`: permite a las entidades principales recuperar una lista de grupos de recursos y sus miembros de AWS Resource Groups desde los recursos que pertenecen a un grupo de recursos.
- `securityhub`: permite a las entidades principales recuperar información sobre los recursos del centro de AWS Security Hub en la cuenta actual.
- `states`: permite a las entidades principales iniciar y recuperar información de AWS Step Functions configurada específicamente para que la use Systems Manager.
- `support`: permite a las entidades principales recuperar información sobre comprobaciones y casos en AWS Trusted Advisor.
- `tag`: permite a las entidades principales recuperar información sobre todos los recursos etiquetados, o que estuvieron etiquetados previamente, que se encuentran en una Región de AWS específica de una cuenta.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AmazonSSMServiceRolePolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AmazonSSMAutomationRole

Puede adjuntar la política `AmazonSSMAutomationRole` a las identidades de IAM. Esta política proporciona permisos para que el servicio Automatización de AWS Systems Manager ejecute las actividades definidas en los manuales de procedimientos de Automatización.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `lambda`: permite a las entidades principales invocar funciones de Lambda con nombres que comiencen por “Automatización”. Esto es necesario para que los manuales de procedimientos de Automatización ejecuten funciones de Lambda como parte de su flujo de trabajo.
- `ec2`: permite a las entidades principales realizar diversas operaciones de Amazon EC2, como crear, copiar y anular el registro de imágenes; administrar instantáneas; iniciar, ejecutar, detener y terminar instancias; administrar el estado de las instancias; y crear, eliminar y describir etiquetas. Estos permisos permiten que los manuales de procedimientos de Automatización administren los recursos de Amazon EC2 durante la ejecución.
- `cloudformation`: permite que las entidades principales creen, describan, actualicen y eliminen pilas de AWS CloudFormation. Esto permite que los manuales de procedimientos de Automatización administren la infraestructura como código a través de CloudFormation.
- `ssm`: permite a las entidades principales acceder a todas las acciones de Systems Manager. Este acceso integral es necesario para que los manuales de procedimientos de Automatización interactúen con todas las funciones de Systems Manager.
- `sns`: permite a las entidades principales publicar mensajes en los temas de Amazon SNS con nombres que comiencen por “Automatización”. Esto permite que los manuales de procedimientos de Automatización envíen notificaciones durante la ejecución.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AmazonSSMAutomationRole en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AmazonSSMReadOnlyAccess

Puede adjuntar la política `AmazonSSMReadOnlyAccess` a las identidades de IAM. Esta política otorga acceso de solo lectura a las operaciones de la API de AWS Systems Manager, incluidas `Describe*`, `Get*` y `List*`.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AmazonSSMReadOnlyAccess en la Guía de referencia de políticas administradas de AWS.

Política administrada por AWS: AWSSystemsManagerOpsDataSyncServiceRolePolicy

No puede adjuntar `AWSSystemsManagerOpsDataSyncServiceRolePolicy` a sus entidades de IAM. Esta política está adjunta a un rol vinculado a servicios que permite a Systems Manager realizar acciones en su nombre. Para obtener más información, consulte Uso de roles para crear OpsData y OpsItems para Explorer.

`AWSSystemsManagerOpsDataSyncServiceRolePolicy` permite al rol vinculado a un servicio `AWSServiceRoleForSystemsManagerOpsDataSync` crear y actualizar OpsItems y OpsData desde resultados de AWS Security Hub.

La política permite que Systems Manager complete las siguientes acciones en todos los recursos relacionados (`"Resource": "*"`), excepto cuando se indica lo contrario:

- `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] Las acciones `ssm:GetOpsItem` y `ssm:UpdateOpsItem` solo tienen permisos por la siguiente condición para el servicio de Systems Manager.

```
"Condition": {
    "StringEquals": {
        "aws:ResourceTag/ExplorerSecurityHubOpsItem": "true"
    }
}
```

[2] La acción `ssm:AddTagsToResource` solo tiene permisos para los siguientes recursos.

```
arn:aws:ssm:*:*:opsitem/*
```

[3] Las acciones `ssm:UpdateServiceSetting` y `ssm:GetServiceSetting` solo tienen permisos para los siguientes recursos.

```
arn:aws:ssm:*:*:servicesetting/ssm/opsitem/*
arn:aws:ssm:*:*:servicesetting/ssm/opsdata/*
```

[4] Las acciones `securityhub:BatchUpdateFindings` son permisos denegados por la siguiente condición para el servicio de Systems Manager.

```
{
   "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
     }
    }
```

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSSystemsManagerOpsDataSyncServiceRolePolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada por AWS: AmazonSSMManagedEC2InstanceDefaultPolicy

Solo debe adjuntar `AmazonSSMManagedEC2InstanceDefaultPolicy` a roles de IAM para las instancias de Amazon EC2 para las que desee tener permiso para usar la funcionalidad de Systems Manager. No debe asignar esta función a otras entidades de IAM, como los usuarios y los grupos de IAM, ni a roles de IAM que sirvan para otros fines. Para obtener más información, consulte Administración automática de instancias EC2 con la configuración de administración de hosts predeterminada.

Esta política otorga permisos que permiten a SSM Agent en su instancia de Amazon EC2 comunicarse con el servicio Systems Manager en la nube para realizar diversas tareas. También

otorga permisos para los dos servicios que proporcionan tokens de autorización para garantizar que las operaciones se realicen en la instancia correcta.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite que las entidades principales recuperen documentos, ejecuten comandos con Run Command, establezcan sesiones con Session Manager, recopilen un inventario de la instancia, y analicen las revisiones y el cumplimiento de las mismas mediante Patch Manager.
- `ssmmessages`: permite a las entidades principale acceder, para cada instancia, a un token de autorización personalizado creado por Amazon Message Gateway Service. Systems Manager valida el token de autorización personalizado cotejándolo con el Nombre de recurso de Amazon (ARN) de la instancia, proporcionado en la operación de la API. Este acceso es necesario para garantizar que SSM Agent realice las operaciones de la API en la instancia correcta.
- `ec2messages`: permite a las entidades principale acceder, para cada instancia, a un token de autorización personalizado creado por Amazon Message Delivery Service. Systems Manager valida el token de autorización personalizado cotejándolo con el Nombre de recurso de Amazon (ARN) de la instancia, proporcionado en la operación de la API. Este acceso es necesario para garantizar que SSM Agent realice las operaciones de la API en la instancia correcta.

Para obtener información relacionada con los puntos de conexión de `ssmmessages` y de `ec2messages`, incluidas las diferencias entre ambos, consulte Operaciones de la API relacionadas con el agente (puntos de conexión de `ssmmessages` y `ec2messages`).

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AmazonSSMManagedEC2InstanceDefaultPolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: SSMQuickSetupRolePolicy

No se puede adjuntar SSMQuickSetupRolePolicy a las entidades de IAM. Esta política está adjunta a un rol vinculado a servicios que permite a Systems Manager realizar acciones en su nombre. Para obtener más información, consulte Uso de roles para mantener el estado y la consistencia de los recursos aprovisionados de Quick Setup.

Esta política otorga permisos de solo lectura que permiten a Systems Manager comprobar el estado de la configuración, garantizar el uso coherente de los parámetros y los recursos aprovisionados,

además de corregir los recursos cuando se detecta una desviación. También concede permisos administrativos para crear un rol vinculado al servicio.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite a las entidades principales leer sincronizaciones de datos de recursos de información y documentos de SSM en Systems Manager y en las cuentas de administrador delegado. Esto es necesario para que la Quick Setup pueda determinar el estado en el que deben estar configurados los recursos.
- `organizations`: permite a las entidades principales leer información sobre las cuentas de los miembros que pertenecen a una organización, tal como se configuró en AWS Organizations. Esto es necesario para que Quick Setup pueda identificar todas las cuentas de una organización en las que se van a realizar comprobaciones de estado de los recursos.
- `cloudformation`: permite a las entidades principales leer la información de AWS CloudFormation. Esto es necesario para que Quick Setup pueda recopilar datos sobre las pilas de AWS CloudFormation que se utilizan para administrar el estado de los recursos y las operaciones de los conjuntos de pilas de CloudFormation.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte SSMQuickSetupRolePolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupDeploymentRolePolicy

La política administrada de `AWSQuickSetupDeploymentRolePolicy` admite varios tipos de configuración de Quick Setup. Estos tipos de configuración crean roles y automatizaciones de IAM que configuran servicios y características de Amazon Web Services utilizados con frecuencia, mediante las prácticas recomendadas.

Puede adjuntar `AWSQuickSetupDeploymentRolePolicy` a sus entidades de IAM.

Esta política concede los permisos administrativos necesarios para crear recursos asociados a las siguientes configuraciones de Quick Setup:

- Instalar la administración de host de Amazon EC2 mediante Quick Setup
- Cree un registrador de configuración de AWS Config mediante Quick Setup
- Implemente el paquete de conformidad de AWS Config mediante Quick Setup

- Configurar DevOps Guru con Quick Setup
- Implemente paquetes de Distributor mediante Quick Setup
- Detenga e inicie las instancias EC2 automáticamente según una programación mediante Quick Setup

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `iam`: permite a las entidades principales gestionar y eliminar los roles de IAM necesarios para las tareas de configuración de automatización y gestionar las políticas de roles de automatización.
- `cloudformation`: permite a las entidades principales crear y administrar conjuntos de pilas.
- `config`: permite a las entidades principales crear, administrar y eliminar paquetes de conformidad.
- `events`: permite a las entidades principales crear, actualizar y eliminar reglas de eventos para acciones programadas.
- `resource-groups`: permite a las entidades principales recuperar las consultas de recursos asociadas a grupos de recursos a los que se dirigen las configuraciones de Quick Setup.
- `ssm`: permite a las entidades principales crear manuales de procedimientos de automatización y asociaciones que apliquen las configuraciones de Quick Setup.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupDeploymentRolePolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupPatchPolicyDeploymentRolePolicy

La política administrada de `AWSQuickSetupPatchPolicyDeploymentRolePolicy` admite el tipo de Cómo configurar las revisiones para las instancias de una organización mediante una política de parches Quick Setup Quick Setup. Este tipo de configuración ayuda a automatizar la aplicación de revisiones en aplicaciones y nodos en una sola cuenta o en toda la organización.

Puede adjuntar `AWSQuickSetupPatchPolicyDeploymentRolePolicy` a sus entidades de IAM. Systems Manager también adjunta esta política a un rol de servicio que permite a Systems Manager realizar acciones en su nombre.

Esta política concede permisos administrativos que permiten que Quick Setup cree recursos asociados a la configuración de la política de revisiones.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `iam`: permite a las entidades principales gestionar y eliminar los roles de IAM necesarios para las tareas de configuración de automatización y gestionar las políticas de roles de automatización.
- `cloudformation`: permite a las entidades principales leer la información de las pilas de AWS CloudFormation y controlar las pilas de AWS CloudFormation que se crearon por Quick Setupmediante conjuntos de pilas de AWS CloudFormation.
- `ssm`: permite a las entidades principales crear, actualizar, leer y eliminar los manuales de procedimientos de automatización necesarios para las tareas de configuración, así como crear, actualizar y eliminar asociaciones de State Manager.

- `resource-groups`: permite a las entidades principales recuperar las consultas de recursos asociadas a grupos de recursos a los que se dirigen las configuraciones de Quick Setup.

- `s3`: permite a las entidades principales enumerar los buckets de Amazon S3 y gestionarlos para almacenar los registros de acceso a las políticas de revisiones.
- `lambda`: permite a las entidades principales gestionar las funciones de corrección de AWS Lambda que mantienen las configuraciones en el estado correcto.
- `logs`: permite a las entidades principales describir y administrar grupos de registros para recursos de configuración de Lambda.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupPatchPolicyDeploymentRolePolicy en la Guía de referencia de políticas administradas de AWS.

Política adminsitrada de AWS: AWSQuickSetupPatchPolicyBaselineAccess

La política administrada de `AWSQuickSetupPatchPolicyBaselineAccess` admite el tipo de Cómo configurar las revisiones para las instancias de una organización mediante una política de parches Quick Setup Quick Setup. Este tipo de configuración ayuda a automatizar la aplicación de revisiones en aplicaciones y nodos en una sola cuenta o en toda la organización.

Puede adjuntar `AWSQuickSetupPatchPolicyBaselineAccess` a sus entidades de IAM. Systems Manager también adjunta esta política a un rol de servicio que permite a Systems Manager realizar acciones en su nombre.

Esta política proporciona permisos de solo lectura para acceder a las líneas de base de revisiones configuradas por un administrador de la Cuenta de AWS actual o de la organización que utiliza Quick Setup. Las líneas de base de revisiones se almacenan en un bucket de Amazon S3 y se pueden utilizar para aplicar revisiones a instancias en una sola cuenta o en toda la organización.

Detalles de los permisos

Esta política incluye el siguiente permiso.

- `s3`: permite a las entidades principales leer las anulaciones de la línea de base de revisiones almacenadas en buckets de Amazon S3.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupPatchPolicyBaselineAccessy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: **`AWSSystemsManagerEnableExplorerExecutionPolicy`**

La política administrada de `AWSSystemsManagerEnableExplorerExecutionPolicy` admite la habilitación de Explorer, una herramienta de AWS Systems Manager.

Puede adjuntar `AWSSystemsManagerEnableExplorerExecutionPolicy` a sus entidades de IAM. Systems Manager también adjunta esta política a un rol de servicio que permite a Systems Manager realizar acciones en su nombre.

Esta política concede permisos administrativos para la habilitación de Explorer. Esto incluye permisos para actualizar la configuración de servicios de Systems Manager relacionados y para crear un rol vinculado al servicio para Systems Manager.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `config`: permite que las entidades principales ayuden a habilitar Explorer mediante el acceso de solo lectura a los detalles del registrador de configuraciones.
- `iam`: permite que las entidades principales ayuden a habilitar Explorer
- `ssm`: permite que las entidades principales inicien un flujo de trabajo de automatización que habilita Explorer.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSSystemsManagerEnableExplorerExecutionPolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: `AWSSystemsManagerEnableConfigRecordingExecutionPolicy`

La política administrada de `AWSSystemsManagerEnableConfigRecordingExecutionPolicy` admite los tipos de configuración de Cree un registrador de configuración de AWS Config mediante Quick Setup Quick Setup. Este tipo de configuración habilita a Quick Setup a realizar un seguimiento y registrar los cambios en los tipos de recursos de AWS que elija para AWS Config. También habilita a Quick Setup para que configure las opciones de entrega y notificación de los datos registrados.

Puede adjuntar `AWSSystemsManagerEnableConfigRecordingExecutionPolicy` a sus entidades de IAM. Systems Manager también adjunta esta política a un rol de servicio que permite a Systems Manager realizar acciones en su nombre.

Esta política concede permisos administrativos que permiten a Quick Setup habilitar y configurar el registro de la configuración de AWS Config.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `s3`: permite que las entidades principales creen y configuren buckets de Amazon S3 para la entrega de grabaciones de configuración.
- `sns`: permite que las entidades principales enumeren y creen temas de Amazon SNS.
- `config`: permite que las entidades principales configuren e inicien el registrador de configuración y ayudar a habilitar Explorer.
- `iam`: permite que las entidades principales creen, obtengan y pasen un rol vinculado al servicio para AWS Config; y además crear un rol vinculado al servicio para Systems Manager; y ayudar a habilitar Explorer.
- `ssm`: permite que las entidades principales inicien un flujo de trabajo de automatización que habilita Explorer.
- `compute-optimizer`: permite que las entidades principales ayuden a habilitar Explorer al proporcionar acceso de solo lectura para determinar si un recurso está inscrito con AWS Compute Optimizer.

- `support`: permite que las entidades principales ayuden a habilitar Explorer al proporcionar acceso de solo lectura para determinar si un recurso está inscrito con AWS Compute Optimizer.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSSystemsManagerEnableConfigRecordingExecutionPolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupDevOpsGuruPermissionsBoundary

> **Note**
>
> Esta política es un límite de permisos. Un límite de permisos establece los permisos máximos que una política basada en la identidad puede conceder a una entidad de IAM. No debe usar ni adjuntar políticas de límites de permisos de Quick Setup por su cuenta. Las políticas de límites de permisos de Quick Setup solo deben adjuntarse a las funciones administradas de Quick Setup. Para obtener más información sobre los límites de los permisos, consulte Límites de permisos para las entidades de IAM en la Guía del usuario de IAM.

La política administrada `AWSQuickSetupDevOpsGuruPermissionsBoundary` admite el tipo Configurar DevOps Guru con Quick Setup. El tipo de configuración habilita Amazon DevOps Guru, con tecnología de machine learning. El servicio DevOps Guru puede ayudar a mejorar el rendimiento operativo y la disponibilidad de la aplicación.

Al crear una configuración de `AWSQuickSetupDevOpsGuruPermissionsBoundary` mediante Quick Setup, el sistema aplica este límite de permisos a los roles de IAM que se crean al implementar la configuración. El límite de permisos limita el alcance de los roles que crea Quick Setup.

Esta política concede permisos administrativos que permiten a Quick Setup habilitar y configurar Amazon DevOps Guru.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `iam`: permite que las entidades principales creen funciones vinculadas a servicios para DevOps Guru y Systems Manager, y enumerar los roles que ayudan a habilitar Explorer.

- `cloudformation`: permite que las entidades principales enumeren y describar las pilas de AWS CloudFormation.
- `sns`: permite que las entidades principales enumeren y creen temas de Amazon SNS.
- `devops-guru`: permite a las entidades principales configuren DevOps Guru y agreguen un canal de notificaciones.
- `config`: permite que las entidades principales ayuden a habilitar Explorer mediante el acceso de solo lectura a los detalles del registrador de configuraciones.
- `ssm`: permite que las entidades principales inicien un flujo de trabajo de automatización que habilite Explorer; y lean y actualicen la configuración del servicio Explorer.
- `compute-optimizer`: permite que las entidades principales ayuden a habilitar Explorer al proporcionar acceso de solo lectura para determinar si un recurso está inscrito con AWS Compute Optimizer.
- `support`: permite que las entidades principales ayuden a habilitar Explorer al proporcionar acceso de solo lectura para determinar si un recurso está inscrito con AWS Compute Optimizer.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupDevOpsGuruPermissionsBoundary en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupDistributorPermissionsBoundary

> **Note**
>
> Esta política es un límite de permisos. Un límite de permisos establece los permisos máximos que una política basada en la identidad puede conceder a una entidad de IAM. No debe usar ni adjuntar políticas de límites de permisos de Quick Setup por su cuenta. Las políticas de límites de permisos de Quick Setup solo deben adjuntarse a las funciones administradas de Quick Setup. Para obtener más información sobre los límites de los permisos, consulte Límites de permisos para las entidades de IAM en la Guía del usuario de IAM.

La política administrada de `AWSQuickSetupDistributorPermissionsBoundary` admite los tipos de configuración de Implemente paquetes de Distributor mediante Quick Setup Quick Setup. El tipo de configuración permite la distribución de paquetes de software, como agentes, a sus instancias de Amazon Elastic Compute Cloud (Amazon EC2) mediante Distribuidor, una herramienta de AWS Systems Manager.

Al crear una configuración de `AWSQuickSetupDistributorPermissionsBoundary` mediante Quick Setup, el sistema aplica este límite de permisos a los roles de IAM que se crean al implementar la configuración. El límite de permisos limita el alcance de los roles que crea Quick Setup.

Esta política concede permisos administrativos que permiten que Quick Setup habilite la distribución de paquetes de software, como agentes, a sus instancias de Amazon EC2 mediante Distributor.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `iam`: permite que las entidades principales obtengan y pasen el rolde automatización de Distributor; que creen, lean, actualicen y elimien el rol de instancia predeterminada; transfieran el rol de instancia predeterminada a Amazon EC2 y Systems Manager; adjunten políticas de administración de instancias a los roles de instancia; creen un rol vinculado al servicio para Systems Manager; agreguen el rol de instancia predeterminada a los perfiles de instancia; lean información sobre los roles de IAM y los perfiles de instancia; y creen el perfil de instancia predeterminada.
- `ec2`: permite que las entidades principales asocien el perfil de instancia predeterminada a las instancias EC2 y ayuden a habilitarExplorer.
- `ssm`: permite que las entidades principales incien los flujos de trabajo de automatización que configuran las instancias e instalan paquetes; y ayudan a iniciar el flujo de trabajo de automatización que habilita Explorer; y leen y actualizan la configuración de servicio de Explorer.
- `config`: permite que las entidades principales ayuden a habilitar Explorer mediante el acceso de solo lectura a los detalles del registrador de configuraciones.
- `compute-optimizer`: permite que las entidades principales ayuden a habilitar Explorer al proporcionar acceso de solo lectura para determinar si un recurso está inscrito con AWS Compute Optimizer.
- `support`: permite que las entidades principales ayuden a habilitar Explorer al proporcionar acceso de solo lectura para determinar si un recurso está inscrito con AWS Compute Optimizer.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupDistributorPermissionsBoundary en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupSSMHostMgmtPermissionsBoundary

> **Note**
>
> Esta política es un límite de permisos. Un límite de permisos establece los permisos máximos que una política basada en la identidad puede conceder a una entidad de IAM. No debe usar ni adjuntar políticas de límites de permisos de Quick Setup por su cuenta. Las políticas de límites de permisos de Quick Setup solo deben adjuntarse a las funciones administradas de Quick Setup. Para obtener más información sobre los límites de los permisos, consulte Límites de permisos para las entidades de IAM en la Guía del usuario de IAM.

La política administrada de `AWSQuickSetupSSMHostMgmtPermissionsBoundary` admite los tipos de configuración de Instalar la administración de host de Amazon EC2 mediante Quick Setup Quick Setup. Este tipo de configuración ajusta los roles de IAM y habilita las herramientas de Systems Manager más utilizadas para administrar de forma segura las instancias de Amazon EC2.

Al crear una configuración de `AWSQuickSetupSSMHostMgmtPermissionsBoundary` mediante Quick Setup, el sistema aplica este límite de permisos a los roles de IAM que se crean al implementar la configuración. El límite de permisos limita el alcance de los roles que crea Quick Setup.

Esta política concede permisos administrativos que permiten que Quick Setup habilite y configure las herramientas de Systems Manager necesarias para administrar de forma segura las instancias de EC2.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `iam`: permite que las entidades principales obtengan y transfieran el rol de servicio para la automatización. Permite que las entidades principales creen, lean, actualicen y elimien el rol de instancia predeterminada; transfieran el rol de instancia predeterminada a Amazon EC2 y Systems Manager; adjunten políticas de administración de instancias a los roles de instancia; creen un rol vinculado al servicio para Systems Manager; agreguen el rol de instancia predeterminada a los perfiles de instancia; lean información sobre los roles de IAM y los perfiles de instancia; y creen el perfil de instancia predeterminada.
- `ec2`: permite que las entidades principales asocien y cancelen la asociación del perfil de instancia predeterminada a las instancias EC2.

- `ssm`: permite que las entidades principales inicien flujos de trabajo de Automatización que habilitan Explorer; lean y actualicen la configuración de servicio de Explorer; configuren instancias y habiliten las herramientas de Systems Manager en las instancias.
- `compute-optimizer`: permite que las entidades principales ayuden a habilitar Explorer al proporcionar acceso de solo lectura para determinar si un recurso está inscrito con AWS Compute Optimizer.
- `support`: permite que las entidades principales ayuden a habilitar Explorer al proporcionar acceso de solo lectura para determinar si un recurso está inscrito con AWS Compute Optimizer.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupSSMHostMgmtPermissionsBoundary en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupPatchPolicyPermissionsBoundary

> **Note**
>
> Esta política es un límite de permisos. Un límite de permisos establece los permisos máximos que una política basada en la identidad puede conceder a una entidad de IAM. No debe usar ni adjuntar políticas de límites de permisos de Quick Setup por su cuenta. Las políticas de límites de permisos de Quick Setup solo deben adjuntarse a las funciones administradas de Quick Setup. Para obtener más información sobre los límites de los permisos, consulte Límites de permisos para las entidades de IAM en la Guía del usuario de IAM.

La política administrada de `AWSQuickSetupPatchPolicyPermissionsBoundary` admite el tipo de Cómo configurar las revisiones para las instancias de una organización mediante una política de parches Quick Setup Quick Setup. Este tipo de configuración ayuda a automatizar la aplicación de revisiones en aplicaciones y nodos en una sola cuenta o en toda la organización.

Al crear una configuración de `AWSQuickSetupPatchPolicyPermissionsBoundary` mediante Quick Setup, el sistema aplica este límite de permisos a los roles de IAM que se crean al implementar la configuración. El límite de permisos limita el alcance de los roles que crea Quick Setup.

Esta política concede permisos administrativos que permiten que Quick Setup habilite y configure políticas de revisiones en Patch Manager, una herramienta de AWS Systems Manager.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `iam`: permite que las entidades principales obtengan el rol de automatización de Patch Manager; pasen las funciones de automatización a las operaciones de aplicación de revisiones de Patch Manager; creen el rol de instancia predeterminada, `AmazonSSMRoleForInstancesQuickSetup`; pasen el rol de instancia predeterminada a Amazon EC2 y Systems Manager; adjunten políticas gestionadas de AWS seleccionadas al rol de instancia; creen un rol vinculado al servicio para Systems Manager; agreguen el rol de instancia predeterminada a los perfiles de instancia; lean información sobre los perfiles y roles de las instancias; creen un perfil de instancia predeterminado; y etiqueten funciones que tienen permisos de lectura de anulaciones de línea de base de revisiones.
- `ssm`: permite que las entidades principales actualicen el rol de instancia que gestiona Systems Manager; gestionen las asociaciones creadas por las políticas de revisiones de Patch Manager creadas enQuick Setup; etiqueten las instancias a las que se dirige una configuración de política de revisiones; lean información sobre las instancias y el estado de las revisiones; inicien los flujos de trabajo de automatización que configuran, habilitan y corrigen las revisiones de instancias; inicien los flujos de trabajo de automatización que habilitan Explorer; ayuden a habilitar Explorer; y lean y actualicen la configuración del servicio de Explorer.
- `ec2`: permite que las entidades principales asocien y desasocien el perfil de instancia predeterminado con las instancias de EC2; etiqueten las instancias a las que se dirige una configuración de política de revisiones; etiqueten las instancias a las que se dirige una configuración de política de revisiones; y ayuden a habilitar Explorer.
- `s3`: permite que las entidades principales creen y configuren buckets de S3 para almacenar las anulaciones de línea de base de revisiones.
- `lambda`: permite que las entidades principales invoquen funciones AWS Lambda que configuran la aplicación de revisiones y realizan operaciones de limpieza una vez eliminada la configuración de una política de revisiones de Quick Setup.
- `logs`: permite que las entidades principales configuren el registro para las funciones AWS Lambda de Quick Setup Patch Manager.
- `config`: permite que las entidades principales ayuden a habilitar Explorer mediante el acceso de solo lectura a los detalles del registrador de configuraciones.
- `compute-optimizer`: permite que las entidades principales ayuden a habilitar Explorer al proporcionar acceso de solo lectura para determinar si un recurso está inscrito con AWS Compute Optimizer.

- `support`: permite que las entidades principales ayuden a habilitar Explorer al proporcionar acceso de solo lectura para determinar si un recurso está inscrito con AWS Compute Optimizer.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupPatchPolicyPermissionsBoundary en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupSchedulerPermissionsBoundary

> **Note**
>
> Esta política es un límite de permisos. Un límite de permisos establece los permisos máximos que una política basada en la identidad puede conceder a una entidad de IAM. No debe usar ni adjuntar políticas de límites de permisos de Quick Setup por su cuenta. Las políticas de límites de permisos de Quick Setup solo deben adjuntarse a las funciones administradas de Quick Setup. Para obtener más información sobre los límites de los permisos, consulte Límites de permisos para las entidades de IAM en la Guía del usuario de IAM.

La política administrada de `AWSQuickSetupSchedulerPermissionsBoundary` admite los tipos de configuración de Detenga e inicie las instancias EC2 automáticamente según una programación mediante Quick Setup Quick Setup. Este tipo de configuración le permite detener e iniciar las instancias de EC2 y otros recursos en los momentos que especifique.

Al crear una configuración de `AWSQuickSetupSchedulerPermissionsBoundary` mediante Quick Setup, el sistema aplica este límite de permisos a los roles de IAM que se crean al implementar la configuración. El límite de permisos limita el alcance de los roles que crea Quick Setup.

Esta política concede permisos administrativos que permiten a Quick Setup habilitar y configurar operaciones programadas en instancias EC2 y otros recursos.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `iam`: permite que las entidades principales recuperen y transfieran roles para las acciones de automatización de la administración de instancias; administren, pasen y adjunten roles de instancia predeterminadas para la administración de instancias de EC2; creen perfiles de instancia

predeterminados; agreguen roles de instancia predeterminados a los perfiles de instancia; creen un rol vinculado a un servicio para Systems Manager; lean información sobre los roles de IAM y los perfiles de instancia; asocien un perfil de instancia predeterminado a las instancias de EC2; e inicien flujos de trabajo de Automatización para configurar las instancias y habilitar las herramientas de Systems Manager.

- `ssm`: permite que las entidades principales inicien los flujos de trabajo de automatización que habilitan Explorer; y lean y actualicen la configuración de servicio de Explorer.
- ec2: permite que las entidades principales localicen las instancias de destino e iniciarlas y detenerlas según una programación.
- `config`: permite que las entidades principales ayuden a habilitar Explorer mediante el acceso de solo lectura a los detalles del registrador de configuraciones.
- `compute-optimizer`: permite que las entidades principales ayuden a habilitar Explorer al proporcionar acceso de solo lectura para determinar si un recurso está inscrito con AWS Compute Optimizer.
- `support`: permite que las entidades principales ayuden a habilitar Explorer al proporcionar acceso de solo lectura a las verificaciónes de AWS Trusted Advisor de una cuenta.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupSchedulerPermissionsBoundary en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupCFGCPacksPermissionsBoundary

> **Note**
>
> Esta política es un límite de permisos. Un límite de permisos establece los permisos máximos que una política basada en la identidad puede conceder a una entidad de IAM. No debe usar ni adjuntar políticas de límites de permisos de Quick Setup por su cuenta. Las políticas de límites de permisos de Quick Setup solo deben adjuntarse a las funciones administradas de Quick Setup. Para obtener más información sobre los límites de los permisos, consulte Límites de permisos para las entidades de IAM en la Guía del usuario de IAM.

La política administrada de `AWSQuickSetupCFGCPacksPermissionsBoundary` admite el tipo de configuración Implemente el paquete de conformidad de AWS Config mediante Quick Setup Quick Setup. Este tipo de configuración implementa paquetes de conformidad AWS Config. Los

paquetes de conformidad son conjunto de reglas de AWS Config y acciones correctivas que se puedan implementar como una sola entidad.

Al crear una configuración de `AWSQuickSetupCFGCPacksPermissionsBoundary` mediante Quick Setup, el sistema aplica este límite de permisos a los roles de IAM que se crean al implementar la configuración. El límite de permisos limita el alcance de los roles que crea Quick Setup.

Esta política concede permisos administrativos que permiten a Quick Setup implementar paquetes de conformidad de AWS Config.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `iam`: permite que las entidades principales creen, obtengan y pasen un rol vinculado a un servicio para AWS Config.
- `sns`: permite que las entidades principales enumeren las aplicaciones de la plataforma en Amazon SNS.
- `config`: permite que las entidades principales implementen paquetes de conformidad de AWS Config, obtengan el estado de los paquetes de conformidad y obtengan información sobre los registradores de configuración.
- `ssm`: permite que las entidades principales obtengan información sobre los documentos SSM y los flujos de trabajo de automatización; obtengan información sobre las etiquetas de los recursos; y obtengan información sobre la configuración del servicio y la actualicen.
- `compute-optimizer`: permite que las entidades principales obtengan el estado de suscripción de una cuenta.
- `support`: permite que las entidades principales obtengan información sobre las verificaciones de AWS Trusted Advisor.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupCFGCPacksPermissionsBoundary en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWS-SSM-DiagnosisAutomation-AdministrationRolePolicy

La política `AWS-SSM-DiagnosisAutomation-AdministrationRolePolicy` otorga permisos para diganosticar problemas con los nodos que interactúan con los servicios de Systems Manager al iniciar los flujos de trabajo de Automatización en las cuentas y regiones donde se administren los nodos.

Puede adjuntar `AWS-SSM-DiagnosisAutomation-AdministrationRolePolicy` a sus entidades de IAM. Systems Manager también adjunta esta política a un rol de servicio que permite a Systems Manager realizar acciones de diagnóstico en su nombre.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite a las entidades principales ejecutar manuales de procedimientos de Automatización que diagnostican los problemas de los nodos, acceden al estado de ejecución de un flujo de trabajo y recuperan información sobre la ejecución de automatización. La política otorga permisos para describir las ejecuciones de automatización, describir las ejecuciones de los pasos de automatización, obtener detalles sobre la ejecución de la automatización e iniciar las ejecuciones de automatización para los documentos relacionados con el diagnóstico.
- `kms`: permite a las entidades principales utilizar claves de AWS Key Management Service especificadas por el cliente para el descifrado y la generación de claves de datos, al acceder a objetos cifrados en los buckets de Amazon S3 que se utilizan para las operaciones de diagnóstico. Estos permisos están restringidos a las claves etiquetadas con `SystemsManagerManaged` y utilizadas a través del servicio Amazon S3 con requisitos de contexto de cifrado específicos.
- `sts`: permite a las entidades principales asumir roles de ejecución de diagnósticos para ejecutar los manuales de procedimientos de Automatización en la misma cuenta. Este permiso está restringido a los roles con el patrón de nomenclatura `AWS-SSM-DiagnosisExecutionRole` e incluye una condición para garantizar que la cuenta de recursos coincida con la cuenta de la entidad principal.
- `iam`: permite a las entidades principales transferir el rol de administración de diagnósticos a Systems Manager para ejecutar los manuales de procedimientos de Automatización. Este permiso está restringido a las funciones con el patrón de nomenclatura `AWS-SSM-DiagnosisAdminRole` y solo se puede transferir al servicio Systems Manager.

- `s3`: permite a las entidades principales acceder, leer, escribir y eliminar objetos en los buckets de Amazon S3 que se utilizan para las operaciones de diagnóstico. Estos permisos están restringidos a los buckets con el mismo patrón de nomenclatura `do-not-delete-ssm-diagnosis-` e incluyen condiciones para garantizar que las operaciones se realicen dentro de la misma cuenta.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWS-SSM-DiagnosisAutomation-AdministrationRolePolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWS-SSM-DiagnosisAutomation-ExecutionRolePolicy

La política administrada `AWS-SSM-DiagnosisAutomation-ExecutionRolePolicy` otorga permisos administrativos para ejecutar manuales de procedimientos de Automatización en una región y Cuenta de AWS específicas para diagnosticar problemas con los nodos administrados que interactúen con los servicios de Systems Manager.

Puede adjuntar `AWS-SSM-DiagnosisAutomation-ExecutionRolePolicy` a sus entidades de IAM. Systems Manager también adjunta esta política a un rol de servicio que permite a Systems Manager realizar acciones en su nombre.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ec2`: permite a las entidades principales describir los recursos de Amazon EC2 y Amazon VPC, así como sus configuraciones, para diagnosticar problemas con los servicios de Systems Manager. Esto incluye permisos para describir las VPC, los atributos de la VPC, los puntos de conexión de VPC, las subredes, los grupos de seguridad, las instancias y las puertas de enlace de Internet.
- `ssm`: permite a las entidades principales ejecutar manuales de procedimientos de Automatización específicos de diagnóstico y acceder al estado del flujo de trabajo de automatización y a los metadatos de ejecución. Esto incluye permisos para describir las ejecuciones de los pasos de automatización, describir la información de las instancias, describir las ejecuciones de automatización, obtener detalles de la ejecución de la automatización e iniciar las ejecuciones de automatización para documentos de diagnóstico de EC2 específicos no administrados por AWS.
- `kms`: permite a las entidades principales utilizar claves de AWS Key Management Service especificadas por el cliente para el descifrado y la generación de claves de datos, al acceder a objetos cifrados en los buckets de Amazon S3 que se utilizan para las operaciones de diagnóstico. Estos permisos están restringidos a las claves etiquetadas con `SystemsManagerManaged` y

utilizadas a través del servicio Amazon S3 con requisitos de contexto de cifrado específicos para buckets de diagnóstico.

- `iam`: permite a las entidades principales transferir el rol de ejecución de diagnósticos a Systems Manager para ejecutar documentos de Automatización. Este permiso está restringido a los roles con el patrón de nomenclatura `AWS-SSM-DiagnosisExecutionRole` y solo se puede transferir al servicio Systems Manager.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWS-SSM-DiagnosisAutomation-ExecutionRolePolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWS-SSM-RemediationAutomation-AdministrationRolePolicy

La política `AWS-SSM-RemediationAutomation-AdministrationRolePolicy` proporciona permisos para solucionar problemas con los servicios de Systems Manager mediante la ejecución de las actividades definidas en los documentos de Automatización, que se utilizan principalmente para ejecutar los documentos de Automatización. Esta política permite iniciar flujos de trabajo de Automatización en cuentas y regiones donde se administran los nodos para abordar los problemas de conectividad y configuración.

Puede adjuntar `AWS-SSM-RemediationAutomation-AdministrationRolePolicy` a sus entidades de IAM. Systems Manager también adjunta esta política a un rol de servicio que permite a Systems Manager realizar acciones de remediación en su nombre.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite a las entidades principales ejecutar manuales de procedimientos de Automatización específicos que solucionan los problemas de los nodos, acceden al estado de ejecución de un flujo de trabajo y recuperan la información de ejecución de la automatización. La política otorga permisos para describir las ejecuciones de automatización, describir las ejecuciones de los pasos de automatización, obtener detalles sobre las ejecuciones de la automatización e iniciar las ejecuciones de automatización para los documentos relacionados con la corrección.
- `kms`: permite a las entidades principales utilizar claves de AWS Key Management Service especificadas por el cliente para el descifrado y la generación de claves de datos, al acceder a objetos cifrados en los buckets de Amazon S3 que se utilizan para las

operaciones de remediación. Estos permisos están restringidos a las claves etiquetadas con `SystemsManagerManaged` y utilizadas a través del servicio Amazon S3 con requisitos de contexto de cifrado específicos.

- `sts`: permite a las entidades principales asumir roles de ejecución de remediación para ejecutar los manuales de procedimientos de Automatización en la misma cuenta. Este permiso está restringido a los roles con el patrón de nomenclatura `AWS-SSM-RemediationExecutionRole` e incluye una condición para garantizar que la cuenta de recursos coincida con la cuenta de la entidad principal.
- `iam`: permite a las entidades principales transferir el rol de administración de remediación a Systems Manager para ejecutar los manuales de procedimientos de Automatización. Este permiso está restringido a los roles con el patrón de nomenclatura `AWS-SSM-RemediationAdminRole` y solo se puede transferir al servicio Systems Manager.
- `s3`: permite a las entidades principales acceder, leer, escribir y eliminar objetos en los buckets de Amazon S3 que se utilizan para las operaciones de remediación. Estos permisos están restringidos a los buckets con el mismo patrón de nomenclatura `do-not-delete-ssm-diagnosis-` e incluyen condiciones para garantizar que las operaciones se realicen dentro de la misma cuenta.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWS-SSM-RemediationAutomation-AdministrationRolePolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWS-SSM-RemediationAutomation-ExecutionRolePolicy

La política administrada `AWS-SSM-RemediationAutomation-ExecutionRolePolicy` otorga permisos administrativos para ejecutar manuales de procedimientos de Automatización en una región y cuenta de destino específicas para resolver problemas de redes y conectividad con los nodos administrados que interactúen con los servicios de Systems Manager. Esta política permite las actividades de corrección definidas en los documentos de Automatización, que se utilizan principalmente para ejecutar los documentos de Automatización a fin de abordar problemas de conectividad y configuración.

También puede asociar la política a sus entidades de IAM. Systems Manager también adjunta esta política a un rol de servicio que permite a Systems Manager realizar acciones de remediación en su nombre.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite a las entidades principales recuperar información sobre las ejecuciones de Automatización y sus ejecuciones escalonadas, e iniciar manuales de procedimientos de Automatización de remediación específicos, incluidos los documentos `AWS-OrchestrateUnmanagedEC2Actions` y `AWS-RemediateSSMAgent`. La política otorga permisos para describir las ejecuciones de automatización, describir las ejecuciones de los pasos de automatización, obtener detalles sobre las ejecuciones de la automatización e iniciar las ejecuciones de automatización para los documentos relacionados con la corrección.
- `ec2`: permite a las entidades principales describir y modificar los recursos de red de Amazon VPC para solucionar problemas de conectividad. Esto incluye:
  - Descripción de los atributos de Amazon VPC, subredes, puntos de conexión de Amazon VPC y grupos de seguridad.
  - Creación de puntos de conexión de Amazon VPC para los servicios de Systems Manager (`ssm`, `ssmmessages`, y `ec2messages`) con las etiquetas necesarias.
  - Modificación de los atributos de Amazon VPC para activar la compatibilidad con DNS y los nombres de host.
  - Creación y administración de los grupos de seguridad con etiquetas específicas para el acceso a los puntos de conexión de Amazon VPC.
  - Autorización y revocación de las reglas de los grupos de seguridad para el acceso a HTTPS con las etiquetas adecuadas.
  - Creación de etiquetas en los puntos de conexión de Amazon VPC, los grupos de seguridad y las reglas de los grupos de seguridad durante la creación de los recursos.
- `kms`: permite a las entidades principales utilizar claves de AWS Key Management Service especificadas por el cliente para el descifrado y la generación de claves de datos, al acceder a objetos cifrados en los buckets de Amazon S3 que se utilizan para las operaciones de remediación. Estos permisos están restringidos a las claves etiquetadas con `SystemsManagerManaged` y utilizadas a través del servicio Amazon S3 con requisitos de contexto de cifrado específicos.
- `iam`: permite a las entidades principales transferir el rol de ejecución de remediación Systems Manager para ejecutar manuales de procedimientos de Automatización. Este permiso está restringido a los roles con el patrón de nomenclatura `AWS-SSM-RemediationExecutionRole` y solo se puede transferir al servicio Systems Manager.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWS-SSM-RemediationAutomation-ExecutionRolePolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupSSMManageResourcesExecutionPolicy

Esta política otorga permisos que permiten a Quick Setup ejecutar el manual de procedimientos de Automatización `AWSQuickSetupType-SSM-SetupResources`. Este manual de procedimientos crea roles de IAM para las asociaciones de Quick Setup, que a su vez se crean mediante una implementación de `AWSQuickSetupType-SSM`. También concede permisos para limpiar un bucket de Amazon S3 asociado durante una operación de eliminación de Quick Setup.

Puede adjuntar la política a sus entidades de IAM. Systems Manager también adjunta esta política a un rol de servicio que permite a Systems Manager realizar acciones en su nombre.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `iam`: permite a las entidades principales enumerar y administrar los roles de IAM para utilizarlos con las operaciones del Explorador de Systems Manager de Quick Setup y ver, adjuntar y separar las políticas de IAM para utilizarlas con Quick Setup y el Explorador de Systems Manager. Estos permisos son necesarios para que Quick Setup pueda crear los roles necesarios para algunas de sus operaciones de configuración.
- `s3`: permite a las entidades principales recuperar información sobre los objetos de los buckets de Amazon S3 de la cuenta de la entidad principal que se utilizan específicamente en las operaciones de configuración de Quick Setup. También les permite eliminarlos. Esto es necesario para poder eliminar los objetos de S3 que ya no se necesitan después de la configuración.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupSSMManageResourcesExecutionPolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupSSMLifecycleManagementExecutionPolicy

La política `AWSQuickSetupSSMLifecycleManagementExecutionPolicy` otorga permisos administrativos que permite a Quick Setup ejecutar un recurso personalizado de AWS

CloudFormation en los eventos del ciclo de vida durante la implementación de Quick Setup en Systems Manager.

Puede adjuntar esta política a sus entidades de IAM. Systems Manager también adjunta esta política a un rol de servicio que permite a Systems Manager realizar acciones en su nombre.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite a las entidades principales obtener información sobre las ejecuciones de automatización e iniciarlas para configurar determinadas operaciones de Quick Setup.
- `iam`: permite a las entidades principales transferir roles de IAM para configurar determinados recursos de Quick Setup.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupSSMLifecycleManagementExecutionPolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupSSMDeploymentRolePolicy

La política administrada `AWSQuickSetupSSMDeploymentRolePolicy` concede permisos administrativos para que Quick Setup pueda crear recursos que se utilizan durante el proceso de incorporación de Systems Manager.

Aunque puede adjuntar esta política a sus entidades de IAM manualmente, no se recomienda. Quick Setup crea entidades que adjuntan esta política a un rol de servicio que permite a Systems Manager realizar acciones en su nombre.

Esta política no está relacionada con la política `SSMQuickSetupRolePolicy`, que se utiliza para conceder permisos para el rol `AWSServiceRoleForSSMQuickSetup` vinculado al servicio.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite a las entidades principales administrar las asociaciones de determinados recursos que se crean mediante plantillas de AWS CloudFormation y un conjunto específico de documentos de SSM; administrar los roles y sus políticas para el diagnóstico y la corrección de los nodos administrados mediante plantillas de AWS CloudFormation; y adjuntar y eliminar políticas de los eventos del ciclo de vida de Quick Setup.

- `iam`: permite a las entidades principales etiquetar roles y transferir permisos de roles para los servicios de Systems Manager y Lambda. Además, permite transferir permisos de roles para operaciones de diagnóstico.
- `lambda`: permite a las entidades principales etiquetar y administrar las funciones durante todo el ciclo de vida de Quick Setup de la cuenta principal mediante plantillas de AWS CloudFormation.
- `cloudformation`: permite a las entidades principales leer la información de AWS CloudFormation. Esto es necesario para que Quick Setup pueda recopilar datos sobre las pilas de AWS CloudFormation que se utilizan para administrar el estado de los recursos y las operaciones de los conjuntos de pilas de CloudFormation.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupSSMDeploymentRolePolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupSSMDeploymentS3BucketRolePolicy

La política `AWSQuickSetupSSMDeploymentS3BucketRolePolicy` otorga permisos para enumerar todos los buckets de S3 de una cuenta, así como para administrar y recuperar información sobre buckets específicos de la cuenta de la entidad principal que se administran mediante plantillas de AWS CloudFormation.

Puede adjuntar `AWSQuickSetupSSMDeploymentS3BucketRolePolicy` a sus entidades de IAM. Systems Manager también adjunta esta política a un rol de servicio que permite a Systems Manager realizar acciones en su nombre.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `s3`: permite a las entidades principales enumerar todos los buckets de S3 de una cuenta, así como administrar y recuperar información sobre buckets específicos de la cuenta de la entidad principal que se administran con plantillas de AWS CloudFormation.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupSSMDeploymentS3BucketRolePolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupEnableDHMCExecutionPolicy

Esta política otorga permisos administrativos que permiten a las entidades principales ejecutar el manual de procedimientos de Automatización `AWSQuickSetupType-EnableDHMC`, que habilita la configuración de administración de hosts predeterminada. La configuración de administración de hosts predeterminada permite que Systems Manager administre las instancias de Amazon EC2 de forma automática como instancias administradas. Una instancia administrada es una instancia EC2 configurada para usarla con Systems Manager. Esta política también concede permisos para crear roles de IAM que se especifican en la configuración del servicio de Systems Manager como roles predeterminados de SSM Agent.

Puede adjuntar `AWSQuickSetupEnableDHMCExecutionPolicy` a sus entidades de IAM. Systems Manager también adjunta esta política a un rol de servicio que permite a Systems Manager realizar acciones en su nombre.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite a las entidades principales actualizar y obtener información sobre la configuración del servicio de Systems Manager.
- `iam`: permite a las entidades principales crear y recuperar información sobre los roles de IAM para las operaciones de Quick Setup.

Para ver más detalles sobre la política, incluyendo la última versión del documento de política JSON, consulte AWSQuickSetupEnableDHMCExecutionPolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupEnableAREXExecutionPolicy

Esta política concede permisos administrativos que permiten a Systems Manager ejecutar el manual de procedimientos de Automatización `AWSQuickSetupType-EnableAREX`, que permite a Explorador de recursos de AWS el uso con Systems Manager. Resource Explorer permite ver los recursos de su cuenta con una experiencia de búsqueda similar a la de un motor de búsqueda de Internet. La política también otorga permisos para administrar los índices y vistas de Resource Explorer.

Puede adjuntar `AWSQuickSetupEnableAREXExecutionPolicy` a sus entidades de IAM. Systems Manager también adjunta esta política a un rol de servicio que permite a Systems Manager realizar acciones en su nombre.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `iam`: permite que las entidades principales creen un rol vinculado al servicio en el servicio de AWS Identity and Access Management (IAM).
- `resource-explorer-2`: permite a las entidades principales recuperar información sobre vistas e índices de Resource Explorer; crear vistas e índices de Resource Explorer; y cambiar el tipo de índice de los índices que se muestran en Quick Setup.

Para ver más detalles sobre la política, incluyendo la última versión del documento de política JSON, consulte AWSQuickSetupEnableAREXExecutionPolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupManagedInstanceProfileExecutionPolicy

Esta política concede permisos administrativos que permiten a Systems Manager crear un perfil de instancia de IAM predeterminado para la herramienta Quick Setup y adjuntarlo a las instancias de Amazon EC2 que aún no tienen un perfil de instancia adjuntado. La política también permite a Systems Manager adjuntar permisos a los perfiles de instancia existentes. Esto se hace para garantizar que están vigentes los permisos necesarios para que Systems Manager se comunique con SSM Agent en instancias de EC2.

Puede adjuntar `AWSQuickSetupManagedInstanceProfileExecutionPolicy` a sus entidades de IAM. Systems Manager también adjunta esta política a un rol de servicio que permite a Systems Manager realizar acciones en su nombre.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite a las entidades principales iniciar los flujos de trabajo de automatización asociados a los procesos de Quick Setup.

- `ec2`: permite a las entidades principales adjuntar perfiles de instancia de IAM a las instancias de EC2 administradas por Quick Setup.
- `iam`: permite a las entidades principales crear, actualizar y recuperar información sobre los roles de IAM que se utilizan en los procesos de Quick Setup, crear perfiles de instancias de IAM y adjuntar la política administrada `AmazonSSMManagedInstanceCore` a los perfiles de instancia de IAM.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupManagedInstanceProfileExecutionPolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupFullAccess

Esta política otorga permisos administrativos que permiten el acceso total a las acciones y los datos de la API de Quick Setup de AWS Systems Manager en la AWS Management Console y los SDK de AWS, así como un acceso limitado a otros recursos de Servicio de AWS necesarios para las operaciones de Quick Setup.

Puede adjuntar la política de `AWSQuickSetupFullAccess` a las identidades de IAM.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite a las entidades principales habilitar Explorer, realizar operaciones de sincronización de datos de recursos en State Manager; y realizar operaciones con documentos de comandos de SSM y manuales de procedimientos de Automatización.

  Explorer, State Manager, Documentos y Automatización son herramientas de Systems Manager.
- `cloudformation`: permite a las entidades principales realizar las operaciones de AWS CloudFormation necesarias para aprovisionar recursos entre las Regiones de AWS y Cuentas de AWS.
- `ec2`: permite a las entidades principales seleccionar los parámetros necesarios para una configuración determinada y proporcionar validación en la AWS Management Console.
- `iam`: permite a las entidades principales crear los roles de servicio y roles vinculados al servicio necesarios para las operaciones de Quick Setup.
- `organizations`: permite a las entidades principales leer el estado de las cuentas de una organización de AWS Organizations; recuperar la estructura de una organización; habilitar

el acceso de confianza y registrar una cuenta de administrador delegado desde la cuenta de administración.

- `resource-groups`: permite a las entidades principales seleccionar los parámetros necesarios para una configuración determinada y proporcionar validación en la AWS Management Console.
- `s3`: permite a las entidades principales seleccionar los parámetros necesarios para una configuración determinada y proporcionar validación en la AWS Management Console.
- `ssm-quicksetup`: permite a las entidades principales realizar acciones de solo lectura en Quick Setup.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupFullAccess en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWSQuickSetupReadOnlyAccess

Esta política concede permisos de solo lectura que permiten a las entidades principales ver datos e informes de Quick Setup de AWS Systems Manager, incluida la información de otros recursos de Servicio de AWS necesarios para las operaciones de Quick Setup.

Puede adjuntar la política de `AWSQuickSetupReadOnlyAccess` a las identidades de IAM.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite a las entidades principales leer los documentos de comandos de SSM y los manuales de procedimientos de Automatización, así como recuperar el estado de las ejecuciones de las asociaciones de State Manager.
- `cloudformation`: permite a las entidades principales iniciar las operaciones necesarias para recuperar el estado de las implementaciones de AWS CloudFormation.
- `organizations`: permite a las entidades principales leer el estado de las cuentas de una organización de AWS Organizations.
- `ssm-quicksetup`: permite a las entidades principales realizar acciones de solo lectura en Quick Setup.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupReadOnlyAccess en la Guía de referencia de políticas administradas de AWS.

Política administrada por AWS: AWSQuickSetupManageJITNAResourcesExecutionPolicy

La política administrada `AWSQuickSetupManageJITNAResourcesExecutionPolicy` permite que Quick Setup, una herramienta de Systems Manager, configure el acceso a los nodos justo a tiempo.

Puede adjuntar `AWSQuickSetupManageJITNAResourcesExecutionPolicy` a sus entidades de IAM. Systems Manager también adjunta esta política a un rol de servicio que permite a Systems Manager realizar acciones en su nombre.

Con esta política, se conceden los permisos administrativos que avalan que Systems Manager cree recursos asociados al acceso a los nodos justo a tiempo.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite que las entidades principales obtengan y actualicen la configuración del servicio que especifica el proveedor de identidades para el acceso a los nodos justo a tiempo.
- `iam`: permite que las entidades principales creen, etiqueten y obtengan roles, adjunten políticas de roles para las políticas administradas de acceso a los nodos justo a tiempo y creen roles vinculados al servicio para el acceso a los nodos justo a tiempo y las notificaciones.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupManageJITNAResourcesExecutionPolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada por AWS: AWSQuickSetupJITNADeploymentRolePolicy

La política administrada `AWSQuickSetupJITNADeploymentRolePolicy` permite que Quick Setup implemente el tipo de configuración necesario del acceso a los nodos justo a tiempo.

Puede adjuntar `AWSQuickSetupJITNADeploymentRolePolicy` a sus entidades de IAM. Systems Manager también adjunta esta política a un rol de servicio que permite a Systems Manager realizar acciones en su nombre.

Con esta política, se conceden los permisos administrativos que avalan que Systems Manager cree recursos asociados al acceso a los nodos justo a tiempo.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `cloudformation`: permite que las entidades principales creen, actualicen, eliminen y lean pilas de AWS CloudFormation.
- `ssm`: permite que las entidades principales creen, eliminen, actualicen y lean asociaciones de State Manager invocadas por AWS CloudFormation.
- `iam`: permite que las entidades principales creen, eliminen, lean y etiqueten los roles de IAM invocados por AWS CloudFormation.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSQuickSetupJITNADeploymentRolePolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada por AWS: AWSSystemsManagerJustInTimeAccessServicePolicy

La política administrada `AWSSystemsManagerJustInTimeAccessServicePolicy` proporciona permisos a los recursos de AWS administrados o utilizados por la característica de acceso a los nodos justo a tiempo de Systems Manager.

No puede adjuntar `AWSSystemsManagerJustInTimeAccessServicePolicy` a sus entidades de IAM. Esta política está adjunta a un rol vinculado a servicios que permite a Systems Manager realizar acciones en su nombre. Para obtener más información, consulte Cómo utilizar roles para permitir el acceso a los nodos justo a tiempo.

Con esta política, se conceden los permisos administrativos que avalan el acceso a recursos asociados al acceso a los nodos justo a tiempo.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite que las entidades principales creen, describan y actualicen OpsItems, describan sesiones y enumeren etiquetas para nodos administrados.
- `ssm-guiconnect`: permite que las entidades principales enumeren las conexiones.

- `identitystore`: permite obtener los ID de usuarios y grupos, describir usuarios y enumerar la pertenencia a grupos.
- `sso-directory`: permite que las entidades principales describan usuarios y determinen si un usuario es miembro de un grupo.
- `sso`: permite que las entidades principales describan las regiones registradas y enumeren las instancias y las asociaciones de directorios.
- `cloudwatch`: permite que las entidades principales introduzcan datos métricos para el espacio de nombres `AWS/SSM/JustInTimeAccess`.
- `ec2`: permite que las entidades principales describan las etiquetas.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSSystemsManagerJustInTimeAccessServicePolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada por AWS: AWSSystemsManagerJustInTimeAccessTokenPolicy

La política administrada `AWSSystemsManagerJustInTimeAccessTokenPolicy` permite que Systems Manager genere los tokens de acceso utilizados en el acceso a los nodos justo a tiempo.

Puede adjuntar `AWSSystemsManagerJustInTimeAccessTokenPolicy` a sus entidades de IAM.

Con esta política, se conceden los permisos administrativos que permiten que Systems Manager inicie sesiones y genere los tokens de acceso para el acceso a los nodos justo a tiempo.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite que las entidades principales envíen comandos, enumeren invocaciones de comandos e inicien, reanuden y finalicen sesiones de Session Manager.
- `ssm-guiconnect`: permite que las entidades principales inicien, describan y finalicen las conexiones con el protocolo para escritorios remotos (RDP) de Systems Manager GUI Connect.
- `kms`: permite que las entidades principales creen concesiones y generen datos de claves.
- `sso`: permite que las entidades principales enumeren las asociaciones de directorios.
- `identitystore`: permite que las entidades principales describan un usuario.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSSystemsManagerJustInTimeAccessTokenPolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada por AWS: AWSSystemsManagerJustInTimeAccessTokenSessionPolicy

La política administrada `AWSSystemsManagerJustInTimeAccessTokenSessionPolicy` permite que Systems Manager aplique permisos restringidos a un token de acceso a los nodos justo a tiempo.

Puede adjuntar `AWSSystemsManagerJustInTimeAccessTokenSessionPolicy` a sus entidades de IAM.

Con esta política, se conceden los permisos administrativos que avalan que Systems Manager restrinja los permisos para los tokens de acceso a los nodos justo a tiempo.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite que las entidades principales inicien sesiones de Session Manager utilizando el documento `SSM-SessionManagerRunShell`. Además, cuando se llaman primero a través de `ssm-guiconnect`, les permite iniciar sesiones utilizando el documento `AWS-StartPortForwardingSession`, listar invocaciones de comandos y enviar comandos con el documento `AWSSSO-CreateSSOUser`.
- `ssm-guiconnect`: permite que las entidades principales cancelen, obtengan e inicien conexiones en todos los recursos.
- `kms`: permite que las entidades principales creen concesiones y generen claves de datos para claves etiquetadas con `SystemsManagerJustInTimeNodeAccessManaged` cuando se llaman a través de `ssm-guiconnect` mediante un servicio de AWS.
- `sso`: permite que las entidades principales enumeren las asociaciones de directorios cuando se llaman a través de `ssm-guiconnect`.
- `identitystore`: permite que las entidades principales describan un usuario cuando se llaman a través de `ssm-guiconnect`.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSSystemsManagerJustInTimeAccessTokenSessionPolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada por AWS: AWSSystemsManagerJustInTimeNodeAccessRolePropagationPolicy

La política administrada `AWSSystemsManagerJustInTimeNodeAccessRolePropagationPolicy` permite que Systems Manager comparta las políticas de denegación de acceso desde la cuenta de administrador delegado con las cuentas de miembros y replique las políticas en varias Regiones de AWS.

Puede adjuntar `AWSSystemsManagerJustInTimeNodeAccessRolePropagationPolicy` a sus entidades de IAM.

Con esta política, se conceden los permisos administrativos necesarios para que Systems Manager comparta y cree políticas de denegación de acceso. Esto garantiza que las políticas de denegación de acceso se apliquen a todas las cuentas de una organización de AWS Organizations y a las regiones configuradas para el acceso a los nodos justo a tiempo.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `ssm`: permite que las entidades principales administren los documentos y las políticas de recursos de SSM.
- `ssm-quicksetup`: permite que las entidades principales lean los administradores de configuración de Quick Setup.
- `organizations`: permite que las entidades principales enumeren detalles sobre una organización de AWS Organizations y los administradores delegados.
- `ram`: permite que las entidades principales creen, etiqueten y describan recursos compartidos.
- `iam`: permite que las entidades principales describan un rol de servicio.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSSystemsManagerJustInTimeNodeAccessRolePropagationPolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada por AWS: AWSSystemsManagerNotificationsServicePolicy

La política administrada `AWSSystemsManagerNotificationsServicePolicy` permite que Systems Manager envíe notificaciones por correo electrónico para las solicitudes de acceso a los nodos justo a tiempo a los aprobadores de las solicitudes de acceso.

No puede adjuntar `AWSSystemsManagerJustInTimeAccessServicePolicy` a sus entidades de IAM. Esta política está adjunta a un rol vinculado a servicios que permite a Systems Manager realizar acciones en su nombre. Para obtener más información, consulte Cómo utilizar roles para enviar notificaciones de solicitudes de acceso a los nodos justo a tiempo.

Con esta política, se conceden los permisos administrativos que permiten que Systems Manager envíe notificaciones por correo electrónico para las solicitudes de acceso a los nodos justo a tiempo a los aprobadores de las solicitudes de acceso.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `identitystore`: permite que las entidades principales enumeren y describan usuarios y la pertenencia a grupos.
- `sso`: permite que las entidades principales enumeren instancias y directorios, y describan regiones registradas.
- `sso-directory`: permite que las entidades principales describan usuarios y enumeren los miembros de un grupo.
- `iam`: permite que las entidades principales obtengan información acerca de los roles.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWSSystemsManagerNotificationsServicePolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWS-SSM-Automation-DiagnosisBucketPolicy

La política administrada `AWS-SSM-Automation-DiagnosisBucketPolicy` concede permisos para diagnosticar problemas en los nodos que interactúan con los servicios de AWS Systems Manager al permitir el acceso a los buckets de S3 que se utilizan para diagnosticar y solucionar problemas.

Puede adjuntar la política `AWS-SSM-Automation-DiagnosisBucketPolicy` a las identidades de IAM. Systems Manager también adjunta esta política a un rol de IAM que permite a Systems Manager realizar acciones de diagnóstico en su nombre.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `s3`: permite a las entidades principales acceder y escribir objetos en un bucket de Amazon S3.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWS-SSM-Automation-DiagnosisBucketPolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWS-SSM-RemediationAutomation-OperationalAccountAdministrationRolePolicy

La política administrada `AWS-SSM-RemediationAutomation-OperationalAccountAdministrationRolePolicy` concede permisos para que una cuenta operativa diagnostique problemas con los nodos mediante permisos específicos de la organización.

Puede adjuntar `AWS-SSM-RemediationAutomation-OperationalAccountAdministrationRolePolicy` a las identidades de IAM. Systems Manager también adjunta esta política a un rol de IAM que permite a Systems Manager realizar acciones de diagnóstico en su nombre.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `organizations`: permite a las entidades principales enumerar una raíz de la organización y obtener las cuentas de los miembros para determinar las de destino.
- `sts`: permite a las entidades principales asumir roles de ejecución de correcciones para ejecutar los documentos de Automatización de SSM en todas las cuentas y regiones de la misma organización.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWS-SSM-RemediationAutomation-OperationalAccountAdministrationRolePolicy en la Guía de referencia de políticas administradas de AWS.

Política administrada de AWS: AWS-SSM-DiagnosisAutomation-OperationalAccountAdministrationRolePolicy

La política administrada `AWS-SSM-DiagnosisAutomation-OperationalAccountAdministrationRolePolicy` concede permisos para que una cuenta operativa diagnostique problemas con los nodos mediante permisos específicos de la organización.

Puede adjuntar la política `AWS-SSM-DiagnosisAutomation-OperationalAccountAdministrationRolePolicy` a las identidades de IAM. Systems Manager también adjunta esta política a un rol de IAM que permite a Systems Manager realizar acciones de diagnóstico en su nombre.

Detalles de los permisos

Esta política incluye los siguientes permisos.

- `organizations`: permite a las entidades principales enumerar una raíz de la organización y obtener las cuentas de los miembros para determinar las de destino.
- `sts`: permite a las entidades principales asumir roles de ejecución de diagnósticos para ejecutar los documentos de Automatización de SSM en todas las cuentas y regiones de la misma organización.

Para ver más detalles sobre la política, incluida la versión más reciente del documento de política JSON, consulte AWS-SSM-DiagnosisAutomation-OperationalAccountAdministrationRolePolicy en la Guía de referencia de políticas administradas de AWS.

Actualizaciones de Systems Manager para las políticas administradas de AWS

La siguiente tabla muestra los detalles de las actualizaciones de las políticas administradas de AWS para Systems Manager debido a que este servicio comenzó a realizar el seguimiento de estos cambios el 12 de marzo de 2021. Para obtener información sobre otras políticas administradas para el servicio Systems Manager, consulte Políticas administradas adicionales para Systems Manager más adelante en este tema. Para obtener alertas automáticas sobre cambios en esta página, suscríbase a la fuente RSS en la página de Systems Manager Historial del documento.

Cambio	Descripción	Fecha
AmazonSSMAutomationRole: actualización de la documentación	Systems Manager agregó documentación exhaustiva para la política `AmazonSSMAutomationRole` existente, que proporciona permisos para que el servicio Automatización de Systems Manager ejecute las actividades definidas en los manuales de procedimientos de Automatización.	15 de julio de 2025
AWS-SSM-RemediationAutomation-ExecutionRolePolicy: actualización de una política existente	Systems Manager actualizó la política administrada para mejorar la seguridad de la API ssm:StartAutomationExecution al exigir permisos tanto para los tipos de recursos de “document” como “automation-execution”. La política actualizada proporciona permisos más completos y detallados para la ejecución de la automatización de la remediación, incluidas descripciones mejoradas de las capacidades de remediación de redes, permisos de creación de puntos de conexión de Amazon VPC más específicos, permisos detallados de administración de grupos de seguridad y controles mejorados de	16 de julio de 2025

Cambio	Descripción	Fecha
	etiquetado de recursos para las operaciones de remediación.	
AWS-SSM-RemediationAutomation-AdministrationPolicy: actualización a una política existente	Systems Manager actualizó la política administrada para respaldar las mejoras en la autorización de la API orientadas a las operaciones de automatización de la remediación. La política actualizada mejora los permisos para ejecutar las actividades definidas en los documentos de automatización, con controles de seguridad y patrones de acceso a los recursos mejorados para los flujos de trabajo de remediación.	16 de julio de 2025

Cambio	Descripción	Fecha
AWS-SSM-DiagnosisAutomation-ExecutionPolicy: actualización a una política existente	Systems Manager actualizó la política administrada para proporcionar permisos más detallados y precisos para la ejecución de la automatización del diagnóstico. La política actualizada incluye descripciones mejoradas para el acceso a los recursos de Amazon EC2 y Amazon VPC, permisos de automatización de SSM más específicos y descripciones de permisos de IAM y AWS KMS mejoradas con las restricciones de recursos adecuadas.	16 de julio de 2025
AWS-SSM-DiagnosisAutomation-AdministrationPolicy: actualización a una política existente	Systems Manager actualizó la política administrada para proporcionar permisos y condiciones de seguridad más específicos para las operaciones de automatización del diagnóstico. La política actualizada proporciona controles de seguridad mejorados para el uso de claves de AWS KMS, el acceso a los buckets de Amazon S3 y las suposiciones de roles, con condiciones más estrictas basadas en los recursos y restricciones a nivel de cuenta.	16 de julio de 2025

Cambio	Descripción	Fecha
AWSQuickSetupDeploymentRolePolicy: actualización de una política	Systems Manager agregó permisos a la política administrada `AWSQuickSetupDeploymentRolePolicy` para acceder al manual de procedimientos AWSQuickSetupType-ManageInstanceProfile, propiedad de Amazon. Este permiso posibilita que Quick Setup cree asociaciones mediante el uso de la política administrada en lugar de políticas insertadas.	14 de julio de 2025
AWSSystemsManagerJustInTimeNodeAccessRolePropagationPolicy: actualización de una política	Systems Manager agregó permisos para avalar que Systems Manager etiquete un recurso compartido por AWS Resource Access Manager para acceder a los nodos justo a tiempo.	30 de abril de 2025
AWSQuickSetupManageJITNAResourcesExecutionPolicy: actualización de una política	Systems Manager agregó permisos para avalar que Systems Manager etiquete roles de IAM creados para el acceso a los nodos justo a tiempo.	30 de abril de 2025

Cambio	Descripción	Fecha
AWSSystemsManagerJustInTimeAccessTokenSessionPolicy: política nueva	Systems Manager agregó una política nueva que permite que Systems Manager aplique permisos restringidos a un token del acceso a los nodos justo a tiempo.	30 de abril de 2025
AWSSystemsManagerNotificationsServicePolicy: política nueva	Systems Manager agregó una política nueva que permite que Systems Manager envíe notificaciones por correo electrónico para las solicitudes de acceso a los nodos justo a tiempo a los aprobadores de las solicitudes de acceso.	30 de abril de 2025
AWSSystemsManagerJustInTimeNodeAccessRolePropagationPolicy: política nueva	Systems Manager agregó una política nueva que permite que Systems Manager replique las políticas de aprobación en diferentes regiones.	30 de abril de 2025
AWSSystemsManagerJustInTimeAccessTokenPolicy: política nueva	Systems Manager agregó una política nueva que permite que Systems Manager genere los tokens de acceso utilizados para el acceso a los nodos justo a tiempo.	30 de abril de 2025

Cambio	Descripción	Fecha
AWSSystemsManagerJustInTimeAccessServicePolicy: política nueva	Systems Manager agregó una política nueva para proporcionar permisos a los recursos de AWS que se administran o utilizan mediante la característica de acceso a los nodos justo a tiempo de Systems Manager.	30 de abril de 2025
AWSQuickSetupManageJITNAResourcesExecutionPolicy: política nueva	Systems Manager agregó una política nueva que permite que Quick Setup, una herramienta de Systems Manager, cree los roles de IAM necesarios para el acceso a los nodos justo a tiempo.	30 de abril de 2025
AWSQuickSetupJITNADeploymentRolePolicy: política nueva	Systems Manager agregó una política nueva que concede los permisos que avalan que Quick Setup implemente el tipo de configuración necesario del acceso a los nodos justo a tiempo.	30 de abril de 2025
AWSSystemsManagerJustInTimeNodeAccessRolePropagationPolicy: actualización de una política	Systems Manager agregó permisos para avalar que Systems Manager etiquete un recurso compartido por AWS Resource Access Manager para acceder a los nodos justo a tiempo.	30 de abril de 2025

Cambio	Descripción	Fecha
AWSQuickSetupManageJITNAResourcesExecutionPolicy: actualización de una política	Systems Manager agregó permisos para avalar que Systems Manager etiquete roles de IAM creados para el acceso a los nodos justo a tiempo.	30 de abril de 2025
AWSSystemsManagerJustInTimeAccessTokenSessionPolicy: política nueva	Systems Manager agregó una política nueva que permite que Systems Manager aplique permisos restringidos a un token del acceso a los nodos justo a tiempo.	30 de abril de 2025
AWSSystemsManagerNotificationsServicePolicy: política nueva	Systems Manager agregó una política nueva que permite que Systems Manager envíe notificaciones por correo electrónico para las solicitudes de acceso a los nodos justo a tiempo a los aprobadores de las solicitudes de acceso.	30 de abril de 2025
AWSSystemsManagerJustInTimeNodeAccessRolePropagationPolicy: política nueva	Systems Manager agregó una política nueva que permite que Systems Manager replique las políticas de aprobación en diferentes regiones.	30 de abril de 2025
AWSSystemsManagerJustInTimeAccessTokenPolicy: política nueva	Systems Manager agregó una política nueva que permite que Systems Manager genere los tokens de acceso utilizados para el acceso a los nodos justo a tiempo.	30 de abril de 2025

Cambio	Descripción	Fecha
AWSSystemsManagerJustInTimeAccessServicePolicy: política nueva	Systems Manager agregó una política nueva para proporcionar permisos a los recursos de AWS que se administran o utilizan mediante la característica de acceso a los nodos justo a tiempo de Systems Manager.	30 de abril de 2025
AWSQuickSetupManageJITNAResourcesExecutionPolicy: política nueva	Systems Manager agregó una política nueva que permite que Quick Setup, una herramienta de Systems Manager, cree los roles de IAM necesarios para el acceso a los nodos justo a tiempo.	30 de abril de 2025
AWSQuickSetupJITNADeploymentRolePolicy: política nueva	Systems Manager agregó una política nueva que concede los permisos que avalan que Quick Setup implemente el tipo de configuración necesario del acceso a los nodos justo a tiempo.	30 de abril de 2025
AWS-SSM-DiagnosisAutomation-OperationalAccountAdministrationRolePolicy: política nueva	Systems Manager agregó una nueva política administrada que concede permisos para que una cuenta operativa diagnostique problemas con los nodos mediante permisos específicos de la organización.	21 de noviembre de 2024

Cambio	Descripción	Fecha
AWS-SSM-RemediationAutomation-OperationalAccountAdministrationRolePolicy : política nueva	Systems Manager agregó una nueva política administrada que concede permisos para que una cuenta operativa diagnostique problemas con los nodos mediante permisos específicos de la organización.	21 de noviembre de 2024
AWS-SSM-Automation-DiagnosisBucketPolicy : política nueva	Systems Manager agregó una nueva política para permitir el inicio de flujos de trabajo de Automatización que diagnostican problemas con los nodos administrados en cuentas y regiones específicas.	21 de noviembre de 2024
AmazonSSMServiceRolePolicy : actualización de una política actual	Systems Manager agregó nuevos permisos que permiten a Explorador de recursos de AWS recopilar detalles sobre las instancias de Amazon EC2 y mostrar los resultados en los widgets del nuevo panel de Systems Manager.	21 de noviembre de 2024
SSMQuickSetupRolePolicy : actualización de una política actual	Systems Manager actualizó la política administrada `SSMQuickSetupRolePolicy` . Esta actualización permite que el rol `AWSServiceRoleForSSMQuickSetup` , asociado y vinculado al servicio, administre las sincronizaciones de los datos de los recursos.	21 de noviembre de 2024

Cambio	Descripción	Fecha
AWS-SSM-DiagnosisAutomation-AdministrationRolePolicy: política nueva	Systems Manager agregó una nueva política para permitir el inicio de flujos de trabajo de Automatización que diagnostican problemas con los nodos administrados en una cuenta y regiones específicas.	21 de noviembre de 2024
AWS-SSM-DiagnosisAutomation-ExecutionRolePolicy: política nueva	Systems Manager agregó una nueva política para permitir el inicio de flujos de trabajo de Automatización que diagnostican problemas con los nodos administrados en una región y cuenta de destino.	21 de noviembre de 2024
AWS-SSM-RemediationAutomation-AdministrationRolePolicy: política nueva	Systems Manager agregó una nueva política para permitir el inicio de flujos de trabajo de Automatización que resuelven problemas con los nodos administrados en cuentas y regiones específicas.	21 de noviembre de 2024
AWS-SSM-RemediationAutomation-ExecutionRolePolicy: política nueva	Systems Manager agregó una nueva política para permitir el inicio de flujos de trabajo de Automatización que resuelven problemas con los nodos administrados en una cuenta y región específicas.	21 de noviembre de 2024

Cambio	Descripción	Fecha
AWSQuickSetupSSMDeploymentRolePolicy: actualización de una política	Systems Manager agregó permisos para avalar que Systems Manager etiquete los roles de IAM y Lambda creados para la consola unificada.	7 de mayo de 2025
AWSQuickSetupSSMManageResourcesExecutionPolicy : política nueva	Systems Manager agregó una nueva política para respaldar la ejecución de una operación de Quick Setup que crea roles de IAM para las asociaciones de Quick Setup, que a su vez se crean mediante una implementación de `AWSQuickSetupType-SSM` .	21 de noviembre de 2024
AWSQuickSetupSSMLifecycleManagementExecutionPolicy : política nueva	Systems Manager agregó una nueva política para permitir que Quick Setup ejecute un recurso de AWS CloudFormation personalizado en los eventos del ciclo de vida durante una implementación de Quick Setup.	21 de noviembre de 2024
AWSQuickSetupSSMDeploymentRolePolicy : política nueva	Systems Manager agregó una nueva política para admitir la concesión de permisos administrativos que permiten a Quick Setup crear recursos que se utilizan durante el proceso de incorporación de Systems Manager.	21 de noviembre de 2024

Cambio	Descripción	Fecha
AWSQuickSetupSSMDeploymentS3BucketRolePolicy : política nueva	Systems Manager agregó una nueva política para admitir la administración y la recuperación de información sobre buckets específicos de la cuenta de la entidad principal que se administran con plantillas de AWS CloudFormation.	21 de noviembre de 2024
AWSQuickSetupEnableDHMCExecutionPolicy : política nueva	Systems Manager presenta una nueva política que permite a Quick Setup crear un rol de IAM que, a su vez, utilice la política AmazonSSMManagedEC2InstanceDefaultPolicy existente. Esta política contiene todos los permisos necesarios para que SSM Agent se comunique con el servicio de Systems Manager. La nueva política también permite modificar la configuración del servicio de Systems Manager.	21 de noviembre de 2024
AWSQuickSetupEnableAREXExecutionPolicy : política nueva	Systems Manager agregó una nueva política que permite que Quick Setup cree un rol vinculado al servicio de Explorador de recursos de AWS para acceder a las vistas y los índices de agregadores de Resource Explorer.	21 de noviembre de 2024

Cambio	Descripción	Fecha
AWSQuickSetupManagedInstanceProfileExecutionPolicy: política nueva	Systems Manager agregó una nueva política que permite a Quick Setup crear un perfil de instancia de Quick Setup predeterminado y adjuntarlo a cualquier instancia de Amazon EC2 que no tenga uno asociado. Esta nueva política también permite que Quick Setup adjunte permisos a los perfiles existentes para garantizar que se concedan todos los permisos necesarios de Systems Manager.	21 de noviembre de 2024
AWSQuickSetupFullAccess: política nueva	Systems Manager agregó una nueva politica que permite el acceso total a las acciones y los datos de la API de Quick Setup de AWS Systems Manager en la AWS Management Console y los SDK de AWS, así como un acceso limitado a otros recursos de Servicio de AWS necesarios para las operaciones de Quick Setup.	21 de noviembre de 2024

Cambio	Descripción	Fecha
AWSQuickSetupReadOnlyAccess : política nueva	Systems Manager agregó una nueva política que concede permisos de solo lectura que permiten a las entidades principales ver datos e informes de Quick Setup de AWS Systems Manager, incluida la información de otros recursos de Servicio de AWS necesarios para las operaciones de Quick Setup.	21 de noviembre de 2024
SSMQuickSetupRolePolicy : actualización de una política actual	Systems Manager agregó nuevos permisos para permitir a Quick Setup comprobar el estado de los conjuntos de pilas adicionales de AWS CloudFormation que ha creado.	13 de agosto de 2024
AmazonSSMManagedEC2InstanceDefaultPolicy : actualización de una política actual	Systems Manager ha añadido identificadores de declaración (SID) a la política de JSON para `AmazonSSMManagedEC2InstanceDefaultPolicy` . Estos SID proporcionan descripciones en línea del propósito de cada declaración de política.	18 de julio de 2024

Cambio	Descripción	Fecha
SSMQuickSetupRolePolicy : política nueva	Systems Manager agregó una nueva política que permite a Quick Setup comprobar el estado de los recursos desplegados y corregir las instancias que se han desviado de la configuración original.	3 de julio de 2024
AWSQuickSetupDeploymentRolePolicy : política nueva	Systems Manager agregó una nueva política para admitir varios tipos de configuración de Configuración Rápida que crean roles y automatizaciones de IAM, que a su vez configuran servicios y características de Amazon Web Services de uso frecuente, mediante las prácticas recomendadas.	3 de julio de 2024
AWSQuickSetupPatchPolicyDeploymentRolePolicy : política nueva	Systems Manager agregó una nueva política que permite a Quick Setup crear recursos asociados a las configuraciones de Patch Manager de las políticas de revisiones de Quick Setup.	3 de julio de 2024
AWSQuickSetupPatchPolicyBaselineAccess: nueva política	Systems Manager agregó una nueva política que permite a Quick Setup acceder a las líneas de base de revisiones en Patch Manager con permisos de solo lectura.	3 de julio de 2024

Cambio	Descripción	Fecha
AWSSystemsManagerEnableExplorerExecutionPolicy: nueva política	Systems Manager agregó una nueva política que permite a Quick Setup conceder permisos administrativos para habiltiar Explorer.	3 de julio de 2024
AWSSystemsManagerEnableConfigRecordingExecutionPolicy: nueva política	Systems Manager agregó una nueva política que permite a Quick Setup habilitar y configurar el registro de la configuración de AWS Config.	3 de julio de 2024
AWSQuickSetupDevOpsGuruPermissionsBoundary: nueva política	Systems Manager agregó una nueva política que permite a Quick Setup habilitar y configurar Amazon DevOps Guru.	3 de julio de 2024
AWSQuickSetupDistributorPermissionsBoundary: nueva política	Systems Manager agregó una nueva política para permitir que Quick Setup habilite y configure Distribuidor, una herramienta de AWS Systems Manager.	3 de julio de 2024
AWSQuickSetupSSMHostMgmtPermissionsBoundary: nueva política	Systems Manager agregó una nueva política que permite que Quick Setup habilite y configure las herramientas de Systems Manager para administrar de forma segura las instancias de Amazon EC2.	3 de julio de 2024

Cambio	Descripción	Fecha
AWSQuickSetupPatchPolicyPermissionsBoundary: nueva política	Systems Manager agregó una nueva política que permite que Quick Setup habilite y configure las políticas de revisiones en Patch Manager, una herramienta de AWS Systems Manager.	3 de julio de 2024
AWSQuickSetupSchedulerPermissionsBoundary: nueva política	Systems Manager agregó una nueva política que permite a Quick Setup habilitar y configurar las operaciones programadas en las instancias de Amazon EC2 y otros recursos.	3 de julio de 2024
AWSQuickSetupCFGCPacksPermissionsBoundary: nueva política	Systems Manager agregó una nueva política que permite a Quick Setup implementar paquetes de conformidad con AWS Config.	3 de julio de 2024
AWSSystemsManagerOpsDataSyncServiceRolePolicy: actualización de una política actual	OpsCenter actualizó la política para mejorar la seguridad del código de servicio dentro del rol vinculado al servicio para que Explorer administre las operaciones relacionadas con OPSData.	3 de julio de 2023

Cambio	Descripción	Fecha
AmazonSSMManagedEC2InstanceDefaultPolicy: política nueva	Systems Manager agregó una política nueva para permitir la funcionalidad de Systems Manager en las instancias de Amazon EC2 sin necesidad de usar un perfil de instancia de IAM.	18 de agosto de 2022
AmazonSSMServiceRolePolicy: actualización a una política existente.	Systems Manager agregó nuevos permisos para permitir a Explorer crear una regla administrada cuando active Security Hub desde Explorer o OpsCenter. Se agregaron nuevos permisos para verificar que la configuración y el optimizador de computación cumplen con los requisitos necesarios antes de permitir OpsData.	27 de abril de 2021
AWSSystemsManagerOpsDataSyncServiceRolePolicy: política nueva	Systems Manager agregó una política nueva para crear y actualizar OpsItems y OpsData desde resultados de Security Hub en Explorer y OpsCenter.	27 de abril de 2021
AmazonSSMServiceRolePolicy: actualización de una política actual	Systems Manager agregó nuevos permisos para permitir la visualización de detalles agregados de OpsData y OpsItems de varias cuentas y Regiones de AWS en Explorer.	24 de marzo de 2021

Cambio	Descripción	Fecha
Systems Manager comenzó el seguimiento de los cambios.	Systems Manager comenzó el seguimiento de los cambios de las políticas administradas de AWS.	12 de marzo de 2021

Políticas administradas adicionales para Systems Manager

Además de las políticas administradas antes mencionadas en este tema, Systems Manager también admite las siguientes políticas.

- `AmazonSSMAutomationApproverAccess`: política administrada por AWS que permite el acceso para ver ejecuciones de automatización y enviar decisiones de aprobación de automatización en espera de aprobación.
- `AmazonSSMDirectoryServiceAccess`: política administrada por AWS que permite a SSM Agent acceder a AWS Directory Service en nombre del usuario para realizar solicitudes de unión al dominio por parte del nodo administrado.
- `AmazonSSMFullAccess`: política administrada por AWS que concede acceso total a la API y los documentos de Systems Manager.
- `AmazonSSMMaintenanceWindowRole`: política administrada por AWS que proporciona periodos de mantenimiento con permisos para la API de Systems Manager.
- `AmazonSSMManagedInstanceCore`: política administrada de AWS que permite que un nodo utilice la funcionalidad básica del servicio Systems Manager.
- `AmazonSSMPatchAssociation`: política administrada por AWS que proporciona acceso a las instancias secundarias para realizar la asociación de revisiones.
- `AmazonSSMReadOnlyAccess`: política administrada por AWS que concede acceso a operaciones de la API de solo lectura de Systems Manager, como `Get*` y `List*`.
- `AWSSSMOpsInsightsServiceRolePolicy`: política administrada por AWS que proporciona permisos para crear y actualizar información operativa de OpsItems en Systems Manager. Se utiliza para proporcionar permisos a través del rol vinculado al servicio `AWSServiceRoleForAmazonSSM_OpsInsights`.
- `AWSSystemsManagerAccountDiscoveryServicePolicy`: política administrada por AWS que concede a Systems Manager permiso para descubrir la información de la Cuenta de AWS.

- AWSSystemsManagerChangeManagementServicePolicy: política administrada por AWS que proporciona acceso a recursos de AWS administrados o utilizados por el marco de administración de cambios de Systems Manager y utilizados por el rol vinculado al servicio AWSServiceRoleForSystemsManagerChangeManagement.
- AmazonEC2RoleforSSM— Esta política ya no se admite y no debe utilizarse. En su lugar, utilice la política AmazonSSMManagedInstanceCore para permitir la funcionalidad principal del servicio de Systems Manager en las instancias EC2. Para obtener información, consulte Configuración de permisos de instancia requeridos para Systems Manager.

Solución de problemas de identidades de AWS Systems Manager y accesos

Utilice la siguiente información para diagnosticar y solucionar los problemas comunes que puedan surgir cuando trabaje con AWS Systems Manager y AWS Identity and Access Management (IAM).

Temas

- No tengo autorización para realizar una acción en Systems Manager
- No tengo autorización para realizar la operación iam:PassRole
- Quiero permitir a personas externas a mi Cuenta de AWS el acceso a mis recursos de AWS

No tengo autorización para realizar una acción en Systems Manager

Si la AWS Management Console le indica que no está autorizado para llevar a cabo una acción, debe ponerse en contacto con su administrador para recibir ayuda. El administrador es la persona que le proporcionó las credenciales de inicio de sesión.

En el siguiente ejemplo, el error se produce cuando el usuario, mateojackson, intenta utilizar la consola para ver detalles sobre un documento, pero no tiene permisos ssm:GetDocument.

```
User: arn:aws:ssm::123456789012:user/mateojackson isn't authorized to perform:
 ssm:GetDocument on resource: MyExampleDocument
```

En este caso, Mateo pide a su administrador que actualice sus políticas de forma que pueda obtener acceso al recurso MyExampleDocument mediante la acción ssm:GetDocument.

No tengo autorización para realizar la operación iam:PassRole

Si recibe un error que indica que no tiene autorización para realizar la acción `iam:PassRole`, las políticas deben actualizarse a fin de permitirle pasar un rol a Systems Manager.

Algunos servicios de AWS le permiten transferir un rol existente a dicho servicio en lugar de crear un nuevo rol de servicio o uno vinculado al servicio. Para ello, debe tener permisos para transferir el rol al servicio.

En el siguiente ejemplo, el error se produce cuando un usuario de IAM denominado `marymajor` intenta utilizar la consola para realizar una acción en Systems Manager. Sin embargo, la acción requiere que el servicio cuente con permisos que otorguen un rol de servicio. Mary no tiene permisos para transferir el rol al servicio.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

En este caso, las políticas de Mary se deben actualizar para permitirle realizar la acción `iam:PassRole`.

Si necesita ayuda, póngase en contacto con su administrador de AWS. El administrador es la persona que le proporcionó las credenciales de inicio de sesión.

Quiero permitir a personas externas a mi Cuenta de AWS el acceso a mis recursos de AWS

Puede crear un rol que los usuarios de otras cuentas o las personas externas a la organización puedan utilizar para acceder a sus recursos. Puede especificar una persona de confianza para que asuma el rol. En el caso de los servicios que admitan las políticas basadas en recursos o las listas de control de acceso (ACL), puede utilizar dichas políticas para conceder a las personas acceso a sus recursos.

Para más información, consulte lo siguiente:

- Para obtener información acerca de si Systems Manager admite estas características, consulte Cómo funciona AWS Systems Manager con IAM.
- Para obtener información acerca de cómo proporcionar acceso a los recursos de las Cuenta de AWS de su propiedad, consulte Proporcionar acceso a un usuario de IAM a otra cuenta de AWS de la que es propietario en la Guía del usuario de IAM.

- Para obtener información acerca de cómo proporcionar acceso a tus recursos a Cuentas de AWS de terceros, consulte Proporcionar acceso a Cuentas de AWS que son propiedad de terceros en la Guía del usuario de IAM.
- Para obtener información sobre cómo proporcionar acceso mediante una federación de identidades, consulte Proporcionar acceso a usuarios autenticados externamente (identidad federada) en la Guía del usuario de IAM.
- Para conocer sobre la diferencia entre los roles y las políticas basadas en recursos para el acceso entre cuentas, consulte Cross account resource access in IAM en la Guía del usuario de IAM.

Cómo utilizar roles vinculados a servicios de Systems Manager

AWS Systems Manager utiliza roles vinculados a servicios de AWS Identity and Access Management (IAM). Un rol vinculado a servicios es un tipo único de rol de IAM que está vinculado directamente a Systems Manager. Los roles vinculados a servicios están predefinidos por Systems Manager e incluyen todos los permisos que el servicio requiere para llamar a otros Servicios de AWS en su nombre.

> **Note**
>
> Un rol de servicio es diferente de un rol vinculado a servicio. Un rol de servicio es un tipo de rol de AWS Identity and Access Management (IAM) que concede permisos a un Servicio de AWS para que pueda acceder a recursos de AWS. Solo unos pocos casos de Systems Manager requieren un rol de servicio. Cuando cree un rol de servicio para Systems Manager, puede elegir los permisos que va a conceder para que pueda acceder a otros recursos de AWS o pueda interactuar con ellos.

Con un rol vinculado a servicios, resulta más sencillo configurar Systems Manager, porque no es preciso agregar los permisos necesarios manualmente. Systems Manager define los permisos de los roles vinculados con su propio servicio y, a menos que esté definido de otra manera, solo Systems Manager puede asumir sus roles. Los permisos definidos incluyen las políticas de confianza y de permisos, y que la política de permisos no se puede adjuntar a ninguna otra entidad de IAM.

Solo puede eliminar un rol vinculado a servicios después de eliminar sus recursos relacionados. De esta forma, se protegen los recursos de Systems Manager, ya que se evita que se puedan eliminar accidentalmente permisos de acceso a los recursos.

> **Note**
>
> Para los nodos que no son de EC2 en un entorno híbrido y multinube, necesita disponer un rol de IAM adicional que permita a las máquinas comunicarse con el servicio de Systems Manager. Esta es la rol de servicio de IAM para Systems Manager. Este rol concede a AWS Security Token Service (AWS STS) la confianza AssumeRole para el servicio Systems Manager. La acción `AssumeRole` devuelve un conjunto de credenciales de seguridad temporales (consistente en un ID de clave de acceso, una clave de acceso secreta y un token de seguridad). Estas credenciales temporales se usan para obtener acceso a recursos de AWS a los que normalmente no tendría acceso. Para obtener más información, consulte Creación del rol de servicio de IAM requerido para Systems Manager en entornos híbridos y multinube y AssumeRole en la Referencia de la API de AWS Security Token Service.

Para obtener información sobre otros servicios que admiten roles vinculados al servicio, consulte Servicios de AWS que funcionan con IAM y busque los servicios que muestran Sí en la columna Roles vinculados a servicios. Elija una opción Sí con un enlace para ver la documentación acerca del rol vinculado al servicio en cuestión.

Temas

- Uso de roles para recopilar datos de inventario y ver OpsData
- Uso de roles para recopilar información de la Cuenta de AWS para OpsCenter y Explorer
- Uso de roles para crear OpsData y OpsItems para Explorer
- Uso de roles para crear OpsItems con información operativa en Systems Manager OpsCenter
- Uso de roles para mantener el estado y la consistencia de los recursos aprovisionados de Quick Setup
- Uso de roles para exportar OpsData de Explorer
- Cómo utilizar roles para permitir el acceso a los nodos justo a tiempo
- Cómo utilizar roles para enviar notificaciones de solicitudes de acceso a los nodos justo a tiempo

Uso de roles para recopilar datos de inventario y ver OpsData

Systems Manager utiliza el rol vinculado a servicio denominado **`AWSServiceRoleForAmazonSSM`**. AWS Systems Manager utiliza este rol de servicio de IAM para administrar recursos de AWS en su nombre.

Permisos de roles vinculados al servicio de inventario, OpsData y OpStems

El rol vinculado a servicio `AWSServiceRoleForAmazonSSM` solo confía en `ssm.amazonaws.com` para asumir este rol.

Puede utilizar el rol vinculado a servicio de Systems Manager `AWSServiceRoleForAmazonSSM` para lo siguiente:

- La herramienta Inventario de Systems Manager utiliza el rol vinculado a servicios `AWSServiceRoleForAmazonSSM` para recopilar metadatos de inventario de etiquetas y grupos de recursos.
- La herramienta Explorer utiliza el rol vinculado a servicios `AWSServiceRoleForAmazonSSM` para habilitar la visualización de OpsData y OpsItems de varias cuentas. Este rol vinculado a servicios también permite a Explorer crear una regla administrada cuando habilita a Security Hub como origen de datos desde Explorer o OpsCenter.

> **Important**
>
> Antes, la consola de Systems Manager ofrecía la posibilidad de elegir el rol vinculado a servicio de IAM `AWSServiceRoleForAmazonSSM` administrado de AWS que utilizar como rol de mantenimiento para las tareas. Ya no se recomienda utilizar este rol y su política asociada, `AmazonSSMServiceRolePolicy`, para tareas de periodo de mantenimiento. Si está utilizando actualmente este rol para tareas de periodo de mantenimiento, le recomendamos que deje de hacerlo. En su lugar, cree su propio rol de IAM que permita la comunicación entre Systems Manager y otros Servicios de AWS cuando se ejecuten las tareas de periodo de mantenimiento.
> Para obtener más información, consulte Configuración de Maintenance Windows.

La política administrada que se utiliza para proporcionar permisos para el rol `AWSServiceRoleForAmazonSSM` es `AmazonSSMServiceRolePolicy`. Para obtener información detallada acerca de los permisos que concede, consulte Política administrada de AWS: AmazonSSMServiceRolePolicy.

Creación del rol vinculado al servicio **AWSServiceRoleForAmazonSSM** de Systems Manager

Puede utilizar la consola de IAM para crear un rol vinculado a servicios con el caso de uso de EC2. El uso de comandos para IAM en la AWS Command Line Interface (AWS CLI) o mediante la API de IAM, crea un rol vinculado a servicios con el nombre de servicio `ssm.amazonaws.com`. Para obtener más información, consulte Creating a service-linked role en la Guía del usuario de IAM.

Si elimina este rol vinculado a servicios y necesita crearlo de nuevo, puede utilizar el mismo proceso para volver a crear el rol en su cuenta.

Modificación del rol vinculado al servicio **AWSServiceRoleForAmazonSSM** de Systems Manager

Systems Manager no le permite modificar el rol vinculado al servicio `AWSServiceRoleForAmazonSSM`. Después de crear un rol vinculado a un servicio, no puede cambiarle el nombre, ya que varias entidades pueden hacer referencia a él. Sin embargo, puede editar la descripción del rol mediante IAM. Para obtener más información, consulte Editar un rol vinculado a servicios en la Guía del usuario de IAM.

Eliminación del rol vinculado al servicio **AWSServiceRoleForAmazonSSM** de Systems Manager

Si ya no necesita utilizar ninguna característica ni ningún servicio que requiera un rol vinculado a un servicio, le recomendamos que elimine el rol. De esta forma no conservará una entidad no utilizada que no se monitorice ni se mantenga de forma activa. Puede utilizar la consola de IAM, la AWS CLI o la API de IAM para eliminar manualmente el rol vinculado a servicios. Para ello, primero debe limpiar manualmente los recursos del rol vinculado a servicio y, a continuación, eliminarlo manualmente.

Dado que al rol vinculado a servicios `AWSServiceRoleForAmazonSSM` lo pueden utilizar varias herramientas, asegúrese de que ninguna de ellas esté utilizando el rol antes de intentar eliminarlo.

- Inventario: si elimina el rol vinculado a servicios utilizado por la herramienta Inventario, los datos de Inventario relacionados con las etiquetas y los grupos de recursos dejarán de estar sincronizados. Debe limpiar los recursos del rol vinculado a servicio antes de eliminarlo manualmente.
- Explorer: si elimina el rol vinculado a servicios utilizado por la herramienta Explorer, OpsData y OpsItems entre cuentas y entre regiones ya no son visibles.

> **Note**
>
> Si el servicio Systems Manager está utilizando el rol cuando se intentan eliminar etiquetas o grupos de recursos, es posible que la eliminación falle. En tal caso, espere unos minutos e intente de nuevo la operación.

Para eliminar los recursos de Systems Manager que se utilizan en **`AWSServiceRoleForAmazonSSM`**

1. Para eliminar etiquetas, consulte Add and delete tags on an individual resource (Adición y eliminación de etiquetas en un recurso individual).
2. Para eliminar grupos de recursos, consulte Eliminación de grupos de AWS Resource Groups.

Para eliminar manualmente el rol vinculado al servicio **`AWSServiceRoleForAmazonSSM`** mediante IAM

Utilice la consola de IAM, la AWS CLI o la API de IAM para eliminar el rol vinculado a servicios `AWSServiceRoleForAmazonSSM`. Para obtener más información, consulte Eliminación de un rol vinculado a servicios en la Guía del usuario de IAM.

Regiones admitidas para el rol vinculado al servicio **`AWSServiceRoleForAmazonSSM`** de Systems Manager

Systems Manager admite el uso del rol vinculado al servicio `AWSServiceRoleForAmazonSSM` en todas las Regiones de AWS en las que el servicio está disponible. Para obtener más información, consulte Puntos de conexión y cuotas de AWS Systems Manager.

Uso de roles para recopilar información de la Cuenta de AWS para OpsCenter y Explorer

Systems Manager utiliza el rol vinculado a servicio denominado **`AWSServiceRoleForAmazonSSM_AccountDiscovery`**. AWS Systems Manager utiliza este rol de servicio de IAM para llamar a otros Servicios de AWS con objeto de descubrir información sobre la Cuenta de AWS.

Permisos de roles vinculados al servicio de detección de cuentas de Systems Manager

El rol vinculado al servicio `AWSServiceRoleForAmazonSSM_AccountDiscovery` depende de los siguientes servicios para asumir el rol:

- `accountdiscovery.ssm.amazonaws.com`

La política de permisos del rol permite que Systems Manager realice las siguientes acciones en los recursos especificados:

- `organizations:DescribeAccount`
- `organizations:DescribeOrganizationalUnit`
- `organizations:DescribeOrganization`
- `organizations:ListAccounts`
- `organizations:ListAWSServiceAccessForOrganization`
- `organizations:ListChildren`
- `organizations:ListParents`
- `organizations:ListDelegatedServicesForAccount`
- `organizations:ListDelegatedAdministrators`
- `organizations:ListRoots`

Debe configurar permisos para permitir a una entidad de IAM (como un usuario, grupo o rol) crear, editar o eliminar un rol vinculado a servicios. Para obtener más información, consulte Permisos de roles vinculados a servicios en la Guía del usuario de IAM.

Creación del rol vinculado al servicio **`AWSServiceRoleForAmazonSSM_AccountDiscovery`** de Systems Manager

Debe crear un rol vinculado a servicios si desea utilizar Explorer y OpsCenter, herramientas de Systems Manager, en varias Cuentas de AWS. En OpsCenter, debe crear manualmente un rol vinculado a servicios. Para obtener más información, consulte (Opcional) Configuración manual de OpsCenter para administrar OpsItems de forma centralizada entre cuentas.

En Explorer, si crea una sincronización de datos de recursos mediante Systems Manager en la AWS Management Console, puede crear el rol vinculado a servicios mediante la elección del botón Create role (Crear rol). Si desea crear una sincronización de datos de recursos mediante programación,

debe crear el rol antes de crear la sincronización de datos de recursos. Puede crear el rol con la operación CreateServiceLinkedRole de la API.

Modificación del rol vinculado al servicio **AWSServiceRoleForAmazonSSM_AccountDiscovery** de Systems Manager

Systems Manager no le permite modificar el rol vinculado al servicio `AWSServiceRoleForAmazonSSM_AccountDiscovery`. Después de crear un rol vinculado a un servicio, no puede cambiarle el nombre, ya que varias entidades pueden hacer referencia a él. Sin embargo, puede editar la descripción del rol mediante IAM. Para obtener más información, consulte Editar un rol vinculado a servicios en la Guía del usuario de IAM.

Eliminación del rol vinculado al servicio **AWSServiceRoleForAmazonSSM_AccountDiscovery** de Systems Manager

Si ya no necesita usar una característica o servicio que requieran un rol vinculado a un servicio, le recomendamos que elimine dicho rol. De esta forma no conservará una entidad no utilizada que no se monitorice ni se mantenga de forma activa. Sin embargo, debe limpiar el rol vinculado a servicios antes de eliminarlo manualmente.

Limpieza del rol vinculado al servicio de **AWSServiceRoleForAmazonSSM_AccountDiscovery**

Para poder utilizar IAM con objeto de eliminar el rol vinculado al servicio `AWSServiceRoleForAmazonSSM_AccountDiscovery`, antes debe eliminar todas las sincronizaciones de datos del recurso de Explorer.

> **Note**
>
> Si el servicio Systems Manager está utilizando el rol cuando intenta eliminar los recursos, la eliminación podría producir un error. En tal caso, espere unos minutos e intente de nuevo la operación.

Eliminar manualmente el rol vinculado al servicio **AWSServiceRoleForAmazonSSM_AccountDiscovery**

Utilice la consola de IAM, la AWS CLI o la API de AWS para eliminar el rol vinculado a servicios de `AWSServiceRoleForAmazonSSM_AccountDiscovery`. Para obtener más información, consulte Eliminar un rol vinculado a un servicio en la Guía del usuario de IAM.

Regiones admitidas para el rol vinculado al servicio **`AWSServiceRoleForAmazonSSM_AccountDiscovery`** de Systems Manager

Systems Manager admite el uso de roles vinculados a servicios en todas las regiones en las que el servicio está disponible. Para obtener más información, consulte Puntos de conexión y cuotas de AWS Systems Manager.

Actualizaciones del rol vinculado al servicio de AWSServiceRoleForAmazonSSM_AccountDiscovery

Puede consultar los detalles sobre las actualizaciones del rol vinculado al servicio de AWSServiceRoleForAmazonSSM_AccountDiscovery, ya que este servicio comenzó a realizar el seguimiento de estos cambios. Para obtener alertas automáticas sobre cambios en esta página, suscríbase a la fuente RSS en la página de Systems Manager Historial del documento.

Cambio	Descripción	Fecha
Nuevos permisos agregados	Este rol vinculado a servicios ahora incluye los permisos `organizations:DescribeOrganizationalUnit` y `organizations:ListRoots`. Estos permisos habilitan a una cuenta de administración de AWS Organizations o a una cuenta de administrador delegado de Systems Manager a trabajar con OpsItems en varias cuentas. Para obtener más información, consulte (Opcional) Configuración manual de OpsCenter para administrar OpsItems de forma centralizada entre cuentas.	17 de octubre de 2022

Uso de roles para crear OpsData y OpsItems para Explorer

Systems Manager usa el rol vinculado a servicio denominado **`AWSServiceRoleForSystemsManagerOpsDataSync`**. AWS Systems Manager usa este rol de servicio de IAM para que Explorer cree OpsData y OpsItems.

Permisos de roles vinculados al servicio de sincronización de OpsData de Systems Manager

El rol vinculado al servicio `AWSServiceRoleForSystemsManagerOpsDataSync` depende de los siguientes servicios para asumir el rol:

- `opsdatasync.ssm.amazonaws.com`

La política de permisos del rol permite que Systems Manager realice las siguientes acciones en los recursos especificados:

- Systems Manager Explorer requiere que un rol vinculado al servicio otorgue permiso para actualizar una búsqueda de seguridad cuando se actualiza un OpsItem, que cree y actualice un OpsItem y que desactive el origen de datos de Security Hub cuando los clientes eliminen una regla administrada por SSM.

La política administrada que se utiliza para proporcionar permisos para el rol `AWSServiceRoleForSystemsManagerOpsDataSync` es `AWSSystemsManagerOpsDataSyncServiceRolePolicy`. Para obtener información detallada acerca de los permisos que concede, consulte Política administrada por AWS: AWSSystemsManagerOpsDataSyncServiceRolePolicy.

Debe configurar permisos para permitir a una entidad de IAM (como un usuario, grupo o rol) crear, editar o eliminar un rol vinculado a servicios. Para obtener más información, consulte Permisos de roles vinculados a servicios en la Guía del usuario de IAM.

Creación del rol vinculado al servicio **`AWSServiceRoleForSystemsManagerOpsDataSync`** de Systems Manager

No necesita crear manualmente un rol vinculado a servicios. Cuando se habilita Explorer en la AWS Management Console, Systems Manager se encarga de crear el rol vinculado a servicio.

> **Important**
>
> Este rol vinculado a servicios se puede mostrar en su cuenta si se ha completado una acción en otro servicio que utilice las características compatibles con este rol. Además, si utilizaba el servicio de Systems Manager antes del 1 de enero de 2017, cuando comenzó a admitir los roles vinculados a servicios, Systems Manager creó el rol `AWSServiceRoleForSystemsManagerOpsDataSync` en su cuenta. Para obtener más información, consulte Un nuevo rol ha aparecido en mi cuenta de IAM.

Si elimina este rol vinculado a servicios y necesita crearlo de nuevo, puede utilizar el mismo proceso para volver a crear el rol en su cuenta. Cuando se habilita Explorer en la AWS Management Console, Systems Manager se encarga de volver a crear el rol vinculado a servicio.

También puede utilizar la consola de IAM para crear un rol vinculado a servicios con el caso de uso del rol de servicio de AWS que permite que Explorer cree OpsData y OpsItems. En la AWS CLI o la API de AWS, cree un rol vinculado al servicio con el nombre de servicio `opsdatasync.ssm.amazonaws.com`. Para obtener más información, consulte Creación de un rol vinculado a un servicio en la Guía del usuario de IAM. Si elimina este rol vinculado al servicio, puede utilizar este mismo proceso para volver a crear el rol.

Modificación del rol vinculado al servicio **`AWSServiceRoleForSystemsManagerOpsDataSync`** de Systems Manager

Systems Manager no le permite modificar el rol vinculado al servicio `AWSServiceRoleForSystemsManagerOpsDataSync`. Después de crear un rol vinculado a un servicio, no puede cambiarle el nombre, ya que varias entidades pueden hacer referencia a él. Sin embargo, puede editar la descripción del rol mediante IAM. Para obtener más información, consulte Editar un rol vinculado a servicios en la Guía del usuario de IAM.

Eliminación del rol vinculado al servicio **`AWSServiceRoleForSystemsManagerOpsDataSync`** de Systems Manager

Si ya no necesita usar una característica o servicio que requieran un rol vinculado a un servicio, le recomendamos que elimine dicho rol. De esta forma no conservará una entidad no utilizada que no se monitorice ni se mantenga de forma activa. Sin embargo, debe limpiar los recursos de su rol vinculado al servicio antes de eliminarlo manualmente.

> **Note**
>
> Si el servicio Systems Manager está utilizando el rol cuando intenta eliminar los recursos, la eliminación podría producir un error. En tal caso, espere unos minutos e intente de nuevo la operación.

El procedimiento para eliminar los recursos de Systems Manager que utiliza el rol `AWSServiceRoleForSystemsManagerOpsDataSync` depende de si ha configurado Explorer o OpsCenter para integrarse en Security Hub.

Para eliminar los recursos de Systems Manager que se utiliza el rol **`AWSServiceRoleForSystemsManagerOpsDataSync`**

- Para impedir que Explorer cree nuevos OpsItems para los resultados de Security Hub, consulte Cómo dejar de recibir resultados.
- Para evitar que OpsCenter cree nuevos hallazgos de OpsItems para Security Hub, consulte

Para eliminar manualmente el rol vinculado al servicio **`AWSServiceRoleForSystemsManagerOpsDataSync`** mediante IAM

Puede usar la consola de IAM, la AWS CLI o la API de AWS para eliminar el rol vinculado a un servicio de `AWSServiceRoleForSystemsManagerOpsDataSync`. Para obtener más información, consulte Eliminar un rol vinculado a un servicio en la Guía del usuario de IAM.

Regiones admitidas para el rol vinculado al servicio **`AWSServiceRoleForSystemsManagerOpsDataSync`** de Systems Manager

Systems Manager admite el uso de roles vinculados a servicios en todas las regiones en las que el servicio esté disponible. Para obtener más información, consulte Puntos de conexión y cuotas de AWS Systems Manager.

Systems Manager no admite el uso de roles vinculados a servicios en todas las regiones en las que el servicio está disponible. Puede usar el rol `AWSServiceRoleForSystemsManagerOpsDataSync` en las siguientes regiones.

Región de AWS name	Identidad de la región	Compatibilidad en Systems Manager
Este de EE. UU. (Norte de Virginia)	us-east-1	Sí
Este de EE. UU. (Ohio)	us-east-2	Sí
Oeste de EE. UU. (Norte de California)	us-west-1	Sí
Oeste de EE. UU. (Oregón)	us-west-2	Sí
Asia Pacífico (Bombay)	ap-south-1	Sí
Asia Pacífico (Osaka)	ap-northeast-3	Sí
Asia Pacífico (Seúl)	ap-northeast-2	Sí
Asia-Pacífico (Singapur)	ap-southeast-1	Sí
Asia Pacífico (Sídney)	ap-southeast-2	Sí
Asia-Pacífico (Tokio)	ap-northeast-1	Sí
Canadá (centro)	ca-central-1	Sí
Europe (Fráncfort)	eu-central-1	Sí
Europa (Irlanda)	eu-west-1	Sí
Europa (Londres)	eu-west-2	Sí
Europa (París)	eu-west-3	Sí
Europa (Estocolmo)	eu-north-1	Sí
América del Sur (São Paulo)	sa-east-1	Sí
AWS GovCloud (US)	us-gov-west-1	No

Uso de roles para crear OpsItems con información operativa en Systems Manager OpsCenter

Systems Manager utiliza el rol vinculado a servicio denominado **`AWSServiceRoleForAmazonSSM_OpsInsights`**. AWS Systems Manager utiliza este rol de servicio de IAM para crear y actualizar OpsItems con información operativa en Systems Manager OpsCenter.

Permisos de roles vinculados al servicio **`AWSServiceRoleForAmazonSSM_OpsInsights`** para OpsItems de información operativa de Systems Manager

El rol vinculado al servicio `AWSServiceRoleForAmazonSSM_OpsInsights` depende de los siguientes servicios para asumir el rol:

- `opsinsights.ssm.amazonaws.com`

La política de permisos del rol permite que Systems Manager realice las siguientes acciones en los recursos especificados:

JSON

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

Debe configurar permisos para permitir a una entidad de IAM (como un usuario, grupo o rol) crear, editar o eliminar un rol vinculado a servicios. Para obtener más información, consulte Permisos de roles vinculados a servicios en la Guía del usuario de IAM.

Creación del rol vinculado al servicio **AWSServiceRoleForAmazonSSM_OpsInsights** de Systems Manager

Debe crear un rol vinculado a servicios. Si habilita información operativa mediante Systems Manager en la AWS Management Console, puede crear el rol vinculado a servicios mediante la elección del botón Enable (Habilitar).

Cómo modificar el rol vinculado al servicio **AWSServiceRoleForAmazonSSM_OpsInsights** de Systems Manager

Systems Manager no le permite editar el rol vinculado a servicios `AWSServiceRoleForAmazonSSM_OpsInsights`. Después de crear un rol vinculado al servicio, no podrá cambiar el nombre del rol, ya que varias entidades podrían hacer referencia al rol. Sin embargo, sí puede editar la descripción del rol con IAM. Para obtener más información, consulte Editar un rol vinculado a servicios en la Guía del usuario de IAM.

Cómo eliminar el rol vinculado al servicio **AWSServiceRoleForAmazonSSM_OpsInsights** de Systems Manager

Si ya no necesita usar una característica o servicio que requieran un rol vinculado a un servicio, le recomendamos que elimine dicho rol. De esta forma, no tiene una entidad no utilizada que no se monitoree ni mantenga de forma activa. Sin embargo, debe limpiar el rol vinculado a servicios antes de eliminarlo manualmente.

Limpieza del rol vinculado al servicio de **AWSServiceRoleForAmazonSSM_OpsInsights**

Para poder utilizar IAM con objeto de eliminar el rol vinculado a servicio `AWSServiceRoleForAmazonSSM_OpsInsights`, antes debe desactivar la información operativa en Systems Manager OpsCenter. Para obtener más información, consulte Análisis de la información operativa para reducir OpsItems.

Eliminar manualmente el rol vinculado al servicio **AWSServiceRoleForAmazonSSM_OpsInsights**

Utilice la consola de IAM, la AWS CLI o la API de AWS para eliminar el rol vinculado a servicios de `AWSServiceRoleForAmazonSSM_OpsInsights`. Para obtener más información, consulte Eliminar un rol vinculado a un servicio en la Guía del usuario de IAM.

Regiones admitidas para el rol vinculado al servicio **AWSServiceRoleForAmazonSSM_OpsInsights** de Systems Manager

Systems Manager no permite el uso de los roles vinculados al servicio en todas las regiones en las que el servicio está disponible. Puede utilizar el rol AWSServiceRoleForAmazonSSM_OpsInsights en las siguientes regiones.

Nombre de la región	Identidad de la región	Compatibilidad en Systems Manager
Este de EE. UU. (Norte de Virginia)	us-east-1	Sí
Este de EE. UU. (Ohio)	us-east-2	Sí
Oeste de EE. UU. (Norte de California)	us-west-1	Sí
Oeste de EE. UU. (Oregón)	us-west-2	Sí
Asia-Pacífico (Bombay)	ap-south-1	Sí
Asia-Pacífico (Tokio)	ap-northeast-1	Sí
Asia-Pacífico (Seúl)	ap-northeast-2	Sí
Asia-Pacífico (Singapur)	ap-southeast-1	Sí
Asia-Pacífico (Sídney)	ap-southeast-2	Sí

Nombre de la región	Identidad de la región	Compatibilidad en Systems Manager
Asia-Pacífico (Hong Kong)	ap-east-1	Sí
Canadá (centro)	ca-central-1	Sí
Europa (Fráncfort)	eu-central-1	Sí
Europa (Irlanda)	eu-west-1	Sí
Europa (Londres)	eu-west-2	Sí
Europa (París)	eu-west-3	Sí
Europa (Estocolmo)	eu-north-1	Sí
Europa (Milán)	eu-south-1	Sí
América del Sur (São Paulo)	sa-east-1	Sí
Medio Oriente (Baréin)	me-south-1	Sí
África (Ciudad del Cabo)	af-south-1	Sí
AWS GovCloud (US)	us-gov-west-1	Sí
AWS GovCloud (US)	us-gov-east-1	Sí

Uso de roles para mantener el estado y la consistencia de los recursos aprovisionados de Quick Setup

Systems Manager usa el rol vinculado a un servicio denominado **`AWSServiceRoleForSSMQuickSetup`**.

Permisos de roles vinculados a servicios de **`AWSServiceRoleForSSMQuickSetup`** para Systems Manager

El rol vinculado al servicio `AWSServiceRoleForSSMQuickSetup` depende de los siguientes servicios para asumir el rol:

- `ssm-quicksetup.amazonaws.com`

AWS Systems Manager utiliza este rol de servicio de IAM para comprobar el estado de la configuración, garantizar el uso consistente de los parámetros y los recursos aprovisionados y corregir los recursos cuando se detecta una desviación.

La política de permisos del rol permite que Systems Manager realice las siguientes acciones en los recursos especificados:

- `ssm` (Systems Manager): lee información sobre el estado en el que deben estar los recursos configurados, incluso en cuentas de administrador delegado.
- `iam` (AWS Identity and Access Management): esto es necesario para que las sincronizaciones de datos de recursos sean accesibles en todas las organizaciones de AWS Organizations.
- `organizations` (AWS Organizations): lee información sobre las cuentas de los miembros que pertenecen a una organización, tal como se configuró en Organizaciones.
- `cloudformation` (AWS CloudFormation): lee información sobre las pilas de CloudFormation que se utilizan para administrar el estado de los recursos y las operaciones de los conjuntos de pilas de CloudFormation.

La política administrada que se utiliza para proporcionar permisos para el rol `AWSServiceRoleForSSMQuickSetup` es `SSMQuickSetupRolePolicy`. Para obtener información detallada acerca de los permisos que concede, consulte Política administrada de AWS: SSMQuickSetupRolePolicy.

Debe configurar permisos para permitir a una entidad de IAM (como un usuario, grupo o rol) crear, editar o eliminar un rol vinculado a servicios. Para obtener más información, consulte Permisos de roles vinculados a servicios en la Guía del usuario de IAM.

Creación del rol vinculado al servicio **`AWSServiceRoleForSSMQuickSetup`** de Systems Manager

No es necesario crear de forma manual el rol vinculado al servicio AWSServiceRoleForSSMQuickSetup. Cuando crea una configuración de Quick Setup en la AWS Management Console, Systems Manager crea el rol vinculado a servicios por usted.

Modificación del rol vinculado al servicio **AWSServiceRoleForSSMQuickSetup** de Systems Manager

Systems Manager no le permite editar el rol vinculado a servicios `AWSServiceRoleForSSMQuickSetup`. Después de crear un rol vinculado al servicio, no podrá cambiar el nombre del rol, ya que varias entidades podrían hacer referencia al rol. Sin embargo, sí puede editar la descripción del rol con IAM. Para obtener más información, consulte Editar un rol vinculado a servicios en la Guía del usuario de IAM.

Eliminación del rol vinculado al servicio **AWSServiceRoleForSSMQuickSetup** de Systems Manager

Si ya no necesita usar una característica o servicio que requieran un rol vinculado a un servicio, le recomendamos que elimine dicho rol. De esta forma, no tiene una entidad no utilizada que no se monitoree ni mantenga de forma activa. Sin embargo, debe limpiar el rol vinculado a servicios antes de eliminarlo manualmente.

Limpieza del rol vinculado al servicio de **AWSServiceRoleForSSMQuickSetup**

Para poder utilizar IAM para eliminar el rol vinculado al servicio de `AWSServiceRoleForSSMQuickSetup`, primero debe eliminar las configuraciones de Quick Setup que utilizan el rol. Para obtener más información, consulte Edición y eliminación de la configuración.

Eliminar manualmente el rol vinculado al servicio **AWSServiceRoleForSSMQuickSetup**

Utilice la consola de IAM, la AWS CLI o la API de AWS para eliminar el rol vinculado a servicios de `AWSServiceRoleForSSMQuickSetup`. Para obtener más información, consulte los temas siguientes:

- Eliminación de un rol vinculado al servicio en la Guía del usuario de IAM
- delete-configuration-manager en la sección Quick Setup en la Referencia de la AWS CLI
- DeleteConfigurationManager en la Referencia de la API de Quick Setup

Regiones admitidas para el rol vinculado al servicio **AWSServiceRoleForSSMQuickSetup** de Systems Manager

Systems Manager no permite el uso de los roles vinculados al servicio en todas las regiones en las que el servicio está disponible. Puede utilizar el rol AWSServiceRoleForSSMQuickSetup en las siguientes regiones.

- Este de EE. UU. (Ohio)
- Este de EE. UU. (Norte de Virginia)
- Oeste de EE. UU. (Norte de California)
- Oeste de EE. UU. (Oregón)
- Asia-Pacífico (Bombay)
- Asia-Pacífico (Seúl)
- Asia-Pacífico (Singapur)
- Asia-Pacífico (Sídney)
- Asia-Pacífico (Tokio)
- Canadá (centro)
- Europa (Fráncfort)
- Europa (Estocolmo)
- Europa (Irlanda)
- Europa (Londres)
- Europa (París)
- América del Sur (São Paulo)

Uso de roles para exportar OpsData de Explorer

AWS Systems Manager Explorer utiliza el rol de servicio AmazonSSMExplorerExportRole para exportar los datos de operaciones (OpsData) mediante el manual de procedimientos de automatización `AWS-ExportOpsDataToS3`.

Permisos de roles vinculados a servicios de Explorer

El rol vinculado a servicio `AmazonSSMExplorerExportRole` solo confía en `ssm.amazonaws.com` para asumir este rol.

Puede usar el rol vinculado a servicio `AmazonSSMExplorerExportRole` para exportar los datos de operaciones (OpsData) mediante el manual de procedimientos de automatización `AWS-ExportOpsDataToS3`. Puede exportar 5000 elementos OpsData de Explorer como un archivo de valores separados por comas (.csv) a un bucket de Amazon Simple Storage Service (Amazon S3).

La política de permisos del rol permite que Systems Manager realice las siguientes acciones en los recursos especificados:

- `s3:PutObject`
- `s3:GetBucketAcl`
- `s3:GetBucketLocation`
- `sns:Publish`
- `logs:DescribeLogGroups`
- `logs:DescribeLogStreams`
- `logs:CreateLogGroup`
- `logs:PutLogEvents`
- `logs:CreateLogStream`
- `ssm:GetOpsSummary`

Debe configurar permisos para permitir a una entidad de IAM (como un usuario, grupo o rol) crear, editar o eliminar un rol vinculado a servicios. Para obtener más información, consulte Permisos de roles vinculados a servicios en la Guía del usuario de IAM.

Creación del rol vinculado al servicio **`AmazonSSMExplorerExportRole`** de Systems Manager

Systems Manager crea el rol vinculado a servicio `AmazonSSMExplorerExportRole` cuando se exportan OpsData mediante Explorer en la consola de Systems Manager. Para obtener más información, consulte Exportación de OpsData desde Systems Manager Explorer.

Si elimina este rol vinculado a servicios y necesita crearlo de nuevo, puede utilizar el mismo proceso para volver a crear el rol en su cuenta.

Modificación del rol vinculado al servicio **`AmazonSSMExplorerExportRole`** de Systems Manager

Systems Manager no le permite modificar el rol vinculado al servicio `AmazonSSMExplorerExportRole`. Después de crear un rol vinculado a un servicio, no puede cambiarle el nombre, ya que varias entidades pueden hacer referencia a él. Sin embargo, puede editar la descripción del rol mediante IAM. Para obtener más información, consulte Editar un rol vinculado a servicios en la Guía del usuario de IAM.

Eliminación del rol vinculado al servicio **AmazonSSMExplorerExportRole** de Systems Manager

Si ya no necesita utilizar ninguna característica ni ningún servicio que requiera un rol vinculado a un servicio, le recomendamos que elimine el rol. De esta forma no conservará una entidad no utilizada que no se monitorice ni se mantenga de forma activa. Puede utilizar la consola de IAM, la AWS CLI o la API de IAM para eliminar manualmente el rol vinculado a servicios. Para ello, primero debe limpiar manualmente los recursos del rol vinculado a servicio y, a continuación, eliminarlo manualmente.

> **Note**
>
> Si el servicio Systems Manager está utilizando el rol cuando se intentan eliminar etiquetas o grupos de recursos, es posible que la eliminación falle. En tal caso, espere unos minutos e intente de nuevo la operación.

Para eliminar los recursos de Systems Manager que se utilizan en **AmazonSSMExplorerExportRole**

1. Para eliminar etiquetas, consulte Add and delete tags on an individual resource (Adición y eliminación de etiquetas en un recurso individual).
2. Para eliminar grupos de recursos, consulte Eliminación de grupos de AWS Resource Groups.

Para eliminar manualmente el rol vinculado al servicio **AmazonSSMExplorerExportRole** mediante IAM

Utilice la consola de IAM, la AWS CLI o la API de IAM para eliminar el rol vinculado a servicios `AmazonSSMExplorerExportRole`. Para obtener más información, consulte Eliminación de un rol vinculado a servicios en la Guía del usuario de IAM.

Regiones admitidas para el rol vinculado al servicio **AmazonSSMExplorerExportRole** de Systems Manager

Systems Manager admite el uso del rol vinculado al servicio `AmazonSSMExplorerExportRole` en todas las Regiones de AWS en las que el servicio está disponible. Para obtener más información, consulte Puntos de conexión y cuotas de AWS Systems Manager.

Cómo utilizar roles para permitir el acceso a los nodos justo a tiempo

Systems Manager utiliza el rol vinculado a un servicio denominado **`AWSServiceRoleForSystemsManagerJustInTimeAccess`**. AWS Systems Manager utiliza este rol de servicio de IAM para habilitar el acceso a los nodos justo a tiempo.

Permisos de roles vinculados a un servicio para el acceso a los nodos justo a tiempo de Systems Manager

El rol vinculado al servicio `AWSServiceRoleForSystemsManagerJustInTimeAccess` depende de los siguientes servicios para asumir el rol:

- `ssm.amazonaws.com`

La política de permisos del rol permite que Systems Manager realice las siguientes acciones en los recursos especificados:

- `ssm:CreateOpsItem`
- `ssm:GetOpsItem`
- `ssm:UpdateOpsItem`
- `ssm:DescribeOpsItems`
- `ssm:DescribeSessions`
- `ssm:ListTagsForResource`
- `ssm-guiconnect:ListConnections`
- `identitystore:ListGroupMembershipsForMember`
- `identitystore:DescribeUser`
- `identitystore:GetGroupId`
- `identitystore:GetUserId`
- `sso-directory:DescribeUsers`
- `sso-directory:IsMemberInGroup`
- `sso:ListInstances`
- `sso:DescribeRegisteredRegions`
- `sso:ListDirectoryAssociations`
- `ec2:DescribeTags`

La política administrada que se utiliza para proporcionar permisos para el rol `AWSServiceRoleForSystemsManagerJustInTimeAccess` es `AWSSystemsManagerEnableJustInTimeAccessPolicy`. Para obtener información detallada acerca de los permisos que concede, consulte Política administrada por AWS: AWSSystemsManagerJustInTimeAccessServicePolicy.

Debe configurar permisos para permitir a una entidad de IAM (como un usuario, grupo o rol) crear, editar o eliminar un rol vinculado a servicios. Para obtener más información, consulte Permisos de roles vinculados a servicios en la Guía del usuario de IAM.

Cómo crear el rol vinculado al servicio **`AWSServiceRoleForSystemsManagerJustInTimeAccess`** de Systems Manager

No necesita crear manualmente un rol vinculado a servicios. Cuando habilita el acceso a los nodos justo a tiempo en la AWS Management Console, Systems Manager crea el rol vinculado a un servicio por usted.

> **Important**
>
> Este rol vinculado a servicios se puede mostrar en su cuenta si se ha completado una acción en otro servicio que utilice las características compatibles con este rol. Además, si utilizaba el servicio de Systems Manager antes del 19 de noviembre de 2024, cuando comenzó a admitir los roles vinculados a un servicio, Systems Manager creó el rol `AWSServiceRoleForSystemsManagerJustInTimeAccess` en su cuenta. Para obtener más información, consulte Un nuevo rol ha aparecido en mi cuenta de IAM.

Si elimina este rol vinculado a servicios y necesita crearlo de nuevo, puede utilizar el mismo proceso para volver a crear el rol en su cuenta. Cuando habilita el acceso a los nodos justo a tiempo en la AWS Management Console, Systems Manager se encarga de volver a crear el rol vinculado al servicio.

También puede utilizar la consola de IAM para crear un rol vinculado a un servicio con el caso de uso en el que el rol de servicio de AWS permite que Systems Manager habilite el acceso a los nodos justo a tiempo. En la AWS CLI o la API de AWS, cree un rol vinculado al servicio con el nombre de servicio `ssm.amazonaws.com`. Para obtener más información, consulte Creación de un rol vinculado a un servicio en la Guía del usuario de IAM. Si elimina este rol vinculado al servicio, puede utilizar este mismo proceso para volver a crear el rol.

Cómo modificar el rol vinculado al servicio **AWSServiceRoleForSystemsManagerJustInTimeAccess** de Systems Manager

Systems Manager no le permite modificar el rol vinculado al servicio `AWSServiceRoleForSystemsManagerJustInTimeAccess`. Después de crear un rol vinculado a un servicio, no puede cambiarle el nombre, ya que varias entidades pueden hacer referencia a él. Sin embargo, puede editar la descripción del rol mediante IAM. Para obtener más información, consulte Editar un rol vinculado a servicios en la Guía del usuario de IAM.

Cómo eliminar el rol vinculado al servicio **AWSServiceRoleForSystemsManagerJustInTimeAccess** de Systems Manager

Si ya no necesita usar una característica o servicio que requieran un rol vinculado a un servicio, le recomendamos que elimine dicho rol. De esta forma no conservará una entidad no utilizada que no se monitorice ni se mantenga de forma activa. Sin embargo, debe limpiar los recursos de su rol vinculado al servicio antes de eliminarlo manualmente.

> **Note**
>
> Si el servicio Systems Manager está utilizando el rol cuando intenta eliminar los recursos, la eliminación podría producir un error. En tal caso, espere unos minutos e intente de nuevo la operación.

Para eliminar manualmente el rol vinculado al servicio **AWSServiceRoleForSystemsManagerJustInTimeAccess** mediante IAM

Puede usar la consola de IAM, la AWS CLI o la API de AWS para eliminar el rol vinculado a un servicio de `AWSServiceRoleForSystemsManagerJustInTimeAccess`. Para obtener más información, consulte Eliminar un rol vinculado a un servicio en la Guía del usuario de IAM.

Regiones admitidas para el rol vinculado al servicio **AWSServiceRoleForSystemsManagerJustInTimeAccess** de Systems Manager

Región de AWS name	Identidad de la región	Compatibilidad en Systems Manager
Este de EE. UU. (Norte de Virginia)	us-east-1	Sí

Región de AWS name	Identidad de la región	Compatibilidad en Systems Manager
Este de EE. UU. (Ohio)	us-east-2	Sí
Oeste de EE. UU. (Norte de California)	us-west-1	Sí
Oeste de EE. UU. (Oregón)	us-west-2	Sí
Asia-Pacífico (Bombay)	ap-south-1	Sí
Asia Pacífico (Osaka)	ap-northeast-3	Sí
Asia-Pacífico (Seúl)	ap-northeast-2	Sí
Asia-Pacífico (Singapur)	ap-southeast-1	Sí
Asia-Pacífico (Sídney)	ap-southeast-2	Sí
Asia-Pacífico (Tokio)	ap-northeast-1	Sí
Canadá (centro)	ca-central-1	Sí
Europa (Fráncfort)	eu-central-1	Sí
Europa (Irlanda)	eu-west-1	Sí
Europa (Londres)	eu-west-2	Sí
Europa (París)	eu-west-3	Sí
Europa (Estocolmo)	eu-north-1	Sí
América del Sur (São Paulo)	sa-east-1	Sí
AWS GovCloud (US)	us-gov-west-1	No

Cómo utilizar roles para enviar notificaciones de solicitudes de acceso a los nodos justo a tiempo

Systems Manager utiliza el rol vinculado al servicio denominado **`AWSServiceRoleForSystemsManagerNotifications`**. AWS Systems Manager utiliza este rol de servicio de IAM para enviar notificaciones a los aprobadores de las solicitudes de acceso.

Permisos de roles vinculados a un servicio para las notificaciones de acceso a los nodos justo a tiempo de Systems Manager

El rol vinculado al servicio `AWSServiceRoleForSystemsManagerNotifications` depende de los siguientes servicios para asumir el rol:

- `ssm.amazonaws.com`

La política de permisos del rol permite que Systems Manager realice las siguientes acciones en los recursos especificados:

- `identitystore:ListGroupMembershipsForMember`
- `identitystore:ListGroupMemberships`
- `identitystore:DescribeUser`
- `sso:ListInstances`
- `sso:DescribeRegisteredRegions`
- `sso:ListDirectoryAssociations`
- `sso-directory:DescribeUser`
- `sso-directory:ListMembersInGroup`
- `iam:GetRole`

La política administrada que se utiliza para proporcionar permisos para el rol `AWSServiceRoleForSystemsManagerNotifications` es `AWSSystemsManagerNotificationsServicePolicy`. Para obtener información detallada acerca de los permisos que concede, consulte Política administrada por AWS: AWSSystemsManagerNotificationsServicePolicy.

Debe configurar permisos para permitir a una entidad de IAM (como un usuario, grupo o rol) crear, editar o eliminar un rol vinculado a servicios. Para obtener más información, consulte Permisos de roles vinculados a servicios en la Guía del usuario de IAM.

Cómo crear el rol vinculado al servicio **AWSServiceRoleForSystemsManagerNotifications** de Systems Manager

No necesita crear manualmente un rol vinculado a servicios. Cuando habilita el acceso a los nodos justo a tiempo en la AWS Management Console, Systems Manager crea el rol vinculado a un servicio por usted.

> **Important**
>
> Este rol vinculado a servicios se puede mostrar en su cuenta si se ha completado una acción en otro servicio que utilice las características compatibles con este rol. Además, si utilizaba el servicio de Systems Manager antes del 19 de noviembre de 2024, cuando comenzó a admitir los roles vinculados a un servicio, Systems Manager creó el rol `AWSServiceRoleForSystemsManagerNotifications` en su cuenta. Para obtener más información, consulte Un nuevo rol ha aparecido en mi cuenta de IAM.

Si elimina este rol vinculado a servicios y necesita crearlo de nuevo, puede utilizar el mismo proceso para volver a crear el rol en su cuenta. Cuando habilita el acceso a los nodos justo a tiempo en la AWS Management Console, Systems Manager se encarga de volver a crear el rol vinculado al servicio.

También puede utilizar la consola de IAM para crear un rol vinculado a un servicio con el caso de uso en el que el rol de servicio de AWS permite que Systems Manager envíe notificaciones sobre el acceso a los aprobadores de las solicitudes. En la AWS CLI o la API de AWS, cree un rol vinculado al servicio con el nombre de servicio `ssm.amazonaws.com`. Para obtener más información, consulte Creación de un rol vinculado a un servicio en la Guía del usuario de IAM. Si elimina este rol vinculado al servicio, puede utilizar este mismo proceso para volver a crear el rol.

Cómo modificar el rol vinculado al servicio **AWSServiceRoleForSystemsManagerNotifications** de Systems Manager

Systems Manager no le permite modificar el rol vinculado al servicio `AWSServiceRoleForSystemsManagerNotifications`. Después de crear un rol vinculado a un

servicio, no puede cambiarle el nombre, ya que varias entidades pueden hacer referencia a él. Sin embargo, puede editar la descripción del rol mediante IAM. Para obtener más información, consulte Editar un rol vinculado a servicios en la Guía del usuario de IAM.

Cómo eliminar el rol vinculado al servicio **AWSServiceRoleForSystemsManagerNotifications** de Systems Manager

Si ya no necesita usar una característica o servicio que requieran un rol vinculado a un servicio, le recomendamos que elimine dicho rol. De esta forma no conservará una entidad no utilizada que no se monitorice ni se mantenga de forma activa. Sin embargo, debe limpiar los recursos de su rol vinculado al servicio antes de eliminarlo manualmente.

> **Note**
>
> Si el servicio Systems Manager está utilizando el rol cuando intenta eliminar los recursos, la eliminación podría producir un error. En tal caso, espere unos minutos e intente de nuevo la operación.

Para eliminar manualmente el rol vinculado al servicio **AWSServiceRoleForSystemsManagerNotifications** mediante IAM

Puede usar la consola de IAM, la AWS CLI o la API de AWS para eliminar el rol vinculado a un servicio de `AWSServiceRoleForSystemsManagerNotifications`. Para obtener más información, consulte Eliminar un rol vinculado a un servicio en la Guía del usuario de IAM.

Regiones admitidas para el rol vinculado al servicio **AWSServiceRoleForSystemsManagerNotifications** de Systems Manager

Región de AWS name	Identidad de la región	Compatibilidad en Systems Manager
Este de EE. UU. (Norte de Virginia)	us-east-1	Sí
Este de EE. UU. (Ohio)	us-east-2	Sí
Oeste de EE. UU. (Norte de California)	us-west-1	Sí
Oeste de EE. UU. (Oregón)	us-west-2	Sí

Región de AWS name	Identidad de la región	Compatibilidad en Systems Manager
Asia-Pacífico (Bombay)	ap-south-1	Sí
Asia Pacífico (Osaka)	ap-northeast-3	Sí
Asia-Pacífico (Seúl)	ap-northeast-2	Sí
Asia-Pacífico (Singapur)	ap-southeast-1	Sí
Asia-Pacífico (Sídney)	ap-southeast-2	Sí
Asia-Pacífico (Tokio)	ap-northeast-1	Sí
Canadá (centro)	ca-central-1	Sí
Europa (Fráncfort)	eu-central-1	Sí
Europa (Irlanda)	eu-west-1	Sí
Europa (Londres)	eu-west-2	Sí
Europa (París)	eu-west-3	Sí
Europa (Estocolmo)	eu-north-1	Sí
América del Sur (São Paulo)	sa-east-1	Sí
AWS GovCloud (US)	us-gov-west-1	No

Registro y monitorización en AWS Systems Manager

La supervisión es un aspecto importante del mantenimiento de la fiabilidad, la disponibilidad y el rendimiento de AWS Systems Manager y sus soluciones de AWS. Debe recopilar datos de monitoreo de todas las partes de su solución de AWS para que pueda depurar un error multipunto si se produce. AWS proporciona varias herramientas para monitorear sus recursos de Systems Manager y otros recursos, y responder a posibles incidentes.

Registros de AWS CloudTrail

CloudTrail proporciona un registro de las acciones que realiza un usuario, un rol o un Servicio de AWS en Systems Manager. Mediante la información recopilada por CloudTrail, puede determinar la solicitud que se realizó a Systems Manager, la dirección IP desde la que se realizó, quién la realizó y cuándo, etc. Para obtener más información, consulte Registro de llamadas a la API de AWS Systems Manager con AWS CloudTrail.

Alarmas de Amazon CloudWatch

Las alarmas de Amazon CloudWatch permiten vigilar una sola métrica durante un periodo que especifique para las instancias de Amazon Elastic Compute Cloud (Amazon EC2) y otros recursos. Si la métrica supera un límite determinado, se envía una notificación a un tema de Amazon Simple Notification Service (Amazon SNS) o a una política de AWS Auto Scaling. Las alarmas de CloudWatch no invocan acciones porque se encuentren en un estado particular. En su lugar, el estado debe haber cambiado y debe mantenerse durante el número de periodos especificado. Para obtener más información, consulte Uso de las alarmas de Amazon CloudWatch en la Guía del usuario de Amazon CloudWatch.

Paneles de Amazon CloudWatch

Los paneles de CloudWatch son páginas de inicio personalizables en la consola de CloudWatch que puede utilizar para monitorear sus recursos en una vista única, incluso aquellos que se reparten entre diferentes Regiones de AWS. Puede utilizar los paneles de CloudWatch para crear vistas personalizadas de las métricas y las alarmas para sus recursos de AWS. Para obtener más información, consulte Uso de paneles de Amazon CloudWatch alojados por Systems Manager.

Amazon EventBridge

Con Amazon EventBridge, puede configurar reglas para recibir una alerta de los cambios que se produzcan en los recursos de Systems Manager y para dirigir a EventBridge para que realice acciones basadas en el contenido de esos eventos. EventBridge admite una serie de eventos emitidos por varias herramientas de Systems Manager. Para obtener más información, consulte Monitoreo de eventos de Systems Manager con Amazon EventBridge.

Registros de SSM Agent y Amazon CloudWatch Logs

El SSM Agent escribe información acerca de ejecuciones, acciones programadas, errores y estados en los archivos de registro en cada nodo. Puede ver archivos de registro conectándose manualmente a un nodo. Se recomienda enviar automáticamente los datos de registro del agente a un grupo de registros de CloudWatch Logs para su análisis. Para obtener más información,

consulte Envío de registros de nodos a los Registros de CloudWatch (agente de CloudWatch) unificado y Visualización de registros de SSM Agent.

Conformidad de AWS Systems Manager

Puede utilizar Cumplimiento, una herramienta de AWS Systems Manager, para analizar la flota de nodos administrados en busca de cumplimiento de revisiones e incoherencias de configuración. Puede recopilar y agregar datos de varias Cuentas de AWS y Regiones de AWS, y luego desglosarlas en recursos específicos que no sean conformes. De forma predeterminada, Cumplimiento muestra datos de cumplimiento actuales sobre la aplicación de revisiones en Patch Manager, una herramienta de AWS Systems Manager, y asociaciones en State Manager, una herramienta de AWS Systems Manager. Para obtener más información, consulte Conformidad de AWS Systems Manager.

AWS Systems Manager Explorer

Explorer, una herramienta de AWS Systems Manager, es un panel de operaciones personalizable que transmite información sobre sus recursos de AWS. Explorer muestra una vista agregada de los datos de operaciones (OpsData) de sus Cuentas de AWS y en todas las Regiones de AWS. En Explorer, OpsData incluye metadatos sobre instancias EC2, detalles de conformidad de revisiones y elementos de trabajo operativos (OpsItems). Explorer proporciona un contexto sobre cómo OpsItems se distribuyen entre las unidades de negocio o las aplicaciones, cómo se presentan a lo largo del tiempo y cómo varían según la categoría. Puede agrupar y filtrar la información en Explorer para centrarse en los elementos que son relevantes para usted y que requieren que se tomen medidas. Para obtener más información, consulte AWS Systems Manager Explorer.

AWS Systems Manager OpsCenter

OpsCenter, una herramienta de AWS Systems Manager, proporciona una ubicación central en la que los ingenieros de operaciones y los profesionales de TI pueden ver, investigar y resolver elementos de trabajo operativos (OpsItems) relacionados con los recursos de AWS. OpsCenter agrega y estandariza OpsItems en todos los servicios, al tiempo que proporciona datos de investigación contextuales sobre cada OpsItem, OpsItems relacionados y recursos relacionados. OpsCenter también proporciona manuales de procedimientos de Automatización, una herramienta de AWS Systems Manager, que puede utilizar para resolver problemas rápidamente. OpsCenter se integra a Amazon EventBridge. Esto significa que puede crear reglas de EventBridge que generan automáticamente OpsItems para cualquier Servicio de AWS que publica eventos en EventBridge. Para obtener más información, consulte AWS Systems Manager OpsCenter.

Amazon Simple Notification Service

Puede configurar Amazon Simple Notification Service (Amazon SNS) para que envíe notificaciones sobre el estado de los comandos que envía a través de Run Command o Maintenance Windows, herramientas de AWS Systems Manager. Amazon SNS coordina y administra el envío y la entrega de las notificaciones a los clientes o puntos de enlace que estén suscritos a temas de Amazon SNS. Puede recibir una notificación siempre que un comando cambie a un nuevo estado o a un estado específico, como, por ejemplo, `Failed` o `Timed Out`. En los casos en que un comando se envía a varios nodos, puede recibir una notificación por cada copia del comando enviada a un nodo concreto. Para obtener más información, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.

AWS Trusted Advisor y AWS Health Dashboard

Trusted Advisor aprovecha las prácticas recomendadas aprendidas al atender a cientos de miles de clientes de AWS. Trusted Advisor inspecciona su entorno de AWS y realiza recomendaciones cuando surge la oportunidad de ahorrar dinero, mejorar el rendimiento y la disponibilidad del sistema o ayudar a cerrar deficiencias de seguridad. Todos los clientes de AWS tienen acceso a cinco comprobaciones de Trusted Advisor. Los clientes con un plan Business o Enterprise de AWS Support pueden ver todas las verificaciones de Trusted Advisor. Para obtener más información, consulte AWS Trusted Advisor en la Guía del usuario de AWS Support y la Guía del usuario de AWS Health.

Más información

- Registro y monitorización en AWS Systems Manager

Validación de conformidad en AWS Systems Manager

En este tema se aborda la conformidad de AWS Systems Manager con los programas de garantía de terceros. Para obtener información sobre cómo ver los datos de conformidad de los nodos administrados, consulte Conformidad de AWS Systems Manager.

Auditores externos evalúan la seguridad y la conformidad de Systems Manager como parte de varios programas de conformidad de AWS. Estos incluyen SOC, PCI, FedRAMP, HIPAA y otros.

Para obtener una lista de Servicios de AWS en el ámbito de programas de conformidad específicos, consulte Servicios de AWS en el ámbito del programa de conformidad. Para obtener información general, consulte Programas de conformidad de AWS.

Puede descargar los informes de auditoría de terceros utilizando AWS Artifact. Para obtener más información, consulte Descarga de informes en AWS Artifact.

Su responsabilidad de conformidad al utilizar Systems Manager se determina en función de la sensibilidad de los datos, los objetivos de cumplimiento de su empresa y la legislación y los reglamentos correspondientes. AWS proporciona los siguientes recursos para ayudar con la conformidad:

- Guías de inicio rápido de seguridad y conformidad: estas guías de implementación tratan consideraciones sobre arquitectura y ofrecen pasos para implementar los entornos de referencia centrados en la seguridad y la conformidad en AWS.
- Documento técnico sobre arquitectura para seguridad y conformidad de HIPAA: en este documento técnico, se describe cómo las empresas pueden utilizar AWS para crear aplicaciones conformes con HIPAA.
- Recursos de conformidad de AWS: este conjunto de manuales y guías podría aplicarse a su sector y ubicación.
- Evaluación de recursos con reglas en la Guía para desarrolladores de AWS Config: el servicio de AWS Config evalúa en qué medida las configuraciones de sus recursos cumplen las prácticas internas, las directrices del sector y las normativas.
- AWS Security Hub: este Servicio de AWS proporciona una vista integral de su estado de seguridad en AWS que lo ayuda a verificar la conformidad con los estándares y las prácticas recomendadas del sector de seguridad.

Resiliencia en AWS Systems Manager

La infraestructura global de AWS se divide en Regiones de AWS y zonas de disponibilidad. Las Regiones de AWSproporcionan varias zonas de disponibilidad físicamente independientes y aisladas que se encuentran conectadas mediante redes con un alto nivel de rendimiento y redundancia, además de baja latencia. Con las zonas de disponibilidad, puede diseñar y utilizar aplicaciones y bases de datos que conmutan automáticamente entre zonas sin interrupción. Las zonas de disponibilidad tienen una mayor disponibilidad, tolerancia a errores y escalabilidad que las infraestructuras tradicionales de uno o varios centros de datos.

Para obtener más información sobre las Regiones de AWS y las zonas de disponibilidad, consulte la Infraestructura global de AWS.

Seguridad de la infraestructura en AWS Systems Manager

Como se trata de un servicio administrado, AWS Systems Manager está protegido por la seguridad de red global de AWS. Para obtener información sobre los servicios de seguridad de AWS y cómo AWSprotege la infraestructura, consulte Seguridad en la nube de AWS. Para diseñar su entorno de AWS con las prácticas recomendadas de seguridad de infraestructura, consulte Protección de la infraestructura en Portal de seguridad de AWS Well-Architected Framework.

Puede utilizar llamadas a la API publicadas en AWS para obtener acceso a Systems Manager a través de la red. Los clientes deben admitir lo siguiente:

- Seguridad de la capa de transporte (TLS). Exigimos TLS 1.2 y recomendamos TLS 1.3.
- Conjuntos de cifrado con confidencialidad directa total (PFS) como DHE (Ephemeral Diffie-Hellman) o ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). La mayoría de los sistemas modernos como Java 7 y posteriores son compatibles con estos modos.

Además, las solicitudes deben estar firmadas mediante un ID de clave de acceso y una clave de acceso secreta que esté asociada a una entidad de seguridad de IAM principal. También puede utilizar AWS Security Token Service (AWS STS) para generar credenciales de seguridad temporales para firmar solicitudes.

Configuración y análisis de vulnerabilidades en AWS Systems Manager

AWS gestiona las tareas de seguridad básicas, como la configuración del firewall y la recuperación de desastres. Estos procedimientos han sido revisados y certificados por los terceros pertinentes. Para obtener más detalles, consulte los siguientes recursos de :

- Validación de conformidad en AWS Systems Manager
- Modelo de responsabilidad compartida
- Prácticas recomendadas para seguridad, identidad y conformidad

Prácticas recomendadas de seguridad para Systems Manager

AWS Systems Manager proporciona un número de características de seguridad que debe tener en cuenta a la hora de desarrollar e implementar sus propias políticas de seguridad. Las siguientes

prácticas recomendadas son directrices generales y no constituyen una solución de seguridad completa. Puesto que es posible que estas prácticas recomendadas no sean adecuadas o suficientes para el entorno, considérelas como consideraciones útiles en lugar de como normas.

Temas

- Prácticas recomendadas preventivas de seguridad de Systems Manager
- Prácticas recomendadas de monitorización y auditoría de Systems Manager

Prácticas recomendadas preventivas de seguridad de Systems Manager

Las siguientes prácticas recomendadas para Systems Manager pueden serle de utilidad para evitar incidentes de seguridad.

Implementación del acceso a los privilegios mínimos

Cuando concede permisos, debe decidir a quién concede cada permiso y para qué recurso de Systems Manager se lo concede. Permita las acciones específicas que desea permitir en estos recursos. Por lo tanto, debe conceder únicamente los permisos obligatorios para realizar una tarea. La implementación del acceso con privilegios mínimos es esencial a la hora de reducir los riesgos de seguridad y el impacto que podrían causar los errores o los intentos malintencionados.

Las siguientes herramientas están disponibles para implementar el acceso a los privilegios mínimos:

- Políticas de IAM y Límites de permisos para las entidades de IAM
- Políticas de control de servicios

Uso de la configuración recomendada para SSM Agent cuando esté configurado para usar un proxy

Si configura SSM Agent para usar un proxy, utilice la variable `no_proxy` con la dirección IP del servicio de metadatos de la instancia de Systems Manager para garantizar que las llamadas a Systems Manager no adopten la identidad del servicio de proxy.

Para obtener más información, consulte Configuración de SSM Agent para utilizar un proxy en nodos de Linux y Configurar el SSM Agent para usar un proxy para las instancias de Windows Server.

Uso de parámetros SecureString para cifrar y proteger datos secretos

En Parameter Store, una herramienta de AWS Systems Manager, un parámetro `SecureString` es toda información confidencial que debe almacenarse o a la que se hace referencia de

forma segura. Si tiene datos que no desea que los usuarios modifiquen o a los que no deban hacer referencia en texto sin formato, tales como contraseñas o claves de licencias, cree esos parámetros utilizando el tipo de datos `SecureString`. Parameter Store utiliza una AWS KMS key en AWS Key Management Service (AWS KMS) para cifrar el valor del parámetro. AWS KMS utiliza una clave administrada por el cliente o una Clave administrada de AWS cuando cifra el valor del parámetro. Para lograr la máxima seguridad, le recomendamos usar su propia clave de KMS. Si utiliza la Clave administrada de AWS, cualquier usuario con permiso para ejecutar las acciones GetParameter y GetParameters en su cuenta podrá ver o recuperar el contenido de todos los parámetros de `SecureString`. Si utiliza claves administradas por el cliente para cifrar los valores seguros `SecureString`, puede usar políticas de IAM y políticas de claves para administrar los permisos para cifrar y descifrar parámetros.

Es más difícil establecer políticas de control de acceso para estas operaciones si se utiliza una Clave administrada de AWS. Por ejemplo, si utiliza una Clave administrada de AWS para cifrar parámetros `SecureString` y no desea que los usuarios trabajen con parámetros `SecureString`, sus políticas de IAM deben denegar explícitamente el acceso a la clave predeterminada.

Para obtener más información, consulte Restringir el acceso a los parámetros de Parameter Store mediante políticas de IAM y Uso de AWS Systems ManagerParameter Store de AWS KMS en la Guía para desarrolladores de AWS Key Management Service.

Definición de allowedValues y allowedPattern para parámetros de documentos

Puede validar la entrada del usuario para los parámetros de documentos de Systems Manager (documentos de SSM) mediante la definición de `allowedValues` y `allowedPattern`. En `allowedValues`, debe definir una matriz de valores permitidos para el parámetro. Si un usuario introduce un valor que no está permitido, la ejecución no se iniciará. En `allowedPattern`, debe definir una expresión regular que valide si la entrada del usuario coincide con el patrón definido para el parámetro. Si la entrada del usuario no coincide con el patrón permitido, la ejecución no se iniciará.

Para obtener más información sobre `allowedValues` y `allowedPattern`, consulte Elementos y parámetros de datos.

Bloqueo del uso compartido público de documentos

A menos que su caso de uso requiera que se permita el uso compartido público, le recomendamos que active la configuración de bloqueo de uso compartido público para sus

documentos de SSM en la sección Preferences (Preferencias) en la consola de documentos de Systems Manager.

Uso de una Amazon Virtual Private Cloud (Amazon VPC) y puntos de enlace de la VPC

Puede usar Amazon VPC para lanzar recursos de AWS en una red virtual que haya definido. Esta red virtual es muy similar a la red tradicional que usaría en su propio centro de datos, pero con los beneficios que supone utilizar la infraestructura escalable de AWS.

Implementar un punto de conexión de VPC permite conectar de forma privada su VPC a los Servicios de AWS y a los servicios de punto de conexión de VPC basados en AWS PrivateLink compatibles sin necesidad de una puerta de enlace de Internet, un dispositivo NAT, una conexión de VPN ni una conexión de AWS Direct Connect. Las instancias de su VPC no necesitan direcciones IP públicas para comunicarse con los recursos del servicio. El tráfico entre su VPC y el servicio no sale de la red de Amazon.

Para obtener más información sobre la seguridad de Amazon VPC, consulte Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager y Privacidad del tráfico entre redes en Amazon VPC en la Guía del usuario de Amazon VPC.

Restrinja a los usuarios de Session Manager a las sesiones mediante comandos interactivos y documentos de sesión SSM específicos.

Session Manager, una herramienta de AWS Systems Manager, proporciona varios métodos para comenzar sesiones en los nodos administrados. Para las conexiones más seguras, puede requerir que los usuarios se conecten con el método de comandos interactivos para limitar la interacción del usuario a un comando o secuencia de comandos específicos. Esto le ayuda a administrar las acciones interactivas que puede realizar un usuario. Para obtener más información, consulte Inicio de una sesión (comandos interactivos y no interactivos).

Para mayor seguridad, puede limitar el acceso de instancias específicas de Amazon EC2 a Session Manager y a documentos de sesión específicos de Session Manager. Puede conceder o revocar el acceso a Session Manager de esta manera mediante políticas AWS Identity and Access Management (IAM). Para obtener más información, consulte Paso 3: controlar el acceso de la sesión a los nodos administrados.

Proporción de permisos de nodos temporales para flujos de trabajo de Automation

Durante un flujo de trabajo de Automatización, una herramienta de AWS Systems Manager, es posible que los nodos necesiten permisos que solo se requieran para esa ejecución, pero no para otras operaciones de Systems Manager. Por ejemplo, un flujo de trabajo de Automation puede requerir un nodo para llamar a una determinada operación de la API o acceder a un recurso de

AWS en particular durante el flujo de trabajo. Si estas llamadas o recursos son aquellos a los que desea limitar el acceso, puede proporcionar permisos temporales y suplementarios para los nodos dentro del propio manual de procedimientos de Automation, en lugar de agregar los permisos al perfil de instancias de IAM. Al final del flujo de trabajo de Automation, se retiran los permisos temporales. Para obtener más información, consulte Cómo proporcionar permisos de instancia temporales con Automations de AWS Systems Manager en el Blog de administración y gobernanza de AWS.

Mantención de las herramientas de AWS y Systems Manager actualizadas

AWS publica periódicamente versiones actualizadas de herramientas y complementos que puede usar en su operaciones de AWS y Systems Manager. Mantener estos recursos actualizados garantiza que los usuarios y los nodos de la cuenta tengan acceso a la funcionalidad y las características de seguridad más recientes de estas herramientas.

- SSM Agent: AWS Systems Manager Agent (SSM Agent) es el software de Amazon que se puede instalar y configurar en una instancia de Amazon Elastic Compute Cloud (Amazon EC2), un servidor local o en una máquina virtual. SSM Agent permite que Systems Manager actualice, administre y configure estos recursos. Recomendamos comprobar si hay nuevas versiones o automatizar las actualizaciones del agente, al menos cada dos semanas. Para obtener más información, consulta Automatización de las actualizaciones de SSM Agent. También recomendamos verificar la firma de SSM Agent como parte de su proceso de actualización. Para obtener más información, consulta Verificación de la firma de SSM Agent.
- AWS CLI: la AWS Command Line Interface (AWS CLI) es una herramienta de código abierto que le permite interactuar con Servicios de AWS mediante el uso de comandos en el shell de la línea de comandos. Para actualizar la AWS CLI, ejecute el mismo comando utilizado para instalar la AWS CLI. Recomendamos crear una tarea programada en el equipo local que ejecute el comando apropiado para su sistema operativo al menos una vez cada dos semanas. Para obtener información acerca de la instalación de comandos, consulte Instalación de la versión 2 de AWS CLI en la Guía del usuario de AWS Command Line Interface.
- AWS Tools for Windows PowerShell: las Tools for Windows PowerShell son un conjunto de módulos de PowerShell basados en la funcionalidad expuesta por el AWS SDK para .NET. Las AWS Tools for Windows PowerShell permiten realizar operaciones mediante scripts en sus recursos de AWS desde la línea de comandos de PowerShell. De forma periódica, cuando se publiquen versiones actualizadas de Tools for Windows PowerShell, debería actualizar la versión que ejecuta localmente. Para obtener información, consulte Actualización de AWS Tools for Windows PowerShell en Windows o Actualización de AWS Tools for Windows PowerShell en Linux o macOS en la Guía del usuario del simulador de políticas de IAM.

- Complemento de Session Manager: si los usuarios de la organización con permisos para utilizar Session Manager desean conectarse a un nodo mediante la AWS CLI, antes deben instalar el complemento de Session Manager en sus equipos locales. Para actualizar el complemento, ejecute el mismo comando utilizado para instalarlo. Recomendamos crear una tarea programada en el equipo local que ejecute el comando apropiado para su sistema operativo al menos una vez cada dos semanas. Para obtener más información, consulta Instalación del complemento de Session Manager para la AWS CLI.
- Agente de CloudWatch: puede configurar y utilizar el agente de CloudWatch para recopilar métricas y registros de las instancias EC2, las instancias locales y las máquinas virtuales. Estos registros se pueden enviar a los Registros de Amazon CloudWatch para su monitoreo y análisis. Recomendamos comprobar si hay nuevas versiones o automatizar las actualizaciones del agente, al menos cada dos semanas. Para las actualizaciones más simples, use la Configuración Rápida de AWS Systems Manager. Para obtener más información, consulta AWS Systems Manager Quick Setup.

Prácticas recomendadas de monitorización y auditoría de Systems Manager

Las siguientes prácticas recomendadas para Systems Manager le pueden ser de utilidad para detectar los incidentes y los posibles puntos débiles de la seguridad.

Identificación y auditoría de todos los recursos de Systems Manager

La identificación de sus activos de TI es un aspecto fundamental de seguridad y control. Tiene que identificar todos sus recursos de Systems Manager para evaluar sus medidas de seguridad y tomar así las medidas pertinentes respecto a las posibles áreas de debilidad.

Utilice Tag Editor para identificar los recursos que precisan más seguridad o una auditoría y utilice dichas etiquetas cuando tenga que buscarlos. Para obtener más información, consulte Búsqueda de recursos para etiquetar en la Guía del usuario de AWS Resource Groups.

Cree grupos de recursos para sus recursos de Systems Manager. Para obtener más información, consulte What are resource groups?

Implementación del monitoreo mediante las herramientas de monitoreo de Amazon CloudWatch

El monitoreo es una parte importante del mantenimiento de la fiabilidad, la seguridad, la disponibilidad y el rendimiento de Systems Manager y sus soluciones de AWS. Amazon

CloudWatch ofrece varias herramientas y servicios para ayudarlo a supervisar Systems Manager y sus otros Servicios de AWS. Para obtener más información, consulte Envío de registros de nodos a los Registros de CloudWatch (agente de CloudWatch) unificado y Monitoreo de eventos de Systems Manager con Amazon EventBridge.

Uso de CloudTrail

AWS CloudTrail proporciona un registro de las medidas adoptadas por un usuario, un rol o un Servicio de AWS en Systems Manager. Mediante la información recopilada por CloudTrail, puede determinar la solicitud que se realizó a Systems Manager, la dirección IP desde la que se realizó, quién la realizó y cuándo, etc. Para obtener más información, consulte Registro de llamadas a la API de AWS Systems Manager con AWS CloudTrail.

Activar AWS Config

AWS Config le permite examinar, auditar y evaluar las configuraciones de sus recursos de AWS. AWS Config monitorea las configuraciones de los recursos, lo que le permite evaluar las configuraciones registradas respecto a las configuraciones de seguridad requeridas. Con AWS Config, puede revisar los cambios en las configuraciones y las relaciones entre los recursos de AWS, investigar los historiales detallados de configuración de recursos y determinar la conformidad general con respecto a las configuraciones especificadas en las pautas internas. Esto le puede ser de utilidad para simplificar las auditorías de conformidad, los análisis de seguridad, la administración de cambios y la resolución de problemas operativos. Para obtener más información, consulte Configuración de AWS Config mediante la consola en la Guía para desarrolladores de AWS Config. Al especificar los tipos de recursos para registras, asegúrese de incluir los recursos de Systems Manager.

Monitoreo de los avisos de seguridad de AWS

Debe comprobar con regularidad los avisos sobre seguridad publicados en Trusted Advisor para su Cuenta de AWS. Puede hacer esto mediante programación con describe-trusted-advisor-checks.

Además, supervise de forma activa la dirección principal de email registrada en cada una de sus Cuentas de AWS. AWS contactará con usted, a través de esta dirección de email, para informarle sobre los problemas de seguridad que surjan y que pudieran afectarle.

Los problemas operativos de AWS con gran alcance se publican en AWS Service Health Dashboard. Los problemas operativos también se publican en las cuentas individuales a través del Personal Health Dashboard. Para obtener más información, consulte la documentación de AWS Health.

Más información

- Prácticas recomendadas para seguridad, identidad y conformidad
- Introducción: siga las prácticas recomendadas de seguridad a medida que se configuran los recursos de AWS (Blog de seguridad de AWS)
- Security best practices in IAM (Prácticas recomendadas de seguridad en IAM)
- Prácticas recomendadas de seguridad de AWS CloudTrail
- Prácticas recomendadas de seguridad para Amazon S3
- Prácticas recomendadas de seguridad para AWS Key Management Service

Ejemplos de código de Systems Manager usando SDK de AWS

En los siguientes ejemplos de código se muestra cómo utilizar Systems Manager con un kit de desarrollo de software (SDK) de AWS.

Los conceptos básicos son ejemplos de código que muestran cómo realizar las operaciones esenciales dentro de un servicio.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Las acciones muestran cómo llamar a las funciones de servicio individuales y es posible verlas en contexto en los escenarios relacionados.

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Introducción

Hola, Systems Manager

En los siguientes ejemplos de código, se muestra cómo empezar a utilizar Systems Manager.

Java

SDK para Java 2.x

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de 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);
    }
}
```

- Para obtener más información acerca de la API, consulte ListDocuments en la Referencia de la API de AWS SDK for Java 2.x.

JavaScript

SDK para JavaScript (v3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
import { paginateListDocuments, SSMClient } from "@aws-sdk/client-ssm";

// Call ListDocuments and display the result.
export const main = async () => {
  const client = new SSMClient();
  const listDocumentsPaginated = [];
  console.log(
    "Hello, AWS Systems Manager! Let's list some of your documents:\n",
  );
  try {
    // The paginate function is a wrapper around the base command.
    const paginator = paginateListDocuments({ client }, { MaxResults: 5 });
    for await (const page of paginator) {
      listDocumentsPaginated.push(...page.DocumentIdentifiers);
    }
  } catch (caught) {
```

```
    console.error(`There was a problem saying hello: ${caught.message}`);
    throw caught;
  }

  for (const { Name, DocumentFormat, CreatedDate } of listDocumentsPaginated) {
    console.log(`${Name} - ${DocumentFormat} - ${CreatedDate}`);
  }
};

// Call function if run directly.
import { fileURLToPath } from "node:url";
if (process.argv[1] === fileURLToPath(import.meta.url)) {
  main();
}
```

- Para obtener más información acerca de la API, consulte ListDocuments en la Referencia de la API de AWS SDK para JavaScript.

Python

SDK para Python (Boto3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
import boto3
from botocore.exceptions import ClientError


def hello_systems_manager(ssm_client):
    """
    Use the AWS SDK for Python (Boto3) to create an AWS Systems Manager
    client and list the first 5 documents in your account.
    This example uses the default settings specified in your shared credentials
    and config files.
```

```
    :param ssm_client: A Boto3 AWS Systems Manager Client object. This object
 wraps
                            the low-level AWS Systems Manager service API.
    """
    print("Hello, AWS Systems Manager! Let's list some of your documents:\n")

    paginator = ssm_client.get_paginator("list_documents")
    page_iterator = paginator.paginate(PaginationConfig={"MaxItems": 5})
    for page in page_iterator:
        for document in page["DocumentIdentifiers"]:
            print(f"  {document['Name']}")


if __name__ == "__main__":
    try:
        hello_systems_manager(boto3.client("ssm"))
    except ClientError as err:
        print("Hello systems manager had an error.")
        print(err.response["Error"]["Code"])
        print(err.response["Error"]["Message"])
```

- Para obtener información sobre la API, consulte ListDocuments en la Referencia de la API de AWS SDK para Python (Boto3).

Ejemplos de código

- Ejemplos básicos para Systems Manager mediante los SDK de AWS
  - Hola, Systems Manager
  - Conceptos básicos de Systems Manager con un SDK de AWS
  - Acciones para Systems Manager usando SDK de AWS
    - Utilizar AddTagsToResource con una CLI
    - Utilizar CancelCommand con una CLI
    - Utilizar CreateActivation con una CLI
    - Utilizar CreateAssociation con una CLI
    - Utilizar CreateAssociationBatch con una CLI
    - Uso de CreateDocument con un SDK de AWS o la CLI
    - Uso de CreateMaintenanceWindow con un SDK de AWS o la CLI

- Uso de CreateOpsItem con un SDK de AWS o la CLI
- Utilizar CreatePatchBaseline con una CLI
- Utilizar DeleteActivation con una CLI
- Utilizar DeleteAssociation con una CLI
- Uso de DeleteDocument con un SDK de AWS o la CLI
- Uso de DeleteMaintenanceWindow con un SDK de AWS o la CLI
- Cómo utilizar DeleteOpsItem con un AWS SDK
- Utilizar DeleteParameter con una CLI
- Utilizar DeletePatchBaseline con una CLI
- Utilizar DeregisterManagedInstance con una CLI
- Utilizar DeregisterPatchBaselineForPatchGroup con una CLI
- Utilizar DeregisterTargetFromMaintenanceWindow con una CLI
- Utilizar DeregisterTaskFromMaintenanceWindow con una CLI
- Utilizar DescribeActivations con una CLI
- Utilizar DescribeAssociation con una CLI
- Utilizar DescribeAssociationExecutionTargets con una CLI
- Utilizar DescribeAssociationExecutions con una CLI
- Utilizar DescribeAutomationExecutions con una CLI
- Utilizar DescribeAutomationStepExecutions con una CLI
- Utilizar DescribeAvailablePatches con una CLI
- Utilizar DescribeDocument con una CLI
- Utilizar DescribeDocumentPermission con una CLI
- Utilizar DescribeEffectiveInstanceAssociations con una CLI
- Utilizar DescribeEffectivePatchesForPatchBaseline con una CLI
- Utilizar DescribeInstanceAssociationsStatus con una CLI
- Utilizar DescribeInstanceInformation con una CLI
- Utilizar DescribeInstancePatchStates con una CLI
- Utilizar DescribeInstancePatchStatesForPatchGroup con una CLI
- Utilizar DescribeInstancePatches con una CLI

- Utilizar DescribeMaintenanceWindowExecutionTaskInvocations con una CLI

- Utilizar DescribeMaintenanceWindowExecutionTasks con una CLI
- Utilizar DescribeMaintenanceWindowExecutions con una CLI
- Utilizar DescribeMaintenanceWindowTargets con una CLI
- Utilizar DescribeMaintenanceWindowTasks con una CLI
- Utilizar DescribeMaintenanceWindows con una CLI
- Uso de DescribeOpsItems con un SDK de AWS o la CLI
- Uso de DescribeParameters con un SDK de AWS o la CLI
- Utilizar DescribePatchBaselines con una CLI
- Utilizar DescribePatchGroupState con una CLI
- Utilizar DescribePatchGroups con una CLI
- Utilizar GetAutomationExecution con una CLI
- Utilizar GetCommandInvocation con una CLI
- Utilizar GetConnectionStatus con una CLI
- Utilizar GetDefaultPatchBaseline con una CLI
- Utilizar GetDeployablePatchSnapshotForInstance con una CLI
- Utilizar GetDocument con una CLI
- Utilizar GetInventory con una CLI
- Utilizar GetInventorySchema con una CLI
- Utilizar GetMaintenanceWindow con una CLI
- Utilizar GetMaintenanceWindowExecution con una CLI
- Utilizar GetMaintenanceWindowExecutionTask con una CLI
- Uso de GetParameter con un SDK de AWS o la CLI
- Utilizar GetParameterHistory con una CLI
- Utilizar GetParameters con una CLI
- Utilizar GetPatchBaseline con una CLI
- Utilizar GetPatchBaselineForPatchGroup con una CLI
- Utilizar ListAssociationVersions con una CLI
- Utilizar ListAssociations con una CLI
- Uso de ListCommandInvocations con un SDK de AWS o la CLI
- Utilizar ListCommands con una CLI

- Utilizar ListComplianceItems con una CLI
- Utilizar ListComplianceSummaries con una CLI
- Utilizar ListDocumentVersions con una CLI
- Utilizar ListDocuments con una CLI
- Utilizar ListInventoryEntries con una CLI
- Utilizar ListResourceComplianceSummaries con una CLI
- Utilizar ListTagsForResource con una CLI
- Utilizar ModifyDocumentPermission con una CLI
- Utilizar PutComplianceItems con una CLI
- Utilizar PutInventory con una CLI
- Uso de PutParameter con un SDK de AWS o la CLI
- Utilizar RegisterDefaultPatchBaseline con una CLI
- Utilizar RegisterPatchBaselineForPatchGroup con una CLI
- Utilizar RegisterTargetWithMaintenanceWindow con una CLI
- Utilizar RegisterTaskWithMaintenanceWindow con una CLI
- Utilizar RemoveTagsFromResource con una CLI
- Uso de SendCommand con un SDK de AWS o la CLI
- Utilizar StartAutomationExecution con una CLI
- Utilizar StartSession con una CLI
- Utilizar StopAutomationExecution con una CLI
- Utilizar UpdateAssociation con una CLI
- Utilizar UpdateAssociationStatus con una CLI
- Utilizar UpdateDocument con una CLI
- Utilizar UpdateDocumentDefaultVersion con una CLI
- Uso de UpdateMaintenanceWindow con un SDK de AWS o la CLI
- Utilizar UpdateManagedInstanceRole con una CLI
- Uso de UpdateOpsItem con un SDK de AWS o la CLI
- Utilizar UpdatePatchBaseline con una CLI

Ejemplos básicos para Systems Manager mediante los SDK de AWS

Los siguientes ejemplos de código muestran cómo utilizar ejemplos básicos de AWS Systems Manager con AWS SDK.

Ejemplos

- Hola, Systems Manager
- Conceptos básicos de Systems Manager con un SDK de AWS
- Acciones para Systems Manager usando SDK de AWS
  - Utilizar AddTagsToResource con una CLI
  - Utilizar CancelCommand con una CLI
  - Utilizar CreateActivation con una CLI
  - Utilizar CreateAssociation con una CLI
  - Utilizar CreateAssociationBatch con una CLI
  - Uso de CreateDocument con un SDK de AWS o la CLI
  - Uso de CreateMaintenanceWindow con un SDK de AWS o la CLI
  - Uso de CreateOpsItem con un SDK de AWS o la CLI
  - Utilizar CreatePatchBaseline con una CLI
  - Utilizar DeleteActivation con una CLI
  - Utilizar DeleteAssociation con una CLI
  - Uso de DeleteDocument con un SDK de AWS o la CLI
  - Uso de DeleteMaintenanceWindow con un SDK de AWS o la CLI
  - Cómo utilizar DeleteOpsItem con un AWS SDK
  - Utilizar DeleteParameter con una CLI
  - Utilizar DeletePatchBaseline con una CLI
  - Utilizar DeregisterManagedInstance con una CLI
  - Utilizar DeregisterPatchBaselineForPatchGroup con una CLI
  - Utilizar DeregisterTargetFromMaintenanceWindow con una CLI
  - Utilizar DeregisterTaskFromMaintenanceWindow con una CLI
  - Utilizar DescribeActivations con una CLI

- Utilizar DescribeAssociation con una CLI
- Utilizar DescribeAssociationExecutionTargets con una CLI
- Utilizar DescribeAssociationExecutions con una CLI
- Utilizar DescribeAutomationExecutions con una CLI
- Utilizar DescribeAutomationStepExecutions con una CLI
- Utilizar DescribeAvailablePatches con una CLI
- Utilizar DescribeDocument con una CLI
- Utilizar DescribeDocumentPermission con una CLI
- Utilizar DescribeEffectiveInstanceAssociations con una CLI
- Utilizar DescribeEffectivePatchesForPatchBaseline con una CLI
- Utilizar DescribeInstanceAssociationsStatus con una CLI
- Utilizar DescribeInstanceInformation con una CLI
- Utilizar DescribeInstancePatchStates con una CLI
- Utilizar DescribeInstancePatchStatesForPatchGroup con una CLI
- Utilizar DescribeInstancePatches con una CLI
- Utilizar DescribeMaintenanceWindowExecutionTaskInvocations con una CLI
- Utilizar DescribeMaintenanceWindowExecutionTasks con una CLI
- Utilizar DescribeMaintenanceWindowExecutions con una CLI
- Utilizar DescribeMaintenanceWindowTargets con una CLI
- Utilizar DescribeMaintenanceWindowTasks con una CLI
- Utilizar DescribeMaintenanceWindows con una CLI
- Uso de DescribeOpsItems con un SDK de AWS o la CLI
- Uso de DescribeParameters con un SDK de AWS o la CLI
- Utilizar DescribePatchBaselines con una CLI
- Utilizar DescribePatchGroupState con una CLI
- Utilizar DescribePatchGroups con una CLI
- Utilizar GetAutomationExecution con una CLI
- Utilizar GetCommandInvocation con una CLI
- Utilizar GetConnectionStatus con una CLI

- Utilizar GetDefaultPatchBaseline con una CLI

- Utilizar GetDeployablePatchSnapshotForInstance con una CLI
- Utilizar GetDocument con una CLI
- Utilizar GetInventory con una CLI
- Utilizar GetInventorySchema con una CLI
- Utilizar GetMaintenanceWindow con una CLI
- Utilizar GetMaintenanceWindowExecution con una CLI
- Utilizar GetMaintenanceWindowExecutionTask con una CLI
- Uso de GetParameter con un SDK de AWS o la CLI
- Utilizar GetParameterHistory con una CLI
- Utilizar GetParameters con una CLI
- Utilizar GetPatchBaseline con una CLI
- Utilizar GetPatchBaselineForPatchGroup con una CLI
- Utilizar ListAssociationVersions con una CLI
- Utilizar ListAssociations con una CLI
- Uso de ListCommandInvocations con un SDK de AWS o la CLI
- Utilizar ListCommands con una CLI
- Utilizar ListComplianceItems con una CLI
- Utilizar ListComplianceSummaries con una CLI
- Utilizar ListDocumentVersions con una CLI
- Utilizar ListDocuments con una CLI
- Utilizar ListInventoryEntries con una CLI
- Utilizar ListResourceComplianceSummaries con una CLI
- Utilizar ListTagsForResource con una CLI
- Utilizar ModifyDocumentPermission con una CLI
- Utilizar PutComplianceItems con una CLI
- Utilizar PutInventory con una CLI
- Uso de PutParameter con un SDK de AWS o la CLI
- Utilizar RegisterDefaultPatchBaseline con una CLI
- Utilizar RegisterPatchBaselineForPatchGroup con una CLI
- Utilizar RegisterTargetWithMaintenanceWindow con una CLI

- Utilizar RegisterTaskWithMaintenanceWindow con una CLI
- Utilizar RemoveTagsFromResource con una CLI
- Uso de SendCommand con un SDK de AWS o la CLI
- Utilizar StartAutomationExecution con una CLI
- Utilizar StartSession con una CLI
- Utilizar StopAutomationExecution con una CLI
- Utilizar UpdateAssociation con una CLI
- Utilizar UpdateAssociationStatus con una CLI
- Utilizar UpdateDocument con una CLI
- Utilizar UpdateDocumentDefaultVersion con una CLI
- Uso de UpdateMaintenanceWindow con un SDK de AWS o la CLI
- Utilizar UpdateManagedInstanceRole con una CLI
- Uso de UpdateOpsItem con un SDK de AWS o la CLI
- Utilizar UpdatePatchBaseline con una CLI

Hola, Systems Manager

En los siguientes ejemplos de código, se muestra cómo empezar a utilizar Systems Manager.

Java

SDK para Java 2.x

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de 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);
    }
}
```

- Para obtener más información acerca de la API, consulte ListDocuments en la Referencia de la API de AWS SDK for Java 2.x.

JavaScript

SDK para JavaScript (v3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
import { paginateListDocuments, SSMClient } from "@aws-sdk/client-ssm";

// Call ListDocuments and display the result.
export const main = async () => {
  const client = new SSMClient();
  const listDocumentsPaginated = [];
  console.log(
    "Hello, AWS Systems Manager! Let's list some of your documents:\n",
  );
  try {
    // The paginate function is a wrapper around the base command.
    const paginator = paginateListDocuments({ client }, { MaxResults: 5 });
    for await (const page of paginator) {
      listDocumentsPaginated.push(...page.DocumentIdentifiers);
    }
```

```
  } catch (caught) {
    console.error(`There was a problem saying hello: ${caught.message}`);
    throw caught;
  }

  for (const { Name, DocumentFormat, CreatedDate } of listDocumentsPaginated) {
    console.log(`${Name} - ${DocumentFormat} - ${CreatedDate}`);
  }
};

// Call function if run directly.
import { fileURLToPath } from "node:url";
if (process.argv[1] === fileURLToPath(import.meta.url)) {
  main();
}
```

- Para obtener más información acerca de la API, consulte ListDocuments en la Referencia de la API de AWS SDK para JavaScript.

Python

SDK para Python (Boto3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
import boto3
from botocore.exceptions import ClientError


def hello_systems_manager(ssm_client):
    """
    Use the AWS SDK for Python (Boto3) to create an AWS Systems Manager
    client and list the first 5 documents in your account.
    This example uses the default settings specified in your shared credentials
    and config files.
```

```
    :param ssm_client: A Boto3 AWS Systems Manager Client object. This object
 wraps
                            the low-level AWS Systems Manager service API.
    """
    print("Hello, AWS Systems Manager! Let's list some of your documents:\n")

    paginator = ssm_client.get_paginator("list_documents")
    page_iterator = paginator.paginate(PaginationConfig={"MaxItems": 5})
    for page in page_iterator:
        for document in page["DocumentIdentifiers"]:
            print(f"  {document['Name']}")


if __name__ == "__main__":
    try:
        hello_systems_manager(boto3.client("ssm"))
    except ClientError as err:
        print("Hello systems manager had an error.")
        print(err.response["Error"]["Code"])
        print(err.response["Error"]["Message"])
```

- Para obtener información sobre la API, consulte ListDocuments en la Referencia de la API de AWS SDK para Python (Boto3).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Conceptos básicos de Systems Manager con un SDK de AWS

En el siguiente ejemplo de código, se muestra cómo:

- Crear un período de mantenimiento.
- Modificar la programación del período de mantenimiento.
- Cree un documento.
- Enviar un comando a una instancia de EC2 especificada.
- Crear un OpsItem.
- Actualizar y resolver el OpsITem.

- Eliminar el período de mantenimiento, el OpsITem y el documento.

Java

SDK para Java 2.x

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
import software.amazon.awssdk.services.ssm.model.DocumentAlreadyExistsException;
import software.amazon.awssdk.services.ssm.model.SsmException;

import java.util.Scanner;
public class SSMScenario {
    public static final String DASHES = new String(new char[80]).replace("\0",
 "-");

    public static void main(String[] args) {
        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);
        SSMActions actions = new SSMActions();
        String documentName;
        String windowName;
        String instanceId = args[0];
        String title = "Disk Space Alert" ;
        String source = "EC2" ;
        String category = "Availability" ;
        String severity = "2" ;

        System.out.println(DASHES);
        System.out.println("""
                Welcome to the AWS Systems Manager SDK Basics scenario.
                This Java program demonstrates how to interact with AWS Systems
Manager using the AWS SDK for Java (v2).
                AWS Systems Manager is the operations hub for your AWS
applications and resources and a secure end-to-end management solution.
                The program's primary functionalities 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 AWS SSM resources.
                Upon completion of the program, all AWS resources are cleaned up.
                Let's get started...

                """);
        waitForInputToContinue(scanner);
        System.out.println(DASHES);

        System.out.println("1. Create an SSM 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 = null;
        try {
            winId = actions.createMaintenanceWindow(windowName);
            waitForInputToContinue(scanner);
            System.out.println("The maintenance window ID is: " + winId);
        } catch (DocumentAlreadyExistsException e) {
            System.err.println("The SSM maintenance window already exists.
Retrieving existing window ID...");
            String existingWinId = actions.createMaintenanceWindow(windowName);
            System.out.println("Existing window ID: " + existingWinId);
        } catch (SsmException e) {
```

```
            System.err.println("SSM error: " + e.getMessage());
            return;
        } catch (RuntimeException e) {
            System.err.println("Unexpected error: " + e.getMessage());
            return;
        }
        waitForInputToContinue(scanner);
        System.out.println(DASHES);

        System.out.println("2. Modify the maintenance window by changing the
schedule");
        waitForInputToContinue(scanner);
        try {
            actions.updateSSMMaintenanceWindow(winId, windowName);
            waitForInputToContinue(scanner);
            System.out.println("The SSM maintenance window was successfully
updated");
        } catch (SsmException e) {
            System.err.println("SSM error: " + e.getMessage());
            return;
        } catch (RuntimeException e) {
            System.err.println("Unexpected error: " + e.getMessage());
            return;
        }
        waitForInputToContinue(scanner);
        System.out.println(DASHES);

        System.out.println("3. Create an SSM document that defines the actions
that Systems Manager performs on your managed nodes.");
        System.out.println("Please enter the document name (default is
ssmdocument):");
        String doc = scanner.nextLine();
        documentName = doc.isEmpty() ? "ssmdocument" : doc;
        try {
            actions.createSSMDoc(documentName);
            waitForInputToContinue(scanner);
            System.out.println("The SSM document was successfully created");
        } catch (DocumentAlreadyExistsException e) {
            System.err.println("The SSM document already exists. Moving on");
        } catch (SsmException e) {
            System.err.println("SSM error: " + e.getMessage());
            return;
        } catch (RuntimeException e) {
            System.err.println("Unexpected error: " + e.getMessage());
```

```
        }
        waitForInputToContinue(scanner);
        System.out.println(DASHES);

        System.out.println("4. Now we are going to run a command on an EC2
instance");
        waitForInputToContinue(scanner);
        String commandId="";
        try {
            commandId = actions.sendSSMCommand(documentName, instanceId);
            waitForInputToContinue(scanner);
            System.out.println("The command was successfully sent. Command ID: "
+ commandId);
        } catch (SsmException e) {
            System.err.println("SSM error: " + e.getMessage());
        } catch (InterruptedException e) {
            System.err.println("Thread was interrupted: " + e.getMessage());
        } catch (RuntimeException e) {
            System.err.println("Unexpected error: " + e.getMessage());
        }
        waitForInputToContinue(scanner);
        System.out.println(DASHES);

        System.out.println("5. Lets get the time when the specific command was
sent to the specific managed node");
        waitForInputToContinue(scanner);
        try {
            actions.displayCommands(commandId);
            System.out.println("The command invocations were successfully
displayed.");
        } catch (SsmException e) {
            System.err.println("SSM error: " + e.getMessage());
            return;
        } catch (RuntimeException e) {
            System.err.println("Unexpected error: " + e.getMessage());
            return;
        }
        waitForInputToContinue(scanner);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("""
             6. Now we will create an SSM OpsItem.
```

```
            A SSM OpsItem is a feature provided by Amazon's Systems Manager
(SSM) 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.
           """);

       waitForInputToContinue(scanner);
       String opsItemId;
       try {
           opsItemId = actions.createSSMOpsItem(title, source, category,
severity);
           System.out.println(opsItemId + " was created");
       } catch (SsmException e) {
           System.err.println("SSM error: " + e.getMessage());
           return;
       } catch (RuntimeException e) {
           System.err.println("Unexpected error: " + e.getMessage());
           return;
       }
       waitForInputToContinue(scanner);
       System.out.println(DASHES);

       System.out.println(DASHES);
       System.out.println("7. Now we will update the SSM OpsItem "+opsItemId);
       waitForInputToContinue(scanner);
       String description = "An update to "+opsItemId ;
       try {
           actions.updateOpsItem(opsItemId, title, description);
       } catch (SsmException e) {
           System.err.println("SSM error: " + e.getMessage());
           return;
       } catch (RuntimeException e) {
           System.err.println("Unexpected error: " + e.getMessage());
           return;
       }

       System.out.println(DASHES);
```

```
        System.out.println("8. Now we will get the status of the SSM OpsItem
"+opsItemId);
        waitForInputToContinue(scanner);
        try {
            actions.describeOpsItems(opsItemId);
        } catch (SsmException e) {
            System.err.println("SSM error: " + e.getMessage());
            return;
        } catch (RuntimeException e) {
            System.err.println("Unexpected error: " + e.getMessage());
            return;
        }

        System.out.println(DASHES);
        System.out.println("9. Now we will resolve the SSM OpsItem "+opsItemId);
        waitForInputToContinue(scanner);
        try {
            actions.resolveOpsItem(opsItemId);
        } catch (SsmException e) {
            System.err.println("SSM error: " + e.getMessage());
            return;
        } catch (RuntimeException e) {
            System.err.println("Unexpected error: " + e.getMessage());
            return;
        }

        System.out.println(DASHES);
        System.out.println("10. Would you like to delete the AWS Systems Manager
resources? (y/n)");
        String delAns = scanner.nextLine().trim();
        if (delAns.equalsIgnoreCase("y")) {
            System.out.println("You selected to delete the resources.");
            waitForInputToContinue(scanner);
            try {
                actions.deleteMaintenanceWindow(winId);
                actions.deleteDoc(documentName);
            } catch (SsmException e) {
                System.err.println("SSM error: " + e.getMessage());
                return;
            } catch (RuntimeException e) {
                System.err.println("Unexpected error: " + e.getMessage());
                return;
            }
        } else {
```

```
            System.out.println("The AWS Systems Manager resources will not be
 deleted");
        }
        System.out.println(DASHES);

        System.out.println("This concludes the AWS Systems Manager SDK Basics
 scenario.");
        System.out.println(DASHES);
    }

    private static void waitForInputToContinue(Scanner scanner) {
        while (true) {
            System.out.println("");
            System.out.println("Enter 'c' followed by <ENTER> to continue:");
            String input = scanner.nextLine();

            if (input.trim().equalsIgnoreCase("c")) {
                System.out.println("Continuing with the program...");
                System.out.println("");
                break;
            } else {
                // Handle invalid input.
                System.out.println("Invalid input. Please try again.");
            }
        }
    }
}
```

Una clase contenedora para métodos del SDK de Systems Manager.

```
public class SSMActions {

    private static SsmAsyncClient ssmAsyncClient;

    private static SsmAsyncClient getAsyncClient() {
        if (ssmAsyncClient == null) {
            SdkAsyncHttpClient httpClient = NettyNioAsyncHttpClient.builder()
                    .maxConcurrency(100)
                    .connectionTimeout(Duration.ofSeconds(60))
                    .readTimeout(Duration.ofSeconds(60))
                    .writeTimeout(Duration.ofSeconds(60))
                    .build();
```

```
            ClientOverrideConfiguration overrideConfig =
ClientOverrideConfiguration.builder()
                    .apiCallTimeout(Duration.ofMinutes(2))
                    .apiCallAttemptTimeout(Duration.ofSeconds(90))
                    .retryPolicy(RetryPolicy.builder()
                            .numRetries(3)
                            .build())
                    .build();

            ssmAsyncClient = SsmAsyncClient.builder()
                    .region(Region.US_EAST_1)
                    .httpClient(httpClient)
                    .overrideConfiguration(overrideConfig)
                    .build();
        }
        return ssmAsyncClient;
    }

    /**
     * Deletes an AWS SSM document asynchronously.
     *
     * @param documentName The name of the document to delete.
     * <p>
     * This method initiates an asynchronous request to delete an SSM document.
     * If an exception occurs, it handles the error appropriately.
     */
    public void deleteDoc(String documentName) {
        DeleteDocumentRequest documentRequest = DeleteDocumentRequest.builder()
                .name(documentName)
                .build();

        CompletableFuture<Void> future = CompletableFuture.runAsync(() -> {
            getAsyncClient().deleteDocument(documentRequest)
                    .thenAccept(response -> {
                        System.out.println("The SSM document was successfully
deleted.");
                    })
                    .exceptionally(ex -> {
                        throw new CompletionException(ex);
                    }).join();
        }).exceptionally(ex -> {
            Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
```

```
            if (cause instanceof SsmException) {
                throw new RuntimeException("SSM error: " + cause.getMessage(),
cause);
            } else {
                throw new RuntimeException("Unexpected error: " +
cause.getMessage(), cause);
            }
        });

        try {
            future.join();
        } catch (CompletionException ex) {
            throw ex.getCause() instanceof RuntimeException ? (RuntimeException)
ex.getCause() : ex;
        }
    }

    /**
     * Deletes an AWS SSM Maintenance Window asynchronously.
     *
     * @param winId The ID of the Maintenance Window to delete.
     * <p>
     * This method initiates an asynchronous request to delete an SSM Maintenance
Window.
     * If an exception occurs, it handles the error appropriately.
     */
    public void deleteMaintenanceWindow(String winId) {
        DeleteMaintenanceWindowRequest windowRequest =
DeleteMaintenanceWindowRequest.builder()
                .windowId(winId)
                .build();

        CompletableFuture<Void> future = CompletableFuture.runAsync(() -> {
            getAsyncClient().deleteMaintenanceWindow(windowRequest)
                    .thenAccept(response -> {
                        System.out.println("The maintenance window was
successfully deleted.");
                    })
                    .exceptionally(ex -> {
                        throw new CompletionException(ex);
                    }).join();
        }).exceptionally(ex -> {
            Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
```

```
            if (cause instanceof SsmException) {
                throw new RuntimeException("SSM error: " + cause.getMessage(),
cause);
            } else {
                throw new RuntimeException("Unexpected error: " +
cause.getMessage(), cause);
            }
        });

        try {
            future.join();
        } catch (CompletionException ex) {
            throw ex.getCause() instanceof RuntimeException ? (RuntimeException)
ex.getCause() : ex;
        }
    }

    /**
     * Resolves an AWS SSM OpsItem asynchronously.
     *
     * @param opsID The ID of the OpsItem to resolve.
     * <p>
     * This method initiates an asynchronous request to resolve an SSM OpsItem.
     * If an exception occurs, it handles the error appropriately.
     */
    public void resolveOpsItem(String opsID) {
        UpdateOpsItemRequest opsItemRequest = UpdateOpsItemRequest.builder()
                .opsItemId(opsID)
                .status(OpsItemStatus.RESOLVED)
                .build();

        CompletableFuture<Void> future = CompletableFuture.runAsync(() -> {
            getAsyncClient().updateOpsItem(opsItemRequest)
                    .thenAccept(response -> {
                        System.out.println("OpsItem resolved successfully.");
                    })
                    .exceptionally(ex -> {
                        throw new CompletionException(ex);
                    }).join();
        }).exceptionally(ex -> {
            Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
            if (cause instanceof SsmException) {
```

```
                throw new RuntimeException("SSM error: " + cause.getMessage(),
cause);
            } else {
                throw new RuntimeException("Unexpected error: " +
cause.getMessage(), cause);
            }
        });

        try {
            future.join();
        } catch (CompletionException ex) {
            throw ex.getCause() instanceof RuntimeException ? (RuntimeException)
ex.getCause() : ex;
        }
    }

    /**
     * Describes AWS SSM OpsItems asynchronously.
     *
     * @param key The key to filter OpsItems by (e.g., OPS_ITEM_ID).
     *
     * This method initiates an asynchronous request to describe SSM OpsItems.
     * If the request is successful, it prints the title and status of each
OpsItem.
     * If an exception occurs, it handles the error appropriately.
     */
    public void describeOpsItems(String key) {
        OpsItemFilter filter = OpsItemFilter.builder()
                .key(OpsItemFilterKey.OPS_ITEM_ID)
                .values(key)
                .operator(OpsItemFilterOperator.EQUAL)
                .build();

        DescribeOpsItemsRequest itemsRequest = DescribeOpsItemsRequest.builder()
                .maxResults(10)
                .opsItemFilters(filter)
                .build();

        CompletableFuture<Void> future = CompletableFuture.runAsync(() -> {
            getAsyncClient().describeOpsItems(itemsRequest)
                    .thenAccept(itemsResponse -> {
                        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());
                        }
                    })
                    .exceptionally(ex -> {
                        throw new CompletionException(ex);
                    }).join();
        }).exceptionally(ex -> {
            Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
            if (cause instanceof SsmException) {
                throw new RuntimeException("SSM error: " + cause.getMessage(),
cause);
            } else {
                throw new RuntimeException("Unexpected error: " +
cause.getMessage(), cause);
            }
        });

        try {
            future.join();
        } catch (CompletionException ex) {
            throw ex.getCause() instanceof RuntimeException ? (RuntimeException)
ex.getCause() : ex;
        }
    }

    /**
     * Updates the AWS SSM OpsItem asynchronously.
     *
     * @param opsItemId The ID of the OpsItem to update.
     * @param title The new title of the OpsItem.
     * @param description The new description of the OpsItem.
     * <p>
     * This method initiates an asynchronous request to update an SSM OpsItem.
     * If the request is successful, it completes without returning a value.
     * If an exception occurs, it handles the error appropriately.
     */
    public void updateOpsItem(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());

        CompletableFuture<Void> future =
getOpsItem(opsItemId).thenCompose(opsItem -> {
            UpdateOpsItemRequest request = UpdateOpsItemRequest.builder()
                    .opsItemId(opsItemId)
                    .title(title)
                    .operationalData(operationalData)
                    .status(opsItem.statusAsString())
                    .description(description)
                    .build();

            return getAsyncClient().updateOpsItem(request).thenAccept(response ->
{
                System.out.println(opsItemId + " updated successfully.");
            }).exceptionally(ex -> {
                throw new CompletionException(ex);
            });
        }).exceptionally(ex -> {
            Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
            if (cause instanceof SsmException) {
                throw new RuntimeException("SSM error: " + cause.getMessage(),
cause);
            } else {
                throw new RuntimeException("Unexpected error: " +
cause.getMessage(), cause);
            }
        });

        try {
            future.join();
        } catch (CompletionException ex) {
            throw ex.getCause() instanceof RuntimeException ? (RuntimeException)
ex.getCause() : ex;
        }
    }


    private static CompletableFuture<OpsItem> getOpsItem(String opsItemId) {
        GetOpsItemRequest request =
GetOpsItemRequest.builder().opsItemId(opsItemId).build();
```

```
        return
getAsyncClient().getOpsItem(request).thenApply(GetOpsItemResponse::opsItem);
    }

    /**
     * Creates an SSM OpsItem asynchronously.
     *
     * @param title The title of the OpsItem.
     * @param source The source of the OpsItem.
     * @param category The category of the OpsItem.
     * @param severity The severity of the OpsItem.
     * @return The ID of the created OpsItem.
     * <p>
     * This method initiates an asynchronous request to create an SSM OpsItem.
     * If the request is successful, it returns the OpsItem ID.
     * If an exception occurs, it handles the error appropriately.
     */
    public String createSSMOpsItem(String title, String source, String category,
String severity) {
        CreateOpsItemRequest opsItemRequest = CreateOpsItemRequest.builder()
                .description("Created by the SSM Java API")
                .title(title)
                .source(source)
                .category(category)
                .severity(severity)
                .build();

        CompletableFuture<CreateOpsItemResponse> future =
getAsyncClient().createOpsItem(opsItemRequest);

        try {
            CreateOpsItemResponse response = future.join();
            return response.opsItemId();
        } catch (CompletionException e) {
            Throwable cause = e.getCause();
            if (cause instanceof SsmException) {
                throw (SsmException) cause;
            } else {
                throw new RuntimeException(cause);
            }
        }
    }

    /**
```

```
    * Displays the date and time when the specific command was invoked.
    *
    * @param commandId The ID of the command to describe.
    * <p>
    * This method initiates an asynchronous request to list command invocations
and prints the date and time of each command invocation.
    * If an exception occurs, it handles the error appropriately.
    */
   public void displayCommands(String commandId) {
       ListCommandInvocationsRequest commandInvocationsRequest =
ListCommandInvocationsRequest.builder()
               .commandId(commandId)
               .build();

       CompletableFuture<ListCommandInvocationsResponse> future =
getAsyncClient().listCommandInvocations(commandInvocationsRequest);
       future.thenAccept(response -> {
           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()));
           }
       }).exceptionally(ex -> {
           Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
           if (cause instanceof SsmException) {
               throw (SsmException) cause;
           } else {
               throw new RuntimeException(cause);
           }
       }).join();
   }

   /**
    * Sends a SSM command to a managed node asynchronously.
    *
    * @param documentName The name of the document to use.
    * @param instanceId The ID of the instance to send the command to.
    * @return The command ID.
    * <p>
    * This method initiates asynchronous requests to send a SSM command to a
managed node.
```

```
    * It waits until the document is active, sends the command, and checks the
command execution status.
    */
   public String sendSSMCommand(String documentName, String instanceId) throws
InterruptedException, SsmException {
       // Before we use Document to send a command - make sure it is active.
       CompletableFuture<Void> documentActiveFuture =
CompletableFuture.runAsync(() -> {
           boolean isDocumentActive = false;
           DescribeDocumentRequest request = DescribeDocumentRequest.builder()
                   .name(documentName)
                   .build();

           while (!isDocumentActive) {
               CompletableFuture<DescribeDocumentResponse> response =
getAsyncClient().describeDocument(request);
               String documentStatus =
response.join().document().statusAsString();
               if (documentStatus.equals("Active")) {
                   System.out.println("The SSM document is active and ready to
use.");
                   isDocumentActive = true;
               } else {
                   System.out.println("The SSM document is not active. Status: "
+ documentStatus);
                   try {
                       Thread.sleep(5000);
                   } catch (InterruptedException e) {
                       throw new RuntimeException(e);
                   }
               }
           }
       });

       documentActiveFuture.join();

       // Create the SendCommandRequest.
       SendCommandRequest commandRequest = SendCommandRequest.builder()
               .documentName(documentName)
               .instanceIds(instanceId)
               .build();

       // Send the command.
```

```
        CompletableFuture<SendCommandResponse> commandFuture =
getAsyncClient().sendCommand(commandRequest);
        final String[] commandId = {null};

        commandFuture.whenComplete((commandResponse, ex) -> {
            if (commandResponse != null) {
                commandId[0] = commandResponse.command().commandId();
                System.out.println("Command ID: " + commandId[0]);

                // Wait for the command execution to complete.
                GetCommandInvocationRequest invocationRequest =
GetCommandInvocationRequest.builder()
                        .commandId(commandId[0])
                        .instanceId(instanceId)
                        .build();

                try {
                    System.out.println("Wait 5 secs");
                    TimeUnit.SECONDS.sleep(5);

                    // Retrieve the command execution details.
                    CompletableFuture<GetCommandInvocationResponse>
invocationFuture = getAsyncClient().getCommandInvocation(invocationRequest);
                    invocationFuture.whenComplete((commandInvocationResponse,
invocationEx) -> {
                        if (commandInvocationResponse != null) {
                            // 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);
                            }
                        } else {
                            Throwable invocationCause = (invocationEx instanceof
CompletionException) ? invocationEx.getCause() : invocationEx;
                            throw new CompletionException(invocationCause);
                        }
                    }).join();
                } catch (InterruptedException e) {
                    throw new RuntimeException(e);
```

```
                }
            } else {
                Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
                if (cause instanceof SsmException) {
                    throw (SsmException) cause;
                } else {
                    throw new RuntimeException(cause);
                }
            }
        }).join();

        return commandId[0];
    }

    /**
     * Creates an AWS SSM document asynchronously.
     *
     * @param docName The name of the document to create.
     * <p>
     * This method initiates an asynchronous request to create an SSM document.
     * If the request is successful, it prints the document status.
     * If an exception occurs, it handles the error appropriately.
     */
    public void createSSMDoc(String docName) throws SsmException {
        String jsonData = """
        {
        "schemaVersion": "2.2",
        "description": "Run a simple shell command",
        "mainSteps": [
            {
                "action": "aws:runShellScript",
                "name": "runEchoCommand",
                "inputs": {
                  "runCommand": [
                    "echo 'Hello, world!'"
                  ]
                }
              }
            ]
        }
        """;

        CreateDocumentRequest request = CreateDocumentRequest.builder()
```

```
                .content(jsonData)
                .name(docName)
                .documentType(DocumentType.COMMAND)
                .build();

        CompletableFuture<CreateDocumentResponse> future =
getAsyncClient().createDocument(request);
        future.thenAccept(response -> {
            System.out.println("The status of the SSM document is " +
response.documentDescription().status());
        }).exceptionally(ex -> {
            Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
            if (cause instanceof DocumentAlreadyExistsException) {
                throw new CompletionException(cause);
            } else if (cause instanceof SsmException) {
                throw new CompletionException(cause);
            } else {
                throw new RuntimeException(cause);
            }
        }).join();
    }

    /**
     * Updates an SSM maintenance window asynchronously.
     *
     * @param id The ID of the maintenance window to update.
     * @param name The new name for the maintenance window.
     * <p>
     * This method initiates an asynchronous request to update an SSM maintenance
window.
     * If the request is successful, it prints a success message.
     * If an exception occurs, it handles the error appropriately.
     */
    public void updateSSMMaintenanceWindow(String id, String name) throws
SsmException {
        UpdateMaintenanceWindowRequest updateRequest =
UpdateMaintenanceWindowRequest.builder()
                .windowId(id)
                .allowUnassociatedTargets(true)
                .duration(24)
                .enabled(true)
                .name(name)
                .schedule("cron(0 0 ? * MON *)")
```

```
                .build();

        CompletableFuture<UpdateMaintenanceWindowResponse> future =
getAsyncClient().updateMaintenanceWindow(updateRequest);
        future.whenComplete((response, ex) -> {
            if (response != null) {
                System.out.println("The SSM maintenance window was successfully
updated");
            } else {
                Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
                if (cause instanceof SsmException) {
                    throw new CompletionException(cause);
                } else {
                    throw new RuntimeException(cause);
                }
            }
        }).join();
    }

    /**
     * Creates an SSM maintenance window asynchronously.
     *
     * @param winName The name of the maintenance window.
     * @return The ID of the created or existing maintenance window.
     * <p>
     * This method initiates an asynchronous request to create an SSM maintenance
window.
     * If the request is successful, it prints the maintenance window ID.
     * If an exception occurs, it handles the error appropriately.
     */
    public String createMaintenanceWindow(String winName) throws SsmException,
DocumentAlreadyExistsException {
        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();
```

```
        CompletableFuture<CreateMaintenanceWindowResponse> future =
getAsyncClient().createMaintenanceWindow(request);
        final String[] windowId = {null};
        future.whenComplete((response, ex) -> {
            if (response != null) {
                String maintenanceWindowId = response.windowId();
                System.out.println("The maintenance window id is " +
maintenanceWindowId);
                windowId[0] = maintenanceWindowId;
            } else {
                Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
                if (cause instanceof DocumentAlreadyExistsException) {
                    throw new CompletionException(cause);
                } else if (cause instanceof SsmException) {
                    throw new CompletionException(cause);
                } else {
                    throw new RuntimeException(cause);
                }
            }
        }).join();

        if (windowId[0] == null) {
            MaintenanceWindowFilter filter = MaintenanceWindowFilter.builder()
                    .key("name")
                    .values(winName)
                    .build();

            DescribeMaintenanceWindowsRequest winRequest =
DescribeMaintenanceWindowsRequest.builder()
                    .filters(filter)
                    .build();

            CompletableFuture<DescribeMaintenanceWindowsResponse> describeFuture
= getAsyncClient().describeMaintenanceWindows(winRequest);
            describeFuture.whenComplete((describeResponse, describeEx) -> {
                if (describeResponse != null) {
                    List<MaintenanceWindowIdentity> windows =
describeResponse.windowIdentities();
                    if (!windows.isEmpty()) {
                        windowId[0] = windows.get(0).windowId();
                        System.out.println("Window ID: " + windowId[0]);
                    } else {
                        System.out.println("Window not found.");
```

```
                        windowId[0] = "";
                    }
                } else {
                    Throwable describeCause = (describeEx instanceof
 CompletionException) ? describeEx.getCause() : describeEx;
                    throw new RuntimeException("Error describing maintenance
 windows: " + describeCause.getMessage(), describeCause);
                }
            }).join();
        }

        return windowId[0];
    }
}
```

- Para obtener detalles sobre la API, consulte los siguientes temas en la Referencia de la API de AWS SDK for Java 2.x.
  - CreateDocument
  - CreateMaintenanceWindow
  - CreateOpsItem
  - DeleteMaintenanceWindow
  - ListCommandInvocations
  - SendCommand
  - UpdateOpsItem

JavaScript

SDK para JavaScript (v3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
import {
  Scenario,
```

```
  ScenarioAction,
  ScenarioInput,
  ScenarioOutput,
} from "@aws-doc-sdk-examples/lib/scenario/index.js";
import { fileURLToPath } from "node:url";
import {
  CreateDocumentCommand,
  CreateMaintenanceWindowCommand,
  CreateOpsItemCommand,
  DeleteDocumentCommand,
  DeleteMaintenanceWindowCommand,
  DeleteOpsItemCommand,
  DescribeOpsItemsCommand,
  DocumentAlreadyExists,
  OpsItemStatus,
  waitUntilCommandExecuted,
  CancelCommandCommand,
  paginateListCommandInvocations,
  SendCommandCommand,
  UpdateMaintenanceWindowCommand,
  UpdateOpsItemCommand,
  SSMClient,
} from "@aws-sdk/client-ssm";
import { parseArgs } from "node:util";

/**
 * @typedef {{
 *   ssmClient: import('@aws-sdk/client-ssm').SSMClient,
 *   documentName?: string
 *   maintenanceWindow?: string
 *   winId?: int
 *   ec2InstanceId?: string
 *   requestedDateTime?: Date
 *   opsItemId?: string
 *   askToDeleteResources?: boolean
 * }} State
 */

const defaultMaintenanceWindow = "ssm-maintenance-window";
const defaultDocumentName = "ssmdocument";
// The timeout duration is highly dependent on the specific setup and environment
 necessary. This example handles only the most common error cases, and uses a
 much shorter duration than most productions systems would use.
const COMMAND_TIMEOUT_DURATION_SECONDS = 30; // 30 seconds
```

```
const pressEnter = new ScenarioInput("continue", "Press Enter to continue", {
  type: "confirm",
});

const greet = new ScenarioOutput(
  "greet",
  `Welcome to the AWS Systems Manager SDK Getting Started scenario.
    This program demonstrates how to interact with Systems Manager using the AWS
 SDK for JavaScript V3.
    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...`,
  { header: true },
);

const createMaintenanceWindow = new ScenarioOutput(
  "createMaintenanceWindow",
  "Step 1: Create a Systems Manager maintenance window.",
);

const getMaintenanceWindow = new ScenarioInput(
  "maintenanceWindow",
  "Please enter the maintenance window name:",
  { type: "input", default: defaultMaintenanceWindow },
);

export const sdkCreateMaintenanceWindow = new ScenarioAction(
  "sdkCreateMaintenanceWindow",
  async (/** @type {State} */ state) => {
    try {
      const response = await state.ssmClient.send(
        new CreateMaintenanceWindowCommand({
          Name: state.maintenanceWindow,
          Schedule: "cron(0 10 ? * MON-FRI *)", //The schedule of the maintenance
 window in the form of a cron or rate expression.
          Duration: 2, //The duration of the maintenance window in hours.
```

```
          Cutoff: 1, //The number of hours before the end of the maintenance
 window that Amazon Web Services Systems Manager stops scheduling new tasks for
 execution.
          AllowUnassociatedTargets: true, //Allow the maintenance window to run
 on managed nodes, even if you haven't registered those nodes as targets.
        }),
      );
      state.winId = response.WindowId;
    } catch (caught) {
      console.error(caught.message);
      console.log(
        `An error occurred while creating the maintenance window. Please fix the
 error and try again. Error message: ${caught.message}`,
      );
      throw caught;
    }
  },
);

const modifyMaintenanceWindow = new ScenarioOutput(
  "modifyMaintenanceWindow",
  "Modify the maintenance window by changing the schedule.",
);

const sdkModifyMaintenanceWindow = new ScenarioAction(
  "sdkModifyMaintenanceWindow",
  async (/** @type {State} */ state) => {
    try {
      await state.ssmClient.send(
        new UpdateMaintenanceWindowCommand({
          WindowId: state.winId,
          Schedule: "cron(0 0 ? * MON *)",
        }),
      );
    } catch (caught) {
      console.error(caught.message);
      console.log(
        `An error occurred while modifying the maintenance window. Please fix the
 error and try again. Error message: ${caught.message}`,
      );
      throw caught;
    }
  },
);
```

```
const createSystemsManagerActions = new ScenarioOutput(
  "createSystemsManagerActions",
  "Create a document that defines the actions that Systems Manager performs on
 your EC2 instance.",
);

const getDocumentName = new ScenarioInput(
  "documentName",
  "Please enter the document: ",
  { type: "input", default: defaultDocumentName },
);

const sdkCreateSSMDoc = new ScenarioAction(
  "sdkCreateSSMDoc",
  async (/** @type {State} */ state) => {
    const contentData = `{
                "schemaVersion": "2.2",
                "description": "Run a simple shell command",
                "mainSteps": [
                    {
                        "action": "aws:runShellScript",
                        "name": "runEchoCommand",
                        "inputs": {
                          "runCommand": [
                            "echo 'Hello, world!'"
                          ]
                        }
                    }
                ]
            }`;
    try {
      await state.ssmClient.send(
        new CreateDocumentCommand({
          Content: contentData,
          Name: state.documentName,
          DocumentType: "Command",
        }),
      );
    } catch (caught) {
      console.log(`Exception type: (${typeof caught})`);
      if (caught instanceof DocumentAlreadyExists) {
        console.log("Document already exists. Continuing...\n");
      } else {
```

```
        console.error(caught.message);
        console.log(
          `An error occurred while creating the document. Please fix the error
 and try again. Error message: ${caught.message}`,
        );
        throw caught;
      }
    }
  },
);

const ec2HelloWorld = new ScenarioOutput(
  "ec2HelloWorld",
  `Now you have the option of running a command on an EC2 instance that echoes
 'Hello, world!'. In order to run this command, you must provide the instance ID
 of a Linux EC2 instance. If you do not already have a running Linux EC2 instance
 in your account, you can create one using the AWS console. For information
 about creating an EC2 instance, see https://docs.aws.amazon.com/AWSEC2/latest/
UserGuide/ec2-launch-instance-wizard.html.`,
);

const enterIdOrSkipEC2HelloWorld = new ScenarioInput(
  "enterIdOrSkipEC2HelloWorld",
  "Enter your EC2 InstanceId or press enter to skip this step: ",
  { type: "input", default: "" },
);

const sdkEC2HelloWorld = new ScenarioAction(
  "sdkEC2HelloWorld",
  async (/** @type {State} */ state) => {
    try {
      const response = await state.ssmClient.send(
        new SendCommandCommand({
          DocumentName: state.documentName,
          InstanceIds: [state.ec2InstanceId],
          TimeoutSeconds: COMMAND_TIMEOUT_DURATION_SECONDS,
        }),
      );
      state.CommandId = response.Command.CommandId;
    } catch (caught) {
      console.error(caught.message);
      console.log(
        `An error occurred while sending the command. Please fix the error and
 try again. Error message: ${caught.message}`,
```

```
      );
      throw caught;
    }
  },
  {
    skipWhen: (/** @type {State} */ state) =>
      state.enterIdOrSkipEC2HelloWorld === "",
  },
);

const sdkGetCommandTime = new ScenarioAction(
  "sdkGetCommandTime",
  async (/** @type {State} */ state) => {
    const listInvocationsPaginated = [];
    console.log(
      "Let's get the time when the specific command was sent to the specific
 managed node.",
    );

    console.log(
      `First, we'll wait for the command to finish executing. This may take up to
 ${COMMAND_TIMEOUT_DURATION_SECONDS} seconds.`,
    );
    const commandExecutedResult = waitUntilCommandExecuted(
      { client: state.ssmClient },
      {
        CommandId: state.CommandId,
        InstanceId: state.ec2InstanceId,
      },
    );
    // This is necessary because the TimeoutSeconds of SendCommandCommand is only
 for the delivery, not execution.
    try {
      await new Promise((_, reject) =>
        setTimeout(
          reject,
          COMMAND_TIMEOUT_DURATION_SECONDS * 1000,
          new Error("Command Timed Out"),
        ),
      );
    } catch (caught) {
      if (caught.message === "Command Timed Out") {
        commandExecutedResult.state = "TIMED_OUT";
      } else {
```

```
        throw caught;
      }
    }

    if (commandExecutedResult.state !== "SUCCESS") {
      console.log(
        `The command with id: ${state.CommandId} did not execute in the allotted
 time. Canceling command.`,
      );
      state.ssmClient.send(
        new CancelCommandCommand({
          CommandId: state.CommandId,
        }),
      );
      state.enterIdOrSkipEC2HelloWorld === "";
      return;
    }

    for await (const page of paginateListCommandInvocations(
      { client: state.ssmClient },
      { CommandId: state.CommandId },
    )) {
      listInvocationsPaginated.push(...page.CommandInvocations);
    }
    /**
     * @type {import('@aws-sdk/client-ssm').CommandInvocation}
     */
    const commandInvocation = listInvocationsPaginated.shift(); // Because the
 call was made with CommandId, there's only one result, so shift it off.
    state.requestedDateTime = commandInvocation.RequestedDateTime;

    console.log(
      `The command invocation happened at: ${state.requestedDateTime}.`,
    );
  },
  {
    skipWhen: (/** @type {State} */ state) =>
      state.enterIdOrSkipEC2HelloWorld === "",
  },
);

const createSSMOpsItem = new ScenarioOutput(
  "createSSMOpsItem",
```

```
  `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.`,
);

const sdkCreateSSMOpsItem = new ScenarioAction(
  "sdkCreateSSMOpsItem",
  async (/** @type {State} */ state) => {
    try {
      const response = await state.ssmClient.send(
        new CreateOpsItemCommand({
          Description: "Created by the System Manager Javascript API",
          Title: "Disk Space Alert",
          Source: "EC2",
          Category: "Performance",
          Severity: "2",
        }),
      );
      state.opsItemId = response.OpsItemId;
    } catch (caught) {
      console.error(caught.message);
      console.log(
        `An error occurred while creating the ops item. Please fix the error and
 try again. Error message: ${caught.message}`,
      );
      throw caught;
    }
  },
);

const updateOpsItem = new ScenarioOutput(
  "updateOpsItem",
  (/** @type {State} */ state) =>
    `Now we will update the OpsItem: ${state.opsItemId}`,
);

const sdkUpdateOpsItem = new ScenarioAction(
  "sdkUpdateOpsItem",
  async (/** @type {State} */ state) => {
    try {
```

```
      const _response = await state.ssmClient.send(
        new UpdateOpsItemCommand({
          OpsItemId: state.opsItemId,
          Description: `An update to ${state.opsItemId}`,
        }),
      );
    } catch (caught) {
      console.error(caught.message);
      console.log(
        `An error occurred while updating the ops item. Please fix the error and
 try again. Error message: ${caught.message}`,
      );
      throw caught;
    }
  },
);

const getOpsItemStatus = new ScenarioOutput(
  "getOpsItemStatus",
  (/** @type {State} */ state) =>
    `Now we will get the status of the OpsItem: ${state.opsItemId}`,
);

const sdkOpsItemStatus = new ScenarioAction(
  "sdkGetOpsItemStatus",
  async (/** @type {State} */ state) => {
    try {
      const response = await state.ssmClient.send(
        new DescribeOpsItemsCommand({
          OpsItemId: state.opsItemId,
        }),
      );
      state.opsItemStatus = response.OpsItemStatus;
    } catch (caught) {
      console.error(caught.message);
      console.log(
        `An error occurred while describing the ops item. Please fix the error
 and try again. Error message: ${caught.message}`,
      );
      throw caught;
    }
  },
);
```

```
const resolveOpsItem = new ScenarioOutput(
  "resolveOpsItem",
  (/** @type {State} */ state) =>
    `Now we will resolve the OpsItem: ${state.opsItemId}`,
);

const sdkResolveOpsItem = new ScenarioAction(
  "sdkResolveOpsItem",
  async (/** @type {State} */ state) => {
    try {
      const _response = await state.ssmClient.send(
        new UpdateOpsItemCommand({
          OpsItemId: state.opsItemId,
          Status: OpsItemStatus.RESOLVED,
        }),
      );
    } catch (caught) {
      console.error(caught.message);
      console.log(
        `An error occurred while updating the ops item. Please fix the error and
 try again. Error message: ${caught.message}`,
      );
      throw caught;
    }
  },
);

const askToDeleteResources = new ScenarioInput(
  "askToDeleteResources",
  "Would you like to delete the Systems Manager resources created during this
 example run?",
  { type: "confirm" },
);

const confirmDeleteChoice = new ScenarioOutput(
  "confirmDeleteChoice",
  (/** @type {State} */ state) => {
    if (state.askToDeleteResources) {
      return "You chose to delete the resources.";
    }
    return "The Systems Manager resources will not be deleted. Please delete them
 manually to avoid charges.";
  },
);
```

```
export const sdkDeleteResources = new ScenarioAction(
  "sdkDeleteResources",
  async (/** @type {State} */ state) => {
    try {
      await state.ssmClient.send(
        new DeleteOpsItemCommand({
          OpsItemId: state.opsItemId,
        }),
      );
      console.log(`The ops item: ${state.opsItemId} was successfully deleted.`);
    } catch (caught) {
      console.log(
        `There was a problem deleting the ops item: ${state.opsItemId}. Please
 delete it manually. Error: ${caught.message}`,
      );
    }

    try {
      await state.ssmClient.send(
        new DeleteMaintenanceWindowCommand({
          Name: state.maintenanceWindow,
          WindowId: state.winId,
        }),
      );
      console.log(
        `The maintenance window: ${state.maintenanceWindow} was successfully
 deleted.`,
      );
    } catch (caught) {
      console.log(
        `There was a problem deleting the maintenance window: ${state.opsItemId}.
 Please delete it manually. Error: ${caught.message}`,
      );
    }

    try {
      await state.ssmClient.send(
        new DeleteDocumentCommand({
          Name: state.documentName,
        }),
      );
      console.log(
        `The document: ${state.documentName} was successfully deleted.`,
```

```
      );
    } catch (caught) {
      console.log(
        `There was a problem deleting the document: ${state.documentName}. Please
 delete it manually. Error: ${caught.message}`,
      );
    }
  },
  { skipWhen: (/** @type {{}} */ state) => !state.askToDeleteResources },
);

const goodbye = new ScenarioOutput(
  "goodbye",
  "This concludes the Systems Manager Basics scenario for the AWS Javascript SDK
 v3. Thank you!",
);

const myScenario = new Scenario(
  "SSM Basics",
  [
    greet,
    pressEnter,
    createMaintenanceWindow,
    getMaintenanceWindow,
    sdkCreateMaintenanceWindow,
    modifyMaintenanceWindow,
    pressEnter,
    sdkModifyMaintenanceWindow,
    createSystemsManagerActions,
    getDocumentName,
    sdkCreateSSMDoc,
    ec2HelloWorld,
    enterIdOrSkipEC2HelloWorld,
    sdkEC2HelloWorld,
    sdkGetCommandTime,
    pressEnter,
    createSSMOpsItem,
    pressEnter,
    sdkCreateSSMOpsItem,
    updateOpsItem,
    pressEnter,
    sdkUpdateOpsItem,
    getOpsItemStatus,
    pressEnter,
```

```
    sdkOpsItemStatus,
    resolveOpsItem,
    pressEnter,
    sdkResolveOpsItem,
    askToDeleteResources,
    confirmDeleteChoice,
    sdkDeleteResources,
    goodbye,
  ],
  { ssmClient: new SSMClient({}) },
);

/** @type {{ stepHandlerOptions: StepHandlerOptions }} */
export const main = async (stepHandlerOptions) => {
  await myScenario.run(stepHandlerOptions);
};

// Invoke main function if this file was run directly.
if (process.argv[1] === fileURLToPath(import.meta.url)) {
  const { values } = parseArgs({
    options: {
      yes: {
        type: "boolean",
        short: "y",
      },
    },
  });
  main({ confirmAll: values.yes });
}
```

- Para obtener detalles sobre la API, consulte los siguientes temas en la Referencia de la API de AWS SDK para JavaScript.
  - CreateDocument
  - CreateMaintenanceWindow
  - CreateOpsItem
  - DeleteMaintenanceWindow
  - ListCommandInvocations
  - SendCommand
  - UpdateOpsItem

Python

SDK para Python (Boto3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

Ejecutar un escenario interactivo en un símbolo del sistema.

```
class SystemsManagerScenario:
    """Runs an interactive scenario that shows how to get started using Amazon
 Systems Manager."""

    def __init__(self, document_wrapper, maintenance_window_wrapper,
 ops_item_wrapper):
        """
        :param document_wrapper: An object that wraps Systems Manager document
 functions.
        :param maintenance_window_wrapper: An object that wraps Systems Manager
 maintenance window functions.
        :param ops_item_wrapper: An object that wraps Systems Manager OpsItem
 functions.
        """
        self.document_wrapper = document_wrapper
        self.maintenance_window_wrapper = maintenance_window_wrapper
        self.ops_item_wrapper = ops_item_wrapper

    def run(self):
        """Demonstrates how to use the AWS SDK for Python (Boto3) to get started
 with Systems Manager."""
        try:
            print("-" * 88)
            print(
                """
Welcome to the AWS Systems Manager SDK Getting Started scenario.
This program demonstrates how to interact with Systems Manager using the AWS SDK
 for Python (Boto3).
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..."""
            )
            q.ask("Please hit Enter")

            print("-" * 88)
            print("Create a Systems Manager maintenance window.")
            maintenance_window_name = q.ask(
                "Please enter the maintenance window name (default is ssm-
maintenance-window):",
            )
            if not maintenance_window_name:
                maintenance_window_name = "ssm-maintenance-window"

            self.maintenance_window_wrapper.create(
                name=maintenance_window_name,
                schedule="cron(0 10 ? * MON-FRI *)",
                duration=2,
                cutoff=1,
                allow_unassociated_targets=True,
            )

            print("-" * 88)
            print("Modify the maintenance window by changing the schedule")
            q.ask("Please hit Enter")

            self.maintenance_window_wrapper.update(
                name=maintenance_window_name,
                schedule="cron(0 0 ? * MON *)",
                duration=24,
                cutoff=1,
                allow_unassociated_targets=True,
                enabled=True,
            )

            print("-" * 88)
            print(
                "Create a document that defines the actions that Systems Manager
 performs on your EC2 instance."
```

```
            )
            document_name = q.ask(
                "Please enter the document name (default is ssmdocument):"
            )

            if not document_name:
                document_name = "ssmdocument"

            self.document_wrapper.create(
                name=document_name,
                content="""
{
    "schemaVersion": "2.2",
    "description": "Run a simple shell command",
    "mainSteps": [
        {
            "action": "aws:runShellScript",
            "name": "runEchoCommand",
            "inputs": {
              "runCommand": [
                "echo 'Hello, world!'"
              ]
            }
        }
    ]
}
            """,
            )

            self.document_wrapper.wait_until_active()

            print(
                """
Now you have the option of running a command on an EC2 instance that echoes
 'Hello, world!'.
In order to run this command, you must provide the instance ID of a Linux EC2
 instance. If you do
not already have a running Linux EC2 instance in your account, you can create one
 using the AWS console.
For information about creating an EC2 instance, see
https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-launch-instance-
wizard.html.
            """
            )
```

```
            if q.ask(
                "Would you like to run a command on an EC2 instance? (y/n)",
                q.is_yesno,
            ):
                instance_id = q.ask(
                    "Please enter the instance ID of the EC2 instance:",
 q.non_empty
                )
                command_id = self.document_wrapper.send_command(
                    instance_ids=[instance_id]
                )

                self.document_wrapper.wait_command_executed(
                    command_id=command_id, instance_id=instance_id
                )

                print("-" * 88)
                print(
                    "Lets get the time when the specific command was sent to the
 specific managed node"
                )
                q.ask("Please hit Enter")


 self.document_wrapper.list_command_invocations(instance_id=instance_id)

            print("-" * 88)
            print("-" * 88)
            print(
                """
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.
            """
            )
            q.ask("Please hit Enter")
```

```
            self.ops_item_wrapper.create(
                title="Disk Space Alert",
                description="Created by the Systems Manager Python (Boto3) API",
                source="EC2",
                category="Performance",
                severity="2",
            )

            print("-" * 88)
            print("-" * 88)
            print(f"Now we will update  the OpsItem {self.ops_item_wrapper.id}")
            q.ask("Please hit Enter")

            self.ops_item_wrapper.update(
                title="Disk Space Alert",
                description=f"An update to {self.ops_item_wrapper.id}",
            )

            print(
                f"Now we will get the status of the OpsItem
{self.ops_item_wrapper.id}"
            )
            q.ask("Please hit Enter")

            # It may take a second for the ops item to be available
            counter = 0
            while not self.ops_item_wrapper.describe() and counter < 5:
                counter += 1
                time.sleep(1)

            print(f"Now we will resolve the OpsItem {self.ops_item_wrapper.id}")
            q.ask("Please hit Enter")

            self.ops_item_wrapper.update(status="Resolved")

            print("-" * 88)
            print("-" * 88)
            if q.ask(
                "Would you like to delete the Systems Manager resources? (y/n)",
                q.is_yesno,
            ):
                print("You selected to delete the resources.")
                self.cleanup()
```

```
            else:
                print("The Systems Manager resources will not be deleted")

            print("-" * 88)
            print("This concludes the Systems Manager SDK Getting Started
 scenario.")
            print("-" * 88)

        except Exception:
            self.cleanup()
            raise

    def cleanup(self):
        self.maintenance_window_wrapper.delete()
        self.ops_item_wrapper.delete()
        self.document_wrapper.delete()


if __name__ == "__main__":
    try:
        scenario = SystemsManagerScenario(
            DocumentWrapper.from_client(),
            MaintenanceWindowWrapper.from_client(),
            OpsItemWrapper.from_client(),
        )
        scenario.run()
    except Exception:
        logging.exception("Something went wrong with the demo.")
```

Definir una clase que ajuste las acciones del documento y el comando.

```
class DocumentWrapper:
    """Encapsulates AWS Systems Manager Document actions."""

    def __init__(self, ssm_client):
        """
        :param ssm_client: A Boto3 Systems Manager client.
        """
        self.ssm_client = ssm_client
        self.name = None

    @classmethod
```

```
    def from_client(cls):
        ssm_client = boto3.client("ssm")
        return cls(ssm_client)


    def create(self, content, name):
        """
        Creates a document.

        :param content: The content of the document.
        :param name: The name of the document.
        """
        try:
            self.ssm_client.create_document(
                Name=name, Content=content, DocumentType="Command"
            )
            self.name = name
        except self.ssm_client.exceptions.DocumentAlreadyExists:
            print(f"Document {name} already exists.")
            self.name = name
        except ClientError as err:
            logger.error(
                "Couldn't create %s. Here's why: %s: %s",
                name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise


    def delete(self):
        """
        Deletes an AWS Systems Manager document.
        """
        if self.name is None:
            return

        try:
            self.ssm_client.delete_document(Name=self.name)
            print(f"Deleted document {self.name}.")
            self.name = None
        except ClientError as err:
            logger.error(
                "Couldn't delete %s. Here's why: %s: %s",
```

```
                self.name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise


    def send_command(self, instance_ids):
        """
        Sends a command to one or more instances.

        :param instance_ids: The IDs of the instances to send the command to.
        :return: The ID of the command.
        """
        try:
            response = self.ssm_client.send_command(
                InstanceIds=instance_ids, DocumentName=self.name,
TimeoutSeconds=3600
            )
            return response["Command"]["CommandId"]
        except ClientError as err:
            logger.error(
                "Couldn't send command to %s. Here's why: %s: %s",
                self.name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise


    def describe(self):
        """
        Describes the document.

        :return: Document status.
        """
        try:
            response = self.ssm_client.describe_document(Name=self.name)
            return response["Document"]["Status"]
        except ClientError as err:
            logger.error(
                "Couldn't get %s. Here's why: %s: %s",
                self.name,
                err.response["Error"]["Code"],
```

```
                err.response["Error"]["Message"],
            )
            raise


    def wait_until_active(self, max_attempts=20, delay=5):
        """
        Waits until the document is active.

        :param max_attempts: The maximum number of attempts for checking the
status.
        :param delay: The delay in seconds between each check.
        """
        attempt = 0
        status = ""
        while attempt <= max_attempts:
            status = self.describe()
            if status == "Active":
                break
            attempt += 1
            time.sleep(delay)

        if status != "Active":
            logger.error("Document is not active.")
        else:
            logger.info("Document is active.")

    def wait_command_executed(self, command_id, instance_id):
        """
        Waits until the command is executed on the instance.

        :param command_id: The ID of the command.
        :param instance_id: The ID of the instance.
        """

        waiter = self.ssm_client.get_waiter("command_executed")
        waiter.wait(CommandId=command_id, InstanceId=instance_id)

    def list_command_invocations(self, instance_id):
        """
        Lists the commands for an instance.

        :param instance_id: The ID of the instance.
        :return: The list of commands.
```

```
        """
        try:
            paginator = self.ssm_client.get_paginator("list_command_invocations")
            command_invocations = []
            for page in paginator.paginate(InstanceId=instance_id):
                command_invocations.extend(page["CommandInvocations"])
            num_of_commands = len(command_invocations)
            print(
                f"{num_of_commands} command invocation(s) found for instance
{instance_id}."
            )

            if num_of_commands > 10:
                print("Displaying the first 10 commands:")
                num_of_commands = 10
            date_format = "%A, %d %B %Y %I:%M%p"
            for command in command_invocations[:num_of_commands]:
                print(
                    f"   The time of command invocation is
{command['RequestedDateTime'].strftime(date_format)}"
                )
        except ClientError as err:
            logger.error(
                "Couldn't list commands for %s. Here's why: %s: %s",
                instance_id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
```

Definir una clase que ajuste las acciones de OpsItems.

```
class OpsItemWrapper:
    """Encapsulates AWS Systems Manager OpsItem actions."""

    def __init__(self, ssm_client):
        """
        :param ssm_client: A Boto3 Systems Manager client.
        """
```

```
        self.ssm_client = ssm_client
        self.id = None

    @classmethod
    def from_client(cls):
        """
        :return: A OpsItemWrapper instance.
        """
        ssm_client = boto3.client("ssm")
        return cls(ssm_client)


    def create(self, title, source, category, severity, description):
        """
        Create an OpsItem

        :param title: The OpsItem title.
        :param source: The OpsItem source.
        :param category: The OpsItem category.
        :param severity: The OpsItem severity.
        :param description: The OpsItem description.

        """
        try:
            response = self.ssm_client.create_ops_item(
                Title=title,
                Source=source,
                Category=category,
                Severity=severity,
                Description=description,
            )
            self.id = response["OpsItemId"]
        except self.ssm_client.exceptions.OpsItemLimitExceededException as err:
            logger.error(
                "Couldn't create ops item because you have exceeded your open
OpsItem limit. "
                "Here's why: %s: %s",
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        except ClientError as err:
            logger.error(
                "Couldn't create ops item %s. Here's why: %s: %s",
```

```
                title,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise

    def delete(self):
        """
        Delete the OpsItem.
        """
        if self.id is None:
            return
        try:
            self.ssm_client.delete_ops_item(OpsItemId=self.id)
            print(f"Deleted ops item with id {self.id}")
            self.id = None
        except ClientError as err:
            logger.error(
                "Couldn't delete ops item %s. Here's why: %s: %s",
                self.id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise


    def describe(self):
        """
        Describe an OpsItem.
        """
        try:
            paginator = self.ssm_client.get_paginator("describe_ops_items")
            ops_items = []
            for page in paginator.paginate(
                OpsItemFilters=[
                    {"Key": "OpsItemId", "Values": [self.id], "Operator":
"Equal"}
                ]
            ):
                ops_items.extend(page["OpsItemSummaries"])

            for item in ops_items:
                print(
```

```
                    f"The item title is {item['Title']} and the status is
{item['Status']}"
                )
            return len(ops_items) > 0
        except ClientError as err:
            logger.error(
                "Couldn't describe ops item %s. Here's why: %s: %s",
                self.id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise


    def update(self, title=None, description=None, status=None):
        """
        Update an OpsItem.

        :param title: The new OpsItem title.
        :param description: The new OpsItem description.
        :param status: The new OpsItem status.
        :return:
        """
        args = dict(OpsItemId=self.id)
        if title is not None:
            args["Title"] = title
        if description is not None:
            args["Description"] = description
        if status is not None:
            args["Status"] = status
        try:
            self.ssm_client.update_ops_item(**args)
        except ClientError as err:
            logger.error(
                "Couldn't update ops item %s. Here's why: %s: %s",
                self.id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
```

Definir una clase que ajuste las acciones de la ventana de mantenimiento.

```
class MaintenanceWindowWrapper:
    """Encapsulates AWS Systems Manager maintenance window actions."""

    def __init__(self, ssm_client):
        """
        :param ssm_client: A Boto3 Systems Manager client.
        """
        self.ssm_client = ssm_client
        self.window_id = None
        self.name = None

    @classmethod
    def from_client(cls):
        ssm_client = boto3.client("ssm")
        return cls(ssm_client)


    def create(self, name, schedule, duration, cutoff,
 allow_unassociated_targets):
        """
        Create an AWS Systems Manager maintenance window.

        :param name: The name of the maintenance window.
        :param schedule: The schedule of the maintenance window.
        :param duration: The duration of the maintenance window.
        :param cutoff: The cutoff time of the maintenance window.
        :param allow_unassociated_targets: Allow the maintenance window to run on
 managed nodes, even
                                           if you haven't registered those nodes
 as targets.
        """
        try:
            response = self.ssm_client.create_maintenance_window(
                Name=name,
                Schedule=schedule,
                Duration=duration,
                Cutoff=cutoff,
                AllowUnassociatedTargets=allow_unassociated_targets,
            )
```

```
            self.window_id = response["WindowId"]
            self.name = name
            logger.info("Created maintenance window %s.", self.window_id)
        except ParamValidationError as error:
            logger.error(
                "Parameter validation error when trying to create maintenance
window %s. Here's why: %s",
                self.window_id,
                error,
            )
            raise
        except ClientError as err:
            logger.error(
                "Couldn't create maintenance window %s. Here's why: %s: %s",
                name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise


    def delete(self):
        """
        Delete the associated AWS Systems Manager maintenance window.
        """
        if self.window_id is None:
            return

        try:
            self.ssm_client.delete_maintenance_window(WindowId=self.window_id)
            logger.info("Deleted maintenance window %s.", self.window_id)
            print(f"Deleted maintenance window {self.name}")
            self.window_id = None
        except ClientError as err:
            logger.error(
                "Couldn't delete maintenance window %s. Here's why: %s: %s",
                self.window_id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise


    def update(
```

```
        self, name, enabled, schedule, duration, cutoff,
allow_unassociated_targets
    ):
        """
        Update an AWS Systems Manager maintenance window.

        :param name: The name of the maintenance window.
        :param enabled: Whether the maintenance window is enabled to run on
managed nodes.
        :param schedule: The schedule of the maintenance window.
        :param duration: The duration of the maintenance window.
        :param cutoff: The cutoff time of the maintenance window.
        :param allow_unassociated_targets: Allow the maintenance window to run on
managed nodes, even
                                           if you haven't registered those nodes
as targets.
        """
        try:
            self.ssm_client.update_maintenance_window(
                WindowId=self.window_id,
                Name=name,
                Enabled=enabled,
                Schedule=schedule,
                Duration=duration,
                Cutoff=cutoff,
                AllowUnassociatedTargets=allow_unassociated_targets,
            )
            self.name = name
            logger.info("Updated maintenance window %s.", self.window_id)
        except ParamValidationError as error:
            logger.error(
                "Parameter validation error when trying to update maintenance
window %s. Here's why: %s",
                self.window_id,
                error,
            )
            raise
        except ClientError as err:
            logger.error(
                "Couldn't update maintenance window %s. Here's why: %s: %s",
                self.name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
```

```
            raise
```

- Para obtener información sobre la API, consulte los siguientes temas en la Referencia de la API de AWS SDK para Python (Boto3).
  - CreateDocument
  - CreateMaintenanceWindow
  - CreateOpsItem
  - DeleteMaintenanceWindow
  - ListCommandInvocations
  - SendCommand
  - UpdateOpsItem

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Acciones para Systems Manager usando SDK de AWS

En los siguientes ejemplos de código se muestra cómo llevar a cabo acciones individuales de Systems Manager con AWS SDK. En cada ejemplo se incluye un enlace a GitHub, con instrucciones de configuración y ejecución del código.

Los siguientes ejemplos incluyen solo las acciones que se utilizan con mayor frecuencia. Para ver una lista completa, consulte la Referencia de la API de AWS Systems Manager.

Ejemplos

- Utilizar AddTagsToResource con una CLI
- Utilizar CancelCommand con una CLI
- Utilizar CreateActivation con una CLI
- Utilizar CreateAssociation con una CLI
- Utilizar CreateAssociationBatch con una CLI
- Uso de CreateDocument con un SDK de AWS o la CLI
- Uso de CreateMaintenanceWindow con un SDK de AWS o la CLI

- Uso de CreateOpsItem con un SDK de AWS o la CLI
- Utilizar CreatePatchBaseline con una CLI
- Utilizar DeleteActivation con una CLI
- Utilizar DeleteAssociation con una CLI
- Uso de DeleteDocument con un SDK de AWS o la CLI
- Uso de DeleteMaintenanceWindow con un SDK de AWS o la CLI
- Cómo utilizar DeleteOpsItem con un AWS SDK
- Utilizar DeleteParameter con una CLI
- Utilizar DeletePatchBaseline con una CLI
- Utilizar DeregisterManagedInstance con una CLI
- Utilizar DeregisterPatchBaselineForPatchGroup con una CLI
- Utilizar DeregisterTargetFromMaintenanceWindow con una CLI
- Utilizar DeregisterTaskFromMaintenanceWindow con una CLI
- Utilizar DescribeActivations con una CLI
- Utilizar DescribeAssociation con una CLI
- Utilizar DescribeAssociationExecutionTargets con una CLI
- Utilizar DescribeAssociationExecutions con una CLI
- Utilizar DescribeAutomationExecutions con una CLI
- Utilizar DescribeAutomationStepExecutions con una CLI
- Utilizar DescribeAvailablePatches con una CLI
- Utilizar DescribeDocument con una CLI
- Utilizar DescribeDocumentPermission con una CLI
- Utilizar DescribeEffectiveInstanceAssociations con una CLI
- Utilizar DescribeEffectivePatchesForPatchBaseline con una CLI
- Utilizar DescribeInstanceAssociationsStatus con una CLI
- Utilizar DescribeInstanceInformation con una CLI
- Utilizar DescribeInstancePatchStates con una CLI
- Utilizar DescribeInstancePatchStatesForPatchGroup con una CLI

- Utilizar DescribeInstancePatches con una CLI
- Utilizar DescribeMaintenanceWindowExecutionTaskInvocations con una CLI
- Utilizar DescribeMaintenanceWindowExecutionTasks con una CLI
- Utilizar DescribeMaintenanceWindowExecutions con una CLI
- Utilizar DescribeMaintenanceWindowTargets con una CLI
- Utilizar DescribeMaintenanceWindowTasks con una CLI
- Utilizar DescribeMaintenanceWindows con una CLI
- Uso de DescribeOpsItems con un SDK de AWS o la CLI
- Uso de DescribeParameters con un SDK de AWS o la CLI
- Utilizar DescribePatchBaselines con una CLI
- Utilizar DescribePatchGroupState con una CLI
- Utilizar DescribePatchGroups con una CLI
- Utilizar GetAutomationExecution con una CLI
- Utilizar GetCommandInvocation con una CLI
- Utilizar GetConnectionStatus con una CLI
- Utilizar GetDefaultPatchBaseline con una CLI
- Utilizar GetDeployablePatchSnapshotForInstance con una CLI
- Utilizar GetDocument con una CLI
- Utilizar GetInventory con una CLI
- Utilizar GetInventorySchema con una CLI
- Utilizar GetMaintenanceWindow con una CLI
- Utilizar GetMaintenanceWindowExecution con una CLI
- Utilizar GetMaintenanceWindowExecutionTask con una CLI
- Uso de GetParameter con un SDK de AWS o la CLI
- Utilizar GetParameterHistory con una CLI
- Utilizar GetParameters con una CLI
- Utilizar GetPatchBaseline con una CLI
- Utilizar GetPatchBaselineForPatchGroup con una CLI

- Utilizar ListAssociationVersions con una CLI
- Utilizar ListAssociations con una CLI
- Uso de ListCommandInvocations con un SDK de AWS o la CLI
- Utilizar ListCommands con una CLI
- Utilizar ListComplianceItems con una CLI
- Utilizar ListComplianceSummaries con una CLI
- Utilizar ListDocumentVersions con una CLI
- Utilizar ListDocuments con una CLI
- Utilizar ListInventoryEntries con una CLI
- Utilizar ListResourceComplianceSummaries con una CLI
- Utilizar ListTagsForResource con una CLI
- Utilizar ModifyDocumentPermission con una CLI
- Utilizar PutComplianceItems con una CLI
- Utilizar PutInventory con una CLI
- Uso de PutParameter con un SDK de AWS o la CLI
- Utilizar RegisterDefaultPatchBaseline con una CLI
- Utilizar RegisterPatchBaselineForPatchGroup con una CLI
- Utilizar RegisterTargetWithMaintenanceWindow con una CLI
- Utilizar RegisterTaskWithMaintenanceWindow con una CLI
- Utilizar RemoveTagsFromResource con una CLI
- Uso de SendCommand con un SDK de AWS o la CLI
- Utilizar StartAutomationExecution con una CLI
- Utilizar StartSession con una CLI
- Utilizar StopAutomationExecution con una CLI
- Utilizar UpdateAssociation con una CLI
- Utilizar UpdateAssociationStatus con una CLI
- Utilizar UpdateDocument con una CLI
- Utilizar UpdateDocumentDefaultVersion con una CLI

- Uso de UpdateMaintenanceWindow con un SDK de AWS o la CLI
- Utilizar UpdateManagedInstanceRole con una CLI
- Uso de UpdateOpsItem con un SDK de AWS o la CLI
- Utilizar UpdatePatchBaseline con una CLI

Utilizar **AddTagsToResource** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `AddTagsToResource`.

CLI

AWS CLI

Ejemplo 1: adición de etiquetas a un periodo de mantenimiento

En el siguiente ejemplo de `add-tags-to-resource` se agrega una etiqueta al periodo de mantenimiento especificado.

```
aws ssm add-tags-to-resource \
    --resource-type "MaintenanceWindow" \
    --resource-id "mw-03eb9db428EXAMPLE" \
    --tags "Key=Stack,Value=Production"
```

Este comando no genera ninguna salida.

Ejemplo 2: adición de etiquetas a un parámetro

En el siguiente ejemplo de `add-tags-to-resource` se agregan dos etiquetas al parámetro especificado.

```
aws ssm add-tags-to-resource \
    --resource-type "Parameter" \
    --resource-id "My-Parameter" \
    --tags '[{"Key":"Region","Value":"East"},{"Key":"Environment",
 "Value":"Production"}]'
```

Este comando no genera ninguna salida.

Ejemplo 3: adición de etiquetas a un documento de SSM

En el siguiente ejemplo de `add-tags-to-resource` se agrega una etiqueta al documento especificado.

```
aws ssm add-tags-to-resource \
    --resource-type "Document" \
    --resource-id "My-Document" \
    --tags "Key=Quarter,Value=Q322"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Etiquetado de recursos de Systems Manager en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte AddTagsToResource en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se actualiza un periodo de mantenimiento con etiquetas nuevas. No se obtienen resultados si el comando se ejecuta correctamente. La sintaxis utilizada en este ejemplo requiere la versión 3 o posterior de PowerShell.

```
$option1 = @{Key="Stack";Value=@("Production")}
Add-SSMResourceTag -ResourceId "mw-03eb9db42890fb82d" -ResourceType
 "MaintenanceWindow" -Tag $option1
```

Ejemplo 2: con la versión 2 de PowerShell, debe usar New-Object para crear cada etiqueta. No se obtienen resultados si el comando se ejecuta correctamente.

```
$tag1 = New-Object Amazon.SimpleSystemsManagement.Model.Tag
$tag1.Key = "Stack"
$tag1.Value = "Production"

Add-SSMResourceTag -ResourceId "mw-03eb9db42890fb82d" -ResourceType
 "MaintenanceWindow" -Tag $tag1
```

- Para obtener información sobre la API, consulte AddTagsToResource en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se actualiza un periodo de mantenimiento con etiquetas nuevas. No se obtienen resultados si el comando se ejecuta correctamente. La sintaxis utilizada en este ejemplo requiere la versión 3 o posterior de PowerShell.

```
$option1 = @{Key="Stack";Value=@("Production")}
Add-SSMResourceTag -ResourceId "mw-03eb9db42890fb82d" -ResourceType
 "MaintenanceWindow" -Tag $option1
```

Ejemplo 2: con la versión 2 de PowerShell, debe usar New-Object para crear cada etiqueta. No se obtienen resultados si el comando se ejecuta correctamente.

```
$tag1 = New-Object Amazon.SimpleSystemsManagement.Model.Tag
$tag1.Key = "Stack"
$tag1.Value = "Production"

Add-SSMResourceTag -ResourceId "mw-03eb9db42890fb82d" -ResourceType
 "MaintenanceWindow" -Tag $tag1
```

- Para obtener información sobre la API, consulte AddTagsToResource en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V2).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`CancelCommand`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `CancelCommand`.

CLI

AWS CLI

Ejemplo 1: cancelación de un comando para todas las instancias

En el siguiente ejemplo de `cancel-command` se intenta cancelar el comando especificado que ya se está ejecutando en todas las instancias.

```
aws ssm cancel-command \
```

```
    --command-id "662add3d-5831-4a10-b64a-f2ff3EXAMPLE"
```

Este comando no genera ninguna salida.

Ejemplo 2: cancelación de un comando para instancias específicas

En el siguiente ejemplo de `cancel-command` se intenta cancelar un comando únicamente en la instancia especificada.

```
aws ssm cancel-command \
    --command-id "662add3d-5831-4a10-b64a-f2ff3EXAMPLE"
    --instance-ids "i-02573cafcfEXAMPLE"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Etiquetado de parámetros de Systems Manager en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte CancelCommand en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se intenta cancelar un comando. No se obtienen resultados si la operación se ejecuta correctamente.

```
Stop-SSMCommand -CommandId "9ded293e-e792-4440-8e3e-7b8ec5feaa38"
```

- Para obtener información sobre la API, consulte CancelCommand en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se intenta cancelar un comando. No se obtienen resultados si la operación se ejecuta correctamente.

```
Stop-SSMCommand -CommandId "9ded293e-e792-4440-8e3e-7b8ec5feaa38"
```

- Para obtener información sobre la API, consulte CancelCommand en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`CreateActivation`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `CreateActivation`.

CLI

AWS CLI

Creación de la activación de una instancia administrada

En el siguiente ejemplo de `create-activation` se crea la activación de una instancia administrada.

```
aws ssm create-activation \
    --default-instance-name "HybridWebServers" \
    --iam-role "HybridWebServersRole" \
    --registration-limit 5
```

Salida:

```
{
    "ActivationId": "5743558d-563b-4457-8682-d16c3EXAMPLE",
    "ActivationCode": "dRmgnYaFv567vEXAMPLE"
}
```

Para obtener más información, consulte Paso 4: crear una activación híbrida para un entorno híbrido y multinube en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte CreateActivation en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se crea una instancia administrada.

```
New-SSMActivation -DefaultInstanceName "MyWebServers" -IamRole
 "SSMAutomationRole" -RegistrationLimit 10
```

Salida:

```
ActivationCode       ActivationId
--------------       ------------
KWChhOxBTiwDcKE9BlKC 08e51e79-1e36-446c-8e63-9458569c1363
```

- Para obtener información sobre la API, consulte CreateActivation en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se crea una instancia administrada.

```
New-SSMActivation -DefaultInstanceName "MyWebServers" -IamRole
 "SSMAutomationRole" -RegistrationLimit 10
```

Salida:

```
ActivationCode       ActivationId
--------------       ------------
KWChhOxBTiwDcKE9BlKC 08e51e79-1e36-446c-8e63-9458569c1363
```

- Para obtener información sobre la API, consulte CreateActivation en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`CreateAssociation`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `CreateAssociation`.

CLI

AWS CLI

Ejemplo 1: asociación de un documento mediante los ID de instancia

En este ejemplo se asocia un documento de configuración a una instancia mediante los ID de instancia.

```
aws ssm create-association \
    --instance-id "i-0cb2b964d3e14fd9f" \
    --name "AWS-UpdateSSMAgent"
```

Salida:

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

Para obtener más información, consulte CreateAssociation en la Referencia de la API de AWS Systems Manager.

Ejemplo 2: asociación de un documento mediante destinos

En este ejemplo, se asocia un documento de configuración a una instancia mediante destinos.

```
aws ssm create-association \
    --name "AWS-UpdateSSMAgent" \
    --targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f"
```

Salida:

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

Para obtener más información, consulte CreateAssociation en la Referencia de la API de AWS Systems Manager.

Ejemplo 3: creación de una asociación que solo se ejecuta una vez

En este ejemplo se crea una nueva asociación que solo se ejecuta una vez en la fecha y hora especificadas. Las asociaciones creadas con una fecha en el pasado o en el presente (cuando se procesan, la fecha queda en el pasado) se ejecutan inmediatamente.

```
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
```

Salida:

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

Para obtener más información, consulte CreateAssociation en la Referencia de la API de AWS Systems Manager o Referencia: expresiones cron y rate para Systems Manager en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte CreateAssociation en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se asocia un documento de configuración a una instancia mediante los ID de instancia.

```
New-SSMAssociation -InstanceId "i-0cb2b964d3e14fd9f" -Name "AWS-UpdateSSMAgent"
```

Salida:

```
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 :
```

Ejemplo 2: en este ejemplo se asocia un documento de configuración a una instancia mediante destinos.

```
$target = @{Key="instanceids";Values=@("i-0cb2b964d3e14fd9f")}
New-SSMAssociation -Name "AWS-UpdateSSMAgent" -Target $target
```

Salida:

```
Name                  : AWS-UpdateSSMAgent
InstanceId            :
Date                  : 3/1/2017 6:22:21 PM
Status.Name           :
Status.Date           :
Status.Message        :
Status.AdditionalInfo :
```

Ejemplo 3: en este ejemplo se asocia un documento de configuración a una instancia mediante destinos y parámetros.

```
$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
```

Salida:

```
Name                  : Configure-CloudWatch
InstanceId            :
Date                  : 5/17/2018 3:17:44 PM
Status.Name           :
Status.Date           :
Status.Message        :
Status.AdditionalInfo :
```

Ejemplo 4: en este ejemplo se crea una asociación a todas las instancias de la región con **`AWS-GatherSoftwareInventory`**. También se proporcionan archivos personalizados y ubicaciones de registro en los parámetros que se van a recopilar

```
$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)"
```

Salida:

```
Name                  : AWS-GatherSoftwareInventory
InstanceId            :
Date                  : 6/9/2019 8:57:56 AM
Status.Name           :
Status.Date           :
Status.Message        :
```

```
Status.AdditionalInfo :
```

- Para obtener información sobre la API, consulte CreateAssociation en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se asocia un documento de configuración a una instancia mediante los ID de instancia.

```
New-SSMAssociation -InstanceId "i-0cb2b964d3e14fd9f" -Name "AWS-UpdateSSMAgent"
```

Salida:

```
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 :
```

Ejemplo 2: en este ejemplo se asocia un documento de configuración a una instancia mediante destinos.

```
$target = @{Key="instanceids";Values=@("i-0cb2b964d3e14fd9f")}
New-SSMAssociation -Name "AWS-UpdateSSMAgent" -Target $target
```

Salida:

```
Name                  : AWS-UpdateSSMAgent
InstanceId            :
Date                  : 3/1/2017 6:22:21 PM
Status.Name           :
Status.Date           :
Status.Message        :
Status.AdditionalInfo :
```

Ejemplo 3: en este ejemplo se asocia un documento de configuración a una instancia mediante destinos y parámetros.

```
$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
```

Salida:

```
Name                 : Configure-CloudWatch
InstanceId           :
Date                 : 5/17/2018 3:17:44 PM
Status.Name          :
Status.Date          :
Status.Message       :
Status.AdditionalInfo :
```

Ejemplo 4: en este ejemplo se crea una asociación a todas las instancias de la región con **`AWS-GatherSoftwareInventory`**. También se proporcionan archivos personalizados y ubicaciones de registro en los parámetros que se van a recopilar

```
$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)"
```

Salida:

```
Name                 : AWS-GatherSoftwareInventory
InstanceId           :
Date                 : 6/9/2019 8:57:56 AM
Status.Name          :
Status.Date          :
```

```
Status.Message        :
Status.AdditionalInfo :
```

- Para obtener información sobre la API, consulte CreateAssociation en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`CreateAssociationBatch`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `CreateAssociationBatch`.

CLI

AWS CLI

Creación de varias asociaciones

En este ejemplo se asocia un documento de configuración a varias instancias. El resultado devuelve una lista de operaciones correctas y con errores, si corresponde.

Comando:

```
aws ssm create-association-batch --entries "Name=AWS-
UpdateSSMAgent,InstanceId=i-1234567890abcdef0" "Name=AWS-
UpdateSSMAgent,InstanceId=i-9876543210abcdef0"
```

Salida:

```
{
  "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": []
}
```

- Para obtener información sobre la API, consulte CreateAssociationBatch en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se asocia un documento de configuración a varias instancias. El resultado devuelve una lista de operaciones correctas y con errores, si corresponde.

```
$option1 = @{InstanceId="i-0cb2b964d3e14fd9f";Name=@("AWS-UpdateSSMAgent")}
$option2 = @{InstanceId="i-0000293ffd8c57862";Name=@("AWS-UpdateSSMAgent")}
New-SSMAssociationFromBatch -Entry $option1,$option2
```

Salida:

```
Failed  Successful
------  ----------
{}      {Amazon.SimpleSystemsManagement.Model.FailedCreateAssociation,
 Amazon.SimpleSystemsManagement.Model.FailedCreateAsso...
```

Ejemplo 2: en este ejemplo se muestran todos los detalles de una operación correcta.

```
$option1 = @{InstanceId="i-0cb2b964d3e14fd9f";Name=@("AWS-UpdateSSMAgent")}
$option2 = @{InstanceId="i-0000293ffd8c57862";Name=@("AWS-UpdateSSMAgent")}
(New-SSMAssociationFromBatch -Entry $option1,$option2).Successful
```

- Para obtener información sobre la API, consulte CreateAssociationBatch en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se asocia un documento de configuración a varias instancias. El resultado devuelve una lista de operaciones correctas y con errores, si corresponde.

```
$option1 = @{InstanceId="i-0cb2b964d3e14fd9f";Name=@("AWS-UpdateSSMAgent")}
$option2 = @{InstanceId="i-0000293ffd8c57862";Name=@("AWS-UpdateSSMAgent")}
New-SSMAssociationFromBatch -Entry $option1,$option2
```

Salida:

```
Failed  Successful
------  ----------
{}      {Amazon.SimpleSystemsManagement.Model.FailedCreateAssociation,
 Amazon.SimpleSystemsManagement.Model.FailedCreateAsso...
```

Ejemplo 2: en este ejemplo se muestran todos los detalles de una operación correcta.

```
$option1 = @{InstanceId="i-0cb2b964d3e14fd9f";Name=@("AWS-UpdateSSMAgent")}
$option2 = @{InstanceId="i-0000293ffd8c57862";Name=@("AWS-UpdateSSMAgent")}
(New-SSMAssociationFromBatch -Entry $option1,$option2).Successful
```

- Para obtener información sobre la API, consulte CreateAssociationBatch en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Uso de **`CreateDocument`** con un SDK de AWS o la CLI

Los siguientes ejemplos de código muestran cómo utilizar `CreateDocument`.

Los ejemplos de acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Puede ver esta acción en contexto en el siguiente ejemplo de código:

- Conceptos básicos

CLI

AWS CLI

Creación de un documento

En el siguiente ejemplo de `create-document` se crea un documento de Systems Manager.

```
aws ssm create-document \
    --content file://exampleDocument.yml \
    --name "Example" \
    --document-type "Automation" \
```

```
    --document-format YAML
```

Salida:

```
{
    "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": []
    }
}
```

Para obtener más información, consulte Crear contenido en el documento de SSM en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte CreateDocument en la Referencia de comandos de la AWS CLI.

Java

SDK para Java 2.x

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
/**
 * Creates an AWS SSM document asynchronously.
 *
 * @param docName The name of the document to create.
 * <p>
 * This method initiates an asynchronous request to create an SSM document.
 * If the request is successful, it prints the document status.
 * If an exception occurs, it handles the error appropriately.
 */
public void createSSMDoc(String docName) throws SsmException {
    String jsonData = """
    {
    "schemaVersion": "2.2",
    "description": "Run a simple shell command",
    "mainSteps": [
        {
            "action": "aws:runShellScript",
            "name": "runEchoCommand",
            "inputs": {
              "runCommand": [
                "echo 'Hello, world!'"
              ]
            }
          }
        ]
    }
```

```
        """;

        CreateDocumentRequest request = CreateDocumentRequest.builder()
                .content(jsonData)
                .name(docName)
                .documentType(DocumentType.COMMAND)
                .build();

        CompletableFuture<CreateDocumentResponse> future =
getAsyncClient().createDocument(request);
        future.thenAccept(response -> {
            System.out.println("The status of the SSM document is " +
response.documentDescription().status());
        }).exceptionally(ex -> {
            Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
            if (cause instanceof DocumentAlreadyExistsException) {
                throw new CompletionException(cause);
            } else if (cause instanceof SsmException) {
                throw new CompletionException(cause);
            } else {
                throw new RuntimeException(cause);
            }
        }).join();
    }
```

- Para obtener información sobre la API, consulte CreateDocument en la Referencia de la API de AWS SDK for Java 2.x.

JavaScript

SDK para JavaScript (v3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
import { CreateDocumentCommand, SSMClient } from "@aws-sdk/client-ssm";
```

```
import { parseArgs } from "node:util";

/**
 * Create an SSM document.
 * @param {{ content: string, name: string, documentType?: DocumentType }}
 */
export const main = async ({ content, name, documentType }) => {
  const client = new SSMClient({});
  try {
    const { documentDescription } = await client.send(
      new CreateDocumentCommand({
        Content: content, // The content for the new SSM document. The content
 must not exceed 64KB.
        Name: name,
        DocumentType: documentType, // Document format type can be JSON, YAML, or
 TEXT. The default format is JSON.
      }),
    );
    console.log("Document created successfully.");
    return { DocumentDescription: documentDescription };
  } catch (caught) {
    if (caught instanceof Error && caught.name === "DocumentAlreadyExists") {
      console.warn(`${caught.message}. Did you provide a new document name?`);
    } else {
      throw caught;
    }
  }
};
```

- Para obtener información sobre la API, consulte CreateDocument en la Referencia de la API de AWS SDK para JavaScript.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se crea un documento en su cuenta. El documento debe estar en formato JSON. Para obtener más información sobre cómo escribir un documento de configuración, consulte Configuration Document en la Referencia de la API de SSM.

```
New-SSMDocument -Content (Get-Content -Raw "c:\temp\RunShellScript.json") -Name
 "RunShellScript" -DocumentType "Command"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte CreateDocument en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se crea un documento en su cuenta. El documento debe estar en formato JSON. Para obtener más información sobre cómo escribir un documento de configuración, consulte Configuration Document en la Referencia de la API de SSM.

```
New-SSMDocument -Content (Get-Content -Raw "c:\temp\RunShellScript.json") -Name
 "RunShellScript" -DocumentType "Command"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte CreateDocument en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Python

SDK para Python (Boto3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
class DocumentWrapper:
    """Encapsulates AWS Systems Manager Document actions."""

    def __init__(self, ssm_client):
        """
        :param ssm_client: A Boto3 Systems Manager client.
        """
        self.ssm_client = ssm_client
        self.name = None

    @classmethod
    def from_client(cls):
        ssm_client = boto3.client("ssm")
        return cls(ssm_client)
```

```
    def create(self, content, name):
        """
        Creates a document.

        :param content: The content of the document.
        :param name: The name of the document.
        """
        try:
            self.ssm_client.create_document(
                Name=name, Content=content, DocumentType="Command"
            )
            self.name = name
        except self.ssm_client.exceptions.DocumentAlreadyExists:
            print(f"Document {name} already exists.")
            self.name = name
        except ClientError as err:
            logger.error(
                "Couldn't create %s. Here's why: %s: %s",
                name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
```

- Para obtener información sobre la API, consulte CreateDocument en la Referencia de la API de AWS SDK para Python (Boto3).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Uso de **`CreateMaintenanceWindow`** con un SDK de AWS o la CLI

Los siguientes ejemplos de código muestran cómo utilizar `CreateMaintenanceWindow`.

Los ejemplos de acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Puede ver esta acción en contexto en el siguiente ejemplo de código:

- Conceptos básicos

CLI

AWS CLI

Ejemplo 1: creación de un periodo de mantenimiento

En el siguiente ejemplo de `create-maintenance-window` se crea un nuevo periodo de mantenimiento que, cada cinco minutos durante un máximo de dos horas (según sea necesario), impide que se inicien nuevas tareas una hora después de que finalice el periodo de mantenimiento, permite los destinos no asociados (instancias que no ha registrado con el periodo de mantenimiento) e indica, mediante el uso de etiquetas personalizadas, que su creador tiene la intención de utilizarlo en un tutorial.

```
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"
```

Salida:

```
{
    "WindowId": "mw-0c50858d01EXAMPLE"
}
```

Ejemplo 2: creación de un periodo de mantenimiento que se ejecuta solo una vez

En el siguiente ejemplo de `create-maintenance-window` se crea un nuevo periodo de mantenimiento que solo se ejecuta una vez en la fecha y hora especificadas.

```
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"
```

Salida:

```
{
    "WindowId": "mw-01234567890abcdef"
}
```

Para obtener más información, consulte Maintenance Windows en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte CreateMaintenanceWindow en la Referencia de comandos de la AWS CLI.

Java

SDK para Java 2.x

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
   /**
    * Creates an SSM maintenance window asynchronously.
    *
    * @param winName The name of the maintenance window.
    * @return The ID of the created or existing maintenance window.
    * <p>
    * This method initiates an asynchronous request to create an SSM maintenance
window.
    * If the request is successful, it prints the maintenance window ID.
    * If an exception occurs, it handles the error appropriately.
    */
   public String createMaintenanceWindow(String winName) throws SsmException,
DocumentAlreadyExistsException {
       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();

        CompletableFuture<CreateMaintenanceWindowResponse> future =
getAsyncClient().createMaintenanceWindow(request);
        final String[] windowId = {null};
        future.whenComplete((response, ex) -> {
            if (response != null) {
                String maintenanceWindowId = response.windowId();
                System.out.println("The maintenance window id is " +
maintenanceWindowId);
                windowId[0] = maintenanceWindowId;
            } else {
                Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
                if (cause instanceof DocumentAlreadyExistsException) {
                    throw new CompletionException(cause);
                } else if (cause instanceof SsmException) {
                    throw new CompletionException(cause);
                } else {
                    throw new RuntimeException(cause);
                }
            }
        }).join();

        if (windowId[0] == null) {
            MaintenanceWindowFilter filter = MaintenanceWindowFilter.builder()
                    .key("name")
                    .values(winName)
                    .build();

            DescribeMaintenanceWindowsRequest winRequest =
DescribeMaintenanceWindowsRequest.builder()
                    .filters(filter)
                    .build();

            CompletableFuture<DescribeMaintenanceWindowsResponse> describeFuture
= getAsyncClient().describeMaintenanceWindows(winRequest);
            describeFuture.whenComplete((describeResponse, describeEx) -> {
                if (describeResponse != null) {
                    List<MaintenanceWindowIdentity> windows =
describeResponse.windowIdentities();
                    if (!windows.isEmpty()) {
                        windowId[0] = windows.get(0).windowId();
                        System.out.println("Window ID: " + windowId[0]);
```

```
                    } else {
                        System.out.println("Window not found.");
                        windowId[0] = "";
                    }
                } else {
                    Throwable describeCause = (describeEx instanceof
 CompletionException) ? describeEx.getCause() : describeEx;
                    throw new RuntimeException("Error describing maintenance
 windows: " + describeCause.getMessage(), describeCause);
                }
            }).join();
        }

        return windowId[0];
    }
```

- Para obtener información sobre la API, consulte CreateMaintenanceWindow en la Referencia de la API de AWS SDK for Java 2.x.

JavaScript

SDK para JavaScript (v3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
import { CreateMaintenanceWindowCommand, SSMClient } from "@aws-sdk/client-ssm";
import { parseArgs } from "node:util";

/**
 * Create an SSM maintenance window.
 * @param {{ name: string, allowUnassociatedTargets: boolean, duration: number,
 cutoff: number, schedule: string, description?: string }}
 */
export const main = async ({
  name,
```

```
  allowUnassociatedTargets, // Allow the maintenance window to run on managed
 nodes, even if you haven't registered those nodes as targets.
  duration, // The duration of the maintenance window in hours.
  cutoff, // The number of hours before the end of the maintenance window that
 Amazon Web Services Systems Manager stops scheduling new tasks for execution.
  schedule, // The schedule of the maintenance window in the form of a cron or
 rate expression.
  description = undefined,
}) => {
  const client = new SSMClient({});

  try {
    const { windowId } = await client.send(
      new CreateMaintenanceWindowCommand({
        Name: name,
        Description: description,
        AllowUnassociatedTargets: allowUnassociatedTargets, // Allow the
 maintenance window to run on managed nodes, even if you haven't registered those
 nodes as targets.
        Duration: duration, // The duration of the maintenance window in hours.
        Cutoff: cutoff, // The number of hours before the end of the maintenance
 window that Amazon Web Services Systems Manager stops scheduling new tasks for
 execution.
        Schedule: schedule, // The schedule of the maintenance window in the form
 of a cron or rate expression.
      }),
    );
    console.log(`Maintenance window created with Id: ${windowId}`);
    return { WindowId: windowId };
  } catch (caught) {
    if (caught instanceof Error && caught.name === "MissingParameter") {
      console.warn(`${caught.message}. Did you provide these values?`);
    } else {
      throw caught;
    }
  }
};
```

- Para obtener información sobre la API, consulte CreateMaintenanceWindow en la Referencia de la API de AWS SDK para JavaScript.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se crea un nuevo periodo de mantenimiento con el nombre especificado que se ejecuta a las 16:00 h todos los martes durante 4 horas, con un límite de 1 hora y que permite la asignación de destinos no asociados.

```
New-SSMMaintenanceWindow -Name "MyMaintenanceWindow" -Duration 4 -Cutoff 1 -
AllowUnassociatedTarget $true -Schedule "cron(0 16 ? * TUE *)"
```

Salida:

```
mw-03eb53e1ea7383998
```

- Para obtener información sobre la API, consulte CreateMaintenanceWindow en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se crea un nuevo periodo de mantenimiento con el nombre especificado que se ejecuta a las 16:00 h todos los martes durante 4 horas, con un límite de 1 hora y que permite la asignación de destinos no asociados.

```
New-SSMMaintenanceWindow -Name "MyMaintenanceWindow" -Duration 4 -Cutoff 1 -
AllowUnassociatedTarget $true -Schedule "cron(0 16 ? * TUE *)"
```

Salida:

```
mw-03eb53e1ea7383998
```

- Para obtener información sobre la API, consulte CreateMaintenanceWindow en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Python

SDK para Python (Boto3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
class MaintenanceWindowWrapper:
    """Encapsulates AWS Systems Manager maintenance window actions."""

    def __init__(self, ssm_client):
        """
        :param ssm_client: A Boto3 Systems Manager client.
        """
        self.ssm_client = ssm_client
        self.window_id = None
        self.name = None

    @classmethod
    def from_client(cls):
        ssm_client = boto3.client("ssm")
        return cls(ssm_client)


    def create(self, name, schedule, duration, cutoff,
 allow_unassociated_targets):
        """
        Create an AWS Systems Manager maintenance window.

        :param name: The name of the maintenance window.
        :param schedule: The schedule of the maintenance window.
        :param duration: The duration of the maintenance window.
        :param cutoff: The cutoff time of the maintenance window.
        :param allow_unassociated_targets: Allow the maintenance window to run on
 managed nodes, even
                                           if you haven't registered those nodes
 as targets.
        """
        try:
```

```
            response = self.ssm_client.create_maintenance_window(
                Name=name,
                Schedule=schedule,
                Duration=duration,
                Cutoff=cutoff,
                AllowUnassociatedTargets=allow_unassociated_targets,
            )
            self.window_id = response["WindowId"]
            self.name = name
            logger.info("Created maintenance window %s.", self.window_id)
        except ParamValidationError as error:
            logger.error(
                "Parameter validation error when trying to create maintenance
window %s. Here's why: %s",
                self.window_id,
                error,
            )
            raise
        except ClientError as err:
            logger.error(
                "Couldn't create maintenance window %s. Here's why: %s: %s",
                name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
```

- Para obtener información sobre la API, consulte CreateMaintenanceWindow en la Referencia de la API de AWS SDK para Python (Boto3).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Uso de **`CreateOpsItem`** con un SDK de AWS o la CLI

Los siguientes ejemplos de código muestran cómo utilizar `CreateOpsItem`.

Los ejemplos de acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Puede ver esta acción en contexto en el siguiente ejemplo de código:

- Conceptos básicos

CLI

AWS CLI

Creación de un OpsItems

En el siguiente ejemplo de `create-ops-item` se utiliza la clave /aws/resources en OperationalData para crear un OpsItem con un recurso relacionado de 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"
```

Salida:

```
{
    "OpsItemId": "oi-1a2b3c4d5e6f"
}
```

Para obtener más información, consulte Creación de OpsItems en la Guía del usuario AWS Systems Manager.

- Para obtener información sobre la API, consulte CreateOpsItem en la Referencia de comandos de la AWS CLI.

Java

SDK para Java 2.x

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
   /**
    * Creates an SSM OpsItem asynchronously.
    *
    * @param title The title of the OpsItem.
    * @param source The source of the OpsItem.
    * @param category The category of the OpsItem.
    * @param severity The severity of the OpsItem.
    * @return The ID of the created OpsItem.
    * <p>
    * This method initiates an asynchronous request to create an SSM OpsItem.
    * If the request is successful, it returns the OpsItem ID.
    * If an exception occurs, it handles the error appropriately.
    */
   public String createSSMOpsItem(String title, String source, String category,
String severity) {
       CreateOpsItemRequest opsItemRequest = CreateOpsItemRequest.builder()
               .description("Created by the SSM Java API")
               .title(title)
               .source(source)
               .category(category)
               .severity(severity)
               .build();

       CompletableFuture<CreateOpsItemResponse> future =
getAsyncClient().createOpsItem(opsItemRequest);

       try {
           CreateOpsItemResponse response = future.join();
           return response.opsItemId();
       } catch (CompletionException e) {
           Throwable cause = e.getCause();
           if (cause instanceof SsmException) {
```

```
                throw (SsmException) cause;
            } else {
                throw new RuntimeException(cause);
            }
        }
    }
```

- Para obtener información sobre la API, consulte CreateOpsItem en la Referencia de la API de AWS SDK for Java 2.x.

JavaScript

SDK para JavaScript (v3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
import { CreateOpsItemCommand, SSMClient } from "@aws-sdk/client-ssm";
import { parseArgs } from "node:util";

/**
 * Create an SSM OpsItem.
 * @param {{ title: string, source: string, category?: string, severity?:
 string }}
 */
export const main = async ({
  title,
  source,
  category = undefined,
  severity = undefined,
}) => {
  const client = new SSMClient({});
  try {
    const { opsItemArn, opsItemId } = await client.send(
      new CreateOpsItemCommand({
        Title: title,
```

```
        Source: source, // The origin of the OpsItem, such as Amazon EC2 or
 Systems Manager.
        Category: category,
        Severity: severity,
      }),
    );
    console.log(`Ops item created with id: ${opsItemId}`);
    return { OpsItemArn: opsItemArn, OpsItemId: opsItemId };
  } catch (caught) {
    if (caught instanceof Error && caught.name === "MissingParameter") {
      console.warn(`${caught.message}. Did you provide these values?`);
    } else {
      throw caught;
    }
  }
};
```

- Para obtener información sobre la API, consulte CreateOpsItem en la Referencia de la API de AWS SDK para JavaScript.

Python

SDK para Python (Boto3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
class OpsItemWrapper:
    """Encapsulates AWS Systems Manager OpsItem actions."""

    def __init__(self, ssm_client):
        """
        :param ssm_client: A Boto3 Systems Manager client.
        """
        self.ssm_client = ssm_client
        self.id = None
```

```
    @classmethod
    def from_client(cls):
        """
        :return: A OpsItemWrapper instance.
        """
        ssm_client = boto3.client("ssm")
        return cls(ssm_client)


    def create(self, title, source, category, severity, description):
        """
        Create an OpsItem

        :param title: The OpsItem title.
        :param source: The OpsItem source.
        :param category: The OpsItem category.
        :param severity: The OpsItem severity.
        :param description: The OpsItem description.

        """
        try:
            response = self.ssm_client.create_ops_item(
                Title=title,
                Source=source,
                Category=category,
                Severity=severity,
                Description=description,
            )
            self.id = response["OpsItemId"]
        except self.ssm_client.exceptions.OpsItemLimitExceededException as err:
            logger.error(
                "Couldn't create ops item because you have exceeded your open
OpsItem limit. "
                "Here's why: %s: %s",
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        except ClientError as err:
            logger.error(
                "Couldn't create ops item %s. Here's why: %s: %s",
                title,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
```

```
            )
            raise
```

- Para obtener información sobre la API, consulte CreateOpsItem en la Referencia de la API de AWS SDK para Python (Boto3).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`CreatePatchBaseline`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `CreatePatchBaseline`.

CLI

AWS CLI

Ejemplo 1: creación de una línea de base de revisiones con aprobación automática

En el siguiente ejemplo de `create-patch-baseline` se crea una línea de base de revisiones para Windows Server que aprueba las revisiones de un entorno de producción siete días después de que Microsoft las publique.

```
aws ssm create-patch-baseline \
    --name "Windows-Production-Baseline-AutoApproval" \
    --operating-system "WINDOWS" \
    --approval-
rules "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=MSRC_SEVERITY,Values=[Critical,1
{Key=CLASSIFICATION,Values=[SecurityUpdates,Updates,UpdateRollups,CriticalUpdates]}]},App
 \
    --description "Baseline containing all updates approved for Windows Server
 production systems"
```

Salida:

```
{
    "BaselineId": "pb-045f10b4f3EXAMPLE"
}
```

Ejemplo 2: creación de una línea de base de revisiones con una fecha límite de aprobación

En el siguiente ejemplo de `create-patch-baseline` se crea una línea de base de revisiones para Windows Server que aprueba todas las revisiones de un entorno de producción que se publicaron el 7 de julio del 2020 o antes.

```
aws ssm create-patch-baseline \
    --name "Windows-Production-Baseline-AutoApproval" \
    --operating-system "WINDOWS" \
    --approval-
rules "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=MSRC_SEVERITY,Values=[Critical,1
{Key=CLASSIFICATION,Values=[SecurityUpdates,Updates,UpdateRollups,CriticalUpdates]}]},App
 \
    --description "Baseline containing all updates approved for Windows Server
 production systems"
```

Salida:

```
{
    "BaselineId": "pb-045f10b4f3EXAMPLE"
}
```

Ejemplo 3: creación de una línea de base de revisiones con las reglas de aprobación almacenadas en un archivo JSON

En el siguiente ejemplo de `create-patch-baseline` se crea una línea de base de revisiones para Amazon Linux 2017.09 que aprueba las revisiones de un entorno de producción siete días después de su publicación, especifica las reglas de aprobación para la línea de base de revisiones y especifica un repositorio personalizado para las revisiones.

```
aws ssm create-patch-baseline \
    --cli-input-json file://my-amazon-linux-approval-rules-and-repo.json
```

Contenido 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"
        }
    ]
}
```

Ejemplo 4: creación de una línea de base de revisiones que especifica las revisiones aprobadas y rechazadas

En el siguiente ejemplo de `create-patch-baseline` se especifican de forma explícita las revisiones que se deben aprobar y rechazar como excepción a las reglas de aprobación predeterminadas.

```
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"
```

Para obtener más información, consulte Creación de una línea de base de revisiones personalizada en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte CreatePatchBaseline en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se crea una línea de base de revisiones que aprueba las revisiones siete días después de que Microsoft las publique, para las instancias administradas que ejecutan Windows Server 2019 en un entorno de producción.

```
$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
```

Salida:

```
pb-0z4z6221c4296b23z
```

- Para obtener información sobre la API, consulte CreatePatchBaseline en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se crea una línea de base de revisiones que aprueba las revisiones siete días después de que Microsoft las publique, para las instancias administradas que ejecutan Windows Server 2019 en un entorno de producción.

```
$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
```

Salida:

```
pb-0z4z6221c4296b23z
```

- Para obtener información sobre la API, consulte CreatePatchBaseline en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DeleteActivation`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DeleteActivation`.

CLI

AWS CLI

Eliminación de la activación de una instancia administrada

En el siguiente ejemplo de `delete-activation` se elimina la activación de una instancia administrada.

```
aws ssm delete-activation \
    --activation-id "aa673477-d926-42c1-8757-1358cEXAMPLE"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Setting Up AWS Systems Manager for Hybrid Environments en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DeleteActivation en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se elimina una activación. No se obtienen resultados si el comando se ejecuta correctamente.

```
Remove-SSMActivation -ActivationId "08e51e79-1e36-446c-8e63-9458569c1363"
```

- Para obtener información sobre la API, consulte DeleteActivation en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se elimina una activación. No se obtienen resultados si el comando se ejecuta correctamente.

```
Remove-SSMActivation -ActivationId "08e51e79-1e36-446c-8e63-9458569c1363"
```

- Para obtener información sobre la API, consulte DeleteActivation en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DeleteAssociation`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DeleteAssociation`.

CLI

AWS CLI

Ejemplo 1: eliminación de una asociación mediante el ID de asociación

En el siguiente ejemplo de `delete-association` se elimina la asociación del ID de asociación especificado. No se obtienen resultados si el comando se ejecuta correctamente.

```
aws ssm delete-association \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Edición y creación de una nueva versión de una asociación en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: eliminación de una asociación

En el siguiente ejemplo de `delete-association` se elimina la asociación entre una instancia y un documento. No se obtienen resultados si el comando se ejecuta correctamente.

```
aws ssm delete-association \
    --instance-id "i-1234567890abcdef0" \
    --name "AWS-UpdateSSMAgent"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Trabajo con asociaciones en Systems Manager en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DeleteAssociation en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se elimina la asociación entre una instancia y un documento. No se obtienen resultados si el comando se ejecuta correctamente.

```
Remove-SSMAssociation -InstanceId "i-0cb2b964d3e14fd9f" -Name "AWS-
UpdateSSMAgent"
```

- Para obtener información sobre la API, consulte DeleteAssociation en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se elimina la asociación entre una instancia y un documento. No se obtienen resultados si el comando se ejecuta correctamente.

```
Remove-SSMAssociation -InstanceId "i-0cb2b964d3e14fd9f" -Name "AWS-
UpdateSSMAgent"
```

- Para obtener información sobre la API, consulte DeleteAssociation en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Uso de **`DeleteDocument`** con un SDK de AWS o la CLI

Los siguientes ejemplos de código muestran cómo utilizar `DeleteDocument`.

CLI

AWS CLI

Eliminación de un documento

En el siguiente ejemplo de `delete-document` se elimina un documento de Systems Manager.

```
aws ssm delete-document \
    --name "Example"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Crear contenido en el documento de SSM en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DeleteDocument en la Referencia de comandos de la AWS CLI.

Java

SDK para Java 2.x

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
    /**
     * Deletes an AWS SSM document asynchronously.
     *
     * @param documentName The name of the document to delete.
     * <p>
     * This method initiates an asynchronous request to delete an SSM document.
```

```
     * If an exception occurs, it handles the error appropriately.
     */
    public void deleteDoc(String documentName) {
        DeleteDocumentRequest documentRequest = DeleteDocumentRequest.builder()
                .name(documentName)
                .build();

        CompletableFuture<Void> future = CompletableFuture.runAsync(() -> {
            getAsyncClient().deleteDocument(documentRequest)
                    .thenAccept(response -> {
                        System.out.println("The SSM document was successfully
deleted.");
                    })
                    .exceptionally(ex -> {
                        throw new CompletionException(ex);
                    }).join();
        }).exceptionally(ex -> {
            Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
            if (cause instanceof SsmException) {
                throw new RuntimeException("SSM error: " + cause.getMessage(),
cause);
            } else {
                throw new RuntimeException("Unexpected error: " +
cause.getMessage(), cause);
            }
        });

        try {
            future.join();
        } catch (CompletionException ex) {
            throw ex.getCause() instanceof RuntimeException ? (RuntimeException)
ex.getCause() : ex;
        }
    }
```

- Para obtener información sobre la API, consulte DeleteDocument en la Referencia de la API de AWS SDK for Java 2.x.

JavaScript

SDK para JavaScript (v3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
import { DeleteDocumentCommand, SSMClient } from "@aws-sdk/client-ssm";
import { parseArgs } from "node:util";

/**
 * Delete an SSM document.
 * @param {{ documentName: string }}
 */
export const main = async ({ documentName }) => {
  const client = new SSMClient({});
  try {
    await client.send(new DeleteDocumentCommand({ Name: documentName }));
    console.log(`Document '${documentName}' deleted.`);
    return { Deleted: true };
  } catch (caught) {
    if (caught instanceof Error && caught.name === "MissingParameter") {
      console.warn(`${caught.message}. Did you provide this value?`);
    } else {
      throw caught;
    }
  }
};
```

- Para obtener información sobre la API, consulte DeleteDocument en la Referencia de la API de AWS SDK para JavaScript.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se elimina un documento. No se obtienen resultados si el comando se ejecuta correctamente.

```
Remove-SSMDocument -Name "RunShellScript"
```

- Para obtener información sobre la API, consulte DeleteDocument en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se elimina un documento. No se obtienen resultados si el comando se ejecuta correctamente.

```
Remove-SSMDocument -Name "RunShellScript"
```

- Para obtener información sobre la API, consulte DeleteDocument en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Python

SDK para Python (Boto3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
class DocumentWrapper:
    """Encapsulates AWS Systems Manager Document actions."""

    def __init__(self, ssm_client):
        """
        :param ssm_client: A Boto3 Systems Manager client.
        """
        self.ssm_client = ssm_client
        self.name = None
```

```
    @classmethod
    def from_client(cls):
        ssm_client = boto3.client("ssm")
        return cls(ssm_client)


    def delete(self):
        """
        Deletes an AWS Systems Manager document.
        """
        if self.name is None:
            return

        try:
            self.ssm_client.delete_document(Name=self.name)
            print(f"Deleted document {self.name}.")
            self.name = None
        except ClientError as err:
            logger.error(
                "Couldn't delete %s. Here's why: %s: %s",
                self.name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
```

- Para obtener información sobre la API, consulte DeleteDocument en la Referencia de la API de AWS SDK para Python (Boto3).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Uso de **`DeleteMaintenanceWindow`** con un SDK de AWS o la CLI

Los siguientes ejemplos de código muestran cómo utilizar `DeleteMaintenanceWindow`.

Los ejemplos de acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Puede ver esta acción en contexto en el siguiente ejemplo de código:

- Conceptos básicos

CLI

AWS CLI

Eliminación de un periodo de mantenimiento

En este ejemplo de `delete-maintenance-window` se elimina el periodo de mantenimiento especificado.

```
aws ssm delete-maintenance-window \
    --window-id "mw-1a2b3c4d5e6f7g8h9"
```

Salida:

```
{
    "WindowId":"mw-1a2b3c4d5e6f7g8h9"
}
```

Para obtener más información, consulte Eliminar un período de mantenimiento (AWS CLI) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DeleteMaintenanceWindow en la Referencia de comandos de la AWS CLI.

Java

SDK para Java 2.x

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
    /**
     * Deletes an AWS SSM Maintenance Window asynchronously.
     *
     * @param winId The ID of the Maintenance Window to delete.
```

```
    * <p>
    * This method initiates an asynchronous request to delete an SSM Maintenance
Window.
    * If an exception occurs, it handles the error appropriately.
    */
   public void deleteMaintenanceWindow(String winId) {
       DeleteMaintenanceWindowRequest windowRequest =
DeleteMaintenanceWindowRequest.builder()
               .windowId(winId)
               .build();

       CompletableFuture<Void> future = CompletableFuture.runAsync(() -> {
           getAsyncClient().deleteMaintenanceWindow(windowRequest)
                   .thenAccept(response -> {
                       System.out.println("The maintenance window was
successfully deleted.");
                   })
                   .exceptionally(ex -> {
                       throw new CompletionException(ex);
                   }).join();
       }).exceptionally(ex -> {
           Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
           if (cause instanceof SsmException) {
               throw new RuntimeException("SSM error: " + cause.getMessage(),
cause);
           } else {
               throw new RuntimeException("Unexpected error: " +
cause.getMessage(), cause);
           }
       });

       try {
           future.join();
       } catch (CompletionException ex) {
           throw ex.getCause() instanceof RuntimeException ? (RuntimeException)
ex.getCause() : ex;
       }
   }
```

- Para obtener información sobre la API, consulte DeleteMaintenanceWindow en la Referencia de la API de AWS SDK for Java 2.x.

JavaScript

SDK para JavaScript (v3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
import { DeleteMaintenanceWindowCommand, SSMClient } from "@aws-sdk/client-ssm";
import { parseArgs } from "node:util";

/**
 * Delete an SSM maintenance window.
 * @param {{ windowId: string }}
 */
export const main = async ({ windowId }) => {
  const client = new SSMClient({});
  try {
    await client.send(
      new DeleteMaintenanceWindowCommand({ WindowId: windowId }),
    );
    console.log(`Maintenance window '${windowId}' deleted.`);
    return { Deleted: true };
  } catch (caught) {
    if (caught instanceof Error && caught.name === "MissingParameter") {
      console.warn(`${caught.message}. Did you provide this value?`);
    } else {
      throw caught;
    }
  }
};
```

- Para obtener información sobre la API, consulte DeleteMaintenanceWindow en la Referencia de la API de AWS SDK para JavaScript.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se elimina un periodo de mantenimiento.

```
Remove-SSMMaintenanceWindow -WindowId "mw-06d59c1a07c022145"
```

Salida:

```
mw-06d59c1a07c022145
```

- Para obtener información sobre la API, consulte DeleteMaintenanceWindow en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se elimina un periodo de mantenimiento.

```
Remove-SSMMaintenanceWindow -WindowId "mw-06d59c1a07c022145"
```

Salida:

```
mw-06d59c1a07c022145
```

- Para obtener información sobre la API, consulte DeleteMaintenanceWindow en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Python

SDK para Python (Boto3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
class MaintenanceWindowWrapper:
```

```
    """Encapsulates AWS Systems Manager maintenance window actions."""

    def __init__(self, ssm_client):
        """
        :param ssm_client: A Boto3 Systems Manager client.
        """
        self.ssm_client = ssm_client
        self.window_id = None
        self.name = None

    @classmethod
    def from_client(cls):
        ssm_client = boto3.client("ssm")
        return cls(ssm_client)


    def delete(self):
        """
        Delete the associated AWS Systems Manager maintenance window.
        """
        if self.window_id is None:
            return

        try:
            self.ssm_client.delete_maintenance_window(WindowId=self.window_id)
            logger.info("Deleted maintenance window %s.", self.window_id)
            print(f"Deleted maintenance window {self.name}")
            self.window_id = None
        except ClientError as err:
            logger.error(
                "Couldn't delete maintenance window %s. Here's why: %s: %s",
                self.window_id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
```

- Para obtener información acerca de la API, consulte DeleteMaintenanceWindow en la Referencia de la API de AWS SDK para Python (Boto3).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Cómo utilizar **`DeleteOpsItem`** con un AWS SDK

En el siguiente ejemplo de código, se muestra cómo utilizar `DeleteOpsItem`.

Python

SDK para Python (Boto3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
class OpsItemWrapper:
    """Encapsulates AWS Systems Manager OpsItem actions."""

    def __init__(self, ssm_client):
        """
        :param ssm_client: A Boto3 Systems Manager client.
        """
        self.ssm_client = ssm_client
        self.id = None

    @classmethod
    def from_client(cls):
        """
        :return: A OpsItemWrapper instance.
        """
        ssm_client = boto3.client("ssm")
        return cls(ssm_client)


    def delete(self):
        """
        Delete the OpsItem.
        """
        if self.id is None:
```

```
            return
        try:
            self.ssm_client.delete_ops_item(OpsItemId=self.id)
            print(f"Deleted ops item with id {self.id}")
            self.id = None
        except ClientError as err:
            logger.error(
                "Couldn't delete ops item %s. Here's why: %s: %s",
                self.id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
```

- Para obtener información sobre la API, consulte DeleteOpsItem en la Referencia de la API de AWS SDK para Python (Boto3).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DeleteParameter`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DeleteParameter`.

CLI

AWS CLI

Eliminación de un parámetro

En el siguiente ejemplo de `delete-parameter` se elimina el parámetro único especificado.

```
aws ssm delete-parameter \
    --name "MyParameter"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Uso de Parameter Store en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DeleteParameter en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se elimina un parámetro. No se obtienen resultados si el comando se ejecuta correctamente.

```
Remove-SSMParameter -Name "helloWorld"
```

- Para obtener información sobre la API, consulte DeleteParameter en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se elimina un parámetro. No se obtienen resultados si el comando se ejecuta correctamente.

```
Remove-SSMParameter -Name "helloWorld"
```

- Para obtener información sobre la API, consulte DeleteParameter en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DeletePatchBaseline`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DeletePatchBaseline`.

CLI

AWS CLI

Eliminación de una línea de base de revisiones

En el siguiente ejemplo de `delete-patch-baseline` se elimina la línea de base de revisiones especificada.

```
aws ssm delete-patch-baseline \
    --baseline-id "pb-045f10b4f382baeda"
```

Salida:

```
{
    "BaselineId": "pb-045f10b4f382baeda"
}
```

Para obtener más información, consulte Actualización o eliminación de una línea de base de revisiones (consola) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DeletePatchBaseline en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se elimina una línea de base de revisiones.

```
Remove-SSMPatchBaseline -BaselineId "pb-045f10b4f382baeda"
```

Salida:

```
pb-045f10b4f382baeda
```

- Para obtener información sobre la API, consulte DeletePatchBaseline en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se elimina una línea de base de revisiones.

```
Remove-SSMPatchBaseline -BaselineId "pb-045f10b4f382baeda"
```

Salida:

```
pb-045f10b4f382baeda
```

- Para obtener información sobre la API, consulte DeletePatchBaseline en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DeregisterManagedInstance`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DeregisterManagedInstance`.

CLI

AWS CLI

Anulación del registro de una instancia administrada

En el siguiente ejemplo de `deregister-managed-instance` se anula el registro de la instancia administrada especificada.

```
aws ssm deregister-managed-instance \
    --instance-id 'mi-08ab247cdfEXAMPLE'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Anulación del registro de nodos administrados en un entorno híbrido y multinube en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DeregisterManagedInstance en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se anula el registro de una instancia administrada. No se obtienen resultados si el comando se ejecuta correctamente.

```
Unregister-SSMManagedInstance -InstanceId "mi-08ab247cdf1046573"
```

- Para obtener información sobre la API, consulte DeregisterManagedInstance en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se anula el registro de una instancia administrada. No se obtienen resultados si el comando se ejecuta correctamente.

```
Unregister-SSMManagedInstance -InstanceId "mi-08ab247cdf1046573"
```

- Para obtener información sobre la API, consulte DeregisterManagedInstance en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DeregisterPatchBaselineForPatchGroup`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DeregisterPatchBaselineForPatchGroup`.

CLI

AWS CLI

Anulación del registro de un grupo de revisiones de una línea de base de revisiones

En el siguiente ejemplo de `deregister-patch-baseline-for-patch-group` se anula el registro del grupo de revisiones especificado de la línea de base de revisiones.

```
aws ssm deregister-patch-baseline-for-patch-group \
    --patch-group "Production" \
    --baseline-id "pb-0ca44a362fEXAMPLE"
```

Salida:

```
{
  "PatchGroup":"Production",
  "BaselineId":"pb-0ca44a362fEXAMPLE"
}
```

Para obtener más información, consulte Añadir un grupo de revisiones a una línea de base de revisiones en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DeregisterPatchBaselineForPatchGroup en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se anula el registro de un grupo de revisiones de una línea de base de revisiones.

```
Unregister-SSMPatchBaselineForPatchGroup -BaselineId "pb-045f10b4f382baeda" -
PatchGroup "Production"
```

Salida:

```
BaselineId           PatchGroup
----------           ----------
pb-045f10b4f382baeda Production
```

- Para obtener información sobre la API, consulte DeregisterPatchBaselineForPatchGroup en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se anula el registro de un grupo de revisiones de una línea de base de revisiones.

```
Unregister-SSMPatchBaselineForPatchGroup -BaselineId "pb-045f10b4f382baeda" -
PatchGroup "Production"
```

Salida:

```
BaselineId           PatchGroup
----------           ----------
pb-045f10b4f382baeda Production
```

- Para obtener información sobre la API, consulte DeregisterPatchBaselineForPatchGroup en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DeregisterTargetFromMaintenanceWindow`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DeregisterTargetFromMaintenanceWindow`.

CLI

AWS CLI

Eliminación de un destino de un periodo de mantenimiento

En el siguiente ejemplo de `deregister-target-from-maintenance-window`, se elimina el destino especificado del periodo de mantenimiento especificado.

```
aws ssm deregister-target-from-maintenance-window \
    --window-id "mw-ab12cd34ef56gh78" \
    --window-target-id "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
```

Salida:

```
{
    "WindowId":"mw-ab12cd34ef56gh78",
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Para obtener más información, consulte Actualizar un período de mantenimiento (AWS CLI) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DeregisterTargetFromMaintenanceWindow en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se elimina un destino de un periodo de mantenimiento.

```
Unregister-SSMTargetFromMaintenanceWindow -WindowTargetId
 "6ab5c208-9fc4-4697-84b7-b02a6cc25f7d" -WindowId "mw-06cf17cbefcb4bf4f"
```

Salida:

```
WindowId             WindowTargetId
--------             --------------
mw-06cf17cbefcb4bf4f 6ab5c208-9fc4-4697-84b7-b02a6cc25f7d
```

- Para obtener información sobre la API, consulte DeregisterTargetFromMaintenanceWindow en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se elimina un destino de un periodo de mantenimiento.

```
Unregister-SSMTargetFromMaintenanceWindow -WindowTargetId
 "6ab5c208-9fc4-4697-84b7-b02a6cc25f7d" -WindowId "mw-06cf17cbefcb4bf4f"
```

Salida:

```
WindowId             WindowTargetId
--------             --------------
mw-06cf17cbefcb4bf4f 6ab5c208-9fc4-4697-84b7-b02a6cc25f7d
```

- Para obtener información sobre la API, consulte DeregisterTargetFromMaintenanceWindow en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DeregisterTaskFromMaintenanceWindow`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DeregisterTaskFromMaintenanceWindow`.

CLI

AWS CLI

Eliminación de una tarea de un periodo de mantenimiento

En el siguiente ejemplo de `deregister-task-from-maintenance-window` se elimina la tarea especificada del periodo de mantenimiento especificado.

```
aws ssm deregister-task-from-maintenance-window \
    --window-id "mw-ab12cd34ef56gh78" \
    --window-task-id "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d5e6c"
```

Salida:

```
{
    "WindowTaskId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d5e6c",
    "WindowId":"mw-ab12cd34ef56gh78"
}
```

Para obtener más información, consulte Tutoriales de Maintenance Windows de Systems Manager (AWS CLI) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DeregisterTaskFromMaintenanceWindow en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se elimina una tarea de un periodo de mantenimiento.

```
Unregister-SSMTaskFromMaintenanceWindow -WindowTaskId "f34a2c47-ddfd-4c85-
a88d-72366b69af1b" -WindowId "mw-03a342e62c96d31b0"
```

Salida:

```
WindowId             WindowTaskId
--------             ------------
mw-03a342e62c96d31b0 f34a2c47-ddfd-4c85-a88d-72366b69af1b
```

- Para obtener información sobre la API, consulte DeregisterTaskFromMaintenanceWindow en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se elimina una tarea de un periodo de mantenimiento.

```
Unregister-SSMTaskFromMaintenanceWindow -WindowTaskId "f34a2c47-ddfd-4c85-
a88d-72366b69af1b" -WindowId "mw-03a342e62c96d31b0"
```

Salida:

```
WindowId             WindowTaskId
--------             ------------
mw-03a342e62c96d31b0 f34a2c47-ddfd-4c85-a88d-72366b69af1b
```

- Para obtener información sobre la API, consulte DeregisterTaskFromMaintenanceWindow en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeActivations`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeActivations`.

CLI

AWS CLI

Descripción de las activaciones

En el siguiente ejemplo de `describe-activations` se muestran detalles sobre las activaciones de su cuenta de AWS.

```
aws ssm describe-activations
```

Salida:

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

Para obtener más información, consulte Paso 4: crear una activación híbrida para un entorno híbrido y multinube en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeActivations en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se proporcionan detalles sobre las activaciones de su cuenta.

```
Get-SSMActivation
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeActivations en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se proporcionan detalles sobre las activaciones de su cuenta.

```
Get-SSMActivation
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeActivations en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeAssociation`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeAssociation`.

CLI

AWS CLI

Ejemplo 1: obtención de los detalles de una asociación

En el siguiente ejemplo de `describe-association` se describe la asociación del ID de asociación especificado.

```
aws ssm describe-association \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Salida:

```
{
    "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"
    }
}
```

Para obtener más información, consulte Edición y creación de una nueva versión de una asociación en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: obtención de los detalles de una asociación para una instancia y un documento específicos

En el siguiente ejemplo de `describe-association` se describe la asociación entre una instancia y un documento.

```
aws ssm describe-association \
    --instance-id "i-1234567890abcdef0" \
```

```
    --name "AWS-UpdateSSMAgent"
```

Salida:

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

Para obtener más información, consulte Edición y creación de una nueva versión de una asociación en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeAssociation en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se describe la asociación entre una instancia y un documento.

```
Get-SSMAssociation -InstanceId "i-0000293ffd8c57862" -Name "AWS-UpdateSSMAgent"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeAssociation en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se describe la asociación entre una instancia y un documento.

```
Get-SSMAssociation -InstanceId "i-0000293ffd8c57862" -Name "AWS-UpdateSSMAgent"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeAssociation en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeAssociationExecutionTargets`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeAssociationExecutionTargets`.

CLI

AWS CLI

Obtención de detalles de la ejecución de una asociación

En el siguiente ejemplo de `describe-association-execution-targets` se describe la ejecución de la asociación especificada.

```
aws ssm describe-association-execution-targets \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
    --execution-id "7abb6378-a4a5-4f10-8312-0123456789ab"
```

Salida:

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

Para obtener más información, consulte Visualización de los historiales de asociación en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeAssociationExecutionTargets en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se muestra el ID del recurso y su estado de ejecución que forman parte de los destinos de ejecución de la asociación

```
Get-SSMAssociationExecutionTarget -AssociationId 123a45a0-
c678-9012-3456-78901234db5e -ExecutionId 123a45a0-c678-9012-3456-78901234db5e |
 Select-Object ResourceId, Status
```

Salida:

```
ResourceId           Status
----------           ------
i-0b1b2a3456f7a890b  Success
i-01c12a45d6fc7a89f  Success
i-0a1caf234f56d7dc8  Success
i-012a3fd45af6dbcfe  Failed
i-0ddc1df23c4a5fb67  Success
```

Ejemplo 2: este comando comprueba la ejecución concreta de una automatización concreta desde ayer, a la que está asociado un documento de comandos. Además, también comprueba si se ha producido un error en la ejecución de la asociación y, de ser así, mostrará los detalles de la invocación del comando junto con el ID de la instancia

```
$AssociationExecution= Get-SSMAssociationExecutionTarget -
AssociationId 1c234567-890f-1aca-a234-5a678d901cb0 -ExecutionId
 12345ca12-3456-2345-2b45-23456789012 |
    Where-Object {$_.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
    }
}
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeAssociationExecutionTargets en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se muestra el ID del recurso y su estado de ejecución que forman parte de los destinos de ejecución de la asociación

```
Get-SSMAssociationExecutionTarget -AssociationId 123a45a0-
c678-9012-3456-78901234db5e -ExecutionId 123a45a0-c678-9012-3456-78901234db5e |
 Select-Object ResourceId, Status
```

Salida:

```
ResourceId           Status
----------           ------
i-0b1b2a3456f7a890b  Success
i-01c12a45d6fc7a89f  Success
i-0a1caf234f56d7dc8  Success
i-012a3fd45af6dbcfe  Failed
i-0ddc1df23c4a5fb67  Success
```

Ejemplo 2: este comando comprueba la ejecución concreta de una automatización concreta desde ayer, a la que está asociado un documento de comandos. Además, también comprueba si se ha producido un error en la ejecución de la asociación y, de ser así, mostrará los detalles de la invocación del comando junto con el ID de la instancia

```
$AssociationExecution= Get-SSMAssociationExecutionTarget -
AssociationId 1c234567-890f-1aca-a234-5a678d901cb0 -ExecutionId
 12345ca12-3456-2345-2b45-23456789012 |
    Where-Object {$_.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
    }
}
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeAssociationExecutionTargets en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeAssociationExecutions`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeAssociationExecutions`.

CLI

AWS CLI

Ejemplo 1: obtención de detalles de todas las ejecuciones de una asociación

En el siguiente ejemplo de `describe-association-executions` se describen todas las ejecuciones de la asociación especificada.

```
aws ssm describe-association-executions \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Salida:

```
{
    "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}"
        },
        ...
    ]
}
```

Para obtener más información, consulte Visualización de los historiales de asociación en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: obtención de detalles de todas las ejecuciones de una asociación después de una fecha y hora específicas

En el siguiente ejemplo de `describe-association-executions` se describen todas las ejecuciones de una asociación después de la fecha y hora especificadas.

```
aws ssm describe-association-executions \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
    --filters "Key=CreatedTime,Value=2019-02-18T16:00:00Z,Type=GREATER_THAN"
```

Salida:

```
{
    "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}"
        },
        ...
    ]
}
```

Para obtener más información, consulte Visualización de los historiales de asociación en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeAssociationExecutions en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se devuelven las ejecuciones del ID de asociación proporcionado

```
Get-SSMAssociationExecution -AssociationId 123a45a0-c678-9012-3456-78901234db5e
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeAssociationExecutions en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se devuelven las ejecuciones del ID de asociación proporcionado

```
Get-SSMAssociationExecution -AssociationId 123a45a0-c678-9012-3456-78901234db5e
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeAssociationExecutions en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeAutomationExecutions`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeAutomationExecutions`.

CLI

AWS CLI

Descripción de una ejecución de automatización

En el siguiente ejemplo de `describe-automation-executions` se muestran detalles sobre una ejecución de Automatización.

```
aws ssm describe-automation-executions \
    --filters Key=ExecutionId,Values=73c8eef8-f4ee-4a05-820c-e354fEXAMPLE
```

Salida:

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

Para obtener más información, consulte Ejecución de una automatización sencilla en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeAutomationExecutions en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se describen todas las ejecuciones de Automatización activas y finalizadas asociadas a su cuenta.

```
Get-SSMAutomationExecutionList
```

Salida:

```
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]]}
```

Ejemplo 2: en este ejemplo se muestra el ExecutionID, el documento y la fecha de inicio y finalización de las ejecuciones cuyo estado de AutomationExecutionStatus no sea “Correcto”

```
Get-SSMAutomationExecutionList | Where-Object AutomationExecutionStatus
 -ne "Success" | Select-Object AutomationExecutionId, DocumentName,
 AutomationExecutionStatus, ExecutionStartTime, ExecutionEndTime | Format-Table -
AutoSize
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeAutomationExecutions en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se describen todas las ejecuciones de Automatización activas y finalizadas asociadas a su cuenta.

```
Get-SSMAutomationExecutionList
```

Salida:

```
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]]}
```

Ejemplo 2: en este ejemplo se muestra el ExecutionID, el documento y la fecha de inicio y finalización de las ejecuciones cuyo estado de AutomationExecutionStatus no sea “Correcto”

```
Get-SSMAutomationExecutionList | Where-Object AutomationExecutionStatus
 -ne "Success" | Select-Object AutomationExecutionId, DocumentName,
 AutomationExecutionStatus, ExecutionStartTime, ExecutionEndTime | Format-Table -
AutoSize
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeAutomationExecutions en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeAutomationStepExecutions`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeAutomationStepExecutions`.

CLI

AWS CLI

Ejemplo 1: descripción de todos los pasos de una ejecución de automatización

En el siguiente ejemplo de `describe-automation-step-executions` se muestran detalles sobre los pasos de una ejecución de Automatización.

```
aws ssm describe-automation-step-executions \
    --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE
```

Salida:

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

Ejemplo 2: descripción de un paso concreto de una ejecución de automatización

En el siguiente ejemplo de `describe-automation-step-executions` se muestran detalles sobre un paso específico de una ejecución de Automatización.

```
aws ssm describe-automation-step-executions \
    --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE \
    --filters Key=StepExecutionId,Values=95e70479-cf20-4d80-8018-7e4e2EXAMPLE
```

Para obtener más información, consulte Running an Automation Workflow Step by Step (Command Line) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeAutomationStepExecutions en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se muestra información acerca de todas las ejecuciones de pasos activas y finalizadas en un flujo de trabajo de Automatización.

```
Get-SSMAutomationStepExecution -AutomationExecutionId e1d2bad3-4567-8901-
ae23-456c7c8901be | Select-Object StepName, Action, StepStatus
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeAutomationStepExecutions en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se muestra información acerca de todas las ejecuciones de pasos activas y finalizadas en un flujo de trabajo de Automatización.

```
Get-SSMAutomationStepExecution -AutomationExecutionId e1d2bad3-4567-8901-
ae23-456c7c8901be | Select-Object StepName, Action, StepStatus
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeAutomationStepExecutions en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeAvailablePatches`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeAvailablePatches`.

CLI

AWS CLI

Obtención de las revisiones disponibles

En el siguiente ejemplo de `describe-available-patches` se obtienen detalles sobre todas las revisiones disponibles para Windows Server 2019 que tienen una gravedad de MSRC crítica.

```
aws ssm describe-available-patches \
    --
filters "Key=PRODUCT,Values=WindowsServer2019" "Key=MSRC_SEVERITY,Values=Critical"
```

Salida:

```
{
    "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"
        },
        ...
    ]
}
```

Obtención de detalles de una revisión específica

En el siguiente ejemplo de `describe-available-patches` se recuperan los detalles sobre la revisión especificada.

```
aws ssm describe-available-patches \
    --filters "Key=PATCH_ID,Values=KB4480979"
```

Salida:

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

Para obtener más información, consulte Cómo funcionan las operaciones de Patch Manager en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeAvailablePatches en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se obtienen todas las revisiones de Windows Server 2012 que tienen una gravedad de MSRC crítica. La sintaxis utilizada en este ejemplo requiere la versión 3 o posterior de PowerShell.

```
$filter1 = @{Key="PRODUCT";Values=@("WindowsServer2012")}
$filter2 = @{Key="MSRC_SEVERITY";Values=@("Critical")}

Get-SSMAvailablePatch -Filter $filter1,$filter2
```

Salida:

```
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
...
```

Ejemplo 2: con la versión 2 de PowerShell, debe usar New-Object para crear cada filtro.

```
$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
```

Ejemplo 3: en este ejemplo se muestran todas las actualizaciones publicadas en los últimos 20 días y aplicables a los productos que coinciden con WindowsServer2019

```
Get-SSMAvailablePatch | Where-Object ReleaseDate -ge (Get-Date).AddDays(-20)
 | Where-Object Product -eq "WindowsServer2019" | Select-Object ReleaseDate,
 Product, Title
```

Salida:

```
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)
```

- Para obtener información sobre la API, consulte DescribeAvailablePatches en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se obtienen todas las revisiones de Windows Server 2012 que tienen una gravedad de MSRC crítica. La sintaxis utilizada en este ejemplo requiere la versión 3 o posterior de PowerShell.

```
$filter1 = @{Key="PRODUCT";Values=@("WindowsServer2012")}
$filter2 = @{Key="MSRC_SEVERITY";Values=@("Critical")}

Get-SSMAvailablePatch -Filter $filter1,$filter2
```

Salida:

```
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
...
```

Ejemplo 2: con la versión 2 de PowerShell, debe usar New-Object para crear cada filtro.

```
$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
```

Ejemplo 3: en este ejemplo se muestran todas las actualizaciones publicadas en los últimos 20 días y aplicables a los productos que coinciden con WindowsServer2019

```
Get-SSMAvailablePatch | Where-Object ReleaseDate -ge (Get-Date).AddDays(-20)
 | Where-Object Product -eq "WindowsServer2019" | Select-Object ReleaseDate,
 Product, Title
```

Salida:

```
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)
```

- Para obtener información sobre la API, consulte DescribeAvailablePatches en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeDocument`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeDocument`.

CLI

AWS CLI

Visualización de los detalles de un documento

En el siguiente ejemplo de `describe-document` se muestran detalles sobre un documento de Systems Manager de su cuenta de AWS.

```
aws ssm describe-document \
    --name "Example"
```

Salida:

```
{
    "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": []
    }
```

```
}
```

Para obtener más información, consulte Crear contenido en el documento de SSM en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeDocument en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se devuelve información sobre un documento.

```
Get-SSMDocumentDescription -Name "RunShellScript"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeDocument en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se devuelve información sobre un documento.

```
Get-SSMDocumentDescription -Name "RunShellScript"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeDocument en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeDocumentPermission`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeDocumentPermission`.

CLI

AWS CLI

Descripción de los permisos de los documentos

En el siguiente ejemplo de `describe-document-permission` se muestran los detalles de los permisos sobre un documento de Systems Manager que se comparte públicamente.

```
aws ssm describe-document-permission \
    --name "Example" \
    --permission-type "Share"
```

Salida:

```
{
    "AccountIds": [
        "all"
    ],
    "AccountSharingInfoList": [
        {
            "AccountId": "all",
            "SharedDocumentVersion": "$DEFAULT"
        }
    ]
}
```

Para obtener más información, consulte Compartir un documento de Systems Manager en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeDocumentPermission en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran todas las versiones de un documento.

```
Get-SSMDocumentVersionList -Name "RunShellScript"
```

Salida:

```
CreatedDate          DocumentVersion IsDefaultVersion Name
-----------          --------------- ---------------- ----
2/24/2017 5:25:13 AM 1               True             RunShellScript
```

- Para obtener información sobre la API, consulte DescribeDocumentPermission en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran todas las versiones de un documento.

```
Get-SSMDocumentVersionList -Name "RunShellScript"
```

Salida:

```
CreatedDate          DocumentVersion IsDefaultVersion Name
-----------          --------------- ---------------- ----
2/24/2017 5:25:13 AM 1               True             RunShellScript
```

- Para obtener información sobre la API, consulte DescribeDocumentPermission en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeEffectiveInstanceAssociations`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeEffectiveInstanceAssociations`.

CLI

AWS CLI

Obtención de los detalles de las asociaciones efectivas de una instancia

En el siguiente ejemplo de `describe-effective-instance-associations` se recuperan los detalles sobre las asociaciones efectivas de una instancia.

Comando:

```
aws ssm describe-effective-instance-associations --instance-
id "i-1234567890abcdef0"
```

Salida:

```
{
```

```
    "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"
        }
    ]
}
```

- Para obtener información sobre la API, consulte DescribeEffectiveInstanceAssociations en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se describen las asociaciones efectivas de una instancia.

```
Get-SSMEffectiveInstanceAssociationList -InstanceId "i-0000293ffd8c57862" -
MaxResult 5
```

Salida:

```
AssociationId                        Content
```

```
-------------                           -------
d8617c07-2079-4c18-9847-1655fc2698b0 {...
```

Ejemplo 2: en este ejemplo se muestra el contenido de las asociaciones efectivas de una instancia.

```
(Get-SSMEffectiveInstanceAssociationList -InstanceId "i-0000293ffd8c57862" -
MaxResult 5).Content
```

Salida:

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

- Para obtener información sobre la API, consulte DescribeEffectiveInstanceAssociations en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se describen las asociaciones efectivas de una instancia.

```
Get-SSMEffectiveInstanceAssociationList -InstanceId "i-0000293ffd8c57862" -
MaxResult 5
```

Salida:

```
AssociationId                        Content
-------------                        -------
d8617c07-2079-4c18-9847-1655fc2698b0 {...
```

Ejemplo 2: en este ejemplo se muestra el contenido de las asociaciones efectivas de una instancia.

```
(Get-SSMEffectiveInstanceAssociationList -InstanceId "i-0000293ffd8c57862" -
MaxResult 5).Content
```

Salida:

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

- Para obtener información sobre la API, consulte DescribeEffectiveInstanceAssociations en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeEffectivePatchesForPatchBaseline`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeEffectivePatchesForPatchBaseline`.

CLI

AWS CLI

Ejemplo 1: obtención de todas las revisiones definidas por una línea de base de revisiones personalizada

En el siguiente ejemplo de `describe-effective-patches-for-patch-baseline` se devuelven las revisiones definidas por una línea de base de revisiones personalizada en la cuenta de AWS actual. Tenga en cuenta que, para una línea de base personalizada, solo se requiere el ID para `--baseline-id`.

```
aws ssm describe-effective-patches-for-patch-baseline \
    --baseline-id "pb-08b654cf9b9681f04"
```

Salida:

```
{
    "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--"
}
```

Ejemplo 2: obtención de todas las revisiones definidas por una línea de base de revisiones administrada de AWS

En el siguiente ejemplo de `describe-effective-patches-for-patch-baseline` se devuelven las revisiones definidas por una línea de base de revisiones administrada de AWS. Tenga en cuenta que, para una línea de base administrada de AWS, se requiere el ARN de la línea de base para `--baseline-id`.

```
aws ssm describe-effective-patches-for-patch-baseline \
    --baseline-id "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-020d361a05defe4ed"
```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte Cómo se seleccionan las revisiones de seguridad en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeEffectivePatchesForPatchBaseline en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran todas las líneas de base de revisiones con una lista de resultados máxima de 1.

```
Get-SSMEffectivePatchesForPatchBaseline -BaselineId "pb-0a2f1059b670ebd31" -
MaxResult 1
```

Salida:

```
Patch                                       PatchStatus
-----                                       -----------
Amazon.SimpleSystemsManagement.Model.Patch
 Amazon.SimpleSystemsManagement.Model.PatchStatus
```

Ejemplo 2: en este ejemplo se muestran los estados de revisión de todas las líneas de base de revisiones con una lista de resultados máxima de 1.

```
(Get-SSMEffectivePatchesForPatchBaseline -BaselineId "pb-0a2f1059b670ebd31" -
MaxResult 1).PatchStatus
```

Salida:

```
ApprovalDate         DeploymentStatus
------------         ----------------
```

```
12/21/2010 6:00:00 PM APPROVED
```

- Para obtener información sobre la API, consulte DescribeEffectivePatchesForPatchBaseline en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran todas las líneas de base de revisiones con una lista de resultados máxima de 1.

```
Get-SSMEffectivePatchesForPatchBaseline -BaselineId "pb-0a2f1059b670ebd31" -
MaxResult 1
```

Salida:

```
Patch                                       PatchStatus
-----                                       -----------
Amazon.SimpleSystemsManagement.Model.Patch
 Amazon.SimpleSystemsManagement.Model.PatchStatus
```

Ejemplo 2: en este ejemplo se muestran los estados de revisión de todas las líneas de base de revisiones con una lista de resultados máxima de 1.

```
(Get-SSMEffectivePatchesForPatchBaseline -BaselineId "pb-0a2f1059b670ebd31" -
MaxResult 1).PatchStatus
```

Salida:

```
ApprovalDate          DeploymentStatus
------------          ----------------
12/21/2010 6:00:00 PM APPROVED
```

- Para obtener información sobre la API, consulte DescribeEffectivePatchesForPatchBaseline en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeInstanceAssociationsStatus`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeInstanceAssociationsStatus`.

CLI

AWS CLI

Descripción del estado de las asociaciones de una instancia

En este ejemplo se muestran los detalles de las asociaciones de una instancia.

Comando:

```
aws ssm describe-instance-associations-status --instance-id "i-1234567890abcdef0"
```

Salida:

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

- Para obtener información sobre la API, consulte DescribeInstanceAssociationsStatus en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se muestran los detalles de las asociaciones de una instancia.

```
Get-SSMInstanceAssociationsStatus -InstanceId "i-0000293ffd8c57862"
```

Salida:

```
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
```

Ejemplo 2: en este ejemplo se comprueba el estado de la asociación de instancias para el ID de instancia dado y, además, se muestra el estado de ejecución de esas asociaciones

```
Get-SSMInstanceAssociationsStatus -InstanceId i-012e3cb4df567e8aa | ForEach-
Object {Get-SSMAssociationExecution -AssociationId .AssociationId}
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeInstanceAssociationsStatus en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se muestran los detalles de las asociaciones de una instancia.

```
Get-SSMInstanceAssociationsStatus -InstanceId "i-0000293ffd8c57862"
```

Salida:

```
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
```

Ejemplo 2: en este ejemplo se comprueba el estado de la asociación de instancias para el ID de instancia dado y, además, se muestra el estado de ejecución de esas asociaciones

```
Get-SSMInstanceAssociationsStatus -InstanceId i-012e3cb4df567e8aa | ForEach-
Object {Get-SSMAssociationExecution -AssociationId .AssociationId}
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeInstanceAssociationsStatus en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (Vr).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeInstanceInformation`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeInstanceInformation`.

CLI

AWS CLI

Ejemplo 1: descripción de la información de las instancias administradas

En el siguiente ejemplo de `describe-instance-information` se recuperan los detalles de cada una de las instancias administradas.

```
aws ssm describe-instance-information
```

Ejemplo 2: descripción de la información sobre una instancia administrada específica

En el siguiente ejemplo de `describe-instance-information` se muestran los detalles de la instancia administrada `i-028ea792daEXAMPLE`.

```
aws ssm describe-instance-information \
    --filters "Key=InstanceIds,Values=i-028ea792daEXAMPLE"
```

Ejemplo 3: descripción de la información sobre las instancias administradas con una clave de etiqueta específica

En el siguiente ejemplo de `describe-instance-information` se muestran los detalles de las instancias administradas que tienen la clave de etiqueta `DEV`.

```
aws ssm describe-instance-information \
    --filters "Key=tag-key,Values=DEV"
```

Salida:

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

Para obtener más información, consulte Managed Instances en la Guía del usuario de AWS.

- Para obtener información sobre la API, consulte DescribeInstanceInformation en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se muestran los detalles de cada una de las instancias.

```
Get-SSMInstanceInformation
```

Salida:

```
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
```

Ejemplo 2: en este ejemplo se muestra cómo utilizar el parámetro -Filter para filtrar los resultados solo a las instancias de AWS Systems Manager de la región **`us-east-1`** con un valor de **`AgentVersion`** de **`2.2.800.0`**. Puede encontrar una lista de valores clave de -Filter válidos en el tema de referencia de la API InstanceInformation (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
```

Salida:

```
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
```

Ejemplo 3: en este ejemplo se muestra cómo utilizar el parámetro -InstanceInformationFilterList para filtrar los resultados solo a las instancias de AWS Systems Manager de la región **us-east-1** con un valor de **PlatformTypes** de **Windows** o **Linux**. Puede encontrar una lista de valores clave de -InstanceInformationFilterList válidos en el tema de referencia de la API InstanceInformationFilter (https://docs.aws.amazon.com/systems-manager/latest/APIReference/API_InstanceInformationFilter.html).

```
$Filters = @{
   Key="PlatformTypes"
   ValueSet=("Windows","Linux")
}
Get-SSMInstanceInformation -Region us-east-1 -InstanceInformationFilterList
 $Filters
```

Salida:

```
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
```

Ejemplo 4: en este ejemplo se enumeran las instancias administradas de SSM y se exportan los vaores de InstanceId, PingStatus, LastPingDateTime y PlatformName a un archivo CSV.

```
Get-SSMInstanceInformation | Select-Object InstanceId, PingStatus,
 LastPingDateTime, PlatformName | Export-Csv Instance-details.csv -
NoTypeInformation
```

- Para obtener información sobre la API, consulte DescribeInstanceInformation en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se muestran los detalles de cada una de las instancias.

```
Get-SSMInstanceInformation
```

Salida:

```
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
```

Ejemplo 2: en este ejemplo se muestra cómo utilizar el parámetro -Filter para filtrar los resultados solo a las instancias de AWS Systems Manager de la región **`us-east-1`** con un valor de **`AgentVersion`** de **`2.2.800.0`**. Puede encontrar una lista de valores clave de -Filter válidos en el tema de referencia de la API InstanceInformation (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
```

Salida:

```
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
```

Ejemplo 3: en este ejemplo se muestra cómo utilizar el parámetro -InstanceInformationFilterList para filtrar los resultados solo a las instancias de AWS Systems Manager de la región **`us-east-1`** con un valor de **`PlatformTypes`** de **`Windows`** o **`Linux`**. Puede encontrar una lista de valores clave de -InstanceInformationFilterList válidos en el tema de referencia de la API InstanceInformationFilter (https://docs.aws.amazon.com/systems-manager/latest/APIReference/API_InstanceInformationFilter.html).

```
$Filters = @{
   Key="PlatformTypes"
   ValueSet=("Windows","Linux")
}
Get-SSMInstanceInformation -Region us-east-1 -InstanceInformationFilterList
 $Filters
```

Salida:

```
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
```

Ejemplo 4: en este ejemplo se enumeran las instancias administradas de SSM y se exportan los vaores de InstanceId, PingStatus, LastPingDateTime y PlatformName a un archivo CSV.

```
Get-SSMInstanceInformation | Select-Object InstanceId, PingStatus,
 LastPingDateTime, PlatformName | Export-Csv Instance-details.csv -
NoTypeInformation
```

- Para obtener información sobre la API, consulte DescribeInstanceInformation en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeInstancePatchStates`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeInstancePatchStates`.

CLI

AWS CLI

Obtención de los estados resumidos de las revisiones en instancias

En este ejemplo de `describe-instance-patch-states` se obtienen los estados resumidos de las revisiones en una instancia.

```
aws ssm describe-instance-patch-states \
    --instance-ids "i-1234567890abcdef0"
```

Salida:

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

Para obtener más información, consulte Conocimiento de los valores del estado de conformidad de parches en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeInstancePatchStates en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se obtienen los estados resumidos de las revisiones en una instancia.

```
Get-SSMInstancePatchState -InstanceId "i-08ee91c0b17045407"
```

Ejemplo 2: en este ejemplo se obtienen los estados resumidos de las revisiones en dos instancias.

```
Get-SSMInstancePatchState -InstanceId "i-08ee91c0b17045407","i-09a618aec652973a9"
```

- Para obtener información sobre la API, consulte DescribeInstancePatchStates en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se obtienen los estados resumidos de las revisiones en una instancia.

```
Get-SSMInstancePatchState -InstanceId "i-08ee91c0b17045407"
```

Ejemplo 2: en este ejemplo se obtienen los estados resumidos de las revisiones en dos instancias.

```
Get-SSMInstancePatchState -InstanceId "i-08ee91c0b17045407","i-09a618aec652973a9"
```

- Para obtener información sobre la API, consulte DescribeInstancePatchStates en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeInstancePatchStatesForPatchGroup`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeInstancePatchStatesForPatchGroup`.

CLI

AWS CLI

Ejemplo 1: obtención de los estados de las instancias de un grupo de revisiones

En el siguiente ejemplo de `describe-instance-patch-states-for-patch-group` se obtienen detalles sobre los estados resumidos de las revisiones por instancia en el grupo de revisiones especificado.

```
aws ssm describe-instance-patch-states-for-patch-group \
    --patch-group "Production"
```

Salida:

```
{
```

```
    "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
```

```
        }
    ]
}
```

Ejemplo 2: obtención de los estados de las instancias de un grupo de revisiones al que le faltan más de cinco revisiones

En el siguiente ejemplo de `describe-instance-patch-states-for-patch-group` se obtienen detalles sobre los estados resumidos de las revisiones del grupo de revisiones especificado en las instancias a las que les faltan más de cinco revisiones.

```
aws ssm describe-instance-patch-states-for-patch-group \
    --filters Key=MissingCount,Type=GreaterThan,Values=5 \
    --patch-group "Production"
```

Salida:

```
{
    "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
        }
    ]
```

```
}
```

Ejemplo 3: obtención de los estados de las instancias de un grupo de revisiones con menos de diez instancias que requieren un reinicio

En el siguiente ejemplo de `describe-instance-patch-states-for-patch-group` se obtienen detalles sobre los estados resumidos de las revisiones del grupo de revisiones especificado en las instancias con menos de diez instancias que requieren un reinicio.

```
aws ssm describe-instance-patch-states-for-patch-group \
    --filters Key=InstalledPendingRebootCount,Type=LessThan,Values=10 \
    --patch-group "Production"
```

Salida:

```
{
    "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
        }
    ]
}
```

Para obtener más información, consulte Conocimiento de los valores del estado de conformidad de parches en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeInstancePatchStatesForPatchGroup en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se obtienen los estados resumidos de las revisiones por instancia en un grupo de revisiones.

```
Get-SSMInstancePatchStatesForPatchGroup -PatchGroup "Production"
```

- Para obtener información sobre la API, consulte DescribeInstancePatchStatesForPatchGroup en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se obtienen los estados resumidos de las revisiones por instancia en un grupo de revisiones.

```
Get-SSMInstancePatchStatesForPatchGroup -PatchGroup "Production"
```

- Para obtener información sobre la API, consulte DescribeInstancePatchStatesForPatchGroup en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeInstancePatches`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeInstancePatches`.

CLI

AWS CLI

Ejemplo 1: obtención de los detalles del estado de una revisión en una instancia

En el siguiente ejemplo de `describe-instance-patches` se recuperan los detalles sobre la instancia especificada.

```
aws ssm describe-instance-patches \
    --instance-id "i-1234567890abcdef0"
```

Salida:

```
{
    "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--"
}
```

Ejemplo 2: obtención de una lista de las revisiones que se encuentran en el estado Faltante en una instancia

En el siguiente ejemplo de `describe-instance-patches` se recupera información sobre las revisiones que se encuentran en el estado Falta en la instancia especificada.

```
aws ssm describe-instance-patches \
    --instance-id "i-1234567890abcdef0" \
    --filters Key=State,Values=Missing
```

Salida:

```
{
    "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--"
}
```

Para obtener más información, consulte Conocimiento de los valores del estado de conformidad de parches en la Guía del usuario de AWS Systems Manager.

Ejemplo 3: obtención de una lista de las revisiones instaladas desde un valor de InstalledTime específico en una instancia

En el siguiente ejemplo de `describe-instance-patches` se recupera información sobre las revisiones instaladas desde un tiempo específico en la instancia especificada combinando el uso de `--filters` y `--query`.

```
aws ssm describe-instance-patches \
    --instance-id "i-1234567890abcdef0" \
    --filters Key=State,Values=Installed \
    --query "Patches[?InstalledTime >= `2023-01-01T16:00:00`]"
```

Salida:

```
{
```

```
    "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--"
}
```

- Para obtener información sobre la API, consulte DescribeInstancePatches en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se obtienen los detalles de conformidad de las revisiones de una instancia.

```
Get-SSMInstancePatch -InstanceId "i-08ee91c0b17045407"
```

- Para obtener información sobre la API, consulte DescribeInstancePatches en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se obtienen los detalles de conformidad de las revisiones de una instancia.

```
Get-SSMInstancePatch -InstanceId "i-08ee91c0b17045407"
```

- Para obtener información sobre la API, consulte DescribeInstancePatches en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeMaintenanceWindowExecutionTaskInvocations`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeMaintenanceWindowExecutionTaskInvocations`.

CLI

AWS CLI

Obtención de las invocaciones de tareas específicas hechas para la ejecución de una tarea en un periodo de mantenimiento

En el siguiente ejemplo de `describe-maintenance-window-execution-task-invocations` se muestran las invocaciones de la tarea especificada que se ejecutó como parte de la ejecución del periodo de mantenimiento especificado.

```
aws ssm describe-maintenance-window-execution-task-invocations \
    --window-execution-id "518d5565-5969-4cca-8f0e-da3b2a638355" \
    --task-id "ac0c6ae1-daa3-4a89-832e-d384503b6586"
```

Salida:

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

Para obtener más información, consulte Ver información sobre tareas y ejecuciones de tareas (AWS CLI) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeMaintenanceWindowExecutionTaskInvocations en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: En este ejemplo se enumeran las invocaciones de una tarea ejecutada como parte de una ejecución de un periodo de mantenimiento.

```
Get-SSMMaintenanceWindowExecutionTaskInvocationList -TaskId "ac0c6ae1-
daa3-4a89-832e-d384503b6586" -WindowExecutionId "518d5565-5969-4cca-8f0e-
da3b2a638355"
```

Salida:

```
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    :
```

- Para obtener información sobre la API, consulte DescribeMaintenanceWindowExecutionTaskInvocations en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: En este ejemplo se enumeran las invocaciones de una tarea ejecutada como parte de una ejecución de un periodo de mantenimiento.

```
Get-SSMMaintenanceWindowExecutionTaskInvocationList -TaskId "ac0c6ae1-
daa3-4a89-832e-d384503b6586" -WindowExecutionId "518d5565-5969-4cca-8f0e-
da3b2a638355"
```

Salida:

```
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    :
```

- Para obtener información sobre la API, consulte DescribeMaintenanceWindowExecutionTaskInvocations en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeMaintenanceWindowExecutionTasks`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeMaintenanceWindowExecutionTasks`.

CLI

AWS CLI

Enumeración de todas las tareas asociadas a la ejecución de un periodo de mantenimiento

En el siguiente ejemplo de `ssm describe-maintenance-window-execution-tasks` se enumeran todas las tareas asociadas a la ejecución del periodo de mantenimiento especificado.

```
aws ssm describe-maintenance-window-execution-tasks \
    --window-execution-id "518d5565-5969-4cca-8f0e-da3b2EXAMPLE"
```

Salida:

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

Para obtener más información, consulte Ver información sobre tareas y ejecuciones de tareas (AWS CLI) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeMaintenanceWindowExecutionTasks en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran las tareas asociadas a la ejecución de un periodo de mantenimiento.

```
Get-SSMMaintenanceWindowExecutionTaskList -WindowExecutionId
 "518d5565-5969-4cca-8f0e-da3b2a638355"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeMaintenanceWindowExecutionTasks en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran las tareas asociadas a la ejecución de un periodo de mantenimiento.

```
Get-SSMMaintenanceWindowExecutionTaskList -WindowExecutionId
 "518d5565-5969-4cca-8f0e-da3b2a638355"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeMaintenanceWindowExecutionTasks en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeMaintenanceWindowExecutions`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeMaintenanceWindowExecutions`.

CLI

AWS CLI

Ejemplo 1: enumeración de todas las ejecuciones de un periodo de mantenimiento

En el siguiente ejemplo de `describe-maintenance-window-executions` se enumeran todas las ejecuciones del periodo de mantenimiento especificado.

```
aws ssm describe-maintenance-window-executions \
    --window-id "mw-ab12cd34eEXAMPLE"
```

Salida:

```
{
    "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"
```

```
        }
    ]
}
```

Ejemplo 2: enumeración de todas las ejecuciones de un periodo de mantenimiento antes de una fecha especificada

En el siguiente ejemplo de `describe-maintenance-window-executions` se enumeran todas las ejecuciones del periodo de mantenimiento especificado antes de la fecha especificada.

```
aws ssm describe-maintenance-window-executions \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --filters "Key=ExecutedBefore,Values=2021-08-03T00:00:00Z"
```

Salida:

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

Ejemplo 3: enumeración de todas las ejecuciones de un periodo de mantenimiento después de una fecha especificada

En el siguiente ejemplo de `describe-maintenance-window-executions` se enumeran todas las ejecuciones del periodo de mantenimiento especificado después de la fecha especificada.

```
aws ssm describe-maintenance-window-executions \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --filters "Key=ExecutedAfter,Values=2021-08-04T00:00:00Z"
```

Salida:

```
{
    "WindowExecutions": [
        {
        "WindowId": "mw-ab12cd34eEXAMPLE",
        "WindowExecutionId": "6027b513-64fe-4cf0-be7d-1191aEXAMPLE",
        "Status": "IN_PROGRESS",
        "StartTime": "2021-08-04T11:00:00.000000-07:00"
        }
    ]
}
```

Para obtener más información, consulte Ver información sobre tareas y ejecuciones de tareas (AWS CLI) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeMaintenanceWindowExecutions en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran todas las ejecuciones de un periodo de mantenimiento.

```
Get-SSMMaintenanceWindowExecutionList -WindowId "mw-03eb9db42890fb82d"
```

Salida:

```
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
```

Ejemplo 2: en este ejemplo se enumeran todas las ejecuciones de un periodo de mantenimiento antes de una fecha especificada.

```
$option1 = @{Key="ExecutedBefore";Values=@("2016-11-04T05:00:00Z")}
```

```
Get-SSMMaintenanceWindowExecutionList -WindowId "mw-03eb9db42890fb82d" -Filter
 $option1
```

Ejemplo 3: en este ejemplo se enumeran todas las ejecuciones de un periodo de mantenimiento después de una fecha especificada.

```
$option1 = @{Key="ExecutedAfter";Values=@("2016-11-04T05:00:00Z")}
Get-SSMMaintenanceWindowExecutionList -WindowId "mw-03eb9db42890fb82d" -Filter
 $option1
```

- Para obtener información sobre la API, consulte DescribeMaintenanceWindowExecutions en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran todas las ejecuciones de un periodo de mantenimiento.

```
Get-SSMMaintenanceWindowExecutionList -WindowId "mw-03eb9db42890fb82d"
```

Salida:

```
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
```

Ejemplo 2: en este ejemplo se enumeran todas las ejecuciones de un periodo de mantenimiento antes de una fecha especificada.

```
$option1 = @{Key="ExecutedBefore";Values=@("2016-11-04T05:00:00Z")}
Get-SSMMaintenanceWindowExecutionList -WindowId "mw-03eb9db42890fb82d" -Filter
 $option1
```

Ejemplo 3: en este ejemplo se enumeran todas las ejecuciones de un periodo de mantenimiento después de una fecha especificada.

```
$option1 = @{Key="ExecutedAfter";Values=@("2016-11-04T05:00:00Z")}
```

```
Get-SSMMaintenanceWindowExecutionList -WindowId "mw-03eb9db42890fb82d" -Filter
 $option1
```

- Para obtener información sobre la API, consulte DescribeMaintenanceWindowExecutions en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeMaintenanceWindowTargets`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeMaintenanceWindowTargets`.

CLI

AWS CLI

Ejemplo 1: enumeración de todos los destinos de un periodo de mantenimiento

En el siguiente ejemplo de `describe-maintenance-window-targets` se enumeran todos los destinos de un periodo de mantenimiento.

```
aws ssm describe-maintenance-window-targets \
    --window-id "mw-06cf17cbefEXAMPLE"
```

Salida:

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

Ejemplo 2: enumeración de todos los destinos de un periodo de mantenimiento que coincida con un valor específico de información del propietario

En este ejemplo de `describe-maintenance-window-targets` se enumeran todos los destinos de un periodo de mantenimiento con un valor específico.

```
aws ssm describe-maintenance-window-targets \
    --window-id "mw-0ecb1226ddEXAMPLE" \
    --filters "Key=OwnerInformation,Values=CostCenter1"
```

Salida:

```
{
    "Targets": [
        {
            "WindowId": "mw-0ecb1226ddEXAMPLE",
            "WindowTargetId": "da89dcc3-7f9c-481d-ba2b-edcb7d0057f9",
            "ResourceType": "INSTANCE",
            "Targets": [
                {
```

```
                    "Key": "tag:Environment",
                    "Values": [
                        "Prod"
                    ]
                }
            ],
            "OwnerInformation": "CostCenter1",
            "Name": "ProdTarget1"
        }
    ]
}
```

Para obtener más información, consulte Ver información sobre períodos de mantenimiento (AWS CLI) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeMaintenanceWindowTargets en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran todos los destinos de un periodo de mantenimiento.

```
Get-SSMMaintenanceWindowTarget -WindowId "mw-06cf17cbefcb4bf4f"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeMaintenanceWindowTargets en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran todos los destinos de un periodo de mantenimiento.

```
Get-SSMMaintenanceWindowTarget -WindowId "mw-06cf17cbefcb4bf4f"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeMaintenanceWindowTargets en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeMaintenanceWindowTasks`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeMaintenanceWindowTasks`.

CLI

AWS CLI

Ejemplo 1: enumeración de todas las tareas de un periodo de mantenimiento

En el siguiente ejemplo de `describe-maintenance-window-tasks` se enumeran todas las tareas del periodo de mantenimiento especificado.

```
aws ssm describe-maintenance-window-tasks \
```

```
    --window-id "mw-06cf17cbefEXAMPLE"
```

Salida:

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

Ejemplo 2: enumeración de todas las tareas de un periodo de mantenimiento que invoca el documento de comandos AWS-RunPowerShellScript

En el siguiente ejemplo de `describe-maintenance-window-tasks` se enumeran todas las tareas del periodo de mantenimiento especificado que invoca el documento de comandos `AWS-RunPowerShellScript`.

```
aws ssm describe-maintenance-window-tasks \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --filters "Key=TaskArn,Values=AWS-RunPowerShellScript"
```

Salida:

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

Ejemplo 3: enumeración de todas las tareas de un periodo de mantenimiento que tienen una prioridad de 3

En el siguiente ejemplo de `describe-maintenance-window-tasks` se enumeran todas las tareas del periodo de mantenimiento especificado que tienen una `Priority` de `3`.

```
aws ssm describe-maintenance-window-tasks \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --filters "Key=Priority,Values=3"
```

Salida:

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

Ejemplo 4: enumeración de todas las tareas de un periodo de mantenimiento que tienen una prioridad de 1 y uso del comando de ejecución

En este ejemplo de `describe-maintenance-window-tasks` se enumeran todas las tareas del periodo de mantenimiento especificado que tienen una `Priority` de 1 y el uso del `Run Command`.

```
aws ssm describe-maintenance-window-tasks \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --filters "Key=Priority,Values=1" "Key=TaskType,Values=RUN_COMMAND"
```

Salida:

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

Para obtener más información, consulte Ver información sobre períodos de mantenimiento (AWS CLI) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeMaintenanceWindowTasks en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran todas las tareas de un periodo de mantenimiento.

```
Get-SSMMaintenanceWindowTaskList -WindowId "mw-06cf17cbefcb4bf4f"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeMaintenanceWindowTasks en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran todas las tareas de un periodo de mantenimiento.

```
Get-SSMMaintenanceWindowTaskList -WindowId "mw-06cf17cbefcb4bf4f"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte DescribeMaintenanceWindowTasks en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribeMaintenanceWindows`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeMaintenanceWindows`.

CLI

AWS CLI

Ejemplo 1: enumeración de todos los periodos de mantenimiento

En el siguiente ejemplo de `describe-maintenance-windows` se enumeran todos los periodos de mantenimiento de su cuenta de AWS en la región actual.

```
aws ssm describe-maintenance-windows
```

Salida:

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

Ejemplo 2: enumeración de todos los periodos de mantenimiento habilitados

En el siguiente ejemplo de `describe-maintenance-windows` se enumeran todos los periodos de mantenimiento habilitados.

```
aws ssm describe-maintenance-windows \
    --filters "Key=Enabled,Values=true"
```

Ejemplo 3: enumeración de los periodos de mantenimiento que coincidan con un nombre específico

En este ejemplo de `describe-maintenance-windows` se enumeran todos los periodos de mantenimiento con el nombre especificado.

```
aws ssm describe-maintenance-windows \
    --filters "Key=Name,Values=MyMaintenanceWindow"
```

Para obtener más información, consulte Ver información sobre períodos de mantenimiento (AWS CLI) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeMaintenanceWindows en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran todos los periodos de mantenimiento de su cuenta.

```
Get-SSMMaintenanceWindowList
```

Salida:

```
Cutoff   : 1
Duration : 4
Enabled  : True
Name     : My-First-Maintenance-Window
WindowId : mw-06d59c1a07c022145
```

- Para obtener información sobre la API, consulte DescribeMaintenanceWindows en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran todos los periodos de mantenimiento de su cuenta.

```
Get-SSMMaintenanceWindowList
```

Salida:

```
Cutoff   : 1
Duration : 4
Enabled  : True
Name     : My-First-Maintenance-Window
WindowId : mw-06d59c1a07c022145
```

- Para obtener información sobre la API, consulte DescribeMaintenanceWindows en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Uso de **`DescribeOpsItems`** con un SDK de AWS o la CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeOpsItems`.

CLI

AWS CLI

Visualización de un conjunto de OpsItems

En el siguiente ejemplo de `describe-ops-items` se muestra una lista de todos los OpsItems abiertos en su cuenta de AWS.

```
aws ssm describe-ops-items \
    --ops-item-filters "Key=Status,Values=Open,Operator=Equal"
```

Salida:

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

Para obtener más información, consulte Trabajo con OpsItems en la AWSGuía del usuario de Systems Manager.

- Para obtener información sobre la API, consulte DescribeOpsItems en la Referencia de comandos de la AWS CLI.

Java

SDK para Java 2.x

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
    /**
     * Describes AWS SSM OpsItems asynchronously.
     *
     * @param key The key to filter OpsItems by (e.g., OPS_ITEM_ID).
     *
     * This method initiates an asynchronous request to describe SSM OpsItems.
     * If the request is successful, it prints the title and status of each
 OpsItem.
     * If an exception occurs, it handles the error appropriately.
     */
```

```
   public void describeOpsItems(String key) {
       OpsItemFilter filter = OpsItemFilter.builder()
               .key(OpsItemFilterKey.OPS_ITEM_ID)
               .values(key)
               .operator(OpsItemFilterOperator.EQUAL)
               .build();

       DescribeOpsItemsRequest itemsRequest = DescribeOpsItemsRequest.builder()
               .maxResults(10)
               .opsItemFilters(filter)
               .build();

       CompletableFuture<Void> future = CompletableFuture.runAsync(() -> {
           getAsyncClient().describeOpsItems(itemsRequest)
                   .thenAccept(itemsResponse -> {
                       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());
                       }
                   })
                   .exceptionally(ex -> {
                       throw new CompletionException(ex);
                   }).join();
       }).exceptionally(ex -> {
           Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
           if (cause instanceof SsmException) {
               throw new RuntimeException("SSM error: " + cause.getMessage(),
cause);
           } else {
               throw new RuntimeException("Unexpected error: " +
cause.getMessage(), cause);
           }
       });

       try {
           future.join();
       } catch (CompletionException ex) {
           throw ex.getCause() instanceof RuntimeException ? (RuntimeException)
ex.getCause() : ex;
       }
   }
```

- Para obtener información sobre la API, consulte DescribeOpsItems en la Referencia de la API de AWS SDK for Java 2.x.

JavaScript

SDK para JavaScript (v3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
import {
  OpsItemFilterOperator,
  OpsItemFilterKey,
  paginateDescribeOpsItems,
  SSMClient,
} from "@aws-sdk/client-ssm";
import { parseArgs } from "node:util";

/**
 * Describe SSM OpsItems.
 * @param {{ opsItemId: string }}
 */
export const main = async ({ opsItemId }) => {
  const client = new SSMClient({});
  try {
    const describeOpsItemsPaginated = [];
    for await (const page of paginateDescribeOpsItems(
      { client },
      {
        OpsItemFilters: {
          Key: OpsItemFilterKey.OPSITEM_ID,
          Operator: OpsItemFilterOperator.EQUAL,
          Values: opsItemId,
        },
      },
    )) {
```

```
      describeOpsItemsPaginated.push(...page.OpsItemSummaries);
    }
    console.log("Here are the ops items:");
    console.log(describeOpsItemsPaginated);
    return { OpsItemSummaries: describeOpsItemsPaginated };
  } catch (caught) {
    if (caught instanceof Error && caught.name === "MissingParameter") {
      console.warn(`${caught.message}. Did you provide this value?`);
    }
    throw caught;
  }
};
```

- Para obtener información sobre la API, consulte DescribeOpsItems en la Referencia de la API de AWS SDK para JavaScript.

Python

SDK para Python (Boto3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
class OpsItemWrapper:
    """Encapsulates AWS Systems Manager OpsItem actions."""

    def __init__(self, ssm_client):
        """
        :param ssm_client: A Boto3 Systems Manager client.
        """
        self.ssm_client = ssm_client
        self.id = None

    @classmethod
    def from_client(cls):
        """
        :return: A OpsItemWrapper instance.
```

```
        """
        ssm_client = boto3.client("ssm")
        return cls(ssm_client)


    def describe(self):
        """
        Describe an OpsItem.
        """
        try:
            paginator = self.ssm_client.get_paginator("describe_ops_items")
            ops_items = []
            for page in paginator.paginate(
                OpsItemFilters=[
                    {"Key": "OpsItemId", "Values": [self.id], "Operator":
"Equal"}
                ]
            ):
                ops_items.extend(page["OpsItemSummaries"])

            for item in ops_items:
                print(
                    f"The item title is {item['Title']} and the status is
{item['Status']}"
                )
            return len(ops_items) > 0
        except ClientError as err:
            logger.error(
                "Couldn't describe ops item %s. Here's why: %s: %s",
                self.id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
```

- Para obtener información sobre la API, consulte DescribeOpsItems en la Referencia de la API de AWS SDK para Python (Boto3).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Uso de **`DescribeParameters`** con un SDK de AWS o la CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribeParameters`.

CLI

AWS CLI

Ejemplo 1: Creación de una lista de todos los parámetros

En el siguiente ejemplo de `describe-parameters` se enumeran todos los parámetros de la cuenta y la región de AWS actuales.

```
aws ssm describe-parameters
```

Salida:

```
{
    "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": []
        }
    ]
}
```

Ejemplo 2: Creación de una lista de todos los parámetros que coinciden con metadatos específicos

En este ejemplo de `describe-parameters` se enumeran todos los parámetros que coinciden con un filtro.

aws ssm describe-parameters --filters "Key=Type,Values=StringList"

Salida:

```
{
    "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": []
        }
    ]
}
```

Para obtener más información, consulte Búsqueda de parámetros de Systems Manager en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribeParameters en la Referencia de comandos de la AWS CLI.

Java

SDK para Java 2.x

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de 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);
        }
    }
}
```

- Para obtener información sobre la API, consulte DescribeParameters en la Referencia de la API de AWS SDK for Java 2.x.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran todos los parámetros.

```
Get-SSMParameterList
```

Salida:

```
Description      :
KeyId            :
LastModifiedDate : 3/3/2017 6:58:23 PM
LastModifiedUser : arn:aws:iam::123456789012:user/admin
Name             : Welcome
Type             : String
```

- Para obtener información sobre la API, consulte DescribeParameters en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran todos los parámetros.

```
Get-SSMParameterList
```

Salida:

```
Description      :
KeyId            :
LastModifiedDate : 3/3/2017 6:58:23 PM
LastModifiedUser : arn:aws:iam::123456789012:user/admin
Name             : Welcome
Type             : String
```

- Para obtener información sobre la API, consulte DescribeParameters en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Rust

SDK para Rust

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de 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(())
}
```

- Para obtener información sobre la API, consulte DescribeParameters en Referencia de la API de AWS SDK para Rust.

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribePatchBaselines`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribePatchBaselines`.

CLI

AWS CLI

Ejemplo 1: enumeración de todas las líneas de base de revisiones

En el siguiente ejemplo de `describe-patch-baselines` se recuperan los detalles de todas las líneas de base de revisiones de su cuenta en la región actual.

```
aws ssm describe-patch-baselines
```

Salida:

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

Ejemplo 2: enumeración de todas las líneas de base de revisiones proporcionadas por AWS

En el siguiente ejemplo de `describe-patch-baselines` se enumeran todas las líneas de base de revisiones proporcionadas por AWS.

```
aws ssm describe-patch-baselines \
    --filters "Key=OWNER,Values=[AWS]"
```

Ejemplo 3: enumeración de todas las líneas de base de revisiones de su propiedad

En el siguiente ejemplo de `describe-patch-baselines` se enumeran todas las líneas de base de revisiones personalizadas creadas en su cuenta en la región actual.

```
aws ssm describe-patch-baselines \
    --filters "Key=OWNER,Values=[Self]"
```

Para obtener más información, consulte Acerca de las líneas de base de revisiones personalizadas y predefinidas en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribePatchBaselines en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran todas las líneas de base de revisiones.

```
Get-SSMPatchBaseline
```

Salida:

```
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...
```

Ejemplo 2: en este ejemplo se enumeran todas las líneas de base de revisiones proporcionadas por AWS. La sintaxis utilizada en este ejemplo requiere la versión 3 o posterior de PowerShell.

```
$filter1 = @{Key="OWNER";Values=@("AWS")}
```

Salida:

```
Get-SSMPatchBaseline -Filter $filter1
```

Ejemplo 3: en este ejemplo se enumeran todas las líneas de base de revisiones en las que usted es propietario. La sintaxis utilizada en este ejemplo requiere la versión 3 o posterior de PowerShell.

```
$filter1 = @{Key="OWNER";Values=@("Self")}
```

Salida:

```
Get-SSMPatchBaseline -Filter $filter1
```

Ejemplo 4: con la versión 2 de PowerShell, debe usar New-Object para crear cada etiqueta.

```
$filter1 = New-Object
 Amazon.SimpleSystemsManagement.Model.PatchOrchestratorFilter
$filter1.Key = "OWNER"
$filter1.Values = "AWS"

Get-SSMPatchBaseline -Filter $filter1
```

Salida:

```
BaselineDescription                     BaselineId
                            BaselineName            DefaultBaselin

                                                    e
------------------                      ----------
                            ------------            --------------
Default Patch Baseline Provided by AWS. arn:aws:ssm:us-
west-2:123456789012:patchbaseline/pb-04fb4ae6142167966 AWS-DefaultPatchBaseline
 True
```

- Para obtener información sobre la API, consulte DescribePatchBaselines en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran todas las líneas de base de revisiones.

```
Get-SSMPatchBaseline
```

Salida:

```
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...
```

Ejemplo 2: en este ejemplo se enumeran todas las líneas de base de revisiones proporcionadas por AWS. La sintaxis utilizada en este ejemplo requiere la versión 3 o posterior de PowerShell.

```
$filter1 = @{Key="OWNER";Values=@("AWS")}
```

Salida:

```
Get-SSMPatchBaseline -Filter $filter1
```

Ejemplo 3: en este ejemplo se enumeran todas las líneas de base de revisiones en las que usted es propietario. La sintaxis utilizada en este ejemplo requiere la versión 3 o posterior de PowerShell.

```
$filter1 = @{Key="OWNER";Values=@("Self")}
```

Salida:

```
Get-SSMPatchBaseline -Filter $filter1
```

Ejemplo 4: con la versión 2 de PowerShell, debe usar New-Object para crear cada etiqueta.

```
$filter1 = New-Object
 Amazon.SimpleSystemsManagement.Model.PatchOrchestratorFilter
$filter1.Key = "OWNER"
$filter1.Values = "AWS"

Get-SSMPatchBaseline -Filter $filter1
```

Salida:

```
BaselineDescription                     BaselineId
                            BaselineName            DefaultBaselin


                                                    e
------------------                      ----------
                            ------------            --------------
Default Patch Baseline Provided by AWS. arn:aws:ssm:us-
west-2:123456789012:patchbaseline/pb-04fb4ae6142167966 AWS-DefaultPatchBaseline
 True
```

- Para obtener información sobre la API, consulte DescribePatchBaselines en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribePatchGroupState`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribePatchGroupState`.

CLI

AWS CLI

Obtención del estado de un grupo de revisiones

En el siguiente ejemplo de `describe-patch-group-state` se recupera el resumen general de conformidad de las revisiones de un grupo de revisiones.

```
aws ssm describe-patch-group-state \
    --patch-group "Production"
```

Salida:

```
{
    "Instances": 21,
    "InstancesWithCriticalNonCompliantPatches": 1,
    "InstancesWithFailedPatches": 2,
    "InstancesWithInstalledOtherPatches": 3,
    "InstancesWithInstalledPatches": 21,
    "InstancesWithInstalledPendingRebootPatches": 2,
    "InstancesWithInstalledRejectedPatches": 1,
    "InstancesWithMissingPatches": 3,
    "InstancesWithNotApplicablePatches": 4,
    "InstancesWithOtherNonCompliantPatches": 1,
    "InstancesWithSecurityNonCompliantPatches": 1,
    "InstancesWithUnreportedNotApplicablePatches": 2
}
```

Para obtener más información, consulte Acerca de los grupos de revisiones <https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-patchgroups.html>__ y Conocimiento de los valores del estado de conformidad de parches en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribePatchGroupState en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se obtiene un resumen general de conformidad de las revisiones de un grupo de revisiones.

```
Get-SSMPatchGroupState -PatchGroup "Production"
```

Salida:

```
Instances                           : 4
InstancesWithFailedPatches          : 1
InstancesWithInstalledOtherPatches : 4
InstancesWithInstalledPatches       : 3
InstancesWithMissingPatches         : 0
InstancesWithNotApplicablePatches   : 0
```

- Para obtener información sobre la API, consulte DescribePatchGroupState en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se obtiene un resumen general de conformidad de las revisiones de un grupo de revisiones.

```
Get-SSMPatchGroupState -PatchGroup "Production"
```

Salida:

```
Instances                           : 4
InstancesWithFailedPatches          : 1
InstancesWithInstalledOtherPatches : 4
InstancesWithInstalledPatches       : 3
InstancesWithMissingPatches         : 0
InstancesWithNotApplicablePatches   : 0
```

- Para obtener información sobre la API, consulte DescribePatchGroupState en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`DescribePatchGroups`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `DescribePatchGroups`.

CLI

AWS CLI

Visualización de los registros de grupos de revisiones

En el siguiente ejemplo de `describe-patch-groups` se enumeran los registros de los grupos de revisiones.

```
aws ssm describe-patch-groups
```

Salida:

```
{
    "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
            }
```

```
        },
        ...
    ]
}
```

Para obtener más información, consulte Creación de un grupo de revisiones <https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-group-tagging.html>__ y Añadir un grupo de revisiones a una línea de base de revisiones en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte DescribePatchGroups en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran los registros de los grupos de revisiones.

```
Get-SSMPatchGroup
```

Salida:

```
BaselineIdentity                                          PatchGroup
----------------                                          ----------
Amazon.SimpleSystemsManagement.Model.PatchBaselineIdentity Production
```

- Para obtener información sobre la API, consulte DescribePatchGroups en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran los registros de los grupos de revisiones.

```
Get-SSMPatchGroup
```

Salida:

```
BaselineIdentity                                          PatchGroup
----------------                                          ----------
Amazon.SimpleSystemsManagement.Model.PatchBaselineIdentity Production
```

- Para obtener información sobre la API, consulte DescribePatchGroups en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`GetAutomationExecution`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `GetAutomationExecution`.

CLI

AWS CLI

Visualización de los detalles sobre una ejecución de automatización

En el siguiente ejemplo de `get-automation-execution` se muestra información detallada sobre una ejecución de Automatización.

```
aws ssm get-automation-execution \
    --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE
```

Salida:

```
{
    "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
        }
    }
}
```

Para obtener más información, consulte Tutorial: Actualizar una AMI de Linux (AWS CLI) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte GetAutomationExecution en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se muestran los detalles de una ejecución de Automatización.

```
Get-SSMAutomationExecution -AutomationExecutionId "4105a4fc-
f944-11e6-9d32-8fb2db27a909"
```

Salida:

```
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...}
```

Ejemplo 2: en este ejemplo se enumeran los detalles de los pasos del identificador de ejecución de automatización indicado

```
Get-SSMAutomationExecution -AutomationExecutionId e1d2bad3-4567-8901-
ae23-456c7c8901be | Select-Object -ExpandProperty StepExecutions | Select-Object
 StepName, Action, StepStatus, ValidNextSteps
```

Salida:

```
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    {}
```

- Para obtener información sobre la API, consulte GetAutomationExecution en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se muestran los detalles de una ejecución de Automatización.

```
Get-SSMAutomationExecution -AutomationExecutionId "4105a4fc-
f944-11e6-9d32-8fb2db27a909"
```

Salida:

```
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...}
```

Ejemplo 2: en este ejemplo se enumeran los detalles de los pasos del identificador de ejecución de automatización indicado

```
Get-SSMAutomationExecution -AutomationExecutionId e1d2bad3-4567-8901-
ae23-456c7c8901be | Select-Object -ExpandProperty StepExecutions | Select-Object
 StepName, Action, StepStatus, ValidNextSteps
```

Salida:

```
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    {}
```

- Para obtener información sobre la API, consulte GetAutomationExecution en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`GetCommandInvocation`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `GetCommandInvocation`.

CLI

AWS CLI

Visualización de los detalles de la invocación de un comando

En el siguiente ejemplo de `get-command-invocation` se muestran todas las invocaciones del comando especificado en la instancia especificada.

```
aws ssm get-command-invocation \
    --command-id "ef7fdfd8-9b57-4151-a15c-db9a12345678" \
    --instance-id "i-1234567890abcdef0"
```

Salida:

```
{
    "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
    }
}
```

Para obtener más información, consulte Descripción de los estados del comando en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte GetCommandInvocation en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se muestran los detalles de un comando ejecutado en una instancia.

```
Get-SSMCommandInvocationDetail -InstanceId "i-0cb2b964d3e14fd9f" -CommandId
 "b8eac879-0541-439d-94ec-47a80d554f44"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte GetCommandInvocation en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se muestran los detalles de un comando ejecutado en una instancia.

```
Get-SSMCommandInvocationDetail -InstanceId "i-0cb2b964d3e14fd9f" -CommandId
 "b8eac879-0541-439d-94ec-47a80d554f44"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte GetCommandInvocation en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`GetConnectionStatus`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `GetConnectionStatus`.

CLI

AWS CLI

Visualización del estado de conexión de una instancia administrada

En este ejemplo de `get-connection-status` se devuelve el estado de conexión de la instancia administrada especificada.

```
aws ssm get-connection-status \
    --target i-1234567890abcdef0
```

Salida:

```
{
    "Target": "i-1234567890abcdef0",
    "Status": "connected"
}
```

- Para obtener información sobre la API, consulte GetConnectionStatus en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se recupera el estado de conexión del Administrador de sesiones de una instancia para determinar si está conectada y lista para recibir las conexiones del Administrador de sesiones.

```
Get-SSMConnectionStatus -Target i-0a1caf234f12d3dc4
```

Salida:

```
Status    Target
------    ------
Connected i-0a1caf234f12d3dc4
```

- Para obtener información sobre la API, consulte GetConnectionStatus en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se recupera el estado de conexión del Administrador de sesiones de una instancia para determinar si está conectada y lista para recibir las conexiones del Administrador de sesiones.

```
Get-SSMConnectionStatus -Target i-0a1caf234f12d3dc4
```

Salida:

```
Status    Target
------    ------
Connected i-0a1caf234f12d3dc4
```

- Para obtener información sobre la API, consulte GetConnectionStatus en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`GetDefaultPatchBaseline`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `GetDefaultPatchBaseline`.

CLI

AWS CLI

Ejemplo 1: visualización de la línea de base de revisiones de Windows predeterminada

En el siguiente ejemplo de `get-default-patch-baseline` se recuperan los detalles de la línea de base de revisiones predeterminada para Windows Server.

```
aws ssm get-default-patch-baseline
```

Salida:

```
{
  "BaselineId": "pb-0713accee01612345",
  "OperatingSystem": "WINDOWS"
}
```

Ejemplo 2: visualización de la línea de base de revisiones predeterminada para Amazon Linux

En el siguiente ejemplo de `get-default-patch-baseline` se recuperan los detalles de la línea de base de revisiones predeterminada para Amazon Linux.

```
aws ssm get-default-patch-baseline \
    --operating-system AMAZON_LINUX
```

Salida:

```
{
    "BaselineId": "pb-047c6eb9c8fc12345",
    "OperatingSystem": "AMAZON_LINUX"
}
```

Para obtener más información, consulte Acerca de las líneas de base de revisiones personalizadas y predefinidas <https://docs.aws.amazon.com/systems-manager/latest/

userguide/sysman-patch-baselines.html>__ y Configuración de una línea de base de revisiones existente como valor predeterminado en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte GetDefaultPatchBaseline en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se muestra la línea de base de revisiones predeterminada.

```
Get-SSMDefaultPatchBaseline
```

Salida:

```
arn:aws:ssm:us-west-2:123456789012:patchbaseline/pb-04fb4ae6142167966
```

- Para obtener información sobre la API, consulte GetDefaultPatchBaseline en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se muestra la línea de base de revisiones predeterminada.

```
Get-SSMDefaultPatchBaseline
```

Salida:

```
arn:aws:ssm:us-west-2:123456789012:patchbaseline/pb-04fb4ae6142167966
```

- Para obtener información sobre la API, consulte GetDefaultPatchBaseline en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **GetDeployablePatchSnapshotForInstance** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `GetDeployablePatchSnapshotForInstance`.

CLI

AWS CLI

Recuperación de la instantánea actual de la línea de base de revisiones que usa una instancia

En el siguiente ejemplo de `get-deployable-patch-snapshot-for-instance` se recuperan los detalles de la instantánea actual correspondiente a la línea de base de revisiones especificada que usa una instancia. Este comando debe ejecutarse desde la instancia con las credenciales de la instancia. Para asegurarse de que usa las credenciales de la instancia, ejecute `aws configure` y especifique solo la región de la instancia. Deje vacíos los campos `Access Key` y `Secret Key`.

Consejo: Use `uuidgen` para generar un `snapshot-id`.

```
aws ssm get-deployable-patch-snapshot-for-instance \
    --instance-id "i-1234567890abcdef0" \
    --snapshot-id "521c3536-930c-4aa9-950e-01234567abcd"
```

Salida:

```
{
    "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"
}
```

Para obtener más información, consulte Nombre del parámetro: Snapshot ID en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte GetDeployablePatchSnapshotForInstance en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se muestra la instantánea actual de la línea de base de revisiones que usa una instancia. Este comando debe ejecutarse desde la instancia con las credenciales de la instancia. Para garantizar que utiliza las credenciales de la instancia, en el ejemplo pasa un objeto **`Amazon.Runtime.InstanceProfileAWSCredentials`** al parámetro Credentials.

```
$credentials = [Amazon.Runtime.InstanceProfileAWSCredentials]::new()
Get-SSMDeployablePatchSnapshotForInstance -SnapshotId "4681775b-098f-4435-
a956-0ef33373ac11" -InstanceId "i-0cb2b964d3e14fd9f" -Credentials $credentials
```

Salida:

```
InstanceId          SnapshotDownloadUrl
----------          -------------------
i-0cb2b964d3e14fd9f https://patch-baseline-snapshot-us-west-2.s3-us-
west-2.amazonaws.com/853d0d3db0f0cafe...1692/4681775b-098f-4435...
```

Ejemplo 2: en este ejemplo se muestra cómo obtener la dirección SnapshotDownloadUrl completa. Este comando debe ejecutarse desde la instancia con las credenciales de la instancia. Para garantizar que utiliza las credenciales de la instancia, en el ejemplo se configura la sesión de PowerShell para que utilice un objeto **`Amazon.Runtime.InstanceProfileAWSCredentials`**.

```
Set-AWSCredential -Credential
 ([Amazon.Runtime.InstanceProfileAWSCredentials]::new())
(Get-SSMDeployablePatchSnapshotForInstance -SnapshotId "4681775b-098f-4435-
a956-0ef33373ac11" -InstanceId "i-0cb2b964d3e14fd9f").SnapshotDownloadUrl
```

Salida:

```
https://patch-baseline-snapshot-us-west-2.s3-us-
west-2.amazonaws.com/853d0d3db0f0cafe...
```

- Para obtener información sobre la API, consulte GetDeployablePatchSnapshotForInstance en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se muestra la instantánea actual de la línea de base de revisiones que usa una instancia. Este comando debe ejecutarse desde la instancia con las credenciales de la instancia. Para garantizar que utiliza las credenciales de la instancia, en el ejemplo pasa un objeto **`Amazon.Runtime.InstanceProfileAWSCredentials`** al parámetro Credentials.

```
$credentials = [Amazon.Runtime.InstanceProfileAWSCredentials]::new()
Get-SSMDeployablePatchSnapshotForInstance -SnapshotId "4681775b-098f-4435-
a956-0ef33373ac11" -InstanceId "i-0cb2b964d3e14fd9f" -Credentials $credentials
```

Salida:

```
InstanceId          SnapshotDownloadUrl
----------          -------------------
i-0cb2b964d3e14fd9f https://patch-baseline-snapshot-us-west-2.s3-us-
west-2.amazonaws.com/853d0d3db0f0cafe...1692/4681775b-098f-4435...
```

Ejemplo 2: en este ejemplo se muestra cómo obtener la dirección SnapshotDownloadUrl completa. Este comando debe ejecutarse desde la instancia con las credenciales de la instancia. Para garantizar que utiliza las credenciales de la instancia, en el ejemplo se configura la sesión de PowerShell para que utilice un objeto **`Amazon.Runtime.InstanceProfileAWSCredentials`**.

```
Set-AWSCredential -Credential
 ([Amazon.Runtime.InstanceProfileAWSCredentials]::new())
(Get-SSMDeployablePatchSnapshotForInstance -SnapshotId "4681775b-098f-4435-
a956-0ef33373ac11" -InstanceId "i-0cb2b964d3e14fd9f").SnapshotDownloadUrl
```

Salida:

```
https://patch-baseline-snapshot-us-west-2.s3-us-
west-2.amazonaws.com/853d0d3db0f0cafe...
```

- Para obtener información sobre la API, consulte GetDeployablePatchSnapshotForInstance en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **GetDocument** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `GetDocument`.

CLI

AWS CLI

Obtención del contenido de un documento

En el siguiente ejemplo de `get-document` se muestra el contenido de un documento de Systems Manager.

```
aws ssm get-document \
    --name "AWS-RunShellScript"
```

Salida:

```
{
    "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"
}
```

Para obtener más información, consulte Documentos de AWS Systems Manager en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte GetDocument en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se devuelve el contenido de un documento.

```
Get-SSMDocument -Name "RunShellScript"
```

Salida:

```
Content
-------
{...
```

Ejemplo 2: en este ejemplo se muestra el contenido completo de un documento.

```
(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 }}"
         }
      }
   ]
}
```

- Para obtener información sobre la API, consulte GetDocument en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se devuelve el contenido de un documento.

```
Get-SSMDocument -Name "RunShellScript"
```

Salida:

```
Content
-------
{...
```

Ejemplo 2: en este ejemplo se muestra el contenido completo de un documento.

```
(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 }}"
            }
         }
      ]
}
```

- Para obtener información sobre la API, consulte GetDocument en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`GetInventory`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `GetInventory`.

CLI

AWS CLI

Visualización del inventario

En este ejemplo se obtienen los metadatos personalizados del inventario.

Comando:

```
aws ssm get-inventory
```

Salida:

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

- Para obtener información sobre la API, consulte GetInventory en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se obtienen los metadatos personalizados del inventario.

```
Get-SSMInventory
```

Salida:

```
Data
     Id
----
     --
{[AWS:InstanceInformation,
 Amazon.SimpleSystemsManagement.Model.InventoryResultItem]} i-0cb2b964d3e14fd9f
```

- Para obtener información sobre la API, consulte GetInventory en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se obtienen los metadatos personalizados del inventario.

```
Get-SSMInventory
```

Salida:

```
Data
     Id
----
     --
{[AWS:InstanceInformation,
 Amazon.SimpleSystemsManagement.Model.InventoryResultItem]} i-0cb2b964d3e14fd9f
```

- Para obtener información sobre la API, consulte GetInventory en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`GetInventorySchema`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `GetInventorySchema`.

CLI

AWS CLI

Visualización del esquema del inventario

En este ejemplo se obtiene una lista con los nombres de los tipos de inventario de la cuenta.

Comando:

```
aws ssm get-inventory-schema
```

Salida:

```
{
  "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--"
}
```

Visualización del esquema de inventario de un tipo de inventario específico

En este ejemplo se devuelve el esquema del inventario de un tipo de inventario AWS:AWSComponent.

Comando:

```
aws ssm get-inventory-schema --type-name "AWS:AWSComponent"
```

- Para obtener información sobre la API, consulte GetInventorySchema en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo devuelve una lista con los nombres de los tipos de inventario de la cuenta.

```
Get-SSMInventorySchema
```

- Para obtener información sobre la API, consulte GetInventorySchema en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo devuelve una lista con los nombres de los tipos de inventario de la cuenta.

```
Get-SSMInventorySchema
```

- Para obtener información sobre la API, consulte GetInventorySchema en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`GetMaintenanceWindow`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `GetMaintenanceWindow`.

CLI

AWS CLI

Obtención de información sobre un periodo de mantenimiento

En el siguiente ejemplo de `get-maintenance-window` se recuperan los detalles sobre el periodo de mantenimiento especificado.

```
aws ssm get-maintenance-window \
    --window-id "mw-03eb9db428EXAMPLE"
```

Salida:

```
{
    "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"
}
```

Para obtener más información, consulte Ver información sobre períodos de mantenimiento (AWS CLI) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte GetMaintenanceWindow en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se obtienen los detalles sobre un periodo de mantenimiento.

```
Get-SSMMaintenanceWindow -WindowId "mw-03eb9db42890fb82d"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte GetMaintenanceWindow en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se obtienen los detalles sobre un periodo de mantenimiento.

```
Get-SSMMaintenanceWindow -WindowId "mw-03eb9db42890fb82d"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte GetMaintenanceWindow en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`GetMaintenanceWindowExecution`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `GetMaintenanceWindowExecution`.

CLI

AWS CLI

Obtención de información sobre la ejecución de una tarea del periodo de mantenimiento

En el siguiente ejemplo de `get-maintenance-window-execution` se muestra información sobre una tarea que se ejecutó como parte de la ejecución del periodo de mantenimiento especificado.

```
aws ssm get-maintenance-window-execution \
    --window-execution-id "518d5565-5969-4cca-8f0e-da3b2EXAMPLE"
```

Salida:

```
{
    "Status": "SUCCESS",
    "TaskIds": [
        "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE"
    ],
    "StartTime": 1487692834.595,
    "EndTime": 1487692835.051,
    "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2EXAMPLE",
}
```

Para obtener más información, consulte Ver información sobre tareas y ejecuciones de tareas (AWS CLI) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte GetMaintenanceWindowExecution en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumera la información sobre una tarea ejecutada como parte de una ejecución de un periodo de mantenimiento.

```
Get-SSMMaintenanceWindowExecution -WindowExecutionId "518d5565-5969-4cca-8f0e-
da3b2a638355"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte GetMaintenanceWindowExecution en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumera la información sobre una tarea ejecutada como parte de una ejecución de un periodo de mantenimiento.

```
Get-SSMMaintenanceWindowExecution -WindowExecutionId "518d5565-5969-4cca-8f0e-
da3b2a638355"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte GetMaintenanceWindowExecution en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **GetMaintenanceWindowExecutionTask** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `GetMaintenanceWindowExecutionTask`.

CLI

AWS CLI

Obtención de información sobre la ejecución de una tarea del periodo de mantenimiento

En el siguiente ejemplo de `get-maintenance-window-execution-task` se muestra información sobre una tarea que forma parte de la ejecución del periodo de mantenimiento especificado.

```
aws ssm get-maintenance-window-execution-task \
    --window-execution-id "518d5565-5969-4cca-8f0e-da3b2EXAMPLE" \
    --task-id "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE"
```

Salida:

```
{
    "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"
}
```

Para obtener más información, consulte Ver información sobre tareas y ejecuciones de tareas (AWS CLI) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte GetMaintenanceWindowExecutionTask en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumera la información sobre una tarea ejecutada que formaba parte de una ejecución de un periodo de mantenimiento.

```
Get-SSMMaintenanceWindowExecutionTask -TaskId "ac0c6ae1-daa3-4a89-832e-
d384503b6586" -WindowExecutionId "518d5565-5969-4cca-8f0e-da3b2a638355"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte GetMaintenanceWindowExecutionTask en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumera la información sobre una tarea ejecutada que formaba parte de una ejecución de un periodo de mantenimiento.

```
Get-SSMMaintenanceWindowExecutionTask -TaskId "ac0c6ae1-daa3-4a89-832e-
d384503b6586" -WindowExecutionId "518d5565-5969-4cca-8f0e-da3b2a638355"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte GetMaintenanceWindowExecutionTask en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Uso de **`GetParameter`** con un SDK de AWS o la CLI

Los siguientes ejemplos de código muestran cómo utilizar `GetParameter`.

CLI

AWS CLI

Ejemplo 1: visualización del valor de un parámetro

En el siguiente ejemplo de `get-parameter` se enumera el valor del parámetro único especificado.

```
aws ssm get-parameter \
    --name "MyStringParameter"
```

Salida:

```
{
```

```
    "Parameter": {
        "Name": "MyStringParameter",
        "Type": "String",
        "Value": "Veni",
        "Version": 1,
        "LastModifiedDate": 1530018761.888,
        "ARN": "arn:aws:ssm:us-east-2:111222333444:parameter/MyStringParameter"
        "DataType": "text"
    }
}
```

Para obtener más información, consulte Uso de Parameter Store en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: descifrar el valor de un parámetro SecureString

En el siguiente ejemplo de `get-parameter` se cambia el valor del parámetro `SecureString` especificado.

```
aws ssm get-parameter \
    --name "MySecureStringParameter" \
    --with-decryption
```

Salida:

```
{
    "Parameter": {
        "Name": "MySecureStringParameter",
        "Type": "SecureString",
        "Value": "16679b88-310b-4895-a943-e0764EXAMPLE",
        "Version": 2,
        "LastModifiedDate": 1582155479.205,
        "ARN": "arn:aws:ssm:us-east-2:111222333444:parameter/
MySecureStringParameter"
        "DataType": "text"
    }
}
```

Para obtener más información, consulte Uso de Parameter Store en la Guía del usuario de AWS Systems Manager.

Ejemplo 3: visualización del valor de un parámetro mediante etiquetas

En el siguiente ejemplo de `get-parameter` se enumera el valor del parámetro único especificado con una etiqueta especificada.

```
aws ssm get-parameter \
    --name "MyParameter:label"
```

Salida:

```
{
    "Parameter": {
        "Name": "MyParameter",
        "Type": "String",
        "Value": "parameter version 2",
        "Version": 2,
        "Selector": ":label",
        "LastModifiedDate": "2021-07-12T09:49:15.865000-07:00",
        "ARN": "arn:aws:ssm:us-west-2:786973925828:parameter/MyParameter",
        "DataType": "text"
    }
}
```

Para obtener más información, consulte Trabajo con etiquetas de parámetros en la Guía del usuario de AWS Systems Manager.

Ejemplo 4: visualización del valor de un parámetro mediante etiquetas

En el siguiente ejemplo de `get-parameter` se enumera el valor de la versión de parámetro único especificado.

```
aws ssm get-parameter \
    --name "MyParameter:2"
```

Salida:

```
{
    "Parameter": {
        "Name": "MyParameter",
        "Type": "String",
        "Value": "parameter version 2",
        "Version": 2,
        "Selector": ":2",
```

```
        "LastModifiedDate": "2021-07-12T09:49:15.865000-07:00",
        "ARN": "arn:aws:ssm:us-west-2:786973925828:parameter/MyParameter",
        "DataType": "text"
    }
}
```

Para obtener más información, consulte Trabajo con etiquetas de parámetros en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte GetParameter en la Referencia de comandos de la AWS CLI.

Rust

SDK para Rust

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
    pub async fn list_path(&self, path: &str) -> Result<Vec<Parameter>, EC2Error>
{
        let maybe_params: Vec<Result<Parameter, _>> = TryFlatMap::new(
            self.inner
                .get_parameters_by_path()
                .path(path)
                .into_paginator()
                .send(),
        )
        .flat_map(|item| item.parameters.unwrap_or_default())
        .collect()
        .await;
        // Fail on the first error
        let params = maybe_params
            .into_iter()
            .collect::<Result<Vec<Parameter>, _>>()?;
        Ok(params)
    }
```

- Para obtener información acerca de la API, consulte GetParameter en la Referencia de la API del SDK de AWS para Rust.

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`GetParameterHistory`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `GetParameterHistory`.

CLI

AWS CLI

Obtención del historial de valores de un parámetro

En el siguiente ejemplo de `get-parameter-history` se enumera el historial de cambios del parámetro especificado, incluido su valor.

```
aws ssm get-parameter-history \
    --name "MyStringParameter"
```

Salida:

```
{
    "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": []
        }
    ]
}
```

Para obtener más información, consulte Trabajo con versiones de parámetros en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte GetParameterHistory en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumera el historial de valores de un parámetro.

```
Get-SSMParameterHistory -Name "Welcome"
```

Salida:

```
Description      :
```

```
KeyId            :
LastModifiedDate : 3/3/2017 6:55:25 PM
LastModifiedUser : arn:aws:iam::123456789012:user/admin
Name             : Welcome
Type             : String
Value            : helloWorld
```

- Para obtener información sobre la API, consulte GetParameterHistory en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumera el historial de valores de un parámetro.

```
Get-SSMParameterHistory -Name "Welcome"
```

Salida:

```
Description      :
KeyId            :
LastModifiedDate : 3/3/2017 6:55:25 PM
LastModifiedUser : arn:aws:iam::123456789012:user/admin
Name             : Welcome
Type             : String
Value            : helloWorld
```

- Para obtener información sobre la API, consulte GetParameterHistory en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`GetParameters`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `GetParameters`.

CLI

AWS CLI

Ejemplo 1: enumeración de los valores de un parámetro

En el siguiente ejemplo de `get-parameters` se enumeran los valores de los tres parámetros especificados.

```
aws ssm get-parameters \
    --names "MyStringParameter" "MyStringListParameter" "MyInvalidParameterName"
```

Salida:

```
{
    "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"
    ]
}
```

Para obtener más información, consulte Uso de Parameter Store en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: enumeración de los nombres y valores de varios parámetros mediante la opción “--query”

En el siguiente ejemplo de `get-parameters` se enumeran los nombres y valores de los parámetros especificados.

```
aws ssm get-parameters \
    --names MyStringParameter MyStringListParameter \
    --query "Parameters[*].{Name:Name,Value:Value}"
```

Salida:

```
[
    {
        "Name": "MyStringListParameter",
        "Value": "alpha,beta,gamma"
    },
    {
        "Name": "MyStringParameter",
        "Value": "Vidi"
    }
]
```

Para obtener más información, consulte Uso de Parameter Store en la Guía del usuario de AWS Systems Manager.

Ejemplo 3: visualización del valor de un parámetro mediante etiquetas

En el siguiente ejemplo de `get-parameter` se enumera el valor del parámetro único especificado con una etiqueta especificada.

```
aws ssm get-parameter \
    --name "MyParameter:label"
```

Salida:

```
{
    "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": []
}
```

Para obtener más información, consulte Trabajo con etiquetas de parámetros en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte GetParameters en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran los valores de un parámetro.

```
Get-SSMParameterValue -Name "Welcome"
```

Salida:

```
InvalidParameters Parameters
----------------- ----------
{}                {Welcome}
```

Ejemplo 2: en este ejemplo se enumeran los detalles del valor.

```
(Get-SSMParameterValue -Name "Welcome").Parameters
```

Salida:

```
Name    Type   Value
----    ----   -----
Welcome String Good day, Sunshine!
```

- Para obtener información sobre la API, consulte GetParameters en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran los valores de un parámetro.

```
Get-SSMParameterValue -Name "Welcome"
```

Salida:

```
InvalidParameters Parameters
----------------- ----------
{}                {Welcome}
```

Ejemplo 2: en este ejemplo se enumeran los detalles del valor.

```
(Get-SSMParameterValue -Name "Welcome").Parameters
```

Salida:

```
Name    Type   Value
----    ----   -----
Welcome String Good day, Sunshine!
```

- Para obtener información sobre la API, consulte GetParameters en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`GetPatchBaseline`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `GetPatchBaseline`.

CLI

AWS CLI

Visualización de una línea de base de revisiones

En el siguiente ejemplo de `get-patch-baseline` se recuperan los detalles de la línea de base de revisiones especificada.

```
aws ssm get-patch-baseline \
    --baseline-id "pb-0123456789abcdef0"
```

Salida:

```
{
    "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": []
}
```

Para obtener más información, consulte Acerca de las líneas de base de revisiones en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte GetPatchBaseline en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se muestran los detalles de una línea de base de revisiones.

```
Get-SSMPatchBaselineDetail -BaselineId "pb-03da896ca3b68b639"
```

Salida:

```
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 : {}
```

- Para obtener información sobre la API, consulte GetPatchBaseline en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se muestran los detalles de una línea de base de revisiones.

```
Get-SSMPatchBaselineDetail -BaselineId "pb-03da896ca3b68b639"
```

Salida:

```
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 : {}
```

- Para obtener información sobre la API, consulte GetPatchBaseline en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`GetPatchBaselineForPatchGroup`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `GetPatchBaselineForPatchGroup`.

CLI

AWS CLI

Visualización de la línea de base de revisiones de un grupo de revisiones

En el siguiente ejemplo de `get-patch-baseline-for-patch-group` se recuperan los detalles sobre la línea de base de revisiones del grupo de revisiones especificado.

```
aws ssm get-patch-baseline-for-patch-group \
```

```
    --patch-group "DEV"
```

Salida:

```
{
    "PatchGroup": "DEV",
    "BaselineId": "pb-0123456789abcdef0",
    "OperatingSystem": "WINDOWS"
}
```

Para obtener más información, consulte Creación de un grupo de revisiones <https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-group-tagging.html>__ y Añadir un grupo de revisiones a una línea de base de revisiones en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte GetPatchBaselineForPatchGroup en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se muestra la línea de base de revisiones para un grupo de revisiones.

```
Get-SSMPatchBaselineForPatchGroup -PatchGroup "Production"
```

Salida:

```
BaselineId           PatchGroup
----------           ----------
pb-045f10b4f382baeda Production
```

- Para obtener información sobre la API, consulte GetPatchBaselineForPatchGroup en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se muestra la línea de base de revisiones para un grupo de revisiones.

```
Get-SSMPatchBaselineForPatchGroup -PatchGroup "Production"
```

Salida:

```
BaselineId           PatchGroup
----------           ----------
pb-045f10b4f382baeda Production
```

- Para obtener información sobre la API, consulte GetPatchBaselineForPatchGroup en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`ListAssociationVersions`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `ListAssociationVersions`.

CLI

AWS CLI

Enumeración de todas las versiones de una asociación para un ID de asociación específico

En el siguiente ejemplo de `list-association-versions` se enumeran todas las versiones de las asociaciones especificadas.

```
aws ssm list-association-versions \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Salida:

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

Para obtener más información, consulte Trabajo con asociaciones en Systems Manager en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte ListAssociationVersions en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se recuperan todas las versiones de la asociación proporcionada.

```
Get-SSMAssociationVersionList -AssociationId 123a45a0-c678-9012-3456-78901234db5e
```

Salida:

```
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}
```

- Para obtener información sobre la API, consulte ListAssociationVersions en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se recuperan todas las versiones de la asociación proporcionada.

```
Get-SSMAssociationVersionList -AssociationId 123a45a0-c678-9012-3456-78901234db5e
```

Salida:

```
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}
```

- Para obtener información sobre la API, consulte ListAssociationVersions en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`ListAssociations`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `ListAssociations`.

CLI

AWS CLI

Ejemplo 1: enumeración de las asociaciones de una instancia específica

En el siguiente ejemplo de list-associations se enumeran todas las asociaciones con el valor UpdateSSMAgent para AssociationName.

```
aws ssm list-associations /
```

```
    --association-filter-list "key=AssociationName,value=UpdateSSMAgent"
```

Salida:

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

Para obtener más información, consulte Trabajo con asociaciones en Systems Manager en la Guía del usuario de Systems Manager.

Ejemplo 2: enumeración de las asociaciones de un documento específico

En el siguiente ejemplo de list-associations se enumeran todas las asociaciones del documento especificado.

```
aws ssm list-associations /
    --association-filter-list "key=Name,value=AWS-UpdateSSMAgent"
```

Salida:

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

Para obtener más información, consulte Trabajo con asociaciones en Systems Manager en la Guía del usuario de Systems Manager.

- Para obtener información sobre la API, consulte ListAssociations en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran todas las asociaciones de una instancia. La sintaxis utilizada en este ejemplo requiere la versión 3 o posterior de PowerShell.

```
$filter1 = @{Key="InstanceId";Value=@("i-0000293ffd8c57862")}
Get-SSMAssociationList -AssociationFilterList $filter1
```

Salida:

```
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}
```

Ejemplo 2: en este ejemplo se enumeran todas las asociaciones de un documento de configuración. La sintaxis utilizada en este ejemplo requiere la versión 3 o posterior de PowerShell.

```
$filter2 = @{Key="Name";Value=@("AWS-UpdateSSMAgent")}
Get-SSMAssociationList -AssociationFilterList $filter2
```

Salida:

```
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}
```

Ejemplo 3: con la versión 2 de PowerShell, debe usar New-Object para crear cada filtro.

```
$filter1 = New-Object Amazon.SimpleSystemsManagement.Model.AssociationFilter
$filter1.Key = "InstanceId"
$filter1.Value = "i-0000293ffd8c57862"

Get-SSMAssociationList -AssociationFilterList $filter1
```

Salida:

```
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}
```

- Para obtener información sobre la API, consulte ListAssociations en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran todas las asociaciones de una instancia. La sintaxis utilizada en este ejemplo requiere la versión 3 o posterior de PowerShell.

```
$filter1 = @{Key="InstanceId";Value=@("i-0000293ffd8c57862")}
Get-SSMAssociationList -AssociationFilterList $filter1
```

Salida:

```
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}
```

Ejemplo 2: en este ejemplo se enumeran todas las asociaciones de un documento de configuración. La sintaxis utilizada en este ejemplo requiere la versión 3 o posterior de PowerShell.

```
$filter2 = @{Key="Name";Value=@("AWS-UpdateSSMAgent")}
Get-SSMAssociationList -AssociationFilterList $filter2
```

Salida:

```
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}
```

Ejemplo 3: con la versión 2 de PowerShell, debe usar New-Object para crear cada filtro.

```
$filter1 = New-Object Amazon.SimpleSystemsManagement.Model.AssociationFilter
$filter1.Key = "InstanceId"
$filter1.Value = "i-0000293ffd8c57862"

Get-SSMAssociationList -AssociationFilterList $filter1
```

Salida:

```
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}
```

- Para obtener información sobre la API, consulte ListAssociations en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Uso de **`ListCommandInvocations`** con un SDK de AWS o la CLI

Los siguientes ejemplos de código muestran cómo utilizar `ListCommandInvocations`.

Los ejemplos de acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Puede ver esta acción en contexto en el siguiente ejemplo de código:

- Conceptos básicos

CLI

AWS CLI

Enumeración de las invocaciones de un comando específico

En el siguiente ejemplo de `list-command-invocations` se enumeran todas las invocaciones de un comando.

```
aws ssm list-command-invocations \
    --command-id "ef7fdfd8-9b57-4151-a15c-db9a12345678" \
    --details
```

Salida:

```
{
    "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
```

```
            }
        }
    ]
}
```

Para obtener más información, consulte Descripción de los estados del comando en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte ListCommandInvocations en la Referencia de comandos de la AWS CLI.

JavaScript

SDK para JavaScript (v3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
import { paginateListCommandInvocations, SSMClient } from "@aws-sdk/client-ssm";
import { parseArgs } from "node:util";

/**
 * List SSM command invocations on an instance.
 * @param {{ instanceId: string }}
 */
export const main = async ({ instanceId }) => {
  const client = new SSMClient({});
  try {
    const listCommandInvocationsPaginated = [];
    // The paginate function is a wrapper around the base command.
    const paginator = paginateListCommandInvocations(
      { client },
      {
        InstanceId: instanceId,
      },
    );
    for await (const page of paginator) {
      listCommandInvocationsPaginated.push(...page.CommandInvocations);
```

```
    }
    console.log("Here is the list of command invocations:");
    console.log(listCommandInvocationsPaginated);
    return { CommandInvocations: listCommandInvocationsPaginated };
  } catch (caught) {
    if (caught instanceof Error && caught.name === "ValidationError") {
      console.warn(`${caught.message}. Did you provide a valid instance ID?`);
    }
    throw caught;
  }
};
```

- Para obtener detalles sobre la API, consulte ListCommandInvocations en la Referencia de la API deAWS SDK para JavaScript.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran todas las invocaciones de un comando.

```
Get-SSMCommandInvocation -CommandId "b8eac879-0541-439d-94ec-47a80d554f44" -
Detail $true
```

Salida:

```
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        :
```

Ejemplo 2: en este ejemplo se enumeran los CommandPlugins para la invocación del comando con el identificador e1eb2e3c-ed4c-5123-45c1-234f5612345f

```
Get-SSMCommandInvocation -CommandId e1eb2e3c-ed4c-5123-45c1-234f5612345f -Detail:
$true | Select-Object -ExpandProperty CommandPlugins
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte ListCommandInvocations en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran todas las invocaciones de un comando.

```
Get-SSMCommandInvocation -CommandId "b8eac879-0541-439d-94ec-47a80d554f44" -
Detail $true
```

Salida:

```
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        :
```

Ejemplo 2: en este ejemplo se enumeran los CommandPlugins para la invocación del comando con el identificador e1eb2e3c-ed4c-5123-45c1-234f5612345f

```
Get-SSMCommandInvocation -CommandId e1eb2e3c-ed4c-5123-45c1-234f5612345f -Detail:
$true | Select-Object -ExpandProperty CommandPlugins
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte ListCommandInvocations en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Python

SDK para Python (Boto3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
class DocumentWrapper:
    """Encapsulates AWS Systems Manager Document actions."""

    def __init__(self, ssm_client):
        """
        :param ssm_client: A Boto3 Systems Manager client.
        """
        self.ssm_client = ssm_client
        self.name = None

    @classmethod
    def from_client(cls):
        ssm_client = boto3.client("ssm")
        return cls(ssm_client)


    def list_command_invocations(self, instance_id):
        """
        Lists the commands for an instance.

        :param instance_id: The ID of the instance.
        :return: The list of commands.
        """
        try:
            paginator = self.ssm_client.get_paginator("list_command_invocations")
            command_invocations = []
            for page in paginator.paginate(InstanceId=instance_id):
                command_invocations.extend(page["CommandInvocations"])
            num_of_commands = len(command_invocations)
            print(
                f"{num_of_commands} command invocation(s) found for instance
 {instance_id}."
```

```
            )

            if num_of_commands > 10:
                print("Displaying the first 10 commands:")
                num_of_commands = 10
            date_format = "%A, %d %B %Y %I:%M%p"
            for command in command_invocations[:num_of_commands]:
                print(
                    f"   The time of command invocation is
{command['RequestedDateTime'].strftime(date_format)}"
                )
        except ClientError as err:
            logger.error(
                "Couldn't list commands for %s. Here's why: %s: %s",
                instance_id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
```

- Para obtener información sobre la API, consulte ListCommandInvocations en la Referencia de la API del AWSSDK para Python (Boto3).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`ListCommands`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `ListCommands`.

CLI

AWS CLI

Ejemplo 1: obtención del estado de un comando específico

En el siguiente ejemplo de `list-commands` se recupera y muestra el estado del comando especificado.

```
aws ssm list-commands \
    --command-id "0831e1a8-a1ac-4257-a1fd-c831bEXAMPLE"
```

Ejemplo 2: obtención del estado de los comandos solicitados después de una fecha específica

En el siguiente ejemplo de `list-commands` se recuperan los detalles de los comandos solicitados después de la fecha especificada.

```
aws ssm list-commands \
    --filter "key=InvokedAfter,value=2020-02-01T00:00:00Z"
```

Ejemplo 3: enumeración de todos los comandos solicitados en una cuenta de AWS

En el siguiente ejemplo de `list-commands` se enumeran todos los comandos que han solicitado los usuarios de la cuenta y la región de AWS actuales.

```
aws ssm list-commands
```

Salida:

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

Para obtener más información, consulte Running Commands Using Systems Manager Run Command en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte ListCommands en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran todos los comandos solicitados.

```
Get-SSMCommand
```

Salida:

```
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            : {}
```

Ejemplo 2: en este ejemplo se obtiene el estado de un comando específico.

```
Get-SSMCommand -CommandId "4b75a163-d39a-4d97-87c9-98ae52c6be35"
```

Ejemplo 3: en este ejemplo se recuperan todos los comandos de SSM invocados después de 2019-04-01T00:00:00Z

```
Get-SSMCommand -Filter @{Key="InvokedAfter";Value="2019-04-01T00:00:00Z"} |
 Select-Object CommandId, DocumentName, Status, RequestedDateTime | Sort-Object -
Property RequestedDateTime -Descending
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte ListCommands en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran todos los comandos solicitados.

```
Get-SSMCommand
```

Salida:

```
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            : {}
```

Ejemplo 2: en este ejemplo se obtiene el estado de un comando específico.

```
Get-SSMCommand -CommandId "4b75a163-d39a-4d97-87c9-98ae52c6be35"
```

Ejemplo 3: en este ejemplo se recuperan todos los comandos de SSM invocados después de 2019-04-01T00:00:00Z

```
Get-SSMCommand -Filter @{Key="InvokedAfter";Value="2019-04-01T00:00:00Z"} |
 Select-Object CommandId, DocumentName, Status, RequestedDateTime | Sort-Object -
Property RequestedDateTime -Descending
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte ListCommands en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`ListComplianceItems`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `ListComplianceItems`.

CLI

AWS CLI

Enumeración de los elementos de conformidad de una instancia específica

En este ejemplo se enumeran todos los elementos de conformidad de la instancia especificada.

Comando:

```
aws ssm list-compliance-items --resource-ids "i-1234567890abcdef0" --resource-
types "ManagedInstance"
```

Salida:

```
{
  "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--"
}
```

Enumeración de los elementos de conformidad de una instancia y un ID de asociación específicos

En este ejemplo se enumeran todos los elementos de conformidad de la instancia y el ID de asociación especificados.

Comando:

```
aws ssm list-compliance-items --resource-ids "i-1234567890abcdef0" --resource-
types "ManagedInstance" --
filters "Key=ComplianceType,Values=Association,Type=EQUAL" "Key=Id,Values=e4c2ed6d-516f-4
aa2a-0123456789ab,Type=EQUAL"
```

Enumeración de los elementos de conformidad de una instancia después de una fecha y hora específicas

En este ejemplo se enumeran todos los elementos de conformidad de una instancia después de la fecha y hora especificada.

Comando:

```
aws ssm list-compliance-items --resource-ids "i-1234567890abcdef0" --resource-
types "ManagedInstance" --
filters "Key=ExecutionTime,Values=2019-02-18T16:00:00Z,Type=GREATER_THAN"
```

- Para obtener información sobre la API, consulte ListComplianceItems en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran los elementos de conformidad para el identificador y el tipo de recurso indicados, filtrando el tipo de conformidad por “Asociación”

```
Get-SSMComplianceItemList -ResourceId i-1a2caf345f67d0dc2 -ResourceType
 ManagedInstance -Filter @{Key="ComplianceType";Values="Association"}
```

Salida:

```
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            :
```

- Para obtener información sobre la API, consulte ListComplianceItems en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran los elementos de conformidad para el identificador y el tipo de recurso indicados, filtrando el tipo de conformidad por “Asociación”

```
Get-SSMComplianceItemList -ResourceId i-1a2caf345f67d0dc2 -ResourceType
 ManagedInstance -Filter @{Key="ComplianceType";Values="Association"}
```

Salida:

```
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            :
```

- Para obtener información sobre la API, consulte ListComplianceItems en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`ListComplianceSummaries`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `ListComplianceSummaries`.

CLI

AWS CLI

Enumeración de los resúmenes de conformidad de todos los tipos de conformidad

En este ejemplo se enumeran los resúmenes de conformidad de todos los tipos de conformidad de su cuenta.

Comando:

```
aws ssm list-compliance-summaries
```

Salida:

```
{
  "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=="
}
```

Enumeración de los resúmenes de conformidad de un tipo de conformidad específico

En este ejemplo se muestra el resumen de conformidad del tipo de conformidad de la revisión.

Comando:

```
aws ssm list-compliance-summaries --
filters "Key=ComplianceType,Values=Patch,Type=EQUAL"
```

- Para obtener información sobre la API, consulte ListComplianceSummaries en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se devuelve un recuento resumido de los recursos conformes y no conformes correspondientes a todos los tipos de conformidad.

```
Get-SSMComplianceSummaryList
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte ListComplianceSummaries en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se devuelve un recuento resumido de los recursos conformes y no conformes correspondientes a todos los tipos de conformidad.

```
Get-SSMComplianceSummaryList
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte ListComplianceSummaries en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`ListDocumentVersions`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `ListDocumentVersions`.

CLI

AWS CLI

Enumeración de las versiones de los documentos

En el siguiente ejemplo de `list-document-versions` se enumeran todas las versiones de un documento de Systems Manager.

```
aws ssm list-document-versions \
    --name "Example"
```

Salida:

```
{
    "DocumentVersions": [
        {
            "Name": "Example",
            "DocumentVersion": "1",
            "CreatedDate": 1583257938.266,
            "IsDefaultVersion": true,
            "DocumentFormat": "YAML",
            "Status": "Active"
        }
    ]
}
```

Para obtener más información, consulte Ejecución de comandos mediante una versión de documento específica en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte ListDocumentVersions en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran todas las versiones de un documento.

```
Get-SSMDocumentVersionList -Name "AWS-UpdateSSMAgent"
```

Salida:

```
CreatedDate       : 6/1/2021 5:19:10 PM
DocumentFormat    : JSON
DocumentVersion   : 1
IsDefaultVersion  : True
Name              : AWS-UpdateSSMAgent
Status            : Active
```

- Para obtener información sobre la API, consulte ListDocumentVersions en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran todas las versiones de un documento.

```
Get-SSMDocumentVersionList -Name "AWS-UpdateSSMAgent"
```

Salida:

```
CreatedDate       : 6/1/2021 5:19:10 PM
DocumentFormat    : JSON
DocumentVersion   : 1
IsDefaultVersion  : True
Name              : AWS-UpdateSSMAgent
Status            : Active
```

- Para obtener información sobre la API, consulte ListDocumentVersions en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`ListDocuments`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `ListDocuments`.

CLI

AWS CLI

Ejemplo 1: enumeración de documentos

En el siguiente ejemplo de `list-documents` se enumeran los documentos propiedad de la cuenta solicitante etiquetados con la etiqueta personalizada.

```
aws ssm list-documents \
    --filters Key=Owner,Values=Self Key=tag:DocUse,Values=Testing
```

Salida:

```
{
    "DocumentIdentifiers": [
        {
            "Name": "Example",
            "Owner": "29884EXAMPLE",
            "PlatformTypes": [
                "Windows",
                "Linux"
            ],
            "DocumentVersion": "1",
            "DocumentType": "Automation",
            "SchemaVersion": "0.3",
            "DocumentFormat": "YAML",
            "Tags": [
                {
                    "Key": "DocUse",
                    "Value": "Testing"
                }
            ]
        }
    ]
}
```

Para obtener más información, consulte Documentos de AWS Systems Manager en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: enumeración de los documentos compartidos

En el siguiente ejemplo de `list-documents` se enumeran los documentos compartidos, incluidos los documentos compartidos privados que no son propiedad de AWS.

```
aws ssm list-documents \
    --filters Key=Name,Values=sharedDocNamePrefix  Key=Owner,Values=Private
```

Salida:

```
{
    "DocumentIdentifiers": [
        {
            "Name": "Example",
            "Owner": "12345EXAMPLE",
            "PlatformTypes": [
                "Windows",
                "Linux"
            ],
            "DocumentVersion": "1",
            "DocumentType": "Command",
            "SchemaVersion": "0.3",
            "DocumentFormat": "YAML",
            "Tags": []
        }
    ]
}
```

Para obtener más información, consulte Documentos de AWS Systems Manager en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte ListDocuments en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se muestran todos los documentos de configuración de su cuenta.

```
Get-SSMDocumentList
```

Salida:

```
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
...
```

Ejemplo 2: en este ejemplo se recuperan todos los documentos de automatización cuyo nombre coincida con “Platform”

```
Get-SSMDocumentList -DocumentFilterList @{Key="DocumentType";Value="Automation"}
 | Where-Object Name -Match "Platform"
```

Salida:

```
DocumentFormat  : JSON
DocumentType    : Automation
DocumentVersion : 7
Name            : KT-Get-Platform
Owner           : 987654123456
PlatformTypes   : {Windows, Linux}
SchemaVersion   : 0.3
Tags            : {}
TargetType      :
VersionName     :
```

- Para obtener información sobre la API, consulte ListDocuments en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se muestran todos los documentos de configuración de su cuenta.

```
Get-SSMDocumentList
```

Salida:

```
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
...
```

Ejemplo 2: en este ejemplo se recuperan todos los documentos de automatización cuyo nombre coincida con “Platform”

```
Get-SSMDocumentList -DocumentFilterList @{Key="DocumentType";Value="Automation"}
 | Where-Object Name -Match "Platform"
```

Salida:

```
DocumentFormat  : JSON
```

```
DocumentType    : Automation
DocumentVersion : 7
Name            : KT-Get-Platform
Owner           : 987654123456
PlatformTypes   : {Windows, Linux}
SchemaVersion   : 0.3
Tags            : {}
TargetType      :
VersionName     :
```

- Para obtener información sobre la API, consulte ListDocuments en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`ListInventoryEntries`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `ListInventoryEntries`.

CLI

AWS CLI

Ejemplo 1: visualización de las entradas de tipos de inventario específicos de una instancia

En el siguiente ejemplo de `list-inventory-entries` se enumeran las entradas de inventario del tipo de inventario AWS:Application en una instancia específica.

```
aws ssm list-inventory-entries \
    --instance-id "i-1234567890abcdef0" \
    --type-name "AWS:Application"
```

Salida:

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

Ejemplo 2: visualización de las entradas de inventario personalizadas asignadas a una instancia

En el siguiente ejemplo de `list-inventory-entries` se muestra una entrada de inventario personalizada asignada a una instancia.

```
aws ssm list-inventory-entries \
    --instance-id "i-1234567890abcdef0" \
    --type-name "Custom:RackInfo"
```

Salida:

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

- Para obtener información sobre la API, consulte ListInventoryEntries en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran todas las entradas de inventario personalizadas de una instancia.

```
Get-SSMInventoryEntriesList -InstanceId "i-0cb2b964d3e14fd9f" -TypeName
 "Custom:RackInfo"
```

Salida:

```
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
```

Ejemplo 2: en este ejemplo se enumeran los detalles.

```
(Get-SSMInventoryEntriesList -InstanceId "i-0cb2b964d3e14fd9f" -TypeName
 "Custom:RackInfo").Entries
```

Salida:

```
Key          Value
---          -----
RackLocation Bay B/Row C/Rack D/Shelf E
```

- Para obtener información sobre la API, consulte ListInventoryEntries en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran todas las entradas de inventario personalizadas de una instancia.

```
Get-SSMInventoryEntriesList -InstanceId "i-0cb2b964d3e14fd9f" -TypeName
 "Custom:RackInfo"
```

Salida:

```
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
```

Ejemplo 2: en este ejemplo se enumeran los detalles.

```
(Get-SSMInventoryEntriesList -InstanceId "i-0cb2b964d3e14fd9f" -TypeName
 "Custom:RackInfo").Entries
```

Salida:

```
Key          Value
---          -----
RackLocation Bay B/Row C/Rack D/Shelf E
```

- Para obtener información sobre la API, consulte ListInventoryEntries en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`ListResourceComplianceSummaries`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `ListResourceComplianceSummaries`.

CLI

AWS CLI

Cómo enumerar los recuentos resumidos de conformidad para cada recurso

En este ejemplo se enumeran los recuentos resumidos de conformidad para cada recurso.

Comando:

```
aws ssm list-resource-compliance-summaries
```

Salida:

```
{
  "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--"
}
```

Cómo enumerar los resúmenes de conformidad de un tipo de conformidad específico para cada recurso

En este ejemplo se enumeran los resúmenes de conformidad de cada recurso del tipo de conformidad de la revisión.

Comando:

```
aws ssm list-resource-compliance-summaries --
filters "Key=ComplianceType,Values=Patch,Type=EQUAL"
```

- Para obtener información sobre la API, consulte ListResourceComplianceSummaries en la Referencia del comando de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se obtiene un recuento resumido para cada recurso. En el resumen se incluye información sobre los estados conformes y no conformes y recuentos detallados de la gravedad de los elementos de conformidad de los productos que coinciden con “Windows10”. Como el valor predeterminado de MaxResult es 100 si no se especifica el parámetro y este valor no es válido, se agrega el parámetro MaxResult y el valor se establece en 50.

```
$FilterValues = @{
  "Key"="Product"
        "Type"="EQUAL"
        "Values"="Windows10"
}
        Get-SSMResourceComplianceSummaryList -Filter $FilterValues -MaxResult 50
```

- Para obtener información sobre la API, consulte ListResourceComplianceSummaries en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se obtiene un recuento resumido para cada recurso. En el resumen se incluye información sobre los estados conformes y no conformes y recuentos detallados de la gravedad de los elementos de conformidad de los productos que coinciden con “Windows10”. Como el valor predeterminado de MaxResult es 100 si no se especifica el parámetro y este valor no es válido, se agrega el parámetro MaxResult y el valor se establece en 50.

```
$FilterValues = @{
  "Key"="Product"
```

```
        "Type"="EQUAL"
        "Values"="Windows10"
}
        Get-SSMResourceComplianceSummaryList -Filter $FilterValues -MaxResult 50
```

- Para obtener información sobre la API, consulte ListResourceComplianceSummaries en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`ListTagsForResource`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `ListTagsForResource`.

CLI

AWS CLI

Cómo enumerar las etiquetas aplicadas a una línea de base de revisiones

En el siguiente ejemplo de `list-tags-for-resource` se muestran las etiquetas de una línea de base de revisiones.

```
aws ssm list-tags-for-resource \
    --resource-type "PatchBaseline" \
    --resource-id "pb-0123456789abcdef0"
```

Salida:

```
{
    "TagList": [
        {
            "Key": "Environment",
            "Value": "Production"
        },
        {
            "Key": "Region",
            "Value": "EMEA"
        }
    ]
```

```
}
```

Para obtener más información, consulte Tagging AWS Resources en la Referencia general de AWS.

- Para ver los detalles de la API, consulte ListTagsForResource en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se enumeran las etiquetas de un periodo de mantenimiento.

```
Get-SSMResourceTag -ResourceId "mw-03eb9db42890fb82d" -ResourceType
 "MaintenanceWindow"
```

Salida:

```
Key   Value
---   -----
Stack Production
```

- Para obtener información sobre la API, consulte ListTagsForResource en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se enumeran las etiquetas de un periodo de mantenimiento.

```
Get-SSMResourceTag -ResourceId "mw-03eb9db42890fb82d" -ResourceType
 "MaintenanceWindow"
```

Salida:

```
Key   Value
---   -----
Stack Production
```

- Para obtener información sobre la API, consulte ListTagsForResource en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`ModifyDocumentPermission`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `ModifyDocumentPermission`.

CLI

AWS CLI

Modificación de los permisos de los documentos

En el siguiente ejemplo de `modify-document-permission` se comparte un documento de Systems Manager públicamente.

```
aws ssm modify-document-permission \
    --name "Example" \
    --permission-type "Share" \
    --account-ids-to-add "All"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Compartir un documento de Systems Manager en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte ModifyDocumentPermission en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se agregan permisos para “compartir” a todas las cuentas de un documento. No se obtienen resultados si el comando se ejecuta correctamente.

```
Edit-SSMDocumentPermission -Name "RunShellScript" -PermissionType "Share" -
AccountIdsToAdd all
```

Ejemplo 2: en este ejemplo se agregan permisos para “compartir” a una cuenta específica de un documento. No se obtienen resultados si el comando se ejecuta correctamente.

```
Edit-SSMDocumentPermission -Name "RunShellScriptNew" -PermissionType "Share" -
AccountIdsToAdd "123456789012"
```

- Para obtener información sobre la API, consulte ModifyDocumentPermission en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se agregan permisos para “compartir” a todas las cuentas de un documento. No se obtienen resultados si el comando se ejecuta correctamente.

```
Edit-SSMDocumentPermission -Name "RunShellScript" -PermissionType "Share" -
AccountIdsToAdd all
```

Ejemplo 2: en este ejemplo se agregan permisos para “compartir” a una cuenta específica de un documento. No se obtienen resultados si el comando se ejecuta correctamente.

```
Edit-SSMDocumentPermission -Name "RunShellScriptNew" -PermissionType "Share" -
AccountIdsToAdd "123456789012"
```

- Para obtener información sobre la API, consulte ModifyDocumentPermission en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`PutComplianceItems`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `PutComplianceItems`.

CLI

AWS CLI

Registro de un tipo de conformidad y de los detalles de conformidad en una instancia designada

En este ejemplo se registra el tipo de conformidad `Custom:AVCheck` en la instancia administrada especificada. No se obtienen resultados si el comando se ejecuta correctamente.

Comando:

```
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"
```

- Para obtener información sobre la API, consulte PutComplianceItems en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se escribe un elemento de conformidad personalizado para la instancia administrada especificada

```
$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"
```

- Para obtener información sobre la API, consulte PutComplianceItems en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se escribe un elemento de conformidad personalizado para la instancia administrada especificada

```
$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"
```

- Para obtener información sobre la API, consulte PutComplianceItems en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`PutInventory`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `PutInventory`.

CLI

AWS CLI

Cómo asignar metadatos de cliente a una instancia

En este ejemplo se asigna información de ubicación de bastidores a una instancia. No se obtienen resultados si el comando se ejecuta correctamente.

Comando (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"}]}]'
```

Comando (Windows):

```
aws ssm put-inventory --instance-id "i-016648b75dd622dab" --
items "TypeName=Custom:RackInfo,SchemaVersion=1.0,CaptureTime=2019-01-22T10:01:01Z,Conten
 B/Row C/Rack D/Shelf F'}]"
```

- Para obtener información sobre la API, consulte PutInventory en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se asigna información de ubicación de bastidores a una instancia. No se obtienen resultados si el comando se ejecuta correctamente.

```
$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
```

- Para obtener información sobre la API, consulte PutInventory en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se asigna información de ubicación de bastidores a una instancia. No se obtienen resultados si el comando se ejecuta correctamente.

```
$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
```

- Para obtener información sobre la API, consulte PutInventory en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Uso de **`PutParameter`** con un SDK de AWS o la CLI

Los siguientes ejemplos de código muestran cómo utilizar `PutParameter`.

CLI

AWS CLI

Ejemplo 1: Cambio del valor de un parámetro

En el siguiente ejemplo de `put-parameter` se cambia el valor del parámetro especificado.

```
aws ssm put-parameter \
    --name "MyStringParameter" \
    --type "String" \
    --value "Vici" \
    --overwrite
```

Salida:

```
{
    "Version": 2,
    "Tier": "Standard"
```

```
}
```

Para obtener más información, consulte Crear un parámetro de Systems Manager (CLI de AWS), Administración de niveles de parámetros y Trabajo con políticas de parámetros en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: Creación de un parámetro avanzado

En el siguiente ejemplo de `put-parameter` se crea un parámetro avanzado.

```
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
```

Salida:

```
{
    "Version": 1,
    "Tier": "Advanced"
}
```

Para obtener más información, consulte Crear un parámetro de Systems Manager (CLI de AWS), Administración de niveles de parámetros y Trabajo con políticas de parámetros en la Guía del usuario de AWS Systems Manager.

Ejemplo 3: cambio de un parámetro estándar a un parámetro avanzado

En el siguiente ejemplo de `put-parameter`, se convierte un parámetro estándar existente en un parámetro avanzado.

```
aws ssm put-parameter \
    --name "MyConvertedParameter" \
    --value "abc123" \
    --type "String" \
    --tier Advanced \
    --overwrite
```

Salida:

```
{
    "Version": 2,
    "Tier": "Advanced"
}
```

Para obtener más información, consulte Crear un parámetro de Systems Manager (CLI de AWS), Administración de niveles de parámetros y Trabajo con políticas de parámetros en la Guía del usuario de AWS Systems Manager.

Ejemplo 4: creación de un parámetro con una política adjunta

En el siguiente ejemplo de `put-parameter` se crea un parámetro avanzado con una política de parámetros adjunta.

```
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\"}}]"
```

Salida:

```
{
    "Version": 1,
    "Tier": "Advanced"
}
```

Para obtener más información, consulte Crear un parámetro de Systems Manager (CLI de AWS), Administración de niveles de parámetros y Trabajo con políticas de parámetros en la Guía del usuario de AWS Systems Manager.

Ejemplo 5: adición de una política a un parámetro existente

En el siguiente ejemplo de `put-parameter` se asocia una política a un parámetro avanzado existente.

```
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
```

Salida:

```
{
    "Version": 2,
    "Tier": "Advanced"
}
```

Para obtener más información, consulte Crear un parámetro de Systems Manager (CLI de AWS), Administración de niveles de parámetros y Trabajo con políticas de parámetros en la Guía del usuario de AWS Systems Manager.

- Para obtener información acerca de la API, consulte PutParameter en la Referencia de comandos de la AWS CLI.

Java

SDK para Java 2.x

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de 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);
        }
    }
}
```

- Para obtener información sobre la API, consulte PutParameter en la Referencia de la API de AWS SDK for Java 2.x.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se crea un parámetro. No se obtienen resultados si el comando se ejecuta correctamente.

```
Write-SSMParameter -Name "Welcome" -Type "String" -Value "helloWorld"
```

Ejemplo 2: en este ejemplo se modifica un parámetro. No se obtienen resultados si el comando se ejecuta correctamente.

```
Write-SSMParameter -Name "Welcome" -Type "String" -Value "Good day, Sunshine!" -
Overwrite $true
```

- Para obtener información sobre la API, consulte PutParameter en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se crea un parámetro. No se obtienen resultados si el comando se ejecuta correctamente.

```
Write-SSMParameter -Name "Welcome" -Type "String" -Value "helloWorld"
```

Ejemplo 2: en este ejemplo se modifica un parámetro. No se obtienen resultados si el comando se ejecuta correctamente.

```
Write-SSMParameter -Name "Welcome" -Type "String" -Value "Good day, Sunshine!" -
Overwrite $true
```

- Para obtener información sobre la API, consulte PutParameter en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Rust

SDK para Rust

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de 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(())
}
```

- Para obtener información sobre la API, consulte PutParameter en la Referencia de la API de AWS SDK para Rust.

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`RegisterDefaultPatchBaseline`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `RegisterDefaultPatchBaseline`.

CLI

AWS CLI

Configuración de la línea de base de revisiones predeterminada

En el siguiente ejemplo de `register-default-patch-baseline` se registra la línea de base de revisiones personalizada especificada como la línea de base de revisiones predeterminada para el tipo de sistema operativo que admite.

```
aws ssm register-default-patch-baseline \
    --baseline-id "pb-abc123cf9bEXAMPLE"
```

Salida:

```
{
    "BaselineId":"pb-abc123cf9bEXAMPLE"
}
```

En el siguiente ejemplo de `register-default-patch-baseline` se registra la línea de base de revisiones predeterminada que ha proporcionado AWS para CentOS como la línea de base de revisiones predeterminada.

```
aws ssm register-default-patch-baseline \
    --baseline-id "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-0574b43a65ea646ed"
```

Salida:

```
{
    "BaselineId":"pb-abc123cf9bEXAMPLE"
}
```

Para obtener más información, consulte Acerca de las líneas de base de revisiones personalizadas y predefinidas en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte RegisterDefaultPatchBaseline en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se registra una línea de base de revisiones como la línea de base de revisiones predeterminada.

```
Register-SSMDefaultPatchBaseline -BaselineId "pb-03da896ca3b68b639"
```

Salida:

```
pb-03da896ca3b68b639
```

- Para obtener información sobre la API, consulte RegisterDefaultPatchBaseline en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se registra una línea de base de revisiones como la línea de base de revisiones predeterminada.

```
Register-SSMDefaultPatchBaseline -BaselineId "pb-03da896ca3b68b639"
```

Salida:

```
pb-03da896ca3b68b639
```

- Para obtener información sobre la API, consulte RegisterDefaultPatchBaseline en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **RegisterPatchBaselineForPatchGroup** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `RegisterPatchBaselineForPatchGroup`.

CLI

AWS CLI

Registro de una línea de base de revisiones de un grupo de revisiones

En el siguiente ejemplo de `register-patch-baseline-for-patch-group` se registra una línea de base de revisiones para un grupo de revisiones.

```
aws ssm register-patch-baseline-for-patch-group \
    --baseline-id "pb-045f10b4f382baeda" \
    --patch-group "Production"
```

Salida:

```
{
    "BaselineId": "pb-045f10b4f382baeda",
    "PatchGroup": "Production"
}
```

Para obtener más información, consulte Creación de un grupo de revisiones <https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-group-tagging.html>__ y Añadir un grupo de revisiones a una línea de base de revisiones en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte RegisterPatchBaselineForPatchGroup en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se registra una línea de base de revisiones para un grupo de revisiones.

```
Register-SSMPatchBaselineForPatchGroup -BaselineId "pb-03da896ca3b68b639" -
PatchGroup "Production"
```

Salida:

```
BaselineId           PatchGroup
----------           ----------
pb-03da896ca3b68b639 Production
```

- Para obtener información sobre la API, consulte RegisterPatchBaselineForPatchGroup en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se registra una línea de base de revisiones para un grupo de revisiones.

```
Register-SSMPatchBaselineForPatchGroup -BaselineId "pb-03da896ca3b68b639" -
PatchGroup "Production"
```

Salida:

```
BaselineId           PatchGroup
----------           ----------
pb-03da896ca3b68b639 Production
```

- Para obtener información sobre la API, consulte RegisterPatchBaselineForPatchGroup en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`RegisterTargetWithMaintenanceWindow`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `RegisterTargetWithMaintenanceWindow`.

CLI

AWS CLI

Ejemplo 1: registro de un único destino con un periodo de mantenimiento

En el siguiente ejemplo de `register-target-with-maintenance-window` se registra una instancia con un periodo de mantenimiento.

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-ab12cd34ef56gh78" \
    --target "Key=InstanceIds,Values=i-0000293ffd8c57862" \
    --owner-information "Single instance" \
    --resource-type "INSTANCE"
```

Salida:

```
{
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Ejemplo 2: registro de varios destinos con un periodo de mantenimiento mediante los ID de instancia

En el siguiente ejemplo de `register-target-with-maintenance-window` se registran dos instancias con un periodo de mantenimiento especificando sus ID de instancia.

```
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"
```

Salida:

```
{
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Ejemplo 3: registro de destinos con un periodo de mantenimiento mediante etiquetas de recursos

En el siguiente ejemplo de `register-target-with-maintenance-window` se registran las instancias con un periodo de mantenimiento especificando las etiquetas de recursos que se han aplicado a las instancias.

```
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"
```

Salida:

```
{
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Ejemplo 4: registro de destinos mediante un grupo de claves de etiquetas

En el siguiente ejemplo de `register-target-with-maintenance-window` se registran instancias que tienen una o más claves asignadas, independientemente de los valores de la clave.

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --resource-type "INSTANCE" \
    --target "Key=tag-key,Values=Name,Instance-Type,CostCenter"
```

Salida:

```
{
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Ejemplo 5: registro de destinos con un nombre de grupo de recursos

En el siguiente ejemplo de `register-target-with-maintenance-window` se registra un grupo de recursos especificado, independientemente del tipo de recursos que contiene.

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
```

```
    --resource-type "RESOURCE_GROUP" \
    --target "Key=resource-groups:Name,Values=MyResourceGroup"
```

Salida:

```
{
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Para obtener más información, consulte Registrar una instancia de destino con el periodo de mantenimiento (AWS CLI) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte RegisterTargetWithMaintenanceWindow en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se registra una instancia con un periodo de mantenimiento.

```
$option1 = @{Key="InstanceIds";Values=@("i-0000293ffd8c57862")}
Register-SSMTargetWithMaintenanceWindow -WindowId "mw-06cf17cbefcb4bf4f" -Target
 $option1 -OwnerInformation "Single instance" -ResourceType "INSTANCE"
```

Salida:

```
d8e47760-23ed-46a5-9f28-927337725398
```

Ejemplo 2: en este ejemplo se registran varias instancias con un periodo de mantenimiento.

```
$option1 =
 @{Key="InstanceIds";Values=@("i-0000293ffd8c57862","i-0cb2b964d3e14fd9f")}
Register-SSMTargetWithMaintenanceWindow -WindowId "mw-06cf17cbefcb4bf4f" -Target
 $option1 -OwnerInformation "Single instance" -ResourceType "INSTANCE"
```

Salida:

```
6ab5c208-9fc4-4697-84b7-b02a6cc25f7d
```

Ejemplo 3: en este ejemplo se registra una instancia con un periodo de mantenimiento mediante etiquetas de EC2.

```
$option1 = @{Key="tag:Environment";Values=@("Production")}
Register-SSMTargetWithMaintenanceWindow -WindowId "mw-06cf17cbefcb4bf4f" -Target
 $option1 -OwnerInformation "Production Web Servers" -ResourceType "INSTANCE"
```

Salida:

```
2994977e-aefb-4a71-beac-df620352f184
```

- Para obtener información sobre la API, consulte RegisterTargetWithMaintenanceWindow en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se registra una instancia con un periodo de mantenimiento.

```
$option1 = @{Key="InstanceIds";Values=@("i-0000293ffd8c57862")}
Register-SSMTargetWithMaintenanceWindow -WindowId "mw-06cf17cbefcb4bf4f" -Target
 $option1 -OwnerInformation "Single instance" -ResourceType "INSTANCE"
```

Salida:

```
d8e47760-23ed-46a5-9f28-927337725398
```

Ejemplo 2: en este ejemplo se registran varias instancias con un periodo de mantenimiento.

```
$option1 =
 @{Key="InstanceIds";Values=@("i-0000293ffd8c57862","i-0cb2b964d3e14fd9f")}
Register-SSMTargetWithMaintenanceWindow -WindowId "mw-06cf17cbefcb4bf4f" -Target
 $option1 -OwnerInformation "Single instance" -ResourceType "INSTANCE"
```

Salida:

```
6ab5c208-9fc4-4697-84b7-b02a6cc25f7d
```

Ejemplo 3: en este ejemplo se registra una instancia con un periodo de mantenimiento mediante etiquetas de EC2.

```
$option1 = @{Key="tag:Environment";Values=@("Production")}
Register-SSMTargetWithMaintenanceWindow -WindowId "mw-06cf17cbefcb4bf4f" -Target
 $option1 -OwnerInformation "Production Web Servers" -ResourceType "INSTANCE"
```

Salida:

```
2994977e-aefb-4a71-beac-df620352f184
```

- Para obtener información sobre la API, consulte RegisterTargetWithMaintenanceWindow en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`RegisterTaskWithMaintenanceWindow`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `RegisterTaskWithMaintenanceWindow`.

CLI

AWS CLI

Ejemplo 1: registro de una tarea de Automatización con un periodo de mantenimiento

En el siguiente ejemplo de `register-task-with-maintenance-window` se registra una tarea de Automatización con un periodo de mantenimiento destinado a una instancia.

```
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"
```

Salida:

```
{
    "WindowTaskId":"11144444-5555-6666-7777-88888888"
}
```

Para obtener más información, consulte Registrar una tarea con el periodo de mantenimiento (AWS CLI) en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: registro de una tarea de Lambda con un periodo de mantenimiento

En el siguiente ejemplo de `register-task-with-maintenance-window` se registra una tarea de Lambda con un periodo de mantenimiento destinado a una instancia.

```
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"
```

Salida:

```
{
    "WindowTaskId":"22244444-5555-6666-7777-88888888"
}
```

Para obtener más información, consulte Registrar una tarea con el periodo de mantenimiento (AWS CLI) en la Guía del usuario de AWS Systems Manager.

Ejemplo 3: registro de una tarea de ejecución de un comando con un periodo de mantenimiento

En el siguiente ejemplo de `register-task-with-maintenance-window` se registra una tarea de ejecución de un comando con un periodo de mantenimiento destinado a una instancia.

```
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
```

Salida:

```
{
    "WindowTaskId":"33344444-5555-6666-7777-88888888"
}
```

Para obtener más información, consulte Registrar una tarea con el periodo de mantenimiento (AWS CLI) en la Guía del usuario de AWS Systems Manager.

Ejemplo 4: registro de una tarea de Step Functions con un periodo de mantenimiento

En el siguiente ejemplo de `register-task-with-maintenance-window` se registra una tarea de Step Functions con un periodo de mantenimiento destinado a una instancia.

```
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"
```

Salida:

```
{
    "WindowTaskId":"44444444-5555-6666-7777-88888888"
}
```

Para obtener más información, consulte Registrar una tarea con el periodo de mantenimiento (AWS CLI) en la Guía del usuario de AWS Systems Manager.

Ejemplo 5: registro de una tarea mediante un ID de destino de un periodo de mantenimiento

En el siguiente ejemplo de `register-task-with-maintenance-window` se registra una tarea mediante un ID de destino de un periodo de mantenimiento. El ID de destino del periodo de mantenimiento estaba en el resultado del comando `aws ssm register-target-with-maintenance-window`. También puede recuperarlo del resultado del comando `aws ssm describe-maintenance-window-targets`.

```
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
```

Salida:

```
{
    "WindowTaskId":"33344444-5555-6666-7777-88888888"
}
```

Para obtener más información, consulte Registrar una tarea con el periodo de mantenimiento (AWS CLI) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte RegisterTaskWithMaintenanceWindow en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se registra una tarea con un periodo de mantenimiento mediante un ID de instancia. El resultado es el ID de la tarea.

```
$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
```

Salida:

```
f34a2c47-ddfd-4c85-a88d-72366b69af1b
```

Ejemplo 2: en este ejemplo se registra una tarea con un periodo de mantenimiento mediante un ID de destino. El resultado es el ID de la tarea.

```
$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
```

Salida:

```
f34a2c47-ddfd-4c85-a88d-72366b69af1b
```

Ejemplo 3: en este ejemplo se crea un objeto de parámetro para el documento de ejecución de comandos **AWS-RunPowerShellScript** y se crea una tarea con un periodo de mantenimiento determinado mediante el ID de destino. El resultado devuelto es el ID de la tarea.

```
$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
```

Salida:

```
f1e2ef34-5678-12e3-456a-12334c5c6cbe
```

Ejemplo 4: en este ejemplo se registra una tarea de Automatización de AWS Systems Manager mediante un documento denominado **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"
```

- Para obtener información sobre la API, consulte RegisterTaskWithMaintenanceWindow en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se registra una tarea con un periodo de mantenimiento mediante un ID de instancia. El resultado es el ID de la tarea.

```
$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
```

Salida:

```
f34a2c47-ddfd-4c85-a88d-72366b69af1b
```

Ejemplo 2: en este ejemplo se registra una tarea con un periodo de mantenimiento mediante un ID de destino. El resultado es el ID de la tarea.

```
$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
```

Salida:

```
f34a2c47-ddfd-4c85-a88d-72366b69af1b
```

Ejemplo 3: en este ejemplo se crea un objeto de parámetro para el documento de ejecución de comandos **AWS-RunPowerShellScript** y se crea una tarea con un periodo de mantenimiento determinado mediante el ID de destino. El resultado devuelto es el ID de la tarea.

```
$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
```

Salida:

```
f1e2ef34-5678-12e3-456a-12334c5c6cbe
```

Ejemplo 4: en este ejemplo se registra una tarea de Automatización de AWS Systems Manager mediante un documento denominado **`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"
```

- Para obtener información sobre la API, consulte RegisterTaskWithMaintenanceWindow en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`RemoveTagsFromResource`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `RemoveTagsFromResource`.

CLI

AWS CLI

Cómo eliminar una etiqueta de una línea de base de revisiones

En el siguiente ejemplo de `remove-tags-from-resource` se eliminan las etiquetas de una línea de base de revisiones.

```
aws ssm remove-tags-from-resource \
```

```
    --resource-type "PatchBaseline" \
    --resource-id "pb-0123456789abcdef0" \
    --tag-keys "Region"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Tagging AWS Resources en la Referencia general de AWS.

- Para obtener información sobre la API, consulte RemoveTagsFromResource en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se elimina una etiqueta de un periodo de mantenimiento. No se obtienen resultados si el comando se ejecuta correctamente.

```
Remove-SSMResourceTag -ResourceId "mw-03eb9db42890fb82d" -ResourceType
 "MaintenanceWindow" -TagKey "Production"
```

- Para obtener información sobre la API, consulte RemoveTagsFromResource en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se elimina una etiqueta de un periodo de mantenimiento. No se obtienen resultados si el comando se ejecuta correctamente.

```
Remove-SSMResourceTag -ResourceId "mw-03eb9db42890fb82d" -ResourceType
 "MaintenanceWindow" -TagKey "Production"
```

- Para obtener información sobre la API, consulte RemoveTagsFromResource en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Uso de **SendCommand** con un SDK de AWS o la CLI

Los siguientes ejemplos de código muestran cómo utilizar `SendCommand`.

Los ejemplos de acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Puede ver esta acción en contexto en el siguiente ejemplo de código:

- Conceptos básicos

CLI

AWS CLI

Ejemplo 1: ejecución de un comando en una o más instancias remotas

En el siguiente ejemplo de `send-command` se ejecuta un comando `echo` en una instancia de destino.

```
aws ssm send-command \
    --document-name "AWS-RunShellScript" \
    --parameters 'commands=["echo HelloWorld"]' \
    --targets "Key=instanceids,Values=i-1234567890abcdef0" \
    --comment "echo HelloWorld"
```

Salida:

```
{
    "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
        }
    }
}
```

Para obtener más información, consulte Running Commands Using Systems Manager Run Command en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: obtención de información sobre la IP de una instancia

En el siguiente ejemplo de `send-command` se obtiene información sobre la IP de una instancia.

```
aws ssm send-command \
    --instance-ids "i-1234567890abcdef0" \
    --document-name "AWS-RunShellScript" \
    --comment "IP config" \
    --parameters "commands=ifconfig"
```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte Running Commands Using Systems Manager Run Command en la Guía del usuario de AWS Systems Manager.

Ejemplo 3: ejecución de un comando en instancias con etiquetas específicas

En el siguiente ejemplo de `send-command` se ejecuta un comando en instancias que tienen la clave de etiqueta “ENV” y el valor “Dev”.

```
aws ssm send-command \
    --targets "Key=tag:ENV,Values=Dev" \
    --document-name "AWS-RunShellScript" \
    --parameters "commands=ifconfig"
```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte Running Commands Using Systems Manager Run Command en la Guía del usuario de AWS Systems Manager.

Ejemplo 4: ejecución de un comando que envía notificaciones de SNS

En el siguiente ejemplo de `send-command` se ejecuta un comando que envía notificaciones de SNS para todos los eventos de notificación y el tipo de notificación `Command`.

```
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"
```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte Running Commands Using Systems Manager Run Command en la Guía del usuario de AWS Systems Manager.

Ejemplo 5: ejecución de un comando que genera un resultado en S3 y CloudWatch

En el siguiente ejemplo de `send-command` se ejecuta un comando que envía los detalles del comando a un bucket de S3 y a un grupo de registro de Registros de CloudWatch.

```
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"
```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte Running Commands Using Systems Manager Run Command en la Guía del usuario de AWS Systems Manager.

Ejemplo 6: ejecución de comandos en varias instancias con etiquetas diferentes

En el siguiente ejemplo de `send-command` se ejecuta un comando en instancias con dos claves y valores de etiqueta diferentes.

```
aws ssm send-command \
    --document-name "AWS-RunPowerShellScript" \
    --parameters commands=["echo helloWorld"] \
    --targets Key=tag:Env,Values=Dev Key=tag:Role,Values=WebServers
```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte Running Commands Using Systems Manager Run Command en la Guía del usuario de AWS Systems Manager.

Ejemplo 7: definición de varias instancias como destino con la misma clave de etiqueta

En el siguiente ejemplo de `send-command` se ejecuta un comando en instancias que tienen la misma clave de etiqueta, pero con valores diferentes.

```
aws ssm send-command \
    --document-name "AWS-RunPowerShellScript" \
    --parameters commands=["echo helloWorld"] \
    --targets Key=tag:Env,Values=Dev,Test
```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte Running Commands Using Systems Manager Run Command en la Guía del usuario de AWS Systems Manager.

Ejemplo 8: ejecución de un comando que usa un documento compartido

En el siguiente ejemplo de `send-command` se ejecuta un documento compartido en una instancia de destino.

```
aws ssm send-command \
    --document-name "arn:aws:ssm:us-east-1:123456789012:document/ExampleDocument"
 \
    --targets "Key=instanceids,Values=i-1234567890abcdef0"
```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte Uso compartido de documentos de SSM en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte SendCommand en la Referencia de comandos de la AWS CLI.

Java

SDK para Java 2.x

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
    /**
     * Sends a SSM command to a managed node asynchronously.
     *
     * @param documentName The name of the document to use.
     * @param instanceId The ID of the instance to send the command to.
     * @return The command ID.
     * <p>
     * This method initiates asynchronous requests to send a SSM command to a
managed node.
     * It waits until the document is active, sends the command, and checks the
command execution status.
     */
    public String sendSSMCommand(String documentName, String instanceId) throws
InterruptedException, SsmException {
        // Before we use Document to send a command - make sure it is active.
```

```
        CompletableFuture<Void> documentActiveFuture =
CompletableFuture.runAsync(() -> {
            boolean isDocumentActive = false;
            DescribeDocumentRequest request = DescribeDocumentRequest.builder()
                    .name(documentName)
                    .build();

            while (!isDocumentActive) {
                CompletableFuture<DescribeDocumentResponse> response =
getAsyncClient().describeDocument(request);
                String documentStatus =
response.join().document().statusAsString();
                if (documentStatus.equals("Active")) {
                    System.out.println("The SSM document is active and ready to
use.");
                    isDocumentActive = true;
                } else {
                    System.out.println("The SSM document is not active. Status: "
+ documentStatus);
                    try {
                        Thread.sleep(5000);
                    } catch (InterruptedException e) {
                        throw new RuntimeException(e);
                    }
                }
            }
        });

        documentActiveFuture.join();

        // Create the SendCommandRequest.
        SendCommandRequest commandRequest = SendCommandRequest.builder()
                .documentName(documentName)
                .instanceIds(instanceId)
                .build();

        // Send the command.
        CompletableFuture<SendCommandResponse> commandFuture =
getAsyncClient().sendCommand(commandRequest);
        final String[] commandId = {null};

        commandFuture.whenComplete((commandResponse, ex) -> {
            if (commandResponse != null) {
                commandId[0] = commandResponse.command().commandId();
```

```
                System.out.println("Command ID: " + commandId[0]);

                // Wait for the command execution to complete.
                GetCommandInvocationRequest invocationRequest =
GetCommandInvocationRequest.builder()
                        .commandId(commandId[0])
                        .instanceId(instanceId)
                        .build();

                try {
                    System.out.println("Wait 5 secs");
                    TimeUnit.SECONDS.sleep(5);

                    // Retrieve the command execution details.
                    CompletableFuture<GetCommandInvocationResponse>
invocationFuture = getAsyncClient().getCommandInvocation(invocationRequest);
                    invocationFuture.whenComplete((commandInvocationResponse,
invocationEx) -> {
                        if (commandInvocationResponse != null) {
                            // 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);
                            }
                        } else {
                            Throwable invocationCause = (invocationEx instanceof
CompletionException) ? invocationEx.getCause() : invocationEx;
                            throw new CompletionException(invocationCause);
                        }
                    }).join();
                } catch (InterruptedException e) {
                    throw new RuntimeException(e);
                }
            } else {
                Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
                if (cause instanceof SsmException) {
                    throw (SsmException) cause;
                } else {
```

```
                    throw new RuntimeException(cause);
                }
            }
        }).join();

        return commandId[0];
    }
```

- Para obtener información sobre la API, consulte SendCommand en la Referencia de la API de AWS SDK for Java 2.x.

JavaScript

SDK para JavaScript (v3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
import { SendCommandCommand, SSMClient } from "@aws-sdk/client-ssm";
import { parseArgs } from "node:util";

/**
 * Send an SSM command to a managed node.
 * @param {{ documentName: string }}
 */
export const main = async ({ documentName }) => {
  const client = new SSMClient({});
  try {
    await client.send(
      new SendCommandCommand({
        DocumentName: documentName,
      }),
    );
    console.log("Command sent successfully.");
    return { Success: true };
  } catch (caught) {
    if (caught instanceof Error && caught.name === "ValidationError") {
```

```
      console.warn(`${caught.message}. Did you provide a valid document name?`);
    } else {
      throw caught;
    }
  }
};
```

- Para obtener información sobre la API, consulte SendCommand en la Referencia de la API de AWS SDK para JavaScript.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se ejecuta un comando echo en una instancia de destino.

```
Send-SSMCommand -DocumentName "AWS-RunPowerShellScript" -Parameter @{commands =
 "echo helloWorld"} -Target @{Key="instanceids";Values=@("i-0cb2b964d3e14fd9f")}
```

Salida:

```
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}
```

Ejemplo 2: en este ejemplo se muestra cómo ejecutar un comando que acepte parámetros anidados.

```
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
```

- Para obtener información sobre la API, consulte SendCommand en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se ejecuta un comando echo en una instancia de destino.

```
Send-SSMCommand -DocumentName "AWS-RunPowerShellScript" -Parameter @{commands =
 "echo helloWorld"} -Target @{Key="instanceids";Values=@("i-0cb2b964d3e14fd9f")}
```

Salida:

```
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}
```

Ejemplo 2: en este ejemplo se muestra cómo ejecutar un comando que acepte parámetros anidados.

```
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
```

- Para obtener información sobre la API, consulte SendCommand en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Python

SDK para Python (Boto3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
class DocumentWrapper:
    """Encapsulates AWS Systems Manager Document actions."""

    def __init__(self, ssm_client):
        """
        :param ssm_client: A Boto3 Systems Manager client.
        """
        self.ssm_client = ssm_client
        self.name = None

    @classmethod
    def from_client(cls):
        ssm_client = boto3.client("ssm")
        return cls(ssm_client)


    def send_command(self, instance_ids):
```

```
        """
        Sends a command to one or more instances.

        :param instance_ids: The IDs of the instances to send the command to.
        :return: The ID of the command.
        """
        try:
            response = self.ssm_client.send_command(
                InstanceIds=instance_ids, DocumentName=self.name,
TimeoutSeconds=3600
            )
            return response["Command"]["CommandId"]
        except ClientError as err:
            logger.error(
                "Couldn't send command to %s. Here's why: %s: %s",
                self.name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
```

- Para obtener información sobre la API, consulte SendCommand en la Referencia de la API de AWS SDK para Python (Boto3).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`StartAutomationExecution`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `StartAutomationExecution`.

CLI

AWS CLI

Ejemplo 1: ejecución de un documento de automatización

En el siguiente ejemplo de `start-automation-execution` se ejecuta un documento de Automatización.

```
aws ssm start-automation-execution \
    --document-name "AWS-UpdateLinuxAmi" \
    --parameters "AutomationAssumeRole=arn:aws:iam::123456789012:role/
SSMAutomationRole,SourceAmiId=ami-EXAMPLE,IamInstanceProfileName=EC2InstanceRole"
```

Salida:

```
{
  "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0a1b2EXAMPLE"
}
```

Para obtener más información, consulte Ejecución manual de un flujo de trabajo de Automatización en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: ejecución de un documento de automatización compartido

En el siguiente ejemplo de `start-automation-execution` se ejecuta un documento de Automatización compartido.

```
aws ssm start-automation-execution \
    --document-name "arn:aws:ssm:us-east-1:123456789012:document/ExampleDocument"
```

Salida:

```
{
  "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0a1b2EXAMPLE"
}
```

Para obtener más información, consulte Uso compartido de documentos de SSM en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte StartAutomationExecution en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se ejecuta un documento que especifica un rol de Automatización, un ID de origen de AMI y un rol de instancia de Amazon EC2.

```
Start-SSMAutomationExecution -DocumentName AWS-UpdateLinuxAmi -
Parameter @{'AutomationAssumeRole'='arn:aws:iam::123456789012:role/
SSMAutomationRole';'SourceAmiId'='ami-
f173cc91';'InstanceIamRole'='EC2InstanceRole'}
```

Salida:

```
3a532a4f-0382-11e7-9df7-6f11185f6dd1
```

- Para obtener información sobre la API, consulte StartAutomationExecution en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se ejecuta un documento que especifica un rol de Automatización, un ID de origen de AMI y un rol de instancia de Amazon EC2.

```
Start-SSMAutomationExecution -DocumentName AWS-UpdateLinuxAmi -
Parameter @{'AutomationAssumeRole'='arn:aws:iam::123456789012:role/
SSMAutomationRole';'SourceAmiId'='ami-
f173cc91';'InstanceIamRole'='EC2InstanceRole'}
```

Salida:

```
3a532a4f-0382-11e7-9df7-6f11185f6dd1
```

- Para obtener información sobre la API, consulte StartAutomationExecution en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`StartSession`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `StartSession`.

CLI

AWS CLI

Ejemplo 1: para iniciar una sesión de Session Manager

Este ejemplo de `start-session` establece una conexión a una instancia para una sesión de Session Manager. Tenga en cuenta que este comando interactivo requiere que el complemento Session Manager esté instalado en la máquina cliente que realiza la llamada.

```
aws ssm start-session \
    --target "i-1234567890abcdef0"
```

Salida:

```
Starting session with SessionId: Jane-Roe-07a16060613c408b5
```

Ejemplo 2: iniciar una sesión de Session Manager mediante SSH

Este ejemplo de `start-session` establece una conexión a una instancia para una sesión de Session Manager mediante SSH. Tenga en cuenta que este comando interactivo requiere que el complemento Session Manager esté instalado en la máquina cliente que realiza la llamada y que el comando use el usuario predeterminado de la instancia, como `ec2-user` en el caso de las instancias EC2 para Linux.

```
ssh -i /path/my-key-pair.pem ec2-user@i-02573cafcfEXAMPLE
```

Salida:

```
Starting session with SessionId: ec2-user-07a16060613c408b5
```

Para obtener más información, consulte Iniciar una sesión e Instalar el complemento Session Manager para la CLI de AWS en la Guía del usuario de AWS.

- Para obtener más información sobre la API, consulte StartSession en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: este ejemplo inicia una conexión a un destino para una sesión de Session Manager, lo que habilita el reenvío de puertos.

```
Start-SSMSession -Target 'i-064578e5e7454488f' -DocumentName 'AWS-
StartPortForwardingSession' -Parameter @{ localPortNumber = '8080'; portNumber =
 '80' }
```

Salida:

```
SessionId    StreamUrl
----------    ----------
random-id0    wss://ssmmessages.amazonaws.com/v1/data-channel/random-id
```

- Para obtener información sobre la API, consulte StartSession en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: este ejemplo inicia una conexión a un destino para una sesión de Session Manager, lo que habilita el reenvío de puertos. Nota: Debe agregar el parámetro Region si aún no lo ha establecido con AWSCredentials.

```
Start-SSMSession -Target 'i-064578e5e7454488f' -DocumentName 'AWS-
StartPortForwardingSession' -Parameter @{ localPortNumber = '8080'; portNumber =
 '80' } -Region 'us-west-1'
```

Salida:

```
Starting session with SessionId: testUser-xi4glew849asyeryde34u4dfsdfy
Port 8080 opened for sessionId testUser-xi4glew849asyeryde34u4dfsdfy.
Waiting for connections...
```

Ejemplo 2: este ejemplo crea una sesión interactiva con una instancia específica para una sesión del Session Manager.

```
Start-SSMSession -Target 'i-1234567890abcdef0' -Region 'us-west-1'
```

Salida:

```
Starting session with SessionId : testUser-xi4glew849asyeryde34u4dfsdfy
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.



Install the latest PowerShell for new features and improvements!


PS C:\Windows\system32> whoami
ec2amaz-fnsdrwv\ec2-test-user
PS C:\Windows\system32>
```

Ejemplo 3: este ejemplo crea una sesión sin conectarse a ella y devuelve las propiedades SessionId, StreamUrl y TokenValue necesarias para conectarse a la sesión.

```
Start-SSMSession -Target 'i-1234567890abcdef0' -Region 'us-west-1' -
DisablePluginInvocation
```

Salida:

```
SessionId        : testUser-xi4glew849asyeryde34u4dfsdfy
StreamUrl        : {StreamUrl value redacted}
TokenValue       : {Token value redacted}
ContentLength    : 1207
HttpStatusCode   : OK
```

- Para obtener información sobre la API, consulte StartSession en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`StopAutomationExecution`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `StopAutomationExecution`.

CLI

AWS CLI

Detención de una ejecución de automatización

En el siguiente ejemplo de `stop-automation-execution` se detiene un documento de Automatización.

```
aws ssm stop-automation-execution
    --automation-execution-id "4105a4fc-f944-11e6-9d32-0a1b2EXAMPLE"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Ejecución manual de un flujo de trabajo de Automatización en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte StopAutomationExecution en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se detiene una ejecución de Automatización. No se obtienen resultados si el comando se ejecuta correctamente.

```
Stop-SSMAutomationExecution -AutomationExecutionId "4105a4fc-
f944-11e6-9d32-8fb2db27a909"
```

- Para obtener información sobre la API, consulte StopAutomationExecution en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se detiene una ejecución de Automatización. No se obtienen resultados si el comando se ejecuta correctamente.

```
Stop-SSMAutomationExecution -AutomationExecutionId "4105a4fc-
f944-11e6-9d32-8fb2db27a909"
```

- Para obtener información sobre la API, consulte StopAutomationExecution en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`UpdateAssociation`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `UpdateAssociation`.

CLI

AWS CLI

Ejemplo 1: actualización de una asociación de documentos

En el siguiente ejemplo de `update-association` se actualiza una asociación con una nueva versión del documento.

```
aws ssm update-association \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
    --document-version "\$LATEST"
```

Salida:

```
{
    "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
    }
}
```

Para obtener más información, consulte Edición y creación de una nueva versión de una asociación en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: actualización de la expresión de programación de una asociación

En el siguiente ejemplo de `update-association` se actualiza la expresión de programación de la asociación especificada.

```
aws ssm update-association \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
    --schedule-expression "cron(0 0 0/4 1/1 * ? *)"
```

Salida:

```
{
    "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
    }
}
```

Para obtener más información, consulte Edición y creación de una nueva versión de una asociación en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte UpdateAssociation en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se actualiza una asociación con una nueva versión del documento.

```
Update-SSMAssociation -AssociationId "93285663-92df-44cb-9f26-2292d4ecc439" -
DocumentVersion "1"
```

Salida:

```
Name                  : AWS-UpdateSSMAgent
InstanceId            :
Date                  : 3/1/2017 6:22:21 PM
Status.Name           :
Status.Date           :
Status.Message        :
Status.AdditionalInfo :
```

- Para obtener información sobre la API, consulte UpdateAssociation en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se actualiza una asociación con una nueva versión del documento.

```
Update-SSMAssociation -AssociationId "93285663-92df-44cb-9f26-2292d4ecc439" -
DocumentVersion "1"
```

Salida:

```
Name                  : AWS-UpdateSSMAgent
InstanceId            :
Date                  : 3/1/2017 6:22:21 PM
Status.Name           :
Status.Date           :
Status.Message        :
Status.AdditionalInfo :
```

- Para obtener información sobre la API, consulte UpdateAssociation en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **UpdateAssociationStatus** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `UpdateAssociationStatus`.

CLI

AWS CLI

Actualización del estado de la asociación

En el siguiente ejemplo de `update-association-status` se actualiza el estado de la asociación entre una instancia y un documento.

```
aws ssm update-association-status \
    --name "AWS-UpdateSSMAgent" \
    --instance-id "i-1234567890abcdef0" \
    --association-
status "Date=1424421071.939,Name=Pending,Message=temp_status_change,AdditionalInfo=Additi
Config-Needed"
```

Salida:

```
{
    "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
    }
}
```

Para obtener más información, consulte Trabajo con asociaciones en Systems Manager en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte UpdateAssociationStatus en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se actualiza el estado de la asociación entre una instancia y un documento de configuración.

```
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"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte UpdateAssociationStatus en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se actualiza el estado de la asociación entre una instancia y un documento de configuración.

```
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"
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte UpdateAssociationStatus en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`UpdateDocument`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `UpdateDocument`.

CLI

AWS CLI

Creación de una nueva versión de un documento

En el siguiente ejemplo de `update-document` se crea una nueva versión de un documento cuando se ejecuta en una computador con Windows. El documento especificado por `--document` debe estar en formato JSON. Tenga en cuenta que se debe hacer referencia a `file://` seguido de la ruta del archivo de contenido. Debido a que $ está al principio del parámetro `--document-version`, en Windows debe escribir el valor entre comillas dobles. En Linux, MacOS o en una línea de comandos de PowerShell, debe escribir el valor entre comillas simples.

Versión de Windows:

```
aws ssm update-document \
    --name "RunShellScript" \
    --content "file://RunShellScript.json" \
    --document-version "$LATEST"
```

Versión de Linux o Mac:

```
aws ssm update-document \
    --name "RunShellScript" \
    --content "file://RunShellScript.json" \
    --document-version '$LATEST'
```

Salida:

```
{
  "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"
  }
}
```

- Para obtener información sobre la API, consulte UpdateDocument en la Referencia de la API de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se crea una nueva versión de un documento con el contenido actualizado del archivo JSON que especifique. El documento debe estar en formato JSON. Puede obtener la versión del documento con el cmdlet “Get-SSMDocumentVersionList”.

```
Update-SSMDocument -Name RunShellScript -DocumentVersion "1" -Content (Get-
Content -Raw "c:\temp\RunShellScript.json")
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte UpdateDocument en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se crea una nueva versión de un documento con el contenido actualizado del archivo JSON que especifique. El documento debe estar en formato JSON. Puede obtener la versión del documento con el cmdlet “Get-SSMDocumentVersionList”.

```
Update-SSMDocument -Name RunShellScript -DocumentVersion "1" -Content (Get-
Content -Raw "c:\temp\RunShellScript.json")
```

Salida:

```
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
```

- Para obtener información sobre la API, consulte UpdateDocument en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`UpdateDocumentDefaultVersion`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `UpdateDocumentDefaultVersion`.

CLI

AWS CLI

Actualización de la versión predeterminada de un documento

En el siguiente ejemplo de `update-document-default-version` se actualiza la versión predeterminada de un documento de Systems Manager.

```
aws ssm update-document-default-version \
    --name "Example" \
    --document-version "2"
```

Salida:

```
{
    "Description": {
        "Name": "Example",
        "DefaultVersion": "2"
    }
}
```

Para obtener más información, consulte Escribir contenido en el documento de SSM en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte UpdateDocumentDefaultVersion en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se actualiza la versión predeterminada de un documento. Puede obtener las versiones de los documentos disponibles con el cmdlet “Get-SSMDocumentVersionList”.

```
Update-SSMDocumentDefaultVersion -Name "RunShellScript" -DocumentVersion "2"
```

Salida:

```
DefaultVersion Name
-------------- ----
2              RunShellScript
```

- Para obtener información sobre la API, consulte UpdateDocumentDefaultVersion en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se actualiza la versión predeterminada de un documento. Puede obtener las versiones de los documentos disponibles con el cmdlet “Get-SSMDocumentVersionList”.

```
Update-SSMDocumentDefaultVersion -Name "RunShellScript" -DocumentVersion "2"
```

Salida:

```
DefaultVersion Name
-------------- ----
2              RunShellScript
```

- Para obtener información sobre la API, consulte UpdateDocumentDefaultVersion en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Uso de **`UpdateMaintenanceWindow`** con un SDK de AWS o la CLI

Los siguientes ejemplos de código muestran cómo utilizar `UpdateMaintenanceWindow`.

CLI

AWS CLI

Ejemplo 1: actualización de un periodo de mantenimiento

En el siguiente ejemplo de `update-maintenance-window` se actualiza el nombre de un periodo de mantenimiento.

```
aws ssm update-maintenance-window \
    --window-id "mw-1a2b3c4d5e6f7g8h9" \
    --name "My-Renamed-MW"
```

Salida:

```
{
    "Cutoff": 1,
    "Name": "My-Renamed-MW",
    "Schedule": "cron(0 16 ? * TUE *)",
    "Enabled": true,
    "AllowUnassociatedTargets": true,
    "WindowId": "mw-1a2b3c4d5e6f7g8h9",
    "Duration": 4
}
```

Ejemplo 2: desactivación de un periodo de mantenimiento

En el siguiente ejemplo de `update-maintenance-window` se desactiva un periodo de mantenimiento.

```
aws ssm update-maintenance-window \
    --window-id "mw-1a2b3c4d5e6f7g8h9" \
    --no-enabled
```

Ejemplo 3: activación de un periodo de mantenimiento

En el siguiente ejemplo de `update-maintenance-window` se activa un periodo de mantenimiento.

```
aws ssm update-maintenance-window \
    --window-id "mw-1a2b3c4d5e6f7g8h9" \
    --enabled
```

Para obtener más información, consulte Actualizar un período de mantenimiento (AWS CLI) en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte UpdateMaintenanceWindow en la Referencia de comandos de la AWS CLI.

Java

SDK para Java 2.x

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
   /**
    * Updates an SSM maintenance window asynchronously.
    *
    * @param id The ID of the maintenance window to update.
    * @param name The new name for the maintenance window.
    * <p>
    * This method initiates an asynchronous request to update an SSM maintenance
window.
    * If the request is successful, it prints a success message.
    * If an exception occurs, it handles the error appropriately.
    */
   public void updateSSMMaintenanceWindow(String id, String name) throws
SsmException {
       UpdateMaintenanceWindowRequest updateRequest =
UpdateMaintenanceWindowRequest.builder()
               .windowId(id)
               .allowUnassociatedTargets(true)
               .duration(24)
               .enabled(true)
               .name(name)
               .schedule("cron(0 0 ? * MON *)")
```

```
                .build();

        CompletableFuture<UpdateMaintenanceWindowResponse> future =
getAsyncClient().updateMaintenanceWindow(updateRequest);
        future.whenComplete((response, ex) -> {
            if (response != null) {
                System.out.println("The SSM maintenance window was successfully
updated");
            } else {
                Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
                if (cause instanceof SsmException) {
                    throw new CompletionException(cause);
                } else {
                    throw new RuntimeException(cause);
                }
            }
        }).join();
    }
```

- Para obtener detalles sobre la API, consulte UpdateMaintenanceWindow en la Referencia de la API de AWS SDK for Java 2.x.

JavaScript

SDK para JavaScript (v3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
import { UpdateMaintenanceWindowCommand, SSMClient } from "@aws-sdk/client-ssm";
import { parseArgs } from "node:util";

/**
 * Update an SSM maintenance window.
 * @param {{ windowId: string, allowUnassociatedTargets?: boolean, duration?:
 number, enabled?: boolean, name?: string, schedule?: string }}
```

```
 */
export const main = async ({
  windowId,
  allowUnassociatedTargets = undefined, //Allow the maintenance window to run on
 managed nodes, even if you haven't registered those nodes as targets.
  duration = undefined, //The duration of the maintenance window in hours.
  enabled = undefined,
  name = undefined,
  schedule = undefined, //The schedule of the maintenance window in the form of a
 cron or rate expression.
}) => {
  const client = new SSMClient({});
  try {
    const { opsItemArn, opsItemId } = await client.send(
      new UpdateMaintenanceWindowCommand({
        WindowId: windowId,
        AllowUnassociatedTargets: allowUnassociatedTargets,
        Duration: duration,
        Enabled: enabled,
        Name: name,
        Schedule: schedule,
      }),
    );
    console.log("Maintenance window updated.");
    return { OpsItemArn: opsItemArn, OpsItemId: opsItemId };
  } catch (caught) {
    if (caught instanceof Error && caught.name === "ValidationError") {
      console.warn(`${caught.message}. Are these values correct?`);
    } else {
      throw caught;
    }
  }
};
```

- Para obtener detalles sobre la API, consulte UpdateMaintenanceWindow en la Referencia de la API de AWS SDK para JavaScript.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se actualiza el nombre de un periodo de mantenimiento.

```
Update-SSMMaintenanceWindow -WindowId "mw-03eb9db42890fb82d" -Name "My-Renamed-
MW"
```

Salida:

```
AllowUnassociatedTargets : False
Cutoff                   : 1
Duration                 : 2
Enabled                  : True
Name                     : My-Renamed-MW
Schedule                 : cron(0 */30 * * * ? *)
WindowId                 : mw-03eb9db42890fb82d
```

Ejemplo 2: en este ejemplo se activa un periodo de mantenimiento.

```
Update-SSMMaintenanceWindow -WindowId "mw-03eb9db42890fb82d" -Enabled $true
```

Salida:

```
AllowUnassociatedTargets : False
Cutoff                   : 1
Duration                 : 2
Enabled                  : True
Name                     : My-Renamed-MW
Schedule                 : cron(0 */30 * * * ? *)
WindowId                 : mw-03eb9db42890fb82d
```

Ejemplo 3: en este ejemplo se desactiva un periodo de mantenimiento.

```
Update-SSMMaintenanceWindow -WindowId "mw-03eb9db42890fb82d" -Enabled $false
```

Salida:

```
AllowUnassociatedTargets : False
Cutoff                   : 1
Duration                 : 2
Enabled                  : False
Name                     : My-Renamed-MW
Schedule                 : cron(0 */30 * * * ? *)
WindowId                 : mw-03eb9db42890fb82d
```

- Para obtener información sobre la API, consulte UpdateMaintenanceWindow en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se actualiza el nombre de un periodo de mantenimiento.

```
Update-SSMMaintenanceWindow -WindowId "mw-03eb9db42890fb82d" -Name "My-Renamed-
MW"
```

Salida:

```
AllowUnassociatedTargets : False
Cutoff                   : 1
Duration                 : 2
Enabled                  : True
Name                     : My-Renamed-MW
Schedule                 : cron(0 */30 * * * ? *)
WindowId                 : mw-03eb9db42890fb82d
```

Ejemplo 2: en este ejemplo se activa un periodo de mantenimiento.

```
Update-SSMMaintenanceWindow -WindowId "mw-03eb9db42890fb82d" -Enabled $true
```

Salida:

```
AllowUnassociatedTargets : False
Cutoff                   : 1
Duration                 : 2
Enabled                  : True
Name                     : My-Renamed-MW
Schedule                 : cron(0 */30 * * * ? *)
WindowId                 : mw-03eb9db42890fb82d
```

Ejemplo 3: en este ejemplo se desactiva un periodo de mantenimiento.

```
Update-SSMMaintenanceWindow -WindowId "mw-03eb9db42890fb82d" -Enabled $false
```

Salida:

```
AllowUnassociatedTargets : False
Cutoff                   : 1
```

```
Duration                  : 2
Enabled                   : False
Name                      : My-Renamed-MW
Schedule                  : cron(0 */30 * * * ? *)
WindowId                  : mw-03eb9db42890fb82d
```

- Para obtener información sobre la API, consulte UpdateMaintenanceWindow en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Python

SDK para Python (Boto3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
class MaintenanceWindowWrapper:
    """Encapsulates AWS Systems Manager maintenance window actions."""

    def __init__(self, ssm_client):
        """
        :param ssm_client: A Boto3 Systems Manager client.
        """
        self.ssm_client = ssm_client
        self.window_id = None
        self.name = None

    @classmethod
    def from_client(cls):
        ssm_client = boto3.client("ssm")
        return cls(ssm_client)


    def update(
        self, name, enabled, schedule, duration, cutoff,
 allow_unassociated_targets
    ):
        """
```

```
        Update an AWS Systems Manager maintenance window.

        :param name: The name of the maintenance window.
        :param enabled: Whether the maintenance window is enabled to run on
managed nodes.
        :param schedule: The schedule of the maintenance window.
        :param duration: The duration of the maintenance window.
        :param cutoff: The cutoff time of the maintenance window.
        :param allow_unassociated_targets: Allow the maintenance window to run on
managed nodes, even
                                           if you haven't registered those nodes
as targets.
        """
        try:
            self.ssm_client.update_maintenance_window(
                WindowId=self.window_id,
                Name=name,
                Enabled=enabled,
                Schedule=schedule,
                Duration=duration,
                Cutoff=cutoff,
                AllowUnassociatedTargets=allow_unassociated_targets,
            )
            self.name = name
            logger.info("Updated maintenance window %s.", self.window_id)
        except ParamValidationError as error:
            logger.error(
                "Parameter validation error when trying to update maintenance
window %s. Here's why: %s",
                self.window_id,
                error,
            )
            raise
        except ClientError as err:
            logger.error(
                "Couldn't update maintenance window %s. Here's why: %s: %s",
                self.name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
```

- Para obtener información sobre las API, consulte UpdateMaintenanceWindow en la Referencia de la API de AWS SDK para Python (Boto3).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`UpdateManagedInstanceRole`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `UpdateManagedInstanceRole`.

CLI

AWS CLI

Actualización del rol de IAM de una instancia administrada

En el siguiente ejemplo de `update-managed-instance-role` se actualiza el perfil de instancia de IAM de una instancia gestionada.

```
aws ssm update-managed-instance-role \
    --instance-id "mi-08ab247cdfEXAMPLE" \
    --iam-role "ExampleRole"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Step 4: Create an IAM Instance Profile for Systems Manager en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte UpdateManagedInstanceRole en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se actualiza el rol de una instancia administrada. No se obtienen resultados si el comando se ejecuta correctamente.

```
Update-SSMManagedInstanceRole -InstanceId "mi-08ab247cdf1046573" -IamRole
 "AutomationRole"
```

- Para obtener información sobre la API, consulte UpdateManagedInstanceRole en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se actualiza el rol de una instancia administrada. No se obtienen resultados si el comando se ejecuta correctamente.

```
Update-SSMManagedInstanceRole -InstanceId "mi-08ab247cdf1046573" -IamRole
 "AutomationRole"
```

- Para obtener información sobre la API, consulte UpdateManagedInstanceRole en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Uso de **`UpdateOpsItem`** con un SDK de AWS o la CLI

Los siguientes ejemplos de código muestran cómo utilizar `UpdateOpsItem`.

Los ejemplos de acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Puede ver esta acción en contexto en el siguiente ejemplo de código:

- Conceptos básicos

CLI

AWS CLI

Cómo actualizar un OpsItem

En el siguiente ejemplo de `update-ops-item` se actualizan la descripción, la prioridad y la categoría de un OpsItem. Además, el comando especifica un tema de SNS al que se envían las notificaciones cuando se edita o cambia este OpsItem.

```
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"
```

Salida:

```
This command produces no output.
```

Para obtener más información, consulte Trabajo con OpsItems en la AWSGuía del usuario de Systems Manager.

- Para obtener información sobre la API, consulte UpdateOpsItem en la Referencia de comandos de la AWS CLI.

Java

SDK para Java 2.x

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
/**
 * Resolves an AWS SSM OpsItem asynchronously.
 *
 * @param opsID The ID of the OpsItem to resolve.
 * <p>
 * This method initiates an asynchronous request to resolve an SSM OpsItem.
 * If an exception occurs, it handles the error appropriately.
 */
public void resolveOpsItem(String opsID) {
    UpdateOpsItemRequest opsItemRequest = UpdateOpsItemRequest.builder()
            .opsItemId(opsID)
            .status(OpsItemStatus.RESOLVED)
            .build();

    CompletableFuture<Void> future = CompletableFuture.runAsync(() -> {
        getAsyncClient().updateOpsItem(opsItemRequest)
                .thenAccept(response -> {
                    System.out.println("OpsItem resolved successfully.");
```

```
                    })
                    .exceptionally(ex -> {
                        throw new CompletionException(ex);
                    }).join();
        }).exceptionally(ex -> {
            Throwable cause = (ex instanceof CompletionException) ?
ex.getCause() : ex;
            if (cause instanceof SsmException) {
                throw new RuntimeException("SSM error: " + cause.getMessage(),
cause);
            } else {
                throw new RuntimeException("Unexpected error: " +
cause.getMessage(), cause);
            }
        });

        try {
            future.join();
        } catch (CompletionException ex) {
            throw ex.getCause() instanceof RuntimeException ? (RuntimeException)
ex.getCause() : ex;
        }
    }
```

- Para obtener información sobre la API, consulte UpdateOpsItem en la referencia de la API de AWS SDK for Java 2.x.

JavaScript

SDK para JavaScript (v3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
import { UpdateOpsItemCommand, SSMClient } from "@aws-sdk/client-ssm";
import { parseArgs } from "node:util";
```

```
/**
 * Update an SSM OpsItem.
 * @param {{ opsItemId: string, status?: OpsItemStatus }}
 */
export const main = async ({
  opsItemId,
  status = undefined, // The OpsItem status. Status can be Open, In Progress, or
 Resolved
}) => {
  const client = new SSMClient({});
  try {
    await client.send(
      new UpdateOpsItemCommand({
        OpsItemId: opsItemId,
        Status: status,
      }),
    );
    console.log("Ops item updated.");
    return { Success: true };
  } catch (caught) {
    if (
      caught instanceof Error &&
      caught.name === "OpsItemLimitExceededException"
    ) {
      console.warn(
        `Couldn't create ops item because you have exceeded your open OpsItem
 limit. ${caught.message}.`,
      );
    } else {
      throw caught;
    }
  }
};
```

- Para obtener información sobre la API, consulte UpdateOpsItem en la referencia de la API de AWS SDK para JavaScript.

Python

SDK para Python (Boto3)

> **Note**
>
> Hay más en GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el Repositorio de ejemplos de código de AWS.

```
class OpsItemWrapper:
    """Encapsulates AWS Systems Manager OpsItem actions."""

    def __init__(self, ssm_client):
        """
        :param ssm_client: A Boto3 Systems Manager client.
        """
        self.ssm_client = ssm_client
        self.id = None

    @classmethod
    def from_client(cls):
        """
        :return: A OpsItemWrapper instance.
        """
        ssm_client = boto3.client("ssm")
        return cls(ssm_client)


    def update(self, title=None, description=None, status=None):
        """
        Update an OpsItem.

        :param title: The new OpsItem title.
        :param description: The new OpsItem description.
        :param status: The new OpsItem status.
        :return:
        """
        args = dict(OpsItemId=self.id)
        if title is not None:
            args["Title"] = title
        if description is not None:
```

```
            args["Description"] = description
        if status is not None:
            args["Status"] = status
        try:
            self.ssm_client.update_ops_item(**args)
        except ClientError as err:
            logger.error(
                "Couldn't update ops item %s. Here's why: %s: %s",
                self.id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
```

- Para obtener información sobre la API, consulte UpdateOpsItem en la Referencia de la API de AWS SDK para Python (Boto3).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Utilizar **`UpdatePatchBaseline`** con una CLI

Los siguientes ejemplos de código muestran cómo utilizar `UpdatePatchBaseline`.

CLI

AWS CLI

Ejemplo 1: actualización de una línea de base de revisiones

En el siguiente ejemplo de `update-patch-baseline` se agregan dos revisiones especificadas como rechazadas y una revisión como aprobada a la línea de base de revisiones especificada.

```
aws ssm update-patch-baseline \
        --baseline-id "pb-0123456789abcdef0" \
        --rejected-patches "KB2032276" "MS10-048" \
        --approved-patches "KB2124261"
```

Salida:

```
{
    "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": []
}
```

Ejemplo 2: cambio del nombre de una línea de base de revisiones

En el siguiente ejemplo de `update-patch-baseline` se cambia el nombre de la línea de base de revisiones especificada.

```
aws ssm update-patch-baseline \
    --baseline-id "pb-0713accee01234567" \
    --name "Windows-Server-2012-R2-Important-and-Critical-Security-Updates"
```

Para obtener más información, consulte Actualización o eliminación de una línea de base de revisiones <https://docs.aws.amazon.com/systems-manager/latest/userguide/patch-baseline-update-or-delete.html>`__ en la Guía del usuario de AWS Systems Manager.

- Para obtener información sobre la API, consulte UpdatePatchBaseline en la Referencia de comandos de la AWS CLI.

PowerShell

Herramientas para PowerShell V4

Ejemplo 1: en este ejemplo se agregan dos revisiones como rechazadas y una revisión como aprobada a una línea de base de revisiones existente.

```
Update-SSMPatchBaseline -BaselineId "pb-03da896ca3b68b639" -RejectedPatch
 "KB2032276","MS10-048" -ApprovedPatch "KB2124261"
```

Salida:

```
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}
```

- Para obtener información sobre la API, consulte UpdatePatchBaseline en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V4).

Herramientas para PowerShell V5

Ejemplo 1: en este ejemplo se agregan dos revisiones como rechazadas y una revisión como aprobada a una línea de base de revisiones existente.

```
Update-SSMPatchBaseline -BaselineId "pb-03da896ca3b68b639" -RejectedPatch
 "KB2032276","MS10-048" -ApprovedPatch "KB2124261"
```

Salida:

```
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}
```

- Para obtener información sobre la API, consulte UpdatePatchBaseline en la Referencia de Cmdlet de las Herramientas de AWS para PowerShell (V5).

Para obtener una lista completa de las guías para desarrolladores de AWS SDK y ejemplos de código, consulte Cómo utilizar este servicio con un AWS SDK. En este tema también se incluye información sobre cómo comenzar a utilizar el SDK y detalles sobre sus versiones anteriores.

Registro y monitorización en AWS Systems Manager

La supervisión es un aspecto importante del mantenimiento de la fiabilidad, la disponibilidad y el rendimiento de AWS Systems Manager y sus soluciones de AWS. Debe recopilar los datos de monitoreo de todas las partes de la solución de AWS para que pueda depurar un error que se produce en distintas partes del código, en caso de que ocurra. Antes de comenzar a monitorear Systems Manager, cree un plan de monitoreo que incluya respuestas a las siguientes preguntas:

- ¿Cuáles son los objetivos de la supervisión?
- ¿Qué recursos va a supervisar?
- ¿Con qué frecuencia va a supervisar estos recursos?
- ¿Qué herramientas de supervisión va a utilizar?
- ¿Quién se encarga de realizar las tareas de monitorización?
- ¿Quién debería recibir una notificación cuando surjan problemas?

Después de definir los objetivos y de crear el plan de monitoreo, el paso siguiente consiste en establecer un punto de referencia para el rendimiento normal de Systems Manager en el entorno. Conviene medir el rendimiento de Systems Manager en varias ocasiones y con diferentes condiciones de carga. A medida que monitorea Systems Manager, guarde un historial de los datos de monitoreo que recopila. Puede comparar el rendimiento actual de Systems Manager con los datos históricos para identificar patrones de rendimiento normal y anomalías en el desempeño, así como crear métodos para solucionarlos.

Por ejemplo, puede monitorizar si se llevan a cabo correctamente o no operaciones como los flujos de trabajo de Automation, la aplicación de líneas de base de revisiones, los eventos de periodos de mantenimiento o la conformidad de la configuración. Automatización es una herramienta de AWS Systems Manager.

También puede monitorear la utilización de la CPU, las E/S de disco y la utilización de la red de los nodos administrados. Si el rendimiento no alcanza los valores del punto de referencia establecido, es posible que deba volver a configurar u optimizar el nodo para reducir la utilización de la CPU, mejorar la E/S de disco o reducir el tráfico de red. Para obtener más información sobre el monitoreo de instancias de EC2, consulte Monitoreo de Amazon EC2 en la Guía del usuario de Amazon EC2.

Temas

- Herramientas de supervisión

- Envío de registros de nodos a los Registros de CloudWatch (agente de CloudWatch) unificado
- Envío de registros de SSM Agent a CloudWatch Logs
- Supervisión de los eventos de las solicitudes de cambio
- Monitoreo de las automatizaciones
- Monitoreo de métricas de Run Command con Amazon CloudWatch
- Registro de llamadas a la API de AWS Systems Manager con AWS CloudTrail
- Registro de salida de acción de Automation con CloudWatch Logs
- Configuración de Registros de Amazon CloudWatch para Run Command
- Monitoreo de eventos de Systems Manager con Amazon EventBridge
- Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS

Herramientas de supervisión

El contenido de este capítulo proporciona información sobre el uso de las herramientas disponibles para monitorear sus recursos de Systems Manager y otros recursos de AWS. Para obtener una lista más completa de herramientas, consulte Registro y monitorización en AWS Systems Manager.

Envío de registros de nodos a los Registros de CloudWatch (agente de CloudWatch) unificado

Puede configurar y utilizar el agente de Amazon CloudWatch para recopilar las métricas y los registros de los nodos en lugar de utilizar el agente de AWS Systems Manager (SSM Agent) para estas tareas. El agente de CloudWatch le permite reunir más métricas de las instancias EC2 que las que están disponibles con SSM Agent. Además, puede reunir las métricas de servidores locales mediante el agente de CloudWatch.

También puede almacenar opciones de configuración del agente en Parameter Store de Systems Manager para utilizarlas con el agente de CloudWatch. Parameter Store es una herramienta de AWS Systems Manager.

> **Note**
>
> AWS Systems Manager admite la migración de SSM Agent al agente de CloudWatch unificado para recopilar registros y métricas solo en versiones de 64 bits de Windows. Para

obtener información acerca de la configuración del agente de CloudWatch unificado en otros sistemas operativos e información completa sobre su uso, consulte Recopilación de métricas y registros de instancias Amazon EC2 y en los servidores en las instalaciones con el agente de CloudWatch. en la Guía del usuario de Amazon CloudWatch.
Puede utilizar el agente de CloudWatch en otros sistemas operativos compatibles, pero no podrá utilizar Systems Manager para realizar una migración de herramientas.

El SSM Agent escribe información acerca de ejecuciones, acciones programadas, errores y estados en los archivos de registro en cada nodo. La conexión manual a un nodo para ver los archivos de registro y solucionar un problema con el SSM Agent puede llevar mucho tiempo. Para monitorear los nodos de forma más eficiente, puede configurar el propio SSM Agent o el agente de CloudWatch para que envíen estos datos de registro a los Registros de Amazon CloudWatch.

Important

El agente unificado de CloudWatch ha sustituido a SSM Agent como herramienta para enviar datos de registro a los Registros de Amazon CloudWatch. El complemento aws:cloudWatch del SSM Agent no es compatible. Recomendamos utilizar solo el agente de CloudWatch unificado para sus procesos de recopilación de registros. Para obtener más información, consulte los temas siguientes:

- Envío de registros de nodos a los Registros de CloudWatch (agente de CloudWatch) unificado
- Migrar la recopilación de registros del nodo de Windows Server al agente de CloudWatch
- Recopilación de métricas, registros y seguimientos con el agente de CloudWatch en la Guía del usuario de Amazon CloudWatch.

Con los Registros de CloudWatch, puede monitorear datos de registro en tiempo real, buscar y filtrar datos de registro creando uno o varios filtros de métricas, así como archivar y recuperar datos históricos si los necesita. Para obtener más información acerca de los Registros de CloudWatch, consulte la Guía del usuario de los Registros de Amazon CloudWatch.

La configuración de un agente para enviar datos de registro a los Registros de Amazon CloudWatch proporciona los siguientes beneficios:

- Almacenamiento centralizado de todos los archivos de registro del SSM Agent.

- Acceso más rápido a los archivos para investigar errores.
- Retención indefinida de los archivos de registro (configurable).
- Se puede realizar el mantenimiento de los registros y obtener acceso a ellos independientemente del estado del nodo.
- Acceso a otras características de CloudWatch como las métricas y las alarmas.

Para obtener más información acerca del monitoreo de la actividad de Session Manager, consulte Registro de la actividad de la sesión y Habilitar y deshabilitar el registro de la sesión.

Migrar la recopilación de registros del nodo de Windows Server al agente de CloudWatch

Si utiliza SSM Agent en nodos de Windows Server compatibles para enviar archivos de registro de SSM Agent a los Registros de Amazon CloudWatch, puede utilizar Systems Manager para cambiar el SSM Agent por el agente de CloudWatch como herramienta de recopilación de registros y migrar las opciones de configuración.

El agente de CloudWatch no se admite en las versiones de 32 bits de Windows Server.

Para las instancias EC2 de Windows Server de 64 bits, puede realizar la migración al agente de CloudWatch de forma automática o manual. Para las máquinas virtuales y los servidores locales, el proceso debe realizarse manualmente.

> **Note**
>
> Durante el proceso de migración, es posible que los datos enviados a CloudWatch se interrumpan o dupliquen. Sus métricas y datos de registro se registrarán de forma precisa otra vez en CloudWatch una vez completada la migración.

Recomendamos que se pruebe la migración en un número limitado de nodos antes de migrar una flota completa al agente de CloudWatch. Tras la migración, si prefiere recopilar los registros con el SSM Agent, puede volver a utilizarlo en su lugar.

> **Important**
>
> En los siguientes casos, no podrá migrar al agente de CloudWatch con los pasos descritos en este tema:

- La configuración existente del SSM Agent especifica varias regiones.
- La configuración existente del SSM Agent especifica varios conjuntos de credenciales de acceso/clave secreta.

En estos casos, será necesario desactivar la recopilación de registros en el SSM Agent e instalar el agente de CloudWatch sin realizar la migración. Para obtener más información, consulte los siguientes temas de la Guía del usuario de Amazon CloudWatch:

- Installing the CloudWatch agent (Instalación del agente de CloudWatch)
- Installing the CloudWatch agent on on-premises servers (Instalación del agente de CloudWatch en servidores locales)

Antes de empezar

Antes de empezar a migrar al agente de CloudWatch para la recopilación de registros, asegúrese de que los nodos en los que realizará la migración cumplen estos requisitos:

- El SO es una versión de 64 bits de Windows Server.
- Está instalada la versión 2.2.93.0 o versiones posteriores del SSM Agent en el nodo.
- El SSM Agent está configurado para el monitoreo del nodo.

Temas

- Migración automática al agente de CloudWatch
- Migración manual al agente de CloudWatch

Migración automática al agente de CloudWatch

Solo para las instancias EC2 para Windows Server puede utilizar la consola de AWS Systems Manager o la AWS Command Line Interface (AWS CLI) para migrar automáticamente al agente de CloudWatch como su herramienta de recopilación de registros.

> **Note**
>
> AWS Systems Manager admite la migración de SSM Agent al agente de CloudWatch unificado para recopilar registros y métricas solo en versiones de 64 bits de Windows. Para

obtener información acerca de la configuración del agente de CloudWatch unificado en otros sistemas operativos e información completa sobre su uso, consulte Recopilación de métricas y registros de instancias Amazon EC2 y en los servidores en las instalaciones con el agente de CloudWatch. en la Guía del usuario de Amazon CloudWatch.
Puede utilizar el agente de CloudWatch en otros sistemas operativos compatibles, pero no podrá utilizar Systems Manager para realizar una migración de herramientas.

Si la migración se ha realizado correctamente, compruebe sus resultados en CloudWatch para asegurarse de que recibe las métricas, los registros o los registros de eventos de Windows esperados. Si está satisfecho con los resultados, puede Almacenar la configuración del agente de CloudWatch en Parameter Store de manera opcional. Si la migración no se realiza correctamente o los resultados no son los esperados, puede intentar Revertir a la recopilación de registros con SSM Agent.

Note

Si desea migrar un archivo de configuración de origen que incluye una entrada `{hostname}`, debe tener en cuenta que la entrada `{hostname}` puede cambiar el valor del campo una vez completada la migración. Por ejemplo, digamos que la siguiente entrada `"LogStream": "{hostname}"` se asigna a un servidor llamado MiServidorDeRegistros001.

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

Después de la migración, esta entrada se asignará a un dominio, como ip-11-1-1-11.production. ExampleCompany.com. Para conservar el valor del nombre de host local, especifique `{local_hostname}` en lugar de `{hostname}`.

Para migrar de forma automática al agente de CloudWatch (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command y, a continuación, seleccione Run command (Ejecutar comando).
3. En la lista Command document (Documento de Command), elija `AmazonCloudWatch-MigrateCloudWatchAgent`.
4. En Status (Estado), elija Enabled (Habilitado).
5. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

   > **Tip**
   >
   > Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

6. En Rate control (Control de velocidad):

   - En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

     > **Note**
     >
     > Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

   - En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

7. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

   > **Note**
   >
   > Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

8. En la sección Notificaciones de SNS, seleccione la casilla de verificación Habilitar notificaciones de SNS si desea recibir notificaciones sobre el estado de ejecución de los comandos.

   Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.

9. Seleccione Ejecutar.

Para migrar de forma automática al agente de CloudWatch (AWS CLI)

- Ejecute el siguiente comando.

```
aws ssm send-command --document-name AmazonCloudWatch-MigrateCloudWatchAgent --
targets Key=instanceids,Values=ID1,ID2,ID3
```

ID1, *ID2* e *ID3* representan los ID de los nodos que desea actualizar, como i-02573cafcfEXAMPLE.

Migración manual al agente de CloudWatch

Para los nodos de Windows Server locales o las instancias EC2 para Windows Server, siga estos pasos para migrar la recopilación de registros al agente de Amazon CloudWatch de forma manual.

> **Note**
>
> Si desea migrar un archivo de configuración de origen que incluye una entrada `{hostname}`, debe tener en cuenta que la entrada `{hostname}` puede cambiar el valor del campo una vez completada la migración. Por ejemplo, digamos que la siguiente entrada `"LogStream": "{hostname}"` se asigna a un servidor llamado MiServidorDeRegistros001.
>
> ```
> {
> "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}"
>      }
> }
> ```
>
> Después de la migración, esta entrada se asignará a un dominio, como ip-11-1-1-11.production.ExampleCompany.com. Para conservar el valor del nombre de host local, especifique `{local_hostname}` en lugar de `{hostname}`.

Uno: instalar el agente de CloudWatch (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command y, a continuación, seleccione Run command (Ejecutar comando).
3. En la lista Command document (Documento de Command), elija `AWS-ConfigureAWSPackage`.
4. En Action (Acción), elija `Install`.
5. En Nombre, escriba **`AmazonCloudWatchAgent`**.

6. En Version (Versión), ingrese **`latest`** si aún no se ha proporcionado de forma predeterminada.
7. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

   > **Tip**
   >
   > Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

8. En Rate control (Control de velocidad):

   - En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

     > **Note**
     >
     > Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

   - En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

9. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

   > **Note**
   >
   > Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración

de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

10. En la sección Notificaciones de SNS, seleccione la casilla de verificación Habilitar notificaciones de SNS si desea recibir notificaciones sobre el estado de ejecución de los comandos.

    Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.

11. Seleccione Ejecutar.

Dos: actualizar el formato JSON de los datos de configuración

- Para actualizar el formato JSON de las opciones de configuración existentes para el agente de CloudWatch, utilice Run Command, una herramienta de AWS Systems Manager, o inicie sesión en el nodo directamente con una conexión de RDP para ejecutar los siguientes comandos de Windows PowerShell en el nodo, de uno en uno.

  ```
  cd ${Env:ProgramFiles}\\Amazon\\AmazonCloudWatchAgent
  ```

  ```
  .\\amazon-cloudwatch-agent-config-wizard.exe --isNonInteractiveWindowsMigration
  ```

  {Env:ProgramFiles} representa la ubicación donde se puede encontrar el directorio de Amazon que contiene el agente de Cloudwatch, normalmente `C:\Program Files`.

Tres: configurar e iniciar el agente de CloudWatch (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command y, a continuación, seleccione Run command (Ejecutar comando).
3. En la lista Command document (Documento de Command), elija `AWS-RunPowerShellScript`.
4. En Commands (Comandos), ingrese los siguientes dos comandos.

```
cd ${Env:ProgramFiles}\Amazon\AmazonCloudWatchAgent
```

```
.\amazon-cloudwatch-agent-ctl.ps1 -a fetch-config -m ec2 -c file:config.json -s
```

{Env:ProgramFiles} representa la ubicación donde se puede encontrar el directorio de Amazon que contiene el agente de Cloudwatch, normalmente `C:\Program Files`.

5. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

> **Tip**
>
> Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

6. En Rate control (Control de velocidad):

   - En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

   > **Note**
   >
   > Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

   - En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

7. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

8. En la sección Notificaciones de SNS, seleccione la casilla de verificación Habilitar notificaciones de SNS si desea recibir notificaciones sobre el estado de ejecución de los comandos.

   Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.

9. Seleccione Ejecutar.

Cuatro: desactivar la recopilación de registros en SSM Agent (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command y, a continuación, seleccione Run command (Ejecutar comando).
3. En la lista Command document (Documento de Command), elija `AWS-ConfigureCloudWatch`.
4. En Status (Estado), elija Disabled (Desactivado).
5. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

> **Tip**
>
> Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

6. En Status (Estado), elija `Disabled` (Habilitado).
7. En Rate control (Control de velocidad):
   - En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

     > **Note**
     >
     > Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

   - En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.
8. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el

rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

9. En la sección Notificaciones de SNS, seleccione la casilla de verificación Habilitar notificaciones de SNS si desea recibir notificaciones sobre el estado de ejecución de los comandos.

   Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.
10. Seleccione Ejecutar.

    Una vez completados estos pasos, compruebe sus registros en CloudWatch para verificar que recibe las métricas, los registros o los registros de eventos de Windows esperados. Si los resultados son satisfactorios, puede Almacenar la configuración del agente de CloudWatch en Parameter Store de manera opcional. Si la migración no se realiza correctamente o los resultados no son los esperados, puede Revertir a la recopilación de registros con SSM Agent.

Almacenar la configuración del agente de CloudWatch en Parameter Store

Puede almacenar el contenido de un archivo de configuración de un agente de CloudWatch en Parameter Store. Al mantener estos datos de configuración en un parámetro, varios nodos pueden obtener sus opciones de configuración de él, evitándose además el hecho de tener que crear o actualizar manualmente archivos de configuración en sus nodos. Por ejemplo, puede utilizar Run Command para escribir el contenido del parámetro en los archivos de configuración de varios nodos, o bien utilizar State Manager, una herramienta de AWS Systems Manager, para ayudar a evitar la desviación de la configuración en las opciones de configuración del agente de CloudWatch en una flota de nodos.

Cuando ejecuta el asistente de configuración del agente de CloudWatch, puede elegir permitir que el asistente guarde sus opciones de configuración como nuevo parámetro en Parameter Store. Para obtener información acerca de cómo se ejecuta el asistente de configuración del agente de CloudWatch, consulte Create the CloudWatch agent configuration file with the wizard (Creación del archivo de configuración del agente de CloudWatch con el asistente) en la Guía del usuario de Amazon CloudWatch.

Si ejecutó el asistente, pero no eligió la opción para guardar la configuración como parámetro, o bien creó el archivo de configuración del agente de CloudWatch manualmente, puede recuperar los datos para guardar como parámetro en su nodo en el siguiente archivo.

```
${Env:ProgramFiles}\Amazon\AmazonCloudWatchAgent\config.json
```

{Env:ProgramFiles} representa la ubicación donde se puede encontrar el directorio de Amazon que contiene el agente de Cloudwatch, normalmente `C:\Program Files`.

Recomendamos mantener una copia de seguridad del código JSON de este archivo en una ubicación distinta del propio nodo.

Para obtener información acerca de cómo crear un parámetro, consulte Creación de parámetros de Parameter Store en Systems Manager.

Para obtener más información acerca de su agente de CloudWatch, consulte Collecting metrics and logs from Amazon EC2 instances and on-premises servers with the CloudWatch agent (Recopilación de métricas y registros de instancias Amazon EC2 y servidores locales con el agente de CloudWatch) en la Guía del usuario de Amazon CloudWatch.

Revertir a la recopilación de registros con SSM Agent

Si desea volver a usar el SSM Agent para recopilar registros, siga estos pasos.

Uno: recuperar los datos de configuración del SSM Agent

1. En el nodo en el que desea volver a recopilar registros con el SSM Agent, encuentre el contenido del archivo de configuración del SSM Agent. Este archivo JSON suele encontrarse en la siguiente ubicación:

   ```
   ${Env:ProgramFiles}\\Amazon\\SSM\\Plugins\\awsCloudWatch\
   \AWS.EC2.Windows.CloudWatch.json
   ```

   {Env:ProgramFiles} representa la ubicación donde se puede encontrar el directorio `Amazon`, normalmente `C:\Program Files`.
2. Copie estos datos en un archivo de texto para su uso en un paso posterior.

   Recomendamos almacenar una copia de seguridad del código JSON en una ubicación distinta del propio nodo.

Dos: desinstalar el agente de CloudWatch (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.

2. En el panel de navegación, elija Run Command y, a continuación, seleccione Run command (Ejecutar comando).
3. En la lista Command document (Documento de Command), elija `AWS-ConfigureAWSPackage`.
4. En Action (Acción), elija Uninstall (Desinstalar).
5. En Nombre, escriba **`AmazonCloudWatchAgent`**.
6. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

   > **Tip**
   >
   > Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

7. En Rate control (Control de velocidad):

   - En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

     > **Note**
     >
     > Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

   - En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

8. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

9. En la sección Notificaciones de SNS, seleccione la casilla de verificación Habilitar notificaciones de SNS si desea recibir notificaciones sobre el estado de ejecución de los comandos.

   Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.

10. Seleccione Ejecutar.

Tres: para volver a activar la recopilación de registros en SSM Agent (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command y, a continuación, seleccione Run command (Ejecutar comando).
3. En la lista Command document (Documento de Command), elija `AWS-ConfigureCloudWatch`.
4. En Status (Estado), elija `Enabled` (Habilitado).
5. En Properties (Propiedades), pegue el contenido de los datos de configuración antiguos guardados en el archivo de texto.
6. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

> **Tip**
>
> Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

7. En Rate control (Control de velocidad):

   - En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

     > **Note**
     >
     > Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

   - En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

8. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

   > **Note**
   >
   > Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el

rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

9. En la sección Notificaciones de SNS, seleccione la casilla de verificación Habilitar notificaciones de SNS si desea recibir notificaciones sobre el estado de ejecución de los comandos.

   Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.

10. Seleccione Ejecutar.

Envío de registros de SSM Agent a CloudWatch Logs

AWS Systems Manager Agent (SSM Agent) es el software de Amazon que se ejecuta en sus instancias de EC2, dispositivos periféricos, servidores en las instalaciones y máquinas virtuales (VM) que están configuradas para Systems Manager. SSM Agent procesa solicitudes del servicio Systems Manager en la nube y configura la máquina tal como se especifica en cada solicitud. Para obtener más información acerca de SSM Agent, consulte Uso de SSM Agent.

Además, siguiendo estos pasos, puede configurar el SSM Agent para enviar datos de registro a Amazon CloudWatch Logs.

Antes de empezar

Cree un grupo de registros de CloudWatch Logs. Para obtener más información, consulte Getting started with CloudWatch Logs (Introducción a CloudWatch Logs) en la Guía del usuario de Amazon CloudWatch Logs.

Para configurar SSM Agent para enviar los registros a CloudWatch

1. Inicie sesión en un nodo y busque el archivo siguiente:

   Linux

   En la mayoría de los tipos de nodos Linux: `/etc/amazon/ssm/seelog.xml.template`.

   En Ubuntu Server 20.10 STR y 20.04, 18.04 y 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. Cambie el nombre del archivo de `seelog.xml.template` a `seelog.xml`.

> **Note**
>
> En Ubuntu Server 20.10 STR y 20.04, 18.04 y 16.04 LTS, el archivo `seelog.xml` se debe crear en el directorio `/etc/amazon/ssm/`. Ejecute los siguientes comandos para crear este directorio y archivo.
>
> ```
> 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. Abra el archivo `seelog.xml` en un editor de texto y localice la siguiente sección:

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. Edite el archivo y agregue un elemento de nombre personalizado después de la etiqueta de cierre **<**/filter**>**. En el ejemplo siguiente, el nombre personalizado se ha especificado como `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. Guarde los cambios y reinicie el SSM Agent o el nodo.
6. Abra la consola de CloudWatch en https://console.aws.amazon.com/cloudwatch/.

7. En el panel de navegación, elija Grupos de registro, y, a continuación, elija el nombre del grupo de registro.

> **Tip**
>
> La transmisión de registros de los datos de archivos de registro del SSM Agent está organizada por ID de nodo.

Supervisión de los eventos de las solicitudes de cambio

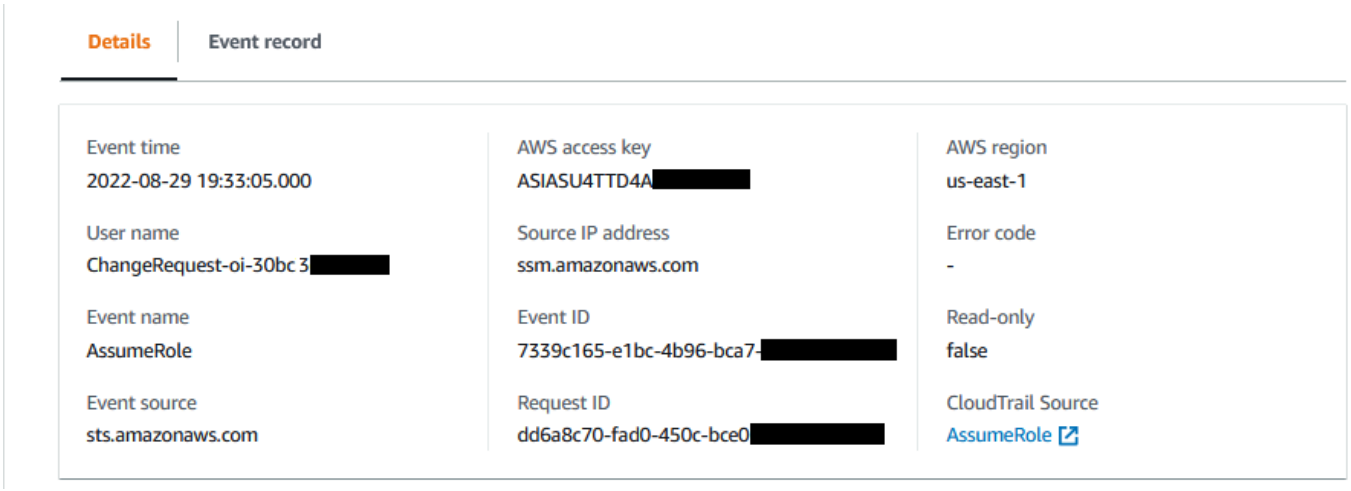
Después de activar la integración con AWS CloudTrail Lake y crear un almacén de datos de eventos, puede ver detalles auditables sobre las solicitudes de cambio que se ejecutan en su cuenta u organización. Esto incluye detalles como los siguientes:

- La identidad del usuario que inició la solicitud de cambio
- La Regiones de AWS donde se hicieron los cambios
- La dirección IP de origen de la solicitud
- La clave de acceso de AWS que se utilizó para la solicitud
- Las acciones de la API que se ejecutan para la solicitud de cambio
- Los parámetros de solicitud que se incluyen para esas acciones
- Los recursos actualizados durante el proceso

Las siguientes son muestras de detalles de eventos que puede ver para una solicitud de cambio después de crear el almacén de datos de eventos en AWS CloudTrail Lake.

Details

La siguiente imagen muestra la información general sobre una solicitud de cambio disponible en la pestaña Details (Detalles). Estos detalles incluyen información como la hora a la que se inició la operación de solicitud de cambio, el ID del usuario que inició la solicitud de cambio, la Región de AWS afectada, así como el ID de evento y el ID de solicitud asociados a la solicitud.



Event record

La siguiente imagen muestra la estructura del contenido JSON que CloudTrail Lake proporciona para un evento de solicitud de cambio. Estos datos se proporcionan en la pestaña Event record (Registro de eventos) de una solicitud de cambio.



> **Important**
>
> Si utiliza Change Manager para una organización, puede completar el siguiente procedimiento mientras inicia sesión en la cuenta de administración o en la cuenta de administrador delegada para Change Manager.
> Sin embargo, para utilizar la cuenta de administrador delegada para completar estos pasos, se debe especificar la misma cuenta de administrador delegado tanto para CloudTrail como para Change Manager.
> Al iniciar sesión en la cuenta de administración para Change Manager, puede añadir o cambiar la cuenta de administrador delegado para CloudTrail en la página Settings (Configuración) de CloudTrail. Esto debe hacerse antes de que la cuenta de administrador delegado pueda crear un almacén de datos de evento para que lo utilice toda la organización.

Para activar el seguimiento de eventos de CloudTrail Lake desde Change Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Change Manager.
3. Elija la pestaña Requests (Solicitudes).
4. Elija cualquier solicitud de cambio existente y luego elija la pestaña Associated events (Eventos asociados).
5. Elija Enable CloudTrail Lake (habilitar CloudTrail Lake).
6. Siga los pasos de Crear un almacén de datos de eventos para eventos de CloudTrail en la Guía del usuario de AWS CloudTrail.

   Para garantizar que se almacenen los datos de eventos para sus solicitudes de cambios, realice las siguientes selecciones a medida que completa el procedimiento:

   - Para Tipo de evento, deje los Eventos de AWS predeterminados y los Eventos de CloudTrail seleccionados.
   - Si usa Change Manager con una organización, seleccione Habilitar para todas las cuentas de mi organización.
   - Para los Eventos de gestión, no desmarque la casilla de verificación Escribir.

Otras opciones que elija cuando crea su almacén de datos de eventos no afectan el almacenamiento de datos de eventos para sus solicitudes de cambio.

Monitoreo de las automatizaciones

Las métricas son el concepto fundamental en Amazon CloudWatch. Una métrica representa una serie de puntos de datos ordenados por tiempo que se publican a CloudWatch. Considere una métrica como una variable que hay que monitorear y los puntos de datos como los valores de esa variable a lo largo del tiempo.

Automatización es una herramienta de AWS Systems Manager. Systems Manager publica métricas sobre el uso de Automation en CloudWatch. Esto le permite establecer alarmas basadas en esas métricas.

Para ver las métricas de Automation en la consola de CloudWatch

1. Abra la consola de CloudWatch en https://console.aws.amazon.com/cloudwatch/.
2. En el panel de navegación, seleccione Métricas.
3. Elija SSM.
4. En la pestaña Metrics (Métricas), elija Usage (Uso) y, a continuación, seleccione By AWS resource (Por recurso de AWS).
5. En el cuadro de búsqueda situado junto a la lista de métricas, ingrese SSM.

Para ver las métricas de Automation mediante la AWS CLI

Abra el símbolo del sistema y utilice el siguiente comando.

```
aws cloudwatch list-metrics \
    --namespace "AWS/Usage"
```

Métricas de Automation

Systems Manager envía las siguientes métricas de Automation a CloudWatch.

Métrica	Descripción
`ConcurrentAutomationUsage`	La cantidad de automatizaciones que se ejecutan al mismo tiempo en la Cuenta de AWS y la Región de AWS actuales.
`QueuedAutomationUsage`	La cantidad de automatizaciones actualmente en cola que no se han iniciado y tienen un estado de `Pending` (Pendiente).

Para obtener más información acerca de cómo trabajar con métricas de CloudWatch, consulte los siguientes temas en la Guía del usuario de Amazon CloudWatch:

- Métricas
- Uso de métricas de Amazon CloudWatch
- Uso de las alarmas de Amazon CloudWatch

Monitoreo de métricas de Run Command con Amazon CloudWatch

Las métricas son el concepto fundamental en Amazon CloudWatch. Una métrica representa una serie de puntos de datos ordenados por tiempo que se publican a CloudWatch. Una métrica es una variable que hay que monitorizar y los puntos de datos son los valores de esa variable a lo largo del tiempo.

AWS Systems Manager publica métricas sobre el estado de los comandos de Run Command en CloudWatch, lo que le permite establecer alarmas basadas en esas métricas. Run Command es una herramienta de AWS Systems Manager. Estas estadísticas se registran a lo largo de un periodo prolongado para que pueda obtener acceso a información histórica y contar con una mejor perspectiva sobre la tasa de éxito de los comandos ejecutados en su Cuenta de AWS.

Los valores de estado terminal de los comandos para cuyas métricas puede realizar un seguimiento son `Success`, `Failed` y `Delivery Timed Out`. Por ejemplo, para un documento de SSM Command configurado para que se ejecute cada hora, puede configurar una alarma que le avise cada vez que no se notifique el estado `Success` para cualquiera de esas horas. Para obtener más información acerca de los valores de estado de los comandos, consulte Descripción de los estados del comando.

Para ver las métricas en la consola de CloudWatch

1. Abra la consola de CloudWatch en https://console.aws.amazon.com/cloudwatch/.
2. En el panel de navegación, seleccione Métricas.
3. En el área Alarmas por servicio de AWS, elija SSM-Run Command para Servicios.

Para ver métricas mediante la AWS CLI

Abra el símbolo del sistema y utilice el siguiente comando.

```
aws cloudwatch list-metrics --namespace "AWS/SSM-RunCommand"
```

Para mostrar todas las métricas disponibles, utilice el siguiente comando.

```
aws cloudwatch list-metrics
```

Métricas y dimensiones de Systems Manager Run Command

Systems Manager envía métricas de comandos de Run Command a CloudWatch una vez cada minuto.

Systems Manager envía las siguientes métricas de comandos a CloudWatch.

> **Note**
>
> Estas métricas utilizan `Count` como unidad, por lo que las estadísticas más útiles son `Sum` y `SampleCount`.

Métrica	Descripción
`CommandsDeliveryTimedOut`	Número de comandos cuyo estado de terminal es `Delivery Timed Out`.
`CommandsFailed`	Número de comandos cuyo estado de terminal es `Failed`.

Métrica	Descripción
CommandsSucceeded	Número de comandos cuyo estado de terminal es Success.

Para obtener más información acerca de cómo trabajar con métricas de CloudWatch, consulte los siguientes temas en la Guía del usuario de Amazon CloudWatch:

- Métricas
- Uso de métricas de Amazon CloudWatch
- Uso de las alarmas de Amazon CloudWatch

Registro de llamadas a la API de AWS Systems Manager con AWS CloudTrail

AWS Systems Manager se integra con AWS CloudTrail, un servicio que proporciona un registro de las acciones realizadas por un usuario, un rol o un Servicio de AWS. CloudTrail captura todas las llamadas a la API de Systems Manager como eventos. Las llamadas capturadas incluyen las llamadas desde la consola de Systems Manager y las llamadas desde el código a las operaciones de la API de Systems Manager. Mediante la información recopilada por CloudTrail, puede determinar la solicitud que se realizó en Systems Manager, la dirección IP desde la que se realizó, cuándo se realizó y detalles adicionales.

Cada entrada de registro o evento contiene información sobre quién generó la solicitud. La información de identidad del usuario le ayuda a determinar lo siguiente:

- Si la solicitud se realizó con las credenciales del usuario raíz o del usuario.
- Si la solicitud se realizó en nombre de un usuario de IAM Identity Center.
- Si la solicitud se realizó con credenciales de seguridad temporales de un rol o fue un usuario federado.
- Si la solicitud la realizó otro Servicio de AWS.

CloudTrail está activado en la Cuenta de AWS cuando usted crea la cuenta y tiene acceso automático al Historial de eventos de CloudTrail. El Historial de eventos de CloudTrail proporciona un registro visible e inmutable, que se puede buscar y descargar, de los últimos 90 días de eventos de

gestión registrados en una Región de AWS. Para obtener más información, consulte Trabajar con el historial de eventos de CloudTrail en la Guía del usuario de AWS CloudTrail. No se cobran cargos de CloudTrail por ver el Historial de eventos.

Para mantener un registro permanente de los eventos en su Cuenta de AWS más allá de los 90 días, cree un registro de seguimiento o un almacén de datos de eventos de CloudTrail Lake.

Registros de seguimiento de CloudTrail

Un registro de seguimiento permite a CloudTrail enviar archivos de registro a un bucket de Amazon S3. Todos los registros de seguimiento que cree con la AWS Management Console son de varias regiones. Puede crear un registro de seguimiento de una sola región o de varias regiones mediante la AWS CLI. Se recomienda crear un registro de seguimiento de varias regiones, ya que registra actividad en todas las Regiones de AWS de su cuenta. Si crea un registro de seguimiento de una sola región, solo podrá ver los eventos registrados en la Región de AWS del registro de seguimiento. Para obtener más información acerca de los registros de seguimiento, consulte Creación de un registro de seguimiento para su Cuenta de AWS y Creación de un registro de seguimiento para una organización en la Guía del usuario de AWS CloudTrail.

Puede crear un registro de seguimiento para enviar una copia de los eventos de administración en curso en su bucket de Amazon S3 sin costo alguno desde CloudTrail; sin embargo, hay cargos por almacenamiento en Amazon S3. Para obtener más información sobre los precios de CloudTrail, consulte Precios de AWS CloudTrail. Para obtener información acerca de los precios de Amazon S3, consulte Precios de Amazon S3.

Almacenes de datos de eventos de CloudTrail Lake

CloudTrail Lake le permite ejecutar consultas basadas en SQL sobre los eventos. CloudTrail Lake convierte los eventos existentes en formato JSON basado en filas al formato ORC de Apache. ORC es un formato de almacenamiento en columnas optimizado para una recuperación rápida de datos. Los eventos se agregan en almacenes de datos de eventos, que son recopilaciones inmutables de eventos en función de criterios que se seleccionan aplicando selectores de eventos avanzados. Los selectores que se aplican a un almacén de datos de eventos controlan los eventos que perduran y están disponibles para la consulta. Para obtener más información acerca de CloudTrail Lake, consulte Trabajar con AWS CloudTrail Lake en la Guía del usuario de AWS CloudTrail.

Los almacenes de datos de eventos de CloudTrail Lake y las consultas generan costos adicionales. Cuando crea un almacén de datos de eventos, debe elegir la opción de precios

que desee utilizar para él. La opción de precios determina el costo de la incorporación y el almacenamiento de los eventos, así como el período de retención predeterminado y máximo del almacén de datos de eventos. Para obtener más información sobre los precios de CloudTrail, consulte Precios de AWS CloudTrail.

Eventos de datos de Systems Manager en CloudTrail

Los eventos de datos ofrecen información sobre las operaciones de recursos realizadas en o dentro de un recurso (por ejemplo, la creación o la apertura de un canal de control). Se denominan también operaciones del plano de datos. Los eventos de datos suelen ser actividades de gran volumen. De forma predeterminada, CloudTrail no registra eventos de datos. El Historial de eventos de CloudTrail no registra los eventos de datos.

Se aplican cargos adicionales a los eventos de datos. Para obtener más información sobre los precios de CloudTrail, consulte Precios de AWS CloudTrail.

Puede registrar eventos de datos para los tipos de recursos de Systems Manager con la consola de CloudTrail, con la AWS CLI o con las operaciones de la API de CloudTrail. Para obtener más información sobre cómo registrar los eventos de datos, consulte Registro de eventos de datos con la AWS Management Console y Registro de eventos de datos con la AWS Command Line Interface en la Guía del usuario de AWS CloudTrail.

En la siguiente tabla se muestran los tipos de recursos de Systems Manager para los que puede registrar eventos de datos. La columna Tipo de evento de datos (consola) muestra el valor que se debe elegir en la lista de tipos de eventos de datos de la consola de CloudTrail. La columna resources.type value muestra el valor de `resources.type`, que especificaría al configurar los selectores de eventos avanzados mediante la AWS CLI o las API de CloudTrail. La columna API de datos registradas en CloudTrail muestra las llamadas a la API registradas en CloudTrail para el tipo de recurso.

Tipo de evento de datos (consola)	resources.type value	API de datos registradas en CloudTrail
Systems Manager	`AWS::SSMMessages::ControlChannel`	• `CreateControlChannel` • `OpenControlChannel`

Tipo de evento de datos (consola)	resources.type value	API de datos registradas en CloudTrail
		Para obtener más información sobre estas operaciones, consulte Acciones definidas por Amazon Message Gateway Service en la Referencia de autorizaciones de servicios.
Nodo administrado de Systems Manager	`AWS::SSM::ManagedNode`	• `RequestManagedInstanceRoleToken` : este evento se genera cuando el agente de AWS Systems Manager (SSM Agent) que se ejecuta en un nodo administrado de Systems Manager solicita credenciales al servicio de credenciales de Systems Manager. Para obtener más información acerca de la operación `RequestManagedInstanceRoleToken` , consulte Validación de equipos activados de manera híbrida mediante una huella digital de hardware.

Puede configurar selectores de eventos avanzados para filtrar según los campos `eventName`, `readOnly` y `resources.ARN` y así registrar solo los eventos que son importantes para usted. Para obtener más información acerca de estos campos, consulte AdvancedFieldSelector en la Referencia de la API de AWS CloudTrail.

Eventos de administración de Systems Manager en CloudTrail

Los eventos de administración proporcionan información sobre las operaciones de administración que se realizan en los recursos de su Cuenta de AWS. Se denominan también operaciones del plano de control. CloudTrail registra los eventos de administración de forma predeterminada.

Systems Manager registra todas las operaciones del plano de control en CloudTrail como eventos de administración. Las operaciones de la API de Systems Manager se documentan en la Referencia de la API de AWS Systems Manager. Por ejemplo, las llamadas a las acciones `CreateMaintenanceWindows`, `PutInventory`, `SendCommand` y `StartSession` generan entradas en los archivos de registros de CloudTrail. Para ver un ejemplo de cómo se configura CloudTrail para monitorear una llamada a la API de Systems Manager, consulte Supervisión de la actividad de la sesión con Amazon EventBridge (consola).

Ejemplos de eventos de Systems Manager

Un evento representa una única solicitud de cualquier origen e incluye información sobre la operación de la API solicitada, la fecha y la hora de la operación o los parámetros de la solicitud, entre otras cosas. Los archivos de registro de CloudTrail no rastrean el orden en la pila de las llamadas a la API públicas, por lo que los eventos no aparecen en un orden específico.

Ejemplos:

- Ejemplos de eventos de administración
- Ejemplos de eventos de datos

Ejemplos de eventos de administración

Ejemplo 1: **`DeleteDocument`**

En el siguiente ejemplo, se muestra un evento de CloudTrail que ilustra la operación `DeleteDocument` en un documento denominado `example-Document` en la región Este de EE. UU. (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"
}
```


Ejemplo 2: **`StartConnection`**

En el ejemplo siguiente, se muestra una evento de CloudTrail para un usuario que inicia una conexión RDP mediante Fleet Manager en la región Este de EE. UU. (Ohio) (us-east-2). La acción de la API subyacente es `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"
}
```

Ejemplos de eventos de datos

Ejemplo 1: **`CreateControlChannel`**

En el ejemplo que sigue se muestra un evento de CloudTrail que ilustra la operación `CreateControlChannel`.

```
{
  "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"
}
```


Ejemplo 2: **`RequestManagedInstanceRoleToken`**

En el ejemplo que sigue se muestra un evento de CloudTrail que ilustra la operación `RequestManagedInstanceRoleToken`.


```
{
    "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"
```

```
}
```

Para obtener información sobre el contenido de los registros de CloudTrail, consulte Contenido de los registros de CloudTrail en la Guía del usuario de AWS CloudTrail.

Registro de salida de acción de Automation con CloudWatch Logs

Automatización, una herramienta de AWS Systems Manager, se integra con Registros de Amazon CloudWatch. Puede enviar la salida desde las acciones de `aws:executeScript` en los manuales de procedimientos hacia el grupo de registros que especifique. Systems Manager no crea grupos ni flujos de registro para los documentos que no utilizan acciones de `aws:executeScript`. Si el documento utiliza `aws:executeScript`, la salida enviada a CloudWatch Logs solo pertenece a esas acciones. Puede utilizar la salida de acción `aws:executeScript` almacenada en el grupo de registros de CloudWatch Logs para depurar y resolver problemas. Si elige un grupo de registros que está cifrado, la salida de acción `aws:executeScript` también está cifrada. La salida de registro desde las acciones de `aws:executeScript` es una configuración del nivel de cuenta.

A fin de enviar la salida de la acción a Registros de CloudWatch para los manuales de procedimientos de propiedad de Amazon, el rol o el usuario que ejecuta la automatización debe tener permisos para las siguientes operaciones:

- `logs:CreateLogGroup`
- `logs:CreateLogStream`
- `logs:DescribeLogGroups`
- `logs:DescribeLogStreams`
- `logs:PutLogEvents`

Para los manuales de procedimientos de los que usted es propietario, se deben agregar los mismos permisos al rol de servicio de IAM (o AssumeRole) que se utiliza para ejecutar el manual de procedimientos.

Para enviar la salida de acción a CloudWatch Logs (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Automation.
3. Elija la pestaña Preferences (Preferencias) y, a continuación, Edit (Editar).

4. Elija la casilla de verificación situada junto a Send output to CloudWatch Logs (Enviar salida a CloudWatch Logs).
5. (Recomendado) Elija la casilla de verificación situada junto a Encrypt log data (Cifrar datos de registro). Con esta opción activada, los datos de registro se cifran con la clave de cifrado del lado del servidor especificado para el grupo de registros. Si no desea cifrar los datos de registro que se envían a CloudWatch Logs, desactive la casilla de verificación. Desactive la casilla de verificación si no se permite el cifrado en el grupo de registros.
6. En CloudWatch Logs log group (Grupo de registros de CloudWatch Logs), para especificar el grupo de registros de CloudWatch Logs existente en la Cuenta de AWS a la que desea que se envíe una salida de acción, seleccione una de las opciones siguientes:
   - Send output to the default log group (Enviar salida al grupo de registros predeterminado): si el grupo de registro predeterminado no existe (`/aws/ssm/automation/executeScript`), la capacidad de Automation lo crea por usted.
   - Choose from a list of log groups (Elegir de una lista de grupos de registro): seleccione un grupo de registros que ya se haya creado en su cuenta para almacenar la salida de acción.
   - Enter a log group name (Escribir un nombre del grupo de registros): escriba el nombre de un grupo de registros que ya se haya creado en su cuenta para almacenar la salida de acción.
7. Seleccione Guardar.

Para enviar la salida de acción a CloudWatch Logs (línea de comando)

1. Abra su herramienta de línea de comandos preferida y ejecute el siguiente comando para actualizar el destino de salida de la acción.

   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"
```

No se obtienen resultados si el comando se ejecuta correctamente.

2. Ejecute el siguiente comando para especificar el grupo de registros al que desea enviar la salida de acción.

   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"
   ```

   No se obtienen resultados si el comando se ejecuta correctamente.

3. Ejecute el siguiente comando para ver la configuración del servicio actual para las preferencias de registro de acciones de Automation en la Cuenta de AWS y la Región de AWS actuales.

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"
```

El comando devuelve información similar a la siguiente.

```
{
    "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"
    }
}
```

Configuración de Registros de Amazon CloudWatch para Run Command

Cuando se envía un comando mediante Run Command, una herramienta de AWS Systems Manager, es posible especificar la ubicación a la que se enviará la salida del comando. De manera predeterminada, Systems Manager devuelve únicamente los 24 000 primeros caracteres de la salida del comando. Si desea ver todos los detalles de la salida del comando, puede especificar un bucket de Amazon Simple Storage Service (Amazon S3). También puede especificar Registros de Amazon CloudWatch. Si especifica Registros de CloudWatch, Run Command enviará periódicamente la totalidad de la salida de los comandos y los registros de errores a Registros de CloudWatch. Puede monitorear los logs de salida prácticamente en tiempo real, buscar frases, valores o patrones específicos y crear alarmas en función de la búsqueda.

Si ha configurado el nodo para que utilicen las políticas administradas de AWS Identity and Access Management (IAM), `AmazonSSMManagedInstanceCore` y `CloudWatchAgentServerPolicy`, el nodo no requiere ninguna configuración adicional para enviar la salida a Registros de CloudWatch. Elija esta opción si se van a enviar comandos desde la consola o bien agregue la sección `cloud-watch-output-config` y el parámetro `CloudWatchOutputEnabled` si se va a utilizar la AWS Command Line Interface, (AWS CLI), AWS Tools for Windows PowerShell o una operación de la API. La sección `cloud-watch-output-config` y el parámetro `CloudWatchOutputEnabled` se describen con más detalle más adelante en este tema.

Para obtener información sobre cómo agregar políticas a un perfil de instancia para instancias de EC2, consulte Configuración de permisos de instancia requeridos para Systems Manager. Para obtener información sobre cómo agregar políticas a un rol de servicio para servidores locales y máquinas virtuales que tenga previsto utilizar como nodos administrados, consulte Creación del rol de servicio de IAM requerido para Systems Manager en entornos híbridos y multinube.

Si va a utilizar una política personalizada en los nodos, deberá actualizar la política en cada nodo para permitir que Systems Manager envíe la salida y los registros a Registros de CloudWatch. Agregue los objetos de política siguientes a la política personalizada. Para obtener más información acerca de la actualización de políticas de IAM, consulte la Edición de políticas de IAM en la Guía del usuario de 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/*"
},
```

Especificación de Registros de CloudWatch al momento de enviar comandos

Para especificar Registros de CloudWatch como salida cuando envíe un comando desde la AWS Management Console, elija CloudWatch Output (Salida de CloudWatch) en la sección Output options (Opciones de salida). Si lo prefiere, puede especificar el nombre del grupo de Registros de CloudWatch al que desea enviar la salida del comando. Si no especifica un nombre de grupo, Systems Manager crea automáticamente un grupo de registros. El grupo de registro utiliza un nombre con el siguiente formato: `/aws/ssm/`*`SystemsManagerDocumentName`*

Si ejecuta comandos mediante la AWS CLI, especifique la sección `cloud-watch-output-config` en el comando. En esta sección podrá especificar el parámetro `CloudWatchOutputEnabled` y, si lo desea, el parámetro `CloudWatchLogGroupName`. A continuación se muestra un ejemplo.

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}'
```

Visualización de la salida de comandos en Registros de CloudWatch

Tan pronto como el comando comienza a ejecutarse, Systems Manager envía la salida a Registros de CloudWatch prácticamente en tiempo real. La salida tiene el siguiente formato en Registros de CloudWatch:

CommandID/*InstanceID*/*PluginID*/stdout

CommandID/*InstanceID*/*PluginID*/stderr

La salida de la ejecución se carga cada 30 segundos o cada vez que el búfer contiene más de 200 KB, lo que suceda antes.

> **Note**
>
> Los flujos de registro únicamente se crean si hay datos de salida disponibles. Por ejemplo, si no hay datos de error para una ejecución, el flujo stderr no se crea.

A continuación, se ofrece un ejemplo de la salida de comandos tal y como se muestra en Registros de CloudWatch.

```
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
```

Monitoreo de eventos de Systems Manager con Amazon EventBridge

Amazon EventBridge es un servicio de bus de eventos sin servidor que le permite conectar sus aplicaciones con datos de varios orígenes. EventBridge proporciona un flujo de datos en tiempo real desde sus propias aplicaciones, aplicaciones de software como servicio (SaaS) y Servicios de AWS y, a continuación, dirige dichos datos a destinos como AWS Lambda. Puede configurar reglas de direccionamiento para determinar adónde enviar sus datos a fin de crear arquitecturas de

aplicaciones que reaccionen en tiempo real a todos sus orígenes de datos. EventBridge le permite crear arquitecturas impulsadas por eventos, que están acopladas y distribuidas débilmente.

Anteriormente, EventBridge se llamaba Amazon CloudWatch Events. EventBridge incluye nuevas características que le permiten recibir eventos de socios de SaaS y de sus propias aplicaciones. Los usuarios existentes de CloudWatch Events pueden obtener acceso a su bus, reglas y eventos predeterminados existentes en la nueva consola de EventBridge y en la consola de CloudWatch Events. EventBridge utiliza la misma API de CloudWatch Events, por lo que todo el uso de la API de CloudWatch Events existente sigue siendo el mismo.

EventBridge puede agregar eventos de docenas de Servicios de AWS a sus reglas y destinos de más de 20 Servicios de AWS.

EventBridge proporciona soporte para eventos de AWS Systems Manager y destinos de Systems Manager.

Tipos de eventos admitidos de Systems Manager

Entre los muchos tipos de eventos de Systems Manager que EventBridge puede detectar se encuentran los siguientes:

- Una actualización del estado de la solicitud de acceso a los nodos justo a tiempo para aprobaciones manuales
- Una solicitud de acceso a los nodos justo a tiempo fallida
- Una ventana de mantenimiento se desactiva.
- Un flujo de trabajo de Automation finaliza correctamente. Automatización es una herramienta de AWS Systems Manager.
- Un nodo administrado no cumple con la conformidad de revisiones.
- Un valor de parámetro se actualiza.

EventBridge admite eventos de las siguientes herramientas de AWS Systems Manager:

- Acceso a los nodos justo a tiempo (los eventos se emiten en la medida de lo posible)
- Automation (los eventos se emiten en la medida de lo posible).
- Change Calendar (Los eventos se emiten en la medida de lo posible).
- Conformidad
- Inventory (Los eventos se emiten en la medida de lo posible).

- Maintenance Windows (Los eventos se emiten en la medida de lo posible).
- Parameter Store (Los eventos se emiten en la medida de lo posible).
- Run Command (Los eventos se emiten en la medida de lo posible).
- State Manager (Los eventos se emiten en la medida de lo posible).

Para obtener más detalles acerca de los tipos de eventos de Systems Manager admitidos, consulte Referencia: patrones y tipos de eventos de Amazon EventBridge para Systems Manager y Ejemplos de eventos de Amazon EventBridge para Systems Manager.

Tipos de destinos admitidos de Systems Manager

EventBridge admite las siguientes tres herramientas de Systems Manager como destinos de una regla de evento:

- ejecución de un flujo de trabajo de Automation
- ejecución de un documento de Command de Run Command (Los eventos se emiten en la medida de lo posible).
- creación de un OpsCenter OpsItem

Para obtener información acerca de las formas sugeridas de utilizar estos destinos, consulte Escenarios de ejemplo: destinos de Systems Manager en reglas de Amazon EventBridge.

Para obtener más información acerca de cómo comenzar a utilizar EventBridge y configurar las reglas, consulte Getting started with Amazon EventBridge (Introducción a Amazon EventBridge) en la Guía del usuario de EventBridge. Para obtener información completa acerca de cómo trabajar con EventBridge, consulte la Guía del usuario de Amazon EventBridge.

Temas

- Configuración de EventBridge para eventos de Systems Manager
- Ejemplos de eventos de Amazon EventBridge para Systems Manager
- Escenarios de ejemplo: destinos de Systems Manager en reglas de Amazon EventBridge

Configuración de EventBridge para eventos de Systems Manager

Puede utilizar Amazon EventBridge para realizar un evento de destino cuando se produzcan cambios de estado u otras condiciones de AWS Systems Manager compatibles. Tiene la opción de crear una

regla que se ejecute siempre que haya una transición de estado o cuando haya una transición a uno o varios estados de interés.

El siguiente procedimiento proporciona pasos generales para crear una regla de EventBridge que interactúe cuando Systems Manager emita un evento especificado. Para obtener una lista de procedimientos en esta guía del usuario que abordan escenarios específicos, consulte More info (Más información) al final de este tema.

> **Note**
>
> Cuando un servicio en su Cuenta de AWS emite un evento, siempre va al bus de eventos predeterminado de su cuenta. Para escribir una regla que responda a eventos de Servicios de AWS de su cuenta, asóciela al bus de eventos predeterminado. Puede crear una regla en un bus de eventos personalizado que busque eventos desde Servicios de AWS, pero esta regla solo se aplica cuando recibe dicho evento desde otra cuenta mediante la entrega de eventos entre cuentas. Para obtener más información, consulte Envío y recepción de eventos de Amazon EventBridge entre Cuentas de AWS en la Guía del usuario de Amazon EventBridge.

Para configurar EventBridge para eventos de Systems Manager

1. Abra la consola de Amazon EventBridge en https://console.aws.amazon.com/events/.
2. En el panel de navegación, seleccione Reglas.
3. Elija Creación de regla.
4. Escriba un nombre y una descripción para la regla.

   Una regla no puede tener el mismo nombre que otra regla de la misma Región de AWS y del mismo bus de eventos.
5. En Bus de eventos, seleccione el bus de eventos que desea asociar a esta regla. Si desea que esta regla responda a eventos coincidentes procedentes de su propia Cuenta de AWS, seleccione default (predeterminado). Cuando un Servicio de AWS en su cuenta emite un evento, siempre va al bus de eventos predeterminado de su cuenta.
6. En Tipo de regla, elija Regla con un patrón de evento.
7. Elija Siguiente.
8. En Origen del evento, elija Eventos de AWS o eventos de socios de EventBridge.

9. En la sección Event pattern (Patrón de eventos), elija Event pattern form (Formulario de patrón de eventos).
10. Para Event source (origen de eventos), elija AWSservices (servicios).
11. En AWS service (Servicio de), elija Systems Manager.
12. En Event type (Tipo de evento), realice una de las siguientes operaciones:

    - Elija All events (Todos los eventos).

      Si elige All Events (Todos los eventos), todos los eventos emitidos por Systems Manager coincidirán con la regla. Tenga en cuenta que esta opción puede dar lugar a muchas acciones de destino de eventos.

    - Elija el tipo de evento de Systems Manager que desea utilizar para esta regla. EventBridge admite eventos de las siguientes herramientas de AWS Systems Manager:
      - Automatización
      - Change Calendar
      - Conformidad
      - Inventario
      - Maintenance Windows
      - Parameter Store
      - Run Command
      - State Manager

    > **Note**
    >
    > Para las acciones de Systems Manager que no son compatibles con EventBridge, puede elegir una llamada a la API de AWS a través de CloudTrail para crear una regla de evento basada en una llamada a la API, que registra CloudTrail. Para ver un ejemplo, consulta Supervisión de la actividad de la sesión con Amazon EventBridge (consola).

13. (Opcional) Para que la regla sea más específica, agregue valores de filtro. Por ejemplo, si eligió State Manager y desea limitar la regla al estado de un objeto de una instancia administrada única a la que se dirige una asociación, en Specific type(s) (Tipos específicos), elija EC2 State Manager Instance Association State Change (Cambio de estado de asociación de instancia de EC2 State Manager).

Para obtener información completa sobre los tipos de detalles compatibles, consulte Referencia: patrones y tipos de eventos de Amazon EventBridge para Systems Manager.

Algunos tipos de detalles tienen otras opciones compatibles, como el estado. Las opciones disponibles dependen de la herramienta que seleccione.

14. Elija Siguiente.
15. En Tipos de destino, seleccione Servicio de AWS.
16. En Select a target (Seleccione un destino), elija un destino, como un tema de Amazon SNS o una función de AWS Lambda. El destino se activa cuando se recibe un evento que coincide con el patrón de eventos definido en la regla.
17. Si hay muchos tipos de destinos, EventBridge necesita permisos para enviar eventos al destino. En estos casos, EventBridge puede crear el rol de AWS Identity and Access Management (IAM) necesario para que se ejecute la regla:
    - Para crear un rol de IAM automáticamente, elija Creación de un nuevo rol para este recurso específico.
    - Para utilizar un rol de IAM que haya creado antes, elija Use existing role (Usar rol existente).
18. (Opcional) Elija Add another target (Agregar otro destino) para agregar otro destino para esta regla.
19. Seleccione Siguiente.
20. (Opcional) Introduzca una o varias etiquetas para la regla. Para obtener más información, consulte Etiquetas de Amazon EventBridge en la Guía del usuario de Amazon EventBridge.
21. Elija Siguiente.
22. Revise los detalles de la regla y seleccione Creación de regla.

Más información

- Creación de un evento de EventBridge que utiliza un manual de procedimientos (consola)
- Transferir datos a Automatización usando transformadores de entrada
- Solución de problemas de conformidad con EventBridge
- Visualización de acciones de eliminación de inventario en EventBridge
- Configuración de las reglas de EventBridge para crear OpsItems
- Configuración de reglas de EventBridge para parámetros y políticas de parámetros

Ejemplos de eventos de Amazon EventBridge para Systems Manager

A continuación, se presentan ejemplos, en formato JSON, de eventos EventBridge admitidos para AWS Systems Manager.

Tipos de eventos de Systems Manager

- Eventos de Automation de AWS Systems Manager
- AWS Systems ManagerEventosChange Calendar
- AWS Systems ManagerEventosChange Manager
- Eventos de AWS Systems Manager Compliance
- AWS Systems ManagerEventosMaintenance Windows
- AWS Systems ManagerEventosParameter Store
- AWS Systems ManagerEventosOpsCenter
- AWS Systems ManagerEventosRun Command
- AWS Systems ManagerEventosState Manager

Eventos de Automation de AWS Systems Manager

Notificación de cambio de estado de paso de Automation

```
{
  "version": "0",
  "id": "eeca120b-a321-433e-9635-dab369006a6b",
  "detail-type": "EC2 Automation Step Status-change Notification",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2024-11-29T19:43:35Z",
  "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, 2024 7:43:25 PM",
```

```
    "StartTime": "Nov 29, 2024 7:43:23 PM",
    "Time": 2630.0,
    "StepName": "runFixedCmds",
    "Action": "aws:runCommand"
  }
}
```

Notificación de cambio de estado de ejecución de Automation

```
{
  "version": "0",
  "id": "d290ece9-1088-4383-9df6-cd5b4ac42b99",
  "detail-type": "EC2 Automation Execution Status-change Notification",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2024-11-29T19:43:35Z",
  "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, 2024 7:43:20 PM",
    "EndTime": "Nov 29, 2024 7:43:26 PM",
    "Time": 5753.0,
    "ExecutedBy": "arn:aws:iam::123456789012:user/userName"
  }
}
```

AWS Systems ManagerEventosChange Calendar

Utilice la información en este tema para planificar y comprender el comportamiento de los eventos de EventBridge para Change Calendar de AWS Systems Manager.

> **Note**
>
> Actualmente no se admiten cambios de estado para calendarios compartidos desde otras Cuentas de AWS.

Integración de Change Calendar en Amazon Eventbridge

Change Calendar de AWS Systems Manager se integra en Amazon EventBridge para notificarle los cambios de estado del calendario. Tenga en cuenta los siguientes comportamientos relacionados con la arquitectura de programación subyacente:

Fiabilidad y temporización de los eventos

- EventBridge entrega las notificaciones en la medida de lo posible, con una tolerancia de programación de hasta 15 minutos.
- Los eventos de cambio de estado reflejan las transiciones generales del estado del calendario, no los eventos individuales del calendario.
- Cuando se producen varios eventos del calendario simultáneamente, EventBridge genera solo un evento por cada cambio de estado del calendario real.
- EventBridge activa eventos solo cuando el estado general del calendario cambia (por ejemplo, de CERRADO a ABIERTO), no para eventos de calendario individuales que no provocan un cambio de estado.
- Los eventos de asesoramiento que no modifican el estado del calendario no activan las notificaciones de EventBridge.

Modificaciones de eventos y consideraciones de temporización

- Si modificas los eventos del calendario dentro de los 15 minutos de la hora de inicio o finalización programada, EventBridge podría generar notificaciones duplicadas o no recibir notificaciones.
- Este comportamiento se debe a que el sistema de programación podría no tener tiempo suficiente para actualizar o cancelar correctamente las notificaciones previamente programadas.
- En el caso de los eventos recurrentes, este comportamiento suele afectar solo a la primera aparición después de la modificación.

Eventos adyacentes y superpuestos

- Cuando los eventos del calendario se programan con una diferencia de 5 minutos entre sí, los eventos de transición de estado pueden ocurrir o no, según el cambio de estado real.
- La creación de eventos superpuestos en ciertos órdenes puede generar eventos de EventBridge adicionales incluso cuando no se produce ningún cambio de estado real.
- Para garantizar un comportamiento predecible, evite crear o modificar eventos del calendario cerca de sus tiempos de ejecución.

Prácticas recomendadas

- Diseñe sus reglas de EventBridge y la automatización posterior para gestionar posibles eventos duplicados.
- Implemente la idempotencia en sus flujos de trabajo de automatización para evitar problemas derivados de la duplicación de notificaciones.
- Configure un tiempo de espera suficiente (de al menos 15 minutos) al crear o modificar los eventos del calendario.
- Pruebe de manera exhaustiva sus integraciones de EventBridge con sus patrones específicos de eventos de calendario.

Calendario ABIERTO

```
{
    "version": "0",
    "id": "47a3f03a-f30d-1011-ac9a-du3bdEXAMPLE",
    "detail-type": "Calendar State Change",
    "source": "aws.ssm",
    "account": "123456789012",
    "time": "2024-09-19T18:00:07Z",
    "region": "us-east-2",
    "resources": [
        "arn:aws:ssm:us-east-2:123456789012:document/MyCalendar"
    ],
    "detail": {
        "state": "OPEN",
        "atTime": "2024-09-19T18:00:07Z",
        "nextTransitionTime": "2024-10-11T18:00:07Z"
    }
}
```

Calendario CERRADO

```
{
    "version": "0",
    "id": "f30df03a-1011-ac9a-47a3-f761eEXAMPLE",
    "detail-type": "Calendar State Change",
    "source": "aws.ssm",
    "account": "123456789012",
    "time": "2024-09-17T21:40:02Z",
    "region": "us-east-2",
```

```
    "resources": [
        "arn:aws:ssm:us-east-2:123456789012:document/MyCalendar"
    ],
    "detail": {
        "state": "CLOSED",
        "atTime": "2024-08-17T21:40:00Z",
        "nextTransitionTime": "2024-09-19T18:00:07Z"
    }
}
```

AWS Systems ManagerEventosChange Manager

Cambiar notificación de actualización del estado de la solicitud: ejemplo 1

```
{
  "version": "0",
  "id": "feab80c1-a8ff-c721-b8b1-96ce70939696",
  "detail-type": "Change Request Status Update",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2024-10-24T10:51:52Z",
  "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": "2024-10-24T10:50:33.180334Z",
    "ops-item-modified-by": "arn:aws:iam::123456789012:user/JohnDoe",
    "ops-item-modified-time": "2024-10-24T10:50:33.180340Z",
    "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"
    ]
  }
```

```
}
```

Cambiar notificación de actualización del estado de la solicitud: ejemplo 2

```
{
  "version": "0",
  "id": "25ce6b03-2e4e-1a2b-2a8f-6c9de8d278d2",
  "detail-type": "Change Request Status Update",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2024-10-24T10:51:52Z",
  "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": "2024-10-24T10:50:33.180334Z",
    "ops-item-modified-by": "arn:aws:iam::123456789012:user/JohnDoe",
    "ops-item-modified-time": "2024-10-24T10:50:33.997163Z",
    "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"
    ]
  }
}
```

Eventos de AWS Systems Manager Compliance

Los siguientes ejemplos corresponden a eventos de AWS Systems Manager Compliance.

Conformidad con la asociación

```
{
  "version": "0",
```

```
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Configuration Compliance State Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2024-07-17T19:03:26Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:managed-instance/i-01234567890abcdef"
  ],
  "detail": {
    "last-runtime": "2024-01-01T10:10:10Z",
    "compliance-status": "compliant",
    "resource-type": "managed-instance",
    "resource-id": "i-01234567890abcdef",
    "compliance-type": "Association"
  }
}
```

Sin conformidad con la asociación

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Configuration Compliance State Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2024-07-17T19:02:31Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:managed-instance/i-01234567890abcdef"
  ],
  "detail": {
    "last-runtime": "2024-01-01T10:10:10Z",
    "compliance-status": "non_compliant",
    "resource-type": "managed-instance",
    "resource-id": "i-01234567890abcdef",
    "compliance-type": "Association"
  }
}
```

Conformidad con los parches

```
{
```

```
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Configuration Compliance State Change",
  "source": "aws.123456789012",
  "account": "123456789012",
  "time": "2024-07-17T19:03:26Z",
  "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"
  }
}
```

Sin conformidad con los parches

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Configuration Compliance State Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2024-07-17T19:02:31Z",
  "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 ManagerEventosMaintenance Windows

A continuación, se muestran ejemplos de los eventos para Systems Manager Maintenance Windows.

Registrar un destino

Los valores de estado válidos son `REGISTERED` y `DEREGISTERED`.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window Target Registration Notification",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2024-11-16T00:58:37Z",
   "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"
   }
}
```

Tipo de ejecución de ventana

Entre los valores de estado válidos se incluyen los siguientes:

- `CANCELLED`
- `CANCELLING`
- `FAILED`
- `IN_PROGRESS`
- `PENDING`
- `SKIPPED_OVERLAPPING`
- `SUCCESS TIMED_OUT`

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window Execution State-change Notification",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2025-06-02T14:52:18Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-0c50858d01EXAMPLE"
   ],
   "detail":{
      "start-time":"2025-06-02T14:48:28.039273Z",
      "end-time":"2025-06-02T14:52:18.083773Z",
      "window-id":"mw-0ed7251d3fcf6e0c2",
      "window-execution-id":"14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE",
      "status":"SUCCESS"
   }
}
```

Tipo de ejecución de tarea

Los valores de estado válidos son `IN_PROGRESS`, `SUCCESS`, `FAILED` y `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":"2025-06-02T14:52:18Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-0c50858d01EXAMPLE"
   ],
   "detail":{
      "start-time":"2025-06-02T14:48:28.039273Z",
      "task-execution-id":"6417e808-7f35-4d1a-843f-123456789012",
      "end-time":"2025-06-02T14:52:18.083773Z",
      "window-id":"mw-0ed7251d3fcf6e0c2",
      "window-execution-id":"14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE",
      "status":"SUCCESS"
   }
```

```
}
```

Destino de tarea procesado

Los valores de estado válidos son `IN_PROGRESS`, `SUCCESS`, `FAILED` y `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":"2025-06-02T14:52:18Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-123456789012345678"
   ],
   "detail":{
      "start-time":"2025-06-02T14:48:28.039273Z",
      "end-time":"2025-06-02T14:52:18.083773Z",
      "window-id":"mw-0ed7251d3fcf6e0c2",
      "window-execution-id":"791b72e0-f0da-4021-8b35-f95dfEXAMPLE",
      "task-execution-id":"c9b05aba-197f-4d8d-be34-e73fbEXAMPLE",
      "window-target-id":"e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE",
      "status":"SUCCESS",
      "owner-information":"Owner"
   }
}
```

Cambio de estado de ventana

Los valores de estado válidos son `ENABLED` y `DISABLED`.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window State-change Notification",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2024-11-16T00:58:37Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-0c50858d01EXAMPLE"
```

```
   ],
   "detail":{
      "window-id":"mw-0c50858d01EXAMPLE",
      "status":"DISABLED"
   }
}
```

AWS Systems ManagerEventosParameter Store

A continuación, se muestran ejemplos de los eventos para Systems Manager Parameter Store.

Crear parámetro

```
{
  "version": "0",
  "id": "6a7e4feb-b491-4cf7-a9f1-bf3703497718",
  "detail-type": "Parameter Store Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2024-05-22T16:43:48Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:parameter/MyExampleParameter"
  ],
  "detail": {
    "operation": "Create",
    "name": "MyExampleParameter",
    "type": "String",
    "description": "Sample Parameter"
  }
}
```

Actualizar parámetro

```
{
  "version": "0",
  "id": "9547ef2d-3b7e-4057-b6cb-5fdf09ee7c8f",
  "detail-type": "Parameter Store Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2024-05-22T16:44:48Z",
```

```
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:parameter/MyExampleParameter"
  ],
  "detail": {
    "operation": "Update",
    "name": "MyExampleParameter",
    "type": "String",
    "description": "Sample Parameter"
  }
}
```

Eliminar parámetro

```
{
  "version": "0",
  "id": "80e9b391-6a9b-413c-839a-453b528053af",
  "detail-type": "Parameter Store Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2024-05-22T16:45:48Z",
  "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 ManagerEventosOpsCenter

OpsCenter OpsItem crear notificación

```
{
  "version": "0",
  "id": "aae66adc-7aac-f0c0-7854-7691e8c079b8",
  "detail-type": "OpsItem Create",
  "source": "aws.ssm",
  "account": "123456789012",
```

```
  "time": "2024-10-19T02:48:11Z",
  "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": "2024-10-19T02:46:53.629361Z",
    "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"
  }
}
```

OpsCenter OpsItem actualizar notificación

```
{
  "version": "0",
  "id": "2fb5b168-b725-41dd-a890-29311200089c",
  "detail-type": "OpsItem Update",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2024-10-19T02:48:11Z",
  "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": "2024-10-19T02:46:54.049271Z",
    "modified-by": "arn:aws:iam::123456789012:user/JohnDoe",
    "modified-time": "2024-10-19T02:46:54.337354Z",
    "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 ManagerEventosRun Command

Notificación de cambio de estado de Run Command


```
{
    "version": "0",
    "id": "51c0891d-0e34-45b1-83d6-95db273d1602",
    "detail-type": "EC2 Command Status-change Notification",
    "source": "aws.ssm",
    "account": "123456789012",
    "time": "2024-07-10T21:51:32Z",
    "region": "us-east-2",
    "resources": ["arn:aws:ec2:us-east-2:123456789012:instance/i-02573cafcfEXAMPLE"],
    "detail": {
        "command-id": "e8d3c0e4-71f7-4491-898f-c9b35bee5f3b",
        "document-name": "AWS-RunPowerShellScript",
        "expire-after": "2024-07-14T22:01:30.049Z",
        "parameters": {
            "executionTimeout": ["3600"],
            "commands": ["date"]
        },
        "requested-date-time": "2024-07-10T21:51:30.049Z",
        "status": "Success"
    }
}
```


Notificación de cambio de estado de invocación de Run Command


```
{
    "version": "0",
    "id": "4780e1b8-f56b-4de5-95f2-95dbEXAMPLE",
    "detail-type": "EC2 Command Invocation Status-change Notification",
    "source": "aws.ssm",
    "account": "123456789012",
    "time": "2024-07-10T21:51:32Z",
    "region": "us-east-2",
    "resources": ["arn:aws:ec2:us-east-2:123456789012:instance/i-02573cafcfEXAMPLE"],
    "detail": {
        "command-id": "e8d3c0e4-71f7-4491-898f-c9b35bee5f3b",
        "document-name": "AWS-RunPowerShellScript",
        "instance-id": "i-02573cafcfEXAMPLE",
        "requested-date-time": "2024-07-10T21:51:30.049Z",
        "status": "Success"
```

```
    }
}
```

AWS Systems ManagerEventosState Manager

Cambio de estado de asociación de State Manager

```
{
   "version":"0",
   "id":"db839caf-6f6c-40af-9a48-25b2ae2b7774",
   "detail-type":"EC2 State Manager Association State Change",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2024-05-16T23:01:10Z",
   "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":"2024-02-13T17:22:54.458Z",
      "last-successful-execution-date":"2024-05-16T23:00:01Z",
      "last-execution-date":"2024-05-16T23:00:01Z",
      "last-updated-date":"2024-02-13T17:22:54.458Z",
      "status":"Success",
      "association-status-aggregated-count":"{\"Success\":1}",
      "schedule-expression":"cron(0 */30 * * * ? *)",
      "association-cwe-version":"1.0"
   }
}
```

Cambio de estado de asociación de instancia de 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":"2024-02-23T15:23:48Z",
   "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-02573cafcfEXAMPLE",
      "document-name":"my-custom-document",
      "document-version":"1",
      "targets":"[{\"key\":\"instanceids\",\"values\":[\"i-02573cafcfEXAMPLE\"]}]",
      "creation-date":"2024-02-23T15:23:48Z",
      "last-successful-execution-date":"2024-02-23T16:23:48Z",
      "last-execution-date":"2024-02-23T16:23:48Z",
      "status":"Success",
      "detailed-status":"",
      "error-code":"testErrorCode",
      "execution-summary":"testExecutionSummary",
      "output-url":"sampleurl",
      "instance-association-cwe-version":"1"
   }
}
```


Escenarios de ejemplo: destinos de Systems Manager en reglas de Amazon EventBridge

Cuando especifica el destino que se va a invocar en una regla de Amazon EventBridge, puede elegir entre más de 20 tipos de destinos y agregar hasta cinco destinos a cada regla.

De los diferentes destinos, puede elegir entre Automatización, OpsCenter y Run Command, que son herramientas de AWS Systems Manager, como acciones de destino cuando se produce un evento de EventBridge.

A continuación, se presentan varios ejemplos de formas en las que puede utilizar estas herramientas como destino de una regla de EventBridge.

Ejemplos de Automation

Puede configurar una regla de EventBridge para iniciar flujos de trabajo de Automation cuando se produzcan eventos como los siguientes:

- Cuando una alarma de Amazon CloudWatch notifica que un nodo administrado no ha pasado una comprobación de estado (`StatusCheckFailed_Instance=1`), ejecute el manual de procedimientos de Automation `AWSSupport-ExecuteEC2Rescue` en el nodo.
- Cuando se produce un evento `EC2 Instance State-change Notification` porque se está ejecutando una nueva instancia de Amazon Elastic Compute Cloud (Amazon EC2), ejecute el manual de procedimientos de Automation `AWS-AttachEBSVolume` en la instancia.
- Cuando se cree y esté disponible un volumen de Amazon Elastic Block Store (Amazon EBS), ejecute el manual de procedimientos de Automation `AWS-CreateSnapshot` en el volumen.

Ejemplos de OpsCenter

Puede configurar una regla de EventBridge para crear un nuevo OpsItem cuando se producen incidentes como los siguientes:

- Se produce un evento de limitación controlada para Amazon DynamoDB o el rendimiento del volumen de Amazon EBS se ha degradado.
- Un grupo de Amazon EC2 Auto Scaling no puede lanzar un nodo o un flujo de trabajo de Systems Manager Automation falla.
- Una instancia de EC2 cambia el estado de `Running` a `Stopped`.

Ejemplos de Run Command

Puede configurar una regla de EventBridge para ejecutar un documento de Systems Manager Command en Run Command cuando se producen eventos como los siguientes:

- Cuando un grupo de Auto Scaling está a punto de finalizar, un script Run Command podría capturar los archivos de registro del nodo antes de que finalice.
- Cuando se crea un nodo nuevo en un grupo de Auto Scaling, una acción de destino Run Command podría producir la activación del rol del servidor web o la instalación de software en el nodo.
- Cuando se descubre un nodo administrado no conforme, una acción de destino Run Command puede actualizar las revisiones en el nodo mediante la ejecución del documento `AWS-RunPatchBaseline`.

Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS

Puede configurar Amazon Simple Notification Service (Amazon SNS) para que envíe notificaciones sobre el estado de los comandos que envía a través de Run Command o Maintenance Windows, que son herramientas de AWS Systems Manager. Amazon SNS coordina y administra el envío y la entrega de las notificaciones a los clientes o puntos de enlace que estén suscritos a temas de Amazon SNS. Puede recibir una notificación siempre que un comando cambie a un estado nuevo o cambie a un estado específico como, por ejemplo, Failed (Error) o Timed Out (Tiempo de espera agotado). En los casos en que un comando se envía a varios nodos, puede recibir una notificación por cada copia del comando enviada a un nodo concreto. Cada copia se denomina una invocación.

Amazon SNS puede entregar las notificaciones como HTTP o HTTPS POST, por email (SMTP, con texto sin formato o con formato JSON) o como mensaje publicado en una cola de Amazon Simple Queue Service (Amazon SQS). Para obtener más información, consulte ¿Qué es Amazon SNS? en la Guía para desarrolladores de Amazon Simple Notification Service. Para ver ejemplos de la estructura de los datos JSON incluidos en la notificación de Amazon SNS que proporciona Run Command y el Maintenance Windows, consulte Ejemplo de notificaciones de Amazon SNS para AWS Systems Manager.

Important

Tenga en cuenta la siguiente información importante.

- Los temas FIFO de Amazon Simple Notification Service no son compatibles.
- No se puede usar Amazon Q Developer en aplicaciones de chat para monitorear Systems Manager con Amazon SNS. Si quiere usar Amazon Q Developer en aplicaciones de chat para monitorear Systems Manager, debe usarlo con Amazon EventBridge. Para obtener información acerca de cómo utilizar EventBridge para supervisar los eventos de Systems Manager, consulte Monitoreo de eventos de Systems Manager con Amazon EventBridge. Para obtener información sobre Amazon EventBridge y Amazon Q Developer en aplicaciones de chat, consulte Tutorial: Creating an EventBridge rule that sends notifications to Amazon Q Developer in chat applications en Amazon Q Developer in chat applications Administrator Guide.

Cómo configurar notificaciones de Amazon SNS para AWS Systems Manager

Run Command y las tareas de Maintenance Windows registradas en un periodo de mantenimiento pueden enviar notificaciones de Amazon SNS para tareas de comandos que introducen los siguientes estados:

- En curso
- Success
- Con error
- Tiempo de espera agotado
- Cancelado

Para obtener información sobre las condiciones que causan que un comando pase a tener uno de estos estados, consulte Descripción de los estados del comando.

> **Note**
>
> Los comandos que se envían cuando se utiliza Run Command también notifican los estados Canceling (Cancelando) y Pending (Pendiente). Estos estados no se capturan en las notificaciones de Amazon SNS.

Notificaciones de Amazon SNS de resumen de comandos

Si configura Run Command o una tarea de Run Command en su periodo de mantenimiento para notificaciones de Amazon SNS, Amazon SNS envía mensajes de resumen que contienen la siguiente información.

Campo	Tipo	Descripción
eventTime	Cadena	La hora a la que se inició el evento. La marca temporal es importante porque Amazon SNS no garantiza el orden de entrega de los mensajes.

Campo	Tipo	Descripción
		Ejemplo: 2016-04-26T13:15:30Z
documentName	Cadena	El nombre del documento de SSM utilizado para ejecutar este comando.
commandId	Cadena	El ID generado por Run Command después de haber enviado el comando.
expiresAfter	Date	Si se llega a esta hora y el comando aún no ha empezado a ejecutarse, no se ejecutará.
outputS3BucketName	Cadena	El bucket de Amazon Simple Storage Service (Amazon S3) donde deben almacenarse las respuestas a la ejecución de comandos.
outputS3KeyPrefix	Cadena	La ruta del directorio de Amazon S3 dentro del bucket donde deben guardarse las respuestas a la ejecución de comandos.
requestedDateTime	Cadena	La hora y la fecha en la que se envió la solicitud a este nodo específico.

Campo	Tipo	Descripción
instanceIds	StringList	Los nodos a los que se dirige el comando. **Note** Los ID de instancia solo se incluyen en el mensaje de resumen si la tarea Run Command dirigió a sus destinos los ID de instancia directamente. Los ID de instancia no se incluyen en el mensaje de resumen si la tarea Run Command se emitió con destino basado en etiquetas.
status	Cadena	El estado del comando.

Notificaciones de Amazon SNS basadas en invocaciones

Si envía un comando a varios nodos, Amazon SNS puede enviar mensajes sobre cada copia o invocación del comando. Los mensajes incluyen la siguiente información.

Campo	Tipo	Descripción
eventTime	Cadena	La hora a la que se inició el evento. La marca temporal es importante porque Amazon SNS no garantiza el orden de entrega de los mensajes.

Campo	Tipo	Descripción
		Ejemplo: 2016-04-26T13:15:30Z
documentName	Cadena	El nombre del documento de Systems Manager (document o SSM) utilizado para ejecutar este comando.
requestedDateTime	Cadena	La hora y la fecha en la que se envió la solicitud a este nodo específico.
commandId	Cadena	El ID generado por Run Command después de haber enviado el comando.
instanceId	Cadena	La instancia que el comando tiene como destino.
status	Cadena	Estado del comando para esta invocación.

Para configurar las notificaciones de Amazon SNS cuando un comando cambie el estado, realice las siguientes tareas.

> **Note**
>
> Si no va a configurar las notificaciones de Amazon SNS para su periodo de mantenimiento, puede omitir la tarea 5 que se encuentra más adelante en este tema.

Temas

- Tarea 1: crear y suscribirse a un tema de Amazon SNS
- Tarea 2: crear una política de IAM para notificaciones de Amazon SNS
- Tarea 3: crear un rol de IAM para notificaciones de Amazon SNS

- Tarea 4: configurar el acceso del usuario
- Tarea 5: asociar la política iam:PassRole al rol de periodo de mantenimiento

Tarea 1: crear y suscribirse a un tema de Amazon SNS

Un Tema de Amazon SNS es un canal de comunicación queRun Command y las tareas de Run Command registradas en un periodo de mantenimiento utilizan para enviar notificaciones sobre el estado de sus comandos. Amazon SNS admite diferentes protocolos de comunicación, incluidos HTTP/S, email y otros Servicios de AWS, como Amazon Simple Queue Service (Amazon SQS). Para comenzar, le recomendamos que empiece con el protocolo de email. Para obtener más información acerca de la creación de un tema, consulte Creating an Amazon SNS topic (Creación de un tema de Amazon SNS) en la Guía para desarrolladores de Amazon Simple Notification Service.

> **Note**
>
> Después de crear el tema, copie o anote el valor de Topic ARN. Tendrá que especificar este ARN al enviar un comando que está configurado para devolver notificaciones de estado.

Después de crear el tema, suscríbase a él especificando un punto de enlace. Si eligió el protocolo de correo electrónico el punto de enlace es la dirección de correo electrónico donde desea recibir las notificaciones. Para obtener más información acerca de cómo se suscribe a un tema, consulte Subscribing to an Amazon SNS topic (Suscripción a un tema de Amazon SNS) en la Guía para desarrolladores de Amazon Simple Notification Service.

Amazon SNS envía un email de confirmación desde AWSNotifications a la dirección de email que especifique. Abra el email y elija el enlace Confirm subscription (Confirmar la suscripción).

Recibirá un mensaje de confirmación de AWS. Amazon SNS estará ahora configurado para recibir notificaciones y enviar la notificación como un email a la dirección especificada.

Tarea 2: crear una política de IAM para notificaciones de Amazon SNS

Utilice el siguiente procedimiento para crear una política de AWS Identity and Access Management (IAM) personalizada que proporcione permisos para iniciar notificaciones de Amazon SNS.

Cómo crear una política personalizada de IAM para notificaciones de Amazon SNS

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.

2. En el panel de navegación, seleccione Policies y, a continuación, seleccione Create Policy. (Si aparece el botón Get Started [Empezar], elíjalo y, a continuación, elija Create Policy [Crear política]).
3. Seleccione la pestaña JSON.
4. Sustituya el contenido predeterminado por uno de los siguientes, en función de si el tema de Amazon SNS utiliza cifrado de AWS KMS:

   SNS topic not encrypted

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sns:Publish"
            ],
            "Resource": "arn:aws:sns:us-east-1:111122223333:sns-topic-
name"
        }
    ]
}
```

   region representa el identificador de Región de AWS compatible con AWS Systems Manager, como `us-east-2` para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

   account-id representa el identificador de 12 dígitos de su Cuenta de AWS en el formato `123456789012`.

   sns-topic-name representa el nombre del tema de Amazon SNS que desea utilizar para la publicación de notificaciones.

SNS topic encrypted

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sns:Publish"
            ],
            "Resource": "arn:aws:sns:us-east-1:111122223333:sns-topic-
name"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:GenerateDataKey",
                "kms:Decrypt"
            ],
            "Resource": "arn:aws:kms:us-east-1:111122223333:key/kms-key-
id"
        }
    ]
}
```

region representa el identificador de Región de AWS compatible con AWS Systems Manager, como `us-east-2` para la región EE. UU. Este (Ohio). Para ver una lista de los valores de *regiones* admitidos, consulte la columna Región en Puntos de conexión de servicio de Systems Manager en la Referencia general de Amazon Web Services.

account-id representa el identificador de 12 dígitos de su Cuenta de AWS en el formato `123456789012`.

sns-topic-name representa el nombre del tema de Amazon SNS que desea utilizar para la publicación de notificaciones.

kms-key-id representa el ID de la clave de KMS de cifrado simétrico en AWS KMS que se utilizará para cifrar y descifrar el tema, en el formato `1234abcd-12ab-34cd-56ef-12345EXAMPLE`.

> **Note**
>
> El uso del cifrado de AWS KMS conlleva un costo. Para obtener más información, consulte Administración de las claves de cifrado y los costos de Amazon SNS en la Guía para desarrolladores de AWS Key Management Service.

5. Elija Siguiente: etiquetas.
6. (Opcional) Agregue uno o varios pares de valor etiqueta-clave para organizar, realizar un seguimiento o controlar el acceso a esta política.
7. Elija Siguiente: Revisar.
8. En la página Review Policy (Revisar política), en Name (Nombre), escriba un nombre para la política insertada. Por ejemplo: **`my-sns-publish-permissions`**.
9. (Opcional) En Description (Descripción), escriba una descripción para la política.
10. Seleccione Crear política.

Tarea 3: crear un rol de IAM para notificaciones de Amazon SNS

Utilice el siguiente procedimiento para crear un rol de IAM para las notificaciones de Amazon SNS. Systems Manager utiliza este rol de servicio para dar comienzo a las notificaciones de Amazon SNS. En todos los procedimientos siguientes, este rol se denomina rol de IAM de Amazon SNS.

Cómo crear un rol de servicio de IAM para las notificaciones de Amazon SNS

1. Inicie sesión en AWS Management Console y abra la consola IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación de la consola de IAM, seleccione Roles y, a continuación, elija Crear rol.
3. Elija el tipo de rol Servicio de AWS y, a continuación, seleccione Systems Manager.
4. Elija el caso de uso de Systems Manager. A continuación, elija Siguiente.

5. En la página Attach permissions policies (Adjuntar políticas de permisos), seleccione la casilla situada a la izquierda del nombre de la política personalizada creada en la tarea 2. Por ejemplo: **`my-sns-publish-permissions`**.
6. (Opcional) Configure un límite de permisos. Se trata de una característica avanzada que está disponible para los roles de servicio, pero no para los roles vinculados a servicios.

   Amplíe la sección Límite de permisos y seleccione Usar un límite de permisos para controlar los permisos máximos de la función. IAM incluye una lista de las políticas administradas por AWS y de las políticas administradas por el cliente de cada cuenta. Seleccione la política que desea utilizar para el límite de permisos o elija Crear política para abrir una pestaña nueva del navegador y crear una política nueva desde cero. Para obtener más información, consulte Creación de políticas de IAM en la Guía del usuario de IAM. Una vez creada la política, cierre la pestaña y vuelva a la pestaña original para seleccionar la política que va a utilizar para el límite de permisos.
7. Elija Siguiente.
8. De ser posible, escriba un nombre o sufijo de nombre para el rol, que pueda ayudarle a identificar su finalidad. Los nombres de rol deben ser únicos en su Cuenta de AWS. No distinguen entre mayúsculas y minúsculas. Por ejemplo, no puede crear funciones denominado tanto **`PRODROLE`** como **`prodrole`**. Dado que varias entidades pueden hacer referencia al rol, no puede editar el nombre del rol después de crearlo.
9. (Opcional) En Descripción, ingrese una descripción para el nuevo rol.
10. Seleccione Editar en las secciones Paso 1: seleccionar entidades de confianza o Paso 2: seleccionar permisos para editar los casos de uso y los permisos del rol.
11. (Opcional) Asocie etiquetas como pares de clave-valor para agregar metadatos al rol. Para obtener más información sobre el uso de etiquetas en IAM, consulte Etiquetado de recursos de IAM en la Guía de usuario de IAM .
12. Revise el rol y, a continuación, seleccione Crear rol.
13. Elija el nombre del rol y copie o anote el valor de Role ARN (ARN del rol). Este nombre de recurso de Amazon (ARN) para el rol se utiliza cuando envía un comando que está configurado para devolver notificaciones de Amazon SNS.
14. Mantenga la página Resumen abierta.

Tarea 4: configurar el acceso del usuario

Si a una entidad de IAM (usuario, rol o grupo) se le asignan permisos de administrador, el usuario o el rol tiene acceso a Run Command y Maintenance Windows, que son herramientas de AWS Systems Manager.

Para las entidades sin permisos de administrador, el administrador debe conceder los siguientes permisos a la entidad de IAM:

- La política administrada `AmazonSSMFullAccess`, o una política que proporcione permisos comparables.
- Permisos `iam:PassRole` para el rol creado en Tarea 3: crear un rol de IAM para notificaciones de Amazon SNS. Por ejemplo:

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::111122223333:role/sns-role-name",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "ssm.amazonaws.com"
                }
            }
        }
    ]
}
```

Para dar acceso, agregue permisos a los usuarios, grupos o roles:

- Usuarios y grupos en AWS IAM Identity Center:

  Cree un conjunto de permisos. Siga las instrucciones de Creación de un conjunto de permisos en la Guía del usuario de AWS IAM Identity Center.
- Usuarios gestionados en IAM a través de un proveedor de identidades:

Cree un rol para la federación de identidades. Siga las instrucciones descritas en Creación de un rol para un proveedor de identidad de terceros (federación) en la Guía del usuario de IAM.

- Usuarios de IAM:
  - Cree un rol que el usuario pueda aceptar. Siga las instrucciones descritas en Creación de un rol para un usuario de IAM en la Guía del usuario de IAM.
  - (No recomendado) Adjunte una política directamente a un usuario o añada un usuario a un grupo de usuarios. Siga las instrucciones descritas en Adición de permisos a un usuario (consola) de la Guía del usuario de IAM.

Cómo configurar el acceso de los usuarios y adjuntar la política **`iam:PassRole`** a una cuenta de usuario

1. En el panel de navegación de IAM, elija Users (Usuarios) y, a continuación, seleccione la cuenta de usuario que desea configurar.
2. En la pestaña Permissions (Permisos), en la lista de políticas, verifique que aparece la política **`AmazonSSMFullAccess`** o que hay una política equivalente que conceda los permisos de cuenta para acceder a Systems Manager.
3. Elija Agregar política insertada.
4. En la página Create policy (Crear política), elija la pestaña Visual editor (Editor visual).
5. Elija Choose a service (Elegir un servicio) y después IAM.
6. En Actions (Acciones), en el cuadro de texto Filter actions (Filtrar acciones), ingrese **`PassRole`** y, a continuación, elija la casilla junto a PassRole.
7. En Resources (Recursos), compruebe que esté seleccionado Specific (Específicos) y elija Add ARN (Agregar ARN).
8. En el campo Specify ARN for role (Especificar el ARN del rol), pegue el ARN del rol de IAM de Amazon SNS que copió al final de la tarea 3. El sistema rellena automáticamente los campos Account (Cuenta) y Role name with path (Nombre del rol con ruta).
9. Elija Add (Agregar).
10. Elija Review policy (Revisar política).
11. En la página Review Policy (Revisar política), ingrese un nombre y, a continuación, elija Create policy (Crear la política).

Tarea 5: asociar la política iam:PassRole al rol de periodo de mantenimiento

Al registrar una tarea de Run Command a un periodo de mantenimiento, debe especificar un rol de servicio de nombre de recurso de Amazon (ARN). Systems Manager utiliza este rol de servicio para ejecutar tareas registradas en el periodo de mantenimiento. Para configurar notificaciones de Amazon SNS para una tarea de Run Command registrada, adjunte una política `iam:PassRole` al rol de servicio del periodo de mantenimiento especificado. Si no tiene intención de configurar la tarea registrada para notificaciones de Amazon SNS, puede omitir esta tarea.

La política `iam:PassRole` permite al rol de servicio del Maintenance Windows transferir el rol de IAM de Amazon SNS creado en la tarea 3 al servicio de Amazon SNS. El siguiente procedimiento muestra cómo adjuntar la política `iam:PassRole` al rol de servicio del Maintenance Windows.

> **Note**
>
> Debe utilizar un rol de servicio personalizado para que su periodo de mantenimiento envíe notificaciones relacionadas con las tareas de Run Command registradas. Para obtener más información, consulta Configuración de Maintenance Windows.
> Si necesita crear un rol de servicio personalizado para las tareas de periodo de mantenimiento, consulte Configuración de Maintenance Windows.

Cómo adjuntar la política de **`iam:PassRole`** a su rol de Maintenance Windows

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, seleccione Roles (Roles) y seleccione el rol de IAM de Amazon SNS creado en la tarea 3.
3. Copie o anote el valor de Role ARN (ARN de rol) y regrese a la sección Roles (Roles) de la consola de IAM.
4. Seleccione el rol de servicio del Maintenance Windows personalizado creado de la lista de Role name (Nombre del rol).
5. En la pestaña Permissions (Permisos), verifique si aparece la política `AmazonSSMMaintenanceWindowRole` o si hay una política equivalente que conceda permisos de periodo de mantenimiento para la API de Systems Manager. Si no es así, elija Agregar permisos, Adjuntar políticas para adjuntarla.
6. Elija Add permissions, Create inline policy (Agregar permisos, Crear política insertada).
7. Seleccione la pestaña Visual editor (Editor visual).

8. En Service (Servicio), seleccione IAM.
9. En Actions (Acciones), en el cuadro de texto Filter actions (Filtrar acciones), ingrese **`PassRole`** y, a continuación, elija la casilla junto a PassRole.
10. En Resources (Recursos), elija Specific (Específico), y, a continuación, elija Add ARN (Agregar ARN).
11. En el cuadro Specify ARN for role (Especificar ARN para rol) pegue el ARN del rol de IAM de Amazon SNS creado en la tarea 3 y, a continuación, elija Add (Agregar).
12. Elija Revisar política.
13. En la página Revisar política, especifique un nombre para la política `PassRole` y, a continuación, elija Crear política.

Ejemplo de notificaciones de Amazon SNS para AWS Systems Manager

Puede configurar Amazon Simple Notification Service (Amazon SNS) para que envíe notificaciones sobre el estado de los comandos que envía a través de Run Command o Maintenance Windows, que son herramientas de AWS Systems Manager.

> **Note**
>
> Esta guía no trata sobre el modo de configurar notificaciones para Run Command o un Maintenance Windows. Para obtener más información acerca de cómo configurar Run Command o un Maintenance Windows para enviar notificaciones de Amazon SNS sobre el estado de los comandos, consulte Cómo configurar notificaciones de Amazon SNS para AWS Systems Manager.

Los siguientes ejemplos muestran la estructura de la salida JSON devuelta mediante notificaciones de Amazon SNS cuando se configura para Run Command o un Maintenance Windows.

Ejemplo de salida JSON para mensajes de resumen de comandos con destino de ID de instancia

```
{
    "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": "amzn-s3-demo-bucket",
    "outputS3KeyPrefix": "runcommand",
    "status": "InProgress",
    "eventTime": "2019-04-25T17:57:09.236Z"
}
```

Ejemplo de salida JSON para mensajes de resumen de comandos con destino basado en etiquetas

```
{
    "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"
}
```

Ejemplo de salida JSON para mensajes de invocación

```
{
    "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"
}
```

Uso de Run Command para enviar un comando que devuelve notificaciones de estado

Los siguientes procedimientos muestran cómo utilizar AWS Command Line Interface (AWS CLI) o la consola de AWS Systems Manager para enviar un comando configurado para devolver notificaciones de estado mediante Run Command, una herramienta de AWS Systems Manager.

Envío de un Run Command que devuelve notificaciones (consola)

Utilice el siguiente procedimiento para enviar un comando a través de Run Command configurado para devolver notificaciones de estado mediante la consola de Systems Manager.

Para enviar un comando que devuelve notificaciones (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command.
3. Elija Run command (Ejecutar comando).
4. En la lista Command document (Documento de Command), elija un documento de Systems Manager.
5. En la sección Command Parameters, especifique los valores de los parámetros obligatorios.
6. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

   > **Tip**
   >
   > Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

7. En Otros parámetros:

   - En Comentario, ingrese la información acerca de este comando.
   - En Tiempo de espera (segundos), especifique el número de segundos que tiene que esperar el sistema antes de indicar que se ha producido un error en la ejecución del comando general.

8. En Rate control (Control de velocidad):

   - En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

     > **Note**
     >
     > Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos

administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

- En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

9. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

10. En la sección SNS Notifications (Notificaciones de SNS), elija Enable SNS notifications (Habilitar notificaciones de SNS.
11. En IAM Role (Rol de IAM), elija el ARN del rol de IAM de Amazon SNS que creó en la tarea 3 en Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.
12. En SNS topic (Tema de SNS), ingrese el ARN del tema de Amazon SNS que se va a utilizar.
13. En Event notifications (Notificaciones de eventos), elija los eventos para los que desea recibir notificaciones.
14. En Change notifications (Notificaciones de cambios), elija si quiere recibir notificaciones únicamente para el resumen de comandos (Command status changes [Cambios del estado del comando]) o para cada copia de un comando enviado a varios nodos (Command status on each instance changes [Cambios del estado del comando en cada instancia]).

15. Seleccione Ejecutar.
16. Busque en su casilla de email un mensaje de Amazon SNS y ábralo. Amazon SNS puede tardar varios minutos en enviar el mensaje por email.

Envío de un Run Command que devuelve notificaciones (CLI)

Utilice el siguiente procedimiento para enviar un comando a través de Run Command configurado para devolver notificaciones de estado mediante la AWS CLI.

Para enviar un comando que devuelve notificaciones (CLI)

1. Abra la AWS CLI.
2. Especifique los parámetros en el siguiente comando para definir destinos en función de los ID de los nodos administrados.

```
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"}'
```

A continuación se muestra un ejemplo.

```
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"}'
```

Comandos alternativos

Especifique los parámetros en el siguiente comando para definir como destinos instancias administradas utilizando etiquetas.

```
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"}'
```

A continuación se muestra un ejemplo.

```
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. Pulse Intro.
4. Busque en su casilla de email un mensaje de Amazon SNS y ábralo. Amazon SNS puede tardar varios minutos en enviar el mensaje por email.

Para obtener más información, consulte send-command en la Referencia de los comandos de AWS CLI.

Uso de un periodo de mantenimiento para enviar un comando que devuelve notificaciones de estado

Los siguientes procedimientos muestran cómo registrar una tarea de Run Command con su periodo de mantenimiento mediante la consola de AWS Systems Manager o la AWS Command Line Interface (AWS CLI). Run Command es una herramienta de AWS Systems Manager. Los procedimientos también describen cómo configurar la tarea de Run Command para devolver notificaciones de estado.

Antes de empezar

Si no ha creado un periodo de mantenimiento ni destinos registrados, consulte Crear y administrar periodos de mantenimiento mediante la consola para ver los pasos que indican cómo crear un periodo de mantenimiento y cómo registrar destinos.

Para recibir notificaciones del servicio de Amazon Simple Notification Service (Amazon SNS), adjunte una política de `iam:PassRole` al rol de servicio del Maintenance Windows especificado en la tarea registrada. Si no ha agregado permisos de `iam:PassRole` a su rol de servicio de Maintenance Windows, consulte Tarea 5: asociar la política iam:PassRole al rol de periodo de mantenimiento.

Registro de una tarea de Run Command a un periodo de mantenimiento que devuelve notificaciones (consola)

Utilice el siguiente procedimiento para registrar una tarea de Run Command configurada para devolver notificaciones de estado a su periodo de mantenimiento a través de la consola de Systems Manager.

Para registrar una tarea de Run Command con el periodo de mantenimiento que devuelve notificaciones (consola)

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Maintenance Windows.
3. Seleccione el periodo de mantenimiento en el cual desea registrar una tarea de Run Command configurada para enviar notificaciones de Amazon Simple Notification Service (Amazon SNS).
4. Elija Actions (Acciones) y, a continuación, elija Register Run Command task (Registrar tarea de Run Command).
5. (Opcional) En el campo Name (Nombre), ingrese el nombre de la tarea.
6. (Opcional) Introduzca una descripción en el campo Description (Descripción).
7. En Command document (Documento de comando), elija un documento de comando.
8. En Task priority (Prioridad de tarea), especifique una prioridad para esta tarea. Cero (`0`) es la prioridad más alta. Las tareas de un periodo de mantenimiento se programan por orden de prioridad. Las tareas que tienen la misma prioridad se programan en paralelo.
9. En la sección Targets (Destinos), seleccione un grupo de destino registrado o destinos no registrados.
10. En Rate control (Control de velocidad):
    - En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

      > **Note**
      >
      > Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos

administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

- En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

11. En el área IAM service role (Rol de servicio de IAM), elija el rol de servicio de Maintenance Windows que tiene permisos de `iam:PassRole` al rol de SNS.

> **Note**
>
> Agregue permisos de `iam:PassRole` al rol Maintenance Windows para permitir que Systems Manager pase el rol de SNS a Amazon SNS. Si no ha agregado `iam:PassRole` permisos, consulte Tarea 5 en el tema Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.

12. (Opcional) En Output options (Opciones de salida), para guardar la salida del comando en un archivo, seleccione el cuadro Enable writing output to S3 (Permitir la escritura de salida en S3). Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancias asignado al nodo administrado, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, verifique que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

13. En la sección SNS notifications (Notificaciones de SNS), realice lo siguiente:

- Elija Enable SNS Notifications (Habilitar notificaciones de SNS).

- En IAM role (Rol de IAM), elija el Nombre de recurso de Amazon (ARN) del rol de IAM de Amazon SNS que ha creado en la tarea 3 en Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS para iniciar Amazon SNS.
- En SNS topic (Tema de SNS), ingrese el ARN del tema de Amazon SNS que se va a utilizar.
- En Event type (Tipo de evento), elija los eventos para los que desea recibir notificaciones.
- En el campo Notification type (Tipo de notificaciones), elija recibir notificaciones para cada copia de un comando enviado a varios nodos (invocaciones) o el resumen de comandos.

14. En la sección Parameters (Parámetros), ingrese los parámetros requeridos en función del documento de Command que elija.
15. Elija Register run command task (Registrar tarea de Run Command).
16. Después de que se ejecute su periodo de mantenimiento la próxima vez, busque en su casilla de email un mensaje de Amazon SNS y ábralo. Amazon SNS puede tardar unos minutos en enviar el mensaje por email.

Registro de una tarea de Run Command a un periodo de mantenimiento que devuelve notificaciones (CLI)

Utilice el siguiente procedimiento para registrar una tarea de Run Command configurada para devolver notificaciones de estado a su periodo de mantenimiento a través de la AWS CLI.

Para registrar una tarea de Run Command con el periodo de mantenimiento que devuelve notificaciones (CLI)

> **Note**
>
> Para administrar mejor las opciones para las tareas, este procedimiento utiliza la opción de comando `--cli-input-json` con valores opcionales almacenados en un archivo JSON.

1. En el equipo local, cree un archivo con el nombre `RunCommandTask.json`.
2. Pegue los siguientes contenidos en el archivo :

```
{
    "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. Reemplace los valores de ejemplo con información acerca de sus propios recursos.

   También puede restaurar opciones omitidas en este ejemplo si desea utilizarlas. Por ejemplo, puede guardar el resultado del comando en un bucket de S3.

   Para obtener más información, consulte register-task-with-maintenance-window en la Referencia de los comandos de AWS CLI.

4. Guarde el archivo.

5. En el directorio en su equipo local donde ha guardado el archivo, ejecute el siguiente comando.

```
aws ssm register-task-with-maintenance-window --cli-input-json file://
RunCommandTask.json
```

> **Important**
>
> Asegúrese de incluir `file://` antes del nombre de archivo. Es obligatorio en este comando.

En caso de éxito, este comando devuelve información similar a la siguiente.

```
{
    "WindowTaskId": "j2l8d5b5c-mw66-tk4d-r3g9-1d4d1EXAMPLE"
}
```

6. Después de que se ejecute su periodo de mantenimiento la próxima vez, busque en su casilla de email un mensaje de Amazon SNS y ábralo. Amazon SNS puede tardar unos minutos en enviar el mensaje por email.

Para obtener más información acerca del registro de tareas para un periodo de mantenimiento mediante la línea de comandos, consulte Registrar tareas con el periodo de mantenimiento.

Integraciones de productos y servicios a Systems Manager

De forma predeterminada, AWS Systems Manager se integra con Servicios de AWS, así como a otros productos y servicios. La siguiente información puede ayudarlo a configurar Systems Manager para que se integre a los productos y los servicios que utilice.

- Cómo integrar con Servicios de AWS
- Integración a otros productos y servicios

Cómo integrar con Servicios de AWS

Mediante el uso de documentos de Systems Manager Command (documentos de SSM) y de manuales de procedimientos de Automation, puede usar AWS Systems Manager para integrarse a los Servicios de AWS. Para obtener más información acerca de estos recursos, consulte Documentos de AWS Systems Manager.

Systems Manager está integrado con los siguientes Servicios de AWS.

Computación

Amazon Elastic Compute Cloud (Amazon EC2)	Amazon EC2 proporciona capacidad informática escalable en la Nube de AWS. El uso de Amazon EC2 elimina la necesidad de invertir inicialmente en hardware, de manera que puede desarrollar e implementar aplicaciones en menos tiempo. Puede usar Amazon EC2 para lanzar tantos servidores virtuales como necesite, configurar la seguridad y las redes, y administrar el almacenamiento. Systems Manager le permite realizar varias tareas en las instancias EC2. Por ejemplo, puede lanzar, configurar, administrar y mantener sus instancias EC2, así como conectarse a ellas de forma segura y solucionar sus problemas. También puede utilizar

	Systems Manager para implementar software, determinar el estado de conformidad y recopilar el inventario de las instancias EC2. Más información • Trabajo con nodos administrados • AWS Systems Manager State Manager • AWS Systems Manager Run Command • AWS Systems Manager Patch Manager • AWS Systems Manager Session Manager • AWS Systems Manager Distributor • Conformidad de AWS Systems Manager • Inventario de AWS Systems Manager
Amazon EC2 Auto Scaling	Auto Scaling (escalado automático) lo ayuda a asegurarse de contar con la cantidad correcta de instancias EC2 disponibles para gestionar la carga de su aplicación. Crea colecciones de instancias EC2, denominadas grupo de escalado automático. Systems Manager le permite automatizar los procedimientos comunes, como la aplicació n de revisiones a las Amazon Machine Image (AMI) que se utilizan en la plantilla de Auto Scaling para el grupo de escalado automático. Más información Actualización de AMIs para grupos de escalado automático

Amazon Elastic Container Service (Amazon ECS)	Amazon ECS es un servicio de administración de contenedores rápido y de alta escalabilidad que le permite ejecutar, detener y administrar contenedores de Docker en un clúster. Systems Manager le permite administrar instancias de contenedor de forma remota e incorporar información confidencial a sus contenedores mediante el almacenamiento en parámetros de Parameter Store, una herramien ta de Systems Manager, y, a continuación, hacer referencia a ellos en la definición de contenedor. Más información • Administración remota de instancias de contenedor mediante AWS Systems Manager • Specifying sensitive data using Systems Manager Parameter Store (Especifi cación de información confidencial mediante el almacén de parámetros de Systems Manager)

AWS Lambda	Lambda es un servicio informático que permite ejecutar código sin aprovisionar ni administr ar servidores. Lambda ejecuta su código solo cuando es necesario y escala de manera automática, desde unas pocas solicitudes por día hasta miles por segundo. Systems Manager le permite utilizar las funciones de Lambda dentro del manual de procedimientos de Automation mediante la acción `aws:invokeLambdaFunction` . Si desea usar parámetros de Parameter Store en funciones AWS Lambda, puede usar la extensión de Lambda para secretos y parámetros de AWS para recuperar los valores de los parámetros y almacenarlos en la caché para usarlos en el futuro. Más información Actualice un golden AMI mediante la Automatio n, AWS Lambda, y Parameter Store Cómo utilizar parámetros Parameter Store en funciones AWS Lambda

Internet de las cosas (IoT)

Dispositivos de núcleo de AWS IoT Greengrass	AWS IoT Greengrass es un servicio en la nube y de tiempo de ejecución de borde de IoT de código abierto que lo ayuda a crear, implementar y administrar aplicaciones de IoT en los dispositivos. Systems Manager ofrece compatibilidad nativa para dispositivos de núcleo de AWS IoT Greengrass.

	Más información Administración de dispositivos periféricos con Systems Manager
Dispositivos de núcleo de AWS IoT	AWS IoT proporciona los servicios en la nube que conectan los dispositivos IoT a otros dispositivos y servicios en la nube de AWS. AWS IoT proporciona software para dispositivos que puede ayudarlo a integrar los dispositivos IoT en soluciones basadas en AWS IoT. Si los dispositivos se pueden conectar a AWS IoT, AWS IoT puede conectarlos a los servicios en la nube que proporciona AWS. Systems Manager admite dispositivos de núcleo de AWS IoT siempre que esos dispositivos estén configurados como nodos administrados en un entorno híbrido y multinube. Más información Administrador de nodos en entornos híbridos y multinube con Systems Manager

Almacenamiento

Amazon Simple Storage Service (Amazon S3)	Amazon S3 es un servicio de almacenamiento para Internet. Está diseñado para facilitar a los desarrolladores la informática a escala de la Web. Amazon S3 tiene una interfaz de servicios web simple que puede utilizar para almacenar y recuperar cualquier cantidad de datos, en cualquier momento y desde cualquier parte de la web. Systems Manager le permite ejecutar scripts remotos y documentos de SSM que estén

	almacenados en Amazon S3. Distributor, una herramienta de AWS Systems Manager, utiliza Amazon S3 para almacenar paquetes. También puede enviar a Amazon S3 resultado s de Run Command y Session Manager, herramientas de AWS Systems Manager. Más información • Ejecución de scripts desde Amazon S3 • Ejecución de documentos de desde ubicaciones remotas • AWS Systems Manager Distributor • Registro de los datos de la sesión con Amazon S3 (consola)

Herramientas para desarrolladores

AWS CodeBuild	CodeBuild es un servicio de compilación completamente administrado en la nube. CodeBuild compila su código fuente, ejecuta pruebas de unidad y produce artefactos listos para implementarse. CodeBuild elimina la necesidad de aprovisionar, administrar y escalar sus propios servidores de compilación. Parameter Store le permite almacenar información confidencial para sus especific aciones y proyectos de compilación. Más información • Referencia de la especificación de compilaci ón de CodeBuild • Creación de un proyecto de compilación en AWS CodeBuild

AWS CDK	AWS Cloud Development Kit (AWS CDK) es un marco para definir la infraestructura de la nube como código, con lenguajes de programación, e implementarla a través de AWS CloudFormation. Application Manager le permite ver sus componentes de CDK agrupados como aplicaciones, ver la estructura de la aplicación, incluidos los recursos subyacentes, ver las alertas, investigar y solucionar los problemas operativos y realizar un seguimiento de los costes en la consola de Application Manager. Más información • Visualización de información general detallada acerca de una aplicación • Visualización de recursos de aplicaciones

Seguridad, identidad y conformidad

AWS Identity and Access Management (IAM)	IAM es un servicio web que lo ayuda a controlar de forma segura el acceso a los recursos de AWS. Utilice IAM para controlar quién está autenticado (ha iniciado sesión) y autorizado (tiene permisos) para utilizar recursos. Systems Manager le permite controlar el acceso a los servicios mediante IAM. Más información • Cómo funciona AWS Systems Manager con IAM

	• Acciones, recursos y claves de condiciones para AWS Systems Manager • Configuración de permisos de instancia requeridos para Systems Manager
AWS Secrets Manager	Secrets Manager permite una administración más fácil de los secretos. Los secretos pueden ser credenciales de base de datos, contraseñ as, claves de API de terceros e incluso texto arbitrario. Parameter Store le permite recuperar secretos de Secrets Manager cuando utiliza otros Servicios de AWS que ya admiten referencias a los parámetros de Parameter Store. Más información Referencia a los secretos de AWS Secrets Manager desde los parámetros de Parameter Store

AWS Security Hub	Security Hub le brinda una vista integral de sus alertas de seguridad de alta prioridad y del estado de conformidad en todas las Cuentas de AWS. Security Hub agrega, organiza y prioriza las alertas de seguridad o los resultados de varios Servicios de AWS. Cuando se activa la integración entre Security Hub y Patch Manager, una herramienta de AWS Systems Manager, Security Hub supervisa el estado de la aplicación de revisiones de sus flotas desde el punto de vista de la seguridad. Los detalles de conformidad de la aplicación de revisiones se exportan automáticamente a Security Hub. Esto le permite utilizar una vista única para monitorear de forma centralizada el estado de conformidad de la aplicación de revisiones y realizar un seguimiento de otros resultados de seguridad. Puede recibir alertas cuando los nodos de la flota se aparten de la conformidad de la aplicación de revisiones y revisar los resultados de esta conformidad en la consola de Security Hub. También puede integrar Security Hub a Explorer y OpsCenter, herramientas de AWS Systems Manager. La integración a Security Hub le permite recibir resultados de Security Hub en Explorer y OpsCenter. Los resultados de Security Hub proporcionan información de seguridad que puede utilizar en Explorer y OpsCenter para agregar y tomar medidas en cuanto a sus problemas operativos, de seguridad y de rendimiento en AWS Systems Manager.

	El uso de Security Hub conlleva un cargo. Para obtener más información, consulte Precios de Security Hub. Más información • Recepción de resultados de AWS Security Hub en Explorer • Comprensión de la integración de OpsCenter con AWS Security Hub • Integración de Patch Manager con AWS Security Hub

Criptografía y PKI

AWS Key Management Service (AWS KMS)	AWS KMS es un servicio administrado que le permite crear y controlar las claves administr adas por el cliente, las claves de cifrado que se utilizan para cifrar los datos. Systems Manager le permite utilizar AWS KMS para crear parámetros `SecureString` y para cifrar los datos de las sesiones de Session Manager. Más información • Cifrado de AWS KMS para los parámetros SecureString del Parameter Store de AWS Systems Manager • Activación del cifrado de datos de sesión con claves de KMS (consola)

Administración y gobierno

AWS CloudFormation	AWS CloudFormation es un servicio que lo ayuda a modelar y configurar los recursos de Amazon Web Services, de manera que pueda dedicar menos tiempo a la administración de dichos recursos y más tiempo a centrarse en las aplicaciones que se ejecutan en AWS. Parameter Store es una fuente de referencias dinámicas. Las referencias dinámicas proporcionan una forma consistente y sólida de especificar valores externos que estén almacenados y se administren en otros servicios de sus plantillas de pila de AWS CloudFormation. Más información Uso de referencias dinámicas para especificar valores de plantillas
AWS CloudTrail	CloudTrail es un servicio de Servicio de AWS que lo ayuda a autorizar la gobernanza, la conformidad y la auditoría de las operaciones y de los riesgos de su Cuenta de AWS. Las acciones que realiza un usuario, rol o servicio de Servicio de AWS se registran como eventos en CloudTrail. Los eventos incluyen las acciones llevadas a cabo en la AWS Management Console, la AWS Command Line Interface (AWS CLI), los AWS SDK y las API. Systems Manager se integra con CloudTrail, que registra la mayoría de las llamadas a la API de Systems Manager como eventos. Estas incluyen las llamadas a la API procedentes de

	la consola de Systems Manager y las llamadas dirigidas a las API de Systems Manager. Más información Registro de llamadas a la API de AWS Systems Manager con AWS CloudTrail
Registros de Amazon CloudWatch	Registros de Amazon CloudWatch le permite centralizar los registros de todos los sistemas, las aplicaciones y los Servicios de AWS que utilice. Esto le permite consultarlos, buscar códigos de error o patrones específicos, filtrarlos en función de campos específicos o archivarlos de forma segura para análisis futuros. Systems Manager admite el envío de registros de SSM Agent, Run Command y Session Manager a los Registros de CloudWatch. Más información • Envío de registros de nodos a los Registros de CloudWatch (agente de CloudWatch) unificado • Configuración de Registros de Amazon CloudWatch para Run Command • Registro de los datos de la sesión con los Registros de Amazon CloudWatch (consola)

Amazon EventBridge	EventBridge entrega una secuencia de eventos de sistema casi en tiempo real que describe los cambios producidos en los recursos de Amazon Web Services. Mediante reglas sencillas que puede configurar rápidamente, puede asignar los eventos y dirigirlos a uno o más flujos o funciones de destino. EventBridge toma conocimiento de los cambios operativo s a medida que se producen. EventBridge responde a estos cambios operativos y toma las medidas correctivas necesarias. Estas acciones incluyen el envío de mensajes para responder al entorno, la activación de las funciones y el registro de la información de estado. Systems Manager tiene varios eventos compatibles con EventBridge, lo que le permite realizar acciones en función del contenido de dichos eventos. Más información Monitoreo de eventos de Systems Manager con Amazon EventBridge **Note** Amazon EventBridge es la forma preferida de administrar sus eventos. CloudWatch Events y EventBridge son el mismo servicio subyacente y la misma API, pero EventBridge ofrece más características. Los cambios que realice en CloudWatch o EventBridge se reflejarán en cada consola. Para

	más información, consulte la Guía del usuario de Amazon EventBridge.
AWS Config	AWS Config proporciona una vista detallada de la configuración de los recursos de AWS de su Cuenta de AWS. Esto incluye cómo se relacionan los recursos entre sí y cómo se han configurado. Esto le permite ver cómo cambian las configuraciones y las relaciones con el paso del tiempo. Systems Manager está integrado a AWS Config, lo que proporciona varias reglas que lo ayudan a obtener visibilidad de sus instancias EC2. Estas reglas lo ayudan a identificar las instancias EC2 administradas por Systems Manager, las configuraciones del sistema operativo, las actualizaciones de nivel del sistema, las aplicaciones instaladas, las configuraciones de red y mucho más. Más información • Tipos de recursos de AWS Config admitidos • Registro de la configuración de software para instancias administradas • Visualización del seguimiento de cambios y del historial de Inventory

AWS Trusted Advisor	Trusted Advisor es una herramienta online que proporciona orientación en tiempo real para ayudarlo a aprovisionar sus recursos de acuerdo con las prácticas recomendadas de AWS. Systems Manager aloja a Trusted Advisor, y es posible ver los datos de Trusted Advisor en Explorer. Más información • AWS Systems Manager Explorer • Introducción a AWS Trusted Advisor

AWS Organizations	Organizations es un servicio de administración de cuentas que le permite agrupar varias Cuentas de AWS en una organización que usted crea y administra de manera centralizada. Organizations incluye capacidades de facturación unificada y administración de cuentas que le permiten satisfacer mejor las necesidades de presupuestos, seguridad y conformidad de su empresa. La integración de Change Manager, una herramienta de AWS Systems Manager, a Organizations permite utilizar una cuenta de administrador delegado para administrar solicitudes de cambio, plantillas de cambio y aprobaciones para toda la organización a través de esta única cuenta. La integración de Organizations a Inventario, una herramienta de AWS Systems Manager, y Explorer le permite agregar los datos de operaciones (OpsData) y de inventario correspondientes a varias Regiones de AWS y Cuentas de AWS. La integración de Quick Setup, una herramienta de AWS Systems Manager, a Organizations automatiza las tareas comunes de configuración de servicios e implementa configuraciones de servicios conforme a las prácticas recomendadas en las unidades organizativas (OU).

Redes y entrega de contenido

AWS PrivateLink	AWS PrivateLink le permite conectar de forma privada su nube virtual privada (VPC) a los servicios admitidos de Servicios de AWS y a los puntos de conexión de VPC, sin la necesidad de contar con una puerta de enlace de Internet, ni un dispositivo NAT, ni una conexión de VPN ni una conexión de AWS Direct Connect. Systems Manager admite nodos administr ados que se conectan a las API de Systems Manager mediante AWS PrivateLink. Esto mejora la posición de seguridad de los nodos administrados porque AWS PrivateLink restringe todo el tráfico de red entre los nodos administrados, Systems Manager y Amazon EC2 a la red de Amazon. Esto significa que no es necesario que los nodos administrados tengan acceso a Internet. Más información Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager

Análisis

Amazon Athena	Athena es un servicio interactivo de consultas que le permite analizar datos directamente en Amazon Simple Storage Service (Amazon S3) usando SQL estándar. Con unas pocas acciones en la AWS Management Console, puede apuntar Athena a los datos almacenad

	os en Amazon S3 y comenzar a utilizar SQL estándar para ejecutar consultas de una única vez y obtener resultados en cuestión de segundos. Systems Manager Inventory se integra a Athena para ayudarlo a consultar los datos de inventario de varias Regiones de AWS y Cuentas de AWS. La integración de Athena utiliza la sincronización de datos de recursos, de modo que pueda ver los datos de inventario de todos los nodos administrados en la página Detailed View (Vista detallada) de la consola de Systems Manager Inventory. Más información • Consulta de datos de Inventory de varias regiones y cuentas • Explicación: cómo utilizar la sincronización de datos de recursos para agregar datos de inventario
AWS Glue	AWS Glue es un servicio completamente administrado de ETL (extracción, transform ación y carga) con el que resulta más rentable y sencillo categorizar los datos, limpiarlos, enriquecerlos y moverlos de manera fiable entre distintos almacenes y flujos de datos. Systems Manager utiliza AWS Glue para rastrear los datos de Inventory en su bucket de S3. Más información Consulta de datos de Inventory de varias regiones y cuentas

Amazon QuickSight	QuickSight es un servicio de análisis empresariales que puede utilizar para crear visualizaciones, llevar a cabo análisis únicos y obtener información empresarial a partir de sus datos. Puede detectar automáticamente los orígenes de datos de AWS y también trabajar con sus orígenes de datos. La sincronización de datos de recursos de Systems Manager envía los datos de inventario recopilados de todos los nodos administrados a un solo bucket de S3. Puede utilizar QuickSight para consultar y analizar los datos agregados. Más información • Cómo crear una sincronización de datos de recursos para Inventory • Explicación: cómo utilizar la sincronización de datos de recursos para agregar datos de inventario

Integración de aplicaciones

Amazon Simple Notification Service (Amazon SNS)	Amazon SNS es un servicio web que coordina y administra la entrega o el envío de mensajes a los puntos de enlace o los clientes suscritos. Systems Manager genera estados para varios servicios que pueden registrarse mediante las notificaciones de Amazon SNS. Más información • Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS

	• Configuración de notificaciones o activación de acciones en función de los eventos de Parameter Store

AWS Management Console

AWS Resource Groups	Resource Groups organiza sus recursos de AWS. Los grupos de recursos facilitan la administración, monitorización y automatización de tareas en grandes cantidades de recursos al mismo tiempo. Los tipos de recursos de Systems Manager, como los nodos administrados, los documentos de SSM, los periodos de mantenimiento, los parámetros de Parameter Store y las líneas de base de revisiones se pueden agregar a los grupos de recursos. Más información ¿Qué son los AWS Resource Groups?

Temas

- Ejecución de scripts desde Amazon S3
- Referencia a los secretos de AWS Secrets Manager desde los parámetros de Parameter Store
- Cifrado de AWS KMS para los parámetros SecureString del Parameter Store de AWS Systems Manager
- Uso de parámetros del Parameter Store en Amazon Elastic Kubernetes Service
- Cómo utilizar parámetros Parameter Store en funciones AWS Lambda

Ejecución de scripts desde Amazon S3

En esta sección, se describe cómo descargar y ejecutar scripts desde Amazon Simple Storage Service (Amazon S3). El siguiente tema incluye información y terminología relacionadas a Amazon S3. Para obtener más información sobre Amazon S3, consulte ¿Qué es Amazon S3?. Puede ejecutar diversos tipos de scripts, incluidos los cuadernos de trabajos de Ansible, Python, Ruby, Shell y PowerShell.

También puede descargar un directorio en el que se incluyen varios scripts. Cuando ejecuta el script principal en el directorio, AWS Systems Manager también ejecuta los scripts a los que se hace referencia y se incluyen en el directorio.

Tenga en cuenta los siguientes detalles importantes acerca de la ejecución de scripts desde Amazon S3:

- Systems Manager no comprueba que el script pueda ejecutarse en un nodo. Antes de descargar y ejecutar el script, verifique que el software necesario esté instalado en el nodo. O bien, puede crear un documento compuesto que instale el software mediante Run Command o State Manager, herramientas de AWS Systems Manager, y, a continuación, descargue y ejecute el script.
- Compruebe que su usuario, rol o grupo tenga los permisos de AWS Identity and Access Management (IAM) que se necesitan para la lectura del bucket de S3.
- Asegúrese de que el perfil de instancias de las instancias de Amazon Elastic Compute Cloud (Amazon EC2) tenga los permisos `s3:ListBucket` y `s3:GetObject`. Si el perfil de instancia no tiene estos permisos, el sistema no puede descargar el script del bucket de S3. Para obtener más información, consulte Uso de perfiles de instancias en la Guía del usuario de IAM.

Ejecutar scripts de shell desde Amazon S3

La siguiente información incluye procedimientos que sirven de ayuda para ejecutar scripts desde Amazon Simple Storage Service (Amazon S3) utilizando la consola de AWS Systems Manager o la AWS Command Line Interface (AWS CLI). Aunque en los ejemplos se utilizan scripts de shell, se pueden sustituir por otros tipos de scripts.

Ejecutar un script de shell desde Amazon S3 (consola)

Ejecutar un script de shell desde Amazon S3

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.

2. En el panel de navegación, elija Run Command.
3. Elija Run command (Ejecutar comando).
4. En la lista Command document (Documento de Command), elija **`AWS-RunRemoteScript`**.
5. En Command parameters haga lo siguiente:

   - En Source Type, seleccione S3.
   - En el cuadro de texto Source Info (Información de la fuente), ingrese la información requerida para acceder a la fuente, con el siguiente formato. Reemplace cada *`example resource placeholder`* con su propia información.

     > **Note**
     >
     > Sustituya https://s3.*`aws-api-domain`* por la URL de su bucket. Puede copiar la URL de su bucket en Amazon S3 en la pestaña Objects (Objetos).

     ```
     {"path":"https://s3.aws-api-domain/path to script"}
     ```

     A continuación se muestra un ejemplo.

     ```
     {"path":"https://amzn-s3-demo-bucket.s3.us-east-2.amazonaws.com/scripts/shell/
     helloWorld.sh"}
     ```

   - En el campo Command Line (Línea de comandos), ingrese los parámetros para la ejecución de scripts. A continuación se muestra un ejemplo.

     ```
     helloWorld.sh argument-1 argument-2
     ```

   - (Opcional) En el campo Working Directory (Directorio de trabajo), ingrese el nombre de un directorio del nodo en el que desee descargar y ejecutar el script.
   - (Opcional) En Execution Timeout, especifique el número de segundos que esperará el sistema antes de fallar en la ejecución del comando de script.
6. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

> **Tip**
>
> Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

7. En Otros parámetros:

   - En Comentario, ingrese la información acerca de este comando.
   - En Tiempo de espera (segundos), especifique el número de segundos que tiene que esperar el sistema antes de indicar que se ha producido un error en la ejecución del comando general.

8. En Rate control (Control de velocidad):

   - En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

     > **Note**
     >
     > Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

   - En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

9. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario

de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

10. En la sección Notificaciones de SNS, seleccione la casilla de verificación Habilitar notificaciones de SNS si desea recibir notificaciones sobre el estado de ejecución de los comandos.

    Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.

11. Seleccione Ejecutar.

Ejecutar un script de shell desde Amazon S3 (línea de comandos)

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Ejecute el siguiente comando. Reemplace cada *example resource placeholder* con su propia información.

   > **Note**
   >
   > Sustituya https://s3.*aws-api-domain*/*script path* por la URL de su bucket. Puede copiar la URL de su bucket en Amazon S3 en la pestaña Objects (Objetos).

   Linux & macOS

   ```
   aws ssm send-command \
       --document-name "AWS-RunRemoteScript" \
       --output-s3-bucket-name "amzn-s3-demo-bucket" \
       --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 "amzn-s3-demo-bucket" ^
    --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 "amzn-s3-demo-bucket" `
    -OutputS3KeyPrefix "key-prefix" `
    -Target @{Key="InstanceIds";Values=@("instance-id")} `
    -Parameter @{
        sourceType = "S3";
        sourceInfo = '{"path": "s3://bucket-name/path/to/script"}';
        commandLine = "script name and arguments"
    }
```

Referencia a los secretos de AWS Secrets Manager desde los parámetros de Parameter Store

AWS Secrets Manager ayuda a organizar y administrar los datos de configuración importantes, como las credenciales, las contraseñas y las claves de licencia. Parameter Store, una herramienta de AWS Systems Manager, se integra en Secrets Manager, lo que le permite recuperar secretos de Secrets Manager cuando se utilizan otros Servicios de AWS que ya admiten las referencias a los parámetros de Parameter Store. Estos servicios incluyen Amazon Elastic Compute Cloud (Amazon EC2), Amazon Elastic Container Service (Amazon ECS), AWS Lambda, AWS CloudFormation, AWS CodeBuild, AWS CodeDeploy y otras herramientas de Systems Manager. Si se utiliza Parameter Store para hacer referencia a los secretos de Secrets Manager, se crea un proceso consistente y seguro para llamar y utilizar secretos y datos de referencia en el código y los scripts de configuración.

Para obtener más información acerca de Secrets Manager, consulte ¿Qué es AWS Secrets Manager? en la Guía del usuario de AWS Secrets Manager.

Restricciones

Tenga en cuenta las siguientes restricciones cuando utilice Parameter Store para hacer referencia a los secretos de Secrets Manager:

- Solo se pueden recuperar secretos de Secrets Manager utilizando las operaciones GetParameter y GetParameters de la API. Las operaciones de modificación y de consulta avanzada de la API, como, por ejemplo, DescribeParameters y GetParametersByPath, no se admiten en Secrets Manager.
- Puede utilizar la AWS Command Line Interface (AWS CLI), las AWS Tools for Windows PowerShell y los SDK para recuperar un secreto mediante Parameter Store.
- Los secretos de Secrets Manager en Parameter Store deben tener un prefijo de `/aws/reference/secretsmanager`. A continuación se muestran algunos ejemplos:

  ```
  /aws/reference/secretsmanager/CFCreds1
  ```

  ```
  /aws/reference/secretsmanager/myapp/db/password
  ```

- Parameter Store respeta las políticas de AWS Identity and Access Management (IAM) adjuntas a los secretos de Secrets Manager. Por ejemplo, si el usuario 1 no tiene acceso al secreto A, el usuario 1 no puede recuperar el secreto A mediante Parameter Store.
- Los parámetros que hacen referencia a los secretos de Secrets Manager no pueden utilizar las características de historial ni de control de versiones de Parameter Store.
- Parameter Store respeta las fases de la versión de Secrets Manager. Si se hace referencia a una fase de la versión, se utilizan letras, números, puntos (.), guiones (-) o guiones bajos (_). Si se especifica cualquier otro símbolo en la fase de la versión, la referencia producirá un error.

Cómo hacer referencia a un secreto de Secrets Manager con Parameter Store

En el siguiente procedimiento, se describe cómo hacer referencia a un secreto de Secrets Manager utilizando las API de Parameter Store. El procedimiento hace referencia a otros procedimientos de la Guía del usuario de AWS Secrets Manager.

> **Note**
>
> Antes de comenzar, compruebe que tiene permiso para hacer referencia a los secretos de Secrets Manager en los parámetros de Parameter Store. Si tiene permisos de administrador en Secrets Manager y Systems Manager, puede hacer referencia o recuperar secretos utilizando las API de Parameter Store. Si hace referencia a un secreto de Secrets Manager en un parámetro de Parameter Store y no tiene permiso para acceder a ese secreto, la referencia producirá un error. Para obtener más información, consulte Autenticación y control de acceso para AWS Secrets Manager en la Guía del usuario de AWS Secrets Manager.

> **Important**
>
> Parameter Store funciona como un servicio de transferencia para las referencias a los secretos de Secrets Manager. Parameter Store no conserva los datos ni los metadatos de los secretos. La referencia es sin estado.

Para hacer referencia a un secreto Secrets Manager mediante el uso de Parameter Store

1. Cree un secreto en Secrets Manager. Para obtener más información, consulte Cree y administre secretos con AWS Secrets Manager.
2. Haga referencia a un secreto utilizando la AWS CLI, AWS Tools for Windows PowerShell o el SDK. Cuando haga referencia a un secreto de Secrets Manager, el nombre debe comenzar por la siguiente ruta reservada: `/aws/reference/secretsmanager/`. Cuando se especifica esta ruta, Systems Manager sabe que tiene que recuperar el secreto desde Secrets Manager en lugar de desde Parameter Store. A continuación, se presentan algunos ejemplos de nombres que hacen referencia correctamente a los secretos de Secrets Manager, `CFCreds1` y `DBPass`, mediante Parameter Store.

   - `/aws/reference/secretsmanager/CFCreds1`
   - `/aws/reference/secretsmanager/DBPass`

   A continuación, se muestra un ejemplo de código Java que hace referencia a una clave de acceso y una clave secreta que están almacenadas en Secrets Manager. En este ejemplo de código, se configura un cliente de Amazon DynamoDB. El código recupera los datos de configuración y las credenciales de Parameter Store. Los datos de configuración se almacenan

como un parámetro de cadena en Parameter Store y las credenciales se almacenan en Secrets Manager. Aunque los datos de configuración y las credenciales se almacenan en servicios independientes, es posible acceder a ambos conjuntos de datos desde Parameter Store mediante la 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();
```

```
}
```

A continuación, se muestran algunos ejemplos de la AWS CLI. Use el comando `aws secretsmanager list-secrets` para encontrar los nombres de sus secretos.

AWS CLI Ejemplo 1 de la : referencia mediante el nombre del secreto

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
```

El comando devuelve información similar a la siguiente.

```
{
    "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 Ejemplo 2 de la : referencia que incluye el ID de versión

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
```

El comando devuelve información similar a la siguiente.

```
{
    "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 Ejemplo 3 de la : referencia que incluye la fase de la versión

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
```

El comando devuelve información similar a la siguiente.

```
{
    "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",
}
```

Cifrado de AWS KMS para los parámetros SecureString del Parameter Store de AWS Systems Manager

Con el Parameter Store de AWS Systems Manager, puede crear parámetros SecureString, que son parámetros que tienen un nombre de parámetro en texto no cifrado y un valor de parámetro cifrado. El Parameter Store utiliza AWS KMS para cifrar y descifrar los valores de los parámetros `SecureString`.

Con el Parameter Store, puede crear, almacenar y administrar datos como parámetros con valores. Puede crear un parámetro en Parameter Store y utilizarlo en varias aplicaciones y servicios de acuerdo con las políticas y los permisos que diseñe. Cuando necesite cambiar el valor de un parámetro, cambiará una instancia, y no tendrá que administrar una modificación propensa a errores en varios orígenes. Parameter Store admite una estructura jerárquica para los nombres de los parámetros, por lo que puede asignar un parámetro para usos específicos.

Para administrar información confidencial, puede crear parámetros `SecureString`. El Parameter Store utiliza AWS KMS keys para cifrar los valores de los parámetros `SecureString` cuando estos se crean o se modifican. También utiliza claves KMS para descifrar los valores de los parámetros cuando se obtiene acceso a ellos. Puede utilizar la Clave administrada de AWS que el Parameter Store crea para su cuenta o especificar su propia clave administrada por el cliente.

> **Important**
>
> El Parameter Store solo admite claves de KMS simétricas. No se puede utilizar una clave KMS asimétrica para cifrar los parámetros. Para obtener ayuda para determinar si una clave de KMS es simétrica o asimétrica, consulte Identificación de distintos tipos de claves en la Guía para desarrolladores de AWS Key Management Service.

Parameter Store admite dos niveles de `SecureString` parámetros: estándar y avanzados. Los parámetros estándar, que no pueden superar los 4096 bytes, se cifran y descifran directamente con la clave KMS que especifique. Para cifrar y descifrar parámetros `SecureString` avanzados, Parameter Store utiliza el cifrado de sobre con el AWS Encryption SDK. Puede convertir un parámetro `SecureString` estándar en un parámetro avanzado, pero no puede convertir un parámetro avanzado en uno estándar. Para obtener más información acerca de la diferencia entre los parámetros `SecureString` estándar y avanzados, consulte Administración de niveles de parámetros.

Temas

- Protección de los parámetros SecureString estándar
- Protección de parámetros Secure String avanzados
- Configurar permisos para cifrar y descifrar valores de parámetros
- Contexto de cifrado de Parameter Store
- Solución de problemas de las claves de KMS en el Parameter Store

Protección de los parámetros SecureString estándar

Parameter Store no realiza ninguna operación criptográfica. En lugar de ello, utiliza AWS KMS para cifrar y descifrar los valores de los parámetros `SecureString`. Cuando se crea o cambia el valor de un parámetro `SecureString` estándar, el Parameter Store llama a la operación Encrypt de AWS KMS. Esta operación utiliza directamente una clave KMS de cifrado simétrica para cifrar el valor del parámetro en lugar de utilizar la clave KMS para generar una clave de datos.

Puede seleccionar la clave de KMS que utiliza el Parameter Store para cifrar el valor del parámetro. Si no especifica una clave de KMS, el Parameter Store utiliza la Clave administrada de AWS que Systems Manager crea automáticamente en su cuenta. Esta clave KMS tiene el alias `aws/ssm`.

Si desea ver la clave KMS `aws/ssm` predeterminada de su cuenta, utilice la operación DescribeKey de la API de AWS KMS. El siguiente ejemplo utiliza el comando `describe-key` de la AWS Command Line Interface (AWS CLI) con el nombre de alias `aws/ssm`.

```
aws kms describe-key \
    --key-id alias/aws/ssm
```

Para crear un parámetro `SecureString` estándar, utilice la operación PutParameter de la API de Systems Manager. Omita el parámetro `Tier` o especifique un valor de `Standard`, que es el valor predeterminado. Incluya un parámetro `Type` con el valor `SecureString`. Para especificar una clave KMS, utilice el parámetro `KeyId`. El valor predeterminado es la Clave administrada de AWS de su cuenta, `aws/ssm`.

A continuación, el Parameter Store llama a la operación `Encrypt` de AWS KMS con la clave de KMS y el valor del parámetro en texto no cifrado. AWS KMS devuelve el valor del parámetro cifrado y el Parameter Store lo almacena junto con el nombre del parámetro.

El siguiente ejemplo utiliza el comando put-parameter de Systems Manager y su parámetro `--type` en la AWS CLI para crear un parámetro `SecureString`. Como el comando omite los parámetros

opcionales `--tier` y `--key-id`, el Parameter Store crea un parámetro `SecureString` estándar y lo cifra con la Clave administrada de AWS.

```
aws ssm put-parameter \
    --name MyParameter \
    --value "secret_value" \
    --type SecureString
```

El siguiente ejemplo similar utiliza el parámetro `--key-id` para especificar una clave KMS administrada por el cliente. En el ejemplo se utiliza un ID de clave KMS para identificar la clave KMS, pero se puede utilizar cualquier identificador de clave KMS válido. Como el comando omite el parámetro `Tier` (`--tier`), Parameter Store crea un parámetro `SecureString` estándar, no uno avanzado.

```
aws ssm put-parameter \
    --name param1 \
    --value "secret" \
    --type SecureString \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Cuando se obtiene un parámetro `SecureString` de Parameter Store, su valor está cifrado. Para obtener un parámetro, utilice la operación GetParameter de la API de Systems Manager.

El siguiente ejemplo utiliza el comando get-parameter de Systems Manager en la AWS CLI para obtener el parámetro `MyParameter` del Parameter Store sin descifrar su valor.

```
aws ssm get-parameter --name MyParameter
```

```
{
    "Parameter": {
        "Type": "SecureString",
        "Name": "MyParameter",
        "Value": "AQECAHgnOkMROh5LaLXkA4j0+vYi6tmM17Lg"
    }
}
```

Para descifrar el valor del parámetro antes de devolverlo, establezca el parámetro `WithDecryption` de `GetParameter` en `true`. Al utilizar `WithDecryption`, el Parameter Store llama a la operación Decrypt de AWS KMS en su nombre para descifrar el valor del parámetro. En

consecuencia, la solicitud `GetParameter` devuelve el parámetro con un valor en texto no cifrado, tal y como se muestra en el siguiente ejemplo.

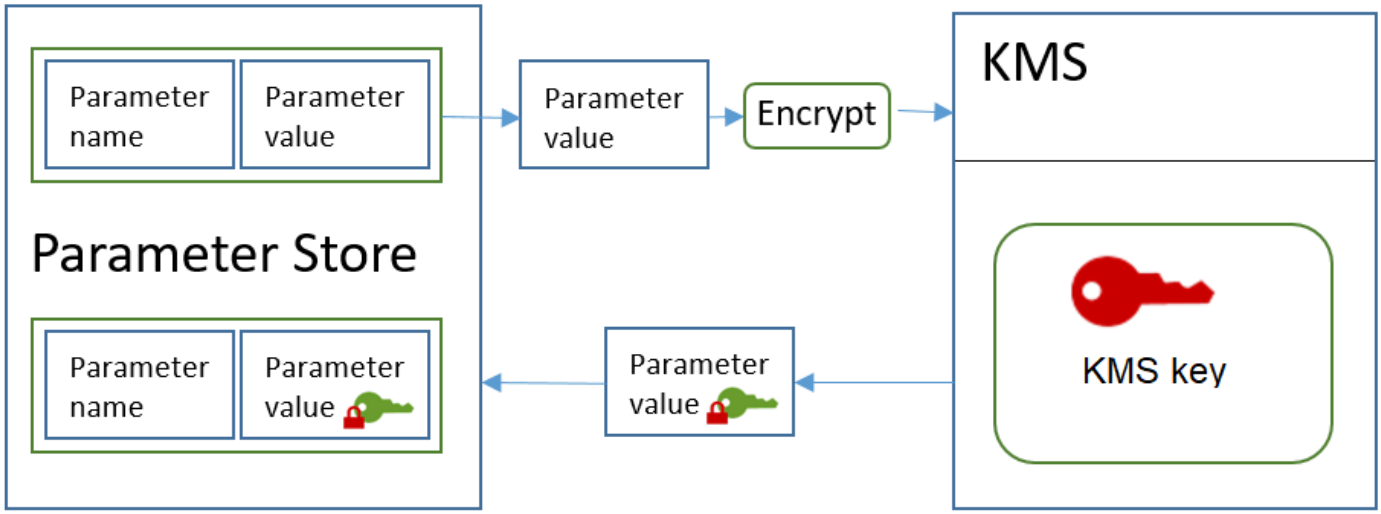
```
aws ssm get-parameter \
    --name MyParameter \
    --with-decryption
```

```
{
    "Parameter": {
        "Type": "SecureString",
        "Name": "MyParameter",
        "Value": "secret_value"
    }
}
```

En el siguiente flujo de trabajo se muestra cómo el Parameter Store utiliza una clave de KMS para cifrar y descifrar un parámetro `SecureString` estándar.

Cifrar un parámetro estándar

1. Cuando se utiliza `PutParameter` para crear un parámetro `SecureString`, Parameter Store envía una solicitud `Encrypt` a AWS KMS. Dicha solicitud incluye el valor del parámetro en texto no cifrado, la clave de KMS que ha elegido y el contexto de cifrado del Parameter Store. Durante la transmisión a AWS KMS, el valor en texto no cifrado del parámetro `SecureString` está protegido por Transport Layer Security (TLS).
2. AWS KMS cifra el valor del parámetro con la clave KMS y el contexto de cifrado especificados. Devuelve el texto cifrado a Parameter Store, que almacena el nombre del parámetro y su valor cifrado.



Descifrar un parámetro estándar

1. Cuando se incluye el parámetro `WithDecryption` en una solicitud `GetParameter`, Parameter Store envía una solicitud `Decrypt` a AWS KMS con el valor del parámetro `SecureString` cifrado y el contexto de cifrado de Parameter Store.
2. AWS KMS utiliza la misma clave KMS y el contexto de cifrado proporcionado para descifrar el valor cifrado. Devuelve el valor del parámetro en texto no cifrado a Parameter Store. Durante la transmisión, los datos en texto no cifrado están protegidos por TLS.
3. Parameter Store devuelve el valor del parámetro en texto no cifrado en la respuesta de `GetParameter`.

Protección de parámetros Secure String avanzados

Cuando se utiliza `PutParameter` para crear un parámetro `SecureString` avanzado, el Parameter Store usa el cifrado de sobre con el AWS Encryption SDK y una AWS KMS key de cifrado simétrica para proteger el valor del parámetro. Cada valor del parámetro avanzado se cifra con una clave de datos única y la clave de datos se cifra con una clave KMS. Puede usar la Clave administrada de AWS de la cuenta (`aws/ssm`) o cualquier clave administrada por el cliente.

El AWS Encryption SDK es una biblioteca cliente de código abierto que le ayuda a cifrar y descifrar datos mediante estándares y prácticas recomendadas del sector. Es compatible con varias plataformas y varios lenguajes de programación, incluida una interfaz de línea de comandos. Puede ver el código fuente y contribuir a su desarrollo en GitHub.

Para cada valor de un parámetro `SecureString`, Parameter Store llama al AWS Encryption SDK para descifrar el valor del parámetro utilizando una clave de datos única que AWS KMS genera (GenerateDataKey). El AWS Encryption SDK devuelve a Parameter Store un mensaje cifrado que incluye el valor del parámetro cifrado y una copia cifrada de la clave de datos única. Parameter Store almacena todo el mensaje cifrado en el valor del parámetro `SecureString`. A continuación, cuando se obtiene el valor de un parámetro `SecureString`, Parameter Store utiliza el AWS Encryption SDK para descifrar el valor del parámetro. Esto requiere una llamada a AWS KMS para descifrar la clave de datos cifrada.

Para crear un parámetro `SecureString` avanzado, utilice la operación PutParameter de la API de Systems Manager. Establezca el valor del parámetro `Tier` en `Advanced`. Incluya un parámetro `Type` con el valor `SecureString`. Para especificar una clave KMS, utilice el parámetro `KeyId`. El valor predeterminado es la Clave administrada de AWS de su cuenta, `aws/ssm`.

```
aws ssm put-parameter \
    --name MyParameter \
    --value "secret_value" \
    --type SecureString \
    --tier Advanced
```

El siguiente ejemplo similar utiliza el parámetro `--key-id` para especificar una clave administrada por el cliente. El ejemplo utiliza el Nombre de recurso de Amazon (ARN) de la clave KMS, pero puede utilizar cualquier identificador de clave KMS válido.

```
aws ssm put-parameter \
    --name MyParameter \
    --value "secret_value" \
    --type SecureString \
    --tier Advanced \
    --key-id arn:aws:kms:us-
east-2:987654321098:key/1234abcd-12ab-34cd-56ef-1234567890ab
```

Cuando se obtiene un parámetro `SecureString` de Parameter Store, su valor es el mensaje cifrado que devuelve el AWS Encryption SDK. Para obtener un parámetro, utilice la operación GetParameter de la API de Systems Manager.

En el siguiente ejemplo se utiliza la operación `GetParameter` de Systems Manager para obtener el parámetro `MyParameter` del Parameter Store sin descifrar su valor.

```
aws ssm get-parameter --name MyParameter
```

```
{
    "Parameter": {
        "Type": "SecureString",
        "Name": "MyParameter",
        "Value": "AQECAHgnOkMROh5LaLXkA4j0+vYi6tmM17Lg"
    }
}
```

Para descifrar el valor del parámetro antes de devolverlo, establezca el parámetro `WithDecryption` de `GetParameter` en `true`. Al utilizar `WithDecryption`, el Parameter Store llama a la operación Decrypt de AWS KMS en su nombre para descifrar el valor del parámetro. En consecuencia, la solicitud `GetParameter` devuelve el parámetro con un valor en texto no cifrado, tal y como se muestra en el siguiente ejemplo.

```
aws ssm get-parameter \
    --name MyParameter \
    --with-decryption
```

```
{
    "Parameter": {
        "Type": "SecureString",
        "Name": "MyParameter",
        "Value": "secret_value"
    }
}
```

No se puede convertir un parámetro `SecureString` avanzado en uno estándar, pero puede convertir un parámetro `SecureString` estándar en uno avanzado. Para convertir un parámetro `SecureString` estándar en un parámetro `SecureString` avanzado, utilice la operación `PutParameter` con el parámetro `Overwrite`. El valor de `Type` debe ser `SecureString` y el valor de `Tier` debe ser `Advanced`. El parámetro `KeyId`, que identifica una clave administrada por el cliente, es opcional. Si lo omite, el Parameter Store utiliza la Clave administrada de AWS de la cuenta. Puede especificar cualquier clave KMS para la que la entidad principal tenga permiso, aunque utilice una clave KMS diferente para cifrar el parámetro estándar.

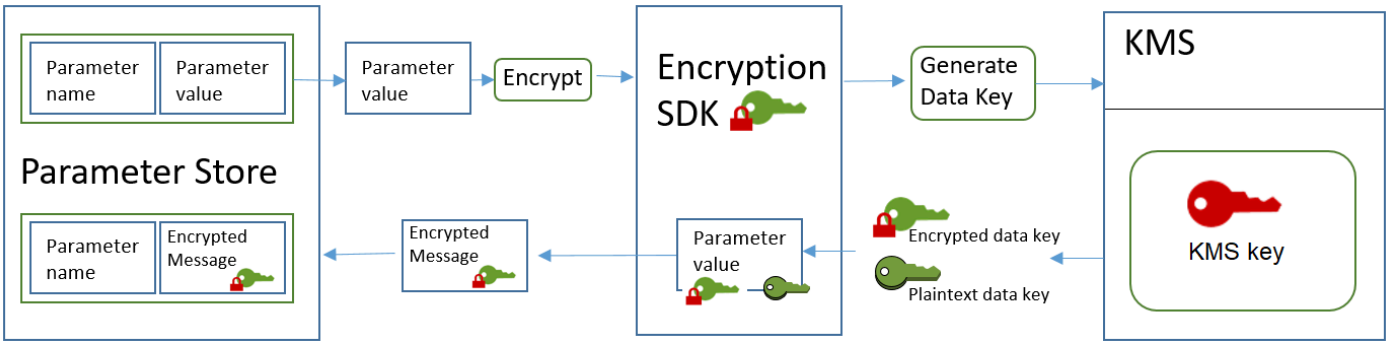
Cuando se utiliza el parámetro `Overwrite`, Parameter Store usa el AWS Encryption SDK para cifrar el valor del parámetro. A continuación, almacena el nuevo mensaje cifrado en Parameter Store.

```
aws ssm put-parameter \
    --name myStdParameter \
    --value "secret_value"  \
    --type SecureString \
    --tier Advanced \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --overwrite
```

El siguiente flujo de trabajo muestra cómo el Parameter Store utiliza una clave de KMS para cifrar y descifrar un parámetro `SecureString` avanzado.

Cifrar un parámetro avanzado

1. Cuando se utiliza `PutParameter` para crear un parámetro `SecureString` avanzado, el Parameter Store utiliza el AWS Encryption SDK y AWS KMS para cifrar el valor del parámetro. El Parameter Store llama al AWS Encryption SDK con el valor del parámetro, la clave de KMS que ha especificado y el contexto de cifrado del Parameter Store.
2. El AWS Encryption SDK envía una solicitud GenerateDataKey a AWS KMS con el identificador de la clave de KMS que ha especificado y el contexto de cifrado del Parameter Store. AWS KMS devuelve dos copias de la clave de datos única: una en texto no cifrado y otra cifrada con la clave de KMS. (El contexto de cifrado se utiliza al cifrar la clave de datos).
3. El AWS Encryption SDK usa la clave de datos en texto no cifrado para cifrar el valor del parámetro. Devuelve un mensaje cifrado que incluye el valor del parámetro cifrado, la clave de datos cifrada y otros datos, incluido el contexto de cifrado de Parameter Store.
4. Parameter Store almacena el mensaje cifrado como el valor del parámetro.



Descifrar un parámetro avanzado

1. Puede incluir el parámetro `WithDecryption` en una solicitud `GetParameter` para obtener un parámetro `SecureString` avanzado. Cuando lo haga, Parameter Store pasará el mensaje cifrado del valor del parámetro a un método de descifrado del AWS Encryption SDK.
2. El AWS Encryption SDK llama a la operación Decrypt (descifrado) de AWS KMS. Pasa la clave de datos cifrada y el contexto de cifrado de Parameter Store del mensaje cifrado.
3. AWS KMS utiliza la clave de KMS y el contexto de cifrado del Parameter Store para descifrar la clave de datos cifrada. A continuación, devuelve la clave de datos en texto no cifrado (descifrada) al AWS Encryption SDK.
4. El AWS Encryption SDK utiliza la clave de datos en texto no cifrado para descifrar el valor del parámetro. Devuelve el valor del parámetro en texto no cifrado a Parameter Store.
5. Parameter Store verifica el contexto de cifrado y devuelve el valor del parámetro en texto no cifrado en la respuesta de `GetParameter`.

Configurar permisos para cifrar y descifrar valores de parámetros

Para cifrar el valor de un parámetro `SecureString` estándar, el usuario necesita el permiso `kms:Encrypt`. Para cifrar el valor de un parámetro `SecureString` avanzado, el usuario necesita el permiso `kms:GenerateDataKey`. Para descifrar cualquier tipo de valor de un parámetro `SecureString`, el usuario necesita el permiso `kms:Decrypt`.

Puede utilizar las políticas de AWS Identity and Access Management (IAM) para conceder o denegar permisos a un usuario para llamar a las operaciones `PutParameter` y `GetParameter` de Systems Manager.

Además, si utiliza claves administradas por el cliente para cifrar valores de parámetros `SecureString`, puede utilizar políticas de IAM y de claves para administrar los permisos de cifrado y descifrado. Sin embargo, no puede establecer políticas de control de acceso para la clave KMS `aws/ssm` predeterminada. Para obtener más información acerca del control del acceso a las clave administradas por el cliente, consulte KMS key access and permissions en la Guía para desarrolladores de AWS Key Management Service.

En el siguiente ejemplo se muestra una política de IAM diseñada para parámetros `SecureString` estándar. Permite que el usuario llame a la operación `PutParameter` de Systems Manager en todos los parámetros de la ruta `FinancialParameters`. La política también permite al usuario llamar a la operación `Encrypt` de AWS KMS en una clave administrada por el cliente de ejemplo.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:PutParameter"
            ],
            "Resource": "arn:aws:ssm:us-east-2:111122223333:parameter/
FinancialParameters/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Encrypt"
            ],
            "Resource": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ]
}
```

En el siguiente ejemplo se muestra una política de IAM diseñada para parámetros `SecureString` avanzados. Permite que el usuario llame a la operación `PutParameter` de Systems Manager en todos los parámetros de la ruta `ReservedParameters`. La política también permite al usuario llamar a la operación `GenerateDataKey` de AWS KMS en una clave administrada por el cliente de ejemplo.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:PutParameter"
            ],
```

```
            "Resource": "arn:aws:ssm:us-east-2:111122223333:parameter/
ReservedParameters/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:GenerateDataKey"
            ],
            "Resource": "arn:aws:kms:us-east-2:111122223333:key/key-id"
        }
    ]
}
```

El último ejemplo también muestra una política de IAM que se puede utilizar para parámetros `SecureString` estándar o avanzados. Permite al usuario llamar a las operaciones `GetParameter` de Systems Manager (y a las operaciones relacionadas) en todos los parámetros de la ruta `ITParameters`. La política también permite al usuario llamar a la operación `Decrypt` de AWS KMS en una clave administrada por el cliente de ejemplo.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetParameter*"
            ],
            "Resource": "arn:aws:ssm:us-east-2:111122223333:parameter/
ITParameters/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ]
```

```
}
```

Contexto de cifrado de Parameter Store

Un contexto de cifrado es un conjunto de pares de clave-valor que contienen datos no secretos arbitrarios. Cuando se incluye un contexto de cifrado en una solicitud para cifrar datos, AWS KMS vincula criptográficamente el contexto de cifrado a los datos cifrados. Para descifrar los datos, es necesario pasar el mismo contexto de cifrado.

También se puede utilizar el contexto de cifrado para identificar una operación criptográfica en los registros y los registros de auditoría. El contexto de cifrado aparece en texto no cifrado en los registros, como los registros de AWS CloudTrail.

El AWS Encryption SDK también tiene un contexto de cifrado, aunque se administra de forma distinta. Parameter Store proporciona el contexto de cifrado al método de cifrado. El AWS Encryption SDK vincula criptográficamente el contexto de cifrado a los datos cifrados. También incluye el contexto de cifrado en texto no cifrado en el encabezado del mensaje cifrado que devuelve. Sin embargo, a diferencia de AWS KMS, los métodos de descifrado de AWS Encryption SDK no toman un contexto de cifrado como entrada. En lugar de ello, cuando se descifran los datos, el AWS Encryption SDK obtiene el contexto de cifrado del mensaje cifrado. Parameter Store verifica que el contexto de cifrado incluye el valor que espera antes de devolverle el valor del parámetro en texto no cifrado.

Parameter Store utiliza el siguiente contexto de cifrado en sus operaciones criptográficas:

- Clave: `PARAMETER_ARN`
- Valor: el nombre de recurso de Amazon (ARN) del parámetro que se va a cifrar.

El formato del contexto de cifrado es el siguiente:

```
"PARAMETER_ARN":"arn:aws:ssm:region-id:account-id:parameter/parameter-name"
```

Por ejemplo, el Parameter Store incluye este contexto de cifrado en las llamadas para cifrar y descifrar el parámetro `MyParameter` en una región y Cuenta de AWS de ejemplo.

```
"PARAMETER_ARN":"arn:aws:ssm:us-east-2:111122223333:parameter/MyParameter"
```

Si el parámetro está en una ruta jerárquica de Parameter Store, la ruta y el nombre se incluyen en el contexto de cifrado. Por ejemplo, este contexto de cifrado se utiliza al cifrar y descifrar el parámetro `MyParameter` en la ruta `/ReadableParameters` en una Cuenta de AWS y región de ejemplo.

```
"PARAMETER_ARN":"arn:aws:ssm:us-east-2:111122223333:parameter/ReadableParameters/
MyParameter"
```

Puede descifrar el valor de un parámetro `SecureString` cifrado llamando a la operación `Decrypt` de AWS KMS con el contexto de cifrado correcto y el valor del parámetro cifrado devuelto por la operación `GetParameter` de Systems Manager. No obstante, le animamos a que utilice la operación `GetParameter` con el parámetro `WithDecryption` para cifrar valores de parámetros de Parameter Store.

También puede incluir el contexto de cifrado en una política de IAM. Por ejemplo, puede permitir a un usuario que descifre únicamente el valor de un parámetro o un conjunto de valores de parámetros determinados.

La siguiente declaración de política de IAM de ejemplo permite al usuario obtener el valor del parámetro `MyParameter` y descifrar su valor mediante la clave KMS especificada. Sin embargo, los permisos solo se aplican cuando el contexto de cifrado coincide con la cadena especificada. Estos permisos no se aplican a ningún otro parámetro ni clave KMS, y la llamada a `GetParameter` da un error si el contexto de cifrado no coincide con la cadena.

Antes de utilizar una declaración de política como esta, sustituya los *`ARN de ejemplo`* por valores válidos.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetParameter*"
            ],
            "Resource": "arn:aws:ssm:us-
east-2:111122223333:parameter/MyParameter"
        },
```

```
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "arn:aws:kms:us-east-2:111122223333:key/key-id",
            "Condition": {
                "StringEquals": {
                    "kms:EncryptionContext:PARAMETER_ARN":"arn:aws:ssm:us-
east-2:111122223333:parameter/MyParameter"
                }
            }
        }
    ]
}
```

Solución de problemas de las claves de KMS en el Parameter Store

Para realizar operaciones en un parámetro `SecureString`, el Parameter Store debe poder utilizar la clave de AWS KMS especificada para la operación que se pretende realizar. La mayoría de los errores del Parameter Store relacionados con las claves de KMS los provocan los siguientes problemas:

- Las credenciales que utiliza una aplicación no tienen permisos para realizar la acción especificada en la clave de KMs.

  Para solucionar este error, ejecute la aplicación con otras credenciales o revise la IAM o la política de claves que impide realizar la operación. Para obtener ayuda con las políticas de claves y de IAM de AWS KMS, consulte KMS key access and permissions en la Guía para desarrolladores de AWS Key Management Service.

- No se encuentra la clave KMS.

  Normalmente, esto ocurre cuando se utiliza un identificador incorrecto para la clave KMS. Localice los identificadores correctos para la clave KMS e intente ejecutar de nuevo el comando.

- La clave KMS no está habilitada. Cuando esto ocurre, Parameter Store devuelve una excepción InvalidKeyId con un mensaje de error detallado de AWS KMS. Si el estado de la clave KMS es `Disabled`, habilítela. Si está `Pending Import`, complete el procedimiento de importación. Si el estado de la clave es `Pending Deletion`, cancele la eliminación de la clave o use una clave KMS distinta.

Para encontrar elestado de clave de una clave KMS, use la operación DescribeKey.

Uso de parámetros del Parameter Store en Amazon Elastic Kubernetes Service

Para mostrar parámetros de Parameter Store, una herramienta de AWS Systems Manager, como archivos montados en pods de Amazon EKS, puede usar el Proveedor de secretos y configuración (ASCP) de AWS para el controlador CSI de Kubernetes Secrets Store. El ASCP funciona con Amazon Elastic Kubernetes Service 1.17+ que ejecute un grupo de nodos de Amazon EC2. No se admiten grupos de nodos de AWS Fargate.

Con el ASCP, puede almacenar y administrar los parámetros en Parameter Store y recuperarlos a través de las cargas de trabajo que se ejecutan en Amazon EKS. Si el parámetro contiene varios pares clave-valor en formato JSON, puede elegir cuáles montar en Amazon EKS. El ASCP utiliza sintaxis JMESPath para consultar los pares clave-valor en el secreto. El ASCP también funciona con secretos de AWS Secrets Manager.

El ASCP ofrece dos métodos de autenticación con Amazon EKS. El primer enfoque utiliza los roles de IAM para cuentas de servicio (IRSA). El segundo enfoque utiliza Pod Identities. Cada enfoque tiene sus beneficios y sus casos de uso.

ASCP con roles de IAM para cuentas de servicio (IRSA)

El ASCP con roles de IAM para IRSA le permite montar parámetros de Parameter Store como archivos en los pods de Amazon EKS. Este enfoque es adecuado en los siguientes casos:

- cuando debe montar los parámetros como archivos en los pods,
- si está utilizando la versión 1.17 o una posterior de Amazon EKS con grupos de nodos de Amazon EC2,
- si desea recuperar pares clave-valor específicos de parámetros con formato JSON.

Para obtener más información, consulte the section called “Integración del ASCP con IRSA para Amazon EKS”.

ASCP con Pod Identity

ASCP con Pod Identity de EKS

El método del ASCP con Pod Identity mejora la seguridad y simplifica la configuración para acceder a los parámetros en Parameter Store. Este enfoque resulta beneficioso en los siguientes casos:

- cuando necesita una administración de permisos más detallada a nivel de pod,
- si está utilizando la versión 1.24 o una posterior de Amazon EKS,
- si desea mejorar el rendimiento y la escalabilidad.

Para obtener más información, consulte the section called “Cómo integrar el ASCP con Pod Identity para Amazon EKS”.

Elección del enfoque correcto

Tenga en cuenta los siguientes factores al decidir entre el ASCP con IRSA y el ASCP con Pod Identity:

- La versión de Amazon EKS: Pod Identity requiere Amazon EKS 1.24+, mientras que el controlador CSI funciona con Amazon EKS 1.17+.
- Requisitos de seguridad: Pod Identity ofrece un control más detallado a nivel de pod.
- Rendimiento: por lo general, Pod Identity funciona mejor en entornos de gran escala.
- Complejidad: Pod Identity simplifica la configuración al eliminar la necesidad de tener cuentas de servicio independientes.

Elija el método que mejor se adapte a sus requisitos específicos y al entorno de Amazon EKS.

Instalación del ASCP para Amazon EKS

En esta sección, se explica cómo instalar el Proveedor de secretos y configuración (ASCP) de AWS para Amazon EKS. Con el ASCP, puede montar parámetros de Parameter Store y secretos de AWS Secrets Manager como archivos en los pods de Amazon EKS.

Requisitos previos

- Un clúster de Amazon EKS
  - Versión 1.24 o posterior de Pod Identity
  - Versión 1.17 o posterior de IRSA
- La AWS CLI debe estar instalada y configurada

- Kubectl debe estar instalado y configurado para su clúster de Amazon EKS
- Helm (versión 3.0 o posterior)

Instalación y configuración del ASCP

El ASCP está disponible en GitHub en el repositorio secres-store-csi-proveedor-aws. El repositorio también contiene archivos YAML de ejemplo para crear y montar un secreto cambiando el valor `objectType` de `secretsmanager` a `ssmparameter`.

Durante la instalación, puede configurar el ASCP para que utilice un punto de conexión FIPS. Para obtener una lista de los puntos de conexión de Systems Manager, consulte Puntos de conexión de servicios de Systems Manager en la Referencia general de Amazon Web Services.

Instalar el ASCP mediante Helm

1. Para asegurarse de que el repositorio apunte al gráfico más reciente, utilice `helm repo update.`.
2. Agregue el gráfico de controladores CSI de Secrets Store.

```
helm repo add secrets-store-csi-driver https://kubernetes-sigs.github.io/secrets-store-csi-driver/charts
```

3. Instale el gráfico. Para configurar la limitación, agregue el siguiente indicador: `--set-json 'k8sThrottlingParams={"qps": "`*`number of queries per second`*`", "burst": "`*`number of queries per second`*`"}'`.

```
helm install -n kube-system csi-secrets-store secrets-store-csi-driver/secrets-store-csi-driver
```

4. Agregue el gráfico del ASCP.

```
helm repo add aws-secrets-manager https://aws.github.io/secrets-store-csi-driver-provider-aws
```

5. Instale el gráfico. Para utilizar un punto de conexión FIPS, agregue el siguiente indicador: `--set useFipsEndpoint=true`.

```
helm install -n kube-system secrets-provider-aws aws-secrets-manager/secrets-store-csi-driver-provider-aws
```

Instalarlo mediante el YAML del repositorio

- Use los siguientes comandos.

```
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
kubectl apply -f https://raw.githubusercontent.com/aws/secrets-store-csi-driver-provider-aws/main/deployment/aws-provider-installer.yaml
```

Verificación de las instalaciones

Para verificar las instalaciones del clúster de EKS, el controlador CSI de Secrets Store y el complemento ASCP, siga estos pasos:

1. Verifique el clúster de EKS:

   ```
   eksctl get cluster --name clusterName
   ```

   Este comando debería devolver información sobre el clúster.

2. Verifique la instalación del controlador CSI de Secrets Store:

   ```
   kubectl get pods -n kube-system -l app=secrets-store-csi-driver
   ```

   Debería ver los pods en ejecución con nombres como `csi-secrets-store-secrets-store-csi-driver-xxx`.

3. Verifique la instalación del complemento ASCP:

   YAML installation

   ```
   $ kubectl get pods -n kube-system -l app=csi-secrets-store-provider-aws
   ```

   Ejemplo de salida:

   ```
   NAME                                     READY   STATUS    RESTARTS   AGE
   csi-secrets-store-provider-aws-12345      1/1     Running   0          2m
   ```

Helm installation

```
$  kubectl get pods -n kube-system -l app=secrets-store-csi-driver-provider-aws
```

Ejemplo de salida:

```
NAME                                                     READY   STATUS    RESTARTS
 AGE
secrets-provider-aws-secrets-store-csi-driver-provider-67890       1/1
 Running   0          2m
```

Debería ver los pods con el estado `Running`.

Después de ejecutar estos comandos, si todo está configurado correctamente, debería ver que todos los componentes se están ejecutando sin errores. Si encuentra algún problema, es posible que deba consultar los registros de los pods que contienen dicho problema para solucionarlo.

Solución de problemas

1. Para verificar los registros del proveedor del ASCP, ejecute:

   ```
   kubectl logs -n kube-system -l app=csi-secrets-store-provider-aws
   ```

2. Verifique el estado de todos los pods en el espacio de nombres `kube-system`.

   Sustituya el *texto del marcador de posición predeterminado* por el ID del pod:

   ```
   kubectl -n kube-system get pods
   ```

   ```
   kubectl -n kube-system logs pod/pod-id
   ```

   Todos los pods relacionados con el controlador CSI y el ASCP deben tener el estado “En ejecución”.

3. Verifique la versión del controlador CSI:

   ```
   kubectl get csidriver secrets-store.csi.k8s.io -o yaml
   ```

Este comando debería devolver información sobre el controlador CSI instalado.

Recursos adicionales

Para obtener más información sobre el uso del ASCP con Amazon EKS, consulte los siguientes recursos:

- Using Pod Identity with Amazon EKS
- AWS Secrets Store CSI Driver on GitHub

Cómo utilizar CSI del Proveedor de secretos y configuración (ASCP) de AWS con Pod Identity para Amazon EKS

La integración del Proveedor de secretos y configuración (ASCP) de AWS con el agente de Pod Identity para Amazon Elastic Kubernetes Service proporciona una seguridad mejorada, una configuración simplificada y un rendimiento mejorado para las aplicaciones que se ejecutan en Amazon EKS. Pod Identity simplifica la autenticación de AWS Identity and Access Management (IAM) para Amazon EKS al recuperar parámetros de AWS Systems Manager Parameter Store o secretos de Secrets Manager.

Pod Identity de Amazon EKS agiliza el proceso de configuración de los permisos de IAM para las aplicaciones de Kubernetes, ya que permite que los permisos se configuren directamente a través de las interfaces de Amazon EKS, lo que reduce el número de pasos y elimina la necesidad de cambiar entre los servicios de Amazon EKS e IAM. Pod Identity permite usar un solo rol de IAM en varios clústeres sin actualizar las políticas de confianza y admite etiquetas de sesión de rol para un control de acceso más detallado. Este enfoque no solo simplifica la administración de políticas al permitir la reutilización de políticas de permisos en todos los roles, sino que también mejora la seguridad al permitir el acceso a los recursos de AWS en función de las etiquetas coincidentes.

Funcionamiento

1. Pod Identity asigna un rol de IAM al pod.
2. El ASCP usa este rol para autenticarse con Servicios de AWS.
3. Si está autorizado, el ASCP recupera los parámetros solicitados y hace que estén disponibles para el pod.

Para obtener más información, consulte Descripción del funcionamiento de Pod Identity de Amazon EKS en la Guía del usuario de Amazon EKS.

Requisitos previos

> **Important**
>
> Pod Identity solo es compatible con Amazon EKS en la nube. No es compatible con Amazon EKS Anywhere, Red Hat OpenShift Service en AWS ni con los clústeres de Kubernetes autoadministrados en las instancias de Amazon EC2.

- Clúster de Amazon EKS (versión 1.24 o posterior)
- Acceso a AWS CLI y al clúster de Amazon EKS a través de `kubectl`
- (Opcional) Acceso a dos Cuentas de AWS para el acceso entre cuentas

Cómo instalar el agente de Pod Identity de Amazon EKS

Para usar Pod Identity con el clúster, debe instalar el complemento del agente de Pod Identity de Amazon EKS.

Cómo instalar el agente de Pod Identity

- Instale el complemento del agente de Pod Identity en el clúster.

  Sustituya el *texto del marcador de posición predeterminado* por sus valores:

  ```
  eksctl create addon \
    --name eks-pod-identity-agent \
    --cluster clusterName \
    --region region
  ```

Cómo configurar el ASCP con Pod Identity

1. Cree una política de permisos que conceda los permisos `ssm:GetParameters` y `ssm:DescribeParameters` a los parámetros a los que el pod necesite acceder.
2. Cree un rol de IAM que pueda ser asumido por la entidad principal de servicio de Amazon EKS para Pod Identity:

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Principal": {
          "Service": "pods.eks.amazonaws.com"
        },
        "Action": [
          "sts:AssumeRole",
          "sts:TagSession"
        ],
        "Resource": "arn:aws:iam::123456789012:role/EKSPodIdentityRole"
      }
    ]
  }
```

Adjunte la política de IAM al rol.

Sustituya el *texto del marcador de posición predeterminado* por sus valores:

```
aws iam attach-role-policy \
  --role-name MY_ROLE \
  --policy-arn POLICY_ARN
```

3. Cree una asociación de Pod Identity. Para ver un ejemplo, consulte Creación de una asociación de Pod Identity en la Guía del usuario de Amazon EKS.
4. Cree `SecretProviderClass` que especifica qué parámetros o secretos se deben montar en el pod:

```
kubectl apply -f kubectl apply -f https://raw.githubusercontent.com/aws/
secrets-store-csi-driver-provider-aws/main/examples/ExampleSecretProviderClass-
PodIdentity.yaml
```

La diferencia clave en `SecretProviderClass` entre los roles de IAM para las cuentas de servicio (IRSA) y Pod Identity es el parámetro opcional `usePodIdentity`. Es un campo

opcional que determina el enfoque de autenticación. Si no se especifica, se utilizarán los roles de IAM para IRSA de manera predeterminada.

- Para usar Pod Identity de EKS, utilice cualquiera de estos valores: `"true", "True", "TRUE", "t", "T"`.
- Para usar IRSA de forma explícita, establézcalo en cualquiera de estos valores: `"false", "False", "FALSE", "f", or "F"`.

5. Implemente el pod que monta los parámetros o secretos en `/mnt/secrets-store`:

```
kubectl apply -f kubectl apply -f https://raw.githubusercontent.com/aws/secrets-store-csi-driver-provider-aws/main/examples/ExampleDeployment-PodIdentity.yaml
```

6. Si utiliza un clúster privado de Amazon EKS, asegúrese de que la VPC en la que se encuentre el clúster tenga un punto de conexión de AWS STS. Para obtener más información sobre la creación de un punto de conexión, consulte Puntos de conexión de VPC de tipo interfaz en la Guía del usuario de AWS Identity and Access Management.

Cómo verificar el montaje del secreto

Para verificar que el parámetro o el secreto se ha montado correctamente, ejecute el siguiente comando.

Sustituya el *texto del marcador de posición predeterminado* por sus valores:

```
kubectl exec -it $(kubectl get pods | awk '/pod-identity-deployment/{print $1}' | head -1) -- cat /mnt/secrets-store/MyParameter
```

Cómo configurar Pod Identity de Amazon EKS para acceder a los parámetros en Parameter Store

1. Cree una política de permisos que conceda los permisos `ssm:GetParameters` y `ssm:DescribeParameters` a los parámetros a los que el pod necesite acceder.
2. Cree un parámetro en Parameter Store, si aún no tiene uno. Para obtener más información, consulta Creación de parámetros de Parameter Store en Systems Manager.

Solución de problemas

Puede ver la mayoría de los errores si describe la implementación del pod.

Cómo ver los mensajes de error del contenedor

1. Obtenga una lista de nombres de pods con el siguiente comando. Si no está utilizando el espacio de nombres predeterminado, use `-n` *`namespace`*.

```
kubectl get pods
```

2. Para describir el pod, en el siguiente comando, en *`pod-id`*, use el ID de pod de los pods que encontró en el paso anterior. Si no está utilizando el espacio de nombres predeterminado, use `-n` *`NAMESPACE`*.

```
kubectl describe pod/pod-id
```

Cómo ver los errores del ASCP

- Para encontrar más información en los registros del proveedor, en el siguiente comando, en *`PODID`*, utilice el ID del pod csi-secrets-store-provider-aws.

```
kubectl -n kube-system get pods
kubectl -n kube-system logs pod/pod-id
```

Uso de CSI del Proveedor de secretos y configuración (ASCP) de AWS con roles de IAM para cuentas de servicio (IRSA)

Temas

- Requisitos previos
- Configurar el control de acceso
- Identificación de los parámetros para montar
- Solución de problemas

Requisitos previos

- Clúster de Amazon EKS (versión 1.17 o posterior)
- Acceso a AWS CLI y al clúster de Amazon EKS a través de `kubectl`

Configurar el control de acceso

El ASCP recupera Pod Identity de Amazon EKS y la cambia por un rol de IAM. Los permisos se establecen en una política de IAM para ese rol de IAM. Cuando el ASCP asume el rol de IAM, le da acceso a los parámetros que usted autorizó. Otros contenedores no pueden acceder a los parámetros a menos que también los asocie con el rol de IAM.

Concesión del acceso al pod de Amazon EKS a los parámetros en Parameter Store

1. Cree una política de permisos que conceda los permisos `ssm:GetParameters` y `ssm:DescribeParameters` a los parámetros a los que el pod necesite acceder.
2. Cree un proveedor OpenID Connect (OIDC) de IAM para el clúster si todavía no tiene uno. Para obtener más información, consulte Crear un proveedor OIDC de IAM para su clúster en la Guía del usuario de Amazon EKS.
3. Cree un rol de IAM para la cuenta de servicio y adjunte la política. Para obtener más información, consulte Crear un rol de IAM para su cuenta de servicio en la Guía del usuario de Amazon EKS.
4. Si utiliza un clúster privado de Amazon EKS, asegúrese de que la VPC en la que se encuentre el clúster tenga un punto de conexión de AWS STS. Para obtener más información sobre la creación de un punto de conexión, consulte Puntos de conexión de VPC de tipo interfaz en la Guía del usuario de AWS Identity and Access Management.

Identificación de los parámetros para montar

Para determinar qué parámetros debe montar el ASCP en Amazon EKS como archivos del sistema de archivos, se debe crear un archivo YAML the section called “SecretProviderClass”. El `SecretProviderClass` contiene una lista de los parámetros que hay que montar y el nombre de archivo con el que montarlos. El `SecretProviderClass` debe estar en el mismo espacio de nombres que el pod de Amazon EKS al que hace referencia.

Montaje de los parámetros como archivos

Las siguientes instrucciones muestran cómo montar los parámetros como archivos utilizando los archivos YAML ExampleSecretProviderClass.yaml y ExampleDeployment.yaml de ejemplo.

Montaje de los parámetros en Amazon EKS

1. Aplique el `SecretProviderClass` al pod:

```
kubectl apply -f ExampleSecretProviderClass.yaml
```

2. Implemente el pod:

```
kubectl apply -f ExampleDeployment.yaml
```

3. El ASCP monta los archivos.

Solución de problemas

Puede ver la mayoría de los errores si describe la implementación del pod.

Ver los mensajes de error del contenedor

1. Obtenga una lista de nombres de pods con el siguiente comando. Si no está utilizando el espacio de nombres predeterminado, use `-n` *`name-space`*.

```
kubectl get pods
```

2. Para describir el pod, en el siguiente comando, en *`pod-id`*, use el ID de pod de los pods que encontró en el paso anterior. Si no está utilizando el espacio de nombres predeterminado, use `-n` *`nameSpace`*.

```
kubectl describe pod/pod-id
```

Ver los errores del ASCP

- Para obtener más información en los registros del proveedor, en el siguiente comando, en *`pod-id`*, utilice el ID del pod csi-secrets-store-provider-aws.

```
kubectl -n kube-system get pods
kubectl -n kube-system logs Pod/pod-id
```

- Comprobación de que la definición de recurso personalizado (CRD) de **`SecretProviderClass`** está instalada:

```
kubectl get crd secretproviderclasses.secrets-store.csi.x-k8s.io
```

Este comando debe devolver información acerca de la definición de recurso personalizado de `SecretProviderClass`.

- Comprobación de que se haya creado el objeto SecretProviderClass.

```
kubectl get secretproviderclass SecretProviderClassName -o yaml
```

Ejemplos de código del Proveedor de secretos y configuración de AWS

Ejemplos de autenticación y control de acceso del ASCP

Ejemplo: política de IAM que permite que el servicio Pod Identity de Amazon EKS (pods.eks.amazonaws.com) asuma el rol y etiquete la sesión:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "pods.eks.amazonaws.com"
      },
      "Action": [
        "sts:AssumeRole",
        "sts:TagSession"
      ],
      "Resource": "arn:aws:iam::111122223333:role/EKSPodIdentityRole"
    }
  ]
}
```

SecretProviderClass

Se debe utilizar YAML para describir qué parámetros hay que montar en Amazon EKS mediante el ASCP. Para ver ejemplos, consulta the section called “Uso de SecretProviderClass”.

Estructura YAML de SecretProviderClass

```
apiVersion: secrets-store.csi.x-k8s.io/v1
kind: SecretProviderClass
metadata:
   name: name
spec:
  provider: aws
  parameters:
    region:
    failoverRegion:
    pathTranslation:
    usePodIdentity:
    preferredAddressType:
    objects:
```

Los campos de los parámetros contienen los detalles de la solicitud de montaje:

region

(Opcional) La Región de AWS del parámetro. Si no utiliza este campo, el ASCP busca la región en la anotación en el nodo. Esta búsqueda agrega una sobrecarga a las solicitudes de montaje, por lo que recomendamos que proporcione la región para los clústeres que utilizan una gran cantidad de pods.

Si también se especifica `failoverRegion`, el ASCP intenta recuperar el parámetro desde ambas regiones. Si alguna de estas regiones devuelve un error `4xx` (por ejemplo, por un problema de autenticación), el ASCP no monta ninguno de los parámetros. Si el parámetro se recupera correctamente desde `region`, el ASCP monta ese valor de parámetro. Si el parámetro no se recupera correctamente desde `region`, pero sí desde `failoverRegion`, el ASCP monta ese valor de parámetro.

`failoverRegion`

(Opcional) Si se incluye este campo, el ASCP intenta recuperar el parámetro desde las regiones definidas en `region` y este campo. Si alguna de estas regiones devuelve un error `4xx` (por ejemplo, por un problema de autenticación), el ASCP no monta ninguno de los parámetros. Si el parámetro se recupera correctamente desde `region`, el ASCP monta ese valor de parámetro. Si el parámetro no se recupera correctamente desde `region`, pero sí desde `failoverRegion`, el ASCP monta ese valor de parámetro. Para ver un ejemplo sobre cómo utilizar este campo, consulte Conmutación por error de parámetros multirregionales.

pathTranslation

(Opcional) Un único carácter de sustitución para utilizarlo si el nombre del archivo de Amazon EKS contiene el carácter separador de ruta, por ejemplo la barra diagonal (/) en Linux. El ASCP no puede crear un archivo montado que contenga un carácter separador de ruta. En su lugar, el ASCP reemplaza el carácter separador de ruta por otro carácter. Si no se utiliza este campo, el carácter de reemplazo es el guion bajo (_), de modo que, por ejemplo, `My/Path/Parameter` se monta como `My_Path_Parameter`.

Para evitar la sustitución de caracteres, ingrese la cadena `False`.

usePodIdentity

(Opcional) Determine el enfoque de autenticación. Si no se especifica, se utilizarán los roles de IAM para las cuentas de servicio (IRSA) de manera predeterminada.

- Para usar Pod Identity de EKS, utilice cualquiera de estos valores: `"true"`, `"True"`, `"TRUE"`, `"t"` o `"T"`.
- Para usar IRSA de forma explícita, establezca cualquiera de estos valores: `"false"`, `"False"`, `"FALSE"`, `"f"` o `"F"`.

preferredAddressType

(Opcional) Especifica el tipo de dirección IP preferido para la comunicación del punto de conexión del agente de Pod Identity. El campo solo se aplica cuando se utiliza la característica Pod Identity de EKS y se ignora cuando se utilizan roles de IAM para cuentas de servicio. Los valores no distinguen entre mayúsculas y minúsculas. Los valores válidos son:

- `"ipv4"`, `"IPv4"` o `"IPV4"`: fuerza el uso del punto de conexión IPv4 del agente de Pod Identity.
- `"ipv6"`, `"IPv6"` o `"IPV6"`: fuerza el uso del punto de conexión IPv6 del agente de Pod Identity.
- no especificado: usa la selección automática del punto de conexión; prueba primero el punto de conexión IPv4 y recurre al punto de conexión IPv6 si IPv4 falla.

objetos

Una cadena que contiene una declaración YAML de los secretos que se van a montar. Se recomienda utilizar una cadena de varias líneas de YAML o una barra vertical (|).

objectName

Obligatorio. Especifica el nombre del parámetro o el secreto que se va a obtener. En el caso de Parameter Store, este es el `Name` del parámetro y puede ser el nombre o el ARN completo del parámetro. En el caso de Secrets Manager, este es el parámetro `SecretId` y puede ser el nombre descriptivo o el ARN completo del secreto.

objectType

Es requerido si no utiliza un ARN de Secrets Manager para `objectName`. En Parameter Store, utilice `ssmparameter`. Para Secrets Manager, utilice `secretsmanager`.

objectAlias

(Opcional) El nombre de archivo del secreto en el pod de Amazon EKS. Si no especifica este campo, el `objectName` aparece como nombre de archivo.

objectVersion

(Opcional) El ID de la versión del parámetro. No se recomienda porque se debe actualizar el ID de la versión cada vez que se actualice el parámetro. Se utiliza la versión más reciente de forma predeterminada. Si se incluye `failoverRegion`, este campo representa el campo `objectVersion` principal.

objectVersionLabel

(Opcional) El alias de la versión. La versión predeterminada es la versión `AWSCURRENT` más reciente. Si se incluye `failoverRegion`, este campo representa el campo `objectVersionLabel` principal.

jmesPath

(Opcional) Un mapa de las claves en el parámetro para los archivos que se van a montar en Amazon EKS. Para utilizar este campo, el valor del parámetro debe estar en formato JSON.

En el siguiente ejemplo se muestra el aspecto de un parámetro codificado JSON.

```
{
    "username" : "myusername",
    "password" : "mypassword"
}
```

Las claves son `username` y `password`. El valor asociado a `username` es `myusername`, y el valor asociado a `password` es `mypassword`.

Si utiliza este campo, debe incluir los subcampos `path` y `objectAlias`.

path

Una clave de un par clave-valor en el JSON del valor del parámetro. Si el campo contiene un guion, aplique escape con comillas simples, por ejemplo: `path: '"hyphenated-path"'`

objectAlias

Nombre de archivo que se va a montar en el pod de Amazon EKS. Si el campo contiene un guion, aplique escape con comillas simples, por ejemplo: `objectAlias: '"hyphenated-alias"'`

failoverObject

(Opcional) Si se especifica este campo, el ASCP intenta recuperar tanto el parámetro especificado en el campo `objectName` principal como el parámetro especificado en el subcampo `failoverObject` `objectName`. Si alguno devuelve un error `4xx` (por ejemplo, por un problema de autenticación), el ASCP no monta ninguno de los parámetros. Si el parámetro se recupera correctamente desde el campo `objectName` principal, el ASCP monta ese valor de parámetro. Si el parámetro no se recupera correctamente desde el campo `objectName` principal, pero sí desde el `objectName` de conmutación por error, el ASCP monta ese valor de parámetro. Si se incluye este campo, se debe incluir el campo `objectAlias`. Para ver un ejemplo sobre cómo utilizar este campo, consulte Conmutación por error a un parámetro diferente.

Este campo se suele utilizar cuando el parámetro de conmutación por error no es una réplica. Para ver un ejemplo sobre cómo especificar una réplica, consulte Conmutación por error de parámetros multirregionales.

objectName

Nombre o ARN completo del parámetro de conmutación por error. Si se utiliza un ARN, la región del ARN debe coincidir con el campo `failoverRegion`.

objectVersion

(Opcional) El ID de la versión del parámetro. Debe coincidir con el campo `objectVersion` principal. No se recomienda porque se debe actualizar el ID de la versión cada vez que se actualice el parámetro. Se utiliza la versión más reciente de forma predeterminada.

objectVersionLabel

(Opcional) El alias de la versión. La versión predeterminada es la versión `AWSCURRENT` más reciente.

Cree una configuración básica de SecretProviderClass para montar los parámetros en los pods de Amazon EKS.

Pod Identity

SecretProviderClass para usar un parámetro en el mismo clúster de Amazon EKS:

```
apiVersion: secrets-store.csi.x-k8s.io/v1
kind: SecretProviderClass
metadata:
  name: aws-parameter-store
spec:
  provider: aws
  parameters:
    objects: |
      - objectName: "MyParameter"
        objectType: "ssmparameter"
    usePodIdentity: "true"
```

IRSA

```
apiVersion: secrets-store.csi.x-k8s.io/v1
kind: SecretProviderClass
metadata:
  name: deployment-aws-parameter
spec:
  provider: aws
  parameters:
    objects: |
        - objectName: "MyParameter"
          objectType: "ssmparameter"
```

Uso de SecretProviderClass

Utilice estos ejemplos para crear configuraciones de `SecretProviderClass` para diferentes situaciones.

Ejemplo: montaje de parámetros por nombre o ARN

En este ejemplo, se muestra cómo montar tres tipos diferentes de parámetros:

- un parámetro especificado por ARN completo,
- un parámetro especificado por su nombre,
- una versión de parámetro de un secreto.

```
apiVersion: secrets-store.csi.x-k8s.io/v1
kind: SecretProviderClass
metadata:
  name: aws-parameters
spec:
  provider: aws
  parameters:
    objects: |
      - objectName: "arn:aws:ssm:us-east-2:777788889999:parameter:MyParameter2-d4e5f6"
      - objectName: "MyParameter3"
        objectType: "ssmparameter"
      - objectName: "MyParameter4"
        objectType: "ssmparameter"
        objectVersionLabel: "AWSCURRENT"
```

Ejemplo: montaje de pares clave-valor de un parámetro

En este ejemplo, se muestra cómo montar pares clave-valor específicos a partir de un parámetro con formato JSON:

```
apiVersion: secrets-store.csi.x-k8s.io/v1
kind: SecretProviderClass
metadata:
  name: aws-parameters
spec:
  provider: aws
  parameters:
    objects: |
      - objectName: "arn:aws:ssm:us-east-2:777788889999:parameter:MyParameter-a1b2c3"
        jmesPath:
            - path: username
              objectAlias: dbusername
            - path: password
```

```
            objectAlias: dbpassword
```

Ejemplo: configuración de conmutación por error

En estos ejemplos, se muestra cómo configurar la conmutación por error para parámetros.

Conmutación por error de parámetros multirregionales

En este ejemplo, se muestra cómo configurar la conmutación por error automática para un parámetro replicado en varias regiones:

```
apiVersion: secrets-store.csi.x-k8s.io/v1
kind: SecretProviderClass
metadata:
  name: aws-parameters
spec:
  provider: aws
  parameters:
    region: us-east-1
    failoverRegion: us-east-2
    objects: |
      - objectName: "MyParameter"
```

Conmutación por error a un parámetro diferente

En este ejemplo, se muestra cómo configurar la conmutación por error a un parámetro diferente (no a una réplica):

```
apiVersion: secrets-store.csi.x-k8s.io/v1
kind: SecretProviderClass
metadata:
  name: aws-parameters
spec:
  provider: aws
  parameters:
    region: us-east-1
    failoverRegion: us-east-2
    objects: |
      - objectName: "arn:aws:ssm:us-east-1:777788889999:parameter:MyParameter-a1b2c3"
        objectAlias: "MyMountedParameter"
        failoverObject:
          - objectName: "arn:aws:ssm:us-
east-2:777788889999:parameter:MyFailoverParameter-d4e5f6"
```

Recursos adicionales

Para obtener más información sobre el uso del ASCP con Amazon EKS, consulte los siguientes recursos:

- Using Pod Identity with Amazon EKS
- AWS Secrets Store CSI Driver on GitHub

Cómo utilizar parámetros Parameter Store en funciones AWS Lambda

Parameter Store, una herramienta de AWS Systems Manager, proporciona un almacenamiento seguro y jerárquico para la administración de los datos de configuración y de los secretos. Puede almacenar datos como contraseñas, cadenas de base de datos, ID de Amazon Machine Image (AMI) y códigos de licencia como valores de parámetros.

Para usar parámetros de Parameter Store en funciones AWS Lambda sin usar un SDK, puede usar la extensión de Lambda para secretos y parámetros de AWS. Esta extensión recupera los valores de los parámetros y los almacena en la caché para usarlos en el futuro. El uso de la extensión Lambda puede reducir sus costos al reducir la cantidad de llamadas a la API a Parameter Store. El uso de la extensión también puede mejorar la latencia, ya que recuperar un parámetro almacenado en caché es más rápido que recuperarlo de Parameter Store.

Una extensión de Lambda es un proceso complementario que se suma a las capacidades de una función Lambda. Una extensión es como un cliente que se ejecuta en paralelo a una invocación de Lambda. Este cliente paralelo puede interactuar con su función en cualquier momento de su ciclo de vida. Para obtener más información sobre las extensiones de Lambda, consulte Extensiones de Lambda en la Guía para desarrolladores de AWS Lambda.

La extensión de Lambda para secretos y parámetros de AWS funciona tanto para Parameter Store como para AWS Secrets Manager. Para aprender a usar la extensión de Lambda con los secretos de Secrets Manager, consulte Utilizar secretos de AWS Secrets Manager en funciones AWS Lambda en la Guía del usuario de AWS Secrets Manager.

Información relacionada

Uso de la extensión de Lambda Parameter and Secrets de AWS para almacenar parámetros y secretos en caché (Compute Blog AWS)

Cómo funciona la extensión

Para utilizar parámetros en una función Lambda sin la extensión de Lambda, debe configurar su función Lambda para recibir actualizaciones de configuración mediante la integración con la acción de la API `GetParameter` para Parameter Store.

Cuando utiliza la extensión de Lambda para secretos y parámetros de AWS, la extensión recupera el valor del parámetro Parameter Store y lo almacena en la memoria caché local. A continuación, el valor almacenado en la caché se utiliza para otras invocaciones hasta que caduque. Los valores en la memoria caché caducan una vez transcurrido su tiempo de vida (TTL). Puede configurar el valor TTL mediante la variable de entorno `SSM_PARAMETER_STORE_TTL`, como se explica más adelante en este tema.

Si el TTL de la caché configurado no ha caducado, se utiliza el valor del parámetro almacenado en la caché. Si el periodo ha caducado, el valor almacenado en caché se invalida y se recupera el valor del parámetro Parameter Store.

Además, el sistema detecta los valores de los parámetros que se utilizan con frecuencia y los mantiene en la memoria caché mientras borra los que están caducados o no se utilizan.

> **Important**
>
> La extensión solo se puede invocar en la fase `INVOKE` de la operación de Lambda y no durante la fase `INIT`.

Detalles de la implementación

Utilice los siguientes detalles para ayudarle a configurar la extensión de Lambda para secretos y parámetros de AWS.

Autenticación

Para autorizar y autenticar las solicitudes de Parameter Store, la extensión usa las mismas credenciales que las que se utilizan para ejecutar la propia función Lambda. Por lo tanto, el rol de (IAM) AWS Identity and Access Management utilizado para ejecutar la función debe tener los siguientes permisos para interactuar con Parameter Store:

- `ssm:GetParameter`: necesario para recuperar parámetros de Parameter Store
- `kms:Decrypt`: necesario si está recuperando parámetros de `SecureString` desde Parameter Store

Para obtener más información, consulte Rol de ejecución de AWS Lambda en la Guía para desarrolladores de AWS Lambda.

Instanciación

Lambda crea instancias independientes correspondientes al nivel de simultaneidad que requiere la función. Cada instancia está aislada y mantiene su propia memoria caché local de los datos de configuración. Para obtener más información sobre las instancias de Lambda y la simultaneidad, consulte Configuración de la simultaneidad reservada en la Guía para desarrolladores de AWS Lambda.

Sin dependencia del SDK

La extensión de Lambda para secretos y parámetros de AWS funciona independientemente de cualquier biblioteca de lenguajes del SDK de AWS. No se requiere un SDK de AWS para realizar solicitudes GET a Parameter Store.

Puerto del Localhost

Utilice `localhost` en sus solicitudes GET. La extensión hace solicitudes al puerto 2773 de localhost. No tiene que especificar un punto de conexión externo o interno para usar la extensión. Para configurar el puerto, establezca la variable de entorno `PARAMETERS_SECRETS_EXTENSION_HTTP_PORT`.

Por ejemplo, en Python, GET URL puede tener un aspecto similar al del siguiente ejemplo.

```
parameter_url = ('http://localhost:' + port + '/systemsmanager/parameters/get/?
name=' + ssm_parameter_path)
```

Cambios en el valor de un parámetro antes de que caduque el TTL

La extensión no detecta cambios en el valor del parámetro y no realiza una actualización automática antes de que caduque el TTL. Si cambia el valor de un parámetro, las operaciones que utilizan el valor del parámetro almacenado en la caché pueden fallar hasta que se actualice la memoria caché de nuevo. Si espera cambios frecuentes en el valor de un parámetro, le recomendamos establecer un valor de TTL más corto.

Requisito de encabezado

Para recuperar los parámetros de la caché de la extensión, el encabezado de la solicitud GET debe incluir una referencia de `X-Aws-Parameters-Secrets-Token`. Configure el token en

`AWS_SESSION_TOKEN`, que Lambda proporciona para todas las funciones en ejecución. El uso de este encabezado indica que el intermediario se encuentra en el entorno de Lambda.

Ejemplo

El siguiente ejemplo en Python muestra una solicitud básica para recuperar el valor de un parámetro almacenado en caché.

```
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)
```

Compatibilidad con ARM

La extensión es compatible con la arquitectura ARM en la mayoría de las Regiones de AWS en las que se admiten las arquitecturas x86_64 y x86. Si utiliza la arquitectura ARM, le sugerimos que compruebe que su arquitectura sea compatible. Para obtener listas completas de los ARN de extensión, consulte ARN de la extensión AWS Parameters and Secrets Lambda.

Registro

Lambda registra la información de ejecución acerca de la extensión junto con la función mediante Registros de Amazon CloudWatch. De forma predeterminada, la extensión registra una cantidad mínima de información en CloudWatch. Para registrar más detalles, establezca la variable de entorno `PARAMETERS_SECRETS_EXTENSION_LOG_LEVEL` en `DEBUG`.

Cómo agregar la extensión a una función Lambda

Para usar la extensión de Lambda para secretos y parámetros de AWS, agregue la extensión a la función Lambda como una capa.

Utilice uno de los métodos siguientes para agregar la extensión a la función.

AWS Management Console (Opción Agregar capa)

1. Abra la consola de AWS Lambda en https://console.aws.amazon.com/lambda/.
2. Elija su función. En el área Layers (Capas), elija Add a layer (Agregar una capa).
3. En el área Choose a layer (Elegir una capa), elija la opción AWS layers (Capas de).
4. En AWS layers (Capas de), elija AWS-Parameters-and-Secrets-Lambda-Extension, elija una versión y, a continuación, elija Add (Agregar).

AWS Management Console(Opción Especificar ARN)

1. Abra la consola de AWS Lambda en https://console.aws.amazon.com/lambda/.
2. Elija su función. En el área Layers (Capas), elija Add a layer (Agregar una capa).
3. En el área Choose a layer (Elegir una capa), elija la opción Specify an ARN (Especificar un ARN).
4. En Specify an ARN (Especificar un ARN), ingrese la extensión de ARN para su Región de AWS y arquitectura y, a continuación, elija Add (Agregar).

AWS Command Line Interface

Ejecute el siguiente comando en la AWS CLI: Reemplace cada *example resource placeholder* con su propia información.

```
aws lambda update-function-configuration \
    --function-name function-name \
    --layers layer-ARN
```

Información relacionada

Uso de capas con su función de Lambda

Configuración de extensiones (archivo de archivo .zip)

Variables de entorno de la extensión AWS Parameters and Secrets Lambda

Puede configurar la extensión si cambia las siguientes variables de entorno. Para ver la configuración actual, establezca `PARAMETERS_SECRETS_EXTENSION_LOG_LEVEL` en `DEBUG`. Para obtener más información, consulte Uso de variables de entorno de AWS Lambda en la Guía para desarrolladores de AWS Lambda.

> **Note**
>
> AWS Lambda registra los detalles de la operación sobre la extensión Lambda y la función Lambda en Registros de Amazon CloudWatch.

Variable de entorno	Detalles	Obligatorio	Valores válidos	Valor predeterminado
`SSM_PARAMETER_STORE_TIMEOUT_MILLIS`	Tiempo de espera, en milisegundos, para las solicitudes a Parameter Store. Un valor de 0 (cero) indica que no hay tiempo de inactividad.	No	Todos los números enteros	0 (cero)
`SECRETS_MANAGER_TIMEOUT_MILLIS`	Tiempo de espera, en milisegundos, para las solicitudes a Secrets Manager. Un valor de 0 (cero) indica que no hay tiempo de inactividad.	No	Todos los números enteros	0 (cero)
`SSM_PARAMETER_STORE_TTL`	Duración máxima válida, en segundos,	No	0 (cero) a 300 s (cinco minutos)	300 s (cinco minutos)

Variable de entorno	Detalles	Obligatorio	Valores válidos	Valor predeterminado
	de un parámetro de la caché antes de que se invalide. Un valor de 0 (cero) indica que se debe omitir la memoria caché. Esta variable se ignora si el valor de `PARAMETERS_SECRETS_EXTENSION_CACHE_SIZE` es 0 (cero).			

Variable de entorno	Detalles	Obligatorio	Valores válidos	Valor predeterminado
SECRETS_MANAGER_TTL	Duración máxima válida, en segundos, de un secreto de la caché antes de que se invalide. Un valor de 0 (cero) indica que se omite la memoria caché. Esta variable se ignora si el valor de PARAMETERS_SECRETS_EXTENSION_CACHE_SIZE es 0 (cero).	No	0 (cero) a 300 s (cinco minutos)	300 s (5 minutos)
PARAMETERS_SECRETS_EXTENSION_CACHE_ENABLED	Determina si la caché está habilitada para la extensión. Valores válidos: TRUE | FALSE	No	TRUE, FALSE	TRUE

Variable de entorno	Detalles	Obligatorio	Valores válidos	Valor predeterminado
`PARAMETERS_SECRETS_EXTENSION_CACHE_SIZE`	El tamaño máximo de la memoria caché en términos de número de elementos. Un valor de 0 (cero) indica que se omite la memoria caché. Esta variable se ignora si ambos valores de TTL de la caché son 0 (cero).	No	0 (cero) a 1000	1 000
`PARAMETERS_SECRETS_EXTENSION_HTTP_PORT`	El puerto del servidor HTTP local.	No	1 a 65535	2773

Variable de entorno	Detalles	Obligatorio	Valores válidos	Valor predeterminado
`PARAMETERS_SECRETS_EXTENSION_MAX_CONNECTIONS`	Cantidad máxima de conexiones para los clientes HTTP que la extensión utiliza para hacer solicitudes a Parameter Store o a Secrets Manager. Esta es una configuración por cliente para la cantidad de conexiones que tanto el cliente de Secrets Manager como el cliente de Parameter Store realizan a los servicios de backend.	No	Mínimo de 1; sin límite máximo.	3

Variable de entorno	Detalles	Obligatorio	Valores válidos	Valor predeterminado
`PARAMETERS_SECRETS_EXTENSION_LOG_LEVEL`	El nivel de detalle indicado en los registros de la extensión. Le recomendamos que utilice `DEBUG` para obtener más detalles sobre la configuración de la memoria caché a medida que configura y prueba la extensión. Los registros de las operaciones de Lambda se envían automáticamente a un grupo de registro de Registros de CloudWatch asociado.	No	`DEBUG | WARN | ERROR | NONE | INFO`	`INFO`

Ejemplos de comandos para usar AWS Systems Manager Parameter Store y la extensión AWS Secrets Manager

Los ejemplos de esta sección muestran las acciones de la API para su uso con AWS Systems Manager Parameter Store y la extensión AWS Secrets Manager.

Comandos de ejemplo para Parameter Store

La extensión Lambda utiliza el acceso de solo lectura a la acción de la API GetParameter.

Para realizar esta acción, realice una llamada HTTP GET similar a la siguiente. Este formato de comando proporciona acceso a los parámetros del nivel de parámetros estándar.

```
GET http://localhost:port/systemsmanager/parameters/get?name=parameter-
name&version=version&label=label&withDecryption={true|false}
```

En este ejemplo, *parameter-name* representa el nombre completo del parámetro, por ejemplo `MyParameter`, para un parámetro que no está en una jerarquía o `%2FDev%2FProduction%2FEast%2FProject-ABC%2FMyParameter` para un parámetro denominado `/Dev/Production/East/Project-ABC/MyParameter` que forma parte de una jerarquía.

> **Note**
>
> Cuando se utilizan llamadas GET, los valores de los parámetros deben codificarse para que HTTP conserve los caracteres especiales. Por ejemplo, en lugar de formatear una ruta jerárquica como `/a/b/c`, codifique los caracteres que puedan interpretarse como parte de la URL, como `%2Fa%2Fb%2Fc`.

versión y *etiqueta* son los selectores disponibles para su uso con la acción `GetParameter`.

```
GET http://localhost:port/systemsmanager/parameters/get/?name=MyParameter&version=5
```

Para llamar a un parámetro de una jerarquía, realice una llamada HTTP GET similar a la siguiente.

```
GET http://localhost:port/systemsmanager/parameters/get?name=%2Fa%2Fb%2F&label=release
```

Para llamar a un parámetro público (global), realice una llamada HTTP GET similar a la siguiente.

```
GET http://localhost:port/systemsmanager/parameters/get/?name=%2Faws%2Fservice%20list
%2F…
```

Para realizar una llamada HTTP GET a un secreto de Secrets Manager mediante referencias de Parameter Store, realice una llamada HTTP GET similar a la siguiente.

```
GET http://localhost:port/systemsmanager/parameters/get?name=%2Faws%2Freference
%2Fsecretsmanager%2F…
```

Para realizar una llamada con el nombre de recurso de Amazon (ARN) para un parámetro, realice una llamada HTTP GET similar a la siguiente.

```
GET http://localhost:port/systemsmanager/parameters/get?name=arn:aws:ssm:us-
east-1:123456789012:parameter/MyParameter
```

Para realizar una llamada que acceda a un parámetro `SecureString` con descifrado, realice una llamada HTTP GET similar a la siguiente.

```
GET http://localhost:port/systemsmanager/parameters/get?
name=MyParameter&withDecryption=true
```

Puede especificar que los parámetros no se descifren si omite `withDecryption` o con una configuración `false` explícita. También puede especificar una versión o una etiqueta, pero no ambas. Si lo hace, solo se utilizará lo primero que se coloque después del signo de interrogación (?) en la URL.

ARN de la extensión AWS Parameters and Secrets Lambda

En las siguientes tablas se proporcionan los ARN de extensión para las arquitecturas y regiones compatibles.

Temas

- ARN de extensión para arquitecturas x86_64 y x86
- ARN de extensión para las arquitecturas ARM64 y Mac with Apple silicon

ARN de extensión para arquitecturas x86_64 y x86

Última actualización: 26 de junio de 2025

Región	ARN
Este de EE. UU. (Ohio)	`arn:aws:lambda:us-east-2:590474943231:layer:AWS-Parame`

Región	ARN
	ters-and-Secrets-Lambda-Ext ension:22
Este de EE. UU. (Norte de Virginia)	arn:aws:lambda:us-east-1:17 7933569100:layer:AWS-Parame ters-and-Secrets-Lambda-Ext ension:18
Oeste de EE. UU. (Norte de California)	arn:aws:lambda:us-west-1:99 7803712105:layer:AWS-Parame ters-and-Secrets-Lambda-Ext ension:18
Oeste de EE. UU. (Oregón)	arn:aws:lambda:us-west-2:34 5057560386:layer:AWS-Parame ters-and-Secrets-Lambda-Ext ension:18
África (Ciudad del Cabo)	arn:aws:lambda:af-south-1:3 17013901791:layer:AWS-Param eters-and-Secrets-Lambda-Ex tension:18
Asia Pacific (Hong Kong)	arn:aws:lambda:ap-east-1:76 8336418462:layer:AWS-Parame ters-and-Secrets-Lambda-Ext ension:20
Asia-Pacífico (Taipéi)	arn:aws:lambda:ap-east-2:89 0742577149:layer:AWS-Parame ters-and-Secrets-Lambda-Ext ension:6

Región	ARN
Región de Asia Pacífico (Hyderabad)	arn:aws:lambda:ap-south-2:070087711984:layer:AWS-Parameters-and-Secrets-Lambda-Extension:15
Asia-Pacífico (Yakarta)	arn:aws:lambda:ap-southeast-3:490737872127:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18
Asia-Pacífico (Melbourne)	arn:aws:lambda:ap-southeast-4:090732460067:layer:AWS-Parameters-and-Secrets-Lambda-Extension:8
Asia-Pacífico (Malasia)	arn:aws:lambda:ap-southeast-5:381492012281:layer:AWS-Parameters-and-Secrets-Lambda-Extension:7
Asia-Pacífico (Bombay)	arn:aws:lambda:ap-south-1:176022468876:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18
Asia-Pacífico (Osaka)	arn:aws:lambda:ap-northeast-3:576959938190:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18
Asia-Pacífico (Seúl)	arn:aws:lambda:ap-northeast-2:738900069198:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18

Región	ARN
Asia-Pacífico (Singapur)	`arn:aws:lambda:ap-southeast-1:044395824272:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18`
Asia-Pacífico (Sídney)	`arn:aws:lambda:ap-southeast-2:665172237481:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18`
Asia-Pacífico (Tailandia)	`arn:aws:lambda:ap-southeast-7:941377119484:layer:AWS-Parameters-and-Secrets-Lambda-Extension:6`
Asia-Pacífico (Tokio)	`arn:aws:lambda:ap-northeast-1:133490724326:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18`
Canadá (centro)	`arn:aws:lambda:ca-central-1:200266452380:layer:AWS-Parameters-and-Secrets-Lambda-Extension:20`
Oeste de Canadá (Calgary)	`arn:aws:lambda:ca-west-1:243964427225:layer:AWS-Parameters-and-Secrets-Lambda-Extension:8`

Región	ARN
China (Pekín)	`arn:aws-cn:lambda:cn-north-1:287114880934:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18arn:aws-cn:lambda:cn-north-1:287114880934:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18`
China (Ningxia)	`arn:aws-cn:lambda:cn-northwest-1:287310001119:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18arn:aws-cn:lambda:cn-northwest-1:287310001119:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18`
Europa (Fráncfort)	`arn:aws:lambda:eu-central-1:187925254637:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18`
Europa (Irlanda)	`arn:aws:lambda:eu-west-1:015030872274:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18`
Europa (Londres)	`arn:aws:lambda:eu-west-2:133256977650:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18`
Europa (Milán)	`arn:aws:lambda:eu-south-1:325218067255:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18`

Región	ARN
Europa (París)	arn:aws:lambda:eu-west-3:780235371811:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18
Región Europa (España)	arn:aws:lambda:eu-south-2:524103009944:layer:AWS-Parameters-and-Secrets-Lambda-Extension:17
Europa (Estocolmo)	arn:aws:lambda:eu-north-1:427196147048:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18
Israel (Tel Aviv)	arn:aws:lambda:il-central-1:148806536434:layer:AWS-Parameters-and-Secrets-Lambda-Extension:8
Región Europa (Zúrich)	arn:aws:lambda:eu-central-2:772501565639:layer:AWS-Parameters-and-Secrets-Lambda-Extension:15
Región de México (centro)	arn:aws:lambda:mx-central-1:241533131596:layer:AWS-Parameters-and-Secrets-Lambda-Extension:5
Medio Oriente (Baréin)	arn:aws:lambda:me-south-1:832021897121:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18

Región	ARN
Medio Oriente (EAU)	`arn:aws:lambda:me-central-1:858974508948:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18`
América del Sur (São Paulo)	`arn:aws:lambda:sa-east-1:933737806257:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18`
AWS GovCloud (Este de EE. UU.)	`arn:aws-us-gov:lambda:us-gov-east-1:129776340158:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18arn:aws-us-gov:lambda:us-gov-east-1:129776340158:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18`
AWS GovCloud (Oeste de EE. UU.)	`arn:aws-us-gov:lambda:us-gov-west-1:127562683043:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18arn:aws-us-gov:lambda:us-gov-west-1:127562683043:layer:AWS-Parameters-and-Secrets-Lambda-Extension:18`

ARN de extensión para las arquitecturas ARM64 y Mac with Apple silicon

Última actualización: 26 de junio de 2025

Región	ARN
Este de EE. UU. (Ohio)	`arn:aws:lambda:us-east-2:590474943231:layer:AWS-Parame`

Región	ARN
	ters-and-Secrets-Lambda-Ext ension-Arm64:22
Este de EE. UU. (Norte de Virginia)	arn:aws:lambda:us-east-1:17 7933569100:layer:AWS-Parame ters-and-Secrets-Lambda-Ext ension-Arm64:18
Región Oeste de EE. UU. (Norte de California)	arn:aws:lambda:us-west-1:99 7803712105:layer:AWS-Parame ters-and-Secrets-Lambda-Ext ension-Arm64:15
Oeste de EE. UU. (Oregón)	arn:aws:lambda:us-west-2:34 5057560386:layer:AWS-Parame ters-and-Secrets-Lambda-Ext ension-Arm64:18
Región África (Ciudad del Cabo)	arn:aws:lambda:af-south-1:3 17013901791:layer:AWS-Param eters-and-Secrets-Lambda-Ex tension-Arm64:15
Región de Asia-Pacífico (Hong Kong)	arn:aws:lambda:ap-east-1:76 8336418462:layer:AWS-Parame ters-and-Secrets-Lambda-Ext ension-Arm64:17
Asia-Pacífico (Taipéi)	arn:aws:lambda:ap-east-2:89 0742577149:layer:AWS-Parame ters-and-Secrets-Lambda-Ext ension-Arm64:2

Región	ARN
Región de Asia Pacífico (Hyderabad)	`arn:aws:lambda:ap-south-2:070087711984:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:7`
Región Asia-Pacífico (Yakarta)	`arn:aws:lambda:ap-southeast-3:490737872127:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:15`
Asia-Pacífico (Melbourne)	`arn:aws:lambda:ap-southeast-4:090732460067:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:7`
Asia-Pacífico (Malasia)	`arn:aws:lambda:ap-southeast-5:381492012281:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:7`
Asia-Pacífico (Bombay)	`arn:aws:lambda:ap-south-1:176022468876:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:18`
Asia-Pacífico (Osaka)	`arn:aws:lambda:ap-northeast-3:576959938190:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:15`
Región de Asia-Pacífico (Seúl)	`arn:aws:lambda:ap-northeast-2:738900069198:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:15`

Región	ARN
Asia-Pacífico (Singapur)	`arn:aws:lambda:ap-southeast-1:044395824272:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:18`
Asia-Pacífico (Sídney)	`arn:aws:lambda:ap-southeast-2:665172237481:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:18`
Asia-Pacífico (Tailandia)	`arn:aws:lambda:ap-southeast-7:941377119484:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:5`
Asia-Pacífico (Tokio)	`arn:aws:lambda:ap-northeast-1:133490724326:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:18`
Región de Canadá (centro)	`arn:aws:lambda:ca-central-1:200266452380:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:17`
Oeste de Canadá (Calgary)	`arn:aws:lambda:ca-west-1:243964427225:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:7`

Región	ARN
China (Pekín)	arn:aws-cn:lambda:cn-north-1:287114880934:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:7arn:aws-cn:lambda:cn-north-1:287114880934:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:7
China (Ningxia)	arn:aws-cn:lambda:cn-northwest-1:287310001119:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:7arn:aws-cn:lambda:cn-northwest-1:287310001119:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:7
Europa (Fráncfort)	arn:aws:lambda:eu-central-1:187925254637:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:18
Europa (Irlanda)	arn:aws:lambda:eu-west-1:015030872274:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:18
Europa (Londres)	arn:aws:lambda:eu-west-2:133256977650:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:18

Región	ARN
Región Europa (Milán)	arn:aws:lambda:eu-south-1:325218067255:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:15
Región Europa (París)	arn:aws:lambda:eu-west-3:780235371811:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:15
Región Europa (España)	arn:aws:lambda:eu-south-2:524103009944:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:9
Región Europa (Estocolmo)	arn:aws:lambda:eu-north-1:427196147048:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:15
Israel (Tel Aviv)	arn:aws:lambda:il-central-1:148806536434:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:7
Región Europa (Zúrich)	arn:aws:lambda:eu-central-2:772501565639:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:7
Región de México (centro)	arn:aws:lambda:mx-central-1:241533131596:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:4

Región	ARN
Región Medio Oriente (Baréin)	`arn:aws:lambda:me-south-1:832021897121:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:15`
Medio Oriente (EAU)	`arn:aws:lambda:me-central-1:858974508948:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:7`
Región de América del Sur (São Paulo)	`arn:aws:lambda:sa-east-1:933737806257:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:15`
AWS GovCloud (Este de EE. UU.)	`arn:aws-us-gov:lambda:us-gov-east-1:129776340158:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:7arn:aws-us-gov:lambda:us-gov-east-1:129776340158:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:7`
AWS GovCloud (Oeste de EE. UU.)	`arn:aws-us-gov:lambda:us-gov-west-1:127562683043:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:7arn:aws-us-gov:lambda:us-gov-west-1:127562683043:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:7`

Integración a otros productos y servicios

AWS Systems Manager incluye capacidad de integración a los productos y los servicios que se indican en la siguiente tabla.

Ansible	Ansible es una plataforma de automatización de TI que facilita la implementación de las aplicaciones y los sistemas. Systems Manager incluye `AWS-ApplyAnsiblePlaybooks` , un documento de Systems Manager (documento de SSM) que permite la creación de asociaciones de State Manager que ejecuten cuadernos de trabajo de Ansible. Más información Creación de asociaciones que ejecuten manuales de estrategia de Ansible
Chef	Chef es una herramienta de automatización de TI que facilita la implementación de las aplicaciones y los sistemas. Systems Manager proporciona el documento de SSM `AWS-ApplyChefRecipes` , que permite la creación de asociaciones en State Manager, una herramienta de AWS Systems Manager, que ejecuten recetas de Chef. Más información Creación de asociaciones que ejecuten recetas de Chef Systems Manager también se integra con los perfiles Chef InSpec, lo que permite la ejecución de análisis de conformidad y el

	control de los nodos conformes y no conformes . Más información Utilización de perfiles de Chef InSpec con la conformidad de Systems Manager
GitHub	GitHub ofrece alojamiento para el control de versiones de desarrollo de software y la colaboración. Systems Manager incluye `AWS-RunDocument` , un documento de SSM que permite la ejecución de otros documentos de SSM almacenados en GitHub; también incluye el documento `AWS-RunRemoteScript` , el cual permite la ejecución de scripts almacenados en GitHub. Más información • Ejecución de documentos de desde ubicaciones remotas • Ejecución de scripts desde GitHub

Jenkins	Jenkins es un servidor de automatización de código abierto que le permite a los desarroll adores crear, probar e implementar software de manera fiable. Automatización, una herramienta de Systems Manager, puede utilizarse como un paso posterior a la creación para preinstalar versiones de las aplicaciones en Amazon Machine Images (AMIs). Más información Actualización de las AMIs mediante Automatiz ación y Jenkins
ServiceNow	ServiceNow es un sistema de administración de servicios empresariales que permite la administración de los servicios y las operacion es de TI. Automatización, Change Manager, Administr ador de incidentes y OpsCenter son herramien tas de Systems Manager que se integran con ServiceNow a través del Conector de administr ación de servicios de AWS. Con esta integraci ón, se puede ver, crear, actualizar, agregar correspondencia y solucionar casos de AWS Support de ServiceNow. Más información Integración con ServiceNow

Terraform	HashiCorp Terraform es una herramienta de software de infraestructura como código (IaC) de código abierto que proporciona un flujo de trabajo de interfaz de la línea de comandos (CLI) para administrar varios servicios en la nube. En el caso de Systems Manager, puede utilizar Terraform para administrar o aprovisionar lo siguiente: Recursos • aws_ssm_activation • aws_ssm_association • aws_ssm_default_patch_baseline • aws_ssm_document • aws_ssm_maintenance_window • aws_ssm_maintenance_window_target • aws_ssm_maintenance_window_task • aws_ssm_parameter • aws_ssm_patch_baseline • aws_ssm_patch_group • aws_ssm_resource_data_sync • aws_ssm_service_setting Origen de datos • aws_ssm_document • aws_ssm_instances • ssm_maintenance_windows • aws_ssm_parameter • aws_ssm_parameters_by_path • aws_ssm_patch_baseline

Temas

- Ejecución de scripts desde GitHub
- Utilización de perfiles de Chef InSpec con la conformidad de Systems Manager
- Integración con ServiceNow

Ejecución de scripts desde GitHub

En este tema, se describe cómo usar `AWS-RunRemoteScript`, un documento predefinido de Systems Manager (documento de SSM) para descargar scripts desde GitHub, incluidos los cuadernos de trabajo de Ansible y los scripts de Python, Ruby y PowerShell. Al utilizar este documento de SSM, ya no es necesario transferir scripts de forma manual a Amazon Elastic Compute Cloud (Amazon EC2) ni empaquetarlos en documentos de SSM. La integración de AWS Systems Manager a GitHub promueve la infraestructura como código, lo que reduce el tiempo que requiere la administración de nodos mientras se estandarizan las configuraciones en la flota.

También puede crear documentos de SSM personalizados que le permitan descargar y ejecutar scripts u otros documentos de SSM desde ubicaciones remotas. Para obtener más información, consulte Creación de documentos compuestos.

También puede descargar un directorio en el que se incluyen varios scripts. Cuando se ejecuta el script principal en el directorio, Systems Manager también ejecuta los scripts a los que se hace referencia y se incluyen en el directorio.

Tenga en cuenta los siguientes detalles importantes acerca de la ejecución de scripts desde GitHub.

- Systems Manager no comprueba que el script pueda ejecutarse en un nodo. Antes de descargar y ejecutar el script, verifique que el software necesario esté instalado en el nodo. O bien, puede crear un documento compuesto que instale el software mediante Run Command o State Manager, herramientas de AWS Systems Manager, y, a continuación, descargue y ejecute el script.
- Usted es responsable de garantizar el cumplimiento de todos los requisitos de GitHub. Esto incluye la actualización de su token de acceso, según sea necesario. Asegúrese de no superar el número de solicitudes autenticadas o sin autenticar. Para obtener más información, consulte la documentación de GitHub.
- Los repositorios GitHub Enterprise no son compatibles.

Temas

- Ejecute cuadernos de trabajo de Ansible desde GitHub
- Ejecute scripts de Python desde GitHub

Ejecute cuadernos de trabajo de Ansible desde GitHub

En esta sección, se incluyen los procedimientos que lo ayudarán a ejecutar cuadernos de trabajo de Ansible desde GitHub a través de la consola o de AWS Command Line Interface (AWS CLI).

Antes de empezar

Si tiene previsto ejecutar un script almacenado en un repositorio privado de GitHub, cree un parámetro AWS Systems Manager `SecureString` para el token de acceso de seguridad de GitHub. No puede obtener acceso a un script en un repositorio privado de GitHub al pasar manualmente el token por SSH. El token de acceso debe pasarse como parámetro `SecureString` de Systems Manager. Para obtener más información acerca de cómo crear un parámetro `SecureString`, consulte Creación de parámetros de Parameter Store en Systems Manager.

Ejecute un cuaderno de trabajo de Ansible desde GitHub (consola)

Ejecute un cuaderno de trabajo de Ansible desde GitHub

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command.
3. Elija Run command (Ejecutar comando).
4. En la lista Command document (Documento de Command), elija **`AWS-RunRemoteScript`**.
5. En Command parameters haga lo siguiente:
   - En Tipo de origen, seleccione GitHub.
   - En el cuadro Source Info (Información de la fuente), ingrese la información requerida para acceder a la fuente, con el siguiente formato.

```
{
  "owner": "owner_name",
  "repository": "repository_name",
  "getOptions": "branch:branch_name",
  "path": "path_to_scripts_or_directory",
  "tokenInfo": "{{ssm-secure:SecureString_parameter_name}}"
```

```
}
```

En este ejemplo, se descarga un archivo llamado `webserver.yml`.

```
{
    "owner": "TestUser1",
    "repository": "GitHubPrivateTest",
    "getOptions": "branch:myBranch",
    "path": "scripts/webserver.yml",
    "tokenInfo": "{{ssm-secure:mySecureStringParameter}}"
}
```

> **Note**
>
> Solo se requiere `"branch"` si el documento SSM se almacena en una sucursal que no sea `master`.
> Para usar la versión de los scripts que están en una confirmación determinada en su repositorio, use `commitID` con `getOptions` en lugar de `branch`. Por ejemplo: `"getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",`

- En el campo Command Line (Línea de comandos), ingrese los parámetros para la ejecución de scripts. A continuación se muestra un ejemplo.

  `ansible-playbook -i “localhost,” --check -c local webserver.yml`

- (Opcional) En el campo Working Directory (Directorio de trabajo), ingrese el nombre de un directorio del nodo en el que desee descargar y ejecutar el script.
- (Opcional) En Execution Timeout, especifique el número de segundos que esperará el sistema antes de fallar en la ejecución del comando de script.

6. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

> **Tip**
>
> Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

7. En Otros parámetros:

   - En Comentario, ingrese la información acerca de este comando.
   - En Tiempo de espera (segundos), especifique el número de segundos que tiene que esperar el sistema antes de indicar que se ha producido un error en la ejecución del comando general.

8. En Rate control (Control de velocidad):

   - En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

     > **Note**
     >
     > Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

   - En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

9. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

   > **Note**
   >
   > Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

10. En la sección Notificaciones de SNS, seleccione la casilla de verificación Habilitar notificaciones de SNS si desea recibir notificaciones sobre el estado de ejecución de los comandos.

    Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.

11. Seleccione Ejecutar.

Ejecute un cuaderno de trabajo de Ansible desde GitHub con AWS CLI

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Ejecute el siguiente comando para descargar y ejecutar un script de GitHub.

```
aws ssm send-command \
    --document-name "AWS-RunRemoteScript" \
    --instance-ids "instance-IDs"\
    --parameters '{"sourceType":["GitHub"],"sourceInfo":["{\"owner
\":\"owner_name\", \"repository\": \"repository_name\", \"path
\": \"path_to_file_or_directory\", \"tokenInfo\":\"{{ssm-
secure:name_of_your_SecureString_parameter}}\" }"],"commandLine":
["commands_to_run"]}'
```

   A continuación, se presenta un ejemplo de comando que se puede ejecutar en un equipo local Linux.

```
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"]}'
```

Ejecute scripts de Python desde GitHub

En esta sección, se incluyen procedimientos que lo ayudarán a ejecutar scripts de Python desde GitHub con la consola de AWS Systems Manager o la AWS Command Line Interface (AWS CLI).

Ejecute un script de Python desde GitHub (consola)

Ejecute un script de Python desde GitHub

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija Run Command.
3. Elija Run command (Ejecutar comando).
4. En la lista Command document (Documento de Command), elija **`AWS-RunRemoteScript`**.
5. En Parámetros de comando haga lo siguiente:
   - En Tipo de origen, seleccione GitHub.
   - En el cuadro Source Info (Información de la fuente), ingrese la información requerida para acceder a la fuente, con el siguiente formato:

```
{
    "owner": "owner_name",
    "repository": "repository_name",
    "getOptions": "branch:branch_name",
    "path": "path_to_document",
    "tokenInfo": "{{ssm-secure:SecureString_parameter_name}}"
}
```

   En el ejemplo siguiente, se descarga un directorio de scripts denominado complex-script.

```
{
    "owner": "TestUser1",
    "repository": "SSMTestDocsRepo",
    "getOptions": "branch:myBranch",
    "path": "scripts/python/complex-script",
    "tokenInfo": "{{ssm-secure:myAccessTokenParam}}"
}
```

> **Note**
>
> Solo se requiere `"branch"` si sus scripts están almacenados en una rama distinta de `master`.
> Para usar la versión de los scripts que están en una confirmación determinada en su repositorio, use `commitID` con `getOptions` en lugar de `branch`. Por ejemplo: `"getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",`

- En Command Line (Línea de comandos), ingrese parámetros para la ejecución de scripts. A continuación se muestra un ejemplo.

```
mainFile.py argument-1 argument-2
```

  En este ejemplo, se ejecuta `mainFile.py`, que posteriormente puede ejecutar otros scripts en el directorio `complex-script`.

- (Opcional) En Working Directory (Directorio de trabajo), ingrese el nombre de un directorio del nodo en el que desee descargar y ejecutar el script.
- (Opcional) En Execution Timeout (Tiempo de espera de ejecución), especifique el número de segundos que esperará el sistema antes de fallar en la ejecución del comando de script.

6. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

> **Tip**
>
> Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.

7. En Otros parámetros:

   - En Comentario, ingrese la información acerca de este comando.
   - En Tiempo de espera (segundos), especifique el número de segundos que tiene que esperar el sistema antes de indicar que se ha producido un error en la ejecución del comando general.

8. En Rate control (Control de velocidad):

- En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

> **Note**
>
> Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

- En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

9. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

10. En la sección Notificaciones de SNS, seleccione la casilla de verificación Habilitar notificaciones de SNS si desea recibir notificaciones sobre el estado de ejecución de los comandos.

    Para obtener más información acerca de la configuración de las notificaciones de Amazon SNS para Run Command, consulte Cómo monitorear los cambios de estado de Systems Manager mediante las notificaciones de Amazon SNS.

11. Seleccione Ejecutar.

Ejecute un script de Python desde GitHub con AWS CLI

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Ejecute el siguiente comando para descargar y ejecutar un script de GitHub.

```
aws ssm send-command --document-name "AWS-RunRemoteScript" --instance-
ids "instance-IDs" --parameters '{"sourceType":["GitHub"],"sourceInfo":
["{\"owner\":\"owner_name\", \"repository\":\"repository_name\", \"path\":
 \"path_to_script_or_directory\"}"],"commandLine":["commands_to_run"]}'
```

   A continuación se muestra un ejemplo.

```
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 "]}'
```

   En este ejemplo, se descarga un directorio de scripts llamado `complex-script`. La entrada `commandLine` ejecuta `mainFile.py`, que luego puede ejecutar otros scripts en el directorio `complex-script`.

Utilización de perfiles de Chef InSpec con la conformidad de Systems Manager

AWS Systems Manager se integra con Chef InSpec. Chef InSpec es un marco de pruebas de código abierto que le permite crear perfiles de lenguaje natural para almacenarlos en GitHub o Amazon Simple Storage Service (Amazon S3). A continuación, puede utilizar Systems Manager para ejecutar análisis de conformidad y ver cuáles nodos son conformes y cuáles no. Un perfil es un requisito de seguridad, conformidad o política de un entorno informático. Por ejemplo, puede crear perfiles que lleven a cabo las siguientes comprobaciones cuando se analicen los nodos con Cumplimiento, una herramienta de AWS Systems Manager:

- Comprobar si determinados puertos están abiertos o cerrados.
- Comprobar si determinadas aplicaciones se están ejecutando.
- Comprobar si determinados paquetes están instalados.
- Comprobar las claves de registro de Windows de determinadas propiedades.

Puede crear perfiles de InSpec solo para las instancias de Amazon Elastic Compute Cloud (Amazon EC2) que administre con Systems Manager. No se admiten servidores en las instalaciones ni máquinas virtuales (VM). El siguiente ejemplo de perfil de Chef InSpec verifica si el puerto 22 está abierto.

```
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 incluye una colección de recursos que le ayudan a escribir rápidamente comprobaciones y controles de auditoría. InSpec utiliza el lenguaje específico de dominio (DSL) de InSpec para escribir estos controles en Ruby. También puede utilizar perfiles creados por una gran comunidad de usuarios de InSpec. Por ejemplo, el proyecto DevSec chef-os-hardening que se encuentra en GitHub incluye decenas de perfiles que lo pueden ayudar a proteger los nodos. Puede crear y almacenar perfiles en GitHub o en Amazon S3.

Funcionamiento

El proceso de utilización de perfiles de InSpec con Compliance funciona de la siguiente manera:

1. Identifique los perfiles predefinidos de InSpec que quiera utilizar o créelos. Puede utilizar perfiles predefinidos de GitHub para comenzar. Para obtener información sobre cómo crear perfiles de InSpec propios, consulte Perfiles de Chef InSpec.
2. Almacene perfiles en un repositorio público o privado de GitHub o en un bucket de S3.
3. Ejecute Compliance con los perfiles de InSpec mediante el documento de Systems Manager (documento de SSM) `AWS-RunInspecChecks`. Puede comenzar un análisis de Cumplimiento con Run Command, una herramienta de AWS Systems Manager, para efectuar análisis bajo

demanda, o bien, puede programar análisis regulares de Cumplimiento a través de State Manager, otra herramienta de AWS Systems Manager.

4. Utilice la API de Compliance o la consola de Compliance para identificar nodos no conformes.

> **Note**
>
> Observe la siguiente información.
>
> - Chef utiliza un cliente en los nodos para procesar el perfil. No es necesario instalar el cliente. Cuando Systems Manager ejecuta el documento de SSM `AWS-RunInspecChecks`, el sistema verifica si el cliente está instalado. Si no lo está, Systems Manager instala el cliente de Chef durante el análisis y lo desinstala al terminar.
> - Como se describe en este tema, cuando se ejecuta el documento de SSM `AWS-RunInspecChecks`, se asigna una entrada de conformidad de tipo `Custom:Inspec` a cada nodo de destino. Para asignar este tipo de conformidad, el documento llama a la operación PutComplianceItems de la API.

Ejecutar un análisis de InSpec Compliance

Esta sección contiene información acerca de cómo ejecutar un análisis de conformidad de InSpec con la consola de Systems Manager y la AWS Command Line Interface (AWS CLI). En el procedimiento con la consola, se muestra cómo configurar State Manager para que ejecute el análisis. En el procedimiento con la AWS CLI, se muestra cómo configurar Run Command para que ejecute el análisis.

Ejecución de un análisis de conformidad de InSpec con State Manager (consola)

Para ejecutar un análisis de InSpec Compliance con State Manager mediante la consola de AWS Systems Manager

1. Abra la consola de AWS Systems Manager en https://console.aws.amazon.com/systems-manager/.
2. En el panel de navegación, elija State Manager.
3. Elija Crear asociación.
4. En la sección Provide association details (Proporcionar detalles de la asociación), escriba un nombre.

5. En la lista Document (Documento), elija **AWS-RunInspecChecks**.
6. En la lista Document version (Versión del documento), elija Latest at runtime (El último durante el tiempo de ejecución).
7. En la sección Parámetros, en la lista Tipo de origen, seleccione GitHub o S3.

   Si selecciona GitHub, introduzca a continuación la ruta hacia un perfil InSpec en un repositorio público o privado de GitHub en el campo Información del origen. A continuación, se muestra un ejemplo de ruta hacia un perfil público proporcionado por el equipo de Systems Manager desde la siguiente ubicación: https://github.com/awslabs/amazon-ssm/tree/master/Compliance/InSpec/PortCheck.

   ```
   {"owner":"awslabs","repository":"amazon-ssm","path":"Compliance/InSpec/
   PortCheck","getOptions":"branch:master"}
   ```

   Si elige S3, escriba a continuación una URL válida hacia un perfil de InSpec en un bucket de S3 en el campo Source Info (Información del origen).

   Para obtener más información sobre cómo se integra Systems Manager en GitHub y Amazon S3, consulte Ejecución de scripts desde GitHub.
8. En la sección Targets (Destinos), para elegir los nodos administrados en los que desea ejecutar esta operación, especifique las etiquetas, seleccione las instancias o los dispositivos de borde manualmente o especifique un grupo de recursos.

   > **Tip**
   >
   > Si un nodo administrado que espera ver no aparece en la lista, consulte Solución de problemas de disponibilidad de nodos administrados para obtener consejos de solución de problemas.
9. En la sección Specify schedule (Especificar programación), utilice las opciones del generador de programaciones para crear una que especifique cuándo desea que se ejecute el análisis de Compliance.
10. En Rate control (Control de velocidad):

    - En Concurrency (Simultaneidad), especifique un número o un porcentaje de los nodos administrados en los que desea ejecutar el comando al mismo tiempo.

> **Note**
>
> Si seleccionó los destinos mediante la especificación de etiquetas aplicadas a nodos administrados o de grupos de recursos de AWS y no está seguro de cuántos nodos administrados tienen destino, limite el número de destinos que puede ejecutar el documento al mismo tiempo. Para ello, especifique un porcentaje.

- En Error threshold (Umbral de errores), especifique cuándo desea parar la ejecución del comando en los demás nodos administrados después de que haya fallado en un número o un porcentaje de los nodos. Por ejemplo, si especifica tres errores, Systems Manager dejará de enviar el comando cuando se reciba el cuarto error. Los nodos administrados que estén procesando el comando todavía pueden enviar errores.

11. (Opcional) En Opciones de salida, para guardar la salida del comando en un archivo, seleccione el cuadro Write command output to an S3 bucket. Ingrese los nombres del bucket y del prefijo (carpeta) en los cuadros.

> **Note**
>
> Los permisos de S3 que conceden la capacidad de escribir datos en un bucket de S3 son los del perfil de instancia (para instancias de EC2) o rol de servicio de IAM (máquinas activadas de manera híbrida) asignados a la instancia, no los del usuario de IAM que realiza esta tarea. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager o Creación de un rol de servicio de IAM para un entorno híbrido. Además, si el bucket de S3 especificado se encuentra en una Cuenta de AWS diferente, asegúrese de que el perfil de instancias o el rol de servicio de IAM asociado al nodo administrado tenga los permisos necesarios para escribir en ese bucket.

12. Elija Crear asociación. El sistema crea la asociación y ejecuta el análisis de Compliance automáticamente.
13. Espere unos minutos para hasta que finalice el análisis y, a continuación, elija Compliance (Conformidad) en el panel de navegación.
14. En Corresponding managed instances (Instancias administradas correspondientes), localice los nodos en los que la columna Compliance Type (Tipo de conformidad) sea Custom:Inspec.
15. Elija un ID de nodo para ver los detalles de los estados no conformes.

Ejecución de un análisis de conformidad de InSpec con Run Command (AWS CLI)

1. Si aún no lo ha hecho, instale y configure la AWS Command Line Interface (AWS CLI).

   Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.

2. Ejecute uno de los siguientes comandos para ejecutar un perfil de InSpec desde GitHub o Amazon S3.

   El comando de la usa los siguientes parámetros:

   - sourceType: GitHub o Amazon S3
   - sourceInfo: URL de la carpeta del perfil InSpec en GitHub o en un bucket de S3. La carpeta debe contener el archivo base InSpec (*.yml) y todos los controles relacionados (*.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"}"]}'
   ```

   A continuación se muestra un ejemplo.

   ```
   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/amzn-s3-demo-
   bucket/Inspec.yml_file\"}"]}'
   ```

   A continuación se muestra un ejemplo.

```
aws ssm send-command --document-name "AWS-RunInspecChecks" --targets
 '[{"Key":"tag:testEnvironment","Values":["webServers"]}]' --
parameters'{"sourceType":["S3"],"sourceInfo":["{\"path\":\"https://s3.aws-api-
domain/amzn-s3-demo-bucket/InSpec/PortCheck.yml\"}"]}'
```

3. Ejecute el siguiente comando para ver un resumen del análisis de Compliance.

```
aws ssm list-resource-compliance-summaries --filters
 Key=ComplianceType,Values=Custom:Inspec
```

4. Ejecute el siguiente comando para ver los detalles de un nodo no conforme.

```
aws ssm list-compliance-items --resource-ids node_ID --resource-type
 ManagedInstance --filters Key=DocumentName,Values=AWS-RunInspecChecks
```

Integración con ServiceNow

ServiceNow ofrece un sistema de administración de servicios basado en la nube para crear y administrar flujos de trabajo en la organización, como para los servicios de TI, los sistemas de tickets y la asistencia. El Conector de administración de servicios de AWS integra ServiceNow con Systems Manager para aprovisionar, administrar y operar recursos de AWS desde ServiceNow. Puede utilizar el Conector de Administración de Servicios AWS para integrar ServiceNow con Automatización, Change Manager, Administración de incidentes y OpsCenter, que son herramientas de AWS Systems Manager.

Con ServiceNow, puede realizar las tareas siguientes:

- Ejecutar manuales de automatización desde Systems Manager.
- Ver, actualizar y resolver incidentes desde OpsItems de Systems Manager.
- Ver y administrar los elementos operativos, como los incidentes, a través de OpsCenter de Systems Manager.
- Ver y ejecutar las solicitudes de cambio de Systems Manager a partir de una lista seleccionada de plantillas de cambios aprobadas previamente.
- Administrar y resolver los incidentes relacionados las aplicaciones alojadas en AWS mediante la integración con Administrador de incidentes.

> **Note**
>
> Para obtener más información sobre la integración con ServiceNow, consulte Configuración de las integraciones de servicios de AWS en la Guía del administrador del Conector de administración de servicios de AWS.

Referencia de AWS Systems Manager

La siguiente información y temas pueden ayudarle a implementar mejor soluciones AWS Systems Manager.

Entidad principal

En AWS Identity and Access Management (IAM), puede conceder o denegarle a un servicio acceso a los recursos con el uso del elemento de la política Principal. El valor del elemento de la política Principal para Systems Manager es `ssm.amazonaws.com`.

Regiones de AWS y puntos de conexión admitidos

Consulte Los puntos de conexión para el servicio de Systems Manager en la Referencia general de Amazon Web Services.

Service Quotas

Consulte las Service Quotas de Systems Manager en la Referencia general de Amazon Web Services.

Referencia de la API

Consulte Referencia de la API de AWS Systems Manager.

Referencia de los comandos de la AWS CLI

Consulte la Sección de AWS Systems Manager de la referencia de los comandos de la AWS CLI.

Referencia de Cmdlet de Herramientas de AWS para PowerShell

Consulte la Sección de AWS Systems Manager de la referencia de Cmdlet de las Herramientas de AWS para PowerShell.

SSM AgentRepositorio en GitHub

Consulte aws/amazon-ssm-agent.

Pregunte

Problemas de Systems Manager en AWS re:Post

Blog de noticias de AWS

Herramientas de administración

Más temas de referencia

- Cómo utilizar este servicio con un AWS SDK
- Referencia: buckets de Amazon S3 para operaciones de aplicación de revisiones
- Referencia: patrones y tipos de eventos de Amazon EventBridge para Systems Manager
- Referencia: expresiones cron y rate para Systems Manager
- Referencia: ec2messages, ssmmessages y otras operaciones de la API
- Referencia: Formatos de cadenas de fecha y hora para Systems Manager

Cómo utilizar este servicio con un AWS SDK

Los kits de desarrollo de software (SDK) de AWS se encuentran disponibles en muchos lenguajes de programación populares. Cada SDK proporciona una API, ejemplos de código y documentación que facilitan a los desarrolladores la creación de aplicaciones en su lenguaje preferido.

Documentación de SDK	Ejemplos de código
AWS SDK para C++	Ejemplos de código de la AWS SDK para C++
AWS CLI	Ejemplos de código de la AWS CLI
AWS SDK para Go	Ejemplos de código de la AWS SDK para Go
AWS SDK para Java	Ejemplos de código de la AWS SDK para Java
AWS SDK para JavaScript	Ejemplos de código de la AWS SDK para JavaScript
AWS SDK para Kotlin	Ejemplos de código de la AWS SDK para Kotlin
AWS SDK para .NET	Ejemplos de código de la AWS SDK para .NET
AWS SDK para PHP	Ejemplos de código de la AWS SDK para PHP
Herramientas de AWS para PowerShell	Ejemplos de código de la Herramientas de AWS para PowerShell

Documentación de SDK	Ejemplos de código
AWS SDK para Python (Boto3)	Ejemplos de código de la AWS SDK para Python (Boto3)
AWS SDK para Ruby	Ejemplos de código de la AWS SDK para Ruby
AWS SDK para Rust	Ejemplos de código de la AWS SDK para Rust
AWS SDK para SAP ABAP	Ejemplos de código de la AWS SDK para SAP ABAP
AWS SDK para Swift	Ejemplos de código de la AWS SDK para Swift

Ejemplo de disponibilidad

¿No encuentra lo que necesita? Solicite un ejemplo de código a través del enlace de Enviar comentarios que se encuentra al final de esta página.

Referencia: buckets de Amazon S3 para operaciones de aplicación de revisiones

En el curso de la realización de diversas operaciones de aplicación de revisiones de Patch Manager, AWS Systems Manager Agent (SSM Agent) accede a determinados buckets de Amazon Simple Storage Service (Amazon S3) que son propiedad de Amazon Web Services (AWS) que los mantiene. Se puede acceder públicamente a estos buckets de S3 y, de forma predeterminada, SSM Agent se conecta a ellos a través de llamadas a HTTP.

Sin embargo, si utiliza un punto de conexión de nube privada virtual (VPC) en las operaciones de Systems Manager, debe conceder un permiso explícito en un perfil de instancia de Amazon Elastic Compute Cloud (Amazon EC2) para Systems Manager o en un rol de servicio para equipos que no sean de EC2 en un entorno híbrido y multinube. De lo contrario, sus recursos no puede acceder a estos buckets públicos.

En este tema de referencia se enumeran los buckets relacionados con la aplicación de revisiones para cada Región de AWS compatible.

Para obtener más información sobre la especificación de estos buckets S3 en los perfiles de instancia EC2, consulte Comunicaciones de SSM Agent con buckets de S3 administrados de AWS.

Para obtener más información sobre los puntos de conexión de VPC con Systems Manager, consulte Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager.

Temas

- Buckets que contienen documentos de comandos SSM para operaciones de aplicación de revisiones (Linux y Windows Server)
- Buckets que contienen documentos de comandos SSM para operaciones de aplicación de revisiones (macOS)
- Buckets que contienen instantáneas de la línea de base de revisiones gestionadas de AWS

Buckets que contienen documentos de comandos SSM para operaciones de aplicación de revisiones (Linux y Windows Server)

Los buckets con el formato `aws-patch-manager-`*`region`*`-`*`unique-suffix` contienen los siguientes documentos que se utilizan en las operaciones de aplicación de revisiones de Patch Manager en los sistemas operativos de Linux y Windows Server:

- `AWS-RunPatchBaseline`
- `AWS-RunPatchBaselineAssociation`
- `AWS-RunPatchBaselineWithHooks`
- `AWS-InstanceRebootWithHooks`
- `AWS-PatchAsgInstance`
- `AWS-PatchInstanceWithRollback`

Nombre de la región	Código de región	Bucket de **`aws-patch-manager-`** ***`region`*****`-`*****`suffix`**
Este de EE. UU. (Ohio)	us-east-2	aws-patch-manager-us-east-2-552881074

Nombre de la región	Código de región	Bucket de **aws-patch-manager-** ***region-suffix***
Este de EE. UU. (Norte de Virginia)	us-east-1	aws-patch-manager-us-east-1-1970c647d
Oeste de EE. UU. (Norte de California)	us-west-1	aws-patch-manager-us-west-1-8badb4304
Oeste de EE. UU. (Oregón)	us-west-2	aws-patch-manager-us-west-2-34d7f99f8
África (Ciudad del Cabo)	af-south-1	aws-patch-manager-af-south-1-bdd5f65a9
Asia-Pacífico (Hong Kong)	ap-east-1	aws-patch-manager-ap-east-1-632356271
Asia-Pacífico (Hyderabad)	ap-south-2	aws-patch-manager-ap-south-2-32f4b4128
Asia-Pacífico (Yakarta)	ap-southeast-3	aws-patch-manager-ap-southeast-3-aa48fc462
Asia-Pacífico (Melbourne)	ap-southeast-4	aws-patch-manager-ap-southeast-4-01e2c40d3
Asia Pacífico (Bombay)	ap-south-1	aws-patch-manager-ap-south-1-cb7c62ff9
Asia Pacífico (Osaka)	ap-northeast-3	aws-patch-manager-ap-northeast-3-67373598a
Asia Pacífico (Seúl)	ap-northeast-2	aws-patch-manager-ap-northeast-2-10467995c
Asia-Pacífico (Singapur)	ap-southeast-1	aws-patch-manager-ap-southeast-1-7fdfd9ef7

Nombre de la región	Código de región	Bucket de **aws-patch-manager-** ***region-suffix***
Asia Pacífico (Sídney)	ap-southeast-2	aws-patch-manager-ap-southeast-2-17283a275
Asia-Pacífico (Tokio)	ap-northeast-1	aws-patch-manager-ap-northeast-1-4849fa78f
Canadá (centro)	ca-central-1	aws-patch-manager-ca-central-1-3148e69e3
Oeste de Canadá (Calgary)	ca-west-1	aws-patch-manager-ca-west-1-9e3a4b2f9
Europe (Fráncfort)	eu-central-1	aws-patch-manager-eu-central-1-9163fdaaf
Europa (Irlanda)	eu-west-1	aws-patch-manager-eu-west-1-5522fb710
Europa (Londres)	eu-west-2	aws-patch-manager-eu-west-2-902a2bc74
Europa (Milán)	eu-south-1	aws-patch-manager-eu-south-1-c52f3f594
Europa (París)	eu-west-3	aws-patch-manager-eu-west-3-29bf85721
Europa (España)	eu-south-2	aws-patch-manager-eu-south-2-a4cf248b1
Europa (Estocolmo)	eu-north-1	aws-patch-manager-eu-north-1-795879e9b
Europa (Zúrich)	eu-central-2	aws-patch-manager-eu-central-2-184ce43c8

Nombre de la región	Código de región	Bucket de **aws-patch-manager-** ***region-suffix***
Israel (Tel Aviv)	il-central-1	aws-patch-manager-il-central-1-e221cb57b
Medio Oriente (Baréin)	me-south-1	aws-patch-manager-me-south-1-a53fc9dce
Medio Oriente (EAU)	me-central-1	aws-patch-manager-me-central-1-2932f2f80
América del Sur (São Paulo)	sa-east-1	aws-patch-manager-sa-east-1-ddf4b6a09

Buckets que contienen documentos de comandos SSM para operaciones de aplicación de revisiones (macOS)

Los buckets con el formato `aws-patchmanager-macos-`*`region`*`-`*`unique-suffix* contienen los siguientes documentos que se utilizan en las operaciones de aplicación de revisiones de Patch Manager en el sistema operativo de macOS:

- `AWS-RunPatchBaseline`
- `AWS-RunPatchBaselineAssociation`
- `AWS-RunPatchBaselineWithHooks`
- `AWS-InstanceRebootWithHooks`
- `AWS-PatchAsgInstance`
- `AWS-PatchInstanceWithRollback`

Nombre de la región	Código de región	Bucket de **aws-patchmanager-macos-** ***region-suffix***
Este de EE. UU. (Ohio)	us-east-2	aws-patchmanager-macos-us-east-2-552881074

Nombre de la región	Código de región	Bucket de **aws-patchmanager-macos-*region-suffix***
Este de EE. UU. (Norte de Virginia)	us-east-1	aws-patchmanager-macos-us-east-1-1970c647d
Oeste de EE. UU. (Norte de California)	us-west-1	aws-patchmanager-macos-us-west-1-8badb4304
Oeste de EE. UU. (Oregón)	us-west-2	aws-patchmanager-macos-us-west-2-34d7f99f8
África (Ciudad del Cabo)	af-south-1	aws-patchmanager-macos-af-south-1-bdd5f65a9
Asia-Pacífico (Hong Kong)	ap-east-1	aws-patchmanager-macos-ap-east-1-632356271
Asia-Pacífico (Hyderabad)	ap-south-2	aws-patchmanager-macos-ap-south-2-32f4b4128
Asia-Pacífico (Yakarta)	ap-southeast-3	aws-patchmanager-macos-ap-southeast-3-aa48fc462
Asia-Pacífico (Melbourne)	ap-southeast-4	aws-patchmanager-macos-ap-southeast-4-01e2c40d3
Asia Pacífico (Bombay)	ap-south-1	aws-patchmanager-macos-ap-south-1-cb7c62ff9
Asia Pacífico (Osaka)	ap-northeast-3	aws-patchmanager-macos-ap-northeast-3-67373598a
Asia Pacífico (Seúl)	ap-northeast-2	aws-patchmanager-macos-ap-northeast-2-10467995c
Asia-Pacífico (Singapur)	ap-southeast-1	aws-patchmanager-macos-ap-southeast-1-7fdfd9ef7

Nombre de la región	Código de región	Bucket de **aws-patchmanager-macos-*region-suffix***
Asia Pacífico (Sídney)	ap-southeast-2	aws-patchmanager-macos-ap-southeast-2-17283a275
Asia-Pacífico (Tokio)	ap-northeast-1	aws-patchmanager-macos-ap-northeast-1-4849fa78f
Canadá (centro)	ca-central-1	aws-patchmanager-macos-ca-central-1-3148e69e3
Oeste de Canadá (Calgary)	ca-west-1	aws-patchmanager-macos-ca-west-1-9e3a4b2f9
Europe (Fráncfort)	eu-central-1	aws-patchmanager-macos-eu-central-1-9163fdaaf
Europa (Irlanda)	eu-west-1	aws-patchmanager-macos-eu-west-1-5522fb710
Europa (Londres)	eu-west-2	aws-patchmanager-macos-eu-west-2-902a2bc74
Europa (Milán)	eu-south-1	aws-patchmanager-macos-eu-south-1-c52f3f594
Europa (París)	eu-west-3	aws-patchmanager-macos-eu-west-3-29bf85721
Europa (España)	eu-south-2	aws-patchmanager-macos-eu-south-2-a4cf248b1
Europa (Estocolmo)	eu-north-1	aws-patchmanager-macos-eu-north-1-795879e9b
Europa (Zúrich)	eu-central-2	aws-patchmanager-macos-eu-central-2-184ce43c8

Nombre de la región	Código de región	Bucket de **aws-patchmanager-macos-*region-suffix***
Israel (Tel Aviv)	il-central-1	aws-patchmanager-macos-il-central-1-e221cb57b
Medio Oriente (Baréin)	me-south-1	aws-patchmanager-macos-me-south-1-a53fc9dce
Medio Oriente (EAU)	me-central-1	aws-patchmanager-macos-me-central-1-2932f2f80
América del Sur (São Paulo)	sa-east-1	aws-patchmanager-macos-sa-east-1-ddf4b6a09

Buckets que contienen instantáneas de la línea de base de revisiones gestionadas de AWS

Los buckets con el formato `patch-baseline-snapshot-`*`region`* o `patch-baseline-snapshot-`*`region`*`-`*`unique-suffix`* contienen instantáneas de la línea de base de revisiones gestionadas de AWS. El acceso a este bucket S3 es necesario si utiliza cualquiera de los siguientes documentos de SSM:

- `AWS-RunPatchBaseline`
- `AWS-RunPatchBaselineAssociation`
- `AWS-RunPatchBaselineWithHooks`
- `AWS-ApplyPatchBaseline` (un documento de SSM heredado)

Nombre de la región	Código de región	Bucket de **patch-baseline-snapshot-***
Este de EE. UU. (Ohio)	us-east-2	patch-baseline-snapshot-us-east-2

Nombre de la región	Código de región	Bucket de **patch-baseline-snapshot-***
Este de EE. UU. (Norte de Virginia)	us-east-1	patch-baseline-snapshot-us-east-1
Oeste de EE. UU. (Norte de California)	us-west-1	patch-baseline-snapshot-us-west-1
Oeste de EE. UU. (Oregón)	us-west-2	patch-baseline-snapshot-us-west-2
África (Ciudad del Cabo)	af-south-1	patch-baseline-snapshot-af-south-1-tbxdb5b9
Asia-Pacífico (Hong Kong)	ap-east-1	patch-baseline-snapshot-ap-east-1
Asia-Pacífico (Hyderabad)	ap-south-2	patch-baseline-snapshot-ap-south2-50209442
Asia-Pacífico (Yakarta)	ap-southeast-3	patch-baseline-snapshot-ap-southeast-3-be0a3174
Asia-Pacífico (Melbourne)	ap-southeast-4	patch-baseline-snapshot-ap-southeast-4-dc6f76ce
Asia Pacífico (Bombay)	ap-south-1	patch-baseline-snapshot-ap-south-1
Asia Pacífico (Osaka)	ap-northeast-3	patch-baseline-snapshot-ap-northeast-3
Asia Pacífico (Seúl)	ap-northeast-2	patch-baseline-snapshot-ap-northeast-2
Asia-Pacífico (Singapur)	ap-southeast-1	patch-baseline-snapshot-ap-southeast-1

Nombre de la región	Código de región	Bucket de **patch-baseline-snapshot-***
Asia Pacífico (Sídney)	ap-southeast-2	patch-baseline-snapshot-ap-southeast-2
Asia-Pacífico (Tokio)	ap-northeast-1	patch-baseline-snapshot-ap-northeast-1
Canadá (centro)	ca-central-1	patch-baseline-snapshot-ca-central-1
Oeste de Canadá (Calgary)	ca-west-1	patch-baseline-snapshot-ca-west-1
Europe (Fráncfort)	eu-central-1	patch-baseline-snapshot-eu-central-1
Europa (Irlanda)	eu-west-1	patch-baseline-snapshot-eu-west-1
Europa (Londres)	eu-west-2	patch-baseline-snapshot-eu-west-2
Europa (Milán)	eu-south-1	patch-baseline-snapshot-eu-south-1
Europa (París)	eu-west-3	patch-baseline-snapshot-eu-west-3
Europa (España)	eu-south-2	patch-baseline-snapshot-eu-south-2-df2c9d70
Europa (Estocolmo)	eu-north-1	patch-baseline-snapshot-eu-north-1
Europa (Zúrich)	eu-central-2	patch-baseline-snapshot-eu-central-2

Nombre de la región	Código de región	Bucket de **patch-baseline-snapshot-***
Israel (Tel Aviv)	il-central-1	patch-baseline-snapshot-il-central-1
Medio Oriente (Baréin)	me-south-1	patch-baseline-snapshot-me-south-1-uduvl7q8
Medio Oriente (EAU)	me-central-1	patch-baseline-snapshot-me-central-1
América del Sur (São Paulo)	sa-east-1	patch-baseline-snapshot-sa-east-1

Referencia: patrones y tipos de eventos de Amazon EventBridge para Systems Manager

> **Note**
>
> Amazon EventBridge es la forma preferida de administrar sus eventos. CloudWatch Events y EventBridge son el mismo servicio subyacente y la misma API, pero EventBridge ofrece más características. Los cambios que realice en CloudWatch o EventBridge se reflejarán en cada consola. Para más información, consulte la Guía del usuario de Amazon EventBridge.

Con Amazon EventBridge, puede crear reglas que coincidan con eventos entrantes y dirigirlos a destinos para su procesamiento.

Un evento indica un cambio en un entorno en sus propias aplicaciones, aplicaciones de software como servicio (SaaS) o un Servicio de AWS. Los eventos se producen en la medida de lo posible. Después de detectar un tipo de evento especificado en una regla, EventBridge lo dirige a un destino especificado para su procesamiento. Los destinos pueden incluir instancias de Amazon Elastic Compute Cloud (Amazon EC2), funciones de AWS Lambda, flujos de Amazon Kinesis, tareas de Amazon Elastic Container Service (Amazon ECS), máquinas de estado de AWS Step Functions,

temas de Amazon Simple Notification Service (Amazon SNS), colas de Amazon Simple Queue Service (Amazon SQS), destinos integrados y muchos más.

Para obtener información acerca de la creación de reglas de EventBridge, consulte los siguientes temas:

- Monitoreo de eventos de Systems Manager con Amazon EventBridge
- Ejemplos de eventos de Amazon EventBridge para Systems Manager
- Getting started with Amazon EventBridge (Introducción a Amazon EventBridge) en la Guía del usuario de Amazon EventBridge

En lo que resta de este tema, se describen los tipos de eventos de Systems Manager que puede incluir en las reglas de EventBridge.

Tipo de evento: Automation

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
Notificación de cambio de estado de ejecución de automatización de EC2	Cambia el estado general de un flujo de trabajo de automatización. Puede agregar uno o más de los siguientes cambios de estado a una regla de evento: • Approved • Cancelado • Con error • PendingApproval • PendingChangeCalendarOverride • Rechazada • Programados • Success • TimedOut

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
Notificación de cambio de estado de paso de automatización de EC2	Cambia el estado de un paso específico en el flujo de trabajo de la automatización. Puede agregar uno o más de los siguientes cambios de estado a una regla de evento: • Cancelado • Con error • Success • TimedOut

Tipo de evento: Change Calendar

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
Cambio de estado del calendario	Cambia el estado de Change Calendar. Puede agregar uno o los dos cambios de estado siguientes a una regla de evento: • OPEN • CLOSED Actualmente, no se admiten cambios de estado para calendarios compartidos desde otras Cuentas de AWS.

Tipo de evento: Change Manager

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
Actualización del estado de la solicitud de cambio	El estado de una solicitud de cambio de Change Manager. Puede realizar los siguientes estados en una regla de evento: • Approved • Rechazada • InProgress

Tipo de evento: conformidad de la configuración

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
Cambio de estado en la conformidad de la configuración	Cambia el estado de un nodo administrado, ya sea para la conformidad de asociación o la conformidad de revisiones. Puede agregar uno o más de los siguientes cambios de estado a una regla de evento: • compliant • non_compliant

Tipo de evento: Inventory

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
Cambio de estado de recursos de Inventory	La eliminación de un inventario personalizado y una llamada PutInventory que utiliza una

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
	versión de esquema anterior. Puede agregar uno o más de los siguientes cambios de estado a una regla de evento: • Evento de tipo de inventario personalizado eliminado en un nodo específico. EventBridge envía un evento por nodo por tipo de inventario personalizado. • Evento de tipo de inventario personalizado eliminado en todos los nodos. • Llamada PutInventory con evento de versión de esquema anterior. EventBridge envía este evento cuando la versión del esquema es menor que el esquema actual. Este evento se aplica a todos los tipos de inventario. Para obtener más información, consulte Uso de EventBridge para monitorear eventos de Inventory.

Tipo de evento: periodo de mantenimiento

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
Notificación de cambio de estado del periodo de mantenimiento	Cambia el estado general de uno o más periodos de mantenimiento. Puede agregar uno o más de los siguientes cambios de estado a una regla de evento: • DISABLED • ENABLED

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
Notificación del registro de destino del periodo de mantenimiento	Cambia el estado de uno o más destinos del periodo de mantenimiento. Puede agregar uno o más de los siguientes cambios de estado a una regla de evento: • DEREGISTERED • REGISTERED • UPDATED
Notificación de cambio de estado de ejecución del periodo de mantenimiento	El estado general de un periodo de mantenimiento cambia mientras se está ejecutando. Puede agregar uno o más de los siguientes cambios de estado a una regla de evento: • CANCELLED • CANCELLING • ERROR • IN_PROGRESS • PENDIENTE • SKIPPED_OVERLAPPING • SUCCESS • TIMED_OUT

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
Notificación de cambio de estado de ejecución de la tarea del periodo de mantenimiento	El estado de una tarea en un periodo de mantenimiento cambia mientras se está ejecutando. Puede agregar uno o más de los siguientes cambios de estado a una regla de evento: • CANCELLED • CANCELLING • ERROR • IN_PROGRESS • SUCCESS • TIMED_OUT

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
Notificación de cambio de estado de invocación de destino de la tarea del periodo de mantenimiento	Cambia el estado de una tarea del periodo de mantenimiento en un destino específico. Esta notificación es totalmente compatible solo para tareas Run Command. Para este tipo de tarea, puede agregar uno o más de los siguientes cambios de estado a una regla de evento: • CANCELLED • CANCELLING • ERROR • IN_PROGRESS • SUCCESS • TIMED_OUT Para Automation, AWS Lambda y las tareas de AWS Step Functions, EventBridge solo informa de los estados `IN_PROGRESS` y `COMPLETE`. `COMPLETE` informa si la tarea se ejecuta correctamente o no.
Notificación de registro de tareas del periodo de mantenimiento	Cambia el estado de una o más tareas del periodo de mantenimiento. Puede agregar uno o más de los siguientes cambios de estado a una regla de evento: • DEREGISTERED • REGISTERED • UPDATED

Tipo de evento: OpsCenter

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
Cree OpsItem	Se produce cuando se crea un OpsItem. Puede añadir reglas para uno de los tipos de OpsItem siguientes: • /aws/issue • /aws/task • /aws/insight • /aws/actionitem
Actualización de OpsItem	Se produce cuando se actualiza un OpsItem. Puede añadir reglas para uno de los tipos de OpsItem siguientes: • /aws/issue • /aws/task • /aws/insight • /aws/actionitem

Tipo de evento: Parameter Store

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
Cambio en Parameter Store	Cambia el estado de un parámetro. Puede agregar uno o más de los siguientes cambios de estado a una regla de evento: • Creación • Actualización • Eliminar

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
	• LabelParameterVersion Para obtener más información, consulte Configuración de reglas de EventBridge para parámetros y políticas de parámetros.
Acción de política de Parameter Store	Se cumple una condición de un cambio de política de parámetro avanzado. Puede agregar uno o más de los siguientes cambios de estado a una regla de evento: • Expiration • ExpirationNotification • NoChangeNotification Para obtener más información, consulte Configuración de reglas de EventBridge para parámetros y políticas de parámetros.

Tipo de evento: Run Command

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
Notificación de cambio de estado de invocación de comando de EC2	Cambia el estado de un comando enviado a una instancia administrada individual. Puede agregar uno o más de los siguientes cambios de estado a una regla de evento: • Success • InProgress • TimedOut

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
	• Cancelado • Con error
Notificación de cambio de estado de comando de EC2	Cambia el estado general de un comando. Puede agregar uno o más de los siguientes cambios de estado a una regla de evento: • Success • InProgress • TimedOut • Cancelado • Con error

Tipo de evento: State Manager

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
Cambio de estado de asociación de EC2 de State Manager	El estado general de una Asociación cambia a medida que se está aplicando. Puede agregar uno o más de los siguientes cambios de estado a una regla de evento: • Con error • Pendiente • Success
Cambio de estado de asociación de instancia de EC2 de State Manager	El estado de un objeto de una instancia administrada única que está dirigida por los cambios de asociación. Puede agregar uno o más de los siguientes cambios de estado a una regla de evento:

Nombre del tipo de evento	Descripción de los eventos que puede agregar a una regla
	• Con error • Pendiente • Success

Referencia: expresiones cron y rate para Systems Manager

Cuando crea una asociación de State Manager o un período de mantenimiento en AWS Systems Manager, debe especificar una programación de cuándo debe llevarse a cabo el período o la asociación. Puede especificar una programación como una entrada basada en el tiempo, lo que se conoce como expresión cron, o una entrada basada en la frecuencia, lo que se conoce como expresión rate.

Información general sobre las expresiones cron y rate

La siguiente información se aplica a las expresiones cron y rate para los periodos de mantenimiento y las asociaciones.

Programaciones de una sola ejecución

Cuando crea una asociación o un periodo de mantenimiento, puede especificar una marca temporal en formato de Hora Universal Coordinada (UTC) para que se ejecute una vez a la hora especificada. Por ejemplo: `"at(2020-07-07T15:55:00)"`

Desplazamientos de la programación

Las asociaciones o los periodos de mantenimiento admiten desplazamientos de la programación solo para expresiones cron. Un desplazamiento de la programación es el número de días que se debe esperar después de la fecha y hora especificadas por una expresión cron antes de ejecutar la asociación o el periodo de mantenimiento.

Maintenance window example

En el ejemplo anterior, la expresión cron programa una ventana de mantenimiento para que se ejecute el tercer martes de cada mes a las 11:30 p.m. Sin embargo, ya que el desplazamiento de la programación es 2, la ventana de mantenimiento no se ejecutará hasta las 11:30 p.m. dos días después.

```
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

En el siguiente comando, la expresión cron programa una asociación para que se ejecute el segundo jueves de cada mes. Sin embargo, dado que la diferencia de horario es 3, la asociación no funcionará hasta el domingo siguiente, tres días después.

```
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**
>
> Para usar una compensación con una asociación, debe especificar la opción `--apply-only-at-cron-interval`. Esta opción indica al sistema que no ejecute ninguna asociación inmediatamente después de que la cree.

Si crea una asociación o un periodo de mantenimiento con una expresión cron dirigida a un día que ya ha pasado en el plazo actual, pero agrega una fecha de desplazamiento de programación que cae en el futuro, la asociación o el periodo de mantenimiento no se ejecutará en ese plazo. Entrará en vigor en el siguiente plazo. Por ejemplo, si especifica una expresión cron que habría ejecutado un periodo de mantenimiento ayer y agrega un desplazamiento de programación de dos días, el periodo de mantenimiento no se ejecutará mañana.

Campos obligatorios

Las expresiones cron para periodos de mantenimiento tienen seis campos obligatorios. Las expresiones cron para las asociaciones tienen cinco. (Actualmente, State Manager no admite la especificación de meses en expresiones cron para asociaciones). Un campo adicional, el campo

Seconds (el primero en una expresión cron) es opcional. Los campos están separados por un espacio.

Ejemplos de expresiones cron

Minutos	Horas	Día del mes	Mes	Día de la semana	Año	Significado
0	10	*	*	?	*	Ejecutar a las 10:00 h (UTC) todos los días
15	12	*	*	?	*	Ejecutar a las 12:15 (UTC) todos los días
0	18	?	*	MON-FRI	*	Ejecutar a las 18:00 (UTC) de lunes a viernes
0	8	1	*	?	*	Ejecutar a las 8:00 (UTC) cada primer día del mes

Valores admitidos

En la siguiente tabla se desglosan los valores compatibles para las entradas cron necesarias.

Valores admitidos para expresiones cron

Campo	Valores	Caracteres comodín
Minutos	0-59	, - * /
Horas	0-23	, - * /
Día del mes	1-31	, - * ? / L W
Mes (solo períodos de mantenimiento)	1-12 o bien JAN-DEC	, - * /
Día de la semana	1-7 o bien SUN-SAT	, - * ? / L #
Año	1970-2199	, - * /

> **Note**
>
> No se puede especificar un valor en los campos day-of-month y day-of-week en la misma expresión cron. Si especifica un valor en uno de los campos, utilice un ? (signo de interrogación) en el otro campo.

Comodines admitidos para expresiones cron

En la tabla siguiente se muestran los valores de comodín que admiten las expresiones cron.

> **Note**
>
> No se admiten expresiones cron que produzcan frecuencias superiores a cinco (5) minutos. La compatibilidad para especificar un valor de día de la semana y uno de día del mes no es completa. Utilice el signo de interrogación (?) en uno de estos campos.

Comodines admitidos para expresiones cron

Comodín	Descripción
,	El comodín `,` (coma) incluye los valores adicionales. En el campo Month, JAN, FEB, MAR incluiría enero, febrero y marzo.
-	El comodín `-` (guion) especifica intervalos. En el campo Day, 1-15 incluiría los días del 1 al 15 del mes especificado.
*	El comodín * (asterisco) incluye todos los valores del campo. En el campo Hours, * incluiría cada hora.
/	El comodín `/` (barra inclinada) especifica incrementos. En el campo Minutes, puede escribir 1/10 para especificar cada décimo minuto, empezando desde el primer minuto de la hora. Por lo tanto, 1/10 especifica los minutos 1, 11, 21 y 31, y así sucesivamente.
?	El comodín ? (signo de interrogación) especifica uno u otro. En el campo Day-of-month puede escribir 7 y si no se preocupó de qué día de la semana era el 7º, podría escribir ? en el campo Day-of-week.
L	El comodín `L` en los campos Day-of-month o Day-of-week especifica el último día del mes o de la semana.
W	El comodín `W` en el campo Día del mes especifica un día de la semana. En el campo Day-of-month, 3W especifica el día más cercano al tercer día de la semana del mes.

Comodín	Descripción
#	El comodín # del campo día de la semana seguido de un número comprendido entre uno y cinco especifica un día determinado del mes. 5#3 especifica el tercer jueves del mes.

Expresiones rate

Las expresiones rate tienen los siguientes dos campos obligatorios. Los campos están separados por espacios.

Campos obligatorios para expresiones rate

Campo	Valores
Valor	número positivo, como 1 o 15

Unidad	`minute`
	`minutes`
	`hour`
	`hours`
	`day`
	`days`

Si el valor es igual a 1, entonces la unidad debe ser singular. Del mismo modo, para valores mayores que 1, la unidad debe ser plural. Por ejemplo, `rate(1 hours)` y `rate(5 hour)` no son válidos, pero `rate(1 hour)` y `rate(5 hours)` son válidos.

Temas

- Expresiones cron y rate para asociaciones
- Expresiones cron y rate para los períodos de mantenimiento

Expresiones cron y rate para asociaciones

En esta sección se incluyen ejemplos de expresiones cron y rate para asociaciones de State Manager. Para crear una de estas expresiones, tenga en cuenta la siguiente información:

- Las asociaciones son compatibles con las siguientes expresiones cron: cada media, 1, 2, 4, 8 o 12 horas; cada día, cada semana, o cada día y hora concretos de la semana; un día concreto de una semana concreta del mes, o el último día x del mes a una hora concreta.
- Las asociaciones admiten las siguientes expresiones rate: intervalos de 30 minutos o más y menos de 31 días.
- Si especifica el campo opcional `Seconds`, su valor puede ser 0 (cero). Por ejemplo: `cron(0 */30 * * * ? *)`
- Para una asociación que recopila metadatos para Inventario, una herramienta de AWS Systems Manager, se recomienda utilizar una expresión de frecuencia.
- State Manager actualmente no admite especificar meses en expresiones cron para asociaciones.

Las asociaciones admiten expresiones cron que incluyen un día de la semana y el signo de número (#) para designar el día x de un mes para ejecutar una asociación. A continuación, se incluye un ejemplo en el que se ejecuta una programación cron el tercer martes de cada mes a las 23.30 h (UTC):

```
cron(30 23 ? * TUE#3 *)
```

A continuación, se incluye un ejemplo que se ejecuta el segundo jueves de cada mes a medianoche (UTC):

```
cron(0 0 ? * THU#2 *)
```

Las asociaciones también admiten el signo (L) para indicar el último día X del mes. A continuación, se incluye un ejemplo en el que se ejecuta una programación cron el último martes de cada mes a medianoche (UTC):

```
cron(0 0 ? * 3L *)
```

Para tener un mayor control sobre el momento en el que se ejecuta una asociación, por ejemplo, si desea ejecutar una asociación dos días después de la revisión del martes, puede especificar un desplazamiento. Un desplazamiento define los días que hay que esperar después del día programado para ejecutar una asociación. Por ejemplo, si especificó una programación cron de

`cron(0 0 ? * THU#2 *)`, puede especificar el número 3 en el campo Desplazamiento de programación para ejecutar la asociación cada domingo después del segundo jueves del mes.

Para utilizar los desplazamientos, debe elegir la opción Apply association only at the next specified Cron interval en la consola o debe especificar el uso del parámetro `--apply-only-at-cron-interval` desde la línea de comandos. Esta opción indica a State Manager que no ejecute ninguna asociación inmediatamente después de crearla.

En la siguiente tabla se presentan ejemplos de expresiones cron para asociaciones.

Ejemplos de cron para asociaciones

Ejemplo	Detalles
cron(0/30 * * * ? *)	Cada 30 minutos
cron(0 0/1 * * ? *)	Cada hora
cron(0 0/2 * * ? *)	Cada 2 horas
cron(0 0/4 * * ? *)	Cada 4 horas
cron(0 0/8 * * ? *)	Cada 8 horas
cron(0 0/12 * * ? *)	Cada 12 horas
cron(15 13 ? * * *)	Cada día a las 13:15
cron(15 13 ? * MON *)	Cada lunes a las 13:15
cron(30 23 ? * TUE#3 *)	El tercer martes de cada mes a las 23.30 h

A continuación mostramos algunos ejemplos de rate para asociaciones.

Ejemplos de rate para asociaciones

Ejemplo	Detalles
rate(30 minutes)	Cada 30 minutos
rate(1 hour)	Cada hora

Ejemplo	Detalles
rate(5 hours)	Cada 5 horas
rate(15 days)	Cada 15 días

Ejemplos de AWS CLI para asociaciones

Para crear asociaciones de un State Manager con el AWS CLI, incluya el parámetro `--schedule-expression` con una expresión cron o rate. En los ejemplos siguientes se utiliza la AWS CLI en un equipo Linux local.

> **Note**
>
> De forma predeterminada, cuando crea una nueva asociación, el sistema la ejecuta inmediatamente después de crearla y, a continuación, de acuerdo con la programación especificada. Especifique `--apply-only-at-cron-interval` para que no se ejecute la asociación inmediatamente después de crearla. Las expresiones rate no admiten este parámetro.

```
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
```

Expresiones cron y rate para los períodos de mantenimiento

En esta sección se incluyen ejemplos de expresiones cron y rate para períodos de mantenimiento.

A diferencia de las asociaciones de State Manager, los periodos de mantenimiento admiten todas las expresiones cron y rate. Esto incluye compatibilidad con los valores del campo de segundos.

Por ejemplo, la siguiente expresión cron de 6 campos ejecuta un periodo de mantenimiento a las 9:30 cada día.

```
cron(30 09 ? * * *)
```

Al agregar un valor al campo `Seconds`, la siguiente expresión cron de 7 campos ejecuta un periodo de mantenimiento a las 9:30:24 cada día.

```
cron(24 30 09 ? * * *)
```

En la siguiente tabla se proporcionan ejemplos de cron de 6 campos adicionales para períodos de mantenimiento.

Ejemplos de cron para los períodos de mantenimiento

Ejemplo	Detalles
cron(0 2 ? * THU#3 *)	2:00 el tercer jueves de cada mes
cron(15 10 ? * * *)	10:15 todos los días
cron(15 10 ? * MON-FRI *)	10:15 cada lunes, martes, miércoles, jueves y viernes
cron(0 2 L * ? *)	2:00 el último día de cada mes
cron(15 10 ? * 6L *)	10:15 el último viernes de cada mes

En la siguiente tabla se proporcionan ejemplos rate para los períodos de mantenimiento.

Ejemplos rate para los períodos de mantenimiento

Ejemplo	Detalles
rate(30 minutes)	Cada 30 minutos
rate(1 hour)	Cada hora
rate(5 hours)	Cada 5 horas
rate(25 days)	Cada 25 días

Ejemplos de AWS CLI para períodos de mantenimiento

Para crear un periodo de mantenimiento con la AWS CLI, incluya el parámetro `--schedule` con una expresión cron o rate o una marca temporal. En los ejemplos siguientes se utiliza la AWS CLI en un equipo Linux local.

```
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
```

Más información

[Expresiones cron] en el sitio web de Wikipedia

Referencia: ec2messages, ssmmessages y otras operaciones de la API

Si supervisa operaciones de la API, es posible que vea llamadas a las siguientes operaciones:

- `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`

Se trata de operaciones especiales utilizadas por AWS Systems Manager, tal y como se describe en el resto de este tema.

Operaciones de la API relacionadas con el agente (puntos de conexión de **`ssmmessages`** y **`ec2messages`**)

Operaciones de la API ssmmessages

Systems Manager usa el punto de conexión `ssmmessages` para los dos siguientes tipos de operaciones de API:

- Operaciones desde Systems Manager Agent (SSM Agent) al servicio Systems Manager en la nube.
- Operaciones desde SSM Agent hasta Session Manager, una herramienta de AWS Systems Manager, en la nube. Este punto de enlace es necesario para crear y eliminar los canales de sesión con el servicio Session Manager en la nube. Además, si se permite la conectividad, SSM Agent recibe los documentos de `Command` a través de Amazon Message Gateway Service. Si no se permite la conectividad, SSM Agent recibe los documentos de `Command` a través de Amazon Message Delivery Service. Para obtener información, consulte Acciones, recursos y claves de condición de Amazon Message Gateway Service.
- Operaciones desde Run Command.

Operaciones de la API ec2messages

Las operaciones de API `ec2messages:*` se realizan en el punto de conexión de Amazon Message Delivery Service. Systems Manager utiliza este punto de conexión para operaciones de la API desde Systems Manager Agent (SSM Agent) al servicio Systems Manager en la nube.

> **Important**
>
> Las operaciones de la API `ec2messages:*` solo se admiten en las Regiones de AWS que se lanzaron antes de 2024. En las regiones lanzadas a partir de 2024, solo se admiten las operaciones de la API `ssmmessages:*`.

Prioridad de conexión del punto de conexión

A partir de la versión 3.3.40.0 de SSM Agent, Systems Manager comenzó a utilizar el punto de conexión `ssmmessages:*` (Amazon Message Gateway Service) siempre que estaba disponible en lugar del punto de conexión de `ec2messages:*` (Amazon Message Delivery Service).

Si proporciona acceso a `ssmmessages:*` en sus políticas de permisos AWS Identity and Access Management (de IAM), SSM Agent se conectara al punto de conexión `ssmmessages:*`, incluso si su perfil de instancia de IAM está configurado para permitir ambos puntos de conexión. Esto incluye políticas para los perfiles de instancia de IAM y los roles de servicio de IAM que haya creado usted mismo, y para los perfiles de instancia de IAM creados mediante la configuración de la administración de hosts de Quick Setup y la configuración predeterminada de la administración de hosts.

Si ha proporcionado permisos para ambos puntos de conexión y ha supervisado las operaciones de la API mediante, por ejemplo, CloudWatch Metrics, no verá ninguna llamada a `ec2messages:*`.

En el caso de las Regiones de AWS lanzadas antes de 2024, puede eliminar los permisos `ec2messages:*` de sus políticas de forma segura en este momento.

Conmutación por error de punto de conexión

Únicamente en el caso de las Regiones de AWS lanzadas antes de 2024: si su perfil de instancia de IAM no proporciona permisos para `ssmmessages:*` en el momento en el que el agente se inicia, sino solo `ec2messages:*`. SSM Agent se conecta al punto de conexión de `ec2messages:*`. Si tiene ambos `ssmmessages:*` y `ec2messages:*` en ese momento se inicia SSM Agent, pero los elimina `ssmmessages:*` después de que se inicie el agente, SSM Agent pronto cambiará la conexión al punto de conexión `ec2messages:*`. En el caso de las regiones lanzadas a partir de 2024, solo se admite el punto de conexión de `ssmmessages:*`.

Para obtener más información sobe los puntos de conexión `ssmmessages` y `ec2messages:*`, consulte los siguientes temas en la Referencia de autorizaciones de servicio de AWS.

- Acciones, recursos y claves de condiciones de Amazon Message Gateway Service (`ssmmessages`).
- Acciones, recursos y claves de condiciones de Amazon Message Delivery Service (`ec2messages:*`)

Operaciones de la API relacionadas con las instancias del espacio de nombres **ssm:***

`DescribeDocumentParameters`

Systems Manager ejecuta esta operación de la API para representar nodos específicos en la consola de Amazon EC2. Los resultados de la operación `DescribeDocumentParameters` se muestran en el nodo Documentos.

`DescribeInstanceProperties`

Systems Manager ejecuta estas operaciones de la API para representar nodos específicos en la consola de Amazon EC2. Los resultados de la operación `DescribeInstanceProperties` se muestran en el nodo Fleet Manager.

`GetCalendar`

Systems Manager ejecuta esta operación de la API para representar documentos de tipo Change Calendar en la consola de Change Calendar.

`GetManifest`

SSM Agent ejecuta esta operación de la API para determinar los requisitos del sistema para instalar o actualizar una versión específica de un paquete de AWS Systems Manager Distributor. Se trata de una operación de API antigua y no estará disponible en Regiones de AWS si se lanzó después de 2017.

`ListInstanceAssociations`

SSM Agent ejecuta esta operación de la API para ver si hay disponible una asociación de State Manager nueva. Esta API es necesaria para que State Manager funcione.

`PutCalendar`

Systems Manager ejecuta esta operación de la API para actualizar los documentos de tipo Change Calendar en la consola de Change Calendar.

`PutConfigurePackageResult`

SSM Agent ejecuta esta operación de la API para publicar las métricas de errores de instalación y latencia de los paquetes de distribuidores públicos en la cuenta del propietario del paquete.

`RegisterManagedInstance`

SSM Agent ejecuta esta operación de la API en los siguientes escenarios:

- Para registrar un servidor o máquina virtual (VM) en las instalaciones con Systems Manager como instancia administrada mediante un código e ID de activación.
- Para registrar las credenciales de AWS IoT Greengrass Version 2.

Esta operación también la llaman las instancias de Amazon EC2 que ejecutan la versión SSM Agent 3.1.x o posterior.

`RequestManagedInstanceRoleToken`

SSM Agent ejecuta esta operación de la API para recuperar credenciales temporales para acceder al nodo administrado.

`UpdateInstanceAssociationStatus`

SSM Agent ejecuta esta operación de la API para actualizar una asociación. Esta operación de la API es necesaria para que State Manager, una herramienta de AWS Systems Manager, funcione.

`UpdateInstanceInformation`

SSM Agent llama al servicio de Systems Manager en la nube cada 5 minutos para ofrecer información de latido. Esta llamada es necesaria para mantener un latido con el agente de modo que el servicio sepa que el agente funciona según lo previsto.

`UpdateManagedInstancePublicKey`

SSM Agent ejecuta esta operación de la API para proporcionar la clave pública después de rotar el par de claves en el nodo administrado. La clave pública se usa para autenticar las solicitudes, firmadas con la clave privada, para obtener credenciales temporales de Systems Manager.

ssm:* espacio de nombres otras operaciones de la API

`ExecuteApi`

Los administradores delegados de OpsItems que administran OpsCenter en OpsItems necesitan acceso a esta acción de la API para poder ver detalles de recursos relacionados sobre Cuentas de AWS en varias . En concreto, esta API concede a un administrador delegado permiso para ver los siguientes detalles de OpsItem en la AWS Management Console: la descripción del OpsItem, las etiquetas, la plantilla de AWS CloudFormation, los cambios de AWS Config, las alarmas de los Registros de CloudWatch y los eventos de AWS CloudTrail. Para obtener más información sobre cómo utilizar OpsItems en las diferentes cuentas, consulte (Opcional) Configuración manual de OpsCenter para administrar OpsItems de forma centralizada entre cuentas. Para obtener más

información sobre los detalles de los recursos relacionados para OpsItems, consulte Adición de recursos relacionados a un OpsItem.

Referencia: Formatos de cadenas de fecha y hora para Systems Manager

Las operaciones de la API de AWS Systems Manager aceptan filtros para limitar el número de resultados regresados por una solicitud. Algunas de estas operaciones de la API aceptan filtros que requieren una cadena con formato para representar una fecha y hora específicas. Por ejemplo, la operación `DescribeSessions` de la API acepta las claves `InvokedAfter` y `InvokedBefore` como algunos de los valores válidos para un objeto `SessionFilter`. Otro ejemplo es la operación `DescribeAutomationExecutions` de la API, que acepta las claves `StartTimeBefore` y `StartTimeAfter` como algunos de los valores válidos para un objeto `AutomationExecutionFilter`. Los valores que proporcione para estas claves al filtrar sus solicitudes deben coincidir con el estándar ISO 8601. Para obtener información sobre ISO 8601, consulte ISO 8601.

Estas cadenas con formato de fecha y hora no se limitan a los filtros. También hay operaciones de la API que requieren una cadena con formato ISO 8601 para representar una fecha y hora específicas cuando se proporcione un valor para un parámetro de solicitud. Por ejemplo, el parámetro de solicitud `AtTime` para la operación `GetCalendarState`. Estas cadenas son difíciles de crear. Utilice los ejemplos de este tema para crear cadenas con formato de fecha y hora para utilizarlas con operaciones de API de Systems Manager.

Dar formato a cadenas de fecha y hora para Systems Manager

A continuación, se muestra un ejemplo de una cadena con formato de fecha y hora ISO 8601.

```
2024-05-08T15:16:43Z
```

Esto representa el 8 de mayo de 2024 a las 15:16 en Hora Universal Coordinada (UTC). La parte de fecha de calendario de la cadena se representa mediante un año de cuatro dígitos, un mes de dos dígitos y un día de dos dígitos separados por guiones. Esto se puede representar en el siguiente formato.

```
YYYY-MM-DD
```

La parte de hora de la cadena comienza con la letra «T» como delimitador y, a continuación, se representa con la hora de dos dígitos, los minutos de dos dígitos y los segundos de dos dígitos separados por dos puntos. Esto se puede representar en el siguiente formato.

```
hh:mm:ss
```

La parte de hora de la cadena termina con la letra «Z», que denota el estándar UTC.

Crear cadenas de fecha y hora personalizadas para Systems Manager

Puede crear cadenas de fecha y hora personalizadas desde su equipo local utilizando su herramienta de línea de comandos preferida. La sintaxis que utilice para crear una cadena con formato de fecha y hora ISO 8601 varía en función del sistema operativo del equipo local. A continuación, se muestran ejemplos de cómo puede utilizar `date` de los coreutils de GNU en Linux o PowerShell en Windows para crear una cadena de fecha y hora con formato ISO 8601.

coreutils

```
date '+%Y-%m-%dT%H:%M:%SZ'
```

PowerShell

```
(Get-Date).ToString("yyyy-MM-ddTH:mm:ssZ")
```

Cuando se trabaja con operaciones de la API de Systems Manager, es posible que necesite crear cadenas de fecha y hora históricas para generar informes o solucionar problemas. A continuación, se muestran ejemplos de cómo puede crear y utilizar cadenas de fecha y hora con formato ISO 8601 históricas personalizadas para la Herramientas de AWS para PowerShell y AWS Command Line Interface (AWS CLI).

AWS CLI

- Recuperar la última semana del historial de comandos de un documento de 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
```

- Recuperar la última semana del historial de ejecución de automatización.

```
lastWeekStamp=$(date '+%Y-%m-%dT%H:%M:%SZ' -d '7 days ago')

aws ssm describe-automation-executions \
    --filters Key=StartTimeAfter,Values=$lastWeekStamp
```

- Recuperar el último mes del historial de sesiones.

```
lastWeekStamp=$(date '+%Y-%m-%dT%H:%M:%SZ' -d '30 days ago')

aws ssm describe-sessions \
    --state History \
    --filters key=InvokedAfter,value=$lastWeekStamp
```

Herramientas de AWS para PowerShell

- Recuperar la última semana del historial de comandos de un documento de 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
```

- Recuperar la última semana del historial de ejecución de automatización.

```
$lastWeekStamp = (Get-Date).AddDays(-7).ToString("yyyy-MM-ddTH:mm:ssZ")

Get-SSMAutomationExecutionList `
    -Filters @{Key="StartTimeAfter";Values=$lastWeekStamp}
```

- Recuperar el último mes del historial de sesiones.

```
$lastWeekStamp = (Get-Date).AddDays(-30).ToString("yyyy-MM-ddTH:mm:ssZ")

Get-SSMSession `
    -State History `
    -Filters @{Key="InvokedAfter";Value=$lastWeekStamp}
```

Casos de uso y prácticas recomendadas

En este tema se enumeran los casos de uso comunes y las prácticas recomendadas de las herramientas de AWS Systems Manager. Si están disponibles, este tema también incluye enlaces a publicaciones de blogs relevantes y documentación técnica.

> **Note**
>
> El título de cada sección que se indica a continuación es un enlace a la sección correspondiente activo en la documentación técnica.

Automatización

- Cree manuales de procedimientos de Automation de autoservicio para la infraestructura.
- Utilice Automatización, una herramienta de AWS Systems Manager, para simplificar la creación de Amazon Machine Images (AMIs) desde AWS Marketplace o AMIs personalizadas, mediante documentos públicos de Systems Manager (documentos de SSM) o mediante la creación de sus propios flujos de trabajo.
- Cree y mantenga AMIs mediante los manuales de procedimientos de Automation `AWS-UpdateLinuxAmi` y `AWS-UpdateWindowsAmi` o mediante manuales de procedimientos de Automation personalizados que cree.

Conformidad

- Como práctica recomendada de seguridad, sugerimos que actualice el rol AWS Identity and Access Management (IAM) que utilizan sus nodos administrados para restringir la capacidad del nodo de utilizar la acción de la API PutComplianceItems. Esta acción de API registra un tipo de conformidad junto con sus detalles en un recurso designado, como una instancia de Amazon EC2 o un nodo administrado. Para obtener más información, consulte Cómo configurar permisos para Compliance.

Inventario

- Utilice Inventario, una herramienta de AWS Systems Manager, con AWS Config para auditar las configuraciones de aplicaciones a lo largo del tiempo.

Maintenance Windows

- Defina una programación para realizar las acciones potencialmente disruptivas en los nodos, por ejemplo, la aplicación de revisiones en el sistema operativo (SO), las actualizaciones de controladores o las instalaciones de software.
- Para obtener información acerca de las diferencias entre State Manager y Maintenance Windows, herramientas de AWS Systems Manager, consulte Elección entre State Manager y Maintenance Windows.

Parameter Store

- Utilice Parameter Store, una herramienta de AWS Systems Manager, para administrar de forma centralizada los valores de configuración globales.
- Cómo AWS Systems ManagerParameter Store utiliza AWS KMS.
- Referencia a los secretos de AWS Secrets Manager desde los parámetros de Parameter Store.

Patch Manager

- Utilice Patch Manager, una herramienta de AWS Systems Manager, para aplicar revisiones a escala y aumentar la visibilidad de la conformidad de la flota en todos los nodos.
- Integre Patch Manager en AWS Security Hub para recibir alertas cuando los nodos de su flota salgan de conformidad y monitoree el estado de aplicación de revisiones de sus flotas desde el punto de vista de la seguridad. El uso de Security Hub conlleva un cargo. Para obtener más información, consulte Precios.
- Utilice solo un método a la vez para analizar los nodos administrados para comprobar la conformidad de las revisiones a fin de evitar sobrescribir involuntariamente los datos de conformidad.

Run Command

- Administrar instancias a escala sin acceso SSH con Run Command de EC2.
- Audite todas las llamadas a la API hechas por Run Command, una herramienta de AWS Systems Manager, o en su nombre mediante AWS CloudTrail.
- Cuando ejecute un comando con Run Command, no incluya información confidencial como texto sin formato, por ejemplo, contraseñas, datos de configuración u otros secretos. Toda la actividad

de la API de Systems Manager de la cuenta se registra en un bucket de S3, para registros de AWS CloudTrail. Esto significa que cualquier usuario con acceso al bucket de S3 puede ver los valores en texto sin formato de esos secretos. Por este motivo, le recomendamos crear y utilizar parámetros `SecureString` para cifrar la información confidencial que utiliza en las operaciones de Systems Manager.

Para obtener más información, consulte Restringir el acceso a los parámetros de Parameter Store mediante políticas de IAM.

> **Note**
>
> De forma predeterminada, los archivos de registro que envía CloudTrail a su bucket se cifran mediante el sistema de Amazon de cifrado del lado del servidor con claves de cifrado administradas mediante Amazon S3 (SSE-S3). Para proporcionar una capa de seguridad que pueda administrarse directamente, en su lugar puede utilizar el cifrado del lado del servidor con claves de AWS KMS administradas (SSE-KMS) para los archivos de registros de CloudTrail.
> Para obtener más información, consulte Cifrado de archivos de registros de CloudTrail con claves administradas por AWS KMS (SSE-KMS) en la Guía del usuario de AWS CloudTrail.

- Usar la característica de control de velocidad y destinos en Run Command para efectuar una operación de comando por fases.
- Utilice permisos de acceso detallados para Run Command (y todas las herramientas de Systems Manager) mediante las políticas de AWS Identity and Access Management (IAM).

Session Manager

- Registre la actividad de la sesión en la Cuenta de AWS con AWS CloudTrail.
- Registre los datos de la sesión en la Cuenta de AWS con Amazon CloudWatch Logs o Amazon S3.
- Control del acceso de las sesiones de usuario a las instancias.
- Restrinja el acceso a los comandos de una sesión.
- Desactive o active los permisos administrativos de la cuenta ssm-user.

State Manager

- Actualice el SSM Agent al menos una vez al mes con el documento `AWS-UpdateSSMAgent` preconfigurado.
- (Windows) Cargue el módulo de PowerShell o DSC en Amazon Simple Storage Service (Amazon S3) y utilice `AWS-InstallPowerShellModule`.
- Usar etiquetas para crear grupos de aplicaciones para los nodos. y, a continuación, dirigirse a los nodos usando el parámetro `Targets` en lugar de especificar el ID de cada nodo.
- Solucione automáticamente los resultados generados por Amazon Inspector con Systems Manager.
- Utilice un repositorio de configuración centralizado para todos los documentos de SSM y comparta los documentos en la organización.
- Para obtener información acerca de las diferencias entre State Manager y Maintenance Windows, consulte Elección entre State Manager y Maintenance Windows.

Nodos administrados

- Systems Manager requiere referencias de horarios precisos para realizar sus operaciones. Si la fecha y la hora del nodo no se establecen correctamente, podrían no coincidir con la fecha de la firma de las solicitudes de la API. Esto podría producir errores o funcionalidad incompleta. Por ejemplo, los nodos con una configuración de hora incorrecta no se incluirán en las listas de nodos administrados.

  Para obtener información sobre cómo configurar la hora en los nodos, consulte Configuración de la hora en una instancia de Amazon EC2.
- En los nodos gestionados por Linux, compruebe la firma de SSM Agent.

Más información

- Prácticas recomendadas de seguridad para Systems Manager

Eliminación de recursos y artefactos de Systems Manager

Como práctica recomendada, se recomienda que elimine los recursos y artefactos de Systems Manager si ya no necesita ver los datos de esos recursos o utilizar los artefactos. En la siguiente

tabla se muestran todas las herramientas o artefactos de Systems Manager y un enlace para obtener más información sobre la eliminación de los recursos o artefactos creados por Systems Manager.

Capacidad o artefacto	Detalles
Application Manager	No puede eliminar una aplicación en Applicati on Manager, pero puede eliminar una aplicació n del servicio mediante la eliminación del elemento subyacente etiquetas, Resource Groups o Pilas de AWS CloudFormation.
Automatización	Si crea recursos de AWS mediante Systems Manager Automation, elimine manualmente esos recursos mediante la correspondiente AWS Management Console. Si ha creado un runbook personalizado, puede eliminar el documento de SSM subyacente. Para obtener más información, consulte Eliminación de documentos de SSM personalizados.
Change Calendar	Puede eliminar un calendario de cambios y un evento de calendario de cambios. Para obtener más información, consulte Eliminación de un calendario de cambios y Eliminación de un evento de Change Calendar.
Change Manager	Puede eliminar una plantilla de cambio. Para obtener más información, consulte Eliminación de plantillas de cambio.
Conformidad	Systems Manager Compliance muestra automáticamente los datos de conformidad sobre el parcheo de Patch Manager y las asociaciones de State Manager. No es posible eliminar estos datos. Si ha configurado una sincronización de datos de recursos para centralizar los datos de cumplimiento en un bucket de S3, puede eliminar la sincroniz

Capacidad o artefacto	Detalles
	ación. Para obtener más información, consulte Eliminación de una sincronización de datos de recursos para Compliance.
Distributor	Puede eliminar paquetes en Distributor. Para obtener más información, consulte Eliminar un paquete de Distributor.
Explorer	Puede desconectarse de las fuentes desde las que Explorer recopila OpsData. Para obtener más información, consulte Edición de orígenes de datos de Systems Manager Explorer. También puede eliminar una sincronización de datos de recursos utilizada por Explorer para agregar OpsData y OpsItems desde múltiples cuentas y de Regiones de AWS en un único bucket de Amazon Simple Storage Service (Amazon S3). Para obtener más información, consulte Eliminación de una sincronización de datos de recursos de Systems Manager Explorer. Para obtener información acerca de la eliminación de un bucket de S3, consulte Eliminación de un bucket en la Guía para desarrolladores de Amazon Simple Email Service.
Fleet Manager	No es posible eliminar un nodo administr ado mediante Fleet Manager. Utilice Amazon Elastic Compute Cloud (Amazon EC2). Para obtener más información, consulte Terminar la instancia (Linux) y Terminar la instancia (Windows).

Capacidad o artefacto	Detalles
Inventario	Puede detener la recopilación de datos de inventario mediante la eliminación de las asociaciones de State Manager que definen la programación y los recursos a partir de los cuales recopilar metadatos. Para obtener más información, consulte Detención de la recopilación de datos y eliminación de datos de inventario. Si ya no desea utilizar AWS Systems Manager Inventory para ver metadatos sobre los recursos de AWS, se recomienda que elimine las sincronizaciones de datos de recursos que se utilizan para la recopilación de datos de inventario. Para obtener más información, consulte Eliminación de una sincronización de datos de recursos de inventario.
Maintenance Windows	Puede eliminar una ventana de mantenimiento, un destino de ventana de mantenimiento y una tarea de ventana de mantenimiento. Para obtener más información, consulte Actualizar o eliminar recursos de un periodo de mantenimiento mediante la consola.
OpsCenter	Puede eliminar un OpsItem individual llamando a la operación de la API Delete OpsItem mediante la AWS Command Line Interface o el AWS SDK. No se puede eliminar un OpsItem en la AWS Management Console. Para obtener más información, consulte Elimine OpsItems.
Parameter Store	Puede eliminar un parámetro que haya creado. Para obtener más información, consulte Eliminación de parámetros de Parameter Store.

Capacidad o artefacto	Detalles
Patch Manager	Puede eliminar una base de referencia de parches personalizada. Para obtener más información, consulte Actualización o eliminación de una línea de base de revisiones personalizada.
Configuración Rápida	Puede eliminar las asociaciones creadas por la Configuración Rápida. Las asociaciones se almacenan y procesan mediante State Manager. Para obtener más información, consulte Eliminación de una asociación.
Run Command	Una vez finalizado el procesamiento de un comando, la información al respecto se almacena en la pestaña Historial de comandos. No es posible eliminar la información de la pestaña Historial de comandos.
Automatización	Una vez finalizado el procesamiento de una automatización, la información al respecto se almacena en la pestaña Ejecuciones. No es posible eliminar la información en la pestaña Ejecuciones.
Rol vinculado a servicios	Systems Manager crea automáticamente roles vinculados a servicios para algunas herramientas. Puede eliminar estos roles. Para obtener más información, consulte Eliminación del rol vinculado al servicio `AWSServiceRoleForAmazonSSM` de Systems Manager.
Session Manager	Session Manager no conserva datos sobre los recursos después de terminar una sesión. Para terminar una sesión, consulte Finalización de una sesión.

Capacidad o artefacto	Detalles
SSM Agent	Puede desinstalar manualmente SSM Agent de los nodos. Para obtener más información, consulte los siguientes temas. • Linux: Instalación y desinstalación manual de SSM Agent en instancias de EC2 para Linux • macOS: Instalación y desinstalación manual de SSM Agent en instancias de EC2 para macOS • Windows Server: Abra el Panel de control y elija Add/remove programs.
State Manager	Puede eliminar una asociación. Para obtener más información, consulte Eliminación de una asociación.
Servicio de documentos de Systems Manager	No es posible eliminar manuales de procedimiento proporcionados por AWS o AWS Support, pero puede eliminar manuales de procedimiento personalizados. Para obtener más información, consulte Eliminación de documentos de SSM personalizados.

Elección entre State Manager y Maintenance Windows

State Manager y Maintenance Windows, herramientas de AWS Systems Manager, pueden llevar a cabo algunos tipos similares de actualizaciones en los nodos administrados. La opción que elija dependerá de si necesita automatizar la conformidad del sistema o realizar tareas de alta prioridad y urgencia durante los periodos que especifique.

State Manager y Maintenance Windows: casos de uso clave

State Manager, una herramienta de AWS Systems Manager, establece y mantiene la configuración de estado de destino para los nodos administrados y recursos de AWS en su Cuenta de AWS. Puede definir combinaciones de configuraciones y destinos como objetos de asociación. State

Manager es la herramienta recomendada si desea mantener todos los nodos administrados de su cuenta en un estado coherente, utilizar Amazon EC2 Auto Scaling para generar nuevos nodos o tener requisitos de generación de informes de conformidad estrictos para los nodos administrados de su cuenta.

Los principales casos de uso de State Manager son los siguientes:

- Escenarios de Auto Scaling: State Manager puede monitorear todos los nodos nuevos lanzados en una cuenta, ya sea de forma manual o a través de grupos de Auto Scaling. Si hay asociaciones en la cuenta que se dirijan a ese nuevo nodo (a través de etiquetas o todos los nodos), esa asociación en concreto se aplicará de forma automática al nuevo nodo.
- Informes de cumplimiento: State Manager puede generar informes de cumplimiento de los estados requeridos para los recursos de su cuenta.
- Compatibilidad con todos los nodos: State Manager puede dirigirse a todos los nodos de una cuenta determinada.

Un periodo de mantenimiento realiza una o varias acciones en los recursos de AWS en un periodo determinado. Puede definir un único periodo de mantenimiento con las horas de inicio y finalización. Puede especificar varias tareas para ejecutar en el periodo de mantenimiento. Utilice Maintenance Windows, una herramienta de AWS Systems Manager, si las operaciones de alta prioridad incluyen la aplicación de revisiones a los nodos administrados, la ejecución de varios tipos de tareas en los nodos durante un periodo de actualización o el control de cuándo se pueden ejecutar las operaciones de actualización en los nodos.

Los principales casos de uso de Maintenance Windows son los siguientes:

- Ejecución de varios documentos: los periodos de mantenimiento pueden ejecutar varias tareas. Cada tarea puede utilizar un tipo de documento diferente. Como resultado, puede crear flujos de trabajo complejos mediante diferentes tareas en un único periodo de mantenimiento.
- Aplicación de revisiones: un periodo de mantenimiento puede proporcionar compatibilidad con la aplicación de revisiones para todos los nodos administrados en una única región con una etiqueta o con un grupo de recursos específicos. La aplicación de revisiones suele implicar la eliminación de nodos (por ejemplo, eliminar nodos de un balanceador de carga), de la aplicación de revisiones y del posterior procesamiento (volver a poner los nodos en producción). Por esta razón, la aplicación de revisiones se puede lograr como una serie de tareas en un periodo de revisión determinado.

> **Note**
>
> Cuando se utiliza un periodo de mantenimiento, la operación de aplicación de revisiones se limita a una sola región en una sola cuenta. Si utiliza una política de revisiones creada en Quick Setup, una herramienta de Systems Manager, puede configurar la aplicación de revisiones para algunas o todas las cuentas y las regiones de una organización creada en AWS Organizations. Para obtener más información, consulte Configuraciones de políticas de revisiones en Quick Setup.

- Acciones de periodo: los periodos de mantenimiento pueden hacer que uno o más conjuntos de acciones comiencen dentro de un periodo específico. Los periodos de mantenimiento no se iniciarán fuera de ese periodo. Las acciones ya iniciadas continúan hasta que finalizan, incluso si esto ocurre fuera del periodo.

En la siguiente tabla se comparan las características principales de State Manager y Maintenance Windows.

Característica	State Manager	Maintenance Windows
Integración de AWS CloudFormation	Las plantillas de AWS CloudFormation admiten las asociaciones de State Manager.	Las plantillas de AWS CloudFormation admiten periodos de mantenimiento, destinos de periodos y tareas de periodos.
Conformidad	Todas las asociaciones de State Manager informan el cumplimiento con respecto al estado requerido del recurso de destino. Puede utilizar el panel de control de Compliance para agregar y ver la conformidad informada.	No se usa.
Integración de Configuration Management	State Manager es compatibl e con soluciones de estado	No se usa.

Característica	State Manager	Maintenance Windows
	específicas externas, como la configuración de estado de destino (DSC) de Microsoft PowerShell, las guías de Ansible y las recetas de Chef. Puede usar las asociaciones de State Manager para probar que las soluciones de Configuration Management funcionan y aplicar a los nodos sus cambios de configuración cuando esté listo.	
Documentos	Las configuraciones de State Manager se pueden definir como documentos de política (para recopilar información de inventario), manuales de procedimientos de Automation, para recursos de AWS, tales como buckets de Amazon Simple Storage Service (Amazon S3), o documentos de Systems Manager Command (documentos de SSM), para nodos administrados.	Las configuraciones de Maintenance Windows se pueden definir como documentos de Automatización (acciones de varios pasos con flujos de trabajo de aprobación opcionales) o documentos de SSM (estado requerido para nodos administrados).

Característica	State Manager	Maintenance Windows
Supervisión	State Manager monitorea los cambios en la configura ción, la asociación o el estado de un nodo (por ejemplo, nuevos nodos conectado s en línea). Cuando State Manager detecta los cambios, la asociación determinada se vuelve a aplicar a los nodos a los que originalmente se dirigieron con esa asociación.	No se usa.
Prioridades de las tareas	No se usa.	A las tareas de un periodo de mantenimiento se les puede asignar una prioridad . Todas las tareas con la misma prioridad se ejecutan en paralelo. Las tareas con prioridades más bajas se ejecutan después de que las tareas con prioridades más altas alcancen un estado final. No hay forma de ejecutar tareas de forma condicion al. Una vez que una tarea de mayor prioridad alcanza su estado final, se ejecuta la siguiente tarea de prioridad , independientemente del estado de la tarea anterior.

Característica	State Manager	Maintenance Windows
Controles de seguridad	State Manager admite dos controles de seguridad cuando se implementan configuraciones en una flota grande. Puede utilizar la simultaneidad máxima para definir cuántos nodos o recursos simultáneos deben tener aplicada la configuración. Puede definir una tasa de error máxima que se puede utilizar para pausar la asociación de State Manager si se produce un cierto número o porcentaje de errores en toda la flota.	El periodo de mantenimiento admite dos controles de seguridad cuando se implementan configuraciones en una flota grande. Puede utilizar la simultaneidad máxima para definir cuántos nodos o recursos simultáneos deben tener aplicada la configuración. Puede definir una tasa de error máxima que se puede utilizar para pausar las acciones en un periodo de mantenimiento si se produce un cierto número o porcentaje de errores en toda la flota.

Característica	State Manager	Maintenance Windows
Programación	Puede ejecutar las asociaciones de State Manager bajo demanda, en un intervalo y a una velocidad determinados o después de que se crean. Esto es útil si desea mantener el estado requerido de los recursos de manera coherente y oportuna. **Important** Expresiones cron para las asociaciones State Manager no admiten el campo months (meses), como `03` o `MAR` para el mes de marzo. Si necesita actualizaciones de configuración mensuales o trimestrales, un periodo de mantenimiento puede satisfacer mejor sus necesidades. Para obtener más información, consulte Referencia: expresiones cron y rate para Systems Manager.	Los periodos de mantenimiento admiten varias opciones de programación, incluidas las expresiones `at` (por ejemplo, `"at(2021-07-07T13:15:30)"`), las expresiones cron y rate, las cron con desplazamientos, la hora de inicio y finalización para ejecutar los periodos de mantenimiento y la hora límite para especificar cuándo detener la programación dentro de un periodo determinado.

Característica	State Manager	Maintenance Windows
Indicación del destino	Las asociaciones de State Manager pueden dirigirse a uno o más nodos mediante el ID de nodo, la etiqueta o el grupo de recursos. State Manager puede dirigirse a todos los nodos administr ados dentro de una cuenta determinada.	Los periodos de mantenimi ento pueden dirigirse a uno o más nodos mediante identific adores de nodos, etiquetas o grupos de recursos.

Característica	State Manager	Maintenance Windows
Tareas dentro de los periodos de mantenimiento	No se usa.	Los periodos de mantenimiento pueden admitir una o más tareas en las que cada una se dirige a un manual de procedimientos de Automation específico o a una acción de documento de Command. Todas las tareas de un periodo de mantenimiento se ejecutan en paralelo, a menos que se establezcan prioridades diferentes para cada tarea. En general, los periodos de mantenimiento permiten ejecutar cuatro tipos de tareas: • Comandos Run Command de AWS Systems Manager • Flujos de trabajo de AWS Systems Manager Automation • Funciones de AWS Lambda • Tareas de AWS Step Functions

Información relacionada

Los recursos relacionados siguientes pueden serle de ayuda cuando trabaje con este servicio.

Precios

Se cobra una tarifa por el uso de algunas herramientas de Systems Manager. Para más información, consulte Precios de AWS Systems Manager.

Biblioteca de documentación de AWS Systems Manager

Documentación de AWS Systems Manager: acceda a toda la documentación de usuario de Systems Manager, incluido AWS AppConfig, el administrador de incidentes y AWS Systems Manager para SAP.

AWS re:Post

AWS re:Post: servicio administrado por AWS de preguntas y respuestas (P y R) que ofrece respuestas de varios orígenes y revisadas por expertos a sus preguntas técnicas.

Blog y pódcast de AWS

Lea las publicaciones en el blog sobre Systems Manager en la Categoría de herramientas de administración de AWS y otras publicaciones etiquetadas con #Systems Manager.

Service Quotas

Revise las Service Quotas de Systems Manager en la Referencia general de Amazon Web Services. A menos que se indique lo contrario, cada cuota se aplica a una sola región en una Cuenta de AWS.

Referencia de autorizaciones de servicio para Systems Manager

En la Referencia de autorizaciones de servicio de AWS, consulte la información sobre las claves de contexto de las acciones, los recursos y las condiciones que puede utilizar en las políticas de AWS Identity and Access Management (IAM) de Systems Manager.

Acuerdo de nivel de servicios de AWS Systems Manager

El Acuerdo de nivel de servicio (SLA) de AWS Systems Manager es una política que rige el uso de Systems Manager y se aplica por separado a cada Cuenta de AWS mediante Systems Manager.

Recursos generales de AWS

Los recursos generales siguientes pueden serle de ayuda cuando trabaje con AWS.

- Clases y talleres: enlaces a cursos basados en roles y especializados, además de laboratorios autoguiados para ayudarlo a desarrollar sus conocimientos sobre AWS y obtener experiencia práctica.
- Centro para desarrolladores de AWS: explore los tutoriales, descargue herramientas y obtenga información sobre los eventos para desarrolladores de AWS.
- Herramientas para desarrolladores de AWS: enlaces a herramientas para desarrolladores, SDK, conjuntos de herramientas de IDE y herramientas de línea de comandos para desarrollar y administrar aplicaciones de AWS.
- Centro de recursos de introducción: aprenda a configurar su Cuenta de AWS, únase a la comunidad de AWS y lance su primera aplicación.
- Tutoriales prácticos: comience con tutoriales paso a paso antes de lanzar su primera aplicación en AWS.
- Documentos técnicos de AWS: enlaces a una lista completa de documentos técnicos de AWS que tratan una gran variedad de temas técnicos, como arquitecturas, seguridad y economía de la nube, escritos por arquitectos de soluciones de AWS o expertos técnicos.
- AWS SupportCentro de : punto para crear y administrar los casos de AWS Support. También incluye enlaces a otros recursos útiles como foros, preguntas técnicas frecuentes, estado de los servicios y de AWS Trusted Advisor.
- Soporte: la página web principal para obtener información acerca de Soporte, un canal de soporte individualizado y de respuesta rápida que le ayudará a crear y ejecutar aplicaciones en la nube.
- Contacta con nosotros – Un punto central de contacto para las consultas relacionadas con la facturación AWS, cuentas, eventos, abuso y demás problemas.
- AWSTérminos del sitio de : información detallada sobre nuestros derechos de autor y marca comercial, su cuenta, licencia y acceso al sitio, entre otros temas.

Historial del documento

En la siguiente tabla se describen los cambios importantes que se han realizado en la documentación desde la versión más reciente de AWS Systems Manager. Para obtener notificaciones sobre las actualizaciones de esta documentación, puede suscribirse a una fuente RSS.

Cambio	Descripción	Fecha
Nueva guía de solución de problemas para los errores de dependencia del bloqueo de paquetes de Zypper en los nodos administrados de SLES	Se agregó una guía completa de solución de problemas para resolver los errores de dependencia del bloqueo de paquetes de Zypper que se producen durante las operaciones `Install` de Patch Manager en los nodos administrados de SUSE Linux Enterprise Server. La nueva sección Problema: errores de dependencia del bloqueo de paquetes de Zypper en los nodos administrados de SLES proporciona información detallada sobre cómo identific ar, diagnosticar y resolver los conflictos de bloqueo de paquetes que impiden la correcta instalación del parche.	28 de julio de 2025
Guía mejorada de solución de problemas para las asociacio nes de inventario omitido	Se actualizó la sección de solución de problemas omitidos para proporcio nar una guía completa sobre la gestión prioritar ia de las asociaciones de	24 de julio de 2025

inventario. La documenta ción mejorada ahora incluye información detallada sobre las asociaciones de inventari o de Quick Setup, escenarios de segmentación explícitos y procedimientos de resolución paso a paso para administrar las asociaciones de inventario conflictivas.

Entre las características clave se incluyen las siguientes:

- Explicación del orden de prioridad de las asociacio nes de inventario (Quick Setup, asociaciones explícitas y globales)
- Escenarios comunes con ejemplos específicos y resultados esperados
- Pasos de resolución detallados para eliminar configuraciones de Quick Setup cuando se utilizan asociaciones de inventario personalizadas
- Notas importantes sobre el comportamiento de la recopilación de datos y la prioridad de las asociacio nes

Política administrada actualizada: AWS-SSM-RemediationAutomation-ExecutionRolePolicy	Systems Manager ha actualizado la política administrada `AWS-SSM-RemediationAutomation-ExecutionRolePolicy` para mejorar la seguridad de la API ssm:StartAutomationExecution al requerir permisos tanto para los tipos de recursos “document” como de “automation-execution”. Esta actualización se ha realizado para mejorar la seguridad de la API ssm:StartAutomationExecution, ya que exige permisos tanto para los tipos de recursos “document” como “automation-execution”, lo que proporciona permisos más completos y detallados para la ejecución de la automatización de las correcciones. Para obtener más información, consulte Actualizaciones de Systems Manager a políticas administradas de AWS.	16 de julio de 2025

Política administrada actualizada: AWS-SSM-RemediationAutomation-AdministrationRolePolicy	Systems Manager ha actualizado la política administrada `AWS-SSM-RemediationAutomation-AdministrationRolePolicy` a fin de admitir las mejoras en la autorización de las API para las operaciones de automatización de las correcciones. La política actualizada mejora los permisos para ejecutar las actividades definidas en los documentos de automatización, con controles de seguridad y patrones de acceso a los recursos mejorados para los flujos de trabajo de remediación. Para obtener más información, consulte Actualizaciones de Systems Manager a políticas administradas de AWS.	16 de julio de 2025

Política administrada actualizada: AWS-SSM-DiagnosisAutomation-ExecutionRolePolicy	Systems Manager ha actualizado la política administrada `AWS-SSM-DiagnosisAutomation-ExecutionRolePolicy` a fin de conceder permisos más detallados y precisos para la ejecución de la automatización del diagnóstico. La política actualizada incluye descripciones mejoradas para el acceso a los recursos de Amazon EC2 y Amazon VPC, permisos de automatización de SSM más específicos y descripciones de permisos de IAM y AWS KMS mejoradas con las restricciones de recursos adecuadas. Para obtener más información, consulte Actualizaciones de Systems Manager a políticas administradas de AWS.	16 de julio de 2025

Política administrada actualizada: AWS-SSM-DiagnosisAutomation-AdministrationRolePolicy	Systems Manager ha actualizado la política administrada `AWS-SSM-DiagnosisAutomation-AdministrationRolePolicy` para incluir permisos y condiciones de seguridad más específicos para las operaciones de automatización del diagnóstico. La política actualizada proporciona controles de seguridad mejorados para el uso de claves de AWS KMS, el acceso a los buckets de Amazon S3 y las suposiciones de roles, con condiciones más estrictas basadas en los recursos y restricciones a nivel de cuenta. Para obtener más información, consulte Actualizaciones de Systems Manager a políticas administradas de AWS.	16 de julio de 2025
Se agregó documentación para la política administrada AmazonSSMAutomationRole	Systems Manager ha actualizado la documentación de la política administrada `AmazonSSMAutomationRole` existente. Para obtener más información, consulte Actualizaciones de Systems Manager a políticas administradas de AWS.	15 de julio de 2025

Seguridad mejorada para los documentos SSM con interpolación de variables de entorno	Los documentos de comandos AWS Systems Manager ahora admiten la interpolación de variables de entorno al procesar los parámetros. Esta característica, disponible en la versión 2.2 del esquema y en la versión 3.3.2746.0 o superior de SSM Agent, ayuda a prevenir los ataques de inyección de comandos al tratar los valores de los parámetros como cadenas literales y no como comandos potencialmente ejecutables. Cuando se establece en `ENV_VAR`, el agente crea una variable de entorno denominada con el formato `SSM_`*`parameter-name`* que contiene el valor del parámetro. Para obtener más información, consulte los temas siguientes: • Componentes del documento • Ejemplos de gestión segura de parámetros • Elementos y parámetros de datos • Referencia de complementos del documento de comandos	14 de julio de 2025

La sincronización de datos de recursos de Explorer está disponible en cuatro Regiones de AWS registradas	De forma predeterminada, las sincronizaciones de datos de recursos cross-account/cross-Region de Explorer entre cuentas o regiones no admiten la agregación de datos en las regiones registradas. El 30 de junio de 2025, se agregó compatibilidad para las siguientes regiones registradas. • Europa (Milán) • África (Ciudad del Cabo) • Medio Oriente (Baréin) • Asia-Pacífico (Hong Kong) Tenga en cuenta que no puede usar una cuenta de administrador delegado para crear una sincronización de datos de recursos en las regiones registradas. Debe usar una cuenta de administración de AWS Organizations. Para obtener más información, consulte Acerca de las sincronizaciones de datos de recursos de varias cuentas y regiones.	30 de junio de 2025

Versiones nuevas de la extensión de Lambda para secretos y parámetros de AWS	Ya están disponibles las versiones nuevas de la extensión de Lambda para secretos y parámetros de AWS en todas las regiones y arquitecturas compatibl es. Además, se ha añadido compatibilidad para la región Asia-Pacífico (Taipéi) (ap-east-2).	16 de junio de 2025
Nuevo tema: Integración de Change Calendar a Amazon EventBridge	El tema Integración de Change Calendar a Amazon EventBridge proporciona información sobre esta integración en las siguientes categorías: • Fiabilidad y temporización de los eventos • Modificaciones de eventos y consideraciones de temporización • Eventos adyacentes y superpuestos • Prácticas recomendadas	25 de junio de 2025
Actualización de soporte en tiempo de ejecución para la acción de automatización 'aws:executeScript'	El parámetro `Runtime` de la acción de automatización aws:executeScript ahora es compatible con los scripts de PowerShell 7.4 (`dotnet8`).	24 de junio de 2025

SSM Agent y Patch Manager ahora son compatibles con RHEL 10 y Ubuntu Server 25.04	Ahora puede gestionar las máquinas Red Hat Enterprise Linux 10 y Ubuntu Server 25.04 con Systems Manager, incluida la aplicación de parches a los nodos gestionados mediante Patch Manager. Para obtener las listas completas de las versiones y sistemas operativos, consulte los siguientes temas: • Sistemas operativos compatibles con Systems Manager • Sistemas operativos compatibles con Patch Manager	23 de junio de 2025

Actualizaciones a los temas del acceso a los nodos justo a tiempo	Hemos realizado las siguientes actualizaciones en el contenido del capítulo Acceso a los nodos justo a tiempo mediante Systems Manager: • Se aclaró el soporte de autenticación para el acceso a los nodos justo a tiempo. Para obtener más información, consulte Cómo configurar acceso justo a tiempo con Systems Manager. • Se aclaró el requisito de etiquetar las claves de AWS KMS utilizadas para el cifrado de Session Manager y el registro de RDP en el acceso a nodos justo a tiempo con la clave de etiqueta `SystemsManagerJustInTimeNodeAccessManaged` y el valor de etiqueta `true`. Para obtener más información, consulte Actualización de las preferencias de sesión de acceso a los nodos justo a tiempo y Configuración del cifrado de bucket de S3 para las grabaciones RDP. • Se actualizó la configuración de las notificaciones para las solicitudes de acceso justo a tiempo con detalles	17 de junio de 2025

	de configuración adicional es.	
Grabación RDP mejorada para el acceso a nodos justo a tiempo	El acceso a nodos justo a tiempo de AWS Systems Manager ahora proporciona información adicional sobre las configuraciones de grabación RDP. Cuando se utiliza el cifrado de AWS KMS en los buckets de S3 donde se almacenan las grabaciones, debe dar acceso a la clave de KMS utilizada para el cifrado del bucket en la entidad principal de servicio `ssm-guiconnect` . Esta versión también introduce un nuevo estado de grabación , `ProcessingError` , que indica fallos intermedios durante el procesamiento del vídeo que podrían deberse a fallos de dependencia del servicio o a permisos faltantes . Para obtener más información, consulte Grabación de conexiones RDP.	13 de junio de 2025

Información actualizada sobre la selección e instalación de revisiones en el Patch Manager	Se ha añadido información para explicar cómo el Patch Manager gestiona los paquetes obsoletos durante la selección e instalación de revisiones. Para obtener más información, consulte Cómo se seleccionan las revisiones de seguridad y Cómo se instalan las revisiones.	12 de junio de 2025
Se ha actualizado la política administrada para Quick Setup: AWSQuickSetupDeploymentRolePolicy	Systems Manager ha actualizado la política administrada `AWSQuickSetupDeploymentRolePolicy` para que Quick Setup pueda crear asociaciones mediante esta política administrada en lugar de políticas insertadas. Para obtener más información, consulte Actualizaciones de Systems Manager a políticas administradas de AWS.	10 de junio de 2025

Introducción más detallada a Patch Manager	Para presentar mejor Patch Manager a los clientes y usuarios potenciales, añadimos al capítulo de Patch Manager las tres secciones siguientes: • ¿Cómo puede Patch Manager beneficiar a mi organización? • ¿Quién debe utilizar Patch Manager? • ¿Cuáles son las características principales de Patch Manager?	21 de mayo de 2025
Se ha actualizado la política administrada para Quick Setup: AWSQuickSetupSSMDeploymentRolePolicy	Systems Manager actualizó la política administrada `AWSQuickSetupSSMDeploymentRolePolicy` para proporcionar permisos a fin de etiquetar los roles de IAM y los recursos de Lambda creados para la consola unificada. Para obtener más información, consulte Actualizaciones de Systems Manager a políticas administradas de AWS.	7 de mayo de 2025

Opciones de configuración de características para la consola unificada	Systems Manager ahora ofrece opciones de configuración de características para la consola unificada. Estas configuraciones de características le permiten elegir si desea establecer la DHMC, la recopilación de metadatos de inventario y las actualizaciones de SSM Agent automáticas. Para obtener más información, consulte Configuración de la consola unificada de Systems Manager para una organización y Configuración de la consola unificada de Systems Manager para una sola cuenta y región.	7 de mayo de 2025
Se ha actualizado la política administrada para Quick Setup: AWSSystemsManagerJustInTimeNodeAccessRolePropagationPolicy	Systems Manager actualizó la política administrada `AWSSystemsManagerJustInTimeNodeAccessRolePropagationPolicy` para proporcionar permisos a fin de etiquetar un recurso compartido por AWS Resource Access Manager para el acceso a los nodos justo a tiempo. Para obtener más información, consulte Actualizaciones de Systems Manager a políticas administradas de AWS.	29 de abril de 2025

Se ha actualizado la política administrada para Quick Setup: AWSQuickSetupManageJITNAResourcesExecutionPolicy	Systems Manager actualizó la política administrada `AWSQuickSetupManageJITNAResourcesExecutionPolicy` para proporcionar permisos a fin de etiquetar los roles de IAM creados para el acceso a los nodos justo a tiempo. Para obtener más información, consulte Actualizaciones de Systems Manager a políticas administradas de AWS.	29 de abril de 2025

Se lanza el acceso a los nodos justo a tiempo en Systems Manager	AWS Systems Manager ahora admite un método de acceso justo a tiempo para conectarse a los nodos administrados. Este nuevo método de conexión a los nodos administrados permite mejorar la seguridad de los nodos, ya que proporcio na acceso temporal y con límite de tiempo a los nodos solo cuando es necesario, lo que elimina la necesidad de tener permisos permanentes. Además, Systems Manager proporciona la grabación de sesiones RDP en los nodos de Windows Server para que pueda respetar los requisito s de cumplimiento, realizar análisis de causas raíz, entre otras tareas. Para obtener más información, consulte Acceso a los nodos justo a tiempo mediante Systems Manager.	29 de abril de 2025
Nuevas políticas administr adas para el acceso a los nodos justo a tiempo	En Systems Manager se lanzaron varias políticas administradas nuevas para admitir el acceso a los nodos justo a tiempo. Para obtener más información, consulte Actualizaciones de Systems Manager a políticas administr adas de AWS.	29 de abril de 2025

Versiones nuevas de la extensión de Lambda para secretos y parámetros de AWS	Ya están disponibles las versiones nuevas de la extensión de Lambda para secretos y parámetros de AWS en todas las regiones y arquitecturas compatibles.	23 de abril de 2025
Nuevo tema: “¿En qué consiste el cumplimiento en Patch Manager?”	El cumplimiento de los parches para los nodos administrados de su flota no lo definen AWS ni otros terceros. El nuevo tema ¿En qué consiste el cumplimiento en Patch Manager? aclara que el cumplimiento de las revisione s para un nodo administrado por Systems Manager significa que todas las revisiones se han instalado en el nodo y cumplen los criterios de aprobación especificados en la línea de base de revisione s asociada. El cumplimiento lo define usted, en las reglas que configure en una línea de base de revisiones o en una línea de base de revisione s predefinida por AWS que usted elija para sus operacion es de aplicación de revisiones.	11 de abril de 2025

Versiones nuevas de la extensión de Lambda para secretos y parámetros de AWS	Ya están disponibles las versiones nuevas de la extensión de Lambda para secretos y parámetros de AWS. Además, se ha agregado compatibilidad de extensión de arquitectura para las siguientes regiones: • Asia-Pacífico (Tailandia) (ap-southeast-7) • México (centro) (mx-centr al-1)	3 de abril de 2025
Tema nuevo: Configuración de permisos para Compliance	Como práctica recomendada de seguridad, sugerimos que actualices el rol AWS Identity and Access Management (IAM) que utilizan sus nodos administrados para restringir la capacidad del nodo de utilizar la acción de la API PutCompli anceItems. Esta acción de API registra un tipo de conformid ad junto con sus detalles en un recurso designado, como una instancia de Amazon EC2 o un nodo administrado. Para obtener más informaci ón, consulte Configuración de permisos para Compliance.	11 de marzo de 2025

Tema actualizado: Revisión de la información sobre los nodos	En la consola unificada de Systems Manager, la página Revisión de la información sobre los nodos informa sobre la cantidad de instancias EC2 administradas y no administradas en su organización o cuenta. Aclaramos que Systems Manager incluye una instancia EC2 en el recuento de "No administradas" cuando se encuentra en un estado `Stopped`. Para obtener más información, consulte ¿Qué es una instancia no administrada?.	25 de febrero de 2025

Tema actualizado: Verificación de la firma de SSM Agent	Los paquetes de instaladores Debian y RPM de AWS Systems Manager Agente (SSM Agent) para instancias de Linux están firmados criptográficamente. Puede utilizar la clave pública para verificar que el paquete del agente sea original y que no se haya modificado. Si hay algún tipo de daño o alteración en los archivos, se produce un error en la verificación. Puede verificar la firma del paquete del instalador con RPM o GPG. El tema incluye una nueva clave pública para SSM Agent. Para obtener más información, consulte Verificación de la firma de SSM Agent.	14 de febrero de 2025

Nuevo tema: Verificación de la firma del complemento de Session Manager	Los paquetes de instaladores RPM y Debian del complemento de Session Manager para instancias de Linux están firmados criptográficamente. Puede usar la clave pública para verificar que el binario y el paquete del complemento sean originales y que no se hayan modificado. Si hay algún tipo de daño o alteración en el archivo, se produce un error en la verificación. Puede verificar la firma del paquete del instalador mediante la herramienta GNU Privacy Guard (GPG). Para obtener más información, consulte Verificación de la firma del complemento de Session Manager.	13 de febrero de 2025
Complemento de Session Manager actualizado	El complemento de Session Manager se actualizó recientemente con las siguientes mejoras: se actualizó la versión Go a la 1.23 en el Dockerfile. Se actualizó el paso de configuración de la versión en el archivo README. Para obtener más información sobre el complemento de Session Manager, consulte Instalación del complemento de Session Manager para la AWS CLI.	6 de febrero de 2025

Cuotas de TPS publicadas para las acciones de la API de Automatización	Hemos publicado las cuotas de transacciones por segundo (TPS), anteriormente denominadas limitadas, para varias acciones de la API relacionadas con Automatización de Systems Manager. Para obtener más información, consulte Service Quotas de Systems Manager en Referencia general de Amazon Web Services.	21 de enero de 2025
Cambio de terminología: las capacidades de Systems Manager ahora se denominan “herramientas”	Hasta ahora, las unidades funcionales que componen Systems Manager se denominaban capacidades, como la capacidad Automatización, la capacidad Parameter Store, etc. Estas unidades funcionales ahora se denominan herramientas, como la herramienta Automatización y la herramienta Parameter Store. Esta guía del usuario se ha actualizado para reflejar el cambio.	10 de enero de 2025

Compatibilidad de Patch Manager con versiones de macOS adicionales (AMI de Amazon EC2)	Patch Manager ahora es compatible con las versiones de macOS 12.7, 13.6, 13.7 y 15.x. Para obtener listas completas de los SO y las versiones compatibles con Patch Manager, consulte Sistemas operativos admitidos por Patch Manager.	8 de enero de 2025
Compatibilidad de SSM Agent y Patch Manager con versiones adicionales: CentOS Stream, Ubuntu Server y Windows Server	Patch Manager ahora es compatible con CentOS Stream 9, Ubuntu Server 24.10 y Windows Server 2025. SSM Agent ahora es compatible con Ubuntu Server 24.10 y Windows Server 2025. (La compatibilidad de Agent para CentOS Stream 9 ya estaba disponible con anterioridad). Para obtener las listas completas de las versiones y sistemas operativos, consulte los siguientes temas: • Sistemas operativos compatibles con Systems Manager • Sistemas operativos compatibles con Patch Manager	22 de noviembre de 2024

Tema nuevo: Cifrado de AWS KMS para los parámetros SecureString del Parameter Store	Consulte el siguiente tema para obtener más informaci ón sobre cómo el Parameter Store de AWS Systems Manager usa AWS Key Management Service para cifrar los valores de los parámetros `SecureString` en Parameter Store: • Cifrado de AWS KMS para los parámetros SecureString del Parameter Store	22 de noviembre de 2024
Políticas administradas nuevas y actualizadas para Systems Manager	Para admitir las nuevas características de Systems Manager, lanzaremos varias nuevas políticas administradas que respaldarán las nuevas configuraciones y operacion es de Systems Manager. Además, actualizaremos otras políticas administradas. Para obtener más información, consulte Actualizaciones de Systems Manager para las políticas administradas de AWS.	21 de noviembre de 2024

Una experiencia nueva y simplificada de administración de nodos para Systems Manager	AWS Systems Manager ha lanzado una nueva experiencia de consola unificada para administrar nodos a escala en todas las cuentas y regiones. Ahora puede ver todos los nodos administrados y no administrados de las regiones y Cuentas de AWS de sus organizaciones desde un único lugar. También puede identificar, diagnosticar y corregir los nodos no administrados. Systems Manager ahora también está integrado con Amazon Q Developer (Amazon Q), lo que amplía su capacidad de ver y controlar los nodos desde cualquier parte de la AWS Management Console mediante el ingreso de peticiones en lenguaje natural. Con esta versión, ahora también puede utilizar AWS Organizations para permitir una cuenta de administrador delegado a fin de administrar los nodos de toda la organización desde un punto de vista central. Para obtener más información, consulte los temas siguientes: • ¿Qué es AWS Systems Manager?	21 de noviembre de 2024

	• Configuración de AWS Systems Manager • Completar las tareas de nodos con AWS Systems Manager	
Corrección de errores en el complemento de Session Manager	El complemento de Session Manager se actualizó recientemente con la siguiente corrección de errores: se anuló el cambio que agregaba credenciales a las solicitudes de OpenDataChannel.	20 de noviembre de 2024
Mejoras del complemento de Session Manager	Esta versión quedó obsoleta el 20 de noviembre de 2024. El complemento de Session Manager se actualizó recientemente con las siguientes mejoras. • Se agregaron credencia les a las solicitudes de OpenDataChannel. • Se actualizaron los paquetes dependientes `testify` y `objx`.	6 de noviembre de 2024

Compatibilidad adicional con la versión del sistema operativo de macOS	Systems Manager ahora es compatible con la versión 15.x (Sequoia) del sistema operativo macOS (solo instancias de EC2). Para obtener una lista de los sistemas operativos y las versiones compatibles, consulte Sistemas operativos compatibles con Systems Manager.	6 de noviembre de 2024
Patch Manager: se actualizan los formatos de los nombres de paquetes admitidos en las listas de aprobados y rechazados	En varios sistemas operativos, actualizamos y ampliamos las listas de formatos de los nombres de paquetes que se pueden especificar en las listas Revisiones aprobadas y Revisiones rechazadas en las líneas de base de revisiones. Para obtener más información, consulte Formatos de nombres de paquetes para Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022, Amazon Linux 2023, CentOS, Oracle Linux y Red Hat Enterprise Linux (RHEL).	1 de noviembre de 2024

Compatibilidad adicional con la versión operativa para Patch Manager	Patch Manager ahora admite versiones adicionales de Oracle Linux (8.10 y 9.4) y Ubuntu Server (23.10 y 24.04). Para obtener las listas de todas las versiones y sistemas operativos admitidos, consulte los siguientes temas: • Sistemas operativos compatibles con Systems Manager • Sistemas operativos compatibles con Patch Manager	1 de noviembre de 2024
Compatibilidad de SSM Agent con versiones adicionales CentOS Stream, Oracle Linux y Ubuntu Server	SSM Agent ahora admite CentOS Stream 9, Oracle Linux 8.10 y 9.4, y Ubuntu Server 24.04 LTS, además de las versiones anteriores admitidas. Para obtener una lista completa de los sistemas operativos y las versiones compatibles con Systems Manager, consulte Sistemas operativos compatibles con Systems Manager.	25 de octubre de 2024
Mejora del complemento Session Manager	Se ha agregado compatibilidad para pasar la versión del complemento con las solicitudes de OpenDataChannel.	10 de octubre de 2024

Nuevo: utilizar Fleet Manager para ver los detalles de las conexiones RDP	Ahora puede ver información sobre las conexiones del protocolo de escritorio remoto que han realizado los usuarios en la Cuenta de AWS. Para obtener más información, consulte Visualizar información sobre las conexiones actuales y completadas.	10 de octubre de 2024
Patch Manager ahora admite SLES versión 15.6	Se ha publicado el soporte de revisiones para SUSE Linux Enterprise Server (SLES) 15.6. Ahora puede aplicar revisiones a máquinas de la versión SLES 15.6 mediante Patch Manager. Para obtener una lista completa de los sistemas operativos y las versiones compatibles con Patch Manager, consulte Sistemas operativos compatibles con Patch Manager.	29 de septiembre de 2024

Versiones nuevas de la extensión de Lambda para secretos y parámetros de AWS	Ya están disponibles las versiones nuevas de la extensión de Lambda para secretos y parámetros de AWS. Ya hay disponible compatibilidad para todas las arquitecturas en la región Asia-Pacífico (Malasia) (ap-southeast-5). Además, se ha añadido compatibilidad de extensión de arquitectura para ARM64 y Mac with Apple silicon para las siguientes regiones: • Asia Pacífico (Hyderabad) (ap-south-2) • Asia Pacífico (Melbourne) (ap-southeast-4) • Oeste de Canadá (Calgary) (ca-west-1) • Europa (Zúrich) (eu-central-2) • Europa (España) (eu-south-2) • Medio Oriente (EAU) (me-central-) • China (Pekín) (cn-north-1) • China (Ningxia) (cn-northwest-1) • Israel (Tel Aviv) (il-central-1) • AWS GovCloud (EE. UU. Este) (us-gov-east-1)	19 de septiembre de 2024

	• AWS GovCloud (EE. UU. Oeste) (us-gov-west-1)	
Tema nuevo: configuración de permisos para periodos de mantenimiento mediante la AWS CLI	El tema Configuración de permisos para periodos de mantenimiento mediante la AWS CLI proporciona instrucciones para crear un rol de servicio personalizado (y sus políticas) para ejecutar las tareas de periodos de mantenimiento en nombre de un usuario.	19 de agosto de 2024
SSM Agent y Patch Manager admiten versiones adicionales: AlmaLinux, Oracle Linux y Rocky Linux	SSM Agent Patch Manager ahora son compatibles con las versiones 8.10 y 9.3 y 9.4 de AlmaLinux, y Rocky Linux y la versión 9.3 de Oracle Linux, además de las versiones anteriores compatibles. Para obtener las listas completas de las versiones y sistemas operativos, consulte los siguientes temas: • Sistemas operativos compatibles con Systems Manager • Sistemas operativos compatibles con Patch Manager	14 de agosto de 2024

Nueva condición de la política de IAM para el soporte de Parameter Store: ssm:Policies	Mediante `ssm:Policies` , una condición recientemente admitida para las políticas de IAM, puede impedir que las entidades creen o actualice n parámetros que incluyan una política de parámetros. Para obtener más información, consulte los temas siguientes: • ssm:Policies: impiden la creación o actualización de parámetros que utilizan políticas de parámetros • Claves de condición paraAWS Systems Manager en la Referencia de autorización del servicio.	14 de agosto de 2024
Se ha actualizado la política administrada para Quick Setup: SSMQuickSetupRolePolicy	Systems Manager ha actualizado la política gestionada `SSMQuickSetupRolePolicy` para conceder acceso a conjuntos de pilas adicionales de AWS CloudFormation. Para obtener más información, consulte Actualizaciones de Systems Manager a políticas administradas de AWS.	13 de agosto de 2024

Soporte para el aprovisionamiento y la gestión de los recursos de Systems Manager mediante Terraform	Hemos añadido Terraform de HashiCorp a la lista de integraciones de terceros compatibles con Systems Manager. Terraform es una herramienta de software de infraestructura como código (IaC) de código abierto que proporciona un flujo de trabajo de interfaz de la línea de comandos (CLI) para administrar varios servicios en la nube. Se puede usar Terraform para aprovisionar y administrar varios recursos y orígenes de datos de Systems Manager de uso común. Para obtener información sobre esta y otras integraciones de terceros con Systems Manager, consulte Integración con otros productos y servicios.	1 de agosto de 2024

Nueva experiencia de consola de Quick Setup y API	Systems Manager de Quick Setup ha lanzado una nueva experiencia de consola y una nueva API. Ahora puede interactuar con esta API mediante la consola, la AWS CLI, AWS CloudFormation y los SDK. Puede acceder a la nueva consola mediante la consola de Quick Setup. Para obtener más informaci ón sobre la incorporación a la nueva experiencia de Quick Setup, consulte Introducción a la Configuración Rápida. Para obtener más informaci ón sobre las operaciones de la API disponibles mediante la API de Quick Setup, consulte la Referencia de la API de Quick Setup.	1 de agosto de 2024

Tema nuevo: opciones de listas de revisiones rechazadas en líneas de base de revisiones personalizadas	Para las operaciones de aplicación de revisiones en las que se utiliza una línea de base de revisiones personalizada en Patch Manager, hemos aclarado el comportamiento cuando se asigna la acción a una revisión añadida a la lista de revisiones rechazadas. `Allow as dependency` Como Windows Server no admite el concepto de dependencias de revisiones, se omiten las revisiones que aún no estén instaladas en un nodo gestionado. A las revisiones ya instaladas en el nodo se les asigna el estado `INSTALLED_REJECTED` . Para obtener más información, consulte las opciones de la lista de revisiones rechazadas en las líneas de base de revisiones personalizadas y los valores de conformidad de las revisiones para otros sistemas operativos.	23 de julio de 2024

Tema nuevo: configuración de SSM Agent para usar con el Estándar de procesamiento de la información federal (FIPS).	Hemos proporcionado instrucciones para la configuración de SSM Agent para su uso con el estándar federal de procesamiento de la información (FIPS). Para obtener más información, consulte Configuración de SSM Agent para su uso con el Estándar de procesamiento de la información federal (FIPS).	22 de julio de 2024
Actualización: se ha aclarado la compatibilidad del símbolo @ en los nombres de usuario de Fleet Manager	Si un nombre de usuario del Centro de Identidad de IAM contiene uno o más símbolos @, RDP de Fleet Manager ignora el primer símbolo @ y todos los caracteres que le siguen, independientemente de que @ introduzca o no la parte de dominio de una dirección de correo electrónico. Para obtener más información sobre los caracteres compatibles con los nombres de usuario en las conexiones de RDP de Fleet Manager, consulte Autenticación de las conexiones de escritorio remoto.	21 de julio de 2024

Política administrada actualizada: AmazonSSMManagedEC2InstanceDefaultPolicy	Systems Manager ha actualizado la política gestionada `AmazonSSMManagedEC2InstanceDefaultPolicy` al proporcionar ID de declaración en línea (SID) para aclarar el propósito de cada declaración de política. Para obtener más información, consulte Actualizaciones de Systems Manager a políticas administradas de AWS.	18 de julio de 2024

Cambios de nombre en los buckets de AWS gestionados para las operaciones de aplicación de revisiones de Patch Manager	AWS posee y mantiene varios buckets de Amazon S3 a los que SSM Agent accede mientras se realizan diversas operaciones de aplicación de revisiones de Patch Manager. Se puede acceder públicamente a estos buckets de S3 y, de forma predeterminada, SSM Agent se conecta a ellos a través de llamadas a HTTP. Sin embargo, si utiliza un punto de conexión de nube privada virtual (VPC) en las operaciones de Systems Manager, debe conceder un permiso explícito en un perfil de instancia de Amazon Elastic Compute Cloud (Amazon EC2) para Systems Manager o en un rol de servicio para equipos que no sean de EC2 en un entorno híbrido y multinube. De lo contrario, sus recursos no puede acceder a estos buckets públicos. En la mayoría de los casos, cambiamos los nombres de estos buckets. Por ejemplo, para las operaciones de aplicación de revisiones, el bucket `aws-patchmanager-macos-us-east-2` se sustituye por `aws-`	18 de julio de 2024

`patchmanager-macos-us-east-2-552881074` y el bucket `aws-ssm-us-east-2` se sustituye por. `aws-patch-manager-us-east-2-552881074`
Para obtener más información, consulte los temas siguientes:

- Comunicaciones de SSM Agent con buckets de S3 administrados de AWS
- Referencia: buckets de Amazon S3 para operaciones de aplicación de revisiones

Nuevo rol vinculado a servicio de Quick Setup	Systems Manager ha lanzado una nueva función vinculada al servicio, `AWSServiceRoleForSSMQuickSetup` . Systems Manager utiliza este rol para comprobar el estado de los recursos de la configuración mediante Quick Setup, para garantizar el uso consistente de los parámetros y los recursos aprovisionados y corregir los recursos cuando se detecta una desviación. La política administrada asociada con este rol es `SSMQuickSetupRolePolicy` . Para obtener más información, consulte Permisos de roles vinculados a servicios de AWSServiceRoleForSSMQuickSetup para Systems Manager.	3 de julio de 2024
Nuevas políticas gestionadas para los tipos de configuración de Quick Setup	Systems Manager ha publicado 12 nuevas políticas administradas adicionales para admitir varios tipos de configuración y procesos de Quick Setup. Para obtener más información, consulte Actualizaciones de Systems Manager a políticas administradas de AWS.	3 de julio de 2024

Compatible con RHEL 8.10 y 9.4	Systems Manager y Patch Manager ahora son compatibles con las versiones 8.10 y 9.4 de Red Hat Enterprise Linux. Para obtener más información, consulte Sistemas operativos y tipos de máquinas compatibles y Sistemas operativos compatibles para Patch Manager.	26 de junio de 2024
Patch Manager es compatible con las versiones 8.8 y 8.9: AlmaLinux, Oracle Linux y Rocky Linux	Patch Manager ahora es compatible con las versiones 8.8 y 8.9 de AlmaLinux, Oracle Linux y Rocky Linux, además de las versiones 8.x anteriores. Para obtener listas completas de los sistemas operativos y las versiones compatibles para Patch Manager, consulte Sistemas operativos compatibles con Patch Manager.	17 de junio de 2024

Nuevos parámetros públicos para AMIs de macOS Amazon EC2	Se han publicado parámetros públicos para que sean compatibles con Amazon Machine Images para instancias de macOS Amazon Elastic Compute Cloud. Para obtener más información, consulte los siguientes temas. • Búsqueda de parámetros públicos • Llamada a parámetros públicos de AMI para macOS • Lanzar una instancia de Mac en la Guía del usuario de Amazon EC2 • Instancias de Amazon EC2 Mac en la Guía del usuario de Amazon EC2 • Use los parámetros de AWS Systems Manager en lugar de los identificadores de AMI en las plantillas de lanzamiento en la Guía del usuario de Amazon EC2 Auto Scaling.	17 de junio de 2024

Actualización: Disponibilidad regional de la ruta de parámetro `/aws/service/global-infrastructure`	Hemos aclarado desde qué regiones comerciales se puede consultar la ruta de parámetro público `/aws/service/global-infrastructure` y cómo ejecutar una consulta para la ruta si se trabaja en una Región de AWS comercial diferente. Para obtener información, consulte Llamada a parámetros públicos para servicios, regiones, puntos de conexión, zonas de disponibilidad, zonas locales y zonas de Wavelength de AWS.	12 de junio de 2024
Novedad: capítulo de ejemplos de código	En un nuevo capítulo, Code examples for Systems Manager using AWS SDKs, se proporcionan ejemplos en distintos lenguajes de SDK sobre cómo trabajar con el servicio de Systems Manager.	8 de mayo de 2024

Cambios en la compatibilidad con puntos de conexión de `ec2messages:*`	Para las Regiones de AWS que se lanzan en 2024 o después, los puntos de conexión de `ec2messages:*` no son compatibles con SSM Agent para enviar información de estado y ejecución al servicio de Systems Manager. Las cuentas de estas regiones deben utilizar `ssmmessages:*`. En las regiones que se lanzaron antes de 2024, tanto `ssmmessages:*` como `ec2messages:*` siguen siendo compatibles, pero ahora recomendamos usar solo el punto de conexión de `ssmmessages:*` (Amazon Message Gateway Service). Puede eliminar los permisos `ec2messages:*` de sus políticas de forma segura en este momento. Para obtener más información, consulte Uso de SSM Agent y Operaciones de API relacionadas con el agente (puntos de conexión de ssmmessages y ec2messages).	3 de mayo de 2024

Hay tiempos de ejecución adicionales disponibles para ejecutar scripts en los manuales de procedimientos de Automatización	La acción `aws:executeScript` ahora es compatible con los tiempos de ejecución de Python 3.9, 3.10 y 3.11. Para obtener más información sobre el uso de esta acción, consulte `aws:executeScript`.	23 de abril de 2024
Compatibilidad con las versiones 8.8 y 8.9: AlmaLinux, Oracle Linux y Rocky Linux	Systems Manager ahora es compatible con las versiones 8.8 y 8.9 de AlmaLinux, Oracle Linux y Rocky Linux, además de las versiones 8.x anteriores. Para obtener listas completas de los sistemas operativos y las versiones compatibles, consulte Sistemas operativos compatibles con Systems Manager.	22 de abril de 2024

Patch Manager: cambiar al estado de revisión 'INSTALLED_PENDING_REBOOT'	Anteriormente, solo los parches instalados por Patch Manager podían marcarse como `INSTALLED_PENDING_REBOOT`. Los parches instalados fuera de Patch Manager nunca reciben este estado. Ahora, `INSTALLED_PENDING_REBOOT` se puede aplicar a cualquier revisión que se haya aplicado a un nodo administrado desde la última vez que se reinició. Esto incluye los parches instalados por Patch Manager con la opción `NoReboot` seleccionada y los parches instalados fuera de Patch Manager tras el último reinicio del nodo. Para obtener descripciones de todos los valores de estado de las revisiones de Patch Manager, consulte Descripción de los valores de estado de conformidad de las revisiones.	16 de abril de 2024
Compatible con RHEL 8.9 y 9.3	Systems Manager, incluido Patch Manager, ahora es compatible con las versiones 8.9 y 9.3 de Red Hat Enterprise Linux (RHEL), además de las versiones 8.x y 9.x anteriores.	26 de marzo de 2024

Actualización de tema: políticas administradas de AWS para AWS Systems Manager	El tema Políticas administradas de AWS para AWS Systems Manager proporciona información sobre las cuatro políticas administradas de Systems Manager que se han introducido o actualizado desde el 12 de marzo de 2021. Hemos agregado una sección a este tema con información sobre las otras 12 políticas administradas para su uso con Systems Manager que se crearon o actualizaron por última vez antes de esa fecha. Para obtener más información, consulte Políticas administradas adicionales para Systems Manager.	1 de marzo de 2024

Compatibilidad con el uso compartido entre cuentas de Parameter Store	Ahora puede compartir parámetros avanzados de forma segura y eficiente en sus Cuentas de AWS o dentro de AWS mediante la configuración del uso compartido de recursos. El uso compartido de recursos le permite centralizar la administración de la configuración de las aplicaciones y reducir la sobrecarga operativa que supone compartir los parámetros con todas las cuentas que posea. Los parámetros se pueden compartir entre cuentas mediante la consola Parameter Store, la consola AWS RAM o el AWS CLI. Para obtener más información, consulte Trabajo con parámetros compartidos.	21 de febrero de 2024
Mejora de las acciones de automatización	Ahora puede usar las propiedades `onFailure` y `isCritical` con la acción `aws:approve`. Para obtener más información sobre la acción `aws:approve`, consulte aws:approve — Pausar una automatización para su aprobación manual.	12 de febrero de 2024

Compatibilidad adicional con la versión operativa para Patch Manager	Hemos agregado contenido a la lista de versiones de sistemas operativos compatibles para Patch Manager. También se ha agregado compatibilidad con: • Debian Server 11.x y 12.x • macOS 14.x (Sonoma) • SUSE Linux Enterprise Server (SLES) 15.5 • Ubuntu Server 23.04	4 de enero de 2024
Configuración de las actualizaciones de SSM Agent automatizadas mediante la consola de Application Manager	Ahora puede usar la consola de Application Manager para automatizar las actualizaciones de SSM Agent en las instancias de su aplicación. Para obtener más información, consulte Trabajar con las instancias de su aplicación.	21 de diciembre de 2023
Proceso actualizado para registrar máquinas que no son de Amazon EC2 en entornos híbridos y multinube	Ahora Systems Manager proporciona `ssm-setup-cli` para que pueda registrar máquinas que no sean de Amazon Elastic Compute Cloud (Amazon EC2) en entornos híbridos y multinube. Para obtener más información, consulte Cómo instalar SSM Agent en nodos de Linux híbridos y Cómo instalar SSM Agent en nodos de Windows híbridos.	20 de diciembre de 2023

Administración de los volúmenes de Amazon EBS mediante Fleet Manager	Ahora puede utilizar Fleet Manager, una herramienta de AWS Systems Manager, para administrar volúmenes de Amazon Elastic Block Store en las instancias administradas. Por ejemplo, puede inicializar un volumen de EBS, dar formato a una partición y montar el volumen para que esté disponible para su uso. Para obtener más información, consulte Administración de volúmenes de Amazon EBS en instancias administradas.	14 de diciembre de 2023
Mejora del complemento Session Manager	Incorporación de la compatibilidad para pasar una respuesta de la API StartSession como variable de entorno al complemento Session Manager.	4 de diciembre de 2023

Nueva experiencia de diseño visual para los manuales de procedimientos de automatización	Ahora puede crear y editar manuales de procedimientos mediante una nueva experiencia de diseño visual desarrollada por la automatización de Systems Manager. La experiencia de diseño visual proporciona una interfaz low-code, de arrastrar y soltar, para que pueda crear y editar manuales de procedimientos con mayor facilidad. Para obtener más información, consulte experiencia de diseño visual para manuales de procedimientos de automatización.	26 de noviembre de 2023

Nuevas acciones de automatización de Systems Manager, elementos de datos y mejoras funcionales para manuales de procedimientos	Ahora puede repasar varias acciones de un manual de procedimientos utilizando la acción `aws:loop`. Esta nueva acción es compatible con do `while` y bucles de estilo de `for each`. Además, con el nuevo elemento de datos de variables, puede definir, referenciar y actualizar valores de forma dinámica en el contexto de un manual de procedimientos. Para actualizar el valor de una variable en el manual de procedimientos, utilice la nueva acción `aws:updateVariable` . La automatización también ha añadido compatibilidad con las conversiones dinámicas de tipos de datos para las salidas. Esto significa que si el valor de una salida no coincide con el tipo de datos que especificó, la automatización intentará convertir el tipo de datos. Por ejemplo, si el valor devuelto es un `Integer`, pero el valor `Type` especificado es `String`, el valor de salida final es un valor `String`. Por último, la automatización ahora admite las expresiones de filtro JSONPath para los selectores. Para obtener más información, consulte los temas siguientes:	17 de noviembre de 2023

	• aws:loop: realiza iteraciones sobre los pasos de una automatización • aws:updateVariable: actualiza el valor de una variable del manual de procedimientos • Elementos y parámetros de datos: elementos de datos de nivel superior • Uso de salidas de acción como entradas. • Uso de JSONPath en un manual de procedimientos.	
Se ha actualizado el soporte regional para las conexiones (RDP) de Remote Desktop Protocol	El escritorio remoto de Fleet Manager, que cuenta con tecnología Amazon DCV, brinda una conectividad segura a sus instancias de Windows Server directamente desde la consola de Systems Manager. Se han habilitado las siguientes tres regiones adicionales para las conexiones de escritorio remoto de Fleet Manager: • África (Ciudad del Cabo) (af-south-1) • Asia-Pacífico (Yakarta) (ap-southeast-3) • Israel (Tel Aviv) (il-central-1)	15 de noviembre de 2023

Patch Manager: compatibilidad ampliada con versiones de sistemas operativos para RHEL y macOS	Patch Manager ahora es compatible con las siguientes versiones de sistemas operativos adicionales: • Red Hat Enterprise Linux: versión 8.8 • macOS: 11.5-11.7 (Big Sur) • macOS: 12.0-12.6 (Monterey) • macOS: 13.0-13.5 (Ventura)	23 de octubre de 2023
Nueva API de OpsCenter: DeleteOpsItem	OpsCenter ahora ofrece la API DeleteOpsItem para OpsItems individuales. Para obtener más información, consulte DeleteOpsItem en la Referencia de la API de AWS Systems Manager.	20 de octubre de 2023

Nuevo tipo de configuración de Quick Setup: actualizaciones de SSM Agent para toda la organización	El nuevo tipo de configuración Configuración de la administración de hosts predeterminada permite que un administrador de la organización, según lo definido en AWS Organizations, pueda solicitar que se compruebe automáticamente si existen actualizaciones de SSM Agent y aplicarlas a todas las instancias de EC2 de las cuentas y regiones de la organización. Para obtener más información, consulte Administración de hosts predeterminada para una organización.	16 de octubre de 2023
Nuevo formato de título y descripción de OpsItems creado por Información de aplicaciones de CloudWatch	El título y la descripción de OpsItems creados por Información de aplicaciones de CloudWatch cambiarán a un formato mejorado el 16 de octubre de 2023. Para ver el formato nuevo, consulte Información de aplicaciones de Amazon CloudWatch.	29 de septiembre de 2023

Soporte para varias resoluciones de pantalla en conexiones RDP de Fleet Manager	Cuando se conecta a nodos administrados de Windows Server mediante la opción de Protocolo de escritorio remoto (RDP) en Fleet Manager, puede elegir la resolución de pantalla. Antes, todas las conexiones utilizaban una resolución fija de 720P (1366 x 768). Ahora puede elegir entre las siguientes opciones para cada conexión: • Ajuste automático (determina la resolución óptima en función del tamaño de pantalla detectado) • 1920 x 1080 • 1400 x 900 • 1366 x 768 • 800 x 600 Para obtener información, consulte Conexión a un nodo administrado mediante Escritorio remoto.	22 de septiembre de 2023

Tema nuevo: ID de línea de base de revisiones aleatorios en operaciones de política de revisiones	Agregamos contenido para describir cómo las políticas de revisiones de Quick Setup utilizan el parámetro `BaselineOverride` del documento de SSM Command `AWS-RunPatchBaseline` para generar ID aleatorios para las líneas de base de revisiones cada vez que se ejecuta una operación de política de revisiones. Para obtener información, consulte ID de línea de base de revisiones aleatorios en operaciones de políticas de revisiones	22 de septiembre de 2023

Información operativa nueva para administrar OpsItems	OpsCenter ahora incluye información operativa llamada Recursos que generan la mayor cantidad de OpsItems. Se genera información de este tipo cuando un recurso de AWS tiene más de 10 OpsItems abiertos. Utilice esta información para localizar recursos problemáticos. Utilice el manual de procedimientos `AWS-BulkResolveOpsItems` con la información para resolver rápidamente los OpsItems asociados con un recurso. Para obtener más información, consulte Análisis de la información operativa para reducir OpsItems.	22 de septiembre de 2023
Clave GPG pública actualizada	Se creó una clave pública nueva para verificar la firma de SSM Agent. Para obtener más información, consulte Verificación de la firma de SSM Agent.	5 de septiembre de 2023

Soporte agregado para versiones adicionales de AlmaLinux, Oracle Linux, RHEL y Rocky Linux	Las listas de sistemas operativos compatibles con AWS Systems Manager y Patch Manager se actualizaron para reflejar la compatibilidad con las siguientes versiones adicionales del SO: • AlmaLinux: 9.2 • Oracle Linux: 8.7 y 9.2 • Red Hat Enterprise Linux (RHEL): 8.7, 9.1 y 9.2 • Rocky Linux: 8.6 y 8.7, 9.0–9.2	30 de agosto de 2023
OpsCenter agregó soporte para el formato de Markdown en el campo de descripción de OpsItem.	OpsCenter ahora admite el formato de Markdown en el campo de descripción de OpsItem. Se admiten los siguientes tipos de formato de Markdown: • Párrafos • Interlineado • Líneas horizontales • Encabezados • Formato de texto • Enlaces • Listas Para obtener más información, consulte Uso de Markdown en la consola en la Introducción a la AWS Management Console en la Guía de introducción.	18 de agosto de 2023

Versiones nuevas de la extensión de Lambda para secretos y parámetros de AWS	Ya están disponibles las versiones nuevas de la extensión de Lambda para secretos y parámetros de AWS. Además, se agregó soporte de extensión para las regiones de Asia Pacífico (Melbourne) (ap-southeast-4) e Israel (Tel Aviv) (il-centr al-1) (solo para arquitecturas x86_64 y x86). Para obtener más información, consulte Uso de parámetros de Parameter Store en funciones de AWS Lambda.	16 de agosto de 2023

Actualización: se agregó información sobre los permisos necesarios para los buckets de políticas de revisiones de Quick Setup	Cuando crea una política de revisiones, Quick Setup crea un bucket de Amazon S3 que contiene un archivo denominado `baseline_overrides.json`. Este archivo almacena información sobre las líneas de base de revisiones que especificó para la política de revisiones. Cuando configura la política de revisiones, tiene la opción de seleccionar la casilla de verificación Agregar las políticas de IAM requeridas a los perfiles de instancia existentes asociados a sus instancias. Si decide no seleccionar esta opción, deberá proporcionar de manera manual determinados recursos con permisos para acceder a este bucket o es posible que las operaciones de la política fallen. Para obtener más información, consulte los temas siguientes: • Permisos para el bucket de S3 de la política de revisiones • Problema: error “Invoke-PatchBaselineOperation : Access Denied” o error “Unable to download file	6 de julio de 2023

	from S3” para `baseline_overrides.json`	
Uso de Quick Setup para configurar OpsCenter para la administración de OpsItem en varias cuentas	Quick Setup para OpsCenter ayuda a completar las siguientes tareas de administración de los OpsItems entre cuentas: • Especificación de la cuenta de administrador delegado • Creación de los roles y las políticas de AWS Identity and Access Management (IAM) requeridos • Especificación de una organización de AWS Organizations, o un subconjunto de cuentas de miembro, donde un administrador delegado pueda administrar OpsItems en las cuentas Para obtener más información, consulte (Opcional) Configuración de OpsCenter para administrar OpsItems en las cuentas mediante Quick Setup.	19 de junio de 2023

Actualización de los agentes de lanzamiento de Amazon EC2 mediante Quick Setup	Ahora puede permitir que Systems Manager compruebe cada 30 días si hay una versión nueva del agente de lanzamiento instalada en la instancia. Si hay una nueva versión disponible, Systems Manager actualiza el agente en la instancia. Para obtener más información, consulte Administración de host de Quick Setup.	19 de junio de 2023
Patch Manager ahora admite Ubuntu Server 22.04 LTS	A partir de ahora, puede utilizar Patch Manager para aplicar revisiones a los nodos de Ubuntu Server 22.04 LTS Al igual que otras versiones compatibles de Ubuntu Server, la versión 22.04 LTS utiliza la línea de base de revisiones `AWS-UbuntuDefaultPatchBaseline` administrada de AWS.	15 de mayo de 2023

Systems Manager ahora es compatible con AlmaLinux, incluido Patch Manager	Ahora puede utilizar Systems Manager para administrar los nodos de AlmaLinux 8.3-8.7; 9.0-9.1. Muchas de las reglas que se aplican a RHEL 8 para la aplicación de revisiones también se aplican a AlmaLinux. AlmaLinux utiliza la nueva `AWS-DefaultAlmaLinuxPatchBaseline` . Para obtener más información, consulte los temas siguientes: • Instalación manual de SSM Agent en instancias de AlmaLinux • Cómo se seleccionan las revisiones de seguridad • Cómo se instalan las revisiones • Funcionamiento de las reglas de líneas de base de revisiones en AlmaLinux, RHEL y Rocky Linux.	8 de mayo de 2023
Implementación del agente EC2Launch v2 mediante Quick Setup	Ahora puede implementar el agente EC2Launch v2 mediante Quick Setup. Para obtener más información, consulte Implementación de paquetes de Distributor con Quick Setup.	13 de abril de 2023

Systems Manager ahora es compatible con Amazon Linux 2023	Systems Manager ahora es compatible con el nuevo tipo de instancia de EC2 de Amazon Linux 2023 (AL2023) y con las operaciones de Patch Manager. Muchas de las reglas de aplicación de revisiones que se aplican a Amazon Linux 2 también se aplican a Amazon Linux 2023 (Patch Manager también sigue siendo compatible con la versión preliminar de Amazon Linux 2022). Para obtener más información, consulte los temas siguientes: • Cómo se seleccionan las revisiones de seguridad • Cómo se instalan las revisiones • Funcionamiento las reglas de línea de base de revisiones en Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 y Amazon Linux 2023	23 de marzo de 2023

Contenido de configuración revisado para instancias de Amazon EC2	Hemos revisado el contenido de configuración para las instancias de Amazon EC2. Ahora se recomienda utilizar la configuración de administración de host predeterminados recientemente publicada para los permisos de instancia. Para obtener más información, consulte Configuración de permisos de instancia requeridos para Systems Manager.	15 de febrero de 2023
Administración automática de instancias con la configuración de administración de host predeterminada	Ahora puede administrar automáticamente las instancias de Amazon EC2 en una Región de AWS completa mediante Systems Manager. Para obtener más información, consulte Configuración de administración de host predeterminada.	15 de febrero de 2023

Agregar documentos de SSM a favoritos	Para ayudarlo a encontrar los documentos de SSM de uso frecuente, ahora puede agregar documentos a sus favoritos. Puede marcar como favoritos hasta 20 documentos por tipo de documento, por Cuenta de AWS y Región de AWS. Puede elegir, modificar y ver sus favoritos desde la consola de documentos de Systems Manager. Para obtener más información, consulte la sección Adición de documentos a favoritos.	7 de febrero de 2023
Implemente controles de cambio para la automatización mediante Change Calendar	Al integrar Automatización con Change Calendar, ahora puede implementar controles de cambios en todas las automatizaciones de su Cuenta de AWS. Para obtener más información, consulte Implementación de controles de cambios para la automatización.	24 de enero de 2023

Nuevo flujo de trabajo de aprobación de Change Manager	El flujo de trabajo de aprobación de Change Manager ahora admite aprobaciones por nivel en lugar de aprobaciones por línea. Anteriormente, todos los aprobadores que agregaba a un nivel de aprobación tenían que aprobar una solicitud de cambio. De lo contrario, el nivel no se aprobó. Ahora, usted especifica cuántas aprobaciones se requieren para el nivel y puede agregar esa cantidad o más aprobadores. Por ejemplo, puede requerir tres aprobaciones para un nivel, pero especificar hasta cinco aprobadores. Las aprobaciones de tres de esos aprobadores son suficientes para aprobar el nivel. Para obtener más información, consulte Acerca de las aprobaciones en las plantillas de cambios.	23 de enero de 2023

Nuevo: configure la aplicación de revisiones para toda la organización mediante una política de revisiones en Quick Setup	Con Quick Setup, una herramienta de Systems Manager, ahora puede crear políticas de revisiones con la tecnología de Patch Manager. Una política de revisiones define la programación y la línea de base de revisiones que se utilizarán al aplicar revisiones automáticamente a los nodos administrados. Con una configuración de política de revisiones única, puede definir la aplicación de revisiones para todas las cuentas de todas las regiones de su organización, solo para las cuentas y regiones que elija o para un solo par de cuentas y regiones. Para obtener más información, consulte los siguientes temas. • Uso de políticas de revisiones de Quick Setup • Automatice la aplicación de revisiones en toda la organización mediante una política de revisiones de Quick Setup	22 de diciembre de 2022

Application Manager se integra con Amazon EC2 para mostrar información sobre sus instancias en el contexto de una aplicación.	Application Manager muestra el estado de la instancia y el estado de Amazon EC2 Auto Scaling de una aplicación seleccionada en formato gráfico. La pestaña Instances (Instancias) también incluye una tabla con la siguiente información para cada instancia de la aplicación.
	• Estado de la instancia (pendiente, deteniéndose, en ejecución, detenida)
	• Estado de ping para SSM Agent
	• Estado y nombre del último manual de procedimientos de Automatización de Systems Manager procesado en la instancia
	• Un recuento de las alarmas de Registros de Amazon CloudWatch por estado.
	  • `ALARM`: la métrica o expresión está fuera del umbral definido.
	  • `OK`: la métrica o expresión está dentro del umbral definido.
	  • `INSUFFICIENT_DATA` : la alarma acaba de iniciarse, la métrica no está disponible o no hay suficientes datos disponibl
		22 de diciembre de 2022

	es en la métrica para determinar el estado de la alarma. • Estado de los grupos de escalado automático principal e individual	
Programar la detención y el inicio de las instancias de Amazon EC2 mediante Quick Setup	Ahora puede implementar la solución Programador de recursos para automatizar el inicio y la detención de sus instancias de Amazon EC2 mediante Quick Setup. Para obtener más información, consulte Programación de recursos de instancia de Amazon EC2.	19 de diciembre de 2022

OpsCenter ahora permite trabajar con OpsItems en varias cuentas	OpsCenter admite su uso con OpsItems desde una cuenta de administración (ya sea una cuenta de administración de AWS Organizations o una cuenta de administrador delegado de Systems Manager) y una cuenta de miembro durante una sesión. Una vez configurados, los usuarios pueden realizar los siguientes tipos de acciones: • Crear, ver y actualizar OpsItems en una cuenta de miembro • Vea información detallada sobre los recursos de AWS especificados en OpsItems en una cuenta de miembro • Iniciar manuales de procedimientos de Automatización de Systems Manager para solucionar problemas con los recursos de AWS en una cuenta de miembro Para obtener más información, consulte Configuración de OpsCenter para trabajar con OpsItems en varias cuentas.	16 de noviembre de 2022

Realice un seguimiento de los detalles de las solicitudes de cambio de Change Manager con AWS CloudTrail Lake	Ahora puede usar un almacén de datos de eventos en AWS CloudTrail Lake para recopilar y revisar los detalles de las solicitudes de cambio que se ejecutan en Change Manager para su organizac ión o cuenta. Esta información incluye detalles que se pueden auditar sobre la identidad del usuario que creó la solicitud de cambio, la dirección IP desde la que se realizó la solicitud, las Regiones de AWS en las que se realizaro n los cambios, los recursos de destino y más. Para obtener más información, consulte Supervisión de los eventos de las solicitudes de cambio y Revisión de los detalles, las tareas y los plazos de las solicitudes de cambio.	11 de noviembre de 2022

Controles de tareas adicionales de Automatización de Systems Manager mediante alarmas de CloudWatch	Ahora puede implementar un control adicional al ejecutar automatizaciones en varias cuentas y regiones mediante las alarmas de CloudWatch. Al aplicar una métrica o alarma compuesta de CloudWatch a una automatización, puede controlar el momento de su detención en función de las métricas que defina. Para obtener más información sobre cómo aplicar una alarma de CloudWatch a una automatización que se ejecuta en varias cuentas y regiones, consulte Ejecución de una automatización en varias regiones y cuentas (consola).	9 de noviembre de 2022

Actualizado: “Uso de parámetros de Parameter Store en funciones AWS Lambda”	Hemos proporcionado información adicional para ayudarlo a utilizar la extensión Lambda deAWS parámetro s y secretos para recuperar los valores de los parámetro s y almacenarlos en caché para usarlos en el futuro en las funciones de Lambda. El uso de la extensión Lambda puede reducir sus costos al reducir la cantidad de llamadas a la API a Parameter Store. Para obtener más información, consulte Uso de parámetro s de Parameter Store en funciones AWS Lambda.	25 de octubre de 2022

Controles de tareas adicionales de Systems Manager mediante alarmas de CloudWatch	Ahora puede implementar un control adicional al ejecutar automatizaciones y comandos mediante las alarmas de CloudWatch. También se puede agregar una alarma de CloudWatch a una automatización o comando cuando se registra en una tarea de periodo de mantenimiento o asociación de State Manager. Al aplicar una alarma compuesta de CloudWatch a una automatización o un comando, puede controlar el momento de su detención en función de la métrica que defina. Para obtener más información sobre cómo aplicar una alarma de CloudWatch a una automatización o un comando, consulte los siguientes procedimientos: • Cómo se seleccionan las revisiones de seguridad • Cómo se instalan las revisiones • Funcionamiento de las reglas de línea de base de revisiones en Amazon Linux 1, Amazon Linux 2 y Amazon Linux 2022.	26 de septiembre de 2022

Controles de tareas adicionales de Systems Manager mediante alarmas de CloudWatch	Ahora puede implementar un control adicional al ejecutar automatizaciones y comandos mediante las alarmas de CloudWatch. También se puede agregar una alarma de CloudWatch a una automatización o comando cuando se registra en una tarea de periodo de mantenimiento o asociación de State Manager. Al aplicar una alarma compuesta de CloudWatch a una automatización o un comando, puede controlar el momento de su detención en función de la métrica que defina. Para obtener más información sobre cómo aplicar una alarma de CloudWatch a una automatización o un comando, consulte los siguientes procedimientos: • Ejecución de una automatización sencilla • Ejecución de comandos desde la consola • Creación de una asociación • Asignación de tareas a un periodo de mantenimiento	26 de septiembre de 2022

Clarificación de los requisitos del nivel de instancias avanzadas	Basándonos en los comentarios de los clientes, hemos aclarado los escenarios que requieren que active el nivel de instancias avanzadas en Configuring instance tiers (Configuración de los niveles de instancias).	21 de septiembre de 2022
Implemente el agente de Amazon CloudWatch mediante Quick Setup	Puede implementar el agente de Amazon CloudWatch mediante Quick Setup. Para obtener más información, consulte Implementación de paquetes de Distributor con Quick Setup.	20 de septiembre de 2022

La clave “PatchGroup” ahora es compatible con los grupos de revisiones al permitir los metadatos de la instancia de EC2	Cuando permita las etiquetas en los metadatos de las instancias de EC2, las claves de etiqueta que cree no deben contener espacios. Anteriormente, esto impedía que los clientes agregaran algunas de sus instancias de EC2 a los grupos de revisiones en Patch Manager porque la clave de etiqueta `Patch Group` tenía que aplicarse a las instancias. Patch Manager ahora admite `Patch Group` (con un espacio) y `PatchGroup` (sin espacio) como clave de etiqueta para identificar las instancias de un grupo de revisiones. Las instancias de EC2 en las que se permiten etiquetas en los metadatos de las instancias ahora se pueden agregar a los grupos de revisiones en Patch Manager. Para obtener información, consulte Acerca de los grupos de revisiones.	31 de agosto de 2022

Nuevo tema: “Cómo se calculan las fechas de lanzamiento y actualización de los paquetes”	En las líneas de base de revisiones administradas por AWS, las nuevas revisiones se aprueban automáticamente 7 días después de su lanzamiento o actualización. En las líneas de base de revisiones personalizadas que cree, puede especificar de forma opcional cuántos días se debe esperar después de su lanzamiento o actualización para aprobar automáticamente su instalación. En el caso de Amazon Linux 1 y Amazon Linux 2, varios factores influyen en la forma en que se calculan las fechas de lanzamiento y actualización más recientes. Para ayudarle a evitar resultados inesperados al elegir el tiempo de espera hasta aprobación automática, estos factores se explican en el tema Cómo se calculan las fechas de lanzamiento y actualización de los paquetes.	24 de agosto de 2022

Contenido actualizado: aplicación de revisiones a una AMI y actualización de un grupo de escalado automático	Hemos actualizado el tutorial Actualización de AMIs para grupos de escalado automático para utilizar plantillas de lanzamiento en lugar de configuraciones de lanzamiento. Además, hemos implementado las últimas acciones de automatización y tiempos de ejecución en el contenido del manual de procedimientos.	22 de junio de 2022
Change Manager: impedir que los usuarios creen solicitudes de aprobación automática	Puede configurar plantillas de cambio en Change Manager para admitir aprobaciones automáticas, lo que significa que los usuarios con los permisos de IAM necesarios pueden elegir iniciar la solicitud de cambio sin necesidad de una aprobación adicional. Ahora también puede impedir que usuarios individuales, grupos o roles de IAM envíen solicitudes de aprobación automática, incluso si una plantilla de cambio las admite. Esto se consigue mediante el uso de una nueva clave de condición de IAM, `ssm:AutoApprove`. Para obtener más información, consulte Control del acceso a flujos de trabajo de manual de procedimientos de aprobación automática	15 de junio de 2022

Orientación actualizada para los roles de tareas de periodo de mantenimiento	Antes, la consola de Systems Manager ofrecía la posibilidad de elegir el rol vinculado a servicio de IAM `AWSServiceRoleForAmazonSSM` administrado de AWS que utilizar como rol de mantenimiento para las tareas. Ya no se recomienda utilizar este rol y su política asociada, `AmazonSSMServiceRolePolicy` , para tareas de periodo de mantenimiento. En su lugar, debe crear una política y un rol personalizados para las tareas de periodo de mantenimiento. Para obtener más información, consulte Configuración de Maintenance Windows.	9 de junio de 2022
Compatibilidad con reenvío de puertos a hosts remotos para Session Manager	Session Manager ahora admite sesiones de reenvío de puertos a hosts remotos. No se requiere que Systems Manager administre el host remoto. Para obtener más información, consulte Inicio de una sesión (reenvío de puertos a host remoto).	25 de mayo de 2022

Contenido actualizado: instrucciones para instalar manualmente SSM Agent en instancias Linux de Amazon EC2	En respuesta a los comentarios de los clientes, hemos revisado los temas que proporcionan instrucciones para instalar manualmente SSM Agent en instancias de Amazon EC2. En estos temas ahora se proporcionan comandos que utilizan archivos disponibles globalmente que se pueden copiar y pegar para realizar una instalación rápida en instancias de EC2 en cualquier Región de AWS. Además, en estos temas también se proporciona información para ayudar a crear comandos de instalación que utilicen archivos disponibles en su propia región de trabajo. Este último enfoque se recomienda cuando se va a instalar el agente en varias instancias mediante un script o una plantilla. Para obtener más información, consulte las instrucciones correspondientes al sistema operativo Linux en la sección Instalación manual de SSM Agent en instancias de EC2 para Linux.	9 de mayo de 2022

Nuevo tema: Amazon Machine Images (AMIs) con SSM Agent preinstalado	En respuesta a los comentarios de los clientes, hemos centralizado la información sobre qué AMIs administradas de AWS tienen SSM Agent preinstalado. En este tema también se proporcionan instrucciones sobre cómo verificar que una instancia de Amazon EC2 creada a partir de estas AMIs se ha instalado correctamente y se está ejecutando. Para los casos excepcionales en los que el agente no se instale correctamente, o se instale pero no se inicie, también proporcionamos información sobre cómo iniciar o instalar manualmente el agente en estas instancias. Para obtener detalles, consulte Amazon Machine Images (AMIs) con SSM Agent preinstalado.	8 de mayo de 2022
Nueva sección sobre State Manager	Se ha agregado una nueva sección que da detalles sobre cuándo State Manager ejecuta asociaciones. Para obtener más información, consulte Acerca de la programación de asociaciones.	27 de abril de 2022

Patch Manager ya admite Rocky Linux	A partir de ahora, puede utilizar Patch Manager para aplicar revisiones a los nodos de Rocky Linux. Muchas de las reglas que se utilizan con RHEL 8 para la aplicación de revisiones también se utilizan con Rocky Linux. Rocky Linux 8 usa la nueva versión de `AWS-DefaultRockyLinuxPatchBaseline` . Para obtener más información, consulte los temas siguientes: • Cómo se seleccionan las revisiones de seguridad • Cómo se instalan las revisiones • Funcionamiento de las reglas de líneas de base de revisiones en RHEL, CentOS Stream y Rocky Linux.	14 de abril de 2022

Patch Manager ahora es compatible con CentOS Stream 8	A partir de ahora, puede utilizar Patch Manager para aplicar revisiones a instancias de CentOS Stream 8 e instancias de Red Hat Enterprise Linux (RHEL) 4.4-4.5. Muchas de las reglas que se utilizan con RHEL 8 para la aplicación de revisiones también se utilizan con CentOS Stream 8. CentOS Stream 8 usa `AWS-DefaultCentOSPatchBaseline` . Para obtener más información, consulte los temas siguientes: • Cómo se seleccionan las revisiones de seguridad • Cómo se instalan las revisiones • Funcionamiento de las reglas de líneas de base de revisiones en RHEL y CentOS Stream	4 de abril de 2022

Crear un rol de asunción para Change Manager	Una nueva sección aclara los requisitos para crear e implementar un rol de asunción para Change Manager. Un rol de asunción es un rol de servicio de AWS Identity and Access Management (IAM) que permite que Change Manager pueda ejecutar de forma segura los flujos de trabajo de manuales de procedimientos especificados en una solicitud de cambio aprobada en su nombre. El rol concede a AWS Systems Manager (AWS STS) la confianza `AssumeRole` para Change Manager. Para obtener información, consulte Configuración de roles y permisos para Change Manager.	18 de marzo de 2022
Aprobar o rechazar solicitudes de cambio de Change Manager en bloque	En la consola de Systems Manager, ahora puede seleccionar varias solicitudes de cambio para aprobarlas o rechazarlas con una sola operación. Para obtener información, consulte Revisión y aprobación o rechazo de las solicitudes de cambio (consola).	8 de marzo de 2022

Compatibilidad con nodos administrados de Rocky Linux y Windows Server 2022	Systems Manager es compatible con nodos administrados de Rocky Linux y Windows Server 2022, incluidos dispositivos perimetrales y máquinas híbridas ubicadas en las instalaciones o en otros proveedores de nube. Para utilizar Systems Manager con estos sistemas operativos, debe completar todos los procedimientos de configuración de Systems Manager necesarios, incluidos los procedimientos para entornos híbridos o dispositivos perimetrales, si procede. Para obtener más información, consulte Configuración de Systems Manager. En las máquinas con Rocky Linux también debe instalar manualmente SSM Agent. Para obtener más información, consulte Instalación manual de SSM Agent en instancias de Rocky Linux. En las instancias de Amazon Elastic Compute Cloud (Amazon EC2) de Windows Server 2022, SSM Agent se instala de manera predeterminada.	1 de marzo de 2022

Permitir que Automatización se adapte a sus necesidades de simultaneidad y ver las métricas de uso de Automatización	Ahora puede permitir que Automation ajuste automáticamente la cuota de automatización simultánea y puede ver las métricas de uso de Automation publicadas en CloudWatch. Para obtener más información acerca de la simultaneidad adaptativa, consulte Permitir que Automation se adapte a sus necesidades de simultaneidad. Para obtener más información acerca de cómo ver las métricas de uso de Automation, consulte Monitoring Automation metrics using Amazon CloudWatch.	27 de enero de 2022

Permitir que Automatización se adapte a sus necesidades de simultaneidad y ver las métricas de uso de Automatización	Ahora puede permitir que Automation ajuste automáticamente la cuota de automatización simultánea y puede ver las métricas de uso de Automation publicadas en CloudWatch. Para obtener más información acerca de la simultaneidad adaptativa, consulte Permitir que Automation se adapte a sus necesidades de simultaneidad. Para obtener más información acerca de cómo ver las métricas de uso de Automation, consulte Monitoring Automation metrics using Amazon CloudWatch.	27 de enero de 2022
Documentos de Systems Manager organizados por categorías	Los documentos de Systems Manager propiedad de Amazon ahora se organizan por tipo y categorías para ayudarlo a encontrar los documentos que necesita.	13 de enero de 2022

Crear e invocar integraciones para Automation	Ahora puede enviar mensajes mediante webhooks durante una automatización al crear una integración. Las integraciones se pueden invocar durante una automatización mediante la nueva acción `aws:invokeWebhook` de su manual de procedimientos. Para obtener más información acerca de la creación de integraciones, consulte Creación de integraciones webhook para Automation. Para obtener más información acerca de la acción `aws:invokeWebhook` , consulte aws:invokeWebhook: invocar una integración de webhook de Automation.	13 de enero de 2022
Capacidades no disponibles en las nuevas Región de AWS	Las siguientes herramientas de Systems Manager no están disponibles actualmente en la nueva región Asia-Pacífico (Yakarta). • Application Manager • Change Calendar • Change Manager • Explorer • Fleet Manager • Administrador de incidentes de • Quick Setup	13 de diciembre de 2021

Ver detalles de costos de recursos de una aplicación	Application Manager está integrado con Administración de facturación y costos de AWS a través del widget Cost Explorer. Después de habilitar Cost Explorer en la consola Billing and Cost Managemen t, el widget Cost Explorer en Application Manager muestra los datos de costos de una aplicación o componente de aplicación específicos que no están en contenedores. Puede utilizar filtros en el widget para ver los datos de costos según diferentes periodos de tiempo, detalles y tipos de costos en un gráfico de barras o líneas. Para obtener más informaci ón, consulte Visualización de información general acerca de una aplicación.	7 de diciembre de 2021
Administración de procesos mediante Fleet Manager	Ahora puede utilizar Fleet Manager para administrar los procesos en los nodos. Para obtener más informaci ón, consulte Trabajo con procesos.	6 de diciembre de 2021

Cambio de terminología: las instancias administradas ahora son nodos administrados	Al ser compatible con dispositivos de núcleo de AWS IoT Greengrass, la frase instancia administrada se ha cambiado por nodo administrado en la mayoría de la documentación de Systems Manager. La consola de Systems Manager, las llamadas a API, los mensajes de error y los documentos SSM continúan utilizando el término instancia.	29 de noviembre de 2021

Compatibilidad con dispositivos de borde	Systems Manager admite las siguientes configuraciones de dispositivos de borde. • AWS IoT Greengrass: Systems Manager ahora admite cualquier dispositivo configurado para AWS IoT Greengrass y ejecuta el software AWS IoT Greengrass Core. Para incorporar los dispositivos de núcleo de AWS IoT Greengrass, debe crear un rol de servicio de AWS Identity and Access Management (IAM). También debe utilizar la consola de AWS IoT Greengrass para implementar SSM Agent como un componente de AWS IoT Greengrass en los dispositivos. Para obtener más información, consulte Configuración de AWS Systems Manager en dispositivos de borde. • Dispositivos de borde en un entorno híbrido: Systems Manager también admite dispositivos de núcleo de AWS IoT y dispositivos IoT que no son de AWSdespués de configurarlos como máquinas locales. Para	29 de noviembre de 2021

	incorporar los dispositivos, debe crear un rol de servicio de IAM, crear una activación de nodo administrado para un entorno híbrido e instalar manualmente SSM Agent en los dispositivos. Para obtener más información, consulte Configuración de AWS Systems Manager para entornos híbridos	
Conexión a instancias administradas mediante Escritorio remoto	Ahora puede utilizar Fleet Manager para conectarse a instancias de Windows administradas mediante el protocolo de escritorio remoto (RDP). Estas sesiones de Escritorio remoto con tecnología de Amazon DCV proporcionan conexiones seguras a las instancias directamente desde el navegador. Para obtener más información, consulte Conexión mediante Escritorio remoto.	23 de noviembre de 2021

Especifique la duración máxima de la sesión y proporcione los motivos de las sesiones	Ahora puede especificar la duración máxima de la sesión para todas las sesiones de Session Manager de una Región de AWS en la Cuenta de AWS. Cuando una sesión alcanza la duración especific ada, finaliza. Ahora también puede agregar opcionalmente un motivo al iniciar una sesión. Para obtener más información, consulte Especificación de la duración máxima de la sesión.	16 de noviembre de 2021

Patch Manager ahora es compatible con el sistema operativo Raspberry Pi OS	Ahora puede utilizar Patch Manager para aplicar revisiones a instancias de Raspberry Pi OS. Patch Manager admite la aplicación de revisiones a Raspberry Pi OS 9 (Stretch) y 10 (Buster). Dado que Raspberry Pi OS es un sistema operativo basado en Debian, se aplican muchas de las mismas reglas de revisiones que en Debian Server. Para obtener más información, consulte los temas siguientes: • Cómo se seleccionan las revisiones de seguridad • Cómo se instalan las revisiones • Funcionamiento de las reglas de bases de referencia de parches en Debian Server y Raspberry Pi OS	16 de noviembre de 2021
Acceso al portal de la base de conocimientos de Red Hat	Utilice Fleet Manager para acceder al portal de la base de conocimientos de RHEL para encontrar soluciones, artículos, documentación y videos sobre el uso de productos Red Hat. Para obtener más información, consulte Acceso al portal de la base de conocimientos de Red Hat.	3 de noviembre de 2021

Edición múltiple de OpsItems	OpsCenter ahora admite la edición múltiple de OpsItems. Puede seleccionar varios OpsItems y editar uno de los siguientes campos: Status (Estado), Priority (Prioridad), Severity (Severidad), Category (Categoría). Para obtener más información, consulte Edición de un OpsItems.	15 de octubre de 2021
Creación de parámetros de entrada que rellenas recursos de AWS	Ahora puede crear parámetros de entrada en los manuales de procedimiento de Automation que rellenan los recursos de AWS en la AWS Managemen t Console. Para obtener información, consulte Creación de parámetros de entrada que rellenan recursos de AWS.	14 de octubre de 2021
Nueva opción de detención de invocación de tareas para ventanas de mantenimiento	Ahora puede optar por bloquear el inicio de cualquier nueva invocación de tareas después de que se haya alcanzado el tiempo límite especificado para un período de mantenimiento. Para obtener información, consulte Asignar tareas a un período de mantenimiento (consola).	13 de octubre de 2021

Soporte de Patch Manager para macOS 11.3.1 y 11.4 (Big Sur)macOS	Las instancias de Amazon Elastic Compute Cloud (Amazon EC2) para macOS 11.3.1 y 11.4 (Big Sur) ahora se pueden revisar mediante Patch Manager. Esto se suma al soporte existente para macOS 10.14.x (Mojave) y 10.15.x (Catalina). Para obtener información acerca de cómo trabajar con Patch Manager, consulte AWS Systems ManagerPatch Manager.	1 de octubre de 2021

Application Insights en Application Manager	Application Manager se integra con Amazon CloudWatch Application Insights. Application Insights identifica y configura las métricas clave, los registros y las alarmas entre los recursos de aplicaciones y la pila de tecnología. Application Insights monitorea continuamente las métricas y los registros para detectar y relacionar anomalías y errores. Cuando el sistema detecta errores y anomalías , Application Insights genera CloudWatch Events que puede utilizar para configura r notificaciones o tomar medidas. Puede habilitar y ver Application Insights en las pestañas Overview (Informac ión general) y Monitorin g (Monitoreo) en Applicati on Manager. Para obtener más información acerca de Application Insights, consulte ¿Qué es Información de aplicaciones de Amazon CloudWatch? en la Guía del usuario de Amazon CloudWatch.	21 de septiembre de 2021

Importación de eventos de otros calendarios al Change Calendar	Ahora puedes importar los eventos de un calendario de terceros a un calendari o en Change Calendar. Anteriormente, cada evento debía introducirse de manera manual en un calendario. Después de exportar un calendario de un proveedor de calendario de terceros compatible a un archivo iCalendar (`.ics`), impórtelo en Change Calendar, y los eventos se incluirán en las reglas del calendario abierto o cerrado en Systems Manager. Los proveedores compatibl es incluyen iCloud Calendar, Google Calendar y Microsoft Outlook. Para obtener más información, consulte Importación y administración de eventos desde calendarios de terceros.	8 de septiembre de 2021

Nuevas características de etiquetado y manual de procedimientos en Application Manager	Las mejoras de etiquetado incluyen la posibilidad de agregar etiquetas o eliminarlas de un recurso específico o de todos los recursos en una aplicación Application Manager. Las mejoras del manual de procedimientos incluyen la capacidad de ver una lista filtrada de manuales de procedimientos de un tipo de recurso específico o iniciar un manual de procedimientos en todos los recursos del mismo tipo. Para obtener más información, consulte Trabajo con etiquetas en Application Manager y Trabajo con manual de procedimientos en Application Manager.	31 de agosto de 2021
Ejemplo nuevo: creación de una solicitud de cambio mediante la AWS CLI	Se agregó un ejemplo de creación de una solicitud de cambio con la AWS CLI al capítulo Change Manager. En el ejemplo se utiliza la plantilla de cambio de muestra `AWS-HelloWorldChangeTemplate` y `AWS-HelloWorld runbook` : • Creating change requests (AWS CLI) (Creación de solicitudes de cambio [CLI])	20 de agosto de 2021

Sección nueva: Uso de parámetros en Amazon EKS	Se ha agregado una sección nueva al capítulo Parameter Store. Este tema es una explicación sobre cómo se utilizan los parámetros en clústeres de Amazon EKS. Para obtener más información, consulte Uso de parámetros del Parameter Store en Amazon Elastic Kubernetes Service.	19 de agosto de 2021
Se han actualizado los enlaces de ciclo de vida de Patch Manager	Patch Manager ahora proporciona un enlace de ciclo de vida (la capacidad de ejecutar un documento de Systems Manager Command) para un punto adicional durante una operación de revisión Patch now. Si programa reinicios para una instancia después de ejecutar Aplicar revisiones ahora, puede especificar un enlace de ciclo de vida que se ejecutará una vez finalizado cada reinicio. Para obtener más información, consulte Uso de los enlaces de ciclo de vida 'Patch now' y Acerca del documento de `AWS-RunPatchBaselineWithHooks` SSM.	9 de agosto de 2021

Las aprobaciones automáticas ahora son compatibles para las solicitudes de Change Manager	Ahora, podrá configurar plantillas de cambio en Change Manager para admitir aprobaciones automáticas, lo que significa que los usuarios con los permisos de IAM necesarios pueden elegir iniciar la solicitud de cambio sin necesidad de una aprobación adicional. Los usuarios que tengan acceso a plantillas de aprobación automática pueden elegir especificar aprobadores si lo desean. Para ayudarlo a controlar sus procesos de Change Manager, las aprobaciones siguen siendo necesarias para todas las solicitudes durante los periodos de congelación de cambios. Para obtener más información, consulte los temas siguientes: • Creación de plantillas de cambio • Creación de solicitudes de cambio • Probar la plantilla de cambio de `Hello World` administrada de AWS	30 de julio de 2021

Análisis operacional de OpsCenter	OpsCenter analiza automáticamente OpsItems en su cuenta y genera información. La información lo ayuda a entender cuántos OpsItems duplicados hay en su cuenta y qué fuentes los están creando. La información también proporciona prácticas recomendadas y manuales de procedimientos de Automation recomendados para ayudarlo a resolver los OpsItems duplicados. Para obtener más información, consulte Uso de información operativa.	13 de julio de 2021
Ver instancias detenidas en Fleet Manager	Ahora puede ver qué instancias están `running` (ejecutándose) y qué instancias están `stopped` (detenidas) desde la consola de Fleet Manager. Para obtener más información, consulte AWS Systems ManagerFleet Manager.	12 de julio de 2021
Nuevo tema: Creación de manuales de procedimientos de Automation	Un nuevo tema, Creación de manuales de procedimientos de Automation, proporciona una guía y ejemplos narrativos sobre cómo crear contenido para manuales de procedimientos personalizados de Automation.	8 de julio de 2021

Creación de plantillas y pilas de AWS CloudFormation en Application Manager	Application Manager lo ayuda a aprovisionar y administrar recursos para sus aplicaciones mediante la integración con CloudFormation. Puede crear, editar y eliminar plantillas y pilas de AWS CloudFormation en Application Manager. Application Manager también incluye una biblioteca de plantillas donde puede clonar, crear y almacenar plantillas. Application Manager y CloudFormation muestran la misma información sobre el estado actual de una pila. Las plantillas y las actualizaciones de plantillas se almacenan en Systems Manager hasta que aprovisione la pila, momento en el que los cambios también se muestran en CloudFormation. Para obtener más información, consulte Trabajar con pilas de AWS CloudFormation en Application Manager.	8 de julio de 2021

Tema nuevo: Rotación automática de claves privadas para instancias híbridas de SSM Agent	Un tema de nuevo, Setting up private key auto rotation (Configuración de la rotación automática de clave privada), proporciona instrucciones sobre cómo se fortalece la posición de seguridad mediante la configuración de SSM Agent para rotar automáticamente la clave privada del entorno híbrido.	15 de junio de 2021
Complemento de Session Manager para la versión 1.2.205.0 de la AWS CLI	Se lanzó una nueva versión del complemento Session Manager para la AWS CLI. Para obtener más informaci ón, consulte Versión más reciente del complemento Session Manager e historial de publicaciones.	10 de junio de 2021

Nuevo rol vinculado a servicio de IAM	Cuando habilita información operativa de OpsCenter, Systems Manager crea un nuevo rol vinculado al servicio de AWS Identity and Access Management (IAM) denominado `AWSSSMOpsInsightsServiceRolePolicy`. Para obtener más información acerca de este rol, consulte Uso de roles para crear información operativa de OpsItems en Systems Manager OpsCenter: `AWSSSMOpsInsightsServiceRolePolicy`.	9 de junio de 2021
Nuevo contenido de solución de problemas de Patch Manager para Linux	Un nuevo tema, Errores al momento de la ejecución de `AWS-RunPatchBaseline` en Linux, proporciona descripciones y soluciones para varios problemas que pueden surgir cuando aplique revisiones a instancias administradas con sistemas operativos Linux.	8 de junio de 2021

Compatibilidad mejorada para tareas de periodo de mantenimiento que no requieren destinos especificados (consola)	Ahora puede crear tareas de periodo de mantenimiento en la consola sin tener que especificar un destino en la tarea si no es necesario. Anteriormente, esta opción solo estaba disponible cuando utilizaba la AWS CLI o la API. Esta opción se aplica a Automation , a AWS Lambda, y a tipos de tareas AWS Step Functions. Por ejemplo, si crea una tarea de automatización y los recursos que se van a actualizar se especifican en los parámetros del documento de Automation, ya no es necesario especificar un destino en esa tarea. Para obtener más información, consulte Registro de tareas de periodo de mantenimiento sin destinos, Asignación de tareas a un periodo de mantenimiento (consola) y Programación de automatizaciones con periodos de mantenimiento.	28 de mayo de 2021
Referencia reubicada del manual de procedimientos de Automatización	La referencia del manual de procedimientos de Automatización se ha trasladado a una nueva ubicación. Para obtener más información, consulte Referencia del manual de procedimientos de Automatización de Systems Manager.	10 de mayo de 2021

Lanzamiento de Administrador de incidentes de AWS Systems Manager	Incident Manager es una consola de administración de incidentes diseñada para ayudar a los usuarios a mitigar y recuperarse de incidentes que afectan a sus aplicaciones alojadas en AWS. Para obtener más información, consulte la Guía del usuario de Administrador de incidentes de AWS Systems Manager.	10 de mayo de 2021
State Manager es compatible con Change Calendar	Ahora puede especificar el nombre de Change Calendar o los Nombres de recurso de Amazon (ARN) cuando cree o actualice una asociación de State Manager. State Manager aplica asociaciones solo cuando el calendario de cambios está abierto, no cuando está cerrado. Para obtener más información, consulte Creación de asociaciones y Edición y creación de una nueva versión de una asociación.	6 de mayo de 2021

Clonar documentos de Systems Manager	Con la consola de documentos de Systems Manager, ahora puede copiar contenido desde un documento existente a un documento nuevo que pueda modificar. Para obtener más información, consulte Clonación de un documento de SSM.	4 de mayo de 2021
Integrar Security Hub con Explorer y OpsCenter	A partir de ahora, puede integrar Explorer y OpsCenter con AWS Security Hub. Security Hub le proporciona una visión completa de su estado de seguridad en AWS y lo ayuda a verificar su entorno con los estándares y las prácticas recomendadas del sector de la seguridad. Cuando se integra con Explorer, puede ver los resultados de seguridad en el widget de Security Hub en el panel de Explorer. Cuando se integra con OpsCenter, puede crear OpsItems para obtener los resultados de Security Hub. Para obtener más información, consulte Recibir resultados de AWS Security Hub enExplorer y Recibir resultados de AWS Security Hub en OpsCenter.	27 de abril de 2021

Nuevo tema: Convenciones del documento	Hemos agregado un nuevo tema para que los usuarios puedan consultar las convenciones tipográficas comunes de la Guía del usuario de AWS Systems Manager. Para obtener más información, consulte Convenciones del documento.	21 de abril de 2021
Tema actualizado: Acerca del uso de revisiones en aplicaciones publicadas por Microsoft en Windows Server	El tema Acerca de la aplicación de revisiones en las aplicaciones publicadas por Microsoft en Windows Server ahora aclara que, para que Patch Manager pueda aplicar revisiones a las aplicaciones publicadas por Microsoft en las instancias administradas de Windows Server, debe estar permitida en ellas la opción de actualización de Windows Ofrecerme actualizaciones para otros productos de Microsoft cuando actualice Windows.	12 de abril de 2021

Reorganización de la referencia del manual de procedimientos de Automatización	Para ayudarlo a encontrar los manuales de procedimientos que necesita y navegar por la referencia de manera más eficiente, reorganizamos el contenido en la referencia del manual de procedimientos de Automatización por el Servicio de AWS pertinente. Para ver estos cambios, consulte Referencia del manual de procedimientos de Automatización de Systems Manager.	12 de abril de 2021

Patch Manager: genere informes de conformidad de revisiones .csv	Patch Manager ahora admite la capacidad de generar informes de cumplimiento de revisiones para sus instancias y guardar el informe en un bucket de S3 de su elección, en formato .csv. Luego, con una herramienta como Amazon QuickSight, puede analizar los datos del informe de conformidad de revisiones. Puede generar un informe de conformidad de revisiones para una sola instancia o para todas las instancias de su Cuenta de AWS. Puede generar un informe único en diferido o configurar una programación para que los informes se creen automáticamente. También puede especificar un tema de Amazon Simple Notification Service para proporcionar notificaciones cuando se genera un informe. Para obtener más información, consulte Generación de informes de conformidad de revisiones con CSV.	9 de abril de 2021

Eliminación de etiquetas de parámetros de Parameter Store	Ahora ya puede eliminar etiquetas de parámetros Parameter Store mediante la consola de Systems Manager o la AWS CLI. Para obtener más información, consulte Trabajo con etiquetas de parámetros.	6 de abril de 2021
Programar reinicios de instancia al usar Patch Now (Aplicar revisión ahora)	Patch Manager ahora admite programar un tiempo para que las instancias se reinicien después de instalar las revisiones mediante la característica “Patch Now” (Aplicar revisión ahora). Esto se suma a las opciones existentes para reiniciar instancias solo si es necesario para completar una instalación de la revisión o para omitir todo reinicio después de la operación de revisión. Para obtener información, consulte Aplicación de revisiones a instancias en diferido.	1 de abril de 2021
Nuevo tema: Búsqueda de parámetros públicos	Los parámetros públicos de Parameter Store ahora se pueden encontrar mediante la AWS CLI o la consola de Systems Manager. Para obtener más información, consulte Buscar parámetros públicos.	1 de abril de 2021

Actualizaciones de la característica “Patch Now” (Aplicar revisión ahora): Almacene los registros en S3 y ejecute los enlaces del ciclo de vida	Cuando ejecute la operación de Patch Manager Patch Now (Aplicar revisión ahora), puede elegir un bucket de S3 en el que almacenar automáticamente los registros de revisiones. Además, puede elegir ejecutar documentos de Systems Manager Command (documentos de SSM) como enlaces del ciclo de vida en tres puntos durante la operación: Antes de la instalación, Después de la instalación, y A la salida. Para obtener información, consulte Aplicación de revisiones a instancias en diferido.	31 de marzo de 2021
Systems Manager ahora informa de los cambios en sus políticas administradas por AWS	A partir del 24 de marzo de 2021, los cambios en las políticas administradas se informan en el tema actualizaciones de Systems Manager a políticas administradas de AWS. El primer cambio enumerado es la adición de compatibilidad con la herramienta Explorer para informar de OpsData y OpsItems de varias cuentas y regiones.	24 de marzo de 2021

Explorer permite automáticamente todas las fuentes OpsData para sincronizar datos de recursos en función de las cuentas en AWS Organizations	Cuando cree una sincronización de datos de recursos, si elige una de las opciones de AWS Organizations, Systems Manager permite de forma automática todas las fuentes de OpsData en las Regiones de AWS seleccionadas para todas las Cuentas de AWS en su organización (o en las unidades organizativas seleccionadas). Esto significa, por ejemplo, incluso si no ha activado Explorer en una Región de AWS, si selecciona una opción de AWS Organizations para la sincronización de datos de recursos, entonces, Systems Manager recopilará de forma automática OpsData de esa región. Para obtener más información, consulte Acerca de las sincronizaciones de datos de recursos de varias cuentas y regiones.	24 de marzo de 2021
La capacidad de Automatización de Systems Manager proporciona una nueva variable de sistema para sus manuales de procedimientos	Con la nueva variable de sistema `global:AWS_PARTITION`, puede especificar la partición de AWS en la que se encuentra un recurso cuando cree sus manuales de procedimientos. Para obtener más información, consulte Variables de sistemas de Automation.	18 de marzo de 2021

Permiso de varios niveles de aprobación para solicitudes de cambio de Change Manager	Cuando cree una plantilla de cambios de Change Manager, ahora puede requerir que más de un nivel de aprobadores conceda permiso para que se ejecute una solicitud de cambio. Por ejemplo, puede requerir que los revisores técnicos aprueben primero una solicitud de cambio creada a partir de una plantilla de cambios y, a continuación, requieran un segundo nivel de aprobaciones por parte de uno o más administradores. Para obtener más información, consulte Creación de plantillas de cambios.	4 de marzo de 2021
Patch Manager ahora es compatible con Oracle Linux 8.x	A partir de ahora, puede utilizar Patch Manager para aplicar revisiones a instancia s de Oracle Linux 8.x, a través de la versión 8.3. Para obtener más información, consulte los temas siguientes: • Cómo se seleccionan las revisiones de seguridad • Cómo se instalan las revisiones • Funcionamiento de las reglas de líneas de base de revisiones en Oracle Linux	1 de marzo de 2021

OpsCenter muestra otros OpsItems para un recurso seleccionado	Para ayudarlo a investigar problemas y proporcionar contexto para un problema, puede ver una lista de OpsItems para un recurso específico de AWS. La lista muestra el estado, la severidad y el título de cada OpsItem. La lista también incluye enlaces profundos a cada OpsItem. Para obtener más información, consulte Visualización de otros OpsItems para un recurso específico..	1 de marzo de 2021
Definir preferencias de aplicación de revisiones en tiempo de ejecución	Ahora puede definir las preferencias de revisiones en tiempo de ejecución mediante la característica de anulación de referencia. Para obtener más información, consulte Uso del parámetro BaselineOverride.	25 de febrero de 2021

Nuevo tipo de documento de Systems Manager	Las plantillas de AWS CloudFormation ahora se pueden almacenar como documentos de Systems Manager. El almacenamiento de plantillas de CloudFormation como documentos de Systems Manager le permite beneficiarse de las características de los documentos de Systems Manager, como control de versiones, comparación del contenido entre versiones y compartir entre cuentas. Para obtener más información, consulte Documentos de AWS Systems Manager	9 de febrero de 2021

Aplicar revisiones a instancias con enlaces opcionales	Nuevo documento de SSM `AWS-RunPatchBaselineWithHooks` proporciona enlaces que puede utilizar para ejecutar documentos de SSM en tres puntos durante el ciclo de aplicación de revisiones a instancias. Para obtener información acerca de `AWS-RunPatchBaselineWithHooks`, consulte Acerca del documento de SSM `AWS-RunPatchBaselineWithHooks`. Para obtener una explicación de ejemplo de una operación de aplicación de revisiones que utiliza los tres enlaces, consulte Explicación: actualizar dependencias de aplicaciones, aplicar revisiones a una instancia y realizar una comprobación de estado específica de una aplicación.	2 de febrero de 2021

Nuevo tema: validación de servidores locales y máquinas virtuales mediante una huella digital de hardware	SSM Agent comprueba la identificación de los servidores y las máquinas virtuales locales y de las máquinas virtuales que registra en el servicio mediante una huella digital calculada. La huella digital es una cadena opaca que se conserva en el Almacén que el agente pasa a ciertas API de Systems Manager. Para obtener información acerca de la huella digital del hardware y las instrucciones para configurar un límite de similitud para ayudar en la verificación de máquinas, consulte Validación de servidores y máquinas virtuales locales mediante una huella digital de hardware.	25 de enero de 2021
Tema nuevo: referencia técnica de SSM Agent	El tema referencia técnica de SSM Agent reúne información que lo puede ayudar a implementar AWS Systems Manager SSM Agent y entender cómo funciona el agente. Este tema incluye una sección totalmente nueva, actualizaciones continuas de SSM Agent por Regiones de AWS.	21 de enero de 2021

SSM Agent en Windows Server 2008	A partir del 14 de enero de 2020, Windows Server 2008 ya no es compatible para obtener actualizaciones de características o seguridad de Microsoft. Las AMIs de Windows Server 2008 incluyen SSM Agent, pero el agente ya no se actualiza para este sistema operativo.	5 de enero de 2021
Compatibilidad mejorada para tareas de periodo de mantenimiento que no requieren destinos especificados (AWS CLI y API únicamente)	Ahora puede crear tareas de periodo de mantenimiento sin tener que especificar un destino en la tarea si no es necesario (AWS CLI y API únicamente). Esta opción se aplica a Automation y a tipos de tareas AWS Lambda y AWS Step Functions. Por ejemplo, si crea una tarea de automatización y los recursos que se van a actualizar se especifican en los parámetros del manual de procedimientos de Automation, ya no es necesario especificar un destino en esa tarea. Para obtener más información, consulte Registro de tareas de periodo de mantenimiento sin destinos y Programación de automatizaciones con periodos de mantenimiento.	23 de diciembre de 2020

Nuevas características de Automation	Se ha agregado una nueva propiedad compartida a los manuales de procedimientos de Automatización de Systems Manager. La propiedad `onCancel` le permite especificar a qué paso debe ir la automatización en caso de que un usuario la cancele. Para obtener más información, consulte Propiedades compartidas por todas las acciones.	21 de diciembre de 2020
Nuevo tema: Uso de asociaciones mediante IAM	Se agregó un tema nuevo al capítulo de State Manager de Systems Manager que describe las prácticas recomendadas para crear asociaciones mediante IAM. Para obtener más información, consulte Trabajo con asociaciones mediante IAM.	18 de diciembre de 2020
State Manager ahora es compatible con regiones y cuentas múltiples	Ahora las asociaciones se pueden crear o actualizar con varias regiones o cuentas. Para obtener más información, consulte Creación de asociaciones.	15 de diciembre de 2020

Nueva herramienta: Fleet Manager	Fleet Manager, una herramienta de AWS Systems Manager, es una experiencia de interfaz de usuario (UI) unificada que lo ayuda a administrar de forma remota la flota de servidores que se ejecuta en AWS o en las instalaciones. Con Fleet Manager, puede ver el estado y el rendimiento de toda la flota de servidores desde una sola consola. También puede recopilar datos de instancias individuales para realizar tareas comunes de solución de problemas y administración desde la consola. Para obtener información, consulte AWS Systems Manager.Fleet Manager	15 de diciembre de 2020

Nueva herramienta: Change Manager	Amazon Web Services lanzó Change Manager, un marco de administración de cambios empresariales con el que se pueden solicitar, aprobar, implementar e informar los cambios operativos en la configuración y la infraestr uctura de la aplicación. A partir de una sola cuenta de administrador delegado, si utiliza AWS Organizations, puede administrar los cambios a través de varias Cuentas de AWS en varias Regiones de AWS. De forma alternativa, a través de una cuenta local, puede administrar los cambios de una sola Cuenta de AWS. Utilice Change Manager para administrar los cambios tanto en los recursos de AWS como en los recursos locales. Para obtener información, consulte AWS Systems Manager.C hange Manager	15 de diciembre de 2020

Nueva herramienta: Application Manager	Application Manager lo ayuda a investigar y corregir problemas con sus recursos de AWS en el contexto de sus aplicaciones. Application Manager agrega información de operaciones de múltiples Servicios de AWS y las herramientas de Systems Manager a una única AWS Management Console. Para obtener información, consulte AWS Systems Manager.Application Manager	15 de diciembre de 2020

AWS Systems Manager es compatible con instancias de Amazon EC2 para macOS	Junto con la publicación de la compatibilidad de Amazon Elastic Compute Cloud (Amazon EC2) para instancias de macOS, Systems Manager ahora admite muchas operaciones en instancias de EC2 para macOS. Las versiones compatibles incluyen macOS 10.14.x (Mojave) y 10.15.x (Catalina). Para obtener más información, consulte los siguientes temas. • Para obtener información sobre la instalación de SSM Agent en instancias EC2 para macOS, consulte Uso de SSM Agent en instancias de EC2 para macOS. • Para obtener información sobre la aplicación de revisiones de instancias EC2 para macOS, consulte Cómo se instalan las revisiones y Creación de una base de referencia de parches personalizada (macOS). • Para obtener información general sobre la compatibilidad de instancias de EC2 con macOS, consulte Instancias de Mac Amazon	30 de noviembre de 2020

	EC2 en la Guía del usuario de Amazon EC2.	
Pseudoparámetros de periodo de mantenimiento: nuevo tipo de recurso compatible con {{TARGET_ID}} y {{RESOURCE_ID}}	Un tipo de recurso adicional está ahora disponible para su uso con los pseudoparámetros {{TARGET_ID}} y {{RESOURCE_ID}} . Ahora puede utilizar el tipo de recurso AWS::RDS::DBCluster con estos pseudoparámetros. Para obtener información acerca de los pseudoparámetros del periodo de mantenimiento, consulte Uso de pseudoparámetros al registrar tareas de la ventana de mantenimiento.	27 de noviembre de 2020
Complemento de Session Manager para la versión 1.2.30.0 de la AWS CLI	Se lanzó una nueva versión del complemento Session Manager para la AWS CLI. Para obtener más información, consulte Versión más reciente del complemento Session Manager e historial de publicaciones.	24 de noviembre de 2020
Nuevo tema: Comparación de versiones de documentos de SSM	Ahora puede comparar las diferencias de contenido entre las versiones de documentos de SSM en la consola de documentos de Systems Manager. Para obtener más información, consulte Comparación de versiones de documentos de SSM.	24 de noviembre de 2020

Systems Manager ahora es compatible con las políticas de punto de conexión de VPC	Ahora puede crear políticas para los puntos de enlace de interfaz de VPC para Systems Manager. Para obtener más información, consulte Creación de una política de punto de enlace de la VPC de interfaces.	18 de noviembre de 2020
Nuevo tema: Especificación de un valor de tiempo de espera de sesión inactiva	Ahora puede especificar qué cantidad de tiempo permitir a un usuario para que esté inactivo antes de que una sesión termine con Session Manager. Para obtener más información, consulte Especificar un valor de tiempo de espera de sesión inactiva.	18 de noviembre de 2020
Nueva característica de Session Manager de registro	Ahora puede enviar un flujo continuo de registros de datos de sesión con formato JSON a los Registros de Amazon CloudWatch. Para obtener más información, consulte Streaming de datos de sesión mediante Amazon CloudWatch Logs.	18 de noviembre de 2020
Nuevo tema: Verificación de la firma de SSM Agent	Ahora puede verificar la firma criptográfica del paquete del instalador para SSM Agent en instancias de Linux. Para obtener más información, consulte Esquemas y características de los documentos de SSM.	17 de noviembre de 2020

Nuevo tema: Conocimiento de los estados de las automatizaciones	Se ha agregado un nuevo tema al capítulo Automatización de Systems Manager que describe los estados de las acciones y las automatizaciones. Para obtener más información, consulte Conocimiento de los estados de automatización.	17 de noviembre de 2020
Tipos de fuente nuevos para el complemento `aws:downloadContent`	Ahora se admiten Git y HTTP como tipos de fuente para el complemento `aws:downloadContent`. Para obtener más información, consulte `aws:downloadContent`.	17 de noviembre de 2020
Nueva característica de esquema de documento de Systems Manager (documento de SSM)	En los documentos de SSM con la versión de esquema 2.2 o posterior, el parámetro `precondition` ahora admite hacer referencia a los parámetros de entrada del documento. Para obtener más información, consulte Esquemas y características de los documentos de SSM.	17 de noviembre de 2020

Nuevo origen de datos en Explorer: AWS Config	Explorer ahora muestra información de conformidad de AWS Config, que incluyen un resumen general de reglas conformes y no conformes de AWS Config, el número de recursos conformes y no conformes y los detalles específicos sobre cada uno (cuando se profundiza en una regla o un recurso no conforme). Para obtener más información, consulte Edición de orígenes de datos de Explorer de Systems Manager.	11 de noviembre de 2020
Nuevo tema: Ejecución de grupos de escalado automático con asociaciones	Se agregó una nueva sección a State Manager que describe las prácticas recomendadas para crear asociaciones para ejecutar grupos de escalado automático. Para obtener más información, consulte Ejecución de grupos de escalado automático con asociaciones.	10 de noviembre de 2020
Quick Setup ahora admite tener como destino un grupo de recursos	Quick Setup ahora admite la elección de un grupo de recursos como destino para el tipo de configuración local. Para obtener más información, consulte Elección de destinos para Quick Setup.	5 de noviembre de 2020

Patch Manager agrega compatibilidad con Debian Server 10 LTS, Oracle Linux 7.9 LTS y Ubuntu Server 20.10 STR	A partir de ahora, puede utilizar Patch Manager para aplicar revisiones a instancia s de Debian Server 10 LTS, Oracle Linux 7.9 LTS yUbuntu Server 20.10 STR. Para obtener más información, consulte los temas siguientes: • Requisitos previos de Patch Manager • Cómo se seleccionan las revisiones de seguridad • Cómo se instalan las revisiones • Funcionamiento de las reglas de líneas de base de revisiones en Debian Server • Funcionamiento de las reglas de línea de base de revisiones en Oracle Linux • Funcionamiento de las reglas de líneas de base de revisiones en Ubuntu Server	4 de noviembre de 2020

Nueva compatibilidad de EventBridge con AWS Systems Manager Change Calendar	Amazon EventBridge ofrece ahora soporte para eventos de Change Calendar en reglas de eventos. Cuando cambia el estado de un calendario, EventBridge puede iniciar la acción de destino que ha definido una regla de EventBridge. Para obtener información acerca de cómo trabajar con los eventos EventBridge y Systems Manager, consulte los siguientes temas. • Configuración de EventBridge para eventos de Systems Manager • Referencia: patrones y tipos de eventos de Amazon EventBridge para Systems Manager	4 de noviembre de 2020

Configurar CloudWatch para crear OpsItems desde alarmas	Puede configurar Amazon CloudWatch para crear automáticamente un OpsItem en Systems Manager OpsCenter cuando una alarma entra en el estado `ALARM`. Esto le permite que se puedan diagnosticar y solucionar rápidamente los problemas con recursos de AWS desde una única consola. Para obtener más información, consulte Configuración de CloudWatch para crear OpsItems desde alarmas.	4 de noviembre de 2020
Compatibilidad con Ubuntu Server 20.10	AWS Systems Manager ahora es compatible con la versión a corto plazo (STR) de Ubuntu Server 20.10. Para obtener más información, consulte los temas siguientes: • Sistemas operativos compatibles • Instalar el SSM Agent para un entorno híbrido (Linux) • Manually install SSM Agent on Ubuntu Server instances (Instalación manual de SSM Agent en instancias de Red Hat Enterprise Linux) • Verificación del estado de SSM Agent e inicio del agente	22 de octubre de 2020

Nuevo tema: Permiso de perfiles de shell configurable	Ahora puede permitir perfiles de shell configurables con Session Manager. Cuando permite perfiles de shell configurables, puede personalizar las preferencias dentro de las sesiones, como preferencias de shell, variables de entorno, directorios de trabajo y ejecutar varios comandos cuando se inicia una sesión. Para obtener más información, consulte Permitir perfiles de shell configurables.	21 de octubre de 2020
Los resultados de conformidad de revisiones ahora informan qué CVE se resuelve con qué revisiones	Para la mayoría de los sistemas Linux compatibles, cuando observa los resultados de conformidad de revisiones para las instancias administradas, los detalles que puede ver ahora informan qué problemas del boletín de vulnerabilidades y exposiciones comunes (CVE) se resuelven mediante las revisiones disponibles. Esta información puede ayudarlo a determinar con qué urgencia necesita instalar una revisión faltante o fallido. Para obtener información, consulte Visualización de resultados de conformidad de revisiones.	20 de octubre de 2020

Compatibilidad ampliada para metadatos de revisiones de Linux	Ahora puede ver muchos detalles sobre las revisiones de Linux disponibles en Patch Manager. Puede elegir ver datos de revisiones como arquitectura, fecha de inicio, versión, ID de CVE, ID de Advisory, ID de Bugzilla, repositorio y más. Además, la operación de la API DescribeAvailablePatches se actualizó para admitir sistemas operativos Linux y se filtró según estos nuevos tipos de metadatos de revisiones disponibles. Para obtener más información, consulte los temas siguientes: • Visualización de revisiones disponibles • DescribeAvailablePatches y Revisión en la Referencia de la API de AWS Systems Manager • describe-available-patches en la sección AWS Systems Manager en la Referencia de comandos de la AWS CLI.	16 de octubre de 2020

Complemento de Session Manager para la versión 1.2.7.0 de la AWS CLI	Se lanzó una nueva versión del complemento Session Manager para la AWS CLI. Para obtener más informaci ón, consulte Versión más reciente del complemento Session Manager e historial de publicaciones.	15 de octubre de 2020
Nuevo tema: Esquema del documento de Session	El nuevo tema Esquema de documento de sesión describe los elementos de esquema de un documento de sesión. Esta información puede ayudarlo a crear documentos personali zados de Sesión en los que especifique preferencias para los tipos de sesiones que utilice con Session Manager.	15 de octubre de 2020
Nuevo tema: Búsqueda de texto libre para documentos de SSM	El cuadro de búsqueda en la página Documents (Document os) de Systems Manager ahora admite las búsquedas de texto libre. La búsqueda de texto libre compara el término o los términos de búsqueda que introduce con el nombre del documento en cada documento de SSM. Para obtener más información, consulte Uso de la búsqueda de texto libre.	15 de octubre de 2020

Nuevo tema: solución de problemas de disponibilidad de instancias administradas de Amazon EC2	El nuevo tema Solución de problemas de disponibilidad de instancias administradas de Amazon EC2 lo ayuda a investigar por qué una instancia de Amazon EC2 que ha confirmado que se está ejecutando no está disponible en las listas de instancias administradas disponibles en Systems Manager.	6 de octubre de 2020

Reorganización de capítulos de Parameter Store	Para ayudarlo a encontrar la información que necesita de manera más eficiente, reorganizamos el contenido en el capítulo Parameter Store de la Guía del usuario de AWS Systems Manager. La mayor parte del contenido está ahora organizado en las secciones Setting up Parameter Store (Configuración de Parameter Store) y Working with Parameter Store (Uso de Parameter Store). Además, el tema AWS Systems ManagerParameter Store se ha ampliado para incluir las siguientes secciones: • ¿Cómo puede Parameter Store beneficiar a mi organización? • ¿Quién debe utilizar Parameter Store? • ¿Cuáles son las características de Parameter Store? • ¿Qué es un parámetro?	1 de octubre de 2020

Nuevos temas relacionados con la conformidad de revisiones	Se han agregado los siguientes temas para ayudarlo a identificar las instancias administradas que no cumplen con las revisiones, comprender los diferentes tipos de análisis de conformidad de revisiones y seguir los pasos adecuados para que las instancias cumplan las normas. • Identificación de instancias no conformes • Aplicación de revisiones a instancias no conformes • Visualización de resultados de conformidad de revisiones	24 de septiembre de 2020
Versión 3.0 de SSM Agent	Systems Manager lanzó una nueva versión de SSM Agent.	21 de septiembre de 2020

Temas nuevos y actualizados: Amazon EventBridge sustituye a CloudWatch Events para la administración de eventos	CloudWatch Events y EventBridge son el mismo servicio subyacente y la misma API, pero EventBridge ofrece más características y ahora es la manera preferida de administrar sus eventos en AWS. (Los cambios que realice en CloudWatch o EventBridge se reflejarán en cada consola). Referencias a CloudWatch Events y procedimientos existentes en toda la Guía del usuario de AWS Systems Manager se han actualizado para reflejar la compatibilidad con EventBridge. Además, se agregaron los siguientes temas nuevos. • Supervisión de eventos de Systems Manager • Configuración de EventBridge para eventos de Systems Manager • Ejemplos de tipos de destino de Systems Manager • Referencia: patrones y tipos de eventos de Amazon EventBridge para Systems Manager	18 de septiembre de 2020

Integración de AWS Security Hub y Patch Manager	Ahora puede integrar Patch Manager con AWS Security Hub. Security Hub le proporciona una visión completa de su estado de seguridad en AWS y lo ayuda a verificar su entorno con los estándares y las prácticas recomendadas del sector de la seguridad. Cuando se integra con Patch Manager, Security Hub monitorea el estado de aplicación de revisiones de sus flotas desde el punto de vista seguro. Para obtener más información, consulte Integración de Patch Manager con AWS Security Hub.	17 de septiembre de 2020

Pseudoparámetros de periodo de mantenimiento: nuevos tipos de recursos compatibles con {{TARGET_ID}} y {{RESOURCE_ID}}	Cuando registra una tarea de periodo de mantenimiento, se utiliza la opción `--task-invocation-parameters` para especificar los parámetros que son exclusivos de cada uno de los cuatro tipos de tarea. También puede hacer referencia a determinados valores mediante la sintaxis de pseudoparámetros, como `{{TARGET_ID}}` y `{{RESOURCE_ID}}` . Al ejecutarse la tarea del periodo de mantenimiento, esta pasa los valores correctos en lugar de los marcadores de posición del pseudoparámetro. Dos tipos de recursos adicionales están ahora disponibles para su uso con los pseudoparámetros `{{TARGET_ID}}` y `{{RESOURCE_ID}}` . Ahora puede utilizar los tipos de recursos `AWS::RDS::DBInstance` y `AWS::SSM::ManagedInstance` con estos pseudoparámetros. Para obtener información acerca de los pseudoparámetros del periodo de mantenimiento, consulte Uso de pseudoparámetros al registrar tareas de la ventana de mantenimiento.	14 de septiembre de 2020

Instancias de revisiones en diferido con la nueva opción 'Patch now' (Aplicar revisión ahora)	Ahora puede utilizar la consola de Systems Manager para aplicar revisiones a instancias o buscar revisiones faltantes en cualquier momento. Puede hacerlo sin tener que crear o modificar una programación, o especificar opciones completas de configuración de revisiones para adaptarse a una necesidad inmediata de revisiones. Solo necesita especificar si desea analizar o instalar revisiones e identificar las instancias de destino para la operación. Patch Manager aplica automáticamente la línea de base de revisiones predeterminada actual para los tipos de instancia y aplica las opciones de prácticas recomendadas para cuántas instancias se aplican revisiones a la vez y cuántos errores se permiten antes de que se produzca un error en la operación. Para obtener información, consulte Aplicación de revisiones a instancias en diferido.	9 de septiembre de 2020

Nuevo tema: Verificación del estado de SSM Agent e inicio del agente	El nuevo tema Verificación del estado de SSM Agent e inicio del agente proporcio na comandos para verificar si SSM Agent se ejecuta en cada sistema operativo compatibl e. También proporciona los comandos para iniciar el agente si no se está ejecutand o.	7 de septiembre de 2020
Patch Manager ya admite Ubuntu Server 20.04 LTS	A partir de ahora, puede utilizar Patch Manager para aplicar revisiones a instancias de Ubuntu Server 20.04 LTS. Para obtener más información, consulte los temas siguientes: • Cómo se seleccionan las revisiones de seguridad • Cómo se instalan las revisiones • Funcionamiento de las reglas de líneas de base de revisiones en Ubuntu Server	31 de agosto de 2020
Nuevo tema para casos de uso y prácticas recomendadas	Hemos agregado un nuevo tema para que los usuarios puedan entender rápidamen te las diferencias entre Maintenance Windows y State Manager. Para obtener más información, consulte Elegir entre State Manager y Maintenance Windows.	28 de agosto de 2020

Nuevas características de OpsCenter	OpsCenter incluye nuevas características para ayudarlo a localizar y ejecutar rápidamente manuales de procedimientos de Automation para solucionar problemas. Para obtener más información, consulte Características del manual de procedimientos de Automation en OpsCenter.	19 de agosto de 2020
Nuevo origen de datos en Explorer: casos de AWS Support	Explorer ahora muestra información sobre casos de Soporte. Debe tener una cuenta Enterprise o Business configurada con Soporte. Para obtener más información, consulte Edición de orígenes de datos de Explorer de Systems Manager.	13 de agosto de 2020
Distributor ahora ofrece un paquete de terceros de Trend Micro	Distributor ahora incluye un paquete de terceros de Trend Micro. Puede utilizar Distributor para instalar el agente de Trend Micro Cloud One en las instancias administradas. Trend Micro Cloud One lo ayuda a proteger sus cargas de trabajo en la nube. Para obtener más información, consulte AWSDistributor.	12 de agosto de 2020

El complemento de documento `aws:configurePackage` ahora incluye el parámetro additionalArguments	El complemento `aws:configurePackage` del documento de Systems Manager Command ahora admite proporcionar parámetros adicionales a sus scripts (instalar, desinstalar y actualizar) con el nuevo parámetro `additionalArguments`. Para obtener más información, consulte el tema `aws:configurePackage`.	11 de agosto de 2020
Contenido de AppConfig movido a una guía del usuario independiente	La información sobre AWS AppConfig se ha trasladado a una guía del usuario independiente. Para obtener más información, consulte ¿Qué es AWSAppConfig? AppConfig también tiene una página de aterrizaje de documentación independiente con enlaces a la guía del usuario, la referencia de la API de AppConfig y un nuevo taller de AppConfig.	3 de agosto de 2020

Quick Setup ya admite AWS Organizations	Quick Setup ya admite AWS Organizations, lo que le permite configurar rápidamente los roles de seguridad necesarios y las herramientas de Systems Manager de uso común en varias cuentas y regiones. Para obtener más información, consulte AWS Systems ManagerQuick Setup.	23 de julio de 2020
Nuevo origen de datos en Explorer: conformidad de la asociación	Explorer ahora muestra los datos de conformidad de asociación de State Manager. Para obtener más información, consulte Edición de orígenes de datos de Explorer de Systems Manager.	23 de julio de 2020

Nuevo documento de Systems Manager Command para activar y desactivar Kernel Live Patching	El documento `AWS-ConfigureKernelLivePatching` ya está disponible para utilizar con Run Command cuando quiera activar o desactivar Kernel Live Patching en instancias de Amazon Linux 2. Este documento reemplaza la necesidad de crear sus propios documentos de Command personalizados para estas tareas. Para obtener más información, consulte Utilizar Kernel Live Patching en instancias de Amazon Linux 2	22 de julio de 2020
Cuotas de Automation actualizadas	Se han actualizado las cuotas de servicio para Automation, lo que incluye una cola separada para automatizaciones de control de velocidad. Para obtener más información, consulte AWS Systems Manager Automation.	20 de julio de 2020

Especificar el número de días de desplazamiento de la programación en un periodo de mantenimiento mediante la consola	Con la consola de Systems Manager, ahora puede especificar un número de días de espera después de la fecha y la hora especific adas por una expresión CRON antes de ejecutar un periodo de mantenimiento. (Anterior mente, esta opción solo estaba disponible cuando se utilizaba un AWS SDK o una herramienta de línea de comandos). Por ejemplo, si su expresión CRON programa un periodo de mantenimiento para que se ejecute el tercer martes de cada mes a las 23.30 h (`cron(0 30 23 ? * TUE#3 *)`) y especifica un desplazamiento de programac ión de 2, el periodo no se ejecutará hasta dos días después a las 23.30 h. Para obtener más información, consulte Expresiones cron y rate para Systems Manager y Especificar el número de días de desplazamiento de la programación en un periodo de mantenimiento.	17 de julio de 2020

Actualizar PowerShell con Run Command	Para ayudarlo a actualizar PowerShell a la versión 5.1 en sus instancias Windows Server 2012 y 2012 R2, agregamos una explicación a la Guía del usuario de AWS Systems Manager. Para obtener más información, consulte Actualización de PowerShell con Run Command.	30 de junio de 2020
Patch Manager ahora soporta CentOS 8.0 y 8.1	Ahora puede utilizar Patch Manager para aplicar revisiones a instancias CentOS 8.0 y 8.1. Para obtener más información, consulte los siguientes temas: • Cómo se seleccionan las revisiones de seguridad • Cómo se instalan las revisiones • Funcionamiento de las reglas de líneas de base de revisiones en CentOS • Instalar manualmente SSM Agent en instancias de CentOS • Cómo instalar SSM Agent en nodos de Linux híbridos	27 de junio de 2020

AppConfig se integra con AWS CodePipeline	AppConfig es una acción de implementación integrada para AWS CodePipeline (CodePipeline). CodePipeline es un servicio de entrega continua completamente administrado que lo ayuda a automatizar sus canales de lanzamiento para actualizaciones de infraestructura y aplicaciones rápidas y confiables. CodePipeline automatiza las fases de creación, prueba e implementación de su proceso de publicación cada vez que se produce un cambio de código, de acuerdo con el modelo de publicación que defina. La integración de AppConfig con CodePipeline ofrece los siguientes beneficios. Para obtener más información, consulte Integración de AppConfig con CodePipeline. • Los clientes que utilizan CodePipeline para gestionar la organización ahora tienen un medio liviano de implementar cambios de configuración en sus aplicaciones sin tener que implementar toda su base de código. • Los clientes que deseen utilizar AppConfig para	25 de junio de 2020

	administrar implement aciones de configuración, pero están limitados porque AppConfig no admite su código actual o almacén de configuración, ahora tienen opciones adicional es. CodePipeline admite AWS CodeCommit, GitHub y BitBucket (por nombrar algunos).	
Nuevo capítulo: Integraciones de productos y servicios	Para ayudarlo a comprende r cómo Systems Manager se integra con Servicios de AWS y otros productos y servicios, se agregó un nuevo capítulo a la guía del usuario de AWS Systems Manager. Para obtener más informaci ón, consulte Integraciones de productos y servicios con Systems Manager.	23 de junio de 2020

Reorganización de capítulos de Automation	Para ayudarlo a encontrar lo que necesita, reorganizamos los temas en el capítulo Automation de la Guía del usuario de AWS Systems Manager. Por ejemplo, las acciones de Automation y las referencias de manuales de procedimientos de Automation son ahora secciones de nivel superior del capítulo. Para obtener más información, consulte AWS Systems Manager Automation.	23 de junio de 2020

Especificar el número de días de desplazamiento de la programación en un período de mantenimiento	Con una herramienta de línea de comandos o el AWS SDK, ahora puede especific ar un número de días de espera después de la fecha y la hora especificadas por una expresión CRON antes de ejecutar un periodo de mantenimiento. Por ejemplo, si su expresión CRON programa un periodo de mantenimiento para que se ejecute el tercer martes de cada mes a las 23.30 h (`cron(0 30 23 ? * TUE#3 *)`) y especifica un desplazamiento de programac ión de 2, el periodo no se ejecutará hasta dos días después a las 23.30 h. Para obtener más información, consulte Expresiones cron y rate para Systems Manager y Especificar el número de días de desplazamiento de la programación en un periodo de mantenimiento.	19 de junio de 2020

Soporte de Patch Managerco n Kernel Live Patching en instancias de Amazon Linux 2	Kernel Live Patching para Amazon Linux 2 le permite aplicar revisiones de vulnerabi lidad de seguridad y de errores críticos a un kernel de Linux en ejecución, sin reinicios ni interrupciones en las aplicaciones en ejecución . Ahora puede permitir la característica y aplicar revisiones en caliente del kernel mediante el Patch Manager. Para obtener más información, consulte Utilizar Kernel Live Patching en instancias de Amazon Linux 2	16 de junio de 2020
Patch Manager aumenta la compatibilidad con versiones de Oracle Linux	Anteriormente, Patch Manager solo admitía la versión 7.6 de Oracle Linux. Como se indica en los requisitos previos de Patch Manager, el soporte ahora cubre las versiones 7.5-7.8.	16 de junio de 2020

Situación de ejemplo para utilizar el parámetro `InstallOverrideList` en las operaciones de aplicación de revisiones	El nuevo tema Situación de ejemplo de uso del parámetro `InstallOverrideList` describe una estrategia para utilizar el parámetro `InstallOverrideList` en el documento `AWS-RunPatchBaseline` para aplicar diferentes tipos de revisiones a un grupo de destino, en diferentes programaciones de periodos de mantenimiento, mientras se utiliza una línea de base de revisiones.	11 de junio de 2020
Estrategias de implementación predefinidas para AppConfig	Ahora AppConfig ofrece estrategias de implementación predefinidas. Para obtener más información, consulte Creación de una estrategia de implementación.	10 de junio de 2020

Patch Manager ya admite Red Hat Enterprise Linux (RHEL) 7.8-8.2	Ahora puede utilizar Patch Manager para aplicar revisiones a instancias de RHEL 7.8–8.2. Para obtener más información, consulte los siguientes temas: • Cómo se seleccionan las revisiones de seguridad • Cómo se instalan las revisiones • Funcionamiento de las reglas de líneas de base de revisiones en RHEL • Manually install SSM Agent on Red Hat Enterprise Linux instances (Instalación manual de SSM Agent en instancias de Red Hat Enterprise Linux) • Cómo instalar SSM Agent en nodos de Linux híbridos	9 de junio de 2020

Explorer admite administradores delegados	Si agrega datos de Explorer de varias Regiones de AWS y Cuentas de AWS mediante la sincronización de datos de recursos con AWS Organizations, le sugerimos que configure un administrador delegado para Explorer. Un administrador delegado mejora la seguridad de Explorer al limitar el número de administradores de Explorer que pueden crear o eliminar varias cuentas y sincronizar los datos de recursos de la región a una sola persona. Tampoco es necesario iniciar sesión en la cuenta de administración de AWS Organizations para administrar las sincronizaciones de datos de recursos en Explorer. Para obtener más información, consulte Configuración de un administrador delegado.	3 de junio de 2020

Aplicar la asociación de State Manager solo en el siguiente intervalo cron especificado	Si no desea que una asociación State Manager se ejecute inmediatamente después de crearla, puede elegir la opción Apply association only at the next specified Cron interval (Aplicar la asociación únicamente en el siguiente intervalo Cron especificado) en la consola de Systems Manager. Para obtener más información, consulte Creación de asociaciones.	3 de junio de 2020
Nuevo origen de datos en Explorer: AWS Compute Optimizer	Explorer ahora muestra los datos de AWS Compute Optimizer. Esto incluye un recuento de instancias EC2 subaprovisionadas y sobreaprovisionadas conclusiones de optimización, información sobre precios bajo demanda y recomendaciones para el tipo de instancia y el precio. Para obtener más información, consulte los detalles de la configuración de AWS Compute Optimizer en Configuración de servicios relacionados.	26 de mayo de 2020

Instalación de los Service Packs de Windows y las actualizaciones de versiones secundarias de Linux mediante Patch Manager	El tema nuevo Tutorial: crear una línea de base de revisiones para instalar los Service Packs de Windows (consola) muestra cómo crear una línea de base de revisiones dedicada exclusivamente a instalar los Service Packs de Windows. El tema Crear una línea de base de revisiones personalizada (Linux) se ha actualizado con información acerca de la inclusión de actualizaciones de versiones secundarias para sistemas operativos Linux en las líneas de base de revisiones.	21 de mayo de 2020
Reorganización de capítulos de Parameter Store	Todos los temas relacionados con la configuración o el ajuste de opciones para operaciones de Parameter Store se han consolidado en la sección Setting up Parameter Store (Configuración del almacén de parámetros). Esto incluye los temas Administración de niveles de parámetros y Aumento del rendimiento de Parameter Store, que se han reubicado desde otras partes del capítulo.	18 de mayo de 2020

Nuevo tema para crear cadenas de fecha y hora para interactuar con operaciones de la API de Systems Manager	El nuevo tema Creación de cadenas de fecha y hora con formato para Systems Manager describe cómo crear cadenas de fecha y hora con formato para interactuar con operaciones de la API de Systems Manager.	13 de mayo de 2020
Acerca de los permisos para cifrar parámetros SecureString	El tema nuevo Restricting access to Systems Manager parameters using IAM policies (Restricción del acceso a los parámetros de Systems Manager mediante políticas de IAM) explica la diferenci a entre cifrar los parámetros `SecureString` que utilizan AWS KMS key y utilizar Clave administrada de AWS proporcionada por AWS.	13 de mayo de 2020

Patch Manager ahora es compatible con los sistemas operativos Debian Server y Oracle Linux 7.6	A partir de ahora, puede utilizar Patch Manager para aplicar revisiones a instancias de Debian Server y Oracle Linux. Patch Manager admite la aplicación de revisiones a las versiones 8.x y 9.x de Debian Server y 7.6 de Oracle Linux. Para obtener más información, consulte los temas siguientes: • Cómo se seleccionan las revisiones de seguridad • Cómo se instalan las revisiones • Funcionamiento de las reglas de líneas de base de revisiones en Debian Server • Funcionamiento de las reglas de líneas de base de revisiones en Oracle Linux	7 de mayo de 2020

Crear asociaciones de State Manager que tengan como destino AWS Resource Groups	Además de las etiquetas de establecimiento de destino, las instancias individuales y todas las instancias de su Cuenta de AWS, ahora puede crear asociaciones de State Manager que definan el destino de las instancias en AWS Resource Groups. Para obtener más informaci ón, consulte Acerca de los controles de velocidad y destinos en las asociaciones de State Manager	7 de mayo de 2020

Tipo de datos `aws:ec2:image` nuevo en Parameter Store para validar las ID de AMI	Cuando cree un parámetro `String`, ahora puede especificar un tipo de datos como `aws:ec2:image` para asegurarse de que el valor de parámetro que especifiq ue tenga formato de ID de Amazon Machine Image (AMI) válido. La compatibilidad con los formatos de ID de AMI significa que no tiene que actualizar todos los scripts y las plantillas con un nuevo ID cada vez que cambie la AMI que desea utilizar en sus procesos. Puede crear un parámetro con el tipo de datos `aws:ec2:image` , y para su valor, especifique el ID de una AMI. Esta es la AMI desde la que desea crear nuevas instancias. A continuación, haga referenci a a este parámetro en sus plantillas y comandos. Cuando esté listo para utilizar una AMI diferente, actualice el valor del parámetro. Parameter Store valida el nuevo ID de AMI y no es necesario actualiza r los scripts ni las plantillas. Para obtener más información, consulte Compatibilidad con parámetros nativos para los ID de Amazon Machine Image.	5 de mayo de 2020

Administración de los códigos de salida en los comandos de Run Command	Run Command lo habilita a definir cómo se tratan los códigos de salida en sus scripts. De forma predeterminada, el código de salida del último comando ejecutado en un script se registra como el código de salida de todo el script. Sin embargo, puede incluir una instrucción condicional de shell para salir del script si algún comando anterior al final produce un error utilizando el siguiente enfoque. Para obtener ejemplos, consulte el nuevo tema Tratamiento de códigos de salida en comandos de Run Command.	5 de mayo de 2020

Se han publicado nuevos parámetros públicos para las zonas de disponibilidad y las zonas locales	Se han publicado parámetros públicos para que la información sobre las zonas de disponibilidad y las zonas locales de AWS estén disponibles mediante programación. Estos parámetros se suman a los parámetros públicos de infraestructura global existentes para los Servicios de AWS y las Regiones de AWS. Para obtener más información, consulte Llamada a parámetros públicos para Servicios de AWS, regiones, puntos de conexión, zonas de disponibilidad, zonas locales y zonas de Wavelength.	4 de mayo de 2020
Nuevo origen de datos en Explorer: AWS Trusted Advisor	Explorer ahora muestra los datos de AWS Trusted Advisor. Esto incluye el estado de las comprobaciones y recomendaciones de las prácticas recomendadas en las siguientes áreas: optimización de costos, seguridad, tolerancia a errores, rendimiento y cuotas de servicio. Para obtener más información, consulte los detalles de la configuración de Trusted Advisor en Configuración de servicios relacionados.	4 de mayo de 2020

Crear asociaciones de State Manager que ejecutan recetas de Chef	Puede crear asociaciones de State Manager que ejecuten libros de recetas de Chef utilizando el documento `AWS-ApplyChefRecipes` . Este documento ofrece los siguientes beneficios para ejecutar recetas de Chef: • Admite múltiples versiones de Chef (de Chef 11 a Chef 14). • Instala automáticamente el software cliente de Chef en las instancias de destino. • Opcionalmente, ejecuta comprobaciones de conformidad de Systems Manager en instancias de destino y almacena los resultados de las comprobaciones de conformidad en un bucket de S3. • Ejecuta varios libros de recetas y recetas en una sola ejecución del documento. • Opcionalmente ejecuta recetas en modo `why-run`, para mostrar qué recetas cambiarán en instancias de destino sin realizar cambios. • Opcionalmente aplica atributos JSON personali	19 de marzo de 2020

	zados a las ejecuciones de `chef-client` . Para obtener más informaci ón, consulte Creación de asociaciones que ejecutan recetas de Chef	
Sincronizar los datos de inventario de varias Cuentas de AWS en un bucket central de Simple Storage Service (Amazon S3)	Puede sincronizar los datos de Systems Manager Inventory de varias Cuentas de AWS en un bucket de Amazon S3 central. Las cuentas deben definirse en AWS Organizations. Para obtener más información, consulte Creación de una sincroniz ación de datos de recursos de inventario para varias cuentas definidas en AWS Organizat ions.	16 de marzo de 2020
Almacenar configuraciones de AppConfig en Simple Storage Service (Amazon S3)	Anteriormente, AppConfig solo admitía configuraciones de aplicaciones almacenadas en documentos de Systems Manager (SSM) o parámetros de Parameter Store. Además de estas opciones, AppConfig ahora admite el almacenam iento de configuraciones en Amazon S3. Para obtener más información, consulte Acerca de las configuraciones almacenadas en Amazon S3.	13 de marzo de 2020

De forma predeterminada, SSM Agent se ha instalado en AMIs optimizadas de Amazon ECS	Ahora SSM Agent está instalado de forma predeterminada en AMIs optimizadas para Amazon ECS. Para obtener más información, consulte Trabajar con SSM Agent.	25 de febrero de 2020
Crear configuraciones de AppConfig en la consola	AppConfig ahora le permite crear una configuración de aplicación en la consola al momento de crear un perfil de configuración. Para obtener más información, consulte Creating a configuration and a configuration profile.	13 de febrero de 2020

Aprobar automáticamente solo las revisiones publicadas hasta una fecha especificada	Además de la opción de aprobar automáticamente las revisiones para su instalación un número determinado de días después de su lanzamiento, Patch Manager ahora admite la posibilidad de aprobar automáticamente solo las revisiones publicadas en la fecha especificada o con anterioridad. Por ejemplo, si especifica el 7 de julio de 2020 como fecha límite en la línea de base de revisiones, no se instalarán automáticamente las revisiones publicados a partir del 8 de julio de 2020. Para obtener más información, consulte About custom baselines (Acerca de las bases de referencia personalizadas) y Working with custom patch baselines (console) (Creación de una línea de base de revisiones personalizada [consola]).	12 de febrero de 2020

Utilizar el pseudoparámetro {{RESOURCE_ID}} en las tareas del periodo de mantenimiento	Al registrar una tarea de periodo de mantenimiento, se especifican los parámetros que son únicos para el tipo de tarea. También puede hacer referencia a determinados valores mediante la sintaxis de pseudoparámetros, como `{{TARGET_ID}}` , `{{TARGET_TYPE}}` y `{{WINDOW_TARGET_ID}}` . Al ejecutarse la tarea del periodo de mantenimiento, esta pasa los valores correctos en lugar de los marcadores de posición del pseudoparámetro. Para admitir recursos que formen parte de un grupo de recursos como destino, puede utilizar el pseudoparámetro `{{RESOURCE_ID}}` para pasar valores de recursos como tablas de DynamoDB, buckets de S3 y otros tipos admitidos. Para obtener más información, consulte los temas siguientes en Tutorial: crear y configurar un período de mantenimiento (AWS CLI): • Utilización de pseudoparámetros en el registro de las tareas del periodo de mantenimiento	6 de febrero de 2020

	• Ejemplos: registrar tareas en un periodo de mantenimiento	
Volver a ejecutar rápidamente los comandos	Systems Manager incluye dos opciones que lo ayudarán a volver a ejecutar un comando desde la página Run Command de la consola de AWS Systems Manager. Rerun (Volver a ejecutar): este botón le permite ejecutar el mismo comando sin realizarle cambios. Copy to new (Copiar en nuevo): este botón copia la configuración de un comando en un comando nuevo y le ofrece la opción de editar esa configuración antes de ejecutarlo. Para obtener más información, consulte Volver a ejecutar comandos.	5 de febrero de 2020

Revertir de la capa de instancias avanzadas a la capa de instancias estándar	Si previamente configuró todas las instancias locales que se ejecutan en el entorno híbrido para utilizar el nivel de instancias avanzadas, ahora puede configurar rápidamente esas instancias para que utilicen el nivel de instancias estándar. La reversión al nivel de instancias estándar se aplica a todas las instancias híbridas en una Cuenta de AWS y una única Región de AWS. Revertir al nivel de instancias estándar afecta a la disponibilidad de algunas herramientas de Systems Manager. Para obtener más información, consulte Revertir de la capa de instancias avanzadas a la capa de instancias estándar	16 de enero de 2020

Nueva opción para omitir los reinicios de instancias después de la instalación de la revisión	Anteriormente, las instancias administradas siempre se reiniciaban después de que Patch Manager instalara revisiones en ellas. Un nuevo parámetro `RebootOption` en el documento de SSM `AWS-RunPatchBaseline` le permite especificar si desea que las instancias se reinicien automáticamente después de instalar nuevas revisiones. Para obtener más información, consulte Nombre del parámetro: RebootOption en el tema Acerca del documento de SSM `AWS-RunPatchBaseline`.	15 de enero de 2020
Nuevo tema: “Ejecución de scripts de PowerShell en instancias de Linux”	Un nuevo tema que describe cómo utilizar Run Command para ejecutar scripts de PowerShell en instancias de Linux. Para obtener más información, vea Ejecución de secuencias de comandos de PowerShell en instancias de Linux.	10 de enero de 2020

Actualizaciones en “configurar el SSM Agent para utilizar un proxy”	Los valores que se deben especificar al configurar SSM Agent para utilizar un proxy se han actualizado para reflejar las opciones tanto para los servidores proxy HTTP como para los servidores proxy HTTPS. Para obtener más información, vea Configurar SSM Agent para utilizar un proxy.	9 de enero de 2020

En el nuevo capítulo “Seguridad” se describen las prácticas para proteger los recursos de Systems Manager	El nuevo capítulo Seguridad de la Guía del usuario de AWS Systems Manager lo ayuda a entender cómo aplicar el modelo de responsabilidad compartida cuando se utiliza Systems Manager. Los temas del capítulo muestran cómo configurar Systems Manager para satisfacer sus objetivos de seguridad y conformidad. También puede aprender a utilizar otros Servicios de AWS que lo ayuden a supervisar y proteger los recursos de Systems Manager. **Note** En esta actualización, el capítulo de la guía del usuario Autenticación y control de acceso se ha sustituido por una nueva sección más sencilla, Identity and Access Management para AWS Systems Manager.	24 de diciembre de 2019

Nuevos ejemplos de manuales de procedimientos de Automation personalizados	Se ha agregado un conjunto de manuales de procedimientos de Automation personalizados de ejemplo a la guía del usuario. Estos ejemplos muestran cómo utilizar varias acciones de Automation para simplificar las tareas de implementación, de solución de problemas y de mantenimiento. Además, se han diseñado para ayudarlo a escribir sus propios manuales de procedimientos de Automation. Para obtener más información, consulte Ejemplos de manuales de procedimientos de Automation personalizados. También puede ver el contenido de manuales de procedimientos de Automation administrados por Amazon en la consola de Systems Manager. Para obtener más información, consulte Referencia del manual de procedimientos de Automatización de Systems Manager.	23 de diciembre de 2019

Compatibilidad con Oracle Linux	Systems Manager ahora admite Oracle Linux 7.5 y 7.7. Para obtener información sobre la instalación manual de SSM Agent en instancia s de EC2 para instancias de Oracle Linux, consulte Oracle Linux. Para obtener informaci ón sobre cómo instalar SSM Agent en servidores de Oracle Linux en un entorno híbrido, consulte Cómo instalar SSM Agent en nodos de Linux híbridos.	19 de diciembre de 2019

Inicio de sesiones de Session Manager desde la consola de Amazon EC2	Ahora puede comenzar sesiones de Session Manager desde la consola de Amazon Elastic Compute Cloud (Amazon EC2). Para trabajar con tareas relacionadas con la sesión desde la consola de Amazon EC2 se requieren permisos de IAM diferentes tanto para los usuarios como para los administradores. Puede proporcionar permisos para usar únicamente la consola de Session Manager y la AWS CLI, solo la consola de Amazon EC2 o estas tres herramientas. Para obtener más información, consulte los siguientes temas. • Políticas de IAM predeterminadas de inicio rápido para Session Manager • Inicio de una sesión (consola de Amazon EC2)	18 de diciembre de 2019

Soporte de CloudWatch para alarmas y las métricas de Run Command	Ahora AWS Systems Manager publica métricas sobre el estado de los comandos de Run Command en CloudWatch, lo que le permite establecer alarmas basadas en esas métricas. Los valores de estado terminal de los comandos para cuyas métricas puede realizar un seguimiento son `Success`, `Failed` y `Delivery Timed Out`. Para obtener más información, consulte Monitoreo de métricas Run Command mediante Amazon CloudWatch.	17 de diciembre de 2019

Nueva herramienta de Systems Manager: Change Calendar	Utilice Change Calendar de Systems Manager para especificar períodos de tiempo (eventos) durante los cuales se desea limitar o impedir cambios mediante código (por ejemplo, desde manuales de procedimientos de Automatización de Systems Manager o funciones de AWS Lambda) en los recursos. Un calendario de cambios es un nuevo tipo de documento de Systems Manager que almacena los datos de iCalendar 2.0 en texto sin formato. Para obtener más información, consulte Calendario de cambios de AWS Systems Manager.	11 de diciembre de 2019

Nueva herramienta de Systems Manager: AWSAppConfig	Utilice AppConfig para crear, administrar e implementar rápidamente configuraciones de aplicaciones. AppConfig admite implementaciones controladas en aplicacio nes de cualquier tamaño. Puede utilizar AppConfig con aplicaciones alojadas en instancias EC2, AWS Lambda, contenedores, aplicacio nes móviles o dispositivos IoT. Para evitar errores al implementar configuraciones de aplicaciones, AppConfig incluye validadores. Un validador proporciona una comprobación sintáctica o semántica para garantizar que la configuración que desea implementar funciona según lo previsto. Durante la implement ación de una configuración, AppConfig supervisa la aplicación para asegurars e de que la implementación se realiza correctamente. Si el sistema encuentra un error o si la implement ación dispara una alarma, AppConfig deshace el cambio para minimizar el impacto para los usuarios de la aplicación. Para obtener más información, consulte AWSAppConfig.	25 de noviembre de 2019

Nueva herramienta de Systems Manager: Explorador de Systems Manager	AWS Systems Manager Explorer es un panel de operaciones personalizable que transmite información sobre sus recursos de AWS. Explorer muestra una vista agregada de los datos de operaciones (OpsData) de sus Cuentas de AWS y en todas las Regiones de AWS. En Explorer, OpsData incluye metadatos sobre instancias EC2, detalles de conformidad de revisiones y elementos de trabajo operativos (OpsItems). Explorer proporciona un contexto sobre cómo OpsItems se distribuyen entre las unidades de negocio o las aplicaciones, cómo se presentan a lo largo del tiempo y cómo varían según la categoría. Puede agrupar y filtrar la información en Explorer para centrarse en los elementos que son relevante s para usted y que requieren que se tomen medidas. Cuando identifica problemas de alta prioridad, puede utilizar OpsCenter de Systems Manager para ejecutar manuales de procedimientos de Automation y resolver rápidamente esos problemas . Para obtener más informaci	18 de noviembre de 2019

ón, consulte AWS Systems Manager Explorer.

> **Note**
>
> La configuración de OpsCenter de Systems Manager está integrado con la configuración de Explorer. Si ya ha preparado OpsCenter, deberá completar la instalación integrada para verificar la configuración y las opciones. Si no ha configurado OpsCenter, puede utilizar la instalación integrada para comenzar con ambas herramientas. Para obtener más información, consulte Introducción a Explorer y OpsCenter.

Capacidades mejoradas de búsqueda de parámetros	Las herramientas para buscar parámetros ahora facilitan la búsqueda de parámetros cuando tiene un gran número de ellos en su cuenta o cuando no recuerda el nombre exacto de un parámetro. Con la herramienta de búsqueda, filtre por `contains`. Antes, las herramientas de búsqueda eran compatibles solo con la búsqueda de nombres de parámetros por `equals` y `begins-with` . Para obtener más información, consulte Búsqueda de parámetros de Systems Manager.	15 de noviembre de 2019

Nuevo generador de documentos basado en la consola para Automation | Soporte para ejecutar scripts en pasos de Automation	Ahora puede utilizar la Automatización de Systems Manager para crear y compartir manuales operativos estandarizados para garantizar la coherencia entre usuarios, Cuentas de AWS y Regiones de AWS. Con esta capacidad de ejecutar scripts y agregar documentación insertada en sus manuales de procedimientos de Automation mediante Markdown, puede reducir errores y eliminar pasos manuales como la navegación por procedimientos escritos en wikis y la ejecución de comandos de terminal. Para obtener más información, consulte los siguientes temas. • (Explicación: uso del generador de documentos para crear un manual de procedimientos de automatización personalizado) • `aws:executeScript` (Referencia de acciones de Automation) • Creación de manuales de procedimientos de Automation mediante el generador de documentos	14 de noviembre de 2019

	• Nuevas características de Automation en Systems Manager en el Blog de noticias de AWS	
Realizar una actualización de paquetes in situ utilizando Distributor	Antes, cuando quería instalar una actualización en un paquete utilizando Distributor, su única opción era desinstalar el paquete completo y volver a instalar la nueva versión. Ahora puede elegir realizar una actualización in situ. Durante una actualización in situ, Distributor solo instala los archivos nuevos o modificados desde la última instalación, según el script de actualización que incluya en el paquete. Con esta opción, la aplicación de paquete puede permanecer disponible y no estar desconectada durante la actualización. Para obtener más información, consulte los siguientes temas. • Creación de un paquete • Instalar o actualizar paquetes	11 de noviembre de 2019

Nueva característica de SSM Agent de actualización automática	Con un solo clic, puede configurar todas las instancias de su Cuenta de AWS para que verifiquen y descarguen automáticamente nuevas versiones de SSM Agent. Para ello, elija Agent auto update (Actualización automática del agente) en la página Managed instances (Instancias administradas) de la consola de AWS Systems Manager. Para obtener información, consulte Actualizaciones automáticas en SSM Agent.	5 de noviembre de 2019
Restricción del acceso a Session Manager mediante etiquetas proporcionadas por AWS	Ahora está disponible un segundo método para controlar el acceso de los usuarios a las acciones de sesión. Con este nuevo método, puede crear políticas de acceso de IAM utilizando etiquetas de sesión proporcionadas por AWS en lugar de utilizar la variable `{aws:username}` . El uso de estas etiquetas de sesión proporcionadas por AWS permite que las organizaciones que utilizan ID federados controlen el acceso de los usuarios a las sesiones. Para obtener más información, consulte Permitir que un usuario termine solo las sesiones que inició.	2 de octubre de 2019

Nuevo documento de SSM Command para aplicar manuales de estrategias de Ansible	Puede crear asociaciones de State Manager que ejecuten manuales de estrategias de Ansible mediante el documento `AWS-ApplyAnsiblePlaybooks` . Este documento ofrece los siguientes beneficios para ejecutar cuadernos de trabajo: • Compatibilidad con la ejecución de cuadernos de trabajo complejos • Soporte para descargar manuales de estrategias de GitHub y Amazon Simple Storage Service (Amazon S3) • Compatibilidad con la estructura de cuaderno de trabajo comprimido • Registro optimizado • Capacidad para especific ar qué cuaderno de trabajo ejecutar cuando se empaquetan los cuadernos de trabajo Para obtener más informaci ón, consulte Creación de asociaciones que ejecutan manuales de estrategias de Ansible	24 de septiembre de 2019

Compatibilidad con el enrutamiento de puertos para Session Manager	Session Manager ahora admite sesiones de enrutamiento de puertos. El enrutamiento de puertos le permite crear de forma segura túneles entre las instancias implementadas en subredes privadas sin necesidad de iniciar el servicio SSH en el servidor, abrir el puerto SSH en el grupo de seguridad o utilizar un host bastión. Al igual que los túneles SSH, el enrutamiento de puertos le permite enviar el tráfico de su portátil a los puertos abiertos de su instancia. Una vez configurado el enrutamiento de puertos, puede conectarse al puerto local y obtener acceso a la aplicación de servidor que se ejecuta dentro de la instancia. Para obtener más información, consulte los temas siguientes: • Reenvío de transferencias mediante AWS Systems ManagerSession Manager en el Blog de noticias de AWS. • Inicio de una sesión (enrutamiento de puertos)	29 de agosto de 2019

Especificar una capa predeterminada de parámetros o automatizar la selección de capa	Ahora puede especificar un nivel predeterminado de parámetros para utilizarlo en las solicitudes de creación o actualización de un parámetro que no especifica una capa. Puede establecer la capa predeterminada en parámetros estándar, parámetros avanzados o una nueva opción, capas inteligentes. Las capas inteligentes evalúan cada solicitud PutParameter y crean un parámetro avanzado solo cuando es necesario. (Los parámetros avanzados son obligatorios si el tamaño del valor del parámetro es superior a 4 KB, se asocia una política de parámetros al parámetro o ya se han creado los 10 000 parámetros máximos admitidos para el nivel estándar). Para obtener más información acerca de cómo especificar una capa predeterminada y cómo utilizar capas inteligentes, consulte Especificación de una capa de parámetros predeterminada.	27 de agosto de 2019

La sección “Trabajo con asociaciones” se ha actualizado con los procedimientos de la CLI y PowerShell	La sección Trabajo con asociaciones se ha actualizado para incluir documentación de procedimientos sobre la administración de asociaciones mediante la AWS CLI o Herramientas de AWS para PowerShell. Para obtener más información, consulte Trabajo con asociaciones en Systems Manager.	26 de agosto de 2019
La sección “Uso de ejecuciones de Automation” se ha actualizado con los procedimientos de la CLI y PowerShell	La sección Uso de ejecuciones de Automation se ha actualizado para incluir documentación de procedimientos sobre la administración de asociaciones mediante la AWS CLI o Herramientas de AWS para PowerShell. Para obtener más información, consulte Uso de ejecuciones de Automation.	20 de agosto de 2019

OpsCenter se integra con Application Insights	OpsCenter se integra con Amazon CloudWatch Application Insights para .NET y SQL Server. Esto significa que puede crear automáticamente OpsItems para problemas detectados en sus aplicaciones. Para obtener información acerca de cómo se configura Application Insights para crear OpsItems, consulte Instalación, configuración y administración de la aplicación para el monitoreo en la Guía del usuario de Amazon CloudWatch.	7 de agosto de 2019

Nueva característica de la consola: AWS Systems Manager Quick Setup	Quick Setup es una nueva característica de la consola de Systems Manager que le ayuda a configurar rápidamente varios componentes de Systems Manager en sus instancias EC2. En concreto, la Configuración Rápida le ayuda a configurar los siguientes componentes en las instancias que elija o especifique mediante el uso de etiquetas: • Un rol de perfil de instancias de AWS Identity and Access Management (IAM) para Systems Manager • Una actualización bimensual programada de SSM Agent. • Una recopilación programada de metadatos de Inventory cada 30 minutos. • Un análisis diario de las instancias para identificar las revisiones que faltan. • Instalación y configuración puntual del agente de Amazon CloudWatch. • Una actualización mensual programada del agente de CloudWatch.	7 de agosto de 2019

Para obtener más información, consulte Configuración Rápida de AWS Systems Manager.

Registrar un grupo de recursos como objetivo del periodo de mantenimiento	Además de registrar instancias administradas como destino de un periodo de mantenimiento, ahora puede registrar un grupo de recursos como destino de periodo de mantenimiento. Mantenance Windows admite todos los tipos de recursos de AWS compatibles con AWS Resource Groups, que incluye `AWS::EC2::Instance`, `AWS::DynamoDB::Table`, `AWS::OpsWorks::Instance`, `AWS::Redshift::Cluster`, y otros. En esta versión también puede enviar comandos a un grupo de recursos, por ejemplo, mediante el uso de la consola Run Command o el comando AWS CLI send-command. Para obtener más información, consulte los temas siguientes: • Asignar destinos a un período de mantenimiento (consola) • Ejemplos: registrar destinos con un periodo de mantenimiento • Uso de controles de frecuencia y destinos para enviar comandos a una flota	23 de julio de 2019

Creación del paquete simplificado y el control de versiones con AWS Systems Manager Distributor	Distributor tiene un nuevo flujo de trabajo de creación de paquetes, en un formato simplificado que puede generar un manifiesto de paquete, scripts y hashes de archivo. También puede utilizar el flujo de trabajo simplificado al añadir una versión a un paquete existente .	22 de julio de 2019
Nuevo panel de categorías de documentos para Automatización de Systems Manager	Systems Manager incluye un nuevo panel de categorías de documentos cuando ejecuta una automatización en la consola. Utilice este panel para filtrar los manuales de procedimientos de Automation en función de su propósito.	18 de julio de 2019
Soporte para iniciar sesiones de Session Manager con las credenciales de usuario del sistema operativo	De forma predeterminada, Session Manager las sesiones se inician con las credencia les de una `ssm-user` cuenta generada por el sistema que se ha creado en una instancia administrada. En equipos de Linux, ahora puede en su lugar iniciar sesión con las credenciales de un sistema operativo. Para obtener más información, consulte Encender ejecutar como soporte para instancias de Linux.	9 de julio de 2019

Soporte para iniciar sesiones de Session Manager mediante SSH	A partir de ahora, puede utilizar la AWS CLI para iniciar una sesión de SSH en una instancia administrada mediante Session Manager. Para obtener más informaci ón acerca de la activació n de sesiones de SSH con Session Manager, consulte (Opcional) Habilitar sesiones Session Manager de SSH. Para obtener más informaci ón acerca de cómo iniciar una sesión de SSH utilizand o Session Manager, consulte Inicio de una sesión (SSH).	9 de julio de 2019
Soporte para cambiar contraseñas en las instancias administradas	A partir de ahora, puede restablecer las contraseñas en máquinas que administre mediante Systems Manager (instancias administr adas). Puede restablecer la contraseña mediante la consola de Systems Manager o la AWS CLI. Para obtener más información, consulte Restablecimiento de contraseñ as en instancias administr adas.	9 de julio de 2019

Revisiones de "¿Qué es AWS Systems Manager?"	El contenido de introducción de ¿Qué es AWS Systems Manager? se ha ampliado para ofrecer una introducción más amplia al servicio e incluir las herramientas de Systems Manager que se han lanzado recientemente. Además, se ha reubicado el contenido de esta sección en temas individuales para una mejor visibilidad.	10 de junio de 2019

Nueva herramienta de Systems Manager: OpsCenter	OpsCenter proporciona una ubicación central donde los ingenieros de operaciones y los profesionales de TI pueden ver, investigar y resolver los elementos de trabajo operativo (OpsItems) relacionados con los recursos de AWS. OpsCenter está diseñado para reducir el tiempo de resolución de problemas que afectan a los recursos de AWS. Esta herramienta de Systems Manager agrega y estandariza OpsItems en todos los servicios, al mismo tiempo que proporciona datos de investigación contextual sobre cada OpsItem, OpsItems relacionados y recursos relacionados. OpsCenter también proporciona manuales de procedimientos de Automatización de Systems Manager que puede utilizar para resolver problemas rápidamente. Puede especificar datos que se pueden buscar y personalizar para cada OpsItem. También puede ver informes de resumen generados automáticamente sobre OpsItems por estado y origen. Para obtener más información, consulte AWS Systems ManagerOpsCenter.	6 de junio de 2019

Cambios en el panel de navegación izquierdo de Systems Manager en la AWS Management Console	El panel de navegación izquierdo de Systems Manager en la AWS Management Console contiene encabezados nuevos, entre los que se incluye un encabezado nuevo para Centro de operaciones, que proporciona una agrupación más lógica de las herramientas de Systems Manager.	6 de junio de 2019

Tutorial revisado para crear y configurar un periodo de mantenimiento mediante la AWS CLI	El Tutorial: crear y configurar un período de mantenimiento (AWS CLI) se ha revisado para simplificar los pasos de práctica. Cree un único periodo de mantenimiento, identifique un solo destino y configure una tarea simple para ejecutar en el periodo de mantenimiento. En este tutorial proporcionamos información y ejemplos que puede utilizar para crear sus propios comandos de registro de tareas, incluida información para utilizar pseudoparámetros como, por ejemplo, `{{TARGET_ID}}` . Para obtener información y ejemplos adicionales, consulte los siguientes temas: • Ejemplos: registrar destinos con un periodo de mantenimiento • Ejemplos: registrar tareas en un periodo de mantenimiento • Información sobre las opciones register-task-with-maintenance-windows • Utilización de pseudoparámetros en el registro de las tareas del periodo de mantenimiento	31 de mayo de 2019

Notificaciones sobre las actualizaciones de SSM Agent	Si desea recibir notificaciones sobre actualizaciones de SSM Agent, suscríbase a la página de SSM Agent Release Notes en GitHub.	24 de mayo de 2019
Recepción de notificaciones o activación de acciones en función de cambios en Parameter Store	El tema Configuración de notificaciones o activación de acciones según eventos de Parameter Store ahora lo ayuda a configurar reglas de Amazon EventBridge para responder a cambios en Parameter Store. Puede recibir notificaciones o activar otras acciones cuando se produce alguna de las siguientes situaciones: • Cuando se crea un parámetro, se actualiza o se elimina. • Cuando se crea una versión de etiqueta de parámetro, se actualiza o se elimina. • Cuando vence un parámetro , va a vencer o no ha cambiado en un periodo de tiempo específico.	22 de mayo de 2019

Revisiones importantes del contenido sobre configuración e introducción	Hemos ampliado y reorganizado el contenido de configuración e introducción en la guía del usuario de AWS Systems Manager. El contenido de configuración se ha dividido en dos secciones. Una sección se centra en tareas para configurar Systems Manager y administrar las instancias EC2. La otra se centra en las tareas para configurar Systems Manager y administrar los servidores y las máquinas virtuales locales en un entorno híbrido. Ambas secciones presentan ahora todos los temas de configuración como pasos importantes numerados y en el orden recomendado de realización. Un nuevo capítulo de Introducción se centra en ayudar a los usuarios finales a empezar a utilizar Systems Manager después de que las tareas de configuración de los servicios y la cuenta se hayan completado. • Configuración de AWS Systems Manager • Configuración AWS Systems Manager para entornos híbridos • Introducción a AWS Systems Manager	15 de mayo de 2019

Inclusión de revisiones para aplicaciones publicadas por Microsoft en líneas de base de revisiones (Windows)	Patch Manager es ahora compatible con actualizaciones de revisiones para aplicaciones publicadas por Microsoft en instancias de Windows Server. Anteriormente, solo era compatible con revisiones para el sistema operativo Windows Server. Patch Manager proporciona dos líneas de base de revisiones predefinidas para instancias de Windows Server. La línea de base de revisiones `AWS-WindowsPredefinedPatchBaseline-OS` solo se aplica a revisiones del sistema operativo. `AWS-WindowsPredefinedPatchBaseline-OS-Applications` se aplica tanto al sistema operativo y aplicaciones Windows Server publicadas por Microsoft en Windows. Para obtener más información acerca de cómo crear una línea de base de revisiones personalizada que incluya revisiones para aplicaciones publicadas por Microsoft, consulte el primer procedimiento en Creación de una línea de base de revisiones predeterminada. Además, como parte de esta actualiza	7 de mayo de 2019

	ción, también se están modificando los nombres de las líneas de base de revisiones predefinidas proporcionados por AWS. Para obtener más información, consulte Bases de referencia predefinidas.	
Ejemplos para registrar destinos de periodos de mantenimiento mediante la AWS CLI	El nuevo tema Ejemplos: registro de destinos con un periodo de mantenimiento ofrece tres comandos de ejemplo para demostrar distintas formas en las que puede especificar los destinos de un nuevo periodo de mantenimiento cuando se utiliza la AWS CLI. En este tema también se explica el mejor caso de uso para cada uno de los comandos de ejemplo.	3 de mayo de 2019

Actualizaciones de los temas de grupos de revisiones	El tema Acerca de grupos de revisiones se ha actualizado para incluir una sección sobre cómo las instancias administradas determinan la línea de base de revisiones adecuada que se utiliza durante las operaciones de aplicación de revisiones. Además, se han agregado instrucciones para utilizar la AWS CLI o la consola de Systems Manager para agregar las etiquetas Patch Group (Grupo de revisiones) o PatchGroup a las instancias administradas y cómo agregar un Patch Group o PatchGroup a una línea de base de revisiones. (Tiene que usar **`PatchGroup`** , sin espacio, si tiene etiquetas permitidas en metadatos de instancias de EC2). Para obtener más información, consulte la sección sobre cómo crear un grupo de revisiones y añadir un grupo de revisiones a una referencia de revisiones.	1 de mayo de 2019

Nuevas características de Parameter Store	Parameter Store ofrece las siguientes características nuevas: • Parámetros avanzados : Parameter Store ahora le permite configurar los parámetros individualmente para utilizar un nivel de parámetros estándar (el nivel predeterminado) o un nivel de parámetros avanzados. Los parámetros avanzados ofrecen una cuota de tamaño más grande del valor del parámetro, una cuota mayor para el número de parámetros que puede crear por Cuenta de AWS y Región de AWS, y la capacidad de utilizar políticas de parámetros. Para obtener más información acerca de los parámetros avanzados, consulte Acerca de los parámetros avanzados de Systems Manager. • Políticas de parámetros: las políticas de parámetros lo ayudan a administrar un conjunto creciente de parámetros que le permiten asignar criterios específicos a un parámetro, como,	25 de abril de 2019

por ejemplo, una fecha de vencimiento o el periodo de vida. Las políticas de parámetros son especialmente útiles pues le fuerzan a actualizar o eliminar contraseñas y datos de configuración almacenados en Parameter Store. Las políticas de parámetros solo están disponibles para los parámetros que utilizan la capa de parámetros avanzados. Para obtener más información, consulte la sección sobre el uso de políticas de parámetros.

- Mayor rendimiento: ahora ya puede aumentar la cuota de rendimiento de Parameter Store a un máximo de 1000 transacciones por segundo. Para obtener más información, consulte la sección sobre el aumento del rendimiento de Parameter Store.

Actualizaciones de la sección sobre Automation	La sección sobre automatización se ha actualizado para mejorar la visibilidad. Además, se han agregado cuatro temas nuevos a la sección Automatización: • Ejecutar una automatización paso a paso • Ejecución de una automatización que requiere aprobaciones • Programación de automatizaciones con asociaciones de State Manager	17 de abril de 2019

Cifrado de los datos de sesión utilizando una clave de AWS KMS	De forma predeterminada, Session Manager utiliza TLS 1.2 para cifrar los datos de la sesión transmitidos entre los equipos locales de los usuarios de su cuenta y las instancias EC2. Ahora puede elegir cifrar aún más los datos con una AWS KMS key que se ha creado en AWS Key Management Service. Puede utilizar una clave KMS que se haya creado en su Cuenta de AWS o una que hayan compartido con usted desde otra cuenta. Para obtener información acerca de cómo especificar una clave de KMS para cifrar datos de sesiones, consulte Activar el cifrado de datos AWS KMS de sesión con una clave de (consola) , Crear las preferencias de Session Manager (AWS CLI) o Actualizar las preferencias de Session Manager (AWS CLI).	4 de abril de 2019

Configuración de notificaciones de Amazon SNS para AWS Systems Manager	Se han agregado instrucciones para utilizar la AWS CLI o la consola de Systems Manager o para configurar notificaciones de Amazon SNS para tareas de Run Command y Run Command registradas en un periodo de mantenimiento. Para obtener más información, consulte Configuración de notificaciones de Amazon SNS para AWS Systems Manager.	6 de marzo de 2019

Instancias avanzadas para los servidores y las máquinas virtuales en entornos híbridos	AWS Systems Manager ofrece una capa de instancias estándar y una capa de instancias avanzadas para los servidores y las máquinas virtuales en el entorno híbrido. El nivel de instancias estándar le permite registrar un máximo de 1000 servidores o máquinas virtuales por Cuenta de AWS por Región de AWS. Si tiene que registrar más de 1 000 servidores o máquinas virtuales en una única cuenta y región, utilice la capa de instancias avanzadas. Puede crear las instancias que quiera en el nivel de instancias avanzadas, pero todas las instancias configuradas para Systems Manager están disponibles en el modelo de “pago por uso”. Las instancias avanzadas también le permiten conectarse a sus máquinas híbridas mediante AWS Systems Manager Session Manager. Session Manager proporciona acceso mediante shell interactivo a las instancias. Para obtener más información acerca de cómo habilitar las instancias avanzadas, consulte uso del nivel de instancias avanzadas.	4 de marzo de 2019

Creación de asociaciones de State Manager que utilizan documentos de SSM compartidos	Puede crear asociaciones de State Manager que utilizan manuales de procedimientos de Automation y Command de SSM compartidos desde otras Cuentas de AWS. Crear asociaciones con documentos compartidos de SSM lo ayuda a mantener Amazon EC2 y la infraestructura híbrida en un estado coherente incluso si las instancias no se encuentran en la misma cuenta. Para obtener información acerca de cómo compartir documentos de SSM, consulte Documentos de AWS Systems Manager. Para obtener más información acerca de cómo crear una asociación de State Manager, consulte la sección sobre cómo crear una asociación.	28 de febrero de 2019
Ver listas de eventos de Systems Manager compatibles con las reglas de Amazon EventBridge	El nuevo tema Supervisión de eventos de Systems Manager con Amazon EventBridge proporciona un resumen de los distintos eventos emitidos por Systems Manager para los que puede configurar reglas de supervisión de eventos en EventBridge.	25 de febrero de 2019

Agregar etiquetas cuando cree recursos de Systems Manager	Systems Manager ahora permite agregar etiquetas a determinados tipos de recursos cuando los cree. Los recursos que puede etiquetar al crearlos con la AWS CLI o un SDK incluyen periodos de mantenimiento, referencias de revisiones, parámetros de Parameter Store y documentos de SSM. También puede asignar etiquetas a una instancia administrada al crear una activación para ella. Cuando se utiliza la consola de Systems Manager, se pueden agregar etiquetas a periodos de mantenimiento, referencias de revisiones y parámetros.	24 de febrero de 2019

Creación automática de roles de IAM para Systems Manager Inventory	Anteriormente, había que crear un rol de AWS Identity and Access Management (IAM) y asociarle distintas políticas para ver los datos de inventario en la página Inventory Detail View (Vista detallada de Inventory) en la consola. Ya no es necesario crear este rol ni asociarle políticas. Cuando se elige una sincronización de datos remotos en la página Inventory Detail View (Vista detallada de inventario), Systems Manager crea automáticamente el rol `Amazon-GlueServicePolicyForSSM` y le asigna las políticas Amazon-GlueServicePolicyForSSM-{nombre del bucket de S3} y AWSGlueServiceRole. Para obtener más información, consulte Consultas de datos de Inventory de varias regiones y cuentas.	14 de febrero de 2019

Explicaciones de Maintenance Windows para actualizar SSM Agent	Se han agregado dos nuevos tutoriales a la documentación de Maintenance Windows. Las explicaciones especific an en detalle cómo utilizar la consola de Systems Manager o la AWS CLI para crear un periodo de mantenimiento que mantenga SSM Agent actualizado automáticamente. Para obtener más informaci ón, consulte la sección sobre tutoriales de Maintenance Windows.	11 de febrero de 2019
Uso de parámetros públicos de Parameter Store	Se ha añadido una sección breve donde se describen los parámetros públicos de Parameter Store. Para obtener más información, consulte Uso de parámetros públicos de Systems Manager.	31 de enero de 2019

Uso de la AWS CLI para crear preferencias de Session Manager	Se han agregado instrucciones para utilizar la AWS CLI para crear preferencias de Session Manager, por ejemplo, configuraciones de los Registros de CloudWatch, opciones de registros de buckets de S3 y opciones de cifrado de sesión. Para obtener más información, consulte la sección sobre el uso de la AWS CLI para crear preferencias de Session Manager.	22 de enero de 2019

Ejecutar flujos de trabajo de automatización de Systems Manager mediante State Manager	AWS Systems Manager State Manager ahora permite crear asociaciones que utilizan los manuales de procedimientos de Automation de SSM. Anteriormente, State Manager solo era compatible con los documentos `command` y `policy`, lo que significa que solo se podían crear asociaciones dirigidas a instancias administradas. Al ser compatible con los manuales de procedimientos de Automation de SSM, ahora puede crear asociaciones dirigidas a diferentes tipos de recursos de AWS. Para obtener más información, consulte Ejecución de flujos de trabajo de Automatización de Systems Manager mediante State Manager.	22 de enero de 2019

Actualizaciones de la referencia de las expresiones cron y rate y las opciones de programación de periodos de mantenimiento	El tema de referencia Expresiones cron y rate para Systems Manager se ha revisado. La nueva versión proporciona más ejemplos y mejores explicaciones de cómo utilizar las expresiones cron y rate para programar los periodos de mantenimiento y las asociaciones de State Manager. Además, el nuevo tema sobre programación de Maintenance Windows y opciones de periodos activos explica cómo se relacionan entre sí las distintas opciones que tienen que ver con la programación de periodos de mantenimiento (fecha de inicio, fecha de finalización, zona horaria y frecuencia de programación).	6 de diciembre de 2018
Activación del registro de depuración de SSM Agent	Puede habilitar el registro de depuración de SSM Agent editando el archivo seelog.xml.template en la instancia administrada. Para obtener más información, consulte Activación del registro de depuración de SSM Agent.	30 de noviembre de 2018

Compatibilidad con las arquitecturas de procesadores ARM64	AWS Systems Manager ahora admite versiones ARM64 de los sistemas operativos Amazon Linux 2, Red Hat Enterprise Linux 7.6 y Ubuntu Server (18.04 LTS y 16.04 LTS). Para obtener más información, consulte las instrucciones para instalar Amazon Linux 2, RHEL y Ubuntu Server 18.04 y 16.04 LTS con paquetes Snap. Para obtener más informaci ón sobre el tipo de instancia A1, consulte Instancias de uso general en la Guía del usuario de Amazon EC2.	26 de noviembre de 2018

Creación e implementación de paquetes mediante AWS Systems Manager Distributor	Con AWS Systems Manager Distributor, puede empaquetar su propio software (o buscar paquetes de software de agente proporcionados por AWS, como `AmazonCloudWatchAgent`) para instalarlo en las instancias administradas de AWS Systems Manager. Distributor publica los recursos, tales como paquetes de software, en las instancias administradas de AWS Systems Manager. Cuando publica un paquete, se anuncian versiones específicas del documento del paquete (un documento de Systems Manager que crea cuando agrega el paquete en Distributor) a las instancias administradas que identifique mediante los ID de instancia administradas, los ID de Cuenta de AWS, las etiquetas o una Región de AWS. Para obtener más información, consulte AWS Systems ManagerDistributor.	20 de noviembre de 2018

Ejecución simultánea de flujos de trabajo de AWS Systems Manager Automation en diferentes Regiones de AWS y Cuentas de AWS desde una cuenta central	Puede ejecutar flujos de trabajo de automatización de AWS Systems Manager en varias Regiones de AWS y Cuentas de AWS o unidades organizativas de AWS desde una cuenta de administración de Automation. La ejecución de automatizaciones simultáne as en varias regiones y cuentas o UO reduce el tiempo necesario para administrar los recursos de AWS al tiempo que mejoran la seguridad de un entorno informático. Para obtener más información, consulte Ejecución de flujos de trabajo de Automation en varias Regiones de AWS y Cuentas de AWS.	19 de noviembre de 2018

Consulta de datos de inventario de varias Regiones de AWS y Cuentas de AWS	Systems Manager Inventory se integra con Amazon Athena para ayudarlo a consultar los datos de inventario de varias Regiones de AWS y Cuentas de AWS. La integración de Athena utiliza la sincronización de datos de recursos, de modo que podrá ver los datos de inventario de todas las instancias administradas en la página Inventory Detail View (Vista de detalles de Inventory) en la consola de AWS Systems Manager. Para obtener más información, consulte Consultas de datos de Inventory de varias regiones y cuentas.	15 de noviembre de 2018

Creación de asociaciones de State Manager que ejecutan archivos MOF	Puede ejecutar archivos Managed Object Format (MOF) para aplicar un estado de destino en las instancias administradas de Windows Server con State Manager mediante el documento `AWS-ApplyDSCMofs` de SSM. El documento `AWS-ApplyDSCMofs` cuenta con dos modos de ejecución. En el primer modo, puede configurar la asociación para analizar e informar si las instancias administradas se encuentran actualmente en el estado de destino definido en los archivos MOF especificados. En el segundo, puede ejecutar los archivos MOF y cambiar la configuración de las instancias en función de los recursos y sus valores definidos en los archivos MOF. El documento `AWS-ApplyDSCMofs` le permite descargar y ejecutar los archivos de configuración MOF desde Amazon Simple Storage Service (Amazon S3), un recurso compartido o un sitio web seguro con un dominio HTTPS. Para obtener más información, consulte el tema sobre creación de asociaciones que ejecutan archivos MOF.	15 de noviembre de 2018

Restricción del acceso administrativo en sesiones de Session Manager	Las sesiones de Session Manager se inician con las credenciales de una cuenta de usuario que se crea con permisos de administrador o raíz predeterminados llamados `ssm-user`. La información sobre la restricción del control administrativo para esta cuenta ya está disponible en el tema Activación o desactivación de permisos administrativos de cuentas ssm-user.	13 de noviembre de 2018
Ejemplos de YAML en la referencia de acciones de Automation	La referencia de acciones de automatización ahora incluye una muestra de YAML para cada acción que ya incluye una muestra de JSON.	31 de octubre de 2018
Asignación de los niveles de gravedad de conformidad a las asociaciones	Ahora puede asignar niveles de severidad de conformidad a las asociaciones de State Manager. Estos niveles de gravedad se notifican en el panel de conformidad y también se pueden utilizar para filtrar sus informes de conformidad. Los niveles de gravedad que puede asignar son, entre otros, Crítica, Alta, Medio, Baja y Unspecified (Sin especificar). Para obtener más información, consulte Creación de una asociación (consola).	26 de octubre de 2018

Uso los controles de frecuencia y destinos con Automation y State Manager	Controle la ejecución de automatizaciones y las asociaciones de State Manager en su flota de recursos mediante el uso de destinos, simultaneidades y umbrales de error. Para obtener más información, consulte el tema sobre uso de controles de frecuencia y destinos para ejecutar flujos de trabajo de Automation en una flota y el tema sobre uso de controles de frecuencia y destinos con asociaciones de State Manager.	23 de octubre de 2018
Especificación de intervalos de tiempo activos y zonas horarias internacionales para periodos de mantenimiento	También puede especificar fechas antes y después de las cuales no se debe ejecutar un periodo de mantenimiento (fecha de inicio y fecha de finalización), y puede especificar la zona horaria internacional en la que basar la programación del periodo de mantenimiento. Para obtener más información, consulte el tema sobre creación de un periodo de mantenimiento (consola) y el tema sobre actualización de un periodo de mantenimiento (AWS CLI).	9 de octubre de 2018

Mantenimiento de una lista personalizada de revisiones para su línea de base de revisiones en un bucket de S3	Con el nuevo parámetro “InstallOverrideList” en el documento de SSM Command `AWS-RunPatchBaseline`, se puede especificar una URL de https o una URL de tipo ruta de Amazon Simple Storage Service (Amazon S3) a una lista de revisiones que deben instalarse. Esta lista de instalación de revisiones, que mantiene en un bucket de S3 en formato YAML, invalida las revisiones especificadas por la línea de base de revisiones predeterminada. Para obtener más información, consulte Nombre del parámetro: InstallOverrideList.	5 de octubre de 2018
Mayor control sobre si las dependencias de revisiones se instalan	Anteriormente, si una revisión de su lista de revisiones rechazadas se ha identificado como una dependencia de otra revisión, seguiría instalado. Ahora puede elegir si desea instalar estas dependencias o bloquearlas para que no se instalen. Para obtener más información, consulte Creación de una línea de base de revisiones.	5 de octubre de 2018

Creación de flujos de trabajo dinámicos con bifurcaciones condicionales	La acción de Automation de `aws:branch` le permite crear un flujo de trabajo de automatización dinámico que evalúa varias opciones en un solo paso y, a continuación, salta a otro paso en el manual de procedimientos de Automation en función de los resultados de dicha evaluación. Para obtener más información, consulte Uso de instrucciones condicionales en manuales de procedimientos.	26 de septiembre de 2018
Uso de la AWS CLI para actualizar preferencias de Session Manager	Se han agregado a la Guía del usuario de AWS Systems Manager instrucciones para utilizar la CLI para actualizar las preferencias de Session Manager como, por ejemplo, Registros de CloudWatch y opciones de registro de buckets de S3 . Para obtener más información, consulte Uso de la AWS CLI para actualizar las preferencias de Session Manager.	25 de septiembre de 2018

Actualización del requisito de SSM Agent para Session Manager	Session Manager ahora requiere la versión 2.3.68.0 del SSM Agent o posterior. Para obtener más informaci ón acerca de los requisitos previos de Session Manager, consulte Completar requisitos previos de Session Manager.	17 de septiembre de 2018

Administrar las instancias sin abrir los puertos de entrada o mantener hosts bastión mediante Session Manager	Con Session Manager, una herramienta de AWS Systems Manager completamente administrada, puede administrar instancias de EC2 a través de un intérprete de comandos interactivo basado en navegador con un solo clic o con la AWS CLI. Session Manager proporciona una administración de instancias segura y auditable sin la necesidad de abrir los puertos de entrada, mantener hosts bastión o administrar claves SSH. Session Manager también facilita el cumplimiento con las políticas corporativas que requieren acceso controlado a instancias, prácticas de seguridad estrictas y registros completamente auditables con detalles del acceso a las instancias, a la vez que ofrecen a los usuarios finales un acceso multiplataforma sencillo con un solo clic a las instancias de EC2. Para obtener más información, consulte Más información acerca de Session Manager.	11 de septiembre de 2018

Invocación de otros Servicios de AWS de un flujo de trabajo de Automatización de Systems Manager	Puede invocar otros Servicios de AWS y otras herramientas de Systems Manager en su flujo de trabajo de Automatización mediante tres nuevas acciones de Automatización (o complementos) en sus manuales de procedimientos de Automatización. Para obtener más información, consulte Uso de salidas de acción como entradas.	28 de agosto de 2018
Uso de claves de condición específicas de Systems Manager en las políticas de IAM	El tema Especificación de condiciones en una política se ha actualizado para incluir las claves de condición de IAM para Systems Manager que se pueden incorporar en las políticas. Puede utilizar estas claves para especificar las condiciones en las que se debe aplicar una política. El tema también incluye enlaces a ejemplos de políticas y otros temas relacionados.	18 de agosto de 2018

Agregación de datos de Inventory mediante grupos para ver qué instancias están configuradas o no para recopilar un tipo de inventario	Los grupos le permiten ver rápidamente el número de instancias administradas que están o que no están configuradas para recopilar uno o varios tipos de inventario. Con los grupos, debe especificar uno o varios tipos de Inventory y un filtro que utiliza el operador `exists`. Para obtener más información, consulte Agregación de datos de inventario.	16 de agosto de 2018
Visualización del seguimiento de cambios y del historial de Inventory y de Configuration Compliance	Ahora puede ver el seguimiento de cambios y el historial de Inventory recopilado de las instancias administradas. También puede ver el seguimiento de cambios y el historial de la aplicación de revisiones de Patch Manager y de las asociaciones de State Manager notificados por Configuration Compliance. Para obtener más información, consulte Visualización del seguimiento de cambios y del historial de Inventory.	9 de agosto de 2018

Parameter Store se integra con Secrets Manager	Parameter Store ahora está integrado con AWS Secrets Manager, lo que permite recuperar secretos de Secrets Manager cuando utiliza otros Servicios de AWS que admiten las referencias a los parámetros de Parameter Store. Estos servicios incluyen Amazon EC2, Amazon Elastic Container Service, AWS Lambda, AWS CloudFormation, AWS CodeBuild, AWS CodeDeploy y otras herramientas de Systems Manager. Al utilizar Parameter Store para hacer referencia a los secretos de Secrets Manager, se crea un proceso coherente y seguro para llamar y utilizar secretos y datos de referencia en el código y los scripts de configuración. Para obtener más información, consulte Referencia a los secretos de AWS Secrets Manager desde los parámetros de Parameter Store.	26 de julio de 2018

Asociación de etiquetas a los parámetros de Parameter Store	Una etiqueta de parámetro es un alias definido por el usuario que ayuda a administrar las distintas versiones de un parámetro. Cuando se modifica un parámetro, Systems Manager guarda automáticamente una versión nueva e incrementa en uno el número de la versión. Un rótulo puede ayudarle a recordar el propósito de una versión de un parámetro cuando hay varias versiones. Para obtener información, consulte Rotulación de parámetros.	26 de julio de 2018

Creación de flujos de trabajo dinámicos de Automation	De forma predeterminada, los pasos (o acciones) que defina en la sección mainSteps de un manual de procedimientos de Automation se ejecutan en orden secuencial. Cuando finaliza una acción, comienza la siguiente acción especific ada en la sección mainSteps. En esta versión, ahora puede crear flujos de trabajo de Automation que realizan la bifurcación condicional. Esto significa que puede crear flujos de trabajo de Automatio n que respondan de forma dinámica a los cambios en las condiciones y se bifurquen a un paso determinado. Para obtener información, consulte Uso de instrucciones condicionales en manuales de procedimientos.	18 de julio de 2018

SSM Agent ahora preinstalado en AMIs de Ubuntu Server 16.04 mediante Snap	A partir de las instancias creadas basándose en las AMIs de Ubuntu Server 16.04 identificadas con el código `20180627`, SSM Agent se instala previamente utilizando paquetes Snap. En las instancias creadas a partir de AMIs anteriores, debe seguir utilizando paquetes de instalador deb. Para obtener más información, consulte Acerca de las instalaciones del SSM Agent en instancias de Ubuntu Server 16.04 de 64 bits.	7 de julio de 2018
Revisión de los permisos mínimos de S3 necesarios para SSM Agent	El tema nuevo Permisos mínimos del bucket de S3 para SSM Agent proporciona información sobre los buckets de Amazon Simple Storage Service (Amazon S3) a los que los recursos podrían necesitar tener acceso para realizar operaciones de Systems Manager. Puede especificar estos buckets en una política personalizada si desea limitar el acceso a los buckets de S3 para un perfil de instancia o un punto de enlace de la VPC al mínimo requerido para usar Systems Manager.	5 de julio de 2018

Visualización del historial de ejecución completo para un ID de asociación de State Manager específico	En el tema nuevo Visualización de los historiales de asociación, se describe cómo ver todas las ejecuciones de un ID de asociación específico y, a continuación, los detalles de ejecución de uno o varios recursos.	2 de julio de 2018
Patch Manager presenta compatibilidad con Amazon Linux 2	Ahora puede utilizar Patch Manager para aplicar revisiones a las instancias de Amazon Linux 2. Para obtener información general acerca de la compatibilidad de Patch Manager con los sistemas operativos, consulte Requisitos previos de Patch Manager. Para obtener más información acerca de los pares valor de clave compatibles con Amazon Linux 2 cuando define un filtro de revisiones, consulte PatchFilter en la Referencia de la API de AWS Systems Manager.	26 de junio de 2018
Envío de resultados de comandos a los Registros de Amazon CloudWatch	El nuevo tema Configuración de los Registros de Amazon CloudWatch para Run Command describe cómo enviar salida de Run Command a los Registros de CloudWatch.	18 de junio de 2018

Creación o eliminación rápidas de sincronizaciones de datos de recursos para Inventory mediante AWS CloudFormation	Puede utilizar AWS CloudFormation para crear o eliminar una sincronización de datos de recursos para Systems Manager Inventory. Para utilizar AWS CloudFormation, agregue el recurso AWS::SSM::ResourceDataSync a la plantilla de AWS CloudFormation. Para obtener más información, consulte Trabajo con recursos plantillas de AWS CloudFormation en la Guía del usuario de AWS CloudFormation. También puede crear manualmente una sincronización de datos de recursos para Inventory tal y como se describe en Creación de la sincronización de datos de recursos para Inventory.	11 de junio de 2018

Notificaciones de actualización de la Guía del usuario de AWS Systems Manager ya disponibles a través de RSS	La versión HTML de la Guía del usuario de Systems Manager ahora admite una fuente RSS de las actualizaciones que se documentan en la página Historial de actualizaciones de la documentación de Systems Manager. La fuente RSS incluye las actualizaciones realizadas en junio de 2018 y posteriores. Las actualizaciones anunciadas anteriormente siguen estando disponibles en la página Systems Manager documentation update history (Historial de actualizaciones de la documentación en Systems Manager). Utilice el botón RSS del panel del menú superior para suscribirse a la fuente.	6 de junio de 2018
Especificación de un código de salida en los scripts para reiniciar las instancias administradas	En el tema nuevo Reinicio de instancias administradas desde scripts, se describe cómo indicar a Systems Manager que reinicie las instancias administradas especificando un código de salida en los scripts que se ejecutan con Run Command.	3 de junio de 2018

Creación de un evento en Amazon EventBridge cada vez que se elimina un inventario personalizado	En el tema nuevo Visualización de acciones de eliminación de inventario en EventBridge, se describe cómo configurar Amazon EventBridge para que se cree un evento cada vez que usuario elimine un inventario personalizado.	1 de junio de 2018

Actualizaciones anteriores a junio de 2018

En la siguiente tabla, se describen los cambios importantes de cada versión de la Guía del usuario de AWS Systems Manager anteriores a junio de 2018.

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Realizar un inventario de todas las instancias administradas de la Cuenta de AWS	Puede realizar un inventario de todas las instancias administradas de su Cuenta de AWS creando una asociación de inventario global. Para obtener más información, consulte Inventario de todos los nodos administrados de la Cuenta de AWS. **Note** Las asociaciones de inventario globales están disponibles en la versión 2.0.790.0 y versiones posteriores del SSM Agent. Para obtener más información sobre cómo actualizar el SSM Agent en sus instancias, consulte Actualización de SSM Agent mediante Run Command.	3 de mayo de 2018
El SSM Agent se instala de forma	SSM Agent está instalado, de forma predeterminada, en las AMIs de 32 y 64 bits de Ubuntu Server 18.04 LTS.	2 de mayo de 2018

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
predeterminada en Ubuntu Server		
Nuevo tema	En el tema nuevo Ejecución de comandos mediante una versión de documento específica, se explica cómo se utiliza el parámetro document-version para especificar qué versión de un documento de SSM se va a utilizar cuando se ejecuta el comando.	1 de mayo de 2018
Nuevo tema	En el nuevo tema Eliminación de un inventario personalizado se describe cómo eliminar datos de inventario personalizados de Amazon S3 mediante la AWS CLI. También se describe cómo utilizar `SchemaDeleteOption` para administrar un inventario personalizado mediante la desactivación o eliminación de un tipo de inventario personalizado. Esta nueva característica utiliza la operación DeleteInventory de la API.	19 de abril de 2018
Notificaciones de Amazon SNS para SSM Agent	Puede suscribirse a un tema de Amazon SNS para recibir notificaciones cuando haya disponible una nueva versión de SSM Agent. Para obtener más información, consulte Suscripción a las notificaciones de SSM Agent.	9 de abril de 2018
Compatibilidad con aplicación de revisiones en CentOS	Systems Manager ahora es compatible con la aplicación de revisiones en instancias de CentOS. Para obtener más información acerca de las versiones de CentOS admitidas, consulte Requisitos previos de Patch Manager. Para obtener más información sobre cómo funciona la aplicación de revisiones, consulte Cómo funcionan las operaciones de Patch Manager.	29 de marzo de 2018

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Nueva sección sobre	Para ofrecer una única fuente de información de referencia en la Guía del usuario de AWS Systems Manager, se ha introducido una nueva sección, Referencia de AWS Systems Manager. El contenido adicional se agregará a esta sección a medida que esté disponible.	15 de marzo de 2018
Nuevo tema	El tema nuevo Formatos de nombre de paquete para listas de revisiones aprobadas y rechazadas detalla los formatos de nombre de paquetes que se pueden introducir en las Listas de revisiones aprobados y rechazados para una línea de base de revisiones personalizados. Se proporcionan ejemplos de formato para cada tipo de sistema operativo que admite Patch Manager.	9 de marzo de 2018
Nuevo tema	Systems Manager ahora se integra con Chef Chef InSpec. InSpec es un marco de trabajo de tiempo de ejecución de código abierto que permite crear perfiles de lenguaje natural en GitHub o Amazon S3. A continuación, puede utilizar Systems Manager para ejecutar análisis de conformidad y ver cuáles instancias son conformes y cuáles no. Para obtener más información, consulte Utilización de perfiles de Chef InSpec con la conformidad de Systems Manager.	7 de marzo de 2018
Nuevo tema	En el tema nuevo Cómo utilizar roles vinculados a servicios de Systems Manager, se describe cómo utilizar un rol vinculado a servicio de AWS Identity and Access Management (IAM) con Systems Manager. En la actualidad, los roles vinculados a servicios solo son necesarios cuando se utiliza Systems Manager Inventory para recopilar metadatos sobre las etiquetas y los grupos de recursos.	27 de febrero de 2018

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Temas nuevos y actualizados	Ahora puede utilizar Patch Manager para instalar revisiones que se encuentran en un repositorio de origen distinto del repositorio predeterminado configurado en la instancia. Esto resulta útil para aplicar revisiones a las instancias con actualizaciones que no están relacionadas con la seguridad, con el contenido de Personal Package Archives (PPA) para Ubuntu Server, con actualizaciones para aplicaciones corporativas internas, etc. Los repositorios de origen de revisiones alternativos se especifican al crear una línea de base de revisiones personalizada. Para obtener más información, consulte los temas siguientes: • Cómo especificar un repositorio de origen de parches alternativo (Linux) • Uso de bases de referencia de parches personalizadas • Creación de una base de referencia de parches con repositorios personalizados para distintas versiones del sistema operativo Además, ahora puede utilizar Patch Manager para aplicar revisiones a instancias de SUSE Linux Enterprise Server. Patch Manager admite la aplicación de revisiones a SLES de versiones 12.* (solo de 64 bits). Para obtener más información, consulte la información específica de SLES en los siguientes temas: • Cómo se seleccionan las revisiones de seguridad • Cómo se instalan las revisiones • Funcionamiento de las reglas de bases de referencia de parches en SUSE Linux Enterprise Server	6 de febrero de 2018

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Nuevo tema	El nuevo tema Documentos de comando de SSM para la aplicación de revisiones a nodos administrados describe los siete documentos de SSM disponibles actualmente para ayudarlo a mantener sus instancias administradas actualizadas con las últimas revisiones relacionadas con la seguridad.	10 de enero de 2018
Actualizaciones importantes relacionadas con la compatibilidad de Linux	Se han actualizado diversos temas con la información siguiente: • SSM Agent se instala de forma predeterminada en las AMIs base de Amazon Linux 1 desde 09/2017 en adelante. • Instale manualmente SSM Agent en otras versiones de Linux, incluidas las imágenes que no son base, como las AMIs optimizadas para Amazon ECS.	9 de enero de 2018
Nuevo tema	Un nuevo tema, Documento de comandos SSM para aplicar revisiones: `AWS-RunPatchBaseline`, proporciona detalles acerca de cómo este documento de SSM funciona en los sistemas Windows y Linux. También proporciona información acerca de los dos parámetros disponibles en el documento `AWS-RunPatchBaseline`, `Operation` y `Snapshot ID`.	5 de enero de 2018

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Temas nuevos	Una nueva sección, Cómo funcionan las operaciones de Patch Manager, proporciona detalles técnicos que explican cómo Patch Manager determina cuáles son las revisione s de seguridad que deben instalarse y cómo este servicio los instala en todos los sistemas operativos admitidos. También proporciona información acerca de cómo trabajan las reglas de líneas de base de revisiones en diversas distribuciones del sistema operativo Linux.	2 de enero de 2018
Referencia de acciones de Automatización de Systems Manager renombradas y trasladadas	Basándose en comentarios de los clientes, ahora la referencia de acciones de Automatización se denomina referencia de manual de procedimientos de Automatiz ación de Systems Manager. Además, hemos trasladado la referencia al nodo Shared Resources > Documents para que esté más cerca de del Referencia de complementos del documento de comandos. Para obtener más informaci ón, consulte Referencia de acciones de Automatización de Systems Manager.	20 de diciembre de 2017
Nueva capítulo de monitorización y contenido	Un nuevo capítulo, Registro y monitorización en AWS Systems Manager, proporciona instrucciones para enviar métricas y registrar datos en los Registros de Amazon CloudWatch. Un tema nuevo, Envío de registros de nodos a los Registros de CloudWatch (agente de CloudWatc h) unificado, proporciona instrucciones para migrar tareas de monitoreo en la instancia, solo en instancias de Windows Server de 64 bits, desde SSM Agent al agente de CloudWatch.	14 de diciembre de 2017

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Nuevo capítulo	Un nuevo capítulo, Administración de identidades y accesos en AWS Systems Manager, proporciona información completa sobre el uso de AWS Identity and Access Management (IAM) y AWS Systems Manager para ayudar a asegurar el acceso a los recursos mediante credenciales. Estas credenciales proporcionan los permisos necesarios para obtener acceso a recursos de AWS; como, por ejemplo, el acceso a datos almacenados en buckets de S3, el envío de comandos a instancias EC2 o la lectura de etiquetas en dichas instancias.	11 de diciembre de 2017
Cambios en el panel de navegación izquierdo	Hemos cambiado los encabezados del panel de navegación izquierdo de esta guía del usuario para que coincidan con los encabezados de la nueva consola de AWS Systems Manager.	8 de diciembre de 2017
Varios cambios para re: Invent 2017	• Lanzamiento oficial de AWS Systems Manager: AWS Systems Manager (anteriormente Amazon EC2 Systems Manager) es una interfaz unificada que permite centralizar los datos operativos y automatizar tareas en sus recursos de AWS. Puede obtener acceso a la nueva consola de AWS Systems Manager aquí. Para obtener más información, consulte ¿Qué es AWS Systems Manager? • Compatibilidad con YAML: puede crear documentos de SSM en YAML. Para obtener más información, consulte Documentos de AWS Systems Manager.	29 de noviembre de 2017

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Uso de Run Command para tomar instantáneas habilitadas para VSS de volúmenes de EBS	Con Run Command, Puede tomar instantáneas compatibles con la aplicación de todos los volúmenes de Amazon Elastic Block Store (Amazon EBS) que se hayan adjuntado a Windows en instancias de Amazon EC2. El proceso de instantáneas usa el servicio Volume Shadow Copy Service (VSS) de Windows para crear copias de seguridad de nivel de imagen de aplicaciones con reconocimiento de VSS, incluidos datos de transacciones pendientes entre dichas aplicaciones y el disco. Además, no es necesario que apague sus instancias o las desconecte cuando necesite respaldar todos los volúmenes conectados. Para obtener más información, consulte Toma de instantáneas habilitadas con VSS de Microsoft mediante AWS Systems Manager en la Guía del usuario de Amazon EC2.	20 de noviembre de 2017
Seguridad de Systems Manager mejorada disponible usando puntos de enlace de la VPC	Puede mejorar la posición de seguridad de las instancias administradas (incluidas las instancias administradas en su entorno híbrido) configurando Systems Manager para que use un punto de enlace de la VPC de interfaz. Los puntos de enlace de la interfaz tienen tecnología PrivateLink, una tecnología que le permite obtener acceso de forma privada a las API de Amazon EC2 y Systems Manager mediante direcciones IP privadas. PrivateLink restringe todo el tráfico de red entre las instancias administradas, Systems Manager y EC2 a la red de Amazon (las instancias administradas no tienen acceso a Internet). Asimismo, no necesita una gateway de Internet ni un dispositivo NAT ni una gateway privada virtual. Para obtener más información, consulte Mejora de la seguridad de las instancias de EC2 mediante puntos de conexión de VPC para Systems Manager.	7 de noviembre de 2017

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Compatibilidad de Inventory para archivos, servicios, roles de Windows y el Registro de Windows	Ahora, Inventory de SSM es compatible con la recopilación de la siguiente información de sus instancias administradas. • Archivos: nombre, tamaño, versión, fecha de instalación, horas de modificación y último acceso, etc. • Servicios: nombre, nombre de visualización, estado, servicios relacionados, tipo de servicio, tipo de inicio, etc. • Registro de Windows: ruta de la clave del registro, nombre de valor, tipo de valor y valor. • Roles de Windows: nombre, nombre de visualización, ruta, tipo de característica, estado de instalación, etc. Antes de intentar recopilar información para estos tipos de inventario, actualice el SSM Agent de las instancias cuyo inventario desea realizar. Si ejecuta la versión más reciente del SSM Agent, se asegurará de poder recopilar metadatos de todos los tipos de inventario admitidos. Para obtener información acerca de cómo actualizar el SSM Agent mediante State Manager, consulte Tutorial: actualización automática de SSM Agent con AWS CLI. Para obtener más información acerca de Inventory, consulte Más información acerca de Systems Manager Inventory.	6 de noviembre de 2017
Actualizaciones de la documentación de Automation	Se han solucionado varios errores en la información acerca de la instalación y la configuración del acceso de Automatización de Systems Manager. Para obtener más información, consulte Configuración de Automation.	31 de octubre de 2017

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Integración de Amazon S3 y GitHub	Ejecución de scripts remotos: ahora Systems Manager es compatible con la descarga y la ejecución de scripts desde un repositorio de GitHub privado o público y desde Amazon S3. Utilizando el documento de SSM predefinido `AWS-RunRemoteScript` o el complemento `aws:downloadContent` en un documento de SSM personalizado, puede ejecutar manuales de estrategias de Ansible y scripts en Python, Ruby o PowerShell, por citar solo algunos. Estos cambios mejoran aún más la infraestructura como código cuando utiliza Systems Manager para automatizar la configuración y la implementación de las instancias EC2 y las instancias administradas locales en su entorno híbrido. Para obtener más información, consulte Ejecución de scripts desde GitHub y Ejecución de scripts desde Amazon S3. Creación de documentos de SSM compuestos: ahora Systems Manager es compatible con la ejecución de uno o varios documentos de SSM secundarios desde un documento de SSM principal. Estos documentos principales que ejecutan otros documentos se denominan documentos compuestos. Los documentos compuestos le permiten crear y compartir un conjunto estándar de documentos de SSM secundarios a través de Cuentas de AWS para tareas habituales como, por ejemplo, software antivirus de arranque o instancias que se unen a dominios. Puede ejecutar documentos compuestos y secundarios almacenados en Systems Manager, GitHub o Amazon S3. Después de crear un documento compuesto, puede ejecutarlo. Para ello, utilice el documento de SSM predefinido `AWS-RunDocument`. Para obtener más información,	26 de octubre de 2017

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
	consulte Creación de documentos compuestos y Ejecución de documentos de desde ubicaciones remotas. Referencia del complemento de documento de SSM: para facilitar el acceso, hemos sacado la referencia del complemento de SSM de los documentos de SSM fuera de la referencia de la API de Systems Manager y la hemos trasladado a la guía del usuario. Para obtener más información, consulte Referencia de complementos del documento de comandos.	
Compatibilidad con versiones de parámetros en Parameter Store	Al editar un parámetro, ahora Parameter Store itera automáticamente el número de versión por 1. Puede especificar un nombre de parámetro y un número de versión específico en las llamadas a la API y los documentos de &SSM;. Si no especifica un número de versión, el sistema utiliza automáticamente la versión más reciente. Las versiones de los parámetros proporcionan una capa de protección en caso de que un parámetro se cambie por error. Puede ver los valores de todas las versiones y consultar versiones anteriores en caso de que sea necesario. También puede utilizar versiones de parámetro s para ver cuántas veces ha cambiado un parámetro durante un período de tiempo. Para obtener más informaci ón, consulte Trabajo con versiones de parámetros en Parameter Store.	24 de octubre de 2017

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Soporte para etiquetado de documentos de Systems Manager	Ahora puede utilizar la API AddTagsToResource, la AWS CLI o Herramientas de AWS para PowerShell para etiquetar documentos de Systems Manager con pares de valor de clave. El etiquetado le ayuda a identificar rápidamente recursos específicos según las etiquetas que les haya asignado. Esto es algo que se ha añadido a la compatibilidad de etiquetado de antes para instancias administradas, periodos de mantenimiento, parámetros de Parameter Store y referencias de revisiones.	3 de octubre de 2017
Varias actualizaciones de la documentación para corregir errores o actualizar el contenido en función de los comentarios	• Se ha actualizado Administrador de nodos en entornos híbridos y multinube con Systems Manager con información para Raspbian Linux. • Se ha actualizado Administrar instancias de EC2 con Systems Manager con el nuevo requisito para las instancias de Windows Server. SSM Agent necesita Windows PowerShell 3.0 o posterior para ejecutar determinados documentos de SSM en las instancias de Windows Server (por ejemplo, el documento `AWS-ApplyPatchBaseline` heredado). Compruebe que sus instancias de Windows Server ejecutan Windows Management Framework 3.0 o posterior. El marco incluye PowerShell. Para obtener más información, consulte Windows Management Framework 3.0.	2 de octubre de 2017

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Solución de problemas de instancias de Windows inaccesible mediante el flujo de trabajo de EC2Rescue Automation	EC2Rescue puede ayudarlo a diagnosticar y resolver problemas de las instancias de Windows Server de Amazon EC2. Puede ejecutar la herramienta como un flujo de trabajo de Automatización de Systems Manager mediante el documento AWSSupport-ExecuteEC2Rescue . El documento AWSSupport-ExecuteEC2Rescue se ha diseñado para realizar una combinación de acciones de Systems Manager, AWS CloudFormation y funciones de Lambda que automatizan los pasos que normalmente se necesitan para usar EC2Rescue. Para obtener más información, consulte Ejecutar la herramienta EC2Rescue en instancias inaccesibles.	29 de septiembre de 2017
SSM Agent Instalado de forma predeterminada en Amazon Linux	SSM Agent se instala de forma predeterminada en las AMIs de Amazon Linux desde septiembre de 2017.09 y posterior. Instale manualmente SSM Agent en otras versiones de Linux, tal como se describe en Uso de SSM Agent en instancias de EC2 para Linux.	27 de septiembre de 2017

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Mejoras de Run Command	Run Command contiene las mejoras siguientes. • Puede restringir la ejecución de comandos a determina das instancias mediante la creación de una política de IAM que incluya la condición de que el usuario solo pueda ejecutar comandos en las instancias que estén señaladas con etiquetas específicas de Amazon EC2. Para obtener más información, consulte Restricción de acceso de Run Command basado en etiquetas. • Tiene más opciones para la orientación de instancias con etiquetas de Amazon EC2. Ahora puede especific ar varias claves de etiqueta y varios valores de etiqueta al enviar los comandos. Para obtener más información, consulte Ejecución de comandos a escala.	12 de septiembre de 2017
Systems Manager compatible con Raspbian	Ahora Systems Manager se puede ejecutar en dispositivos Raspbian Jessie y Raspbian Stretch, incluido Raspberry Pi (32 bits).	7 de septiembre de 2017
Envío automátic o de registros de SSM Agent a los Registros de Amazon CloudWatc h.	Ahora puede realizar un simple cambio de configuración en sus instancias para que el SSM Agent envíe archivos de registro a CloudWatch. Para obtener más información, consulte Envío de registros de SSM Agent a CloudWatch Logs.	7 de septiembre de 2017

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Cifrado de sincronización de datos de recursos	Con la sincronización de datos de recursos de Systems Manager puede agregar los datos de inventario recopilados en docenas o centenas de instancias administradas en un bucket de S3 central. Ahora puede cifrar la sincronización de datos de recursos mediante una clave de AWS Key Management Service. Para obtener más información, consulte Explicación: cómo utilizar la sincronización de datos de recursos para agregar datos de inventario.	1 de septiembre de 2017
Nuevos tutoriales de State Manager	Se han agregado dos nuevos tutoriales a la documentación de State Manager: Tutorial: actualización automática de SSM Agent con AWS CLI Tutorial: actualización automática de los controladores PV en las instancias de EC2 de Windows Server	31 de agosto de 2017
Systems Manager Configuration Compliance	Utilice Configuration Compliance para analizar su flota de instancias administradas en busca de incoherencias en la conformidad de revisiones y la configuración. Puede recopilar y agregar datos de varias Cuentas de AWS y Regiones de AWS, y luego desglosarlas en recursos específicos que no sean conformes. De forma predeterminada, Configuration Compliance muestra datos de conformidad sobre las revisiones de Patch Manager y asociaciones de State Manager. También puede personalizar el servicio y crear sus propios tipos de conformidad en función de sus requisitos empresariales o de TI Para obtener más información, consulte Conformidad de AWS Systems Manager.	28 de agosto de 2017

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Nueva acción de Automation: `aws:executeAutomation`	Ejecuta un flujo de trabajo secundario de Automation mediante la llamada a un manual de procedimientos de Automation secundario. Con esta acción puede crear manuales de procedimientos de Automation para la mayoría de los flujos de trabajo más comunes y hacer referencia a esos documentos durante una ejecución de Automation. Esta acción puede simplificar los manuales de procedimientos de Automation mediante la eliminación de la necesidad de duplicar pasos en manuales similares. Para obtener más información, consulte `aws:executeAutomation` : ejecutar otra automatización.	22 de agosto de 2017
Automation como el destino de un evento de CloudWatch	Puede iniciar un flujo de trabajo de Automation especificando un manual de procedimientos de Automation como el objetivo de un evento de Amazon CloudWatch. Puede iniciar los flujos de trabajo según una programación o cuando se produzca un evento del sistema de AWS específico. Para obtener más información, consulte Ejecución de automatizaciones a partir de eventos EventBridge.	21 de agosto de 2017
Control de versiones de asociaciones de State Manager y actualizaciones generales	Ahora puede crear diferentes versiones de asociaciones de State Manager. Existe una cuota de 1000 versiones de cada asociación. También puede especificar nombres para sus asociaciones. Además, la documentación de State Manager se ha actualizado para abordar la información obsoleta y las incoherencias. Para obtener más información, consulte AWS Systems Manager State Manager.	21 de agosto de 2017

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Cambios en Maintenance Windows	En Maintenance Windows se han incluido los siguientes cambios o mejoras: • Antes Maintenance Windows solo podía realizar tareas con Run Command. Ahora puede realizar las tareas con la Automatización de Systems Manager, AWS Lambda y AWS Step Functions. • Puede editar los destinos de un periodo de mantenimi ento, así como especificar un nombre de destino, una descripción y un propietario. • Puede editar las tareas de un periodo de mantenimiento, como, por ejemplo, especificar un nuevo documento de SSM para Run Command y tareas de automatización. • Ahora se admiten todos los parámetros de Run Command, incluidos DocumentHash, DocumentH ashType, TimeoutSeconds, Comment y Notificat ionConfig. • Ahora puede utilizar una marca `safe` cuando intente cancelar el registro de un destino. Si se activa, el sistema devuelve un error si una tarea hace referencia al destino. Para obtener más información, consulte AWS Systems Manager Maintenance Windows.	16 de agosto de 2017

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Nueva acción de Automation: `aws:approve`	Esta nueva acción para los manuales de procedimientos de Automation detiene temporalmente una ejecución de Automation hasta que las entidades principales designadas aprueben o rechacen la acción. Después de que se alcance el número necesario de aprobaciones, se reanuda la ejecución de Automation. Para obtener más información, consulte Referencia de acciones de Automatización de Systems Manager.	10 de agosto de 2017
El rol de asunción de Automatización ya no es necesario	Anteriormente, Automation requería especificar un rol de servicio (o un rol de asunción) para que el servicio tuviera permiso para realizar acciones en su nombre. Automation ya no requiere este rol porque el servicio ahora funciona mediante el contexto del usuario que ha invocado la ejecución. Sin embargo, en las siguientes situaciones sigue siendo necesario especificar un rol de servicio para Automation: • Cuando desea restringir los permisos de un usuario para un recurso, pero desea que el usuario ejecute un flujo de trabajo de Automation que requiera permisos elevados. En este caso, se puede crear un rol de servicio con permisos elevados y permitir al usuario que ejecute el flujo de trabajo. • Las operaciones cuya ejecución se prevé superior a las 12 horas requieren un rol de servicio. Para obtener más información, consulte Configuración de Automation.	3 de agosto de 2017

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Configuration Compliance	Utilice Amazon EC2 Systems Manager Configuration Compliance para analizar su flota de instancias administradas en busca de incoherencias en la conformidad de revisiones y la configuración. Puede recopilar y agregar datos de varias Cuentas de AWS y Regiones de AWS, y luego desglosarlas en recursos específicos que no sean conformes. Para obtener más información, consulte Conformidad de AWS Systems Manager.	8 de agosto de 2017
Mejoras en los documentos de SSM	Los documentos de comando y de política de SSM ahora ofrecen compatibilidad multiplataforma. Esto quiere decir que un solo documento de SSM puede procesar complementos para los sistemas operativos Windows y Linux. La compatibilidad multiplataforma le permite consolidar el número de documentos que administra. La compatibilidad multiplataforma se ofrece en los documentos de SSM que usan la versión de esquema 2.2 o posterior. Los documentos de SSM Command que utilizan la versión de esquema 2.0 o posterior ahora pueden incluir varios complementos del mismo tipo. Por ejemplo, puede crear un documento de Command que llame al complemento `aws:runRunShellScript` varias veces. Para obtener más información sobre los cambios de la versión 2.2 del esquema, consulte Documentos de AWS Systems Manager. Para obtener más información sobre los complementos de SSM, consulte Referencia de los complementos del documento de comandos.	12 de julio de 2017

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
revisiones de Linux	Patch Manager ahora puede aplicar revisiones a las siguientes distribuciones de Linux: Sistemas de 64 y 32 bits • Amazon Linux 2014.03, 2014.09 o posterior • Ubuntu Server 16.04 LTS, 14.04 LTS o 12.04 LTS • Red Hat Enterprise Linux (RHEL) 6.5 o posterior Solo sistemas de 64 bits • Amazon Linux 2015.03, 2015.09 o posterior • Red Hat Enterprise Linux (RHEL) 7.x o posterior Para obtener más información, consulte AWS Systems Manager Patch Manager. Note • Para revisar las instancias de Linux, estas deben ejecutar el SSM Agent, versión 2.0.834.0 o posterior. Para obtener información sobre cómo actualizar el agente, consulte el apartado titulado Ejemplo: actualizar el SSM Agent en Ejecución de comandos desde la consola. • El documento `AWS-ApplyPatchBaseline` de SSM se ha reemplazado por el documento `AWS-RunPatchBaseline` .	6 de julio de 2017

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
Sincronización de datos de recursos	Puede utilizar la sincronización de datos de recursos de Systems Manager para enviar datos de inventario recopilados de todas las instancias administradas a un solo bucket de Amazon S3. A continuación, la sincroniz ación de datos de recursos actualiza automáticamente los datos centralizados cuando se recopilan los nuevos datos de Inventory. Con todos los datos de inventario almacenados en un bucket de S3 de destino, puede utilizar servicios como Amazon Athena y Amazon QuickSight para consultar y analizar los datos agregados. Para obtener más información, consulte Cómo crear una sincronización de datos de recursos para Inventory. Para ver un ejemplo de cómo trabajar con la sincronización de datos de recursos, consulte Explicación: cómo utilizar la sincronización de datos de recursos para agregar datos de inventario.	29 de junio de 2017
Jerarquías de parámetros de Systems Manager	La administración de docenas o centenas de parámetros de Systems Manager como una lista sin formato requiere mucho tiempo y es propensa a errores. Puede utilizar las jerarquías de parámetros como ayuda para organizar y administrar los parámetros de Systems Manager. Una jerarquía es un nombre de parámetro que incluye una ruta definida mediante barras. A continuación se muestra un ejemplo que utiliza tres niveles de jerarquía en el nombre para identificar lo siguiente: /Entorno/Tipo de equipo/Aplicación/Datos `/Dev/DBServer/MySQL/db-string13` Para obtener más información, consulte Trabajo con jerarquías de parámetros en Parameter Store.	22 de junio de 2017

Cambio	Descripción	Fecha de lanzamiento de la nueva versión
SSM Agent Compatibilidad con SUSE Linux Enterprise Server	Puede instalar SSM Agent en SUSE Linux Enterprise Server (SLES) de 64 bits. Para obtener más información, consulte Uso de SSM Agent en instancias de EC2 para Linux.	14 de junio de 2017

Convenciones del documento

A continuación, se muestran las convenciones tipográficas comunes para la Guía del usuario de AWS Systems Manager.

Ejemplos diferenciados para sistemas operativos locales o lenguajes de línea de comandos

Usamos pestañas para presentar diferentes ejemplos de comandos en función del tipo de sistema operativo local de un usuario. La barra invertida (\) se utiliza en ejemplos de Linux y macOS para dividir comandos largos en varias líneas. En los ejemplos de Windows Server, utilizamos el signo de intercalación (^) para dividir los comandos en varias líneas.

Ejemplo:

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
```

Elementos de la interfaz de usuario

Formato: texto en negrita

Ejemplo: Elija File, Properties.

Entrada del usuario (texto que escribe un usuario)

Formato: texto en una fuente monoespaciada

Ejemplo: para el nombre, escriba **`my-new-resource`**.

Texto de marcador de posición de un valor obligatorio

Formato: texto en *`cursiva`*

Ejemplo:

```
aws ec2 register-image --image-location amzn-s3-demo-bucket/image.manifest.xml
```