aws

Panduan Pengguna

AWS Systems Manager

AWS Systems Manager: Panduan Pengguna

Table of Contents

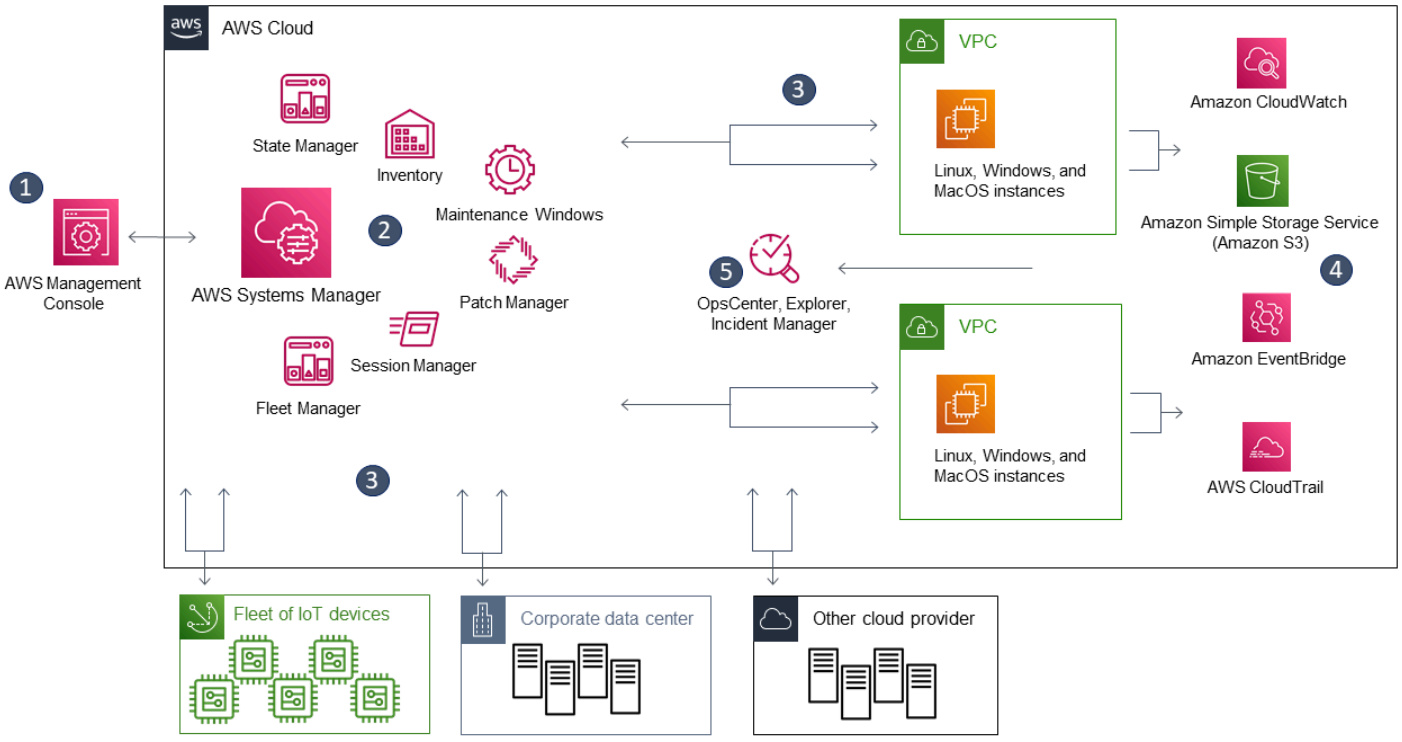
Apakah AWS Systems Manager itu?

AWS Systems Manageradalah hub operasi untuk AndaAWSaplikasi dan sumber daya serta solusi manajemen end-to-end yang aman untukhibrida dan multicloudlingkungan yang memungkinkan operasi aman dalam skala besar.

Cara kerja Systems Manager

Diagram berikut menjelaskan bagaimana beberapa kemampuan Manajer Sistem melakukan tindakan pada sumber daya Anda. Diagram tidak mencakup semua kemampuan. Setiap interaksi yang disebutkan dijelaskan sebelum diagram.

1. Manajer Sistem Akses- Gunakan salah satu opsi yang tersedia untukmengakses Manajer Sistem.
2. Pilih kemampuan Manajer Sistem- Tentukan kemampuan mana yang dapat membantu Anda melakukan tindakan yang ingin Anda lakukan pada sumber daya Anda. Diagram hanya menunjukkan beberapa kemampuan yang administrator TI danDevOpspersonil digunakan untuk mengelola aplikasi dan sumber daya mereka.
3. Verifikasi dan pemrosesan- Manajer Sistem memverifikasi bahwa pengguna, grup, atau peran Anda memiliki yang diperlukanAWS Identity and Access Management(IAM) izin untuk melakukan tindakan yang Anda tentukan. Jika target tindakan Anda adalah node yang dikelola, Agen Manajer Sistem (SSM Agent) berjalan pada node melakukan tindakan. Untuk jenis sumber daya lainnya, Manajer Sistem melakukan tindakan tertentu atau berkomunikasi dengan lainnyaLayanan AWSuntuk melakukan tindakan atas nama Manajer Sistem.
4. Pelaporan- Manajer Sistem,SSM Agent, dan lainnyaLayanan AWSyang melakukan tindakan atas nama status laporan Manajer Sistem. Manajer Sistem dapat mengirim detail status ke yang lainLayanan AWS, jika dikonfigurasi.
5. Kemampuan manajemen operasi Manajer Sistem- Jika diaktifkan, kemampuan manajemen operasi Manajer Sistem sepertiExplorer,OpsCenter, dan Incident Manager mengumpulkan data operasi atau membuat artefak sebagai respons terhadap peristiwa atau kesalahan dengan sumber daya Anda. Artefak ini termasuk item pekerjaan operasional (OpsItems) dan insiden. Kemampuan manajemen operasi Manajer Sistem memberikan wawasan operasional tentang aplikasi dan sumber daya Anda serta solusi remediasi otomatis untuk membantu memecahkan masalah.



Kemampuan Systems Manager

Manajer Sistem mengelompokkan kemampuan ke dalam kategori berikut. Pilih tab di bawah setiap kategori untuk mempelajari lebih lanjut tentang setiap kemampuan.

Topik

- Manajemen aplikasi
- Manajemen perubahan
- Manajemen node
- Manajemen operasi
- Quick Setup
- Sumber Daya Bersama

Manajemen aplikasi

Application Manager

Application ManagermenolongDevOpsinsinyur menyelidiki dan memperbaiki masalah dengan merekaAWSsumber daya dalam konteks aplikasi dan cluster mereka. DalamApplication

Manager, sebuahpenerapanadalah kelompok logisAWSsumber daya yang ingin Anda operasikan sebagai satu unit. Kelompok logis ini dapat mewakili versi yang berbeda dari aplikasi, batas kepemilikan untuk operator, atau lingkungan pengembang, untuk beberapa nama.Application Managerdukungan untuk klaster kontainer mencakup klaster Amazon Elastic Kubernetes Service (Amazon EKS) dan Amazon Elastic Container Service (Amazon ECS).Application Manageragregat informasi operasi dari beberapaLayanan AWSdan kemampuan System Manager untuk satuAWS Management Console.

AppConfig

AppConfigmembantu Anda membuat, mengelola, dan menyebarkan konfigurasi aplikasi dan tanda fitur.AppConfigmendukung penyebaran terkontrol untuk aplikasi dari berbagai ukuran. Anda dapat menggunakanAppConfigdengan aplikasi yang dihosting di instans Amazon EC2,AWS Lambdakontainer, aplikasi mobile, atau perangkat tepi. Untuk mencegah kesalahan saat menerapkan konfigurasi aplikasi,AppConfigtermasuk validator. Sebuah validator menyediakan pemeriksaan sintaksis atau semantik untuk memverifikasi bahwa konfigurasi yang ingin Anda deploy berfungsi sebagaimana yang dimaksud. Selama penerapan konfigurasi,AppConfigmemonitor aplikasi untuk memverifikasi bahwa penyebaran berhasil. Jika sistem menemukan kesalahan atau jika penyebaran memanggil alarm,AppConfigmengembalikan perubahan untuk meminimalkan dampak bagi pengguna aplikasi Anda.

Penyimpanan Parameter

Parameter Storemenyediakan penyimpanan hierarkis yang aman untuk data konfigurasi dan manajemen rahasia. Anda dapat menyimpan data seperti kata sandi, string database, ID instans Amazon Elastic Compute Cloud (Amazon EC2) dan ID Amazon Machine Image (AMI), dan kode lisensi sebagai nilai parameter. Anda dapat menyimpan nilai sebagai teks biasa atau data terenkripsi. Anda kemudian dapat mereferensi nilai dengan menggunakan nama unik yang Anda tentukan ketika Anda membuat parameter.

Manajemen perubahan

Change Manager

Change Manageradalah kerangka kerja manajemen perubahan perusahaan untuk meminta, menyetujui, menerapkan, dan melaporkan perubahan operasional pada konfigurasi dan infrastruktur aplikasi Anda. Dari satu akun administrator yang didelegasikan, jika Anda menggunakan AWS Organizations, Anda dapat mengelola perubahan di beberapa Akun AWS dalam beberapa Wilayah AWS. Atau, menggunakan akun lokal, Anda dapat mengelola

perubahan untuk satu Akun AWS. GunakanChange Manageruntuk mengelola perubahan keduanyaAWSsumber daya dan sumber daya lokal.

Automation

GunakanOtomatisasiuntuk mengotomatiskan tugas pemeliharaan dan penyebaran umum. Anda dapat menggunakan Otomatisasi untuk membuat dan memperbarui Amazon Machine Images (AMIs), menerapkan driver dan pembaruan agen, mengatur ulang kata sandi pada instans Windows Server, mengatur ulang kunci SSH pada instans Linux, dan menerapkan patch OS atau pembaruan aplikasi.

Change Calendar

Change Calendarmembantu Anda mengatur rentang tanggal dan waktu saat tindakan yang Anda tentukan (misalnya, diOtomasi Manajer Sistemrunbook) dapat atau tidak dapat dilakukan diAkun AWS. DalamChange Calendar, rentang ini disebutacara. Saat Anda membuatChange Calendarentri, Anda membuatDokumen Manajer Sistemdari jenis`ChangeCalendar`. DalamChange Calendar, toko dokumeniCalendar 2.0data dalam format plaintext. Peristiwa yang Anda tambahkan keChange Calendarentri menjadi bagian dari dokumen. Anda dapat menambahkan acara secara manual diChange Calendarantarmuka atau impor peristiwa dari kalender pihak ketiga yang didukung menggunakan`.ics`berkas.

Jendela Pemeliharaan

GunakanMaintenance Windowsuntuk mengatur jadwal berulang untuk instans terkelola untuk menjalankan tugas administratif seperti menginstal patch dan pembaruan tanpa mengganggu operasi bisnis kritis.

Manajemen node

SEBUAHnode yang dikelolaadalah mesin apa pun yang dikonfigurasi untuk digunakan dengan Manajer Sistem dihibrida dan multicloudlingkungan.

Compliance

GunakanKepatuhanuntuk memindai armada node terkelola Anda untuk kepatuhan patch dan inkonsistensi konfigurasi. Anda dapat mengumpulkan dan menggabungkan data dari beberapa Akun AWS dan Wilayah AWS, lalu menelusuri ke sumber daya tertentu yang tidak sesuai. Secara default, Kepatuhan menampilkan data kepatuhanPatch Managermenambal danState Managerasosiasi. Anda juga dapat menyesuaikan layanan dan membuat jenis kepatuhan Anda sendiri berdasarkan IT Anda atau persyaratan bisnis.

Fleet Manager

Fleet Manageradalah pengalaman antarmuka pengguna terpadu (UI) yang membantu Anda mengelola node dari jarak jauh. DenganFleet Manager, Anda dapat melihat status kesehatan dan kinerja seluruh armada Anda dari satu konsol. Anda juga dapat mengumpulkan data dari masing-masing perangkat dan instans untuk melakukan pemecahan masalah umum dan tugas manajemen dari konsol. Ini termasuk melihat direktori dan isi file, manajemen registri Windows, manajemen pengguna sistem operasi, dan lebih banyak lagi.

Inventory

Persediaanmengotomatiskan proses pengumpulan inventaris perangkat lunak dari node terkelola Anda. Anda dapat menggunakan Inventory untuk mengumpulkan metadata tentang aplikasi, file, komponen, patch, dan lainnya.

Manajer Sesi

GunakanSession Manageruntuk mengelola perangkat edge Anda dan instans Amazon Elastic Compute Cloud (Amazon EC2) melalui shell berbasis browser satu klik interaktif atau melaluiAWS CLI.Session Managermenyediakan perangkat tepi dan manajemen instans yang aman dan dapat diaudit tanpa perlu membuka port masuk, memelihara host benteng, atau mengelola kunci SSH.Session Managerjuga memungkinkan Anda mematuhi kebijakan perusahaan yang memerlukan akses terkontrol ke perangkat dan instans edge, praktik keamanan yang ketat, dan log yang sepenuhnya dapat diaudit dengan detail akses perangkat edge dan instans, sambil tetap memberikan akses lintas platform satu klik sederhana kepada pengguna akhir ke perangkat edge dan instans EC2 Anda. Untuk menggunakanSession Manager, Anda harus mengaktifkan tingkat lanjutan-lanjutan. Untuk informasi selengkapnya, lihat Mengaktifkan tingkat instans lanjutan.

Run Command

GunakanRun Commanduntuk mengelola konfigurasi node terkelola dari jarak jauh dan aman dalam skala besar. GunakanRun Commanduntuk melakukan perubahan sesuai permintaan seperti memperbarui aplikasi atau menjalankan skrip shell Linux danWindows PowerShellperintah pada target set puluhan atau ratusan node dikelola.

State Manager

GunakanState Manageruntuk mengotomatiskan proses menjaga node terkelola Anda dalam keadaan yang ditentukan. Anda dapat menggunakanState Manageruntuk menjamin bahwa node terkelola Anda di-bootstrap dengan perangkat lunak tertentu saat startup, bergabung keWindowsdomain (Windows Servernode saja), atau ditambal dengan pembaruan perangkat lunak tertentu.

Patch Manager

GunakanPatch Manageruntuk mengotomatiskan proses menambal node terkelola Anda dengan pembaruan terkait keamanan dan jenis pembaruan lainnya. Anda dapat menggunakanPatch Manageruntuk menerapkan tambalan untuk sistem operasi dan aplikasi. (Pada Windows Server, dukungan aplikasi terbatas pada pembaruan untuk aplikasi yang dirilis oleh Microsoft.)

Kemampuan ini memungkinkan Anda memindai node terkelola untuk patch yang hilang dan menerapkan patch yang hilang satu per satu atau ke grup besar node yang dikelola dengan menggunakan tag.Patch Managermenggunakangaris dasar patch, yang dapat mencakup aturan untuk patch yang menyetujui otomatis dalam beberapa hari sejak rilis, dan daftar patch yang disetujui dan ditolak. Anda dapat menginstal patch keamanan secara teratur dengan menjadwalkan patching untuk dijalankan sebagai tugas jendela pemeliharaan Manajer Sistem, atau Anda dapat menambal node yang dikelola sesuai permintaan kapan saja.

Untuk sistem operasi Linux, Anda dapat menentukan repositori yang seharusnya digunakan untuk operasi patch sebagai bagian dari dasar patch Anda. Ini memungkinkan Anda untuk memastikan bahwa pembaruan diinstal hanya dari repositori tepercaya terlepas dari repositori apa yang dikonfigurasi pada node yang dikelola. Untuk Linux, Anda juga memiliki kemampuan untuk memperbarui paket apa pun pada node yang dikelola, bukan hanya yang diklasifikasikan sebagai pembaruan keamanan sistem operasi. Anda juga dapat membuat laporan patch yang dikirim ke bucket S3 pilihan Anda. Untuk satu node terkelola, laporan menyertakan detail semua patch untuk mesin. Untuk laporan pada semua node yang dikelola, hanya ringkasan berapa banyak patch yang hilang disediakan.

Distributor

GunakanDistributoruntuk membuat dan menyebarkan paket ke node yang dikelola. DenganDistributor, Anda dapat mengemas perangkat lunak Anda sendiri — atau menemukanAWSpaket perangkat lunak agen -provided, sepertiAmazonCloudWatchAgent— untuk menginstal pada node yang dikelola Manajer Sistem. Setelah Anda menginstal paket untuk pertama kalinya, Anda dapat menggunakanDistributoruntuk menghapus dan menginstal ulang versi paket baru, atau melakukan pembaruan di tempat yang menambahkan file baru atau yang diubah.Distributormenerbitkan sumber daya, seperti paket perangkat lunak, ke node yang dikelola Manajer Sistem.

Hybrid Activations

Untuk menyiapkan mesin non-EC2 di lingkungan hybrid dan multicloud Anda sebagai node terkelola, buataktivasi hibrida. Setelah Anda menyelesaikan aktivasi, Anda menerima kode

aktivasi dan ID. Kombinasi kode dan ID ini berfungsi seperti ID akses Amazon Elastic Compute Cloud (Amazon EC2) dan kunci rahasia untuk memberikan akses aman ke layanan Manajer Sistem dari instans terkelola Anda.

Anda juga dapat membuat aktivasi untuk perangkat edge jika Anda ingin mengelolanya dengan menggunakan Systems Manager.

Manajemen operasi

Incident Manager

Incident Manager adalah konsol manajemen insiden yang membantu pengguna mengurangi dan memulihkan dari insiden yang memengaruhi aplikasi yang di-host AWS.

Incident Manager meningkatkan resolusi insiden dengan memberi tahu responden tentang dampak, menyoroti data pemecahan masalah yang relevan, dan menyediakan alat kolaborasi untuk membuat layanan kembali aktif dan berjalan. Incident Manager juga mengotomatisasi rencana respon dan memungkinkan eskalasi tim responder.

Explorer

Explorer adalah dasbor operasi yang dapat disesuaikan yang melaporkan informasi tentangAWSsumber daya.Explorermenampilkan tampilan agregat data operasi (OpsData) untuk AndaAkun AWSdan di seberangWilayah AWS. DalamExplorer,OpsDatamenyertakan metadata tentang instans Amazon EC2 Anda, detail kepatuhan patch, dan item pekerjaan operasional (OpsItems).Explorermenyediakan konteks tentang bagaimanaOpsItemsdidistribusikan di seluruh unit bisnis Anda atau aplikasi, bagaimana mereka tren dari waktu ke waktu, dan bagaimana mereka bervariasi menurut kategori. Anda dapat mengelompokkan dan memfilter informasi diExploreruntuk fokus pada item yang relevan bagi Anda dan yang memerlukan tindakan. Saat Anda mengidentifikasi masalah prioritas tinggi, Anda dapat menggunakannyaOpsCenter, kemampuan Manajer Sistem, untuk menjalankan runbook Otomasi dan menyelesaikan masalah tersebut.

OpsCenter

OpsCentermenyediakan lokasi pusat di mana insinyur operasi danITprofesional dapat melihat, menyelidiki, dan menyelesaikan item pekerjaan operasional (OpsItems) terkait denganAWSsumber daya.OpsCenterdirancang untuk mengurangi waktu rata-rata untuk resolusi untuk masalah yang berdampakAWSsumber daya. Kemampuan Manajer Sistem

ini mengumpulkan dan menstandarisasiOpsItemslintas layanan sambil memberikan data investigasi kontekstual tentang masing-masingOpsItem, terkaitOpsItems, dan sumber daya terkait.OpsCenterjuga menyediakan runbook Systems Manager Automation yang dapat Anda gunakan untuk menyelesaikan masalah. Anda dapat menentukan dicari, data kustom untuk masing-masingOpsItem. Anda juga dapat melihat laporan ringkasan yang dibuat secara otomatisOpsItemsberdasarkan status dan sumber.

CloudWatch Dashboards

AmazonCloudWatchDasboradalah halaman yang dapat disesuaikan diCloudWatchkonsol yang dapat Anda gunakan untuk memantau sumber daya Anda dalam satu tampilan, bahkan sumber daya yang tersebar di berbagai wilayah. Anda dapat menggunakanCloudWatchdasbor untuk membuat tampilan metrik dan alarm yang disesuaikan untukAWSsumber daya.

Quick Setup

GunakanQuick Setupuntuk mengkonfigurasi sering digunakanLayanan AWSdan fitur dengan praktik terbaik yang direkomendasikan. Anda dapat menggunakanQuick Setupdalam individuAkun AWSatau di beberapaAkun AWSdanWilayah AWSdengan mengintegrasikan denganAWS Organizations.Quick Setupmenyederhanakan pengaturan layanan, termasuk Manajer Sistem, dengan mengotomatiskan tugas umum atau yang direkomendasikan. Tugas-tugas ini termasuk, misalnya, membuat peran profil instans (IAM) AWS Identity and Access Management dan mengatur praktik terbaik operasional, seperti pemindaian patch berkala dan pengumpulan inventaris.

Sumber Daya Bersama

Documents

Dokumen Systems Manager (dokumen SSM) menentukan tindakan yang dilakukan Systems Manager. Jenis dokumen SSM meliputiPerintahdokumen, yang digunakan olehState ManagerdanRun Command, dan runbook Otomasi, yang digunakan oleh Systems Manager Automation. Systems Manager mencakup puluhan dokumen pra-konfigurasi yang dapat Anda gunakan dengan menentukan parameter di runtime. Dokumen dapat dinyatakan dalam JSON atau YAML, dan menyertakan langkah-langkah dan parameter yang Anda tentukan.

Mengakses Systems Manager

Anda dapat bekerja dengan Manajer Sistem dengan salah satu cara berikut:

Konsol Systems Manager

YangKonsol Manajer Sistemadalah antarmuka berbasis browser untuk mengakses dan menggunakan Systems Manager.

Konsol AWS IoT Greengrass V2

Anda dapat melihat dan mengelola perangkat edge yang dikonfigurasiAWS IoT Greengrassdi dalamKonsol Greengrass.

Alat baris perintah AWS

Dengan menggunakan alat baris perintah AWS, Anda dapat mengeluarkan perintah pada baris perintah sistem untuk menjalankan Systems Manager dan tugas AWS lainnya. Alat-alat yang didukung di Linux, macOS, dan Windows. Menggunakan AWS Command Line Interface (AWS CLI) dapat lebih cepat dan lebih nyaman dibandingkan menggunakan konsol. Alat baris perintah juga berguna jika Anda ingin membangun skrip yang melakukan tugas AWS.

AWS menyediakan dua set alat baris perintah: AWS Command Line Interface dan AWS Tools for Windows PowerShell. Untuk informasi tentang menginstal dan menggunakan AWS CLI, lihat Panduan Pengguna AWS Command Line Interface. Untuk informasi tentang menginstal dan menggunakan Alat untuk WindowsPowerShell, lihatAWS Tools for Windows PowerShellPanduan Pengguna.

> **Note**
>
> Pada instans Windows Server Anda, Windows PowerShell 3.0 atau lebih baru diperlukan untuk menjalankan dokumen SSM tertentu (misalnya, dokumen `AWS-ApplyPatchBaseline` warisan). Verifikasi bahwa instans Windows Server Anda menjalankan Windows Management Framework 3.0 atau lebih baru. Kerangka kerja mencakup Windows PowerShell.

SDK AWS

AWS memberikan kit pengembangan perangkat lunak (SDK) yang terdiri dari perpustakaan dan kode sampel untuk berbagai bahasa dan platform pemrograman (misalnya, Java, Python, Ruby, .NET, iOS dan Android, dan lainnya). SDK menyediakan cara mudah untuk memberikan akses terprogram ke Manajer Sistem. Untuk informasi tentang AWS SDK, termasuk cara mengunduh dan menginstalnya, lihat Alat untuk Amazon Web Services.

Riwayat nama layanan Systems Manager

AWS Systems Manager(Manajer Sistem) sebelumnya dikenal sebagai”Amazon Simple Systems Manager (SSM)“dan”Amazon EC2 Systems Manager (SSM)“. Nama singkatan asli dari layanan ini, "SSM", masih tercermin dalam berbagai sumber daya AWS, termasuk beberapa konsol layanan lainnya. Beberapa contoh:

- Manajer SistemAgent:SSM Agent
- parameter Systems Manager: Parameter SSM
- Titik akhir layanan Systems Manager: `ssm.region.amazonaws.com`
- Jenis sumber daya AWS CloudFormation: `AWS::SSM::Document`
- Pengidentifikasi aturan AWS Config: `EC2_INSTANCE_MANAGED_BY_SSM`
- perintah AWS Command Line Interface (AWS CLI): `aws ssm describe-patch-baselines`
- Nama kebijakan yang dikelola (IAM) AWS Identity and Access Management: `AmazonSSMReadOnlyAccess`
- ARN sumber daya Systems Manager: `arn:aws:ssm:region:account-id:patchbaseline/pb-07d8884178EXAMPLE`

Didukung Wilayah AWS

Manajer Sistem tersedia diWilayah AWStercantum dalamTitik akhir layanan Manajer Sistemdi dalamReferensi Umum Amazon Web Services. Sebelum memulai proses konfigurasi Systems Manager, kami sarankan Anda memverifikasi bahwa layanan tersedia di masing-masing Wilayah AWS yang ingin Anda gunakan.

Untuk mesin non-EC2 dihibrida dan multicloudlingkungan, kami sarankan Anda memilih Wilayah yang paling dekat dengan pusat data atau lingkungan komputasi Anda.

Sistem operasi dan jenis mesin yang didukung

Sebelum bekerja dengan Systems Manager, verifikasi bahwa sistem operasi (OS), versi OS, dan jenis mesin Anda didukung sebagai node terkelola.

Topik

- Sistem operasi yang didukung untuk Systems Manager
- Jenis alat berat yang didukung di lingkungan hybrid dan multicloud

Sistem operasi yang didukung untuk Systems Manager

Bagian berikut mencantumkan versi OS dan OS yang didukung oleh Systems Manager.

> **Note**
>
> Jika Anda berencana untuk mengelola dan mengonfigurasi perangkat AWS IoT Greengrass inti dengan menggunakan Systems Manager, perangkat tersebut harus memenuhi persyaratan AWS IoT Greengrass. Untuk informasi selengkapnya, lihat Menyiapkan perangkat AWS IoT Greengrass inti di Panduan AWS IoT Greengrass Version 2 Pengembang.
> Jika Anda berencana untuk mengelola AWS IoT dan mengonfigurasi perangkat AWS non-edge, perangkat tersebut harus memenuhi persyaratan yang tercantum di sini dan dikonfigurasi sebagai node terkelola lokal untuk Systems Manager. Untuk informasi selengkapnya, lihat AWS Systems ManagerMenyiapkan perangkat edge.

> **Important**
>
> Patch Manager, kemampuan Systems Manager, mungkin tidak mendukung semua versi OS yang tercantum dalam topik ini. Untuk daftar versi OS yang didukung olehPatch Manager, lihatPrasyarat Patch Manager.

Tipe sistem operasi

- Linux
- macOS(Hanya instans Amazon EC2)
- Raspberry Pi OS(sebelumnya Raspbian)
- Windows Server

Linux

AlmaLinux

Versi	x86	x86_64	ARM64
8.3—8.7		✓	✓

Versi	x86	x86_64	ARM64
9.0—9.2		✓	✓

Amazon Linux 1

Versi	x86	x86_64	ARM64
2012.03—2018.03	✓	✓	

> **Note**
>
> Dimulai dengan versi 2015.03, Amazon Linux 1 dirilis dalam x86_64 versi. Amazon Linux 1 mencapai akhir dukungan standarnya pada 31 Desember 2020, dan mencapai akhir masa pakai pada 31 Desember 2023, seperti yang diumumkan dalam Pembaruan di Amazon Linux AMI end-of-life di Blog AWS Berita. AWS tidak lagi menyediakan Amazon Machine Images (AMIs) untuk sistem operasi ini. AWS Systems Manager Namun, terus memberikan dukungan untuk instans Amazon Linux 1 yang ada.

Amazon Linux 2

Versi	x86	x86_64	ARM64
Versi 2.0 dan semua yang lebih baru		✓	✓

Amazon Linux 2023

Versi	x86	x86_64	ARM64
2023.0.20230315.0 dan semua versi yang lebih baru		✓	✓

Bottlerocket

Versi	x86_64	ARM64
1.0.0 dan semua versi yang lebih baru	✓	✓

CentOS

Versi	x86	x86_64	ARM64
6.x[1]	✓	✓	
Versi 7.1 dan 7.x yang lebih baru		✓	✓
8.0—8.5		✓	✓

[1] Untuk menggunakan versi ini, Anda harus menggunakan versi 3.0.x dari versi. SSM Agent Kami merekomendasikan menggunakan versi 3.0.x terbaru yang tersedia dari file. SSM Agent SSM AgentVersi yang lebih baru (3.1 atau yang lebih baru) tidak didukung.

CentOS Aliran

Versi	x86	x86_64	ARM64
8		✓	✓

Debian Server

Versi	x86	x86_64	ARM64
Jessie (8)		✓	
Stretch (9)		✓	✓
Buster (10)		✓	✓
Bullseye(11)		✓	✓

Versi	x86	x86_64	ARM64
Bookworm(12)		✓	✓

Oracle Linux

Versi	x86	x86_64	ARM64
7.5—7.8		✓	
8.1—8.7		✓	
9.0—9.2		✓	

Red Hat Enterprise Linux (RHEL)

Versi	x86	x86_64	ARM64
6.x[1]	✓	✓	
7.0—7.5		✓	
7.6—8.8		✓	✓
9.0—9.2		✓	✓

[1] Untuk menggunakan versi ini, Anda harus menggunakan versi 3.0.x dari versi. SSM Agent Kami merekomendasikan menggunakan versi 3.0.x terbaru yang tersedia dari file. SSM Agent SSM AgentVersi yang lebih baru (3.1 atau yang lebih baru) tidak didukung.

Linux Rocky

Versi	x86	x86_64	ARM64
8.4—8.7		✓	✓
9.0—9.2		✓	✓

SUSE Linux Enterprise Server (SLES)

Versi	x86	x86_64	ARM64
Versi 12 dan 12.x yang lebih baru		✓	
Versi 15 dan 15.x yang lebih baru		✓	✓

Ubuntu Server

Versi	x86	x86_64	ARM64
12.04 LTS dan 14.04 LTS	✓	✓	
16.04 LTS dan 18.04 LTS		✓	✓
20.04 LTS dan 20.10 STR		✓	✓
22,04 LTS		✓	✓
23.04		✓	✓

macOS(Hanya instans Amazon EC2)

Versi	x86	x86_64	Mac with Apple silicon
10.14.x (Mojave)		✓	
10.15.x (Catalina)		✓	
11.x (Big Sur)		✓	✓
12.x (Monterey)		✓	✓

Versi	x86	x86_64	Mac with Apple silicon
13.x (Ventura)		✓	✓
14.x (Sonoma)		✓	✓

> **Note**
>
> macOStidak didukung sama sekali Wilayah AWS. Untuk informasi selengkapnya tentang dukungan Amazon EC2macOS, lihat instans Amazon EC2 Mac di Panduan Pengguna Amazon EC2 untuk Instans Linux.

Raspberry Pi OS(sebelumnya Raspbian)

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

Info lebih lanjut

- Kelola perangkat Raspberry Pi menggunakan AWS Systems Manager

Windows Server

SSM Agentmembutuhkan Windows PowerShell 3.0 atau nanti untuk menjalankan AWS Systems Manager dokumen tertentu (dokumen SSM) pada Windows Server instance (misalnya, dokumen lama`AWS-ApplyPatchBaseline`). Verifikasi bahwa instans Windows Server Anda menjalankan Windows Management Framework 3.0 atau lebih baru. Kerangka kerja ini mencakup Windows PowerShell. Untuk informasi selengkapnya, lihat Windows Management Framework 3.0.

Versi	x86	x86_64	ARM64
2008[1]	✓	✓	

Versi	x86	x86_64	ARM64
2008 R2[1]		✓	
2012 dan 2012 R2		✓	
2016		✓	
2019		✓	
2022		✓	

[1] Per 14 Januari 2020, Windows Server 2008 tidak lagi didukung untuk pembaruan fitur atau keamanan dari Microsoft. Legacy Amazon Machine Images (AMIs) untuk Windows Server 2008 dan 2008 R2 masih menyertakan versi 2 dari SSM Agent prainstal, tetapi Systems Manager tidak lagi secara resmi mendukung versi 2008 dan tidak lagi memperbarui agen untuk versi ini. Windows Server Selain itu, SSM Agent versi 3 mungkin tidak kompatibel dengan semua operasi pada Windows Server 2008 dan 2008 R2. Versi final yang didukung secara resmi SSM Agent untuk versi Windows Server 2008 adalah 2.3.1644.0.

Jenis alat berat yang didukung di lingkungan hybrid dan multicloud

Systems Manager mendukung sejumlah jenis mesin sebagai node terkelola. Node terkelola adalah mesin apa pun yang dikonfigurasi untuk bekerja dengan Systems Manager.

Panduan pengguna ini menggunakan istilah hybrid dan multicloud untuk merujuk ke lingkungan yang berisi kombinasi jenis mesin berikut:

- Instans Amazon Elastic Compute Cloud (Amazon EC2)
- Server di tempat Anda sendiri (server lokal)
- AWS IoT Greengrass perangkat inti
- AWS IoT dan perangkat AWS non-edge
- Mesin virtual (VM), termasuk VM di lingkungan cloud lainnya

Untuk informasi tentang AWS dukungan untuk lingkungan hybrid dan multicloud, lihat AWS Solusi untuk Hybrid dan Multicloud.

Menyiapkan AWS Systems Manager

Selesaikan tugas di bagian ini untuk menyiapkan dan mengonfigurasi peran, akun pengguna, izin, dan sumber daya awalAWS Systems Manager. Tugas yang dijelaskan di bagian ini biasanya dilakukan olehAkun AWS dan administrator sistem. Setelah langkah-langkah ini selesai, pengguna di organisasi Anda dapat menggunakan Systems Manager untuk mengonfigurasi, mengelola, dan mengakses node terkelola. Node terkelola adalah mesin apa pun yang dikonfigurasi untuk digunakan dengan Systems Manager di lingkungan hibrida dan multicloud.

> **Note**
>
> Jika Anda berencana untuk menggunakan instans Amazon EC2 dan sumber daya komputasi Anda sendiri di lingkungan hibrida dan multicloud, ikuti langkah-langkah iniMenyiapkan Systems Manager untuk instans EC2. Topik itu menyajikan langkah-langkah dalam urutan terbaik untuk menyelesaikan pengaturan Systems Manager untuk instans EC2 dan mesin non-EC2.

Jika Anda sudah menggunakan lainnyaLayanan AWS, Anda telah menyelesaikan beberapa langkah berikut. Meskipun, langkah-langkah lain khusus untuk Systems Manager. Oleh karena itu, kami sarankan meninjau seluruh bagian ini untuk memastikan bahwa Anda siap untuk menggunakan semua kemampuan Systems Manager.

Topik

- Pengaturan umum untuk AWS Systems Manager
- Menyiapkan Systems Manager untuk instans EC2
- Menyiapkan Manajer Sistem untuk lingkungan hybrid dan multicloud
- AWS Systems ManagerMenyiapkan perangkat edge
- Menyiapkan administrator yang didelegasikan untuk Manajer Sistem

Pengaturan umum untuk AWS Systems Manager

Jika Anda belum melakukannya, daftar Akun AWS dan buat pengguna administratif.

Mendaftar Akun AWS

Jika Anda tidak memiliki Akun AWS, selesaikan langkah-langkah berikut untuk membuatnya.

Untuk mendaftar Akun AWS

1. Buka https://portal.aws.amazon.com/billing/signup.
2. Ikuti petunjuk secara online.

   Anda akan diminta untuk menerima panggilan telepon dan memasukkan kode verifikasi pada keypad telepon sebagai bagian dari prosedur pendaftaran.

   Saat Anda mendaftar Akun AWS, Pengguna root akun AWS akan dibuat. Pengguna root memiliki akses ke semua Layanan AWS dan sumber daya dalam akun. Sebagai praktik terbaik keamanan, tetapkan akses administratif ke pengguna administratif, dan hanya gunakan pengguna root untuk melakukan tugas yang memerlukan akses pengguna root.

AWS akan mengirimkan email konfirmasi kepada Anda setelah proses pendaftaran selesai. Anda dapat melihat aktivitas akun saat ini dan mengelola akun dengan mengunjungi https://aws.amazon.com/ dan memilih Akun Saya.

Membuat pengguna administratif

Setelah mendaftar Akun AWS, amankan Pengguna root akun AWS, aktifkan AWS IAM Identity Center, dan buat sebuah pengguna administratif sehingga Anda tidak menggunakan pengguna root untuk tugas sehari-hari.

Mengamankan Pengguna root akun AWS Anda

1. Masuk ke AWS Management Console sebagai pemilik akun dengan memilih Pengguna root dan memasukkan alamat email Akun AWS Anda. Di halaman berikutnya, masukkan kata sandi Anda.

   Untuk bantuan masuk menggunakan pengguna root, lihat Masuk sebagai pengguna root dalam Panduan Pengguna AWS Sign-In.
2. Aktifkan autentikasi multi-faktor (MFA) untuk pengguna root Anda.

   Untuk petunjuknya, silakan lihat Mengaktifkan perangkat MFA virtual untuk pengguna root Akun AWS Anda (konsol) dalam Panduan Pengguna IAM.

Membuat pengguna administratif

1. Aktifkan Pusat Identitas IAM.

   Untuk mendapatkan petunjuk, silakan lihat Mengaktifkan AWS IAM Identity Center di Panduan Pengguna AWS IAM Identity Center.

2. Di Pusat Identitas IAM, berikan akses administratif ke sebuah pengguna administratif.

   Untuk mendapatkan tutorial tentang menggunakan Direktori Pusat Identitas IAM sebagai sumber identitas Anda, silakan lihat Mengonfigurasi akses pengguna dengan Direktori Pusat Identitas IAM default di Panduan Pengguna AWS IAM Identity Center.

Masuk sebagai pengguna administratif

- Untuk masuk dengan pengguna Pusat Identitas IAM, gunakan URL masuk yang dikirim ke alamat email Anda saat Anda membuat pengguna Pusat Identitas IAM.

  Untuk bantuan masuk menggunakan pengguna Pusat Identitas IAM, lihat Masuk ke portal akses AWS dalam Panduan Pengguna AWS Sign-In.

Menyiapkan Systems Manager untuk instans EC2

Selesaikan tugas di bagian ini untuk menyiapkan dan sumber daya awal untukAWS Systems Manager. Tugas yang dijelaskan di bagian ini biasanya dilakukan olehAkun AWS dan administrator sistem. Setelah langkah-langkah ini selesai, pengguna di organisasi Anda dapat menggunakan Systems Manager untuk mengonfigurasi,, mengakses instans Amazon Elastic Compute Cloud (Amazon EC2).

> **Note**
>
> Jika Anda berencana menggunakan Systems Manager untuk mengelola dan mengonfigurasi perangkat lokal, ikuti langkah-langkah pengaturan diMenyiapkan Manajer Sistem untuk lingkungan hybrid dan multicloud. Jika Anda berencana untuk menggunakan kedua instans Amazon EC2 dan non-EC2 di lingkungan hibrida dan multicloud, ikuti langkah-langkah ini terlebih dahulu. Bagian ini menyajikan langkah-langkah sesuai urutan yang disarankan untuk mengonfigurasi peran, izin, sumber daya awal untuk digunakan dalam operasi Systems Manager Anda.

Jika Anda sudah menggunakan lainnyaLayanan AWS, Anda telah menyelesaikan beberapa langkah berikut. Meskipun, langkah-langkah lain khusus untuk Systems Manager. Oleh karena itu, kami sarankan meninjau seluruh bagian ini untuk memastikan bahwa Anda siap untuk menggunakan semua kemampuan Systems Manager.

Konten

Langkah 1: Konfigurasikan izin instance untuk Systems Manager

Secara default, AWS Systems Manager tidak memiliki izin untuk melakukan tindakan pada instance Anda. Anda dapat memberikan izin instans di tingkat akun menggunakan peran AWS Identity and Access Management (IAM), atau di tingkat instans menggunakan profil instance. Jika kasus penggunaan Anda memungkinkan, kami sarankan untuk memberikan akses di tingkat akun menggunakan Konfigurasi Manajemen Host Default.

Konfigurasi yang disarankan

Konfigurasi Manajemen Host default memungkinkan Systems Manager mengelola instans Amazon EC2 Anda secara otomatis. Setelah Anda mengaktifkan setelan ini, semua instance yang menggunakan Layanan Metadata Instans Versi 2 (IMDSv2) di Wilayah AWS dan Akun AWS dengan SSM Agent versi 3.2.582.0 atau yang lebih baru diinstal secara otomatis menjadi instance terkelola. Konfigurasi Manajemen Host Default tidak mendukung Layanan Metadata Instans Versi 1. Untuk informasi tentang transisi ke IMDSv2, lihat Transisi menggunakan Layanan Metadata Instans Versi 2 di Panduan Pengguna Amazon EC2 untuk Instans Linux. Untuk informasi tentang memeriksa versi yang SSM Agent diinstal pada instans Anda, lihatMemeriksa nomor SSM Agent versi. Untuk informasi tentang memperbaruiSSM Agent, lihatMemperbarui secara otomatis SSM Agent. Manfaat instans terkelola meliputi:

- Connect ke instans Anda dengan aman menggunakan. Session Manager
- Lakukan pemindaian patch otomatis menggunakanPatch Manager.
- Lihat informasi terperinci tentang instans Anda menggunakan Systems Manager Inventory.
- Lacak dan kelola instance menggunakanFleet Manager.
- Tetap SSM Agent up to date secara otomatis.

Fleet Manager, InventarisPatch Manager,, dan Session Manager merupakan kemampuan AWS Systems Manager.

Konfigurasi Manajemen Host Default memungkinkan pengelolaan instans tanpa menggunakan profil instans dan memastikan bahwa Systems Manager memiliki izin untuk mengelola semua instance di Wilayah dan akun. Jika izin yang diberikan tidak cukup untuk kasus penggunaan, Anda juga dapat menambahkan kebijakan ke peran IAM default yang dibuat oleh Konfigurasi Manajemen Host Default. Atau, jika Anda tidak memerlukan izin untuk semua kemampuan yang disediakan oleh peran IAM default, Anda dapat membuat peran dan kebijakan kustom Anda sendiri. Setiap perubahan yang dibuat pada peran IAM yang Anda pilih untuk Konfigurasi Manajemen Host Default berlaku untuk semua instans Amazon EC2 yang dikelola di Wilayah dan akun. Untuk informasi selengkapnya tentang kebijakan yang digunakan oleh Konfigurasi Manajemen Host Default, lihat. AWS kebijakan terkelola: AmazonsSMManageDEC2 InstanceDefaultPolicy Untuk informasi selengkapnya tentang Konfigurasi Manajemen Host Default, lihatMenggunakan pengaturan Konfigurasi Manajemen Host Default.

Important

Instans yang terdaftar menggunakan Konfigurasi Manajemen Host Default menyimpan informasi pendaftaran secara lokal di direktori `/lib/amazon/ssm` atau`C:\ProgramData\Amazon`. Menghapus direktori ini atau file-file mereka akan mencegah instance memperoleh kredensyal yang diperlukan untuk terhubung ke Systems Manager menggunakan Default Host Management Configuration. Dalam kasus ini, Anda harus menggunakan profil instance untuk memberikan izin yang diperlukan untuk instans Anda, atau membuat ulang instance.

Note

Prosedur ini dimaksudkan untuk dilakukan hanya oleh administrator. Terapkan akses hak istimewa terkecil saat mengizinkan individu untuk mengonfigurasi atau memodifikasi Konfigurasi Manajemen Host Default. Anda harus mengaktifkan Konfigurasi Manajemen Host Default di setiap yang Wilayah AWS Anda inginkan untuk mengelola instans Amazon EC2 Anda secara otomatis.

Untuk mengaktifkan pengaturan Konfigurasi Manajemen Host Default

Anda dapat mengaktifkan Konfigurasi Manajemen Host Default dari Fleet Manager konsol. Agar berhasil menyelesaikan prosedur ini menggunakan alat baris perintah AWS Management Console atau pilihan Anda, Anda harus memiliki izin untuk operasi GetServiceSetting, ResetServiceSetting, dan UpdateServiceSettingAPI. Selain itu, Anda harus memiliki izin untuk `iam:PassRole` izin untuk peran `AWSSystemsManagerDefaultEC2InstanceManagementRole` IAM. Berikut ini adalah contoh kebijakan . Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri.

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

Sebelum memulai, jika Anda memiliki profil instans yang dilampirkan ke instans Amazon EC2 Anda, hapus izin apa pun yang memungkinkan pengoperasian. `ssm:UpdateInstanceInformation` SSM AgentUpaya untuk menggunakan izin profil instance sebelum menggunakan izin Konfigurasi

Manajemen Host Default. Jika Anda mengizinkan `ssm:UpdateInstanceInformation` operasi di profil instans Anda, instans tidak akan menggunakan izin Konfigurasi Manajemen Host Default.

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih Konfigurasi Konfigurasi Manajemen Host Default di bawah tarik-turun Manajemen akun.
4. Aktifkan Aktifkan Konfigurasi Manajemen Host Default.
5. Pilih peran IAM yang digunakan untuk mengaktifkan kemampuan Systems Manager untuk instans Anda. Sebaiknya gunakan peran default yang disediakan oleh Konfigurasi Manajemen Host Default. Ini berisi set izin minimum yang diperlukan untuk mengelola instans Amazon EC2 Anda menggunakan Systems Manager. Jika Anda lebih suka menggunakan peran khusus, kebijakan kepercayaan peran tersebut harus mengizinkan Systems Manager sebagai entitas tepercaya.
6. Pilih Konfigurasi untuk menyelesaikan penyiapan.

Setelah mengaktifkan Konfigurasi Manajemen Host Default, mungkin diperlukan waktu 30 menit agar instans Anda menggunakan kredensyal peran yang Anda pilih. Anda harus mengaktifkan Konfigurasi Manajemen Host Default di setiap Wilayah yang ingin Anda kelola instans Amazon EC2 secara otomatis.

Konfigurasi alternatif

Anda dapat memberikan akses pada tingkat instans individual dengan menggunakan profil instans AWS Identity and Access Management (IAM). Profil instans adalah kontainer yang menyampaikan informasi IAM role ke instans Amazon Elastic Compute Cloud (Amazon EC2) saat peluncuran. Anda dapat membuat profil instans untuk Systems Manager dengan melampirkan satu atau lebih kebijakan IAM yang menentukan izin yang diperlukan untuk peran baru atau peran yang telah Anda buat.

> **Note**
>
> Anda dapat menggunakanQuick Setup, kemampuan AWS Systems Manager, untuk dengan cepat mengkonfigurasi profil instans pada semua instance di Anda Akun AWS. Quick Setupjuga membuat peran layanan IAM (atau mengambil peran), yang memungkinkan Systems Manager menjalankan perintah dengan aman pada instance Anda atas nama Anda. Dengan menggunakanQuick Setup, Anda dapat melewati langkah ini (Langkah 3) dan Langkah 4. Untuk informasi selengkapnya, lihat AWS Systems Manager Quick Setup.

Perhatikan rincian berikut tentang membuat profil instans IAM:

- Jika Anda mengonfigurasi mesin non-EC2 di lingkungan hybrid dan multicloud untuk Systems Manager, Anda tidak perlu membuat profil instans untuk mereka. Sebaliknya, konfigurasi server dan VM Anda untuk menggunakan peran layanan IAM. Untuk informasi lebih lanjut, lihat Membuat peran layanan IAM untuk lingkungan hibrid.
- Jika Anda mengubah profil instans IAM, kredensi instans mungkin perlu beberapa waktu untuk menyegarkan. SSM Agenttidak akan memproses permintaan sampai ini terjadi. Untuk mempercepat proses penyegaran, Anda dapat memulai ulang SSM Agent atau memulai ulang instance.

Tergantung pada apakah Anda membuat peran baru untuk profil instans atau menambahkan izin yang diperlukan ke peran yang ada, gunakan salah satu prosedur berikut.

Untuk membuat profil instans untuk instans terkelola Systems Manager (konsol)

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran, lalu pilih Buat peran.
3. Untuk jenis entitas Tepercaya, pilih Layanan AWS.
4. Segera di bawah Kasus penggunaan, pilih EC2, lalu pilih Berikutnya.
5. Pada halaman Tambahkan izin, lakukan hal berikut:
   - Gunakan kolom Pencarian untuk menemukan kebijakan ManagedInstanceCoreAmazonSSM. Pilih kotak centang di sebelah namanya.

Konsol mempertahankan pilihan Anda bahkan jika Anda mencari kebijakan lain.

- Jika Anda membuat kebijakan bucket S3 kustom di prosedur sebelumnya(Opsional) Buat kebijakan khusus untuk akses bucket S3, cari kebijakan tersebut dan pilih kotak centang di samping namanya.
- Jika Anda berencana untuk menggabungkan instance ke Active Directory yang dikelola oleh AWS Directory Service, cari AmazonSSM DirectoryServiceAccess dan pilih kotak centang di samping namanya.
- Jika Anda berencana untuk menggunakan EventBridge atau CloudWatch Log untuk mengelola atau memantau instans Anda, cari CloudWatchAgentServerPolicydan pilih kotak centang di samping namanya.

6. Pilih Berikutnya.
7. Untuk nama Peran, masukkan nama untuk profil instans baru Anda, seperti**`SSMInstanceProfile`**.

> **Note**
>
> Buat catatan tentang nama peran. Anda akan memilih peran ini ketika Anda membuat instans baru yang ingin Anda kelola dengan menggunakan Systems Manager.

8. (Opsional) Untuk Deskripsi, perbarui deskripsi untuk profil contoh ini.
9. (Opsional) Untuk Tag, tambahkan satu atau beberapa pasangan nilai kunci tag untuk mengatur, melacak, atau mengontrol akses untuk peran ini, lalu pilih Buat peran. Sistem mengembalikan Anda ke halaman Peran.

Untuk menambahkan izin profil instans pada Systems Manager untuk peran yang ada (konsol)

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran, lalu pilih peran yang ada yang ingin Anda kaitkan dengan profil instans untuk operasi Systems Manager.

3. Pada tab Izin, pilih Tambahkan izin, Lampirkan kebijakan.
4. Pada halaman Lampirkan kebijakan, lakukan hal berikut:
   - Gunakan kolom Pencarian untuk menemukan kebijakan ManagedInstanceCoreAmazonSSM. Pilih kotak centang di sebelah namanya.
   - Jika Anda telah membuat kebijakan bucket S3 kustom, cari kebijakan tersebut dan pilih kotak centang di samping namanya. Untuk informasi tentang kebijakan bucket S3 kustom untuk profil instans, lihat (Opsional) Buat kebijakan khusus untuk akses bucket S3.
   - Jika Anda berencana untuk menggabungkan instance ke Active Directory yang dikelola oleh AWS Directory Service, cari AmazonSSM DirectoryServiceAccess dan pilih kotak centang di samping namanya.
   - Jika Anda berencana untuk menggunakan EventBridge atau CloudWatch Log untuk mengelola atau memantau instans Anda, cari CloudWatchAgentServerPolicydan pilih kotak centang di samping namanya.
5. Pilih Lampirkan kebijakan.

Untuk informasi tentang cara memperbarui peran untuk menyertakan entitas tepercaya atau membatasi akses lebih lanjut, lihat Memodifikasi peran dalam Panduan Pengguna IAM.

(Opsional) Buat kebijakan khusus untuk akses bucket S3

Membuat kebijakan kustom untuk akses Amazon S3 hanya jika Anda memerlukan untuk menggunakan VPC endpoint atau menggunakan bucket S3 Anda sendiri dalam operasi Systems Manager Anda. Anda dapat melampirkan kebijakan ini ke peran IAM default yang dibuat oleh Konfigurasi Manajemen Host Default, atau profil instans yang Anda buat di prosedur sebelumnya.

Untuk informasi tentang bucket S3 AWS terkelola yang dapat Anda akses dalam kebijakan berikut, lihat. SSM Agentkomunikasi dengan bucket S3 AWS terkelola

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Kebijakan dan kemudian pilih Buat kebijakan.
3. Pilih tab JSON, dan ganti teks default dengan yang berikut ini.

```
{
    "Version": "2012-10-17",
    "Statement": [
```

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

 (2)                                                                    {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:PutObject",

 "s3:PutObjectAcl", (3)

 "s3:GetEncryptionConfiguration" (4)
            ],
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*",
                "arn:aws:s3:::DOC-EXAMPLE-
BUCKET" (5)
            ]
        }
    ]
}
```

[1] Yang pertama elemen `Statement` hanya diperlukan jika Anda menggunakan VPC endpoint.

[2] Yang kedua elemen `Statement` hanya diperlukan jika Anda menggunakan bucket S3 yang Anda buat untuk digunakan dalam operasi Systems Manager Anda.

[3] Izin daftar kontrol akses `PutObjectAcl` hanya diperlukan jika Anda berencana untuk mendukung akses lintas akun ke bucket S3 di akun lain.

[4] Elemen `GetEncryptionConfiguration` diperlukan jika bucket S3 Anda dikonfigurasi untuk menggunakan enkripsi.

[5] Jika bucket S3 Anda dikonfigurasi untuk menggunakan enkripsi, maka bucket root S3 (misalnya, `arn:aws:s3:::DOC-EXAMPLE-BUCKET`) harus terdaftar dalam bagian Sumber Daya. Pengguna, grup, atau peran Anda harus dikonfigurasi dengan akses ke root bucket.

4. Jika Anda menggunakan VPC endpoint dalam operasi Anda, lakukan hal berikut:

   Dalam elemen `Statement` pertama, ganti masing-masing *`wilayah`* Placeholder dengan pengidentifikasi kebijakan dasar Wilayah AWS yang akan digunakan. Misalnya, gunakan `us-east-2` untuk Wilayah US East (Ohio). Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

   > **Important**
   >
   > Kami menyarankan Anda untuk menghindari menggunakan karakter wildcard (*) di tempat Wilayah tertentu dalam kebijakan ini. Misalnya, gunakan `arn:aws:s3:::aws-ssm-us-east-2/*` dan jangan gunakan `arn:aws:s3:::aws-ssm-*/*`. Menggunakan wildcard dapat menyediakan akses ke bucket S3 yang tidak ingin Anda berikan akses. Jika Anda ingin menggunakan profil instans untuk lebih dari satu Wilayah, kami sarankan Anda mengulangi elemen `Statement` pertama untuk setiap Wilayah.

   -atau-

   Jika Anda tidak menggunakan VPC endpoint dalam operasi Anda, Anda dapat menghapus elemen `Statement`.

5. Jika Anda menggunakan bucket S3 milik Anda sendiri dalam operasi Systems Manager, lakukan hal berikut:

   Di elemen `Statement` kedua, ganti *`DOC-CONTOH-BUCKET`* dengan nama bucket S3 di akun Anda. Anda akan menggunakan bucket ini untuk operasi Systems Manager Anda. Ini memberikan izin objek dalam bucket, gunakan `"arn:aws:s3:::my-bucket-name/*"` sebagai sumber daya. Untuk informasi selengkapnya tentang memberikan izin untuk bucket

atau objek dalam bucket, lihat topik tindakan Amazon S3 di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon dan posting AWS blog Kebijakan IAM dan Kebijakan Bucket dan ACL! Astaga! (Mengontrol Akses ke Sumber Daya S3).

> **Note**
>
> Jika Anda menggunakan lebih dari satu bucket, memberikan untuk masing-masing ARN. Lihat contoh berikut untuk izin pada bucket.
>
> ```
> "Resource": [
> "arn:aws:s3:::DOC-EXAMPLE-BUCKET1/*",
> "arn:aws:s3:::DOC-EXAMPLE-BUCKET2/*"
>              ]
> ```

-atau-

Jika Anda tidak menggunakan bucket S3 milik Anda sendiri dalam operasi Systems Manager, Anda dapat menghapus elemen `Statement` kedua.

6. Pilih Berikutnya: Tanda.
7. (Opsional) Tambahkan tag dengan memilih Tambah tag, dan masukkan tag pilihan untuk kebijakan tersebut.
8. Pilih Berikutnya: Peninjauan.
9. Untuk Nama, masukkan nama untuk mengidentifikasi kebijakan ini, seperti**`SSMInstanceProfileS3Policy`**.
10. Pilih Buat kebijakan.

Pertimbangan kebijakan tambahan untuk instans terkelola

Bagian ini menjelaskan beberapa kebijakan yang dapat Anda tambahkan ke peran IAM default yang dibuat oleh Konfigurasi Manajemen Host Default, atau profil instans Anda. AWS Systems Manager Untuk memberikan izin komunikasi antara instans dan Systems Manager API, sebaiknya buat kebijakan khusus yang mencerminkan kebutuhan sistem dan persyaratan keamanan Anda. Bergantung pada rencana operasi, Anda mungkin memerlukan izin yang direpresentasikan dalam satu atau beberapa kebijakan lainnya.

Kebijakan: **`AmazonSSMDirectoryServiceAccess`**

Hanya diperlukan jika Anda berencana untuk bergabung dengan instans Amazon EC2 untuk Windows Server ke direktori Microsoft AD.

Kebijakan AWS terkelola ini memungkinkan SSM Agent untuk mengakses AWS Directory Service atas nama Anda untuk permintaan bergabung dengan domain oleh instans terkelola. Untuk informasi selengkapnya, lihat Bergabung dengan Instans Windows EC2 dengan mulus di Panduan AWS Directory Service Administrasi.

Kebijakan: **`CloudWatchAgentServerPolicy`**

Diperlukan hanya jika Anda berencana untuk menginstal dan menjalankan CloudWatch agen pada instans Anda untuk membaca metrik dan data log pada sebuah instance dan menuliskannya ke Amazon CloudWatch. Ini membantu Anda memantau, menganalisis, dan dengan cepat menanggapi masalah atau perubahan AWS sumber daya Anda.

Peran IAM default Anda yang dibuat oleh Konfigurasi Manajemen Host Default atau profil instans memerlukan kebijakan ini hanya jika Anda akan menggunakan fitur seperti Amazon EventBridge atau Amazon CloudWatch Logs. (Anda juga dapat membuat kebijakan yang lebih ketat yang, misalnya, membatasi akses penulisan ke aliran CloudWatch log Log tertentu.)

> **Note**
>
> Menggunakan EventBridge dan CloudWatch Log fitur adalah opsional. Namun, kami sarankan untuk mengaturnya di awal proses konfigurasi Systems Manager jika Anda memutuskan untuk menggunakannya. Untuk informasi selengkapnya, lihat Panduan EventBridge Pengguna Amazon dan Panduan Pengguna CloudWatch Log Amazon.

Untuk membuat kebijakan IAM dengan izin untuk kemampuan Systems Manager tambahan, lihat sumber daya berikut:

- Membatasi akses ke parameter Systems Manager menggunakan kebijakan IAM
- Menyiapkan Otomatisasi
- Langkah 2: Verifikasi atau tambahkan izin instance untuk Session Manager

Lampirkan profil instance Systems Manager ke instance (konsol)

1. Masuk ke AWS Management Console dan buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, di bawah Instans, pilih Instans.
3. Arahkan ke dan pilih instans EC2 Anda dari daftar.
4. Pilih menu Actions, pilih Security, Modify IAM role.
5. Untuk IAM role, pilih profil instans yang Anda buat menggunakan prosedur ini Konfigurasi alternatif.
6. Pilih Perbarui peran IAM.

Untuk informasi lebih lanjut tentang melampirkan IAM role ke instans, pilih salah satu dari berikut ini, tergantung pada jenis sistem operasi yang Anda pilih:

- Melampirkan peran IAM ke instans di Panduan Pengguna Amazon EC2 untuk Instans Linux
- Lampirkan peran IAM ke instans di Panduan Pengguna Amazon EC2 untuk Instans Windows

Lanjutkan ke Langkah 2: Buat titik akhir VPC.

Langkah 2: Buat titik akhir VPC

Anda dapat meningkatkan postur keamanan node terkelola Anda (termasuk mesin non-EC2 di lingkungan hybrid dan multicloud) dengan mengonfigurasi untuk AWS Systems Manager menggunakan titik akhir VPC antarmuka di Amazon Virtual Private Cloud (Amazon VPC). Dengan menggunakan antarmuka VPC endpoint (interface endpoint), Anda dapat terhubung ke layanan yang didukung oleh. AWS PrivateLink AWS PrivateLink adalah teknologi yang memungkinkan Anda mengakses Amazon Elastic Compute Cloud (Amazon EC2) dan API Systems Manager secara pribadi dengan menggunakan alamat IP pribadi.

AWS PrivateLink membatasi semua lalu lintas jaringan antara instans terkelola, Systems Manager, dan Amazon EC2 ke jaringan Amazon. Ini berarti instans terkelola Anda tidak memiliki akses ke Internet. Jika Anda menggunakannya AWS PrivateLink, Anda tidak memerlukan gateway internet, perangkat NAT, atau gateway pribadi virtual.

Anda tidak diharuskan untuk mengkonfigurasi AWS PrivateLink, tetapi disarankan. Untuk informasi selengkapnya tentang AWS PrivateLink dan titik akhir VPC, lihat dan titik akhir AWS PrivateLink VPC.

> **Note**
>
> Alternatif untuk menggunakan VPC endpoint adalah untuk memungkinkan akses internet luar pada instans terkelola Anda. Dalam kasus ini, instans terkelola juga harus mengizinkan lalu lintas keluar HTTPS (port 443) ke titik akhir berikut:
>
> - `ssm.region.amazonaws.com`
> - `ssmmessages.region.amazonaws.com`
> - `ec2messages.region.amazonaws.com`
>
> SSM Agentmemulai semua koneksi ke layanan Systems Manager di cloud. Untuk alasan ini, Anda tidak perlu mengonfigurasi firewall Anda untuk mengizinkan lalu lintas masuk ke instans Anda untuk Systems Manager.
> Untuk informasi lebih lanjut tentang panggilan ke titik akhir ini, lihat Referensi: ec2messages, ssmmessages, dan operasi API lainnya.

Tentang Amazon VPC

Anda dapat menggunakan Amazon Virtual Private Cloud (Amazon VPC) untuk menentukan jaringan virtual di area Anda sendiri yang terisolasi secara logis di dalam AWS Cloud, yang dikenal sebagai virtual private cloud (VPC). Anda dapat meluncurkan sumber daya AWS , seperti instans, ke dalam VPC Anda. VPC Anda sangat menyerupai jaringan tradisional yang mungkin Anda operasikan di pusat data Anda sendiri, dengan memanfaatkan infrastruktur terukur dari AWS. Anda dapat mengonfigurasi VPC Anda; Anda dapat memilih baris alamat IP, membuat subnet, dan mengonfigurasi tabel rute, gateway jaringan, dan pengaturan keamanan. Anda dapat menghubungkan instans dalam VPC Anda ke internet. Anda dapat menghubungkan VPC Anda ke pusat data perusahaan Anda sendiri, membuat AWS Cloud perpanjangan pusat data Anda. Untuk melindungi sumber daya di setiap subnet, Anda dapat menggunakan beberapa lapisan keamanan, termasuk grup keamanan dan daftar kontrol akses jaringan. Untuk informasi selengkapnya, silakan lihat ACL Jaringan di Panduan Pengguna Amazon VPC.

Topik

- Pembatasan dan batasan VPC endpoint
- Membuat VPC endpoint untuk Systems Manager
- Penciptaan sebuah kebijakan VPC endpoint antarmuka

Pembatasan dan batasan VPC endpoint

Sebelum Anda mengkonfigurasi VPC endpoints untuk Systems Manager, perhatikan pembatasan dan batasan berikut.

Permintaan Lintas Wilayah

Titik akhir VPC tidak mendukung permintaan lintas wilayah—pastikan Anda membuat titik akhir sama dengan bucket Anda. Wilayah AWS Anda dapat menemukan lokasi bucket Anda dengan menggunakan konsol Amazon S3, atau dengan menggunakan perintah. get-bucket-location Gunakan titik akhir Amazon S3 khusus wilayah untuk mengakses bucket Anda; misalnya, *DOC-EXAMPLE-BUCKET*`.s3-us-west-2.amazonaws.com`. Untuk informasi selengkapnya tentang titik akhir khusus Wilayah untuk Amazon S3, lihat titik akhir Amazon S3 di. Referensi Umum Amazon Web Services Jika Anda menggunakan AWS CLI untuk membuat permintaan ke Amazon S3, setel wilayah default Anda ke wilayah yang sama dengan bucket, atau gunakan `--region` parameter dalam permintaan Anda.

Koneksi peering VPC

Titik akhir antarmuka VPC dapat diakses melalui kedua antar-wilayah dan antar-wilayah Koneksi peering VPC. Untuk informasi selengkapnya tentang permintaan koneksi peering VPC untuk titik akhir antarmuka VPC, lihat Koneksi peering VPC (Kuota) di Panduan Pengguna Amazon Virtual Private Cloud.

Koneksi titik akhir gateway VPC tidak dapat diperpanjang dari VPC. Sumber daya di sisi lain dari Koneksi peering VPC di VPC Anda tidak dapat menggunakan gateway endpoint untuk berkomunikasi dengan sumber daya dalam layanan gateway endpoint. Untuk informasi selengkapnya tentang permintaan koneksi peering VPC untuk titik akhir gateway VPC, lihat Titik akhir VPC (Kuota) di Panduan Pengguna Amazon Virtual Private Cloud

Koneksi masuk

Grup keamanan yang terkait dengan VPC endpoint harus mengizinkan koneksi masuk pada port 443 dari subnet privat pada instans terkelola. Jika koneksi masuk tidak diizinkan, maka instans terkelola tidak dapat terhubung ke SSM dan EC2 endpoint.

Resolusi DNS

Jika Anda menggunakan server DNS khusus, Anda harus menambahkan forwarder bersyarat untuk kueri apa pun ke domain ke server DNS Amazon `amazonaws.com` untuk VPC Anda.

Bucket S3

Kebijakan titik akhir VPC Anda harus mengizinkan akses ke setidaknya bucket Amazon S3 berikut:

- Bucket S3 yang tercantum dalam SSM Agentkomunikasi dengan bucket S3 AWS terkelola.
- Bucket S3 yang digunakan oleh Patch Manager untuk operasi baseline patch di Anda. Wilayah AWS Bucket ini berisi kode yang diambil dan berjalan pada instans oleh layanan dasar patch. Masing-masing Wilayah AWS memiliki bucket operasi baseline patch sendiri dari mana kode diambil ketika dokumen baseline patch dijalankan. Jika kode tidak dapat diunduh, perintah dasar patch akan gagal.

> **Note**
>
> Jika Anda menggunakan firewall lokal dan berencana untuk menggunakannyaPatch Manager, firewall tersebut juga harus mengizinkan akses ke titik akhir dasar patch yang sesuai.

Untuk menyediakan akses ke bucket di dalam Anda Wilayah AWS, sertakan izin berikut dalam kebijakan endpoint Anda.

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

wilayah mewakili pengenal untuk Wilayah AWS didukung oleh AWS Systems Manager, seperti `us-east-2` untuk Wilayah Timur AS (Ohio). Untuk daftar nilai *wilayah* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

Lihat contoh berikut ini.

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

> **Note**
>
> Di Wilayah Timur Tengah (Bahrain) (saya-selatan-1) saja, ember ini menggunakan konvensi penamaan yang berbeda. Untuk ini Wilayah AWS saja, gunakan dua ember berikut sebagai gantinya:

- `patch-baseline-snapshot-me-south-1-uduvl7q8`
- `aws-patch-manager-me-south-1-a53fc9dce`

CloudWatch Log Amazon

Jika Anda tidak mengizinkan instans Anda mengakses internet, buat titik akhir VPC CloudWatch untuk Log untuk menggunakan fitur yang mengirim log ke Log. CloudWatch Untuk informasi selengkapnya tentang membuat titik akhir untuk CloudWatch Log, lihat Membuat titik akhir VPC CloudWatch untuk Log di Panduan Pengguna Log CloudWatch Amazon.

DNS di lingkungan hybrid dan multicloud

Untuk informasi tentang mengonfigurasi DNS agar berfungsi dengan AWS PrivateLink titik akhir di lingkungan hybrid dan multicloud, lihat DNS pribadi untuk titik akhir antarmuka di Panduan Pengguna Amazon VPC. Jika Anda ingin menggunakan DNS Anda sendiri, Anda dapat menggunakan Route 53 Resolver. Untuk informasi selengkapnya, lihat Menyelesaikan kueri DNS antara VPC dan jaringan Anda di Panduan Pengembang Amazon Route 53.

Membuat VPC endpoint untuk Systems Manager

Gunakan informasi berikut untuk membuat antarmuka VPC dan titik akhir gateway untuk AWS Systems Manager. Topik link ini untuk prosedur di Panduan Pengguna Amazon VPC.

Untuk membuat VPC endpoint untuk Systems Manager

Pada langkah pertama dari prosedur ini, buat tiga Titik akhir dan satu opsional yang diperlukan Antarmuka untuk Systems Manager. Tiga titik akhir pertama diperlukan untuk Systems Manager yang bekerja di VPC. Yang keempat`com.amazonaws.`*`region`*`.ssmmessages`,, diperlukan hanya jika Anda menggunakan Session Manager kemampuan.

Pada langkah kedua, Anda membuat gateway titik akhir yang dibutuhkan untuk Systems Manager untuk mengakses Amazon S3.

> **Note**
>
> *`wilayah`* mewakili pengenal untuk Wilayah AWS didukung oleh AWS Systems Manager, seperti `us-east-2` untuk Wilayah Timur AS (Ohio). Untuk daftar nilai *`wilayah`* yang

didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

1. Ikuti langkah-langkah di Buat titik akhir antarmuka untuk membuat titik akhir antarmuka berikut:

   - **com.amazonaws.*region*.ssm**— Titik akhir untuk layanan Systems Manager.
   - **com.amazonaws.*region*.ec2messages**— Systems Manager menggunakan endpoint ini untuk melakukan panggilan dari SSM Agent ke layanan Systems Manager.
   - **com.amazonaws.*region*.ec2**— Jika Anda menggunakan Systems Manager untuk membuat snapshot berkemampuan VSS, Anda perlu memastikan bahwa Anda memiliki titik akhir ke layanan EC2. Tanpa titik akhir EC2 ditentukan, panggilan untuk menghitung volume Amazon EBS terlampir gagal, yang menyebabkan perintah Systems Manager gagal.
   - **com.amazonaws.*region*.ssmmessages**— Titik akhir ini hanya diperlukan jika Anda terhubung ke instans Anda melalui saluran data aman menggunakan. Session Manager Lihat informasi yang lebih lengkap di AWS Systems Manager Session Manager dan Referensi: ec2messages, ssmmessages, dan operasi API lainnya.
   - **com.amazonaws.*region*.kms**Endpoint ini bersifat opsional. Namun, itu dapat dibuat jika Anda ingin menggunakan enkripsi AWS Key Management Service (AWS KMS) untuk Session Manager atau Parameter Store parameter.
   - **com.amazonaws.*region*.logs**Endpoint ini bersifat opsional. Namun, itu dapat dibuat jika Anda ingin menggunakan Amazon CloudWatch Logs (CloudWatch Log) untukSession Manager,Run Command, atau SSM Agent log.

2. Ikuti langkah-langkah di Buat titik akhir gateway untuk membuat titik akhir gateway berikut untuk Amazon S3.

   - **com.amazonaws.*region*.s3**— Systems Manager menggunakan endpoint ini untuk memperbarui SSM Agent dan melakukan operasi patching. Systems Manager juga menggunakan endpoint ini untuk tugas-tugas seperti mengunggah log keluaran yang Anda pilih untuk disimpan di bucket S3, mengambil skrip atau file lain yang Anda simpan di bucket, dan sebagainya. Jika grup keamanan yang terkait dengan instans membatasi lalu lintas keluar, Anda harus menambahkan aturan untuk mengizinkan lalu lintas ke daftar awalan untuk Amazon S3. Untuk informasi selengkapnya, lihat Memodifikasi grup keamanan Anda di AWS PrivateLink Panduan.

Untuk informasi tentang bucket S3 AWS terkelola yang SSM Agent harus dapat diakses, lihat. SSM Agentkomunikasi dengan bucket S3 AWS terkelola Jika Anda menggunakan titik akhir virtual private cloud (VPC) dalam operasi Systems Manager, Anda harus memberikan izin eksplisit dalam profil instans EC2 untuk Systems Manager, atau dalam peran layanan untuk node terkelola non-EC2 di lingkungan hybrid dan multicloud.

Penciptaan sebuah kebijakan VPC endpoint antarmuka

Anda dapat membuat kebijakan untuk titik akhir antarmuka VPC yang dapat Anda tentukan: AWS Systems Manager

- Prinsip-prinsip yang dapat melakukan tindakan.
- Tindakan yang dapat dilakukan
- Sumber daya yang dapat memiliki tindakan yang dilakukan pada mereka

Untuk informasi selengkapnya, lihat Mengontrol akses ke layanan dengan titik akhir VPC di Panduan Pengguna Amazon VPC.

Menyiapkan Manajer Sistem untuk lingkungan hybrid dan multicloud

Anda dapat menggunakanAWS Systems Manageruntuk mengelola instans Amazon Elastic Compute Cloud (EC2) dan sejumlah jenis mesin non-EC2. Bagian ini menjelaskan tugas penyiapan yang dilakukan oleh administrator akun dan sistem untuk mengelola mesin non-EC2 menggunakan Manajer Sistemhybrid dan multicloudlingkungan. Setelah langkah-langkah ini selesai, pengguna yang telah diberikan izin olehAkun AWSadministrator dapat menggunakan Manajer Sistem untuk mengkonfigurasi dan mengelola mesin non-EC2 organisasi mereka.

Setiap mesin yang telah dikonfigurasi untuk digunakan dengan Manajer Sistem disebut anode terkelola.

> **Note**
>
> - Anda dapat mendaftarkan perangkat edge sebagai node terkelola menggunakan langkah aktivasi hibrida yang sama yang digunakan untuk mesin non-EC2 lainnya. Jenis perangkat

tepi ini mencakup keduanyaAWS IoTperangkat dan perangkat selainAWS IoTperangkat. Gunakan proses yang dijelaskan di bagian ini untuk mengatur jenis perangkat tepi ini.

Systems Manager juga mendukung perangkat edge yang menggunakanAWS IoT GreengrassPerangkat lunak inti. Proses penyiapan dan persyaratan untukAWS IoT Greengrassperangkat inti berbeda dari perangkat untukAWS IoTdan perangkat tepi selainAWSperangkat tepi. Untuk informasi tentang pendaftaranAWS IoT Greengrassperangkat untuk digunakan dengan Manajer Sistem, lihatAWS Systems ManagerMenyiapkan perangkat edge.

- Non-EC2macOSmesin tidak didukung untuk lingkungan hybrid dan multicloud Systems Manager.

Jika Anda berencana menggunakan Manajer Sistem untuk mengelola instans Amazon Elastic Compute Cloud (Amazon EC2), atau menggunakan instans Amazon EC2 dan mesin non-EC2 di lingkungan hybrid dan multicloud, ikuti langkah-langkahnyaMenyiapkan Systems Manager untuk instans EC2pertama.

Setelah mengonfigurasi lingkungan hybrid dan multicloud Anda untuk Manajer Sistem, Anda dapat melakukan hal berikut:

- Buat cara yang konsisten dan aman untuk mengelola beban kerja hybrid dan multicloud dari jarak jauh dari satu lokasi menggunakan alat atau skrip yang sama.
- Memusatkan kontrol akses untuk tindakan yang dapat dilakukan pada mesin Anda dengan menggunakanAWS Identity and Access Management(IAM).
- Memusatkan audit operasi yang dilakukan pada mesin Anda dengan melihat aktivitas API yang direkamAWS CloudTrail.

  Untuk informasi tentang penggunaan CloudTrail untuk memantau tindakan Manajer Sistem, lihatPencatatan panggilan AWS Systems Manager API dengan AWS CloudTrail.
- Memusatkan pemantauan dengan mengonfigurasi Amazon EventBridge dan Amazon Simple Notification Service (Amazon SNS) untuk mengirim pemberitahuan tentang keberhasilan eksekusi layanan.

  Untuk informasi tentang penggunaan EventBridge untuk memantau peristiwa Manajer Sistem, lihatPemantauan peristiwa Systems Manager dengan Amazon EventBridge.

Tentang node terkelola

Setelah Anda selesai mengonfigurasi mesin non-EC2 Anda untuk Manajer Sistem seperti yang dijelaskan di bagian ini, mesin yang diaktifkan hibrida Anda tercantum dalamAWS Management Consoledan digambarkan sebagainode terkelola. Di konsol, ID node terkelola yang diaktifkan hibrida dibedakan dari instans Amazon EC2 dengan awalan “mi-”. Instans Amazon EC2 dengan ID yang menggunakan prefiks "i-".

Node terkelola adalah mesin apa pun yang dikonfigurasi untuk Manajer Sistem. Sebelumnya, node terkelola semuanya disebut sebagai instance terkelola. Istilahcontohsekarang mengacu pada instans EC2 saja. Thederegister-managed-instanceperintah dinamai sebelum perubahan terminologi ini.

Untuk informasi selengkapnya, lihat Bekerja dengan node terkelola.

Tingkatan instans

Systems Manager menawarkan tingkat instans standar dan tingkat instans lanjutan untuk node terkelola non-EC2 di lingkungan hybrid dan multicloud Anda. Tingkat instans standar memungkinkan Anda mendaftarkan maksimum 1.000 mesin yang diaktifkan hibrida perAkun AWSperWilayah AWS. Jika Anda perlu mendaftarkan lebih dari 1.000 mesin non-EC2 dalam satu akun dan Wilayah, gunakan tingkat instans lanjutan. Instans lanjutan juga memungkinkan Anda untuk terhubung ke mesin non-EC2 Anda dengan menggunakanAWS Systems Manager Session Manager.Session Managermenyediakan akses shell interaktif ke node terkelola Anda.

Untuk informasi selengkapnya, lihat Mengonfigurasi tingkat instans.

Topik

- Langkah 1: Membuat peran layanan IAM untuk lingkungan hybrid dan multicloud
- Langkah 2: Buat aktivasi hybrid untuk lingkungan hybrid dan multicloud
- Langkah 3: Instal SSM Agent untuk lingkungan hybrid dan multicloud (Linux)
- Langkah 4: Instal SSM Agent untuk lingkungan hybrid dan multicloud () Windows

Langkah 1: Membuat peran layanan IAM untuk lingkungan hybrid dan multicloud

Mesin non-EC2 (Amazon Elastic Compute Cloud) dalam lingkungan hybrid dan multicloud memerlukan peran layanan AWS Identity and Access Management (IAM) untuk berkomunikasi dengan layanan. AWS Systems Manager Memberikan kepercayaan peran AWS Security Token

Service(AWS STS)`AssumeRole` untuk layanan Systems Manager. Anda hanya perlu membuat peran layanan untuk lingkungan hybrid dan multicloud satu kali untuk masing-masing. Akun AWS Namun, Anda dapat memilih untuk membuat beberapa peran layanan untuk aktivasi hibrid yang berbeda jika mesin di lingkungan hybrid dan multicloud Anda memerlukan izin yang berbeda.

Prosedur berikut menjelaskan cara membuat peran layanan yang diperlukan menggunakan konsol Manajer Sistem atau alat baris perintah pilihan Anda.

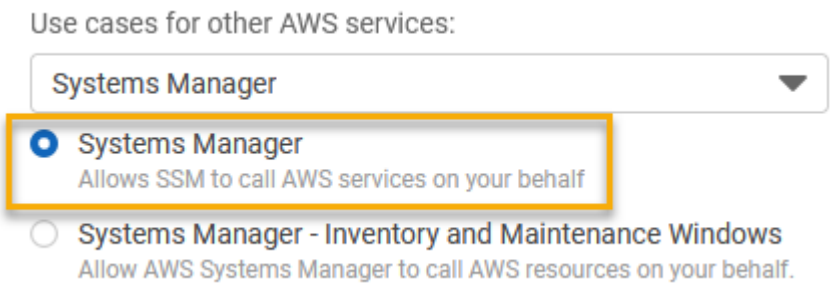
Membuat peran layanan IAM (konsol)

Gunakan prosedur berikut untuk membuat peran layanan untuk aktivasi hibrida. Prosedur ini menggunakan `AmazonSSMManagedInstanceCore` kebijakan untuk fungsi inti Manajer Sistem. Bergantung pada kasus penggunaan, Anda mungkin perlu menambahkan kebijakan tambahan ke peran layanan agar mesin lokal dapat mengakses kemampuan lain atauLayanan AWS. Misalnya, tanpa akses ke bucket Amazon Simple Storage Service (Amazon S3) AWS terkelola yang diperlukan, operasi Patch Manager patching gagal.

Membuat peran layanan (konsol)

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran, lalu pilih Buat peran.
3. Untuk Pilih entitas tepercaya, buat pilihan berikut:

    1. Untuk tipe entitas tepercaya, pilih Layanan AWS.
    2. Untuk Kasus penggunaan untuk kasus lainnya Layanan AWS, pilih Manajer Sistem.
    3. Pilih Manajer Sistem, seperti yang ditunjukkan pada gambar berikut.



4. Pilih Selanjutnya.
5. Pada halaman Tambahkan izin, lakukan hal berikut:

    - Gunakan kolom Pencarian untuk menemukan kebijakan ManagedInstanceCoreAmazonSSM. Pilih kotak centang di samping namanya.

- Konsol mempertahankan pilihan Anda bahkan jika Anda mencari kebijakan lain.
- Jika Anda membuat kebijakan bucket S3 khusus dalam prosedur(Opsional) Buat kebijakan khusus untuk akses bucket S3, cari dan centang kotak di samping namanya.
- Jika Anda berencana untuk bergabung dengan mesin non-EC2 ke Active Directory yang dikelola olehAWS Directory Service, cari AmazonSSM DirectoryServiceAccess dan centang kotak di samping namanya.
- Jika Anda berencana untuk menggunakan EventBridge atau CloudWatch Log untuk mengelola atau memantau node terkelola, cari CloudWatchAgentServerPolicydan centang kotak di samping namanya.

6. Pilih Selanjutnya.
7. Untuk Nama peran, masukkan nama untuk peran server IAM baru Anda, seperti**`SSMServerRole`**.

> **Note**
>
> Buat catatan tentang nama peran. Anda akan memilih peran ini ketika Anda mendaftarkan mesin baru yang ingin Anda kelola dengan menggunakan Manajer Sistem.

8. (Opsional) Untuk Deskripsi, perbarui deskripsi untuk peran server IAM ini.
9. (Opsional) Untuk Tag, tambahkan satu atau lebih pasangan nilai tag-key untuk mengatur, melacak, atau mengontrol akses untuk peran ini.
10. Pilih Buat peran. Sistem mengembalikan Anda ke halaman Peran.

Buat peran layanan IAM (baris perintah)

Gunakan prosedur berikut untuk membuat peran layanan untuk aktivasi hibrida. Prosedur ini menggunakan `AmazonSSMManagedInstanceCore` kebijakan fungsi inti Manajer Sistem. Bergantung pada kasus penggunaan, Anda mungkin perlu menambahkan kebijakan tambahan ke

peran layanan untuk mesin non-EC2 di lingkungan hybrid dan multicloud agar dapat mengakses kemampuan lain atau. Layanan AWS

Persyaratan kebijakan bucket S3

Jika salah satu dari kasus berikut ini benar, Anda harus membuat kebijakan izin IAM khusus untuk bucket Amazon Simple Storage Service (Amazon S3) sebelum menyelesaikan prosedur ini:

- Kasus 1 — Anda menggunakan endpoint VPC untuk menghubungkan VPC Anda secara pribadi ke layanan endpoint VPC yang didukung Layanan AWS dan didukung oleh VPC. AWS PrivateLink
- Kasus 2 — Anda berencana untuk menggunakan bucket Amazon S3 yang Anda buat sebagai bagian dari operasi Manajer Sistem, seperti untuk menyimpan output untuk Run Command perintah atau Session Manager sesi ke bucket S3. Sebelum melanjutkan, ikuti langkah-langkah di Membuat kebijakan bucket S3 kustom untuk profil instans. Informasi tentang kebijakan bucket S3 dalam topik tersebut juga berlaku untuk peran layanan Anda.

AWS CLI

Untuk membuat peran layanan IAM untuk lingkungan hybrid dan multicloud () AWS CLI

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaru AWS CLI.
2. Di komputer lokal Anda, buat file teks dengan nama seperti `SSMService-Trust.json` dengan kebijakan kepercayaan berikut. Pastikan Anda menyimpan file dengan ekstensi file `.json`. Pastikan untuk menentukan Anda Akun AWS dan Wilayah AWS di ARN tempat Anda membuat aktivasi hybrid Anda.

```
{
   "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. BukaAWS CLI, dan di direktori tempat Anda membuat file JSON, jalankan perintah create-role untuk membuat peran layanan. Contoh ini membuat peran bernama `SSMServiceRole`. Anda dapat memilih nama lain jika Anda suka.

   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. Jalankan attach-role-policyperintah sebagai berikut untuk memungkinkan peran layanan yang baru saja Anda buat untuk membuat token sesi. Token sesi memberikan izin node terkelola Anda untuk menjalankan perintah menggunakan Manajer Sistem.

   > **Note**
   >
   > Kebijakan yang Anda tambahkan untuk profil layanan untuk node terkelola dalam lingkungan hybrid dan multicloud adalah kebijakan yang sama yang digunakan untuk membuat profil instans untuk instans Amazon Elastic Compute Cloud (Amazon EC2). Untuk informasi selengkapnya tentang AWS kebijakan yang digunakan dalam perintah berikut, lihat Mengonfigurasi izin instans untuk Manajer Sistem.

(Diperlukan) Jalankan perintah berikut untuk memungkinkan node yang dikelola menggunakan fungsionalitas inti AWS Systems Manager layanan.

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
```

Jika Anda membuat kebijakan bucket S3 khusus untuk peran layanan Anda, jalankan perintah berikut untuk mengizinkan AWS Systems Manager Agen (SSM Agent) mengakses bucket yang Anda tentukan dalam kebijakan. Ganti *account-id* dan *my-bucket-policy-name*dengan Akun AWS ID Anda dan nama bucket Anda.

Linux & 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
```

(Opsional) Jalankan perintah berikut SSM Agent untuk memungkinkan akses AWS Directory Service atas nama Anda untuk permintaan bergabung dengan domain oleh node yang dikelola. Peran layanan Anda memerlukan kebijakan ini hanya jika Anda menggabungkan node ke direktori 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
```

(Opsional) Jalankan perintah berikut untuk memungkinkan CloudWatch agen berjalan di node yang dikelola. Perintah ini memungkinkan untuk membaca informasi pada node dan menuliskannyaCloudWatch. Profil layanan Anda memerlukan kebijakan ini hanya jika Anda akan menggunakan layanan seperti Amazon EventBridge atau Amazon CloudWatch Logs.

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

Tools for PowerShell

Untuk membuat peran layanan IAM untuk lingkungan hybrid dan multicloud () AWS Tools for Windows PowerShell

1. Instal dan konfigurasikan AWS Tools for PowerShell (Alat untuk WindowsPowerShell), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal AWS Tools for PowerShell.

2. Di komputer lokal Anda, buat file teks dengan nama seperti `SSMService-Trust.json` dengan kebijakan kepercayaan berikut. Pastikan Anda menyimpan file dengan ekstensi file `.json`. Pastikan untuk menentukan Anda Akun AWS dan Wilayah AWS di ARN tempat Anda membuat aktivasi hybrid Anda.

   ```
   {
      "Version":"2012-10-17",
   ```

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

3. Buka PowerShell dalam mode administratif, dan di direktori tempat Anda membuat file JSON, jalankan New-IAMRole sebagai berikut untuk membuat peran layanan. Contoh ini membuat peran bernama `SSMServiceRole`. Anda dapat memilih nama lain jika Anda suka.

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

4. Gunakan Register-IAM RolePolicy sebagai berikut untuk memungkinkan peran layanan yang Anda buat untuk membuat token sesi. Token sesi memberikan izin node terkelola Anda untuk menjalankan perintah menggunakan Manajer Sistem.

> **Note**
>
> Kebijakan yang Anda tambahkan untuk profil layanan untuk node terkelola di lingkungan hibrid dan multicloud adalah kebijakan yang sama yang digunakan untuk membuat profil instans untuk instans EC2. Untuk informasi selengkapnya tentang AWS kebijakan yang digunakan dalam perintah berikut, lihat Mengonfigurasi izin instans untuk Manajer Sistem.

(Diperlukan) Jalankan perintah berikut untuk memungkinkan node yang dikelola menggunakan fungsionalitas inti AWS Systems Manager layanan.

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

Jika Anda membuat kebijakan bucket S3 khusus untuk peran layanan Anda, jalankan perintah berikut SSM Agent untuk memungkinkan mengakses bucket yang Anda tentukan dalam kebijakan. Ganti *account-id* dan *my-bucket-policy-name*dengan Akun AWS ID Anda dan nama bucket Anda.

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

(Opsional) Jalankan perintah berikut SSM Agent untuk memungkinkan akses AWS Directory Service atas nama Anda untuk permintaan bergabung dengan domain oleh node yang dikelola. Peran server Anda memerlukan kebijakan ini hanya jika Anda bergabung dengan node Anda ke direktori Microsoft AD.

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

(Opsional) Jalankan perintah berikut untuk memungkinkan CloudWatch agen berjalan di node yang dikelola. Perintah ini memungkinkan untuk membaca informasi pada node dan menuliskannyaCloudWatch. Profil layanan Anda memerlukan kebijakan ini hanya jika Anda akan menggunakan layanan seperti Amazon EventBridge atau Amazon CloudWatch Logs.

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

Lanjutkan ke Langkah 2: Buat aktivasi hybrid untuk lingkungan hybrid dan multicloud.

Langkah 2: Buat aktivasi hybrid untuk lingkungan hybrid dan multicloud

Untuk menyiapkan mesin selain instans Amazon Elastic Compute Cloud (EC2) sebagai node terkelola untuk lingkungan hybrid dan multicloud, Anda membuat dan menerapkan aktivasi hibrid. Setelah Anda berhasil menyelesaikan aktivasi, Anda segera menerima Kode Aktivasi dan ID Aktivasi di bagian atas halaman konsol. Anda menentukan kombinasi Kode dan ID ini saat Anda menginstal AWS Systems Manager SSM Agent pada mesin non-EC2 untuk lingkungan hybrid dan multicloud Anda. Kode dan ID menyediakan akses aman ke layanan Manajer Sistem dari node terkelola Anda.

> **Important**
>
> Systems Manager akan segera mengembalikan Kode Aktivasi dan ID ke konsol atau jendela perintah, tergantung pada bagaimana Anda membuat aktivasi. Salin informasi ini dan simpan di tempat yang aman. Jika Anda menavigasi keluar dari konsol atau menutup jendela perintah, Anda mungkin kehilangan informasi ini. Jika Anda kehilangannya, Anda harus membuat aktivasi baru.

Tentang aktivasi kedaluwarsa

Aktivasi kedaluwarsa adalah jangka waktu ketika Anda dapat mendaftarkan mesin on-premise dengan Systems Manager. Aktivasi kedaluwarsa tidak berdampak pada server atau VM Anda yang sebelumnya Anda daftarkan dengan Systems Manager. Jika aktivasi kedaluwarsa maka Anda tidak dapat mendaftar lebih banyak server atau VM dengan Systems Manager dengan menggunakan aktivasi tertentu. Anda hanya perlu membuat yang baru.

Setiap server lokal dan VM yang sebelumnya Anda daftarkan tetap terdaftar sebagai node yang dikelola Manajer Sistem hingga Anda membatalkan pendaftaran secara eksplisit. Anda dapat membatalkan pendaftaran node yang dikelola pada tab Managed nodes Fleet Manager di konsol Systems Manager dengan menggunakan AWS CLI perintah deregister-managed-instance, atau dengan menggunakan panggilan API. DeregisterManagedInstance

Tentang node terkelola

Sebuah node dikelola adalah setiap mesin dikonfigurasi untukAWS Systems Manager. AWS Systems Managermendukung instans Amazon Elastic Compute Cloud (Amazon EC2), perangkat edge, dan server lokal atau VM, termasuk VM di lingkungan cloud lainnya. Sebelumnya, node yang dikelola semuanya disebut sebagai instance terkelola. Istilah contoh sekarang mengacu pada contoh EC2 saja. deregister-managed-instancePerintah itu dinamai sebelum perubahan terminologi ini.

Tentang tag aktivasi

Jika Anda membuat aktivasi dengan menggunakan AWS Command Line Interface (AWS CLI) atau AWS Tools for Windows PowerShell, Anda dapat menentukan tag. Tag adalah metadata opsional yang Anda tetapkan ke sumber daya. Tag memungkinkan Anda untuk mengkategorikan sumber daya dengan berbagai cara, seperti berdasarkan tujuan, pemilik, atau lingkungan. Berikut adalah contoh perintah AWS CLI untuk berjalan pada mesin Linux on-premise yang mencakup tag opsional.

```
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"
```

Jika Anda menentukan tag saat Anda membuat aktivasi, maka tag tersebut secara otomatis ditetapkan ke node yang dikelola saat Anda mengaktifkannya.

Anda tidak dapat menambahkan tag ke atau menghapus tag dari aktivasi yang ada. Jika Anda tidak ingin secara otomatis menetapkan tag ke server on-premise dan VM menggunakan aktivasi, Anda dapat menambahkan tag ke server tersebut nanti. Lebih khusus lagi, Anda dapat menandai server on-premise dan VM setelah tersambung ke Systems Manager untuk pertama kalinya. Setelah terhubung, mereka diberi ID node terkelola dan terdaftar di konsol Manajer Sistem dengan ID yang diawali dengan “mi-”. Untuk informasi tentang cara menambahkan tag ke node yang dikelola tanpa menggunakan proses aktivasi, lihatMenandai node terkelola.

> **Note**
>
> Anda tidak dapat menetapkan tag untuk aktivasi jika Anda membuatnya dengan menggunakan konsol Systems Manager. Anda harus membuatnya dengan menggunakan alat AWS CLI atau untuk WindowsPowerShell.

Jika Anda tidak lagi ingin mengelola server on-premise atau mesin virtual (VM) dengan menggunakan Systems Manager, Anda dapat membatalkan pendaftaran. Untuk informasi, lihat Menderegistrasi node terkelola dalam lingkungan hybrid dan multicloud.

Topik

- Membuat aktivasi (konsol)
- Buat aktivasi dikelola-node (baris perintah)

Membuat aktivasi (konsol)

Untuk membuat aktivasi dikelola-node

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Pengaktifan hibrida.

   -atau-

   Jika membuka halaman beranda AWS Systems Manager terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, dan kemudian memilih Aktivasi hibrida.
3. Pilih Buat aktivasi.

   -atau-

   Jika Anda mengakses Aktivasi Hibrid untuk pertama kalinya di saat iniWilayah AWS, pilih Buat Aktivasi.
4. (Opsional) Untuk deskripsi Aktivasi, masukkan deskripsi untuk aktivasi ini. Kami sarankan memasukkan deskripsi jika Anda berencana untuk mengaktifkan sejumlah besar server dan VM.
5. Untuk batas Instance, tentukan jumlah total node yang ingin Anda daftarkan AWS sebagai bagian dari aktivasi ini. Nilai default adalah 1 instans .
6. Untuk peran IAM, pilih opsi peran layanan yang memungkinkan server dan VM Anda berkomunikasi AWS Systems Manager di cloud:

   - Opsi 1: Pilih Gunakan peran default yang dibuat oleh sistem untuk menggunakan peran dan kebijakan terkelola yang disediakan olehAWS.
   - Opsi 2: Pilih peran IAM kustom yang ada yang memiliki izin yang diperlukan untuk menggunakan peran kustom opsional yang Anda buat sebelumnya. Peran ini harus memiliki kebijakan hubungan kepercayaan yang menentukan `"Service": "ssm.amazonaws.com"`. Jika IAM role Anda tidak menentukan prinsip ini dalam kebijakan hubungan kepercayaan, Anda akan menerima kesalahan berikut:

     ```
     An error occurred (ValidationException) when calling the CreateActivation
     ```

```
                                        operation: Not existing role:
arn:aws:iam::<accountid>:role/SSMRole
```

Untuk informasi lebih lanjut tentang pembuatan peran, lihat Langkah 1: Membuat peran layanan IAM untuk lingkungan hybrid dan multicloud.

7. Untuk tanggal kedaluwarsa Aktivasi, tentukan tanggal kedaluwarsa untuk aktivasi. Tanggal kedaluwarsa harus di masa depan, dan tidak lebih dari 30 hari ke depan. Nilai default adalah 24 jam.

> **Note**
>
> Jika Anda ingin mendaftarkan node terkelola tambahan setelah tanggal kedaluwarsa, Anda harus membuat aktivasi baru. Tanggal kedaluwarsa tidak berdampak pada node yang terdaftar dan berjalan.

8. (Opsional) Untuk bidang nama contoh default, tentukan nilai nama pengidentifikasi yang akan ditampilkan untuk semua node terkelola yang terkait dengan aktivasi ini.
9. Pilih Buat aktivasi. Systems Manager akan segera mengembalikan kode aktivasi dan ID ke konsol.

Buat aktivasi dikelola-node (baris perintah)

Prosedur berikut menjelaskan cara menggunakan AWS Command Line Interface (AWS CLI) (di Linux atauWindows) atau AWS Tools for PowerShell untuk membuat aktivasi node yang dikelola.

Untuk membuat aktivasi

1. Instal dan konfigurasikan AWS CLI atau AWS Tools for PowerShell, jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaru AWS CLI dan Memasang AWS Tools for PowerShell.
2. Jalankan perintah berikut untuk membuat aktivasi.

Note

- Dalam perintah berikut, ganti *wilayah* dengan informasi Anda sendiri. Untuk daftar nilai *wilayah* yang didukung, lihat kolom Wilayah di endpoint layanan Manajer Sistem di Referensi Umum Amazon Web Services.
- Peran yang Anda tentukan untuk parameter *iam-role* harus memiliki kebijakan hubungan kepercayaan yang menentukan `"Service": "ssm.amazonaws.com"`. Jika (IAM) role Anda AWS Identity and Access Management tidak menentukan prinsip ini dalam kebijakan hubungan kepercayaan, Anda akan menerima kesalahan berikut:

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

  Untuk informasi lebih lanjut tentang pembuatan peran, lihat Langkah 1: Membuat peran layanan IAM untuk lingkungan hybrid dan multicloud.
- Untuk `--expiration-date`, memberikan tanggal dalam format timestamp, seperti `"2021-07-07T00:00:00"`, untuk saat kode aktivasi kedaluwarsa. Anda dapat menentukan tanggal hingga 30 hari sebelumnya. Jika Anda tidak memberikan tanggal kedaluwarsa, kode aktivasi akan kedaluwarsa dalam 24 jam.

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"}
```

Ini contohnya.

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"}
```

Jika aktivasi berhasil dibuat, sistem akan segera mengembalikan kode aktivasi dan ID.

Langkah 3: Instal SSM Agent untuk lingkungan hybrid dan multicloud (Linux)

Topik ini menjelaskan cara menginstal AWS Systems Manager SSM Agent pada mesin Linux non EC2 (Amazon Elastic Compute Cloud) di lingkungan hybrid dan multicloud. Jika Anda berencana untuk menggunakan Windows Server mesin di lingkungan hybrid dan multicloud, lihat langkah selanjutnya. Langkah 4: Instal SSM Agent untuk lingkungan hybrid dan multicloud () Windows

> **Important**
>
> Prosedur ini adalah jenis mesin selain instans EC2 untuk lingkungan hybrid dan multicloud. Untuk mengunduh dan menginstal SSM Agent pada instans EC2 untuk Linux, lihatBekerja denganSSM Agent instans EC2 untuk Linux.

Sebelum Anda mulai, cari Kode Aktivasi dan ID Aktivasi yang dikirimkan kepada Anda setelah Anda menyelesaikan aktivasi hibrida sebelumnyaLangkah 2: Buat aktivasi hybrid untuk lingkungan hybrid dan multicloud. Anda menentukan kode dan ID dalam prosedur berikut.

`wilayah` mewakili pengenal untuk Wilayah AWS didukung oleh AWS Systems Manager, seperti `us-east-2` untuk Wilayah Timur AS (Ohio). Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

Misalnya, SSM Agent untuk mengunduh Amazon Linux, CentOSRHEL, dan SLES 64-bit dari Wilayah AS Timur (Ohio) (us-timur-2), gunakan URL berikut:

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

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

- x86_64

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

- x86

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

- ARM64

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

RHEL 6.x, CentOS 6.x

- x86_64

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

- x86

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

Ubuntu Server

- x86_64

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

- ARM64

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

- x86

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

Debian Server

- x86_64

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

- ARM64

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

Raspberry Pi OS (formerly Raspbian)

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

Untuk menginstal SSM Agent pada mesin non-EC2 di lingkungan hybrid dan multicloud

1. Masuk ke server atau VM di lingkungan hybrid dan multicloud Anda.
2. Jika Anda menggunakan proxy HTTP atau HTTPS, Anda harus mengatur `http_proxy` atau variabel lingkungan `https_proxy` di sesi shell saat ini. Jika Anda tidak menggunakan proxy, Anda dapat melewati langkah ini.

   Untuk server proxy HTTP, masukkan perintah berikut pada baris perintah:

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

   Untuk server proxy HTTPS, masukkan perintah berikut pada baris perintah:

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

3. Copy dan paste salah satu blok perintah berikut ke SSH. Ganti nilai placeholder dengan Kode Aktivasi dan ID Aktivasi yang dihasilkan saat Anda membuat aktivasi simpul terkelola, dan dengan pengenal yang ingin Wilayah AWS Anda unduhSSM Agent, lalu tekan. `Enter`

> **Note**
>
> Perhatikan detail penting berikut ini:
>
> - `sudo` tidak diperlukan jika Anda adalah pengguna root.
> - Unduh `ssm-setup-cli` dari tempat yang Wilayah AWS sama dengan tempat aktivasi hybrid Anda dibuat.
> - `ssm-setup-cli`mendukung `manifest-url` opsi yang menentukan sumber dari mana agen diunduh. Jangan tentukan nilai untuk opsi ini kecuali diperlukan oleh organisasi Anda.
> - Saat mendaftarkan instance, hanya gunakan tautan unduhan yang disediakan`ssm-setup-cli`. `ssm-setup-cli`seharusnya tidak disimpan secara terpisah untuk penggunaan di masa mendatang.
> - Anda dapat menggunakan skrip yang disediakan di sini untuk memvalidasi tanda tangan. `ssm-setup-cli`

`wilayah` mewakili pengenal untuk Wilayah AWS didukung oleh AWS Systems Manager, seperti `us-east-2` untuk Wilayah Timur AS (Ohio). Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

Selain itu, `ssm-setup-cli` termasuk opsi berikut:

- `version`- Nilai yang valid adalah `latest` dan`stable`.
- `downgrade`- Memungkinkan SSM Agent untuk diturunkan ke versi sebelumnya. Tentukan `true` untuk menginstal versi agen yang lebih lama.
- `skip-signature-validation`- Melewatkan validasi tanda tangan selama pengunduhan dan pemasangan agen.

RHEL6.x, dan CentOS 6.x

```
mkdir /tmp/ssm
```

```
curl https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/3.0.1479.0/linux_amd64/
amazon-ssm-agent.rpm -o /tmp/ssm/amazon-ssm-agent.rpm
sudo yum install -y /tmp/ssm/amazon-ssm-agent.rpm
sudo stop amazon-ssm-agent
sudo -E amazon-ssm-agent -register -code "activation-code" -id "activation-id" -region
 "region"
sudo start amazon-ssm-agent
```

Amazon Linux 1

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

Amazon Linux 2, RHEL 7.xOracle Linux, dan CentOS 7.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"
```

RHEL 8.x dan 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(sebelumnya 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"
```

SLES

```
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

- Menggunakan paket.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"
  ```

- Menggunakan paket Snap

  Anda tidak perlu menentukan URL untuk unduhan, karena perintah `snap` secara otomatis mengunduh agen dari menyimpan aplikasi Snap di https://snapcraft.io.

  Pada Ubuntu Server 20.10 STR & 20.04, 18.04, dan 16.04 LTS, file SSM Agent installer, termasuk binari agen dan file konfigurasi, disimpan dalam direktori berikut:. `/snap/amazon-ssm-agent/current/` Jika Anda membuat perubahan ke file konfigurasi dalam direktori ini, maka Anda harus

menyalin file-file ini dari direktori `/snap` ke `/etc/amazon/ssm/`. File log dan perpustakaan belum berubah (`/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**
>
> Kanal kandidat di toko Snap berisi versi terbaruSSM Agent; bukan saluran stabil. Jika Anda ingin melacak informasi SSM Agent versi pada kanal kandidat, jalankan perintah berikut pada node terkelola Ubuntu Server 18.04 dan 16.04 LTS 64-bit Anda.
>
> ```
> sudo snap switch --channel=candidate amazon-ssm-agent
> ```

Perintah mengunduh dan menginstal SSM Agent ke mesin yang diaktifkan hibrida di lingkungan hybrid dan multicloud Anda. Perintah berhentiSSM Agent, dan kemudian mendaftarkan mesin dengan layanan Systems Manager. Mesin sekarang menjadi node terkelola. Instans Amazon EC2 yang dikonfigurasi untuk Systems Manager juga merupakan node yang dikelola. Namun, di konsol Systems Manager, node yang diaktifkan hibrida dibedakan dari instans Amazon EC2 dengan awalan “mi-”.

Lanjutkan ke Langkah 4: Instal SSM Agent untuk lingkungan hybrid dan multicloud () Windows.

Mengatur rotasi otomatis kunci privat

Untuk memperkuat postur keamanan Anda, Anda dapat mengonfigurasi AWS Systems Manager Agent (SSM Agent) untuk secara otomatis memutar kunci pribadi untuk lingkungan hybrid dan multicloud Anda. Anda dapat mengakses fitur ini menggunakan SSM Agent versi 3.0.1031.0 atau yang lebih baru. Hidupkan fitur ini menggunakan prosedur berikut.

Untuk mengkonfigurasi SSM Agent untuk memutar kunci pribadi untuk lingkungan hybrid dan multicloud

1. `/etc/amazon/ssm/`Arahkan ke mesin Linux atau `C:\Program Files\Amazon\SSM` untuk Windows mesin.

2. Salin isi `amazon-ssm-agent.json.template` ke file baru yang bernama `amazon-ssm-agent.json`. Simpan `amazon-ssm-agent.json` di direktori yang sama dimana dengan lokasi `amazon-ssm-agent.json.template`.
3. Cari `Profile`, `KeyAutoRotateDays`. Masukkan jumlah hari yang Anda inginkan antara rotasi kunci privat otomatis.
4. Mulai ulang SSM Agent.

Setiap kali Anda mengubah konfigurasi, restartSSM Agent.

Anda dapat menyesuaikan fitur lain SSM Agent menggunakan prosedur yang sama. Untuk up-to-date daftar properti konfigurasi yang tersedia dan nilai defaultnya, lihat Config Property Definitions.

Deregister dan registrasi ulang node terkelola

Anda dapat membatalkan pendaftaran node terkelola yang diaktifkan hibrida dengan memanggil operasi DeregisterManagedInstanceAPI baik dari Tools atau Tools untuk AWS CLI Windows. PowerShell Berikut ini adalah contoh perintah CLI:

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

Untuk menghapus informasi pendaftaran yang tersisa untuk agen, hapus `IdentityConsumptionOrder` kunci dalam `amazon-ssm-agent.json` file. Kemudian jalankan perintah berikut:

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

Anda dapat mendaftarkan ulang mesin setelah Anda membatalkan pendaftarannya. Gunakan prosedur berikut untuk mendaftarkan ulang mesin. Setelah Anda menyelesaikan prosedur, node terkelola Anda ditampilkan lagi dalam daftar node terkelola.

Untuk mendaftarkan ulang node terkelola pada mesin Linux non-EC2

1. Connect ke mesin Anda.
2. Jalankan perintah berikut. Pastikan untuk mengganti nilai placeholder dengan Kode Aktivasi dan ID Aktivasi yang dihasilkan saat Anda membuat aktivasi simpul terkelola, dan dengan pengenal Wilayah yang ingin Anda unduh. SSM Agent

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

Mengatasi masalah SSM Agent instalasi pada mesin Linux non-EC2

Gunakan informasi berikut untuk membantu Anda memecahkan masalah saat menginstal SSM Agent pada mesin Linux yang diaktifkan hibrida di lingkungan hybrid dan multicloud.

Anda menerima DeliveryTimedOut kesalahan

Masalah: Saat mengkonfigurasi mesin dalam satu Akun AWS sebagai node terkelola untuk yang terpisah Akun AWS, Anda menerima `DeliveryTimedOut` setelah menjalankan perintah untuk menginstal SSM Agent pada mesin target.

Solusi: `DeliveryTimedOut` adalah kode respon yang diharapkan untuk skenario ini. Perintah untuk menginstal SSM Agent pada node target mengubah ID node dari node sumber. Karena ID node telah berubah, node sumber tidak dapat membalas node target bahwa perintah gagal, selesai, atau habis waktu saat mengeksekusi.

Tidak dapat memuat asosiasi simpul

Masalah: Setelah menjalankan perintah install, Anda melihat kesalahan berikut di log SSM Agent kesalahan:

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

Anda melihat kesalahan ini ketika ID mesin tidak bertahan setelah reboot.

Solusi: Untuk mengatasi masalah ini, jalankan perintah berikut. Perintah ini memaksa ID mesin untuk bertahan setelah reboot.

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

Langkah 4: Instal SSM Agent untuk lingkungan hybrid dan multicloud () Windows

Topik ini menjelaskan cara menginstal SSM Agent pada Windows Server mesin untuk lingkungan hybrid dan multicloud. Jika Anda berencana untuk menggunakan mesin Linux non-EC2 di lingkungan hybrid dan multicloud, lihat langkah sebelumnya,. Langkah 3: Instal SSM Agent untuk lingkungan hybrid dan multicloud (Linux)

> **Important**
>
> Prosedur ini adalah mesin non-EC2 (Amazon Elastic Compute Cloud) di lingkungan hybrid dan multicloud. Untuk mengunduh dan menginstal SSM Agent pada instans EC2Windows Server, lihatBekerja dengan SSM Agent instans EC2 untuk Windows Server.

Sebelum Anda mulai, cari Kode Aktivasi dan ID Aktivasi yang dikirimkan kepada Anda setelah Anda menyelesaikan aktivasi hibrida sebelumnyaLangkah 2: Buat aktivasi hybrid untuk lingkungan hybrid dan multicloud. Anda menentukan kode dan ID dalam prosedur berikut.

Untuk menginstal SSM Agent pada Windows Server mesin non-EC2 di lingkungan hybrid dan multicloud

1. Masuk ke server atau VM di lingkungan hybrid dan multicloud Anda.
2. Jika Anda menggunakan proxy HTTP atau HTTPS, Anda harus mengatur `http_proxy` atau variabel lingkungan `https_proxy` di sesi shell saat ini. Jika Anda tidak menggunakan proxy, Anda dapat melewati langkah ini.

   Untuk server proxy HTTP, atur variabel ini:

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

   Untuk server proksi HTTPS, atur variabel ini:

   ```
   http_proxy=http://hostname:port
   https_proxy=https://hostname:port
   ```
3. Buka Windows PowerShell dalam mode tinggi (administratif).
4. Salin dan tempel blok perintah berikut ke dalamWindows PowerShell. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri. Misalnya, Kode Aktivasi dan ID Aktivasi dihasilkan saat Anda membuat aktivasi hibrida, dan dengan pengenal yang ingin Wilayah AWS Anda unduhSSM Agent.

   > **Note**
   >
   > Perhatikan detail penting berikut ini:

- `ssm-setup-cli`mendukung `manifest-url` opsi yang menentukan sumber dari mana agen diunduh. Jangan tentukan nilai untuk opsi ini kecuali diperlukan oleh organisasi Anda.
- Anda dapat menggunakan skrip yang disediakan di sini untuk memvalidasi tanda tangan. `ssm-setup-cli`
- Saat mendaftarkan instance, hanya gunakan tautan unduhan yang disediakan`ssm-setup-cli`. `ssm-setup-cli`seharusnya tidak disimpan secara terpisah untuk penggunaan di masa mendatang.

`wilayah` mewakili pengenal untuk Wilayah AWS didukung oleh AWS Systems Manager, seperti `us-east-2` untuk Wilayah Timur AS (Ohio). Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

Selain itu, `ssm-setup-cli` termasuk opsi berikut:

- `version`- Nilai yang valid adalah `latest` dan`stable`.
- `downgrade`- Mengembalikan agen ke versi sebelumnya.
- `skip-signature-validation`- Melewatkan validasi tanda tangan selama pengunduhan dan pemasangan agen.

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. Tekan `Enter`.

Perintah ini melakukan hal berikut:

- Mengunduh dan menginstal SSM Agent ke mesin.
- Mendaftarkan mesin dengan layanan Systems Manager.
- Mengembalikan tanggapan yang serupa dengan yang berikut:

```
    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
```

Mesin sekarang menjadi node terkelola. Node terkelola ini sekarang diidentifikasi dengan awalan “mi-”. Anda dapat melihat node terkelola pada halaman Managed node diFleet Manager, dengan menggunakan AWS CLI perintah describe-instance-information, atau dengan menggunakan perintah API DescribeInstanceInformation.

Mengatur rotasi otomatis kunci privat

Untuk memperkuat postur keamanan Anda, Anda dapat mengonfigurasi AWS Systems Manager Agent (SSM Agent) untuk secara otomatis memutar kunci pribadi untuk lingkungan hybrid dan multicloud. Anda dapat mengakses fitur ini menggunakan SSM Agent versi 3.0.1031.0 atau yang lebih baru. Hidupkan fitur ini menggunakan prosedur berikut.

Untuk mengkonfigurasi SSM Agent untuk memutar kunci pribadi untuk lingkungan hybrid dan multicloud

1. `/etc/amazon/ssm/`Arahkan ke mesin Linux atau `C:\Program Files\Amazon\SSM` untuk Windows Server mesin.
2. Salin isi `amazon-ssm-agent.json.template` ke file baru yang bernama `amazon-ssm-agent.json`. Simpan `amazon-ssm-agent.json` di direktori yang sama dimana dengan lokasi `amazon-ssm-agent.json.template`.
3. Cari `Profile`, `KeyAutoRotateDays`. Masukkan jumlah hari yang Anda inginkan antara rotasi kunci privat otomatis.
4. Mulai ulang SSM Agent.

Setiap kali Anda mengubah konfigurasi, restartSSM Agent.

Anda dapat menyesuaikan fitur lain SSM Agent menggunakan prosedur yang sama. Untuk up-to-date daftar properti konfigurasi yang tersedia dan nilai defaultnya, lihat Config Property Definitions.

Deregister dan registrasi ulang node terkelola

Anda dapat membatalkan pendaftaran node terkelola dengan memanggil operasi DeregisterManagedInstanceAPI baik dari Tools AWS CLI atau Tools untuk Windows. PowerShell Berikut ini adalah contoh perintah CLI:

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

Untuk menghapus informasi pendaftaran yang tersisa untuk agen, hapus `IdentityConsumptionOrder` kunci dalam `amazon-ssm-agent.json` file. Kemudian jalankan perintah berikut:

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

Anda dapat mendaftarkan ulang mesin setelah Anda membatalkan pendaftarannya. Gunakan prosedur berikut untuk mendaftarkan ulang mesin sebagai node terkelola. Setelah Anda menyelesaikan prosedur, node terkelola Anda ditampilkan lagi dalam daftar node terkelola.

Untuk mendaftarkan ulang node terkelola pada mesin hybrid Windows

1. Connect ke mesin Anda.
2. Jalankan perintah berikut. Pastikan untuk mengganti nilai placeholder dengan Kode Aktivasi dan ID Aktivasi yang dihasilkan saat Anda membuat aktivasi hibrida, dan dengan pengenal Wilayah yang ingin Anda unduh. 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"
```

AWS Systems ManagerMenyiapkan perangkat edge

Bagian ini menjelaskan pengaturan tugas pengaturan yang dilakukan administrator akun dan sistem untuk mengaktifkan konfigurasi dan pengelolaan perangkat inti dan manajemen perangkat AWS IoT Greengrass inti dan sistem. Setelah Anda menyelesaikan tugas ini, pengguna yang telah diberikan izin oleh Akun AWS administrator dapat menggunakan AWS Systems Manager untuk mengkonfigurasi dan mengelola perangkat AWS IoT Greengrass inti organisasi mereka.

> **Note**
>
> - SSM Agentuntuk AWS IoT Greengrass tidak didukung pada macOS dan Windows 10. Anda tidak dapat menggunakan kemampuan Systems Manager untuk mengelola dan mengonfigurasi perangkat edge yang menggunakan sistem operasi ini.
> - Systems Manager juga mendukung perangkat edge yang tidak dikonfigurasi sebagai perangkat AWS IoT Greengrass inti. Untuk menggunakan Systems Manager untuk mengelola perangkat AWS IoT Core dan perangkat AWS non-edge, Anda harus

mengonfigurasinya menggunakan aktivasi hibrida. Untuk informasi selengkapnya, lihat Menyiapkan Manajer Sistem untuk lingkungan hybrid dan multicloud.

- Untuk menggunakan Session Manager dan menambal aplikasi Microsoft dengan perangkat edge Anda, Anda harus mengaktifkan tingkat lanjutan-lanjutan. Untuk informasi selengkapnya, lihat Mengaktifkan tingkat instans lanjutan.

Sebelum kamu memulai

Pastikan perangkat edge Anda memenuhi persyaratan berikut.

- Perangkat edge Anda harus memenuhi persyaratan untuk dikonfigurasi sebagai perangkat AWS IoT Greengrass inti. Untuk informasi selengkapnya, lihat Menyiapkan perangkat AWS IoT Greengrass inti di Panduan AWS IoT Greengrass Version 2 Developer.
- Perangkat edge Anda harus kompatibel dengan AWS Systems Manager Agen (SSM Agent). Untuk informasi selengkapnya, lihat Sistem operasi yang didukung untuk Systems Manager.
- perangkat tepi Anda harus dapat berkomunikasi dengan layanan Systems Manager di cloud. Systems Manager tidak mendukung perangkat tepi terputus.

Tentang menyiapkan perangkat edge

Menyiapkan AWS IoT Greengrass perangkat untuk Systems Manager melibatkan proses berikut.

> **Note**
>
> Untuk informasi tentang mencopot pemasangan SSM Agent dari perangkat edge, lihat Menghapus instalan AWS Systems Manager Agen di Panduan AWS IoT Greengrass Version 2Pengembang.

Langkah 1: Membuat peran layanan IAM untuk perangkat edge

AWS IoT Greengrassperangkat inti memerlukan peran layanan AWS Identity and Access Management (IAM) untuk berkomunikasiAWS Systems Manager. Memberikan kepercayaan peran AWS Security Token Service(AWS STS)`AssumeRole` untuk layanan Systems Manager. Anda hanya perlu membuat peran layanan satu kali untuk setiapAkun AWS. Anda akan menentukan peran ini untuk `RegistrationRole` parameter saat Anda mengkonfigurasi dan menyebarkan SSM Agent

komponen ke AWS IoT Greengrass perangkat Anda. Jika Anda sudah membuat peran ini saat menyiapkan node non-EC2 untuk lingkungan hybrid dan multicloud, Anda dapat melewati langkah ini.

> **Note**
>
> Pengguna di perusahaan atau organisasi Anda yang akan menggunakan Systems Manager di perangkat edge Anda harus telah diberikan izin dari IAM untuk memanggil API Systems Manager.

Persyaratan kebijakan bucket S3

Jika salah satu dari kasus berikut ini benar, Anda harus membuat kebijakan izin IAM khusus untuk bucket Amazon Simple Storage Service (Amazon S3) sebelum menyelesaikan prosedur ini:

- Kasus 1: Anda menggunakan VPC endpoint untuk menghubungkan VPC Anda secara pribadi yang didukung oleh Layanan AWS dan layanan VPC endpoint yang didukung oleh. AWS PrivateLink
- Kasus 2: Anda berencana untuk menggunakan bucket S3 yang Anda buat sebagai salah satu dari bagian operasi Systems Manager Anda, seperti menyimpan output untuk Run Command perintah atau Session Manager sesi untuk bucket S3. Sebelum melanjutkan, ikuti langkah-langkah di Membuat kebijakan bucket S3 kustom untuk profil instans. Informasi tentang kebijakan bucket S3 dalam topik tersebut juga berlaku untuk peran layanan Anda.

  > **Note**
  >
  > Jika perangkat Anda dilindungi oleh firewall dan Anda berencana untuk menggunakannyaPatch Manager, firewall harus mengizinkan akses ke titik akhir `arn:aws:s3:::patch-baseline-snapshot-`*`region`*`/*` dasar patch. *`wilayah`* mewakili pengenal untuk Wilayah AWS didukung oleh AWS Systems Manager, seperti `us-east-2` untuk Wilayah US East (Ohio). Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Wilayah di endpoint layanan Systems Manager di bagian Referensi Umum Amazon Web Services.

AWS CLI

Untuk membuat peran layanan IAM untuk AWS IoT Greengrass lingkungan () AWS CLI

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaru dari AWS CLI.

2. Pada mesin lokal Anda, buat file teks dengan nama seperti `SSMService-Trust.json` kebijakan kepercayaan berikut. Pastikan Anda menyimpan file dengan ekstensi file `.json`.

   > **Note**
   >
   > Perhatikan namanya. Anda akan menentukannya saat Anda menyebarkan SSM Agent ke perangkat AWS IoT Greengrass inti Anda.

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

3. BukaAWS CLI, dan di direktori tempat Anda membuat file JSON, jalankan perintah create-role untuk membuat peran layanan pada peran layanan create-role. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri.

   Linux & macOS

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

   Jendela

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

4. Jalankan attach-role-policyperintah sebagai berikut untuk mengizinkan untuk membuat token sesi pada peran layanan yang baru saja Anda buat untuk membuat token sesi pada peran layanan yang baru saja Anda buat. Token sesi memberikan izin perangkat tepi Anda untuk menjalankan perintah menggunakan Systems Manager.

> **Note**
>
> Kebijakan yang Anda tambahkan untuk profil layanan pada perangkat edge adalah kebijakan yang sama yang digunakan untuk membuat profil instans untuk membuat profil instans untuk instans Amazon Elastic Compute Cloud (Amazon EC2). Untuk informasi selengkapnya tentang kebijakan IAM yang digunakan dalam perintah berikut, lihat Mengatur izin instans untuk Systems Manager.

(Diperlukan) Jalankan perintah berikut untuk mengizinkan perangkat tepi untuk menggunakan fungsi inti layanan layanan pada perangkat tepi untuk menggunakan fungsi inti AWS Systems Manager layanan.

Linux & macOS

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

Jendela

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

Jika Anda membuat kebijakan kustom bucket S3 untuk peran layanan Anda, jalankan perintah berikut untuk mengizinkan AWS Systems Manager Agent (SSM Agent) untuk mengakses bucket yang Anda tentukan dalam kebijakan. Ganti *Account_ID dan my_bucket_policy_name* dengan ID dan nama bucket Anda. Akun AWS

Linux & macOS

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

Jendela

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

(Opsional) Jalankan perintah berikut untuk mengizinkan untuk mengakses atas nama Anda untuk meminta bergabung dengan domain dari perangkat edge SSM Agent untuk mengakses AWS Directory Service atas nama Anda untuk meminta bergabung dengan domain dari perangkat edge untuk mengakses atas nama Anda untuk meminta bergabung dengan domain dari perangkat edge untuk Peran layanan memerlukan kebijakan ini hanya jika Anda bergabung dengan perangkat edge Anda ke direktori Microsoft AD Microsoft AD.

Linux & macOS

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

Jendela

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

(Opsional) Jalankan perintah berikut untuk mengizinkan CloudWatch agen menjalankan perangkat tepi pada perangkat tepi Anda. Perintah ini memungkinkan untuk membaca informasi pada perangkat dan menuliskannya ke perangkat dan menuliskannya ke perangkatCloudWatch. Peran layanan Anda memerlukan kebijakan ini hanya jika Anda akan menggunakan layanan seperti Amazon EventBridge atau Amazon Logs atau Amazon CloudWatch Logs.

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

Tools for PowerShell

Untuk membuat peran layanan IAM untuk AWS IoT Greengrass lingkungan () AWS Tools for Windows PowerShell

1. Instal dan konfigurasikan AWS Tools for PowerShell (Tools untuk WindowsPowerShell), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal AWS Tools for PowerShell.

2. Pada mesin lokal Anda, buat file teks dengan nama seperti `SSMService-Trust.json` kebijakan kepercayaan berikut. Pastikan Anda menyimpan file dengan ekstensi file `.json`.

   > **Note**
   >
   > Perhatikan namanya. Anda akan menentukannya saat Anda menyebarkan SSM Agent ke perangkat AWS IoT Greengrass inti Anda.

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

3. Buka PowerShell dalam mode administratif, dan di direktori tempat Anda membuat file JSON, jalankan New-IAMRole sebagai berikut untuk membuat peran layanan.

   ```
   New-IAMRole `
       -RoleName SSMServiceRole `
   ```

```
    -AssumeRolePolicyDocument (Get-Content -raw SSMService-Trust.json)
```

4. Gunakan Register-IAM RolePolicy sebagai berikut untuk mengizinkan membuat token sesi pada peran layanan yang Anda buat untuk membuat token sesi pada peran layanan yang Anda buat untuk membuat token sesi pada peran layanan yang Anda buat untuk membuat token sesi pada Token sesi memberikan izin perangkat tepi Anda untuk menjalankan perintah menggunakan Systems Manager.

> **Note**
>
> Kebijakan yang Anda tambahkan untuk peran layanan pada perangkat edge di AWS IoT Greengrass lingkungan adalah kebijakan yang sama yang digunakan untuk membuat profil instans untuk membuat profil instans untuk instans EC2. Untuk informasi selengkapnya tentang AWS kebijakan yang digunakan dalam perintah berikut, lihat Mengatur izin instans untuk Systems Manager.

(Diperlukan) Jalankan perintah berikut untuk mengizinkan perangkat tepi untuk menggunakan fungsi inti layanan layanan pada perangkat tepi untuk menggunakan fungsi inti AWS Systems Manager layanan.

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

Jika Anda membuat kebijakan kustom bucket S3 untuk peran layanan Anda, jalankan perintah berikut untuk mengizinkan SSM Agent untuk mengakses bucket yang Anda tentukan dalam kebijakan. Ganti *Account_ID dan my_bucket_policy_name* dengan ID dan nama bucket Anda. Akun AWS

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

(Opsional) Jalankan perintah berikut untuk mengizinkan untuk mengakses atas nama Anda untuk meminta bergabung dengan domain dari perangkat edge SSM Agent untuk mengakses AWS Directory Service atas nama Anda untuk meminta bergabung dengan domain dari perangkat edge untuk mengakses atas nama Anda untuk meminta bergabung dengan

domain dari perangkat edge untuk Peran layanan memerlukan kebijakan ini hanya jika Anda bergabung dengan perangkat edge Anda ke direktori Microsoft AD Microsoft AD.

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

(Opsional) Jalankan perintah berikut untuk mengizinkan CloudWatch agen menjalankan perangkat tepi pada perangkat tepi Anda. Perintah ini memungkinkan untuk membaca informasi pada perangkat dan menuliskannya ke perangkat dan menuliskannya ke perangkatCloudWatch. Peran layanan Anda memerlukan kebijakan ini hanya jika Anda akan menggunakan layanan seperti Amazon EventBridge atau Amazon Logs atau Amazon CloudWatch Logs.

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

Langkah 2: Siapkan AWS IoT Greengrass

Siapkan perangkat edge Anda sebagai perangkat AWS IoT Greengrass inti. Proses penyiapan melibatkan verifikasi sistem operasi yang didukung dan persyaratan sistem, serta menginstal dan mengkonfigurasi perangkat lunak AWS IoT Greengrass Core pada perangkat Anda. Untuk informasi selengkapnya, lihat Menyiapkan perangkat AWS IoT Greengrass inti di Panduan AWS IoT Greengrass Version 2 Developer.

Langkah 3: Perbarui peran pertukaran AWS IoT Greengrass token dan instal SSM Agent di perangkat edge Anda

Langkah terakhir untuk menyiapkan dan mengonfigurasi perangkat AWS IoT Greengrass inti Anda untuk Systems Manager mengharuskan Anda memperbarui peran layanan perangkat AWS IoT Greengrass AWS Identity and Access Management (IAM), yang juga disebut peran pertukaran token, dan menerapkan AWS Systems Manager Agen (SSM Agent) ke perangkat Anda. AWS IoT Greengrass Untuk informasi tentang proses ini, lihat Instal AWS Systems Manager Agen di Panduan AWS IoT Greengrass Version 2 Pengembang.

Setelah Anda men-deploy SSM Agent ke perangkat Anda, AWS IoT Greengrass secara otomatis mendaftarkan perangkat Anda dengan Systems Manager. Tidak diperlukan pendaftaran tambahan. Anda dapat mulai menggunakan kemampuan Systems Manager untuk mengakses, mengelola, dan mengkonfigurasi AWS IoT Greengrass perangkat Anda.

> **Note**
>
> perangkat tepi Anda harus dapat berkomunikasi dengan layanan Systems Manager di cloud. Systems Manager tidak mendukung perangkat tepi terputus.

Menyiapkan administrator yang didelegasikan untuk Manajer Sistem

Saat menyiapkan organisasiAWS Organizations, Anda menetapkan akun manajemen untuk melakukan semua tugas administratif untuk semuaLayanan AWS. Pengguna akun manajemen dapat menetapkan akun administrator yang didelegasikan hanya untuk Manajer Sistem untuk melakukan tugas administratif untukChange Manager,Explorer, dan. OpsCenter AWS Organizationsadalah layanan manajemen akun yang dapat Anda gunakan untuk membuat organisasi dan menetapkan Akun AWS untuk mengelola akun ini secara terpusat. Untuk selengkapnyaAWS Organizations, lihat AWS Organizationsdi Panduan AWS Organizations Pengguna.

Change Manager,Explorer, danOpsCenter, kemampuanAWS Systems Manager, bekerja dengan AWS Organizations untuk melakukan tugas-tugas pada semua akun anggota organisasi Anda. Anda hanya dapat menetapkan satu administrator yang didelegasikan untuk semua kemampuan Manajer Sistem. Akun administrator yang didelegasikan harus menjadi anggota organisasi yang ditugaskan.

Topik

Administrator yang didelegasikan untuk Change Manager

Change Manageradalah kerangka kerja manajemen perubahan perusahaan untuk meminta, menyetujui, menerapkan, dan melaporkan perubahan operasional pada konfigurasi dan infrastruktur aplikasi Anda.

Jika Anda menggunakan Change Manager di seluruh organisasi, tetapkan akun administrator yang didelegasikan untuk mengelola templat perubahan, persetujuan, dan pelaporan untuk semua akun anggota. Menggunakan Pengaturan Cepat, Anda dapat mengatur Change Manager untuk digunakan dengan organisasi dan memilih akun administrator yang didelegasikan. Jika Anda menggunakan Change Manager dengan satuAkun AWS, akun administrator yang didelegasikan tidak diperlukan.

Secara default, Change Manager menampilkan semua tugas terkait perubahan di akun administrator yang didelegasikan. Untuk petunjuk tentang mengonfigurasi administrator yang didelegasikan saat menyiapkan Change Manager organisasi, lihat. Menyiapkan Change Manager untuk organisasi (akun manajemen)

> **Important**
>
> Jika Anda menggunakan Change Manager di seluruh organisasi, sebaiknya selalu membuat perubahan dari akun administrator yang didelegasikan. Meskipun Anda dapat membuat perubahan dari akun lain di organisasi, perubahan tersebut tidak akan dilaporkan atau dapat dilihat dari akun administrator yang didelegasikan.

Administrator yang didelegasikan untuk Explorer

Exploreradalah dasbor operasi yang dapat disesuaikan yang melaporkan tampilan gabungan data operasi (OpsData) untuk AndaAkun AWS, di seluruhWilayah AWS.

Anda dapat mengonfigurasi akun administrator yang didelegasikan untuk Manajer Sistem untuk mengumpulkan Explorer data dari beberapa Wilayah dan akun dengan menggunakan sinkronisasi data sumber daya. AWS Organizations Administrator yang didelegasikan dapat mencari, memfilter, dan mengumpulkan Explorer data menggunakan, AWS Command Line Interface (AWS CLI)AWS Management Console, atau. AWS Tools for Windows PowerShell

Bila Anda menggunakan akun administrator yang didelegasikan untukExplorer, Anda membatasi jumlah administrator yang dapat membuat atau menghapus sinkronisasi data sumber daya multi-akun dan wilayah ke individu. Akun AWS

Anda dapat menyinkronkan data operasi Akun AWS di semua organisasi Anda dengan menggunakanExplorer. Untuk informasi tentang cara menetapkan administrator yang didelegasikanExplorer, lihat. Mengonfigurasi administrator yang didelegasikan

Administrator yang didelegasikan untuk OpsCenter

OpsCentermenyediakan lokasi sentral di mana insinyur operasi dan profesional TI dapat mengelola item pekerjaan operasional (OpsItems) yang terkait dengan AWS sumber daya. Jika ingin digunakan OpsCenter untuk mengelola OpsItems secara terpusat di seluruh akun, Anda harus menyiapkan organisasi. AWS Organizations

Dengan menggunakan Quick Setup forOpsCenter, Anda dapat menetapkan akun administrator yang didelegasikan dan mengonfigurasi OpsCenter untuk mengelola OpsItems secara terpusat. Untuk informasi selengkapnya, lihat (Opsional) Konfigurasi OpsCenter untuk mengelola OpsItems seluruh akun dengan menggunakan Quick Setup.

Memulai dengan AWS Systems Manager

Gunakan tutorial ini untuk mulai menggunakanAWS Systems Manager. Anda akan mempelajari cara meluncurkan instans Amazon Elastic Compute Cloud (Amazon EC2) yang dikelola oleh Systems Manager, dan cara terhubung ke instans yang dikelola.

Karena Systems Manager adalah kumpulan dari beberapa kemampuan, tidak ada satu panduan atau tutorial yang dapat memperkenalkan seluruh layanan. Tutorial ini memberikan pengantar beberapa kemampuan.

Prasyarat

Sebelum memulai, pastikan Anda telah menyelesaikan langkah-langkah di Menyiapkan Systems Manager untuk instans EC2.

Luncurkan instance menggunakanAMI withSSM Agent prainstal

Anda dapat meluncurkan instans Amazon EC2 dengan menggunakanAWS Management Console seperti yang dijelaskan dalam prosedur berikut. Tutorial ini dimaksudkan untuk membantu Anda meluncurkan instans terkelola pertama dengan cepat, jadi tidak mencakup semua opsi yang memungkinkan.

Untuk meluncurkan sebuah instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari dasbor konsol EC2, di kotak Launch instance, pilih Launch instance, lalu pilih Luncurkan instance dari opsi yang muncul.
3. Untuk Nama dan tag, untuk Nama, masukkan nama deskriptif untuk instans Anda.
4. Untuk Gambar Aplikasi dan OS (Amazon Machine Image), lakukan hal berikut:
   a. Pilih tab Mulai Cepat, lalu pilih Amazon Linux. Ini adalah sistem operasi (OS) untuk instans Anda.
   b. Untuk Amazon Machine Image (AMI), pilih versi HVM Amazon Linux 2.
5. Untuk jenis Instans, dari daftar jenis Instans, pilih konfigurasi perangkat keras untuk instans Anda. Pilih jenis`t2.micro` instans, yang dipilih secara default. Jenis`t2.micro` instans memenuhi syarat untuk TingkatAWS Gratis. DiWilayah AWS mana tidak`t2.micro` tersedia,

Anda dapat menggunakan`t3.micro` instance di bawah Tingkat Gratis. Untuk informasi selengkapnya, lihat AWS Tingkat Gratis .

6. Untuk key pair (login), untuk nama pasangan kunci, pilih pasangan kunci.
7. Untuk Pengaturan jaringan, pilih Edit. Untuk nama grup keamanan, perhatikan bahwa wizard membuat dan memilih grup keamanan untuk Anda. Anda dapat menggunakan grup keamanan ini, atau sebagai alternatif, Anda dapat memilih grup keamanan yang Anda buat sebelumnya menggunakan langkah-langkah berikut ini:

   a. Pilih Pilih grup keamanan yang ada.

   b. Dari grup keamanan umum, pilih grup keamanan Anda dari daftar grup keamanan yang sudah ada.
8. Jika Anda tidak menggunakan Konfigurasi Manajemen Host Default, luaskan bagian Detail lanjutan, dan untuk profil instans IAM, pilih profil instans yang Anda buat saat menyiapkanLangkah 1: Konfigurasikan izin instance untuk Systems Manager.
9. Simpan pilihan default untuk pengaturan konfigurasi lain untuk instans Anda.
10. Tinjau ringkasan konfigurasi instans Anda di panel Ringkasan. Saat Anda siap, pilih instans Luncurkan.
11. Halaman konfirmasi memberi tahu Anda bahwa instans Anda sedang diluncurkan. Pilih Lihat semua instans untuk menutup halaman konfirmasi dan kembali ke konsol.
12. Pada layar Instans, Anda dapat melihat status peluncuran. Hanya butuh waktu singkat untuk peluncuran instans.
13. Proses ini mungkin memerlukan waktu beberapa menit sampai instans ditampilkan sebagai pengelolaan dan siap bagi Anda untuk terhubung dengannya. Untuk memeriksa apakah instans Anda lulus pemeriksaan statusnya, lihat informasi ini di kolom Pemeriksaan Status.

Connect ke instans terkelola Anda

Untuk terhubung ke instans yang dikelola Anda

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.

3. Pilih tombol di samping instans yang ingin Anda hubungkan menggunakan RDP.
4. Di menu Tindakan Node, pilih Mulai sesi terminal.
5. Pilih Connect.

Bersihkan instans Anda

Jika Anda selesai bekerja dengan instans terkelola yang Anda buat untuk tutorial ini, akhiri instans tersebut. Mengakhiri instans akan menghapusnya secara efektif.

Untuk mengakhiri instans Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instances (Instans). Dalam daftar instans, pilih instans.
3. Pilih Status Instans, Akhiri instans.
4. Saat diminta konfirmasi, pilih Akhiri.

   Amazon EC2 akan mati dan mengakhiri instans Anda. Setelah instans Anda diakhiri, instans tersebut tetap terlihat di konsol secara singkat, kemudian entri tersebut akan dihapus secara otomatis. Anda tidak dapat menghapus sendiri instans yang telah diakhiri dari tampilan konsol.

Bekerja dengan SSM Agent

AWS Systems Manager Agent (SSM Agent) adalah perangkat lunak Amazon yang berjalan di instans Amazon Elastic Compute Cloud (Amazon EC2), perangkat edge, server lokal, dan mesin virtual (VM). SSM Agentmemungkinkan Systems Manager untuk memperbarui, mengelola, dan mengkonfigurasi sumber daya ini. Agen memproses permintaan dari layanan Systems Manager di AWS Cloud, dan kemudian menjalankannya seperti yang ditentukan dalam permintaan. SSM Agentkemudian mengirimkan informasi status dan eksekusi kembali ke layanan Systems Manager dengan menggunakan Amazon Message Delivery Service(awalan layanan:`ec2messages`) atau Amazon Message Gateway Service(`ssmmessages`).

Jika Anda memantau lalu lintas, Anda akan melihat bahwa node terkelola Anda berkomunikasi dengan `ec2messages.*` atau `ssmmessages.*` titik akhir. Untuk informasi selengkapnya, lihat Referensi: ec2messages, ssmmessages, dan operasi API lainnya. Untuk informasi tentang porting SSM Agent log ke Amazon CloudWatch Logs, lihatPemantauan AWS Systems Manager.

Konten

SSM Agentreferensi teknis

Gunakan informasi dalam topik ini untuk membantu Anda menerapkan AWS Systems Manager Agen (SSM Agent) dan memahami cara kerja agen.

Topik

SSM Agentversi 3.2.xx perilaku kredensi

SSM Agentmenyimpan satu set kredensi sementara di `/var/lib/amazon/ssm/credentials` (untuk Linux danmacOS) atau `%PROGRAMFILES%\Amazon\SSM\credentials` (untukWindows Server) ketika sebuah instance di-onboard menggunakan Konfigurasi Manajemen Host Default di. Quick Setup Kredensi sementara memiliki izin yang Anda tentukan untuk peran IAM yang Anda pilih untuk Konfigurasi Manajemen Host Default. Di Linux, hanya akun root yang dapat mengakses kredensi ini. PadaWindows Server, hanya akun SYSTEM dan Administrator lokal yang dapat mengakses kredensi ini.

SSM Agentkredensialnya diutamakan

Topik ini menjelaskan informasi penting tentang bagaimana izin SSM Agent diberikan untuk melakukan tindakan pada sumber daya Anda.

> **Note**
>
> Support untuk perangkat edge sedikit berbeda. Anda harus mengonfigurasi perangkat edge Anda untuk menggunakan perangkat lunak AWS IoT Greengrass Core, mengonfigurasi peran

layanan AWS Identity and Access Management (IAM), dan menyebarkan SSM Agent ke perangkat Anda dengan menggunakan. AWS IoT Greengrass Untuk informasi selengkapnya, lihat AWS Systems ManagerMenyiapkan perangkat edge.

Ketika SSM Agent diinstal pada mesin, itu memerlukan izin untuk berkomunikasi dengan layanan Systems Manager. Pada instans Amazon Elastic Compute Cloud (Amazon EC2), izin ini disediakan di profil instans yang dilampirkan ke instans. Pada mesin non-EC2, SSM Agent biasanya mendapatkan izin yang diperlukan dari file kredensi bersama, yang terletak di `/root/.aws/credentials` (Linux danmacOS) atau (). `%USERPROFILE%\.aws\credentials` Windows Server Izin yang diperlukan ditambahkan ke file ini selama proses aktivasi hybrid.

Namun, dalam kasus yang jarang terjadi, mesin mungkin berakhir dengan izin yang ditambahkan ke lebih dari satu lokasi tempat SSM Agent memeriksa izin untuk menjalankan tugasnya.

Misalnya, Anda telah mengonfigurasi instans EC2 untuk dikelola oleh Systems Manager. Konfigurasi itu termasuk melampirkan profil instance. Tetapi kemudian Anda memutuskan untuk juga menggunakan instance itu untuk tugas pengembang atau pengguna akhir dan menginstal AWS Command Line Interface (AWS CLI) di atasnya. Instalasi ini menghasilkan izin tambahan yang ditambahkan ke file kredensial pada instans.

Ketika Anda menjalankan perintah Systems Manager pada instance, SSM Agent mungkin mencoba menggunakan kredensial yang berbeda dari yang Anda harapkan untuk digunakan, seperti dari file kredensial alih-alih profil instance. Ini karena SSM Agent mencari kredensi dalam urutan yang ditentukan untuk rantai penyedia kredensi default.

Note

Di Linux danmacOS, SSM Agent berjalan sebagai pengguna root. Oleh karena itu, variabel lingkungan dan file kredensial yang SSM Agent dicari dalam proses ini adalah milik pengguna root only ()`/root/.aws/credentials`. SSM Agenttidak melihat variabel lingkungan atau file kredensional dari pengguna lain pada instance selama pencarian kredensional.

Rantai penyedia default mencari kredensial dalam urutan sebagai berikut:

1. Variabel lingkungan, jika dikonfigurasi (`AWS_ACCESS_KEY_ID` dan `AWS_SECRET_ACCESS_KEY`).

2. file kredensial bersama (`$HOME/.aws/credentials` untuk Linux dan macOS atau `%USERPROFILE%\.aws\credentials` untuk Windows Server) dengan izin yang disediakan oleh, misalnya, aktivasi hibrid atau instalasi AWS CLI .
3. Peran AWS Identity and Access Management (IAM) untuk tugas jika ada aplikasi yang menggunakan definisi tugas Amazon Elastic Container Service (Amazon ECS) atau operasi API. RunTask
4. Profil instans terlampir ke instans Amazon EC2.
5. Peran IAM dipilih untuk Konfigurasi Manajemen Host Default.

Untuk informasi terkait, lihat topik berikut:

- Profil instans untuk instans EC2 — Konfigurasikan izin instans untuk Systems Manager
- Aktivasi hibrida — Buat aktivasi simpul terkelola untuk lingkungan hibrida
- AWS CLI credentials — Konfigurasi dan pengaturan file kredensi di Panduan Pengguna AWS Command Line Interface
- Rantai penyedia kredensial default – Menentukan Kredensial di Panduan Developer AWS SDK for Go

> **Note**
>
> Topik ini dalam Panduan AWS SDK for Go Pengembang menjelaskan rantai penyedia default dalam hal SDK for Go; namun, prinsip yang sama berlaku untuk mengevaluasi kredensialnya. SSM Agent

Tentang akun ssm-user lokal

Dimulai dengan versi 2.3.50.0 dariSSM Agent, agen membuat akun pengguna lokal yang disebut `ssm-user` dan menambahkannya ke `/etc/sudoers.d` direktori (Linux danmacOS) atau ke grup Administrator (). Windows Server Pada versi agen sebelum 2.3.612.0, akun dibuat pertama kali SSM Agent dimulai atau dimulai ulang setelah instalasi. Pada versi 2.3.612.0 dan yang lebih baru, akun `ssm-user` dibuat saat pertama kali sesi dimulai pada sebuah instans. Ini `ssm-user` adalah pengguna OS default ketika sesi dimulaiSession Manager, kemampuan AWS Systems Manager. Anda dapat mengubah izin dengan memindahkan `ssm-user` ke grup yang kurang istimewa atau dengan mengubah file `sudoers`. `ssm-user`Akun tidak dihapus dari sistem saat SSM Agent dihapus.

Windows ServerAktif, SSM Agent menangani pengaturan kata sandi baru untuk `ssm-user` akun saat setiap sesi dimulai. Tidak ada kata sandi yang ditetapkan untuk `ssm-user` pada instans yang dikelola Linux.

Dimulai dengan SSM Agent versi 2.3.612.0, `ssm-user` akun tidak dibuat secara otomatis pada Windows Server mesin yang digunakan sebagai pengontrol domain. Untuk digunakan Session Manager pada pengontrol Windows Server domain, buat `ssm-user` akun secara manual jika belum ada, dan tetapkan izin Administrator Domain kepada pengguna.

> **Important**
>
> Agar `ssm-user` akun dapat dibuat, profil instance yang dilampirkan pada instance harus memberikan izin yang diperlukan. Untuk selengkapnya, lihat Langkah 2: Verifikasi atau tambahkan izin instans untuk Session Manager.

SSM Agentdan Instance Metadata ServiceIMDS

Systems Manager bergantung pada metadata instans EC2 untuk berfungsi dengan benar. Systems Manager dapat mengakses metadata instans menggunakan versi 1 atau versi 2 dari Instance Metadata Service (IMDSv1 dan IMDSv2). Instans Anda harus dapat mengakses alamat IPv4 dari layanan metadata instans: 169.254.169.254. Untuk informasi lebih lanjut, lihat metadata instans dan data pengguna di Panduan Pengguna Amazon EC2 untuk Instans Linux.

Menjaga SSM Agent up-to-date

Versi terbaru dirilis setiap kali kemampuan baru ditambahkan ke Systems Manager atau pembaruan dibuat untuk kemampuan yang ada. SSM Agent Gagal menggunakan agen versi terbaru dapat mencegah node terkelola Anda menggunakan berbagai kemampuan dan fitur Systems Manager. Untuk alasan itu, kami menyarankan Anda mengotomatiskan proses menjaga agar tetap SSM Agent up to date pada mesin Anda. Untuk informasi, lihat Mengotomatiskan pembaruan ke SSM Agent. Berlangganan halaman Catatan SSM Agent Rilis GitHub untuk mendapatkan pemberitahuan tentang SSM Agent pembaruan.

> **Note**
>
> Versi terbaru dirilis setiap kali kemampuan baru ditambahkan ke Systems Manager atau pembaruan dibuat untuk kemampuan yang ada. SSM Agent Gagal menggunakan agen

versi terbaru dapat mencegah node terkelola Anda menggunakan berbagai kemampuan dan fitur Systems Manager. Untuk alasan itu, kami menyarankan Anda mengotomatiskan proses menjaga agar tetap SSM Agent up to date pada mesin Anda. Untuk informasi, lihat Mengotomatiskan pembaruan ke SSM Agent. Berlangganan halaman Catatan SSM Agent Rilis GitHub untuk mendapatkan pemberitahuan tentang SSM Agent pembaruan.
Amazon Machine Images(AMIs) yang menyertakan secara SSM Agent default dapat memakan waktu hingga dua minggu untuk diperbarui dengan versi terbaruSSM Agent. Kami menyarankan Anda mengonfigurasi pembaruan otomatis yang lebih sering keSSM Agent.

Memastikan bahwa direktori SSM Agent instalasi tidak diubah, dipindahkan, atau dihapus

SSM Agentdiinstal di `/var/lib/amazon/ssm/` (Linux danmacOS) dan `%PROGRAMFILES%\Amazon\SSM\` (Windows Server). Direktori instalasi ini berisi file dan folder penting yang digunakan olehSSM Agent, seperti file kredensial, sumber daya untuk komunikasi antar-proses (IPC), dan folder orkestrasi. Tidak ada dalam direktori instalasi yang harus dimodifikasi, dipindahkan, atau dihapus. Jika tidak, SSM Agent mungkin berhenti berfungsi dengan baik.

SSM Agentpembaruan bergulir oleh Wilayah AWS

Setelah SSM Agent pembaruan tersedia di GitHub repositori, dapat memakan waktu hingga dua minggu hingga versi yang diperbarui diluncurkan ke semua Wilayah AWS pada waktu yang berbeda. Untuk alasan ini, Anda mungkin menerima kesalahan “Tidak didukung pada platform saat ini” atau “memperbarui amazon-ssm-agent ke versi yang lebih lama, aktifkan izinkan penurunan versi untuk melanjutkan” saat mencoba menerapkan versi baru SSM Agent di Wilayah.

Untuk menentukan versi yang SSM Agent tersedia untuk Anda, Anda dapat menjalankan `curl` perintah.

Untuk melihat versi agen yang tersedia di bucket unduhan global, jalankan perintah berikut.

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

Untuk melihat versi agen yang tersedia di Wilayah tertentu, jalankan perintah berikut, ganti *`wilayah`* dengan Wilayah tempat Anda bekerja, seperti `us-east-2` untuk Wilayah US East (Ohio).

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

Anda juga dapat membuka file `VERSION` secara langsung di peramban Anda tanpa perintah `curl`.

Menginstal SSM Agent pada VM dan instans lokal

Untuk informasi tentang menginstal SSM Agent pada mesin non-EC2 untuk lingkungan hybrid dan multicloud, lihat Install SSM Agent for a hybrid environment (Linux) dan Install SSM Agent for a hybrid environment (Windows).

Memvalidasi mesin yang diaktifkan hibrida menggunakan sidik jari perangkat keras

Ketika mesin non-EC2 di lingkungan hybrid dan multicloud, SSM Agent mengumpulkan sejumlah atribut sistem (disebut sebagai hash perangkat keras) dan menggunakan atribut ini untuk menghitung sidik jari. Sidik jari adalah string buram yang diteruskan agen ke API Systems Manager tertentu. Sidik jari unik ini mengaitkan penelepon dengan node terkelola yang diaktifkan hibrida tertentu. Agen menyimpan sidik jari dan hash perangkat keras pada disk lokal di lokasi yang disebut Vault.

Agen menghitung hash perangkat keras dan sidik jari saat mesin terdaftar untuk digunakan dengan Systems Manager. Kemudian, sidik jari diteruskan kembali ke layanan Systems Manager ketika agen mengirimkan perintah `RegisterManagedInstance`.

Kemudian, ketika mengirim perintah `RequestManagedInstanceRoleToken`, agen memeriksa sidik jari dan hash perangkat keras di Vault untuk memastikan bahwa atribut mesin saat ini cocok dengan hash perangkat keras yang disimpan. Jika atribut mesin saat ini cocok dengan hash perangkat keras yang disimpan di Vault, agen akan meneruskan sidik jari dari Vault ke `RegisterManagedInstance`, menghasilkan panggilan yang sukses.

Jika atribut mesin saat ini tidak cocok dengan hash perangkat keras yang disimpan, SSM Agent menghitung sidik jari baru, menyimpan hash perangkat keras dan sidik jari baru di Vault, dan meneruskan sidik jari baru ke. `RequestManagedInstanceRoleToken` Ini menyebabkan *`RequestManagedInstanceRoleToken`* gagal, dan agen tidak akan dapat memperoleh token peran untuk menghubungkan ke layanan Systems Manager.

Kegagalan ini dirancang dan digunakan sebagai langkah verifikasi untuk mencegah beberapa node terkelola berkomunikasi dengan layanan Systems Manager sebagai node terkelola yang sama.

Ketika membandingkan atribut mesin saat ini untuk hash perangkat keras yang disimpan di Vault, agen menggunakan logika berikut untuk menentukan apakah hash lama dan yang baru cocok:

- Jika SID (ID sistem/mesin) berbeda, maka tidak ada yang cocok.
- Jika tidak, jika alamat IP sama, maka cocokkan.
- Jika tidak, persentase atribut mesin yang cocok dihitung dan dibandingkan dengan batas kesamaan yang dikonfigurasi pengguna untuk menentukan apakah ada kecocokan.

Batas kesamaan disimpan di Vault, sebagai bagian dari hash perangkat keras.

Batas kesamaan dapat diatur setelah instans terdaftar menggunakan perintah seperti berikut.

Pada mesin Linux:

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

Pada Windows Server mesin yang menggunakan PowerShell:

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

> **Important**
>
> Jika salah satu komponen yang digunakan untuk menghitung sidik jari berubah, ini bisa menyebabkan agen hibernasi. Untuk membantu terhindar dari hibernasi ini, tetapkan batas kesamaan kepada nilai yang rendah, seperti **1**.

SSM Agent pada GitHub

Kode sumber SSM Agent tersedia GitHubsehingga Anda dapat menyesuaikan agen untuk memenuhi kebutuhan Anda. Kami mendorong Anda untuk mengirim permintaan tarik untuk perubahan yang ingin Anda sertakan. Namun, Amazon Web Services tidak menyediakan dukungan untuk menjalankan salinan yang dimodifikasi dari perangkat lunak ini.

Amazon Machine Images(AMIs) dengan SSM Agent prainstal

AWS Systems Manager Agen (SSM Agent) sudah diinstal sebelumnya pada beberapa Amazon Machine Images (AMIs) yang disediakan oleh AWS dan pihak ketiga tepercaya.

Misalnya, saat meluncurkan instans Amazon Elastic Compute Cloud (Amazon EC2) yang dibuat dari AMI instans dengan salah satu sistem operasi berikut, kemungkinan besar Anda akan menemukan bahwa instans sudah diinstal: SSM Agent

- AlmaLinux
- Amazon Linux 1 Base AMI tertanggal 2017.09 dan yang lebih baru
- Amazon Linux 2
- Amazon Linux 2 ECS-Optimized Base AMIs
- Amazon Linux 2023 (AL2023)
- Amazon Linux Amazon yang dioptimalkan EKS AMIs
- macOS10.14.x (Mojave), 10.15.x (Catalina), 11.x (Big Sur), 12.x (Monterey), 13.x (Ventura), dan 14.x (Sonoma)
- SUSE Linux Enterprise Server(SLES) 12 dan 15
- Ubuntu Server16.04, 18.04, 20.04, dan 22.04
- Windows Server 2008-2012 R2 AMIs yang dipublikasikan pada November 2016 atau yang lebih baru
- Windows Server2016, 2019, dan 2022

> **Note**
>
> SSM Agentmungkin sudah diinstal sebelumnya pada AWS dikelola AMIs yang tidak ada dalam daftar ini. Ini biasanya menunjukkan bahwa sistem operasi (OS) tidak sepenuhnya didukung oleh semua kemampuan Systems Manager.
> SSM Agentmungkin juga sudah diinstal sebelumnya pada AMIs ditemukan di AWS Marketplace atau di AMIs repositori Komunitas, tetapi AWS tidak mendukung ini. AMIs

Verifikasi status SSM Agent

Bergantung pada kapan diinisialisasi, instance yang dibuat dari daftar sebelumnya mungkin belum diinstal sebelumnya. AMI SSM Agent Mungkin juga sebuah instance memiliki agen yang sudah diinstal sebelumnya, tetapi agen tidak berjalan. Oleh karena itu, kami menyarankan Anda memeriksa status SSM Agent sebelum Anda mencoba menggunakan Systems Manager pada instans untuk pertama kalinya.

Gunakan prosedur berikut untuk memverifikasi bahwa SSM Agent diinstal dan berjalan pada sebuah instance. Jika Anda menemukan bahwa agen tidak diinstal, Anda dapat menginstalnya secara manual di Linux, macOS, dan Windows Serverinstance.

Untuk memverifikasi instalasi SSM Agent pada sebuah instance

1. Setelah meluncurkan instance baru, tunggu beberapa menit untuk menginisialisasi.
2. Connect ke instans menggunakan metode pilihan Anda. Misalnya, Anda dapat menggunakan SSH untuk terhubung ke instance Linux atau menggunakan Remote Desktop untuk terhubung ke Windows Server instance.
3. Periksa status SSM Agent dengan menjalankan perintah untuk jenis sistem operasi instans Anda.

Sistem operasi	Perintah
Amazon Linux 1	`sudo status amazon-ssm-agent`
Amazon Linux 2 dan Amazon Linux 2023	`sudo systemctl status amazon-ssm-agent`
macOS	Tidak ada perintah untuk memeriksa SSM Agent statusmacOS. Anda dapat memeriksa status dengan mencari dan mengevaluasi file log agen. `/var/log/amazon/ssm/amazon-ssm-agent.log`
SUSE Linux Enterprise Server	`sudo systemctl status amazon-ssm-agent`
Ubuntu Server(32-bit)	`sudo status amazon-ssm-agent`
Ubuntu Server(64-bit - Deb)	`sudo systemctl status amazon-ssm-agent`
Ubuntu Server(64-bit - Jepret)	`sudo systemctl status snap.amazon-ssm-agent.amazon-ssm-agent.service`

Sistem operasi	Perintah
Windows Server	`Get-Service AmazonSSMAgent`

> **Tip**
>
> Untuk melihat perintah untuk memeriksa SSM Agent status pada semua jenis sistem operasi yang didukung oleh Systems Manager, lihatMemeriksa SSM Agent status dan memulai agen.

4. Evaluasi output perintah untuk mempelajari statusSSM Agent.

   Status: Diinstal dan berjalan

   Dalam kebanyakan kasus, output perintah menunjukkan bahwa agen diinstal dan berjalan.

   Contoh berikut menunjukkan bahwa SSM Agent diinstal dan berjalan pada instance Amazon Linux 2.

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

   Contoh berikut menunjukkan bahwa SSM Agent diinstal dan berjalan pada sebuah Windows Server instance.

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

   Status: Diinstal tetapi tidak berjalan

   Dalam beberapa kasus, output perintah menunjukkan bahwa agen diinstal tetapi tidak berjalan.

   Contoh berikut menunjukkan bahwa SSM Agent diinstal tetapi tidak berjalan pada instance Amazon Linux 2.

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

Contoh berikut menunjukkan bahwa SSM Agent diinstal tetapi tidak berjalan pada sebuah Windows Server instance.

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

Jika agen diinstal tetapi tidak berjalan, Anda dapat mengaktifkannya secara manual menggunakan perintah untuk jenis sistem operasi instans Anda.

Sistem operasi	Perintah
Amazon Linux 1	`sudo start amazon-ssm-agent`
Amazon Linux 2 dan 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`

Sistem operasi	Perintah
SUSE Linux Enterprise Server	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`
Ubuntu Server(32-bit)	`sudo start amazon-ssm-agent`
Ubuntu Server(64-bit - Deb)	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`
Ubuntu Server(64-bit - Jepret)	`sudo snap start amazon-ssm-agent`
Windows Server	Jalankan perintah berikut di PowerShell. `Start-Service AmazonSSMAgent`

Status: Tidak diinstal

Dalam beberapa kasus, output perintah menunjukkan bahwa agen tidak diinstal.

Contoh berikut menunjukkan bahwa SSM Agent tidak diinstal pada instance Amazon Linux 2.

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

Contoh berikut menunjukkan bahwa SSM Agent tidak diinstal pada sebuah Windows Server instance.

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

Jika agen tidak diinstal, Anda dapat menginstalnya secara manual menggunakan prosedur untuk jenis sistem operasi Anda:

- Menginstal secara manual SSM Agent pada instans EC2 untuk Linux
- Menginstal secara manual SSM Agent pada instans EC2 untuk macOS
- Menginstal dan menghapus instalasi secara manual SSM Agent pada instans EC2 untuk Windows Server

Bekerja denganSSM Agent instans EC2 untuk Linux

AWS Systems ManagerAgent (SSM Agent) memproses permintaan Systems Manager dan mengkonfigurasi mesin Anda seperti yang ditentukan dalam permintaan. Gunakan prosedur dalam topik berikut untuk menginstal, mengkonfigurasi, menghapus instalanSSM Agent pada sistem operasi Linux.

Topik

- Memverifikasi tanda tangan SSM Agent
- Menginstal secara manual SSM Agent pada instans EC2 untuk Linux
- Mengkonfigurasi SSM Agent untuk menggunakan proxy (Linux)
- Menghapus instalasi SSM Agent dari instance Linux

Memverifikasi tanda tangan SSM Agent

Paket installer AWS Systems Manager Agent (SSM Agent) deb dan rpm untuk instance Linux ditandatangani secara kriptografis. Anda dapat menggunakan kunci publik untuk memverifikasi bahwa paket agen asli dan tidak dimodifikasi. Jika ada kerusakan atau perubahan pada file, verifikasi gagal. Anda dapat memverifikasi tanda tangan dari paket penginstal menggunakan RPM atau GPG. Informasi berikut adalah untuk SSM Agent versi 3.1.1141.0 atau yang lebih baru.

> **Important**
>
> Kunci publik yang ditampilkan nanti dalam topik ini berakhir pada 2025-02-17 (17 Februari 2025). Systems Manager akan menerbitkan kunci publik baru dalam topik ini sebelum yang lama kedaluwarsa. Kami mendorong Anda untuk berlangganan RSS feed untuk topik ini untuk mendapatkan pemberitahuan ketika kunci baru tersedia.

Untuk menemukan file standar yang benar untuk arsitektur instans dan sistem operasi instans Anda, lihat tabel berikut.

wilayah mewakili pengenal untuk Wilayah AWS didukung oleh AWS Systems Manager, seperti us-east-2 untuk Wilayah Timur AS (Ohio). Untuk daftar nilai *wilayah* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

Arsitektur	Sistem operasi	URL file standar	Nama file unduhan agen
x86_64	AlmaLinux, Amazon Linux 1, Amazon Linux 2, Amazon Linux 2023, CentOS, Aliran CentOS,,,, 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-ssm-*region*/latest/debian_amd64/amazon-ssm-agent.deb.sig	amazon-ssm-agent.deb

Arsitektur	Sistem operasi	URL file standar	Nama file unduhan agen
		`https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/debian_amd64/amazon-ssm-agent.deb.sig`	
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`

Arsitektur	Sistem operasi	URL file standar	Nama file unduhan agen
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`

Arsitektur	Sistem operasi	URL file standar	Nama file unduhan agen
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`

GPG

Untuk memverifikasi SSM Agent paket pada server Linux

1. Salin kunci publik berikut, dan simpan ke file bernama `amazon-ssm-agent.gpg`.

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

mQENBGTtIoIBCAD2M1aoGIE0FXynAHM/jtuvdAVVaX3Q4ZejTqrX+Jq8ElAMhxyO
GzHu2CDtCYxtVxXK3unptLVt2kGgJwNbhYC393jDeZx5dCda4Nk2YXX1UK3P461i
axuuXRzMYvfM4RZn+7bJTu635tA07q9Xm6MGD4TCTvsjBfViOxbrxOg5ozWbJdSw
fSR8MwUrRfmFpAefRlYfCEuZ8FHywa9U6jLeWt2O/kqrZliJOAGjGzXtB7EZkqKb
faCCxikjjvhF1awdEqSK4DQorC/OvQc4I5kP5y2CJbtXvXO73QH2yE75JMDIIx9x
rOsIRUoSfK3UrWaOVuAnEEn5ueKzZNqGG1J1ABEBAAG0J1NTTSBBZ2VudCA8c3Nt
LWFnZW50LXNpZ25lckBhbWF6b24uY29tPokBPwQTAQIAKQUCZO0iggIbLwUJAsaY
```

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

2. Impor kunci publik ke dalam keyring Anda, dan perhatikan nilai kunci yang dikembalikan.

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

3. Verifikasi sidik jari. Pastikan untuk mengganti *key-value* dengan nilai dari langkah sebelumnya. Kami menyarankan Anda menggunakan GPG untuk memverifikasi sidik jari bahkan jika Anda menggunakan RPM untuk memverifikasi paket penginstal.

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

Perintah ini mengembalikan output yang serupa dengan yang berikut.

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

Sidik jari harus sesuai dengan yang berikut.

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

Jika sidik jari tidak cocok, jangan instal agen. Kontak AWS Support.

4. Unduh file tanda tangan sesuai dengan arsitektur dan sistem operasi instans Anda jika Anda belum melakukannya.
5. Verifikasi tanda tangan paket penginstal. Pastikan untuk mengganti *nama file tanda tangan* dan *agent-download-filename*dengan nilai yang Anda tentukan saat mengunduh file tanda tangan dan agen, seperti yang tercantum dalam tabel sebelumnya dalam topik ini.

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

   Misalnya, untuk x86_64 arsitektur di Amazon Linux 2:

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

   Perintah ini mengembalikan output yang serupa dengan yang berikut ini.

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

   Jika output berisi frasa `BAD signature`, periksa apakah Anda melakukan prosedur dengan benar. Jika Anda terus mendapatkan tanggapan ini, hubungi AWS Support dan jangan instal agen. Pesan peringatan tentang kepercayaan tidak berarti tanda tangan tidak valid, hanya Anda yang belum memverifikasi kunci publik. Kunci hanya dapat dipercaya jika Anda atau seseorang yang Anda percaya telah menandatanganinya. Jika output menyertakan frasa`Can't check signature: No public key`, verifikasi Anda mengunduh SSM Agent versi 3.1.1141.0 atau yang lebih baru.

RPM

Untuk memverifikasi SSM Agent paket pada server Linux

1. Salin kunci publik berikut, dan simpan ke file bernama `amazon-ssm-agent.gpg`.

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

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

2. Impor kunci publik ke dalam keyring Anda, dan perhatikan nilai kunci yang dikembalikan.

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

3. Verifikasi sidik jari. Pastikan untuk mengganti *key-value* dengan nilai dari langkah sebelumnya. Kami menyarankan Anda menggunakan GPG untuk memverifikasi sidik jari bahkan jika Anda menggunakan RPM untuk memverifikasi paket penginstal.

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

Perintah ini mengembalikan output yang serupa dengan yang berikut.

```
pub   2048R/97DD04ED 2023-08-28 [expires: 2025-02-17]
      Key fingerprint = DE92 C7DA 3E56 E923 31D6  2A36 BC1F 495C 97DD 04ED
```

```
uid                  SSM Agent <ssm-agent-signer@amazon.com>
```

Sidik jari harus sesuai dengan yang berikut.

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

Jika sidik jari tidak cocok, jangan instal agen. Kontak AWS Support.

4. Verifikasi tanda tangan paket penginstal. Pastikan untuk mengganti *nama file tanda tangan* dan *agent-download-filename*dengan nilai yang Anda tentukan saat mengunduh file tanda tangan dan agen, seperti yang tercantum dalam tabel sebelumnya dalam topik ini.

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

Misalnya, untuk x86_64 arsitektur di Amazon Linux 2:

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

Perintah ini mengembalikan output yang serupa dengan yang berikut ini.

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

Jika pgp hilang dari output dan Anda telah mengimpor kunci publik, maka agen tidak ditandatangani. Jika output berisi frasa`NOT OK (MISSING KEYS: (MD5) key-id`), periksa apakah Anda melakukan prosedur dengan benar dan verifikasi bahwa Anda mengunduh SSM Agent versi 3.1.1141.0 atau yang lebih baru. Jika Anda terus mendapatkan tanggapan ini, hubungi AWS Support dan jangan instal agen.

Menginstal secara manual SSM Agent pada instans EC2 untuk Linux

Sebelum Anda menginstal AWS Systems Manager Agent (SSM Agent) secara manual di sistem operasi Amazon Elastic Compute Cloud (Amazon EC2) Linux, tinjau informasi berikut.

SSM AgentURL file instalasi

Anda dapat mengakses file instalasi SSM Agent yang disimpan dalam iklan apa pun Wilayah AWS. Kami juga menyediakan file instalasi dalam bucket Amazon Simple Storage Service (Amazon S3) yang tersedia secara global yang dapat Anda gunakan sebagai alternatif atau sumber cadangan file.

Jika Anda menginstal agen secara manual pada satu atau dua instance, Anda dapat menggunakan perintah dalam prosedur instalasi Cepat yang kami sediakan untuk menghemat waktu. Perintah yang disediakan dalam prosedur ini juga dapat diteruskan ke instans Amazon EC2 sebagai skrip melalui data pengguna.

Jika Anda membuat skrip atau template untuk digunakan untuk menginstal agen pada beberapa contoh, kami sarankan Anda menggunakan file instalasi di atau di dekat Wilayah AWS tempat Anda berada secara geografis. Untuk instalasi massal, ini dapat meningkatkan kecepatan unduhan Anda dan mengurangi latensi. Dalam kasus ini, kami sarankan menggunakan prosedur perintah Buat instalasi kustom dalam topik instalasi.

Amazon Machine Imagesdengan agen yang sudah diinstal sebelumnya

SSM Agentsudah diinstal sebelumnya pada beberapa Amazon Machine Images (AMIs) yang disediakan oleh AWS. Untuk informasi, lihat Amazon Machine Images(AMIs) dengan SSM Agent prainstal.

Instalasi pada jenis mesin lainnya

Jika Anda perlu menginstal agen di server lokal atau mesin virtual (VM) agar dapat digunakan dengan Systems Manager, lihat Menginstal SSM Agent untuk lingkungan hybrid (Linux). Untuk informasi tentang menginstal agen pada perangkat edge, lihatAWS Systems ManagerMenyiapkan perangkat edge.

Menjaga agen tetap up to date

Versi terbaru dirilis setiap kali kemampuan baru ditambahkan ke Systems Manager atau pembaruan dibuat untuk kemampuan yang ada. SSM Agent Gagal menggunakan agen versi terbaru dapat mencegah node terkelola Anda menggunakan berbagai kemampuan dan fitur Systems Manager. Untuk alasan itu, kami menyarankan Anda mengotomatiskan proses menjaga agar tetap SSM Agent up to date pada mesin Anda. Untuk informasi, lihat Mengotomatiskan pembaruan ke SSM Agent. Berlangganan halaman Catatan SSM Agent Rilis GitHub untuk mendapatkan pemberitahuan tentang SSM Agent pembaruan.

Pilih sistem operasi Anda

Untuk melihat prosedur untuk menginstal secara manual SSM Agent pada sistem operasi yang ditentukan, pilih tautan dari daftar berikut:

> **Note**
>
> Untuk daftar versi yang didukung dari masing-masing sistem operasi berikut, lihat Sistem operasi yang didukung untuk Systems Manager.

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

Instal secara manual SSM Agent pada AlmaLinux instance

Gunakan informasi di bagian ini untuk membantu Anda menginstal atau menginstal ulang secara manual SSM Agent pada sebuah AlmaLinux instans.

Sebelum Anda memulai

Sebelum Anda menginstal SSM Agent pada sebuah AlmaLinux instance, perhatikan hal berikut:

- Pastikan bahwa Python 3 diinstal pada instans Anda AlmaLinux . Ini diperlukan agar dapat bekerja SSM Agent dengan baik.
- Untuk informasi penting yang berlaku untuk instalasi SSM Agent pada semua sistem operasi berbasis Linux, lihat. Menginstal secara manual SSM Agent pada instans EC2 untuk Linux

Topik

- Perintah instalasi cepat untuk SSM Agent on AlmaLinux
- Buat perintah instalasi agen kustom untuk AlmaLinux di Wilayah Anda

Perintah instalasi cepat untuk SSM Agent on AlmaLinux

Gunakan langkah-langkah berikut untuk menginstal secara manual SSM Agent pada satu instance. Prosedur ini menggunakan file instalasi yang tersedia secara global.

Sebelum Anda memulai

Sebelum Anda menginstal SSM Agent pada sebuah AlmaLinux instance, perhatikan hal berikut:

- Pastikan bahwa Python 3 diinstal pada instans Anda AlmaLinux . Ini diperlukan agar dapat bekerja SSM Agent dengan baik.

Untuk SSM Agent menginstal AlmaLinux

1. Connect ke AlmaLinux instans Anda menggunakan metode pilihan Anda, seperti SSH.
2. Salin perintah untuk arsitektur instans Anda dan jalankan pada instance.

   > **Note**
   >
   > Meskipun URL dalam perintah berikut menyertakan `ec2-downloads-windows` direktori, ini adalah file instalasi global yang benar untuk AlmaLinux.

   Instans x86_64

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

   Instans ARM64

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

3. (Disarankan) Jalankan perintah berikut untuk memverifikasi bahwa agen sedang berjalan.

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

   Dalam kebanyakan kasus, perintah melaporkan bahwa agen sedang berjalan, seperti yang ditunjukkan pada contoh berikut.

```
# 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--
```

Dalam kasus yang jarang terjadi, perintah melaporkan bahwa agen diinstal tetapi tidak berjalan, seperti yang ditunjukkan pada contoh berikut.

```
# 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--
```

Untuk mengaktifkan agen dalam kasus ini, jalankan perintah berikut.

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

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

Buat perintah instalasi agen kustom untuk AlmaLinux di Wilayah Anda

Saat Anda menginstal SSM Agent pada beberapa instance menggunakan skrip atau template, sebaiknya gunakan file instalasi yang disimpan di tempat Wilayah AWS Anda bekerja.

Untuk perintah berikut, kami memberikan contoh yang menggunakan bucket S3 yang dapat diakses publik di Wilayah Timur AS (Ohio) (). `us-east-2`

> **Tip**
>
> Anda juga dapat mengganti URL global dalam prosedur Perintah instalasi cepat untuk SSM Agent on AlmaLinux sebelumnya dalam topik ini dengan URL Regional kustom yang Anda buat.

Dalam perintah berikut, ganti *wilayah* dengan informasi Anda sendiri. Untuk daftar nilai *wilayah* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

x86_64

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

Lihat contoh berikut ini.

```
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
```

Lihat contoh berikut ini.

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

Menginstal secara manual SSM Agent di instans Amazon Linux 2 dan Amazon Linux 2023

> **Important**
>
> Topik ini menyediakan perintah untuk bekerja dengan instans SSM Agent Amazon Linux 2 dan Amazon Linux 2023. Beberapa perintah ini tidak didukung pada instans Amazon Linux 1.

Sebelum melanjutkan, pastikan Anda melihat topik yang benar untuk tipe instans Anda. Untuk perintah yang dapat dijalankan di instans Amazon Linux 1, lihatMenginstal secara manual SSM Agent di instans Amazon Linux 1.

Dalam kebanyakan kasus, Amazon Machine Images (AMIs) untuk Amazon Linux 2 dan Amazon Linux 2023 yang disediakan oleh AWS datang dengan AWS Systems Manager Agent (SSM Agent) yang sudah diinstal sebelumnya secara default. Untuk informasi selengkapnya, lihat Amazon Machine Images(AMIs) dengan SSM Agent prainstal.

Jika SSM Agent tidak diinstal sebelumnya pada instans Amazon Linux 2 atau Amazon Linux 2023 baru, atau jika Anda perlu menginstal ulang agen secara manual, gunakan informasi di halaman ini untuk membantu Anda.

Sebelum Anda memulai

Sebelum Anda menginstal SSM Agent pada instans Amazon Linux 2 atau Amazon Linux 2023, perhatikan hal berikut:

- Untuk informasi penting yang berlaku untuk instalasi SSM Agent pada semua sistem operasi berbasis Linux, lihat. Menginstal secara manual SSM Agent pada instans EC2 untuk Linux
- Jika Anda menggunakan `yum` perintah untuk memperbarui SSM Agent pada node terkelola setelah agen diinstal atau diperbarui menggunakan dokumen SSM`AWS-UpdateSSMAgent`, Anda mungkin melihat pesan berikut: “Peringatan: RPMDB diubah di luar yum.” Pesan ini diharapkan dan dapat diabaikan dengan aman.

Topik

- Perintah instalasi cepat untuk SSM Agent di Amazon Linux 2 atau Amazon Linux 2023
- Buat perintah instalasi agen khusus untuk Amazon Linux 2 atau Amazon Linux 2023 di Wilayah Anda

Perintah instalasi cepat untuk SSM Agent di Amazon Linux 2 atau Amazon Linux 2023

Gunakan langkah-langkah berikut untuk menginstal secara manual SSM Agent pada satu instance. Prosedur ini menggunakan file instalasi yang tersedia secara global.

Untuk menginstal SSM Agent di Amazon Linux 2 atau Amazon Linux 2023 menggunakan perintah salin dan tempel cepat

1. Connect ke instans Amazon Linux 2 atau Amazon Linux 2023 menggunakan metode pilihan Anda, seperti SSH.
2. Salin perintah untuk arsitektur instans Anda dan jalankan pada instance.

   > **Note**
   >
   > Meskipun URL dalam perintah berikut menyertakan `ec2-downloads-windows` direktori, ini adalah file instalasi global yang benar untuk Amazon Linux 2 dan 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. (Disarankan) Jalankan perintah berikut untuk memverifikasi bahwa agen sedang berjalan.

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

   Dalam kebanyakan kasus, perintah melaporkan bahwa agen sedang berjalan, seperti yang ditunjukkan pada contoh berikut.

   ```
   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--
   ```

   Dalam kasus yang jarang terjadi, perintah melaporkan bahwa agen diinstal tetapi tidak berjalan, seperti yang ditunjukkan pada contoh berikut.

```
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--
```

Untuk mengaktifkan agen dalam kasus ini, jalankan perintah berikut.

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

Buat perintah instalasi agen khusus untuk Amazon Linux 2 atau Amazon Linux 2023 di Wilayah Anda

Saat Anda menginstal SSM Agent pada beberapa instance menggunakan skrip atau template, sebaiknya gunakan file instalasi yang disimpan di tempat Wilayah AWS Anda bekerja.

Untuk perintah berikut, kami memberikan contoh yang menggunakan bucket S3 yang dapat diakses publik di Wilayah Timur AS (Ohio) (). `us-east-2`

> **Tip**
>
> Anda juga dapat mengganti URL global dalam prosedur Perintah instalasi cepat untuk SSM Agent di Amazon Linux 1 sebelumnya dalam topik ini dengan URL Regional kustom yang Anda buat.

Dalam perintah berikut, ganti *`wilayah`* dengan informasi Anda sendiri. Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

x86_64

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

Lihat contoh berikut ini.

```
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
```

Lihat contoh berikut ini.

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

Menginstal secara manual SSM Agent di instans Amazon Linux 1

> **Important**
>
> Amazon Linux 1 mencapai akhir dukungan standarnya pada 31 Desember 2020, dan mencapai akhir masa pakai pada 31 Desember 2023, seperti yang diumumkan dalam Pembaruan di Amazon Linux AMI end-of-life di Blog AWS Berita. AWS tidak lagi menyediakan Amazon Machine Images (AMIs) untuk sistem operasi ini. AWS Systems Manager Namun, terus memberikan dukungan untuk instans Amazon Linux 1 yang ada. Topik ini menyediakan perintah untuk bekerja dengan instans SSM Agent Amazon Linux 1. Beberapa perintah ini tidak didukung pada instans Amazon Linux 2 dan Amazon Linux 2023. Sebelum melanjutkan, verifikasi bahwa Anda melihat topik yang benar untuk jenis instans Anda. Untuk perintah yang dapat dijalankan di instans Amazon Linux 2 atau Amazon Linux 2023, lihat. Menginstal secara manual SSM Agent di instans Amazon Linux 2 dan Amazon Linux 2023

Dalam kebanyakan kasus, Amazon Machine Images (AMIs) untuk Amazon Linux 1 yang disediakan oleh AWS datang dengan AWS Systems Manager Agent (SSM Agent) yang telah diinstal sebelumnya secara default. Untuk informasi selengkapnya, lihat Amazon Machine Images(AMIs) dengan SSM Agent prainstal.

Jika Anda perlu menginstal ulang agen secara manual di Amazon Linux 1, gunakan informasi di halaman ini untuk membantu Anda.

Sebelum Anda memulai

Sebelum Anda menginstal SSM Agent pada instans Amazon Linux 1, perhatikan hal berikut:

- Untuk informasi penting yang berlaku untuk instalasi SSM Agent pada semua sistem operasi berbasis Linux, lihat. Menginstal secara manual SSM Agent pada instans EC2 untuk Linux
- Node terkelola yang dibuat dari Amazon Linux 1 AMI yang menggunakan proxy harus menjalankan versi `Python requests` modul saat ini untuk mendukung Patch Manager operasi. Untuk informasi selengkapnya, lihat Memutakhirkan modul permintaan Python di Amazon Linux 1 instance yang menggunakan server proxy.
- Jika Anda menggunakan `yum` perintah untuk memperbarui SSM Agent pada node terkelola setelah agen diinstal atau diperbarui menggunakan dokumen SSM`AWS-UpdateSSMAgent`, Anda mungkin melihat pesan berikut: “Peringatan: RPMDB diubah di luar yum.” Pesan ini diharapkan dan dapat diabaikan dengan aman.

Topik

Perintah instalasi cepat untuk SSM Agent di Amazon Linux 1

Gunakan langkah-langkah berikut untuk menginstal secara manual SSM Agent pada satu instance. Prosedur ini menggunakan file instalasi yang tersedia secara global.

Untuk menginstal SSM Agent di Amazon Linux 1 menggunakan perintah salin dan tempel cepat

1. Connect ke instans Amazon Linux 1 Anda menggunakan metode pilihan Anda, seperti SSH.
2. Salin perintah untuk arsitektur instans Anda dan jalankan pada instance.

> **Note**
>
> Meskipun URL dalam perintah berikut menyertakan `ec2-downloads-windows` direktori, ini adalah file instalasi global yang benar untuk 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. (Disarankan) Jalankan perintah untuk arsitektur instans Anda untuk memverifikasi bahwa agen sedang berjalan.

   x86_64 dan x86

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

   ARM64

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

   Dalam kebanyakan kasus, perintah melaporkan bahwa agen sedang berjalan, seperti yang ditunjukkan dalam contoh berikut.

   x86_64 dan 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--
   ```

Dalam kasus yang jarang terjadi, perintah melaporkan bahwa agen diinstal tetapi tidak berjalan, seperti yang ditunjukkan pada contoh berikut.

x86_64 dan 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--
```

Untuk mengaktifkan agen dalam kasus ini, jalankan perintah untuk arsitektur instans Anda.

x86_64 dan x86

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

ARM64

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

Buat perintah instalasi agen khusus untuk Amazon Linux 1 di Wilayah Anda

Saat Anda menginstal SSM Agent pada beberapa instance menggunakan skrip atau template, sebaiknya gunakan file instalasi yang disimpan di tempat Wilayah AWS Anda bekerja.

Untuk perintah berikut, kami memberikan contoh yang menggunakan bucket S3 yang dapat diakses publik di Wilayah Timur AS (Ohio) (). `us-east-2`

> **Tip**
>
> Anda juga dapat mengganti URL global dalam prosedur Perintah instalasi cepat untuk SSM Agent di Amazon Linux 1 sebelumnya dalam topik ini dengan URL Regional kustom yang Anda buat.

Dalam perintah berikut, ganti *wilayah* dengan informasi Anda sendiri. Untuk daftar nilai *wilayah* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

x86_64

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

Lihat contoh berikut ini.

```
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
```

Lihat contoh berikut ini.

```
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
```

Lihat contoh berikut ini.

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

Menginstal secara manualSSM Agent pada instans CentOS

Amazon Machine Images(AMIs) untuk CentOS yang disediakan olehAWS tidak datang denganAWS Systems Manager Agen (SSM Agent) terinstal secara default. Untuk daftar yangAWS dikelolaAMIs di mana agen mungkin diinstal sebelumnya, lihatAmazon Machine Images(AMIs) dengan SSM Agent prainstal.

Gunakan informasi di bagian ini untuk membantu Anda menginstal atau menginstal ulang secara manualSSM Agent pada instans CentOS.

Sebelum Anda memulai

Sebelum Anda menginstalSSM Agent pada instans CentOS, perhatikan hal berikut ini:

- Untuk informasi penting yang berlaku untuk instalasiSSM Agent pada semua sistem operasi berbasis Linux, lihatMenginstal secara manual SSM Agent pada instans EC2 untuk Linux.
- Jika Anda menggunakan`yum` perintah untuk memperbaruiSSM Agent pada node terkelola setelah agen telah diinstal atau diperbarui menggunakan dokumen SSM`AWS-UpdateSSMAgent`, Anda mungkin melihat pesan berikut: “Peringatan: RPMDB diubah di luar yum.” Pesan ini diharapkan dan dapat diabaikan dengan aman.

Topik

- InstalSSM Agent di CentOS 8.x
- InstalSSM Agent di CentOS 7.x
- InstalSSM Agent di CentOS

InstalSSM Agent di CentOS 8.x

Amazon Machine Images(AMIs) untuk CentOS 8 yang disediakan olehAWS tidak datang denganAWS Systems Manager Agen (SSM Agent) terinstal secara default. Gunakan informasi di halaman ini untuk membantu Anda menginstal atau menginstal ulang agen pada instans CentOS 8.

Sebelum Anda memulai

Sebelum Anda menginstalSSM Agent di instans CentOS 8, perhatikan hal berikut ini:

- Pastikan bahwa Python 2 atau Python 3 terinstal pada instans CentOS 8 Anda. Hal ini diperlukan agar dapatSSM Agent bekerja dengan baik.

Topik

Perintah instalasi cepat untukSSM Agent di CentOS 8

Gunakan langkah-langkah berikut untuk menginstal secara manualSSM Agent pada satu instance. Prosedur ini menggunakan file instalasi yang tersedia secara global.

Untuk menginstalSSM Agent di CentOS 8.x

1. Connect ke instans CentOS 8 Anda menggunakan metode pilihan Anda, seperti SSH.
2. Salin perintah untuk arsitektur instance Anda dan jalankan pada instance.

   > **Note**
   >
   > Meskipun URL dalam perintah berikut menyertakan`ec2-downloads-windows` direktori, ini adalah file instalasi global yang benar untuk CentOS 8.

   x86_64contoh

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

   Instans ARM64

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

3. (Direkomendasikan) Jalankan perintah berikut untuk memverifikasi bahwa agen menjalankan.

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

Dalam kebanyakan kasus, perintah akan menampilkan bahwa agen menjalankan, seperti yang ditunjukkan dalam contoh berikut.

```
# 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--
```

Dalam kasus yang jarang terjadi, perintah akan menampilkan bahwa agen diinstal tetapi tidak berjalan, seperti yang ditunjukkan dalam contoh berikut.

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

Untuk mengaktifkan agen dalam kasus ini, jalankan perintah berikut.

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

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

Buat perintah instalasi agen kustom untuk CentOS 8 di Wilayah Anda

Ketika Anda menginstalSSM Agent pada beberapa contoh menggunakan skrip atau template, kami sarankan menggunakan file instalasi yang disimpan di tempatWilayah AWS Anda bekerja.

Untuk perintah berikut, kami memberikan contoh yang menggunakan bucket S3 yang dapat diakses publik di Wilayah AS Timur (Ohio) (`us-east-2`).

> **Tip**
>
> Anda juga dapat mengganti URL global dalam prosedurPerintah instalasi cepat untukSSM Agent di CentOS 8 sebelumnya dalam topik ini dengan URL Regional kustom yang Anda buat.

Dalam perintah berikut, ganti *wilayah* dengan informasi Anda sendiri. Untuk daftar nilai *wilayah* yang didukung, lihat kolom Wilayah di endpoint layanan Systems Manager di Referensi Umum Amazon Web Services.

x86_64

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

Lihat contoh berikut.

```
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
```

Lihat contoh berikut.

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

InstalSSM Agent di CentOS 7.x

Amazon Machine Images(AMIs) untuk CentOS 7 yang disediakan olehAWS tidak datang denganAWS Systems Manager Agen (SSM Agent) terinstal secara default. Gunakan informasi di halaman ini untuk membantu Anda menginstal atau menginstal ulang agen pada instans CentOS 7.

Topik

- Perintah instalasi cepat untukSSM Agent di CentOS 7
- Buat perintah instalasi agen kustom untuk CentOS 7 di Wilayah Anda

Perintah instalasi cepat untukSSM Agent di CentOS 7

Gunakan langkah-langkah berikut untuk menginstal secara manualSSM Agent pada satu instance. Prosedur ini menggunakan file instalasi yang tersedia secara global.

Untuk menginstalSSM Agent di CentOS 7.x

1. Connect ke instans CentOS 7 Anda menggunakan metode pilihan Anda, seperti SSH.
2. Salin perintah untuk arsitektur instance Anda dan jalankan pada instance.

> **Note**
>
> Meskipun URL dalam perintah berikut menyertakan`ec2-downloads-windows` direktori, ini adalah file instalasi global yang benar untuk CentOS 7.

Instans x86_64

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

Instans ARM64

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

3. (Disarankan) Jalankan perintah berikut untuk memastikan bahwa agen menjalankan.

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

Dalam kebanyakan kasus, perintah akan menampilkan bahwa agen menjalankan, seperti yang ditunjukkan dalam contoh berikut.

```
# 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--
```

Dalam kasus yang jarang terjadi, perintah akan menampilkan bahwa agen diinstal tetapi tidak berjalan, seperti yang ditunjukkan dalam contoh berikut.

```
# 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--
```

Untuk mengaktifkan agen dalam kasus ini, jalankan perintah berikut.

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

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

Buat perintah instalasi agen kustom untuk CentOS 7 di Wilayah Anda

Ketika Anda menginstalSSM Agent pada beberapa contoh menggunakan skrip atau template, kami sarankan menggunakan file instalasi yang disimpan di tempatWilayah AWS Anda bekerja.

Untuk perintah berikut, kami memberikan contoh yang menggunakan bucket S3 yang dapat diakses publik di Wilayah AS Timur (Ohio) (`us-east-2`).

> **Tip**
>
> Anda juga dapat mengganti URL global dalam prosedurPerintah instalasi cepat untukSSM Agent di CentOS 7 sebelumnya dalam topik ini dengan URL Regional kustom yang Anda buat.

Dalam perintah berikut, ganti *`wilayah`* dengan informasi Anda sendiri. Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Wilayah di endpoint layanan Systems Manager di bagian Referensi Umum Amazon Web Services.

x86_64

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

Lihat contoh berikut.

```
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
```

Lihat contoh berikut.

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

InstalSSM Agent di CentOS

Amazon Machine Images(AMIs) untuk CentOS 6 yang disediakan olehAWS tidak datang denganAWS Systems Manager Agen (SSM Agent) terinstal secara default. Gunakan informasi di halaman ini untuk membantu Anda menginstal atau menginstal ulang agen pada instans CentOS 6.

Topik

- Perintah instalasi cepat untukSSM Agent di CentOS 6
- Buat perintah instalasi agen kustom untuk CentOS 6 di Wilayah Anda

Perintah instalasi cepat untukSSM Agent di CentOS 6

Gunakan langkah-langkah berikut untuk menginstal secara manualSSM Agent pada satu instance. Prosedur ini menggunakan file instalasi yang tersedia secara global.

Untuk menginstalSSM Agent di CentOS

1. Connect ke instans CentOS 6 Anda menggunakan metode pilihan Anda, seperti SSH.
2. Salin perintah untuk arsitektur instance Anda dan jalankan pada instance.

> **Note**
>
> Meskipun URL dalam perintah berikut menyertakan`ec2-downloads-windows` direktori, ini adalah file instalasi global yang benar untuk CentOS 6.

Perintah berikut menentukan direktori versi,`3.0.1479.0` bukan`latest` direktori. Ini karenaSSM Agent versi 3.1 dan yang lebih baru tidak didukung untuk CentOS 6.

Instans x86_64

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

Instans x86

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

3. (Direkomendasikan) Jalankan perintah berikut untuk memverifikasi bahwa agen menjalankan

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

Dalam kebanyakan kasus, perintah akan menampilkan bahwa agen menjalankan, seperti yang ditunjukkan dalam contoh berikut.

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

Dalam kasus yang jarang terjadi, perintah akan menampilkan bahwa agen diinstal tetapi tidak berjalan, seperti yang ditunjukkan dalam contoh berikut.

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

Untuk mengaktifkan agen dalam kasus ini, jalankan perintah berikut.

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

Buat perintah instalasi agen kustom untuk CentOS 6 di Wilayah Anda

Ketika Anda menginstalSSM Agent pada beberapa contoh menggunakan skrip atau template, kami sarankan menggunakan file instalasi yang disimpan di tempatWilayah AWS Anda bekerja.

Untuk perintah berikut, kami memberikan contoh yang menggunakan bucket S3 yang dapat diakses publik di Wilayah AS Timur (Ohio) (`us-east-2`).

> **Tip**
>
> Anda juga dapat mengganti URL global dalam prosedurPerintah instalasi cepat untukSSM Agent di CentOS 6 sebelumnya dalam topik ini dengan URL Regional kustom yang Anda buat.

Dalam perintah berikut, ganti *`wilayah`* dengan informasi Anda sendiri Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Wilayah di endpoint layanan Systems Manager di bagian Referensi Umum Amazon Web Services.

> **Note**
>
> Perintah berikut menentukan direktori versi,`3.0.1390.0` bukan`latest` direktori. Ini karenaSSM Agent versi 3.1 dan yang lebih baru tidak didukung untuk 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
```

Lihat contoh berikut.

```
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
```

Lihat contoh berikut.

```
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
```

Instal secara manualSSM Agent padaCentOS Stream instans

Amazon Machine Images(AMIs) untukCentOS Stream yang disediakan olehAWS tidak datang denganAWS Systems Manager Agen (SSM Agent) terinstal secara default. Untuk daftar yangAWS dikelolaAMIs di mana agen mungkin sudah diinstal sebelumnya, lihatAmazon Machine Images(AMIs) dengan SSM Agent prainstal.

Gunakan informasi di bagian ini untuk membantu Anda menginstal atau menginstal ulang secara manualSSM Agent padaCentOS Stream instance.

Sebelum Anda memulai

Sebelum Anda menginstalSSM AgentCentOS Stream instans, perhatikan hal berikut:

- Untuk informasi penting yang berlaku untuk instalasiSSM Agent pada semua sistem operasi berbasis Linux, lihatMenginstal secara manual SSM Agent pada instans EC2 untuk Linux.

Topik

- Perintah instalasi cepat untukSSM Agent aktifCentOS Stream
- Membuat perintah instalasi agen kustom untukCentOS Stream di Wilayah Anda

Perintah instalasi cepat untukSSM Agent aktifCentOS Stream

Gunakan langkah-langkah berikut untuk menginstal secara manualSSM Agent pada satu instance. Prosedur ini menggunakan file instalasi yang tersedia secara global.

Sebelum Anda memulai

Sebelum Anda menginstalSSM AgentCentOS Stream instans, perhatikan hal berikut:

- Pastikan bahwa Python 2 atau Python 3 terinstal pada instansCentOS Stream 8 Anda. Hal ini diperlukan agar dapatSSM Agent bekerja dengan baik.

UntukSSM Agent menginstalCentOS Stream

1. Connect keCentOS Stream instans Anda menggunakan metode pilihan Anda, seperti SSH.
2. Salin perintah untuk arsitektur instance Anda dan jalankan pada instance.

> **Note**
>
> Meskipun URL dalam perintah berikut menyertakan`ec2-downloads-windows` direktori, ini adalah file instalasi global yang benar untukCentOS Stream.

Instans x86_64

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

Instans ARM64

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

3. (Disarankan) Jalankan perintah berikut untuk memverifikasi bahwa agen menjalankan.

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

Dalam kebanyakan kasus, perintah akan menampilkan bahwa agen menjalankan, seperti yang ditunjukkan dalam contoh berikut.

```
# 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--
```

Dalam kasus yang jarang terjadi, perintah akan menampilkan bahwa agen terinstal tetapi tidak berjalan, seperti yang ditunjukkan dalam contoh berikut.

```
# 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--
```

Untuk mengaktifkan agen dalam kasus ini, jalankan perintah berikut.

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

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

Membuat perintah instalasi agen kustom untukCentOS Stream di Wilayah Anda

Ketika Anda menginstalSSM Agent pada beberapa contoh menggunakan skrip atau template, kami sarankan menggunakan file instalasi yang disimpan di tempatWilayah AWS Anda bekerja.

Untuk perintah berikut, kami memberikan contoh yang menggunakan bucket S3 yang dapat diakses publik di Wilayah AS Timur (Ohio) (`us-east-2`).

> **Tip**
>
> Anda juga dapat mengganti URL global dalam prosedurPerintah instalasi cepat untukSSM Agent aktifCentOS Stream sebelumnya dalam topik ini dengan URL Regional kustom yang Anda buat.

Dalam perintah berikut, ganti *`wilayah`* dengan informasi Anda sendiri. Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Wilayah di endpoint layanan Systems Manager di bagian Referensi Umum Amazon Web Services.

x86_64

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

Lihat contoh berikut.

```
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
```

Lihat contoh berikut.

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

Menginstal secara manualSSM Agent padaDebian Server instance

Amazon Machine Images(AMIs) untukDebian Server yang disediakan olehAWS tidak datang denganAWS Systems Manager Agen (SSM Agent) terinstal secara default. Untuk daftar yangAWS dikelolaAMIs di mana agen mungkin sudah diinstal sebelumnya, lihatAmazon Machine Images(AMIs) dengan SSM Agent prainstal.

Gunakan informasi di bagian ini untuk membantu Anda menginstal atau menginstal ulang secara manualSSM Agent padaDebian Server instans.

Sebelum Anda memulai

Sebelum Anda menginstalSSM AgentDebian Server instans, perhatikan hal berikut:

- Untuk informasi penting yang berlaku untuk instalasiSSM Agent pada semua sistem operasi berbasis Linux, lihatMenginstal secara manual SSM Agent pada instans EC2 untuk Linux.

Topik

- Perintah instalasi cepat untukSSM Agent aktifDebian Server
- Membuat perintah instalasi agen kustom untukDebian Server di Wilayah Anda

Perintah instalasi cepat untukSSM Agent aktifDebian Server

Gunakan langkah-langkah berikut untuk menginstal secara manualSSM Agent pada satu instance. Prosedur ini menggunakan file instalasi yang tersedia secara global.

UntukSSM Agent menginstalDebian Server

1. Connect keDebian Server instans Anda menggunakan metode pilihan Anda, seperti SSH.

2. Jalankan perintah berikut ini untuk membuat direktori sementara pada instans.

```
mkdir /tmp/ssm
```

3. Jalankan perintah berikut ini untuk mengubah ke direktori sementara.

```
cd /tmp/ssm
```

4. Salin perintah untuk arsitektur instance Anda dan jalankan pada instance.

> **Note**
>
> Meskipun URL dalam perintah berikut menyertakan`ec2-downloads-windows` direktori, ini adalah file instalasi global yang benar untukDebian Server.
> UntukDebian Server 8, hanyax86_64 arsitektur yang didukung.

Instans x86_64

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

Instans ARM64

```
wget https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/
debian_arm64/amazon-ssm-agent.deb
```

5. Jalankan perintah berikut.

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

6. (Disarankan) Jalankan perintah berikut ini untuk memverifikasi bahwa agen menjalankan.

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

Dalam kebanyakan kasus, perintah akan menampilkan bahwa agen menjalankan, seperti yang ditunjukkan dalam contoh berikut.

```
# 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--
```

Dalam kasus yang jarang terjadi, perintah akan menampilkan bahwa agen diinstal tetapi tidak berjalan, seperti yang ditunjukkan dalam contoh berikut.

```
# 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--
```

Untuk mengaktifkan agen dalam kasus ini, jalankan perintah berikut.

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

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

Membuat perintah instalasi agen kustom untukDebian Server di Wilayah Anda

Ketika Anda menginstalSSM Agent pada beberapa contoh menggunakan skrip atau template, kami sarankan menggunakan file instalasi yang disimpan di tempatWilayah AWS Anda bekerja.

Untuk perintah berikut, kami memberikan contoh yang menggunakan bucket S3 yang dapat diakses publik di Wilayah AS Timur (Ohio) (`us-east-2`).

> **Tip**
>
> Anda juga dapat mengganti URL global dalam prosedurPerintah instalasi cepat untukSSM Agent aktifDebian Server sebelumnya dalam topik ini dengan URL Regional kustom yang Anda buat.

Dalam perintah berikut, ganti *wilayah* dengan informasi Anda sendiri. Untuk daftar nilai *wilayah* yang didukung, lihat kolom Wilayah di endpoint layanan Systems Manager di bagian Referensi Umum Amazon Web Services.

> **Note**
>
> UntukDebian Server 8, hanyax86_64 arsitektur yang didukung.

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
```

Lihat contoh berikut.

```
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
```

Lihat contoh berikut.

```
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
```

Menginstal secara manualSSM Agent padaOracle Linux instance

Amazon Machine Images(AMIs) untukOracle Linux yang disediakan olehAWS tidak datang denganAWS Systems Manager Agen (SSM Agent) terinstal secara default. Untuk daftar yangAWS dikelolaAMIs di mana agen mungkin sudah diinstal sebelumnya, lihatAmazon Machine Images(AMIs) dengan SSM Agent prainstal.

Gunakan informasi di bagian ini untuk membantu Anda menginstal atau menginstal ulang secara manualSSM Agent padaOracle Linux instans.

Sebelum Anda memulai

Sebelum Anda menginstalSSM AgentOracle Linux instans, perhatikan hal berikut:

- Untuk informasi penting yang berlaku untuk instalasiSSM Agent pada semua sistem operasi berbasis Linux, lihatMenginstal secara manual SSM Agent pada instans EC2 untuk Linux.
- Jika Anda menggunakan`yum` perintah untuk memperbaruiSSM Agent pada node terkelola setelah agen telah diinstal atau diperbarui menggunakan dokumen SSM`AWS-UpdateSSMAgent`, Anda mungkin melihat pesan berikut: “Peringatan: RPMDB diubah di luar yum.” Pesan ini diharapkan dan dapat diabaikan dengan aman.

Topik

- Perintah instalasi cepat untukSSM Agent aktifOracle Linux
- Membuat perintah instalasi agen kustom untukOracle Linux di Wilayah Anda

Perintah instalasi cepat untukSSM Agent aktifOracle Linux

Gunakan langkah-langkah berikut untuk menginstal secara manualSSM Agent pada satu instance. Prosedur ini menggunakan file instalasi yang tersedia secara global.

Untuk menginstalSSM AgentOracle Linux menggunakan perintah copy dan paste cepat

1. Connect keOracle Linux instans Anda menggunakan metode pilihan Anda, seperti SSH.
2. Salin perintah berikut dan jalankan pada instance.

> **Note**
>
> Meskipun URL dalam perintah berikut termasuk`ec2-downloads-windows` direktori, ini adalah file instalasi global yang benar untukOracle Linux.

x86_64

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

3. (Disarankan) Jalankan perintah berikut untuk memverifikasi bahwa agen menjalankan.

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

Dalam kebanyakan kasus, perintah akan melaporkan bahwa agen menjalankan, seperti yang ditunjukkan dalam contoh berikut.

```
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--
```

Dalam kasus yang jarang terjadi, perintah akan melaporkan bahwa agen telah diinstal tetapi tidak berjalan, seperti yang ditunjukkan dalam contoh berikut.

```
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--
```

Untuk mengaktifkan agen dalam kasus ini, jalankan perintah berikut.

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

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

Membuat perintah instalasi agen kustom untukOracle Linux di Wilayah Anda

Ketika Anda menginstalSSM Agent pada beberapa contoh menggunakan skrip atau template, kami sarankan menggunakan file instalasi yang disimpan di tempatWilayah AWS Anda bekerja.

Untuk perintah berikut, kami memberikan contoh yang menggunakan bucket S3 yang dapat diakses publik di Wilayah AS Timur (Ohio) (`us-east-2`).

> **Tip**
>
> Anda juga dapat mengganti URL global dalam prosedurPerintah instalasi cepat untukSSM Agent aktifOracle Linux sebelumnya dalam topik ini dengan URL Regional kustom yang Anda buat.

Dalam perintah berikut, ganti *`wilayah`* dengan informasi Anda sendiri. Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Wilayah di endpoint layanan Systems Manager di bagian Referensi Umum Amazon Web Services.

x86_64

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

Lihat contoh berikut.

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

Menginstal secara manual SSM Agent pada Red Hat Enterprise Linux instance

The Amazon Machine Images (AMIs) for Red Hat Enterprise Linux (RHEL) yang disediakan oleh AWS tidak disertakan dengan AWS Systems Manager Agent (SSM Agent) yang telah diinstal sebelumnya secara default. Untuk daftar AWS terkelola AMIs di mana agen mungkin sudah diinstal sebelumnya, lihatAmazon Machine Images(AMIs) dengan SSM Agent prainstal.

Gunakan informasi di bagian ini untuk membantu Anda menginstal atau menginstal ulang secara manual SSM Agent pada sebuah RHEL instans.

Sebelum Anda memulai

Sebelum Anda menginstal SSM Agent pada sebuah RHEL instance, perhatikan hal berikut:

- Untuk informasi penting yang berlaku untuk instalasi SSM Agent pada semua sistem operasi berbasis Linux, lihat. Menginstal secara manual SSM Agent pada instans EC2 untuk Linux
- Jika Anda menggunakan `yum` perintah untuk memperbarui SSM Agent pada node terkelola setelah agen diinstal atau diperbarui menggunakan dokumen SSM`AWS-UpdateSSMAgent`, Anda mungkin melihat pesan berikut: “Peringatan: RPMDB diubah di luar yum.” Pesan ini diharapkan dan dapat diabaikan dengan aman.

Topik

- Instal SSM Agent pada RHEL 8.x dan 9.x
- InstalSSM Agent padaRHEL 7.x
- InstalSSM Agent padaRHEL 6.x

Instal SSM Agent pada RHEL 8.x dan 9.x

Amazon Machine Images(AMIs) untuk RHEL 8 dan 9 yang disediakan oleh AWS tidak disertakan dengan AWS Systems Manager Agen (SSM Agent) yang telah diinstal sebelumnya secara default. Gunakan informasi di halaman ini untuk membantu Anda menginstal atau menginstal ulang agen pada RHEL 8 dan 9 instance.

Sebelum Anda mulai

Sebelum Anda menginstal SSM Agent pada instance RHEL 8 atau 9, perhatikan hal berikut:

- Pastikan Python 2 atau Python 3 diinstal pada instans RHEL 8 atau 9 Anda. Ini diperlukan agar dapat bekerja SSM Agent dengan baik.

Topik

- Perintah instalasi cepat untuk SSM Agent pada RHEL 8 atau 9
- Buat perintah instalasi agen kustom untuk RHEL 8 dan 9 di Wilayah Anda

Perintah instalasi cepat untuk SSM Agent pada RHEL 8 atau 9

Gunakan langkah-langkah berikut untuk menginstal secara manual SSM Agent pada satu instance. Prosedur ini menggunakan file instalasi yang tersedia secara global.

Untuk menginstal SSM Agent pada RHEL 8.x atau 9.x

1. Connect ke instans RHEL 8 atau 9 Anda menggunakan metode pilihan Anda, seperti SSH.
2. Salin perintah untuk arsitektur instans Anda dan jalankan pada instance.

   > **Note**
   >
   > Meskipun URL dalam perintah berikut menyertakan `ec2-downloads-windows` direktori, ini adalah file instalasi global yang benar untuk RHEL 8 dan 9.

   Instans x86_64

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

   Instans ARM64

   ```
   sudo dnf install -y https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/
   latest/linux_arm64/amazon-ssm-agent.rpm
   ```
3. (Disarankan) Jalankan perintah berikut untuk memverifikasi bahwa agen sedang berjalan.

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

   Dalam kebanyakan kasus, perintah melaporkan bahwa agen sedang berjalan, seperti yang ditunjukkan pada contoh berikut.

   ```
   # 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--
```

Dalam kasus yang jarang terjadi, perintah melaporkan bahwa agen diinstal tetapi tidak berjalan, seperti yang ditunjukkan pada contoh berikut.

```
# 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--
```

Untuk mengaktifkan agen dalam kasus ini, jalankan perintah berikut.

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

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

Buat perintah instalasi agen kustom untuk RHEL 8 dan 9 di Wilayah Anda

Saat Anda menginstal SSM Agent pada beberapa instance menggunakan skrip atau template, sebaiknya gunakan file instalasi yang disimpan di tempat Wilayah AWS Anda bekerja.

Untuk perintah berikut, kami memberikan contoh yang menggunakan bucket S3 yang dapat diakses publik di Wilayah Timur AS (Ohio) (). `us-east-2`

> **Tip**
>
> Anda juga dapat mengganti URL global dalam prosedur Perintah instalasi cepat untuk SSM Agent pada RHEL 8 atau 9 sebelumnya dalam topik ini dengan URL Regional kustom yang Anda buat.

Dalam perintah berikut, ganti *`wilayah`* dengan informasi Anda sendiri. Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

x86_64

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

Lihat contoh berikut ini.

```
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
```

Lihat contoh berikut ini.

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

InstalSSM Agent padaRHEL 7.x

Amazon Machine Images(AMIs) untukRHEL 7 yang disediakan olehAWS tidak datang denganAWS Systems Manager Agen (SSM Agent) terinstal secara default. Gunakan informasi di halaman ini untuk membantu Anda menginstal atau menginstal ulang agen padaRHEL 7 instans.

Topik

- Perintah instalasi cepat untukSSM Agent padaRHEL 7
- Buat perintah instalasi agen kustom untukRHEL 7 di Wilayah Anda

Perintah instalasi cepat untukSSM Agent padaRHEL 7

Gunakan langkah-langkah berikut untuk menginstal secara manualSSM Agent pada satu instance. Prosedur ini menggunakan file instalasi yang tersedia secara global.

Untuk menginstalSSM Agent padaRHEL 7.x

1. Connect ke instansRHEL 7 Anda menggunakan metode pilihan Anda, seperti SSH.
2. Salin perintah untuk arsitektur instance Anda dan jalankan pada instance.

> **Note**
>
> Meskipun URL dalam perintah berikut menyertakan`ec2-downloads-windows` direktori, ini adalah file instalasi global yang benar untukRHEL 7.

Instans x86_64

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

Instans ARM64

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

3. (Direkomendasikan) Jalankan perintah berikut untuk memastikan agen menjalankan.

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

Dalam kebanyakan kasus, perintah akan menampilkan bahwa agen menjalankan, seperti yang ditunjukkan dalam contoh berikut.

```
# 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--
```

Dalam kasus yang jarang terjadi, perintah akan menampilkan bahwa agen diinstal tetapi tidak menjalankan, seperti yang ditunjukkan dalam contoh berikut.

```
# 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--
```

Untuk mengaktifkan agen dalam kasus ini, jalankan perintah berikut.

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

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

Buat perintah instalasi agen kustom untukRHEL 7 di Wilayah Anda

Ketika Anda menginstalSSM Agent pada beberapa contoh menggunakan skrip atau template, kami sarankan menggunakan file instalasi yang disimpan di tempatWilayah AWS Anda bekerja.

Untuk perintah berikut, kami memberikan contoh yang menggunakan bucket S3 yang dapat diakses publik di Wilayah AS Timur (Ohio) (`us-east-2`).

> **Tip**
>
> Anda juga dapat mengganti URL global dalam prosedurPerintah instalasi cepat untukSSM Agent padaRHEL 7 sebelumnya dalam topik ini dengan URL Regional kustom yang Anda buat.

Dalam perintah berikut, ganti *`wilayah`* dengan informasi Anda sendiri. Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Wilayah di endpoint layanan Systems Manager di bagian Referensi Umum Amazon Web Services.

x86_64

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

Lihat contoh berikut.

```
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
```

Lihat contoh berikut.

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

InstalSSM Agent padaRHEL 6.x

Amazon Machine Images(AMIs) untukRHEL 6 yang disediakan olehAWS tidak datang denganAWS Systems Manager Agen (SSM Agent) terinstal secara default. Gunakan informasi di halaman ini untuk membantu Anda menginstal atau menginstal ulang agen padaRHEL 6 instans.

Topik

- Perintah instalasi cepat untukSSM Agent padaRHEL 6
- Buat perintah instalasi agen kustom untukRHEL 6 di Wilayah Anda

Perintah instalasi cepat untukSSM Agent padaRHEL 6

Gunakan langkah-langkah berikut untuk menginstal secara manualSSM Agent pada satu instance. Prosedur ini menggunakan file instalasi yang tersedia secara global.

Untuk menginstalSSM Agent padaRHEL 6.x

1. Connect ke instansRHEL 6 Anda menggunakan metode pilihan Anda, seperti SSH.
2. Salin perintah untuk arsitektur instance Anda dan jalankan pada instance.

> **Note**
>
> Meskipun URL dalam perintah berikut menyertakan`ec2-downloads-windows` direktori, ini adalah file instalasi global yang benar untukRHEL 6.

Perintah berikut menentukan direktori versi,`3.0.1479.0` bukan`latest` direktori. Ini karenaSSM Agent versi 3.1 dan yang lebih baru tidak didukung untukRHEL 6.

Instans x86_64

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

Instans x86

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

3. (Direkomendasikan) Jalankan perintah berikut untuk memastikan agen berjalan.

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

Dalam kebanyakan kasus, perintah akan menampilkan bahwa agen menjalankan, seperti yang ditunjukkan dalam contoh berikut.

```
amazon-ssm-agent start/running, process 1788
```

Dalam kasus yang jarang terjadi, perintah akan menampilkan bahwa agen diinstal tetapi tidak berjalan, seperti yang ditunjukkan dalam contoh berikut.

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

Untuk mengaktifkan agen dalam kasus ini, jalankan perintah berikut.

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

Buat perintah instalasi agen kustom untukRHEL 6 di Wilayah Anda

Ketika Anda menginstalSSM Agent pada beberapa contoh menggunakan skrip atau template, kami sarankan menggunakan file instalasi yang disimpan di tempatWilayah AWS Anda bekerja.

Untuk perintah berikut, kami memberikan contoh yang menggunakan bucket S3 yang dapat diakses publik di Wilayah AS Timur (Ohio) (`us-east-2`).

> **Tip**
>
> Anda juga dapat mengganti URL global dalam prosedurPerintah instalasi cepat untukSSM Agent padaRHEL 6 sebelumnya dalam topik ini dengan URL Regional kustom yang Anda buat.

Dalam perintah berikut, ganti *`wilayah`* dengan informasi Anda sendiri. Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Wilayah di endpoint layanan Systems Manager di bagian Referensi Umum Amazon Web Services.

> **Note**
>
> Perintah berikut menentukan direktori versi,`3.0.1390.0` bukan`latest` direktori. Ini karenaSSM Agent versi 3.1 dan yang lebih baru tidak didukung untukRHEL 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
```

Lihat contoh berikut.

```
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
```

Lihat contoh berikut.

```
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
```

Instal secara manualSSM Agent padaRocky Linux instans

Amazon Machine Images(AMIs) untukRocky Linux yang disediakan olehAWS tidak datang denganAWS Systems Manager Agen (SSM Agent) terinstal secara default. Untuk daftar yangAWS dikelolaAMIs di mana agen mungkin sudah diinstal sebelumnya, lihatAmazon Machine Images(AMIs) dengan SSM Agent prainstal.

Gunakan informasi di bagian ini untuk membantu Anda menginstal atau menginstal ulang secara manualSSM Agent padaRocky Linux instans.

Sebelum Anda memulai

Sebelum Anda menginstalSSM AgentRocky Linux instans, perhatikan hal berikut:

- Untuk informasi penting yang berlaku untuk instalasiSSM Agent pada semua sistem operasi berbasis Linux, lihatMenginstal secara manual SSM Agent pada instans EC2 untuk Linux.

Topik

- Perintah instalasi cepat untukSSM Agent aktifRocky Linux
- Membuat perintah instalasi agen kustom untukRocky Linux di Wilayah Anda

Perintah instalasi cepat untukSSM Agent aktifRocky Linux

Gunakan langkah-langkah berikut untuk menginstal secara manualSSM Agent pada satu instance. Prosedur ini menggunakan file instalasi yang tersedia secara global.

Sebelum Anda memulai

Sebelum Anda menginstalSSM AgentRocky Linux instans, perhatikan hal berikut:

- Pastikan bahwa Python 2 atau Python 3 terinstal padaRocky Linux instans Anda. Hal ini diperlukan agar dapatSSM Agent bekerja dengan baik.

UntukSSM Agent menginstalRocky Linux

1. Connect keRocky Linux instans Anda menggunakan metode pilihan Anda, seperti SSH.
2. Salin perintah untuk arsitektur instance Anda dan jalankan pada instance.

> **Note**
>
> Meskipun URL dalam perintah berikut menyertakan`ec2-downloads-windows` direktori, ini adalah file instalasi global yang benar untukRocky Linux.

Instans x86_64

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

Instans ARM64

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

3. (Disarankan) Jalankan perintah berikut untuk memverifikasi bahwa agen menjalankan.

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

Dalam kebanyakan kasus, perintah akan menampilkan bahwa agen menjalankan, seperti yang ditunjukkan dalam contoh berikut.

```
# 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--
```

Dalam kasus yang jarang terjadi, perintah akan menampilkan bahwa agen terinstal tetapi tidak berjalan, seperti yang ditunjukkan dalam contoh berikut.

```
# 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--
```

Untuk mengaktifkan agen dalam kasus ini, jalankan perintah berikut.

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

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

Membuat perintah instalasi agen kustom untukRocky Linux di Wilayah Anda

Ketika Anda menginstalSSM Agent pada beberapa contoh menggunakan skrip atau template, kami sarankan menggunakan file instalasi yang disimpan di tempatWilayah AWS Anda bekerja.

Untuk perintah berikut, kami memberikan contoh yang menggunakan bucket S3 yang dapat diakses publik di Wilayah AS Timur (Ohio) (`us-east-2`).

> **Tip**
>
> Anda juga dapat mengganti URL global dalam prosedurPerintah instalasi cepat untukSSM Agent aktifRocky Linux sebelumnya dalam topik ini dengan URL Regional kustom yang Anda buat.

Dalam perintah berikut, ganti *`wilayah`* dengan informasi Anda sendiri. Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Wilayah di endpoint layanan Systems Manager di bagian Referensi Umum Amazon Web Services.

x86_64

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

Lihat contoh berikut.

```
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
```

Lihat contoh berikut.

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

Instal secara manual SSM Agent pada SUSE Linux Enterprise Server instans

Dalam kebanyakan kasus, Amazon Machine Images (AMIs) for SUSE Linux Enterprise Server (SLES) yang disediakan oleh AWS come with AWS Systems Manager Agent (SSM Agent) terinstal secara default. Untuk informasi selengkapnya, lihat Amazon Machine Images(AMIs) dengan SSM Agent prainstal.

Jika SSM Agent tidak diinstal sebelumnya pada SLES instans baru, atau jika Anda perlu menginstal ulang agen secara manual, gunakan informasi di halaman ini untuk membantu Anda.

Sebelum Anda memulai

Sebelum Anda menginstal SSM Agent pada SLES instans, perhatikan hal berikut:

- Untuk informasi penting yang berlaku untuk instalasi SSM Agent pada semua sistem operasi berbasis Linux, lihat. Menginstal secara manual SSM Agent pada instans EC2 untuk Linux

Topik

- Perintah instalasi cepat untuk SSM Agent aktif SLES
- Membuat perintah instalasi agen kustom untuk SLES di Wilayah Anda

Perintah instalasi cepat untuk SSM Agent aktif SLES

Gunakan langkah-langkah berikut untuk menginstal secara manual SSM Agent pada satu instance. Prosedur ini menggunakan file instalasi yang tersedia secara global.

Untuk menginstal SSM AgentSLES menggunakan perintah copy dan paste cepat

1. Connect ke SLES instans Anda menggunakan metode pilihan Anda, seperti SSH.

2. Opsi 1: Gunakan `zypper` perintah:

   - Jalankan perintah berikut:

     ```
     sudo zypper install amazon-ssm-agent
     ```

   - Masukkan `y` dalam menanggapi setiap petunjuk.

   Opsi 2: Gunakan `rpm` perintah.

   - Menciptakan direktori sementara pada instans.

     ```
     mkdir /tmp/ssm
     ```

   - Mengubah ke direktori sementara.

     ```
     cd /tmp/ssm
     ```

   - Jalankan perintah berikut satu per satu untuk mengunduh dan menjalankan SSM Agent penginstal.

     > **Note**
     >
     > Meskipun URL dalam perintah berikut menyertakan `ec2-downloads-windows` direktori, ini adalah file instalasi global yang benar untukSLES.

     Instans x86_64:

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

     Instans ARM64:

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

   - Jalankan perintah berikut.

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

- (Disarankan) Jalankan perintah berikut untuk memverifikasi bahwa agen menjalankan.

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

Dalam kebanyakan kasus, perintah akan menampilkan bahwa agen menjalankan, seperti yang ditunjukkan dalam contoh berikut.

```
# 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--
```

Dalam kasus yang jarang terjadi, perintah akan menampilkan bahwa agen diinstal tetapi tidak berjalan, seperti yang ditunjukkan dalam contoh berikut.

```
# 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--
```

Untuk mengaktifkan agen dalam kasus ini, jalankan perintah berikut.

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

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

Membuat perintah instalasi agen kustom untuk SLES di Wilayah Anda

Ketika Anda menginstal SSM Agent pada beberapa contoh menggunakan skrip atau template, kami sarankan menggunakan file instalasi yang disimpan di tempat Wilayah AWS Anda bekerja.

Untuk perintah berikut, kami memberikan contoh yang menggunakan bucket S3 yang dapat diakses publik di Wilayah AS Timur (Ohio) (). `us-east-2`

> **Tip**
>
> Anda juga dapat mengganti URL global dalam prosedur Perintah instalasi cepat untuk SSM Agent di Amazon Linux 1 sebelumnya dalam topik ini dengan URL Regional kustom yang Anda buat.

Dalam perintah berikut, ganti *`wilayah`* dengan informasi Anda sendiri. Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Wilayah di endpoint layanan Systems Manager di Referensi Umum 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
```

Lihat contoh berikut.

```
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
```

Lihat contoh berikut.

```
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
```

Menginstal secara manual SSM Agent pada Ubuntu Server instance

> **Important**
>
> Sebelum Anda menginstal SSM Agent pada versi 64-bitUbuntu Server, pastikan bahwa Anda menggunakan alat instalasi yang benar. Dimulai dengan Amazon Machine Images (AMI) yang diidentifikasi dengan 20180627, SSM Agent sudah diinstal sebelumnya pada versi 16.04 menggunakan paket Snap. Pada instance yang dibuat dari AMI sebelumnya, SSM Agent harus diinstal menggunakan paket installer deb. Untuk informasi selengkapnya, lihat Menentukan SSM Agent versi yang benar untuk diinstal pada instance 64-bit Ubuntu Server 16.04.

Dalam kebanyakan kasus, Amazon Machine Images (AMIs) untuk Ubuntu Server itu disediakan oleh AWS datang dengan AWS Systems Manager Agen (SSM Agent) yang telah diinstal sebelumnya secara default. Untuk informasi selengkapnya, lihat Amazon Machine Images(AMIs) dengan SSM Agent prainstal.

Jika SSM Agent tidak diinstal sebelumnya pada Ubuntu Server instance baru, atau jika Anda perlu menginstal ulang agen secara manual, gunakan informasi di bagian ini untuk membantu Anda.

Sebelum Anda memulai

Sebelum Anda menginstal SSM Agent pada sebuah Ubuntu Server instance, perhatikan hal berikut:

- Untuk informasi penting yang berlaku untuk instalasi SSM Agent pada semua sistem operasi berbasis Linux, lihat. Menginstal secara manual SSM Agent pada instans EC2 untuk Linux

Topik

- InstalSSM Agent padaUbuntu Server 22.04 LTS, 20.10 STR & 20.04, 18.04, dan 16.04 LTS 64-bit (Snap)
- Instal SSM Agent pada Ubuntu Server 16.04 dan 14.04 64-bit (deb)
- InstalSSM Agent padaUbuntu Server 16.04 dan 14.04 32-bit
- Menentukan SSM Agent versi yang benar untuk diinstal pada instance 64-bit Ubuntu Server 16.04

InstalSSM Agent padaUbuntu Server 22.04 LTS, 20.10 STR & 20.04, 18.04, dan 16.04 LTS 64-bit (Snap)

Sebelum Anda memulai

Sebelum Anda menginstalSSM Agent padaUbuntu Server 22.04 LTS, 20.10 STR & 20.04, 18.04, dan 16.04 LTS 64-bit (Snap), perhatikan hal berikut:

Versi 16.04 instalasi oleh Snaps atau deb installer

PadaUbuntu Server 16.04,SSM Agent diinstal menggunakan Snap atau paket instalasi deb, tergantung pada versi 16.04AMI.

SSM Agentlokasi file installer

PadaUbuntu Server 22.04 LTS, 20.10 STR & 20.04, 18.04, dan 16.04 LTS (dengan Snap), fileSSM Agent penginstal, termasuk binari agen dan file config, disimpan dalam direktori berikut:`/snap/amazon-ssm-agent/current/`. Jika Anda membuat perubahan ke file konfigurasi dalam direktori ini, maka Anda harus menyalin file-file ini dari direktori `/snap` ke `/etc/amazon/ssm/`. File log dan perpustakaan belum berubah (`/var/lib/amazon/ssm`, `/var/log/amazon/ssm`).

Menggunakan`candidate` saluran Snap

Saluran kandidat dalam menyimpan Snap berisi versi terbaruSSM Agent (termasuk semua perbaikan bug terbaru); bukan saluran stabil. Untuk mempelajari selengkapnya tentang perbedaan antara kandidat dan saluran stabil, lihat Risk-levels di https://snapcraft.io/docs/channels.

Jika Anda ingin melacak informasiSSM Agent versi pada saluran kandidat, jalankan perintah berikut pada instansUbuntu Server 20.04 & 20.04, 18.04, dan 16.04 LTS 64-bit.

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

Snap direkomendasikan pada versi 18.04 dan lebih baru

PadaUbuntu Server 22.04 LTS, 20.10 STR & 20.04 dan 18.04 LTS, kami sarankan Anda hanya menggunakan Snap. Juga verifikasi bahwa hanya satu instans agen terinstal dan berjalan pada instans Anda. Jika Anda ingin menggunakanSSM Agent tanpa Snap, hapus instalanSSM Agent. Kemudian instal paketSSM Agent sebagai debian menggunakan petunjuk untuk menginstalSSM Agent padaUbuntu Server 16.04 dan 14.04 64-bit (deb). Sebelum menginstal, pastikan Anda tidak menginstal Snap yang tumpang tindih dengan daftar paket yang ingin Anda kelola sebagai paket debian.

`Maximum timeout exceeded`penanggulangan pesan kesalahan

Karena masalah yang diketahui dengan Snap, Anda mungkin melihat kesalahan `Maximum timeout exceeded` dengan perintah `snap`. Jika Anda mendapatkan kesalahan ini, jalankan perintah berikut satu per satu untuk memulai agen, menghentikannya, dan memeriksa statusnya:

```
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
```

Untuk menginstalSSM Agent pada instansUbuntu Server 22.04 LTS, 20.04, 18.04, dan 16.04, dan 16.04 LTS 64-bit (dengan paket Snap)

1. SSM Agentdiinstal, secara default, padaUbuntu Server 22.04 LTS, 20.04, 18.04, dan 16.04 LTS 64-bitAMIs dengan pengidentifikasi`20180627` atau lebih baru.

   Anda dapat menggunakan skrip berikut jika Anda perlu menginstalSSM Agent di server on-premise atau jika Anda perlu menginstal ulang agen. Anda tidak perlu menentukan URL untuk unduhan, karena perintah `snap` secara otomatis mengunduh agen dari toko aplikasi Snap di https://snapcraft.io.

   ```
   sudo snap install amazon-ssm-agent --classic
   ```

2. Jalankan perintah berikut untuk menentukanSSM Agent apakah berjalan.

   ```
   sudo snap list amazon-ssm-agent
   ```

3. Jalankan perintah berikut untuk memulai layanan jika perintah sebelumnya kembali menjadi `amazon-ssm-agent is stopped`, `inactive`, atau `disabled`.

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

4. Periksa status agen.

```
sudo snap services amazon-ssm-agent
```

Instal SSM Agent pada Ubuntu Server 16.04 dan 14.04 64-bit (deb)

> **Important**
>
> Sebelum Anda menginstal SSM Agent pada versi 64-bitUbuntu Server, pastikan bahwa Anda menggunakan alat instalasi koreksi. Dimulai dengan Amazon Machine Images (AMI) yang diidentifikasi dengan 20180627, SSM Agent sudah diinstal sebelumnya pada versi 16.04 menggunakan paket Snap. Pada instance yang dibuat dari AMI sebelumnya, SSM Agent harus diinstal menggunakan paket installer deb. Untuk informasi lebih lanjut, lihatMenentukan SSM Agent versi yang benar untuk diinstal pada instance 64-bit Ubuntu Server 16.04. I f SSM Agent diinstal pada instance Anda bersama dengan Snap dan Anda menginstal atau memperbarui SSM Agent menggunakan paket penginstal deb, instalasi atau SSM Agent operasi mungkin gagal.

Dalam kebanyakan kasus, Amazon Machine Images (AMIs) Ubuntu Server 16.04 yang disediakan oleh AWS datang dengan AWS Systems Manager Agent (SSM Agent) yang telah diinstal sebelumnya secara default. Untuk informasi selengkapnya, lihat Amazon Machine Images(AMIs) dengan SSM Agent prainstal.

Jika SSM Agent tidak diinstal sebelumnya pada instance Ubuntu Server 16.04 baru sebelum versi 20180627, Anda menginstal pada Ubuntu Server 14.04, atau Anda perlu menginstal ulang agen secara manual, gunakan informasi di halaman ini untuk membantu Anda.

Perintah instalasi cepat untuk SSM Agent pada Ubuntu Server 16.04 dan 14.04 64-bit (deb)

Gunakan langkah-langkah berikut untuk menginstal secara manual SSM Agent pada satu instance. Prosedur ini menggunakan file instalasi yang tersedia secara global.

Untuk menginstal SSM Agent pada Ubuntu Server 16.04 dan 14.04 64-bit (deb) menggunakan perintah salin dan tempel cepat

1. Connect ke Ubuntu Server instans Anda menggunakan metode pilihan Anda, seperti SSH.
2. Jalankan perintah berikut untuk membuat direktori sementara pada instance.

```
mkdir /tmp/ssm
```

3. Mengubah ke direktori sementara.

```
cd /tmp/ssm
```

4. Jalankan perintah berikut.

> **Note**
>
> Meskipun URL dalam perintah berikut menyertakan `ec2-downloads-windows` direktori, ini adalah file instalasi global yang benar untuk Ubuntu Server 16.04 dan 14.04 64-bit.

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

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

5. (Disarankan) Jalankan salah satu perintah berikut untuk menentukan apakah SSM Agent sedang berjalan.

   Ubuntu Server16.04

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

   Ubuntu Server14.04

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

   Dalam kebanyakan kasus, perintah melaporkan bahwa agen sedang berjalan.

Dalam kasus yang jarang terjadi, perintah melaporkan bahwa agen diinstal tetapi tidak berjalan, seperti yang ditunjukkan pada contoh berikut.

6. Jalankan salah satu perintah berikut untuk memulai layanan jika perintah sebelumnya kembali menjadi `amazon-ssm-agent is stopped`, `inactive`, atau `disabled`.

   Ubuntu Server16.04:

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

   Ubuntu Server14.04:

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

Buat perintah instalasi khusus untuk SSM Agent pada Ubuntu Server 16.04 dan 14.04 64-bit (deb) di Wilayah Anda

Saat Anda menginstal SSM Agent pada beberapa instance menggunakan skrip atau template, sebaiknya gunakan file instalasi yang disimpan di tempat Wilayah AWS Anda bekerja.

Untuk perintah berikut, kami memberikan contoh yang menggunakan bucket S3 yang dapat diakses publik di Wilayah Timur AS (Ohio) (). `us-east-2`

> **Tip**
>
> Anda juga dapat mengganti URL global dalam prosedur Perintah instalasi cepat untuk SSM Agent pada Ubuntu Server 16.04 dan 14.04 64-bit (deb) sebelumnya dalam topik ini dengan URL Regional kustom yang Anda buat.

Dalam perintah berikut, ganti *`wilayah`* dengan informasi Anda sendiri. Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum 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
```

Lihat contoh berikut.

```
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
```

InstalSSM Agent padaUbuntu Server 16.04 dan 14.04 32-bit

Dalam kebanyakan kasus,Amazon Machine Images (AMIs)Ubuntu Server 16.04 yang disediakan olehAWS come withAWS Systems Manager Agent (SSM Agent) terinstal secara default. Untuk informasi selengkapnya, lihat Amazon Machine Images(AMIs) dengan SSM Agent prainstal.

JikaSSM Agent tidak diinstal sebelumnya pada instansUbuntu Server 16.04 baru, Anda menginstal padaUbuntu Server 14.04, atau Anda perlu menginstal ulang agen secara manual, gunakan informasi di halaman ini untuk membantu Anda.

Perintah penginstalan cepat untukSSM Agent padaUbuntu Server 16.04 dan 14.04 32-bit (deb)

Gunakan langkah-langkah berikut untuk menginstal secara manualSSM Agent pada satu instance. Prosedur ini menggunakan file instalasi yang tersedia secara global.

Untuk menginstalSSM Agent padaUbuntu Server 16.04 dan 14.04 32-bit (deb) menggunakan perintah penginstal dan tempel cepat

1. Connect keUbuntu Server instans Anda menggunakan metode pilihan Anda, seperti SSH.
2. Jalankan perintah berikut ini untuk membuat sebuah direktori sementara pada instans.

   ```
   mkdir /tmp/ssm
   ```
3. Mengubah ke direktori sementara.

   ```
   cd /tmp/ssm
   ```
4. Jalankan perintah berikut.

   > **Note**
   >
   > Meskipun URL dalam perintah berikut menyertakan`ec2-downloads-windows` direktori, ini adalah file instalasi global yang benar untukUbuntu Server 16.04 dan 14.04 32-bit.

```
wget https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/debian_386/
amazon-ssm-agent.deb
```

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

5. (Direkomendasikan) Jalankan salah satu perintah berikut untuk menentukanSSM Agent apakah menjalankan.

   Ubuntu Server16.04

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

   Ubuntu Server14.04

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

   Dalam sebagian besar kasus, perintah akan menampilkan bahwa agen menjalankan.

   Dalam kasus yang jarang, perintah akan menampilkan bahwa agen diinstal tetapi tidak berjalan, seperti yang ditunjukkan dalam contoh berikut.

6. Jalankan salah satu perintah berikut untuk memulai layanan jika perintah sebelumnya kembali menjadi `amazon-ssm-agent is stopped`, `inactive`, atau `disabled`.

   Ubuntu Server16.04:

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

   Ubuntu Server14.04:

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

Buat perintah penginstalan khusus untukSSM Agent padaUbuntu Server 16.04 dan 14.04 32-bit (deb) di Wilayah Anda

Ketika Anda menginstalSSM Agent pada beberapa contoh menggunakan skrip atau template, kami sarankan menggunakan file instalasi yang disimpan di tempatWilayah AWS Anda bekerja.

Untuk perintah berikut, kami memberikan contoh yang menggunakan bucket S3 yang dapat diakses publik di Wilayah AS Timur (Ohio) (`us-east-2`).

> **Tip**
>
> Anda juga dapat mengganti URL global dalam prosedurPerintah penginstalan cepat untukSSM Agent padaUbuntu Server 16.04 dan 14.04 32-bit (deb) sebelumnya dalam topik ini dengan URL Regional kustom yang Anda buat.

Dalam perintah berikut, ganti *`wilayah`* dengan informasi Anda sendiri. Untuk daftar nilai *`wilayah`* yang didukung, lihat kolom Wilayah di endpoint layanan Systems Manager di bagian Referensi Umum 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
```

Lihat contoh berikut.

```
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
```

Menentukan SSM Agent versi yang benar untuk diinstal pada instance 64-bit Ubuntu Server 16.04

> **Important**
>
> Sebelum Anda menginstal SSM Agent pada versi 64-bitUbuntu Server, pastikan bahwa Anda menggunakan alat instalasi koreksi. Dimulai dengan Amazon Machine Images (AMI)

yang diidentifikasi dengan 20180627, SSM Agent sudah diinstal sebelumnya pada versi 16.04 menggunakan paket Snap. Pada instance yang dibuat dari AMI sebelumnya, SSM Agent harus diinstal menggunakan paket installer deb. Untuk informasi selengkapnya, lihat Menentukan SSM Agent versi yang benar untuk diinstal pada instance 64-bit Ubuntu Server 16.04

Ketahuilah bahwa jika sebuah instans memiliki lebih dari satu instalasi SSM Agent (misalnya, satu diinstal menggunakan Snap dan satu diinstal menggunakan penginstal deb), operasi agen Anda tidak akan berfungsi dengan benar.

Anda dapat memverifikasi tanggal pembuatan ID AMI sumber untuk instans yang menggunakan salah satu metode berikut. Prosedur ini hanya berlaku untuk AMIs yang dikelola AWS.

Verifikasi tanggal pembuatan ID AMI sumber (konsol)

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi kiri, pilih Instans.
3. Pilih instans.
4. Pada tab Detail, periksa YYYYMMDD pengenal di bidang nilai di bawah AMInama. Sebagai contoh: `ubuntu/images/hvm-ssd/ubuntu-xenial-16.04-amd64-server-20180627`.

Verifikasi tanggal pembuatan ID AMI sumber (AWS CLI)

- Jalankan perintah berikut.

```
aws ec2 describe-images --image-ids ami-id
```

ami-id mewakili ID dari AMI yang disediakan oleh AWS, seperti `ami-07c8bc5c1ce9598c3`.

Jika berhasil, perintah mengembalikan informasi seperti berikut, di mana Anda dapat memeriksa bidang `CreationDate` dan `Name` untuk informasi.

```
{
    "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"
        }
    ]
}
```

Mengkonfigurasi SSM Agent untuk menggunakan proxy (Linux)

Anda dapat mengonfigurasi AWS Systems Manager Agent (SSM Agent) untuk berkomunikasi melalui proxy HTTP dengan membuat file konfigurasi override dan menambahkan `http_proxyhttps_proxy`, serta `no_proxy` pengaturan ke file. File override juga mempertahankan pengaturan proxy jika Anda menginstal versi yang lebih baru atau yang lebih lama. SSM Agent Bagian ini mencakup prosedur untuk membuat file override di lingkungan upstart dan systemd. Jika Anda ingin menggunakanSession Manager, perhatikan bahwa server proxy HTTPS tidak didukung.

> **Note**
>
> Node terkelola yang dibuat dari Amazon Linux 1 AMI yang menggunakan proxy harus menjalankan versi `Python requests` modul saat ini untuk mendukung Patch Manager operasi. Untuk informasi selengkapnya, lihat Memutakhirkan modul permintaan Python di Amazon Linux 1 instance yang menggunakan server proxy.

Topik

- Konfigurasikan SSM Agent untuk menggunakan proxy (pemula)
- Konfigurasikan SSM Agent untuk menggunakan proxy (systemd)
- Memutakhirkan modul permintaan Python di Amazon Linux 1 instance yang menggunakan server proxy

Konfigurasikan SSM Agent untuk menggunakan proxy (pemula)

Gunakan prosedur berikut untuk membuat file konfigurasi override untuk lingkungan `upstart`.

Untuk mengkonfigurasi SSM Agent untuk menggunakan proxy (pemula)

1. Connect ke instance terkelola tempat Anda menginstalSSM Agent.
2. Buka editor sederhana seperti VIM, dan tergantung pada apakah Anda menggunakan server proksi HTTP atau server proksi HTTPS, tambahkan salah satu konfigurasi berikut.

   Untuk server proxy HTTP:

   ```
   env http_proxy=http://hostname:port
   env https_proxy=http://hostname:port
   env no_proxy=169.254.169.254
   ```

   Untuk server proxy HTTPS:

   ```
   env http_proxy=http://hostname:port
   env https_proxy=https://hostname:port
   env no_proxy=169.254.169.254
   ```

   > **Note**
   >
   > Tambahkan pengaturan `no_proxy` ke file dan menentukan alamat IP yang tercantum di sini. Ini adalah titik akhir metadata instans untuk Systems Manager. Tanpa alamat IP ini, panggilan ke Systems Manager gagal.

3. Simpan file dengan nama `amazon-ssm-agent.override` di lokasi berikut: `/etc/init/`
4. Berhenti dan mulai ulang SSM Agent menggunakan perintah berikut.

   ```
   sudo service stop amazon-ssm-agent
   sudo service start amazon-ssm-agent
   ```

> **Note**
>
> Untuk informasi lebih lanjut tentang bekerja dengan file `.override` di lingkungan Upstart, lihat init: Konfigurasi pekerjaan daemon init upstart.

Konfigurasikan SSM Agent untuk menggunakan proxy (systemd)

Gunakan prosedur berikut untuk mengonfigurasi SSM Agent agar menggunakan proxy di `systemd` lingkungan.

> **Note**
>
> Beberapa langkah dalam prosedur ini berisi instruksi eksplisit untuk Ubuntu Server contoh di mana diinstal menggunakan SSM Agent Snap.

1. Connect ke instance tempat Anda menginstalSSM Agent.
2. Jalankan salah satu perintah berikut, tergantung pada jenis sistem operasi.

   - Pada Ubuntu Server contoh di mana SSM Agent diinstal dengan menggunakan snap:

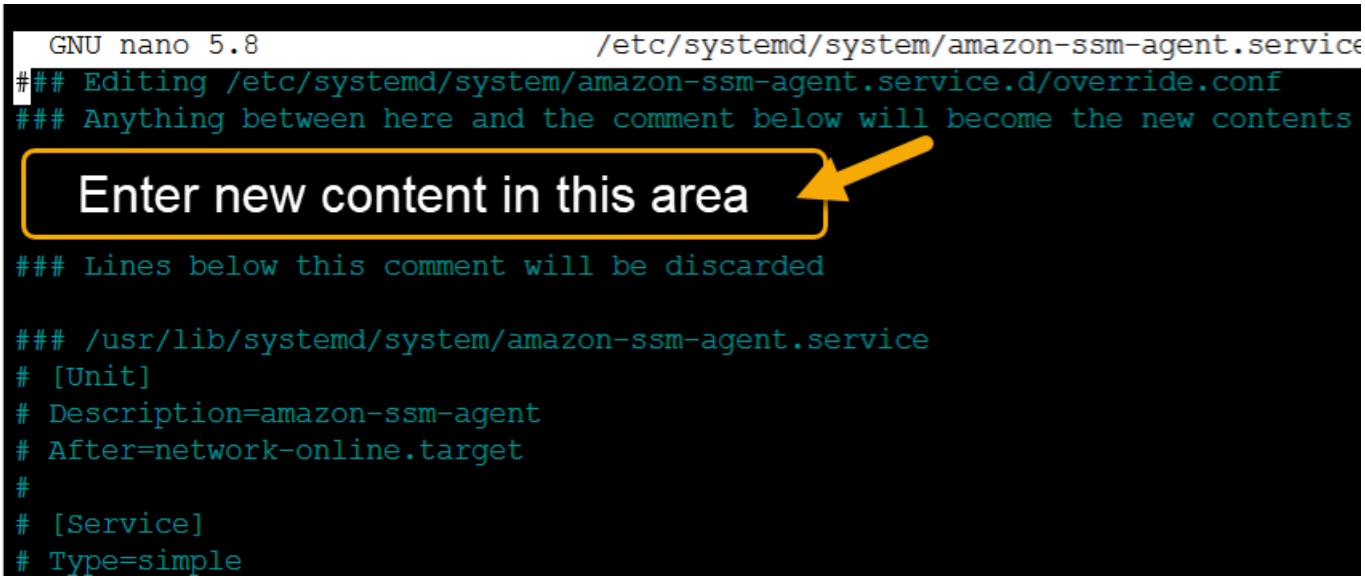
     ```
     sudo systemctl edit snap.amazon-ssm-agent.amazon-ssm-agent
     ```

     Di sistem operasi lain:

     ```
     sudo systemctl edit amazon-ssm-agent
     ```

3. Buka editor sederhana seperti VIM, dan tergantung pada apakah Anda menggunakan server proksi HTTP atau server proksi HTTPS, tambahkan salah satu konfigurasi berikut.

   Pastikan Anda memasukkan informasi di atas komentar yang bertuliskan "`### Lines below this comment will be discarded`“, seperti yang ditunjukkan pada gambar berikut.



Untuk server proxy HTTP:

```
[Service]
Environment="http_proxy=http://hostname:port"
Environment="https_proxy=http://hostname:port"
Environment="no_proxy=169.254.169.254"
```

Untuk server proxy HTTPS:

```
[Service]
Environment="http_proxy=http://hostname:port"
Environment="https_proxy=https://hostname:port"
Environment="no_proxy=169.254.169.254"
```

> **Note**
>
> Tambahkan pengaturan `no_proxy` ke file dan menentukan alamat IP yang tercantum di sini. Ini adalah titik akhir metadata instans untuk Systems Manager. Tanpa alamat IP ini, panggilan ke Systems Manager gagal.

4. Simpan perubahan Anda. Sistem secara otomatis membuat salah satu file berikut, tergantung pada jenis sistem operasi.

   - Pada Ubuntu Server contoh di mana SSM Agent diinstal dengan menggunakan snap:

`/etc/systemd/system/snap.amazon-ssm-agent.amazon-ssm-agent.service.d/`**`override.conf`**

- Di Amazon Linux 2 dan Amazon Linux 2023 instans:

  `/etc/systemd/system/amazon-ssm-agent.service.d/`**`override.conf`**

- Di sistem operasi lain:

  `/etc/systemd/system/amazon-ssm-agent.service.d/`**`amazon-ssm-agent.override`**

5. Mulai ulang SSM Agent dengan menggunakan salah satu perintah berikut, tergantung pada jenis sistem operasi.

   - Pada Ubuntu Server instance yang diinstal dengan menggunakan snap:

     ```
     sudo systemctl daemon-reload && sudo systemctl restart snap.amazon-ssm-agent.amazon-ssm-agent
     ```

   - Di sistem operasi lain:

     ```
     sudo systemctl daemon-reload && sudo systemctl restart amazon-ssm-agent
     ```

> **Note**
>
> Untuk informasi lebih lanjut tentang bekerja dengan file `.override` di lingkungan systemd, lihat Memodifikasi File Unit yang Sudah Ada di Panduan Administrator Sistem Red Hat Enterprise Linux 7.

Memutakhirkan modul permintaan Python di Amazon Linux 1 instance yang menggunakan server proxy

Untuk menambal instance yang menggunakan proxy dan yang dibuat dari Amazon Linux 1 AMIPatch Manager, kemampuan AWS Systems Manager, memerlukan versi terbaru dari `requests` modul Python untuk diinstal pada instance. Kami sarankan untuk selalu meningkatkan ke versi yang paling baru dirilis.

Untuk memastikan versi terbaru dari modul Python `requests` terinstal, ikuti langkah berikut:

1. Masuk ke instans Amazon Linux 1, atau gunakan AWS Systems Manager dokumen (dokumen SSM) `AWS-RunShellScript`Run Command, kemampuan AWS Systems Manager, dan jalankan perintah berikut pada instance.

   ```
   pip list | grep requests
   ```

   - Jika modul diinstal, permintaan mengembalikan nomor versi dalam respons yang serupa dengan yang berikut.

     ```
     requests (1.2.3)
     ```

   - Jika modul tidak diinstal, jalankan perintah berikut untuk menginstalnya.

     ```
     pip install requests
     ```

   - Jika pip sendiri tidak diinstal, jalankan perintah berikut untuk menginstalnya.

     ```
     sudo yum install -y python-pip
     ```

2. Jika modul diinstal, tetapi versi yang tercantum lebih awal dari `2.18.4` (seperti `1.2.3` yang ditunjukkan di langkah sebelumnya), jalankan perintah berikut untuk meningkatkan ke versi terbaru dari modul Python `requests`.

   ```
   pip install requests --upgrade
   ```

Menghapus instalasi SSM Agent dari instance Linux

Gunakan perintah berikut untuk menghapus instalan AWS Systems Manager Agen (SSM Agent).

Amazon Linux 1, Amazon Linux 2, Amazon Linux 2023, Oracle Linux CentOS, dan Red Hat Enterprise Linux

```
sudo yum erase amazon-ssm-agent --assumeyes
```

Debian Server

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

SUSE Linux Enterprise Server (SLES)

- `zypper`instalasi perintah:

```
sudo zypper remove amazon-ssm-agent
```

- `rpm`instalasi perintah:

```
sudo rpm --erase amazon-ssm-agent
```

Ubuntu Server

- instalasi paket deb:

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

- instalasi paket snap:

```
sudo snap remove amazon-ssm-agent
```

Bekerja dengan SSM Agent instans EC2 untuk macOS

AWS Systems Manager (SSM Agent) memproses permintaan Systems Manager dan mengonfigurasi mesin Anda sebagaimana ditentukan dalam permintaan. Gunakan prosedur berikut untuk menginstal, mengkonfigurasi, atau menghapus instalan SSM Agent untukmacOS.

> **Note**
>
> SSM Agentsudah diinstal sebelumnya, secara default, on Amazon Machine Images (AMIs) untukmacOS. Anda tidak perlu menginstal SSM Agent pada instance macOS Amazon Elastic Compute Cloud (Amazon EC2) kecuali Anda telah mencopotnya.

Kode sumber SSM Agent tersedia GitHubsehingga Anda dapat menyesuaikan agen untuk memenuhi kebutuhan Anda. Kami mendorong Anda untuk mengirim permintaan tarik untuk perubahan yang ingin Anda sertakan. Namun, AWS tidak menyediakan dukungan untuk menjalankan salinan yang dimodifikasi dari perangkat lunak ini.

> **Note**
>
> Untuk melihat detail tentang versi yang berbedaSSM Agent, lihat catatan rilis.

Sebelum Anda menginstal secara manual SSM Agent pada sistem macOS operasi, tinjau informasi berikut.

- SSM Agentdiinstal secara default pada instans EC2 berikut dan: Amazon Machine Images
  - macOS 10.14.x (Mojave)
  - macOS 10.15.x (Catalina)
  - macOS11.x () BigSur
  - macOS12.x (Monterey)
  - macOS13.x (Ventura)
  - macOS14.x (Sonoma)

  SSM Agenttidak perlu diinstal secara manual pada instans macOS EC2 kecuali telah dihapus instalasinya.
- Instans EC2 untuk tidak macOS didukung sama sekali. Wilayah AWS Untuk daftar Wilayah yang didukung instans berbasis x86 dan M1 EC2, lihat beban kerja macOSdi macOS FAQ Amazon EC2.
- Versi terbaru dirilis setiap kali kemampuan baru ditambahkan ke Systems Manager atau pembaruan dibuat untuk kemampuan yang ada. SSM Agent Gagal menggunakan agen versi terbaru dapat mencegah node terkelola Anda menggunakan berbagai kemampuan dan fitur Systems Manager. Untuk alasan itu, kami menyarankan Anda mengotomatiskan proses menjaga agar tetap SSM Agent up to date pada mesin Anda. Untuk informasi, lihat Mengotomatiskan pembaruan ke SSM Agent. Berlangganan halaman Catatan SSM Agent Rilis GitHub untuk mendapatkan pemberitahuan tentang SSM Agent pembaruan.

Topik

- Menginstal secara manual SSM Agent pada instans EC2 untuk macOS
- KonfigurasiSSM Agentuntuk menggunakan proxy (macOS)
- Menghapus instalasiSSM AgentdarimacOScontoh

Menginstal secara manual SSM Agent pada instans EC2 untuk macOS

Connect ke macOS instans Anda dan lakukan langkah-langkah berikut untuk menginstal AWS Systems Manager Agent (SSM Agent). Lakukan langkah-langkah ini pada setiap instans yang akan menjalankan perintah menggunakan Systems Manager. Perintah yang disediakan dalam prosedur ini juga dapat diteruskan ke instans Amazon EC2 sebagai skrip melalui data pengguna.

Untuk menginstal SSM Agent pada macOS

1. Unduh file penginstal agen menggunakan perintah berikut.

   Dalam perintah berikut, ganti *wilayah* dengan informasi Anda sendiri. Untuk daftar nilai *wilayah* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

   ```
   sudo wget https://s3.region.amazonaws.com/amazon-ssm-region/latest/darwin_amd64/
   amazon-ssm-agent.pkg
   ```

   Inilah contohnya.

   ```
   sudo wget https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
   darwin_amd64/amazon-ssm-agent.pkg
   ```

2. Gunakan perintah berikut untuk menjalankan SSM Agent installer.

   x86_64:

   ```
   sudo installer -pkg amazon-ssm-agent.pkg -target /
   ```

3. Periksa status agen.

   Untuk menentukan apakah SSM Agent sedang berjalan, periksa log agen di`/var/log/amazon/ssm/amazon-ssm-agent.log`.

4. Jalankan perintah berikut untuk memulai layanan jika log agen menunjukkan bahwa "amazon-ssm-agent dihentikan.”

   ```
   sudo launchctl load -w /Library/LaunchDaemons/com.amazon.aws.ssm.plist && sudo
    launchctl start com.amazon.aws.ssm
   ```

> **Important**
>
> Versi terbaru dirilis setiap kali kemampuan baru ditambahkan ke Systems Manager atau pembaruan dibuat untuk kemampuan yang ada. SSM Agent Gagal menggunakan agen versi terbaru dapat mencegah node terkelola Anda menggunakan berbagai kemampuan dan fitur Systems Manager. Untuk alasan itu, kami menyarankan Anda mengotomatiskan proses menjaga agar tetap SSM Agent up to date pada mesin Anda. Untuk informasi, lihat Mengotomatiskan pembaruan ke SSM Agent. Berlangganan halaman Catatan SSM Agent Rilis GitHub untuk mendapatkan pemberitahuan tentang SSM Agent pembaruan.

KonfigurasiSSM Agentuntuk menggunakan proxy (macOS)

Anda dapat mengonfigurasiAWS Systems ManagerAgen (SSM Agent) untuk berkomunikasi melalui proksi HTTP dengan menambahkan proxy web, proxy web yang aman, dan pengaturan proxy bypass ke konfigurasi JaringanmacOSInstans Amazon Elastic Compute Cloud (Amazon EC2). Untuk informasi lebih lanjut, konsultasikan dokumentasi pengguna macOS Anda.

Menghapus instalasiSSM AgentdarimacOScontoh

macOS tidak mendukung penghapusan instalasi secara native dari file PKG. Untuk menghapus instalanAWS Systems ManagerAgen (SSM Agent) dari Amazon Elastic Compute Cloud (Amazon EC2) untukmacOS, Anda dapat menggunakanAWSskrip terkelola dari lokasi berikut.

https://github.com/aws/amazon-ssm-agent/blob/mainline/Tools/src/update/darwin/uninstall.sh

Bekerja dengan SSM Agent instans EC2 untuk Windows Server

AWS Systems Manager Agent (SSM Agent) sudah diinstal sebelumnya, secara default, pada Amazon Machine Images (AMIs) untuk Windows Server yang disediakan oleh AWS. Support disediakan untuk versi sistem operasi (OS) berikut.

- Windows Server 2008-2012 R2 AMIs yang dipublikasikan pada November 2016 atau yang lebih baru
- Windows Server2016, 2019, dan 2022

Support notes untuk versi sebelumnya

Windows Server AMIs yang dipublikasikan sebelum November 2016 menggunakan layanan EC2Config untuk memproses permintaan dan mengkonfigurasi instans.

Kecuali Anda memiliki alasan khusus untuk menggunakan layanan EC2config, atau versi sebelumnya, untuk memproses permintaan Systems ManagerSSM Agent, kami sarankan Anda mengunduh dan menginstal versi terbaru SSM Agent ke masing-masing instans Amazon Elastic Compute Cloud (Amazon EC2) atau mesin non-EC2 yang dikonfigurasi untuk Systems Manager di lingkungan hybrid dan multicloud.

Per 14 Januari 2020, Windows Server 2008 tidak lagi didukung untuk pembaruan fitur atau keamanan dari Microsoft. Legacy Amazon Machine Images (AMIs) untuk Windows Server 2008 dan 2008 R2 masih menyertakan versi 2 dari SSM Agent prainstal, tetapi Systems Manager tidak lagi secara resmi mendukung versi 2008 dan tidak lagi memperbarui agen untuk versi ini. Windows Server Selain itu, SSM Agent versi 3 mungkin tidak kompatibel dengan semua operasi pada Windows Server 2008 dan 2008 R2. Versi final yang didukung secara resmi SSM Agent untuk versi Windows Server 2008 adalah 2.3.1644.0.

Tetap SSM Agent up to date

Versi terbaru dirilis setiap kali kemampuan baru ditambahkan ke Systems Manager atau pembaruan dibuat untuk kemampuan yang ada. SSM Agent Gagal menggunakan agen versi terbaru dapat mencegah node terkelola Anda menggunakan berbagai kemampuan dan fitur Systems Manager. Untuk alasan itu, kami menyarankan Anda mengotomatiskan proses menjaga agar tetap SSM Agent up to date pada mesin Anda. Untuk informasi, lihat Mengotomatiskan pembaruan ke SSM Agent. Berlangganan halaman Catatan SSM Agent Rilis GitHub untuk mendapatkan pemberitahuan tentang SSM Agent pembaruan.

Untuk melihat detail tentang versi yang berbedaSSM Agent, lihat catatan rilis.

Topik

- Menginstal dan menghapus instalasi secara manual SSM Agent pada instans EC2 untuk Windows Server
- SSM AgentKonfigurasikan untuk menggunakan proxy untuk Windows Server instance

Menginstal dan menghapus instalasi secara manual SSM Agent pada instans EC2 untuk Windows Server

AWS Systems Manager Agent (SSM Agent) sudah diinstal sebelumnya, secara default, pada berikut Amazon Machine Images (AMIs) untuk Windows Server disediakan oleh Amazon:

- Windows Server 2008-2012 R2 AMIs yang dipublikasikan pada November 2016 atau yang lebih baru
- Windows Server2016, 2019, dan 2022

Instal SSM Agent pada instans EC2 untuk Windows Server

Jika perlu, Anda dapat mengunduh dan menginstal versi terbaru secara manual SSM Agent di Amazon Elastic Compute Cloud (Amazon EC2) Windows Server misalnya dengan menggunakan prosedur berikut. Perintah yang disediakan dalam prosedur ini juga dapat diteruskan ke instans Amazon EC2 sebagai skrip melalui data pengguna.

SSM Agentmemerlukan Windows PowerShell 3.0 atau yang lebih baru untuk menjalankan AWS Systems Manager dokumen tertentu (dokumen SSM) pada Windows Server instance (misalnya, dokumen lama`AWS-ApplyPatchBaseline`). Verifikasi bahwa instans Windows Server Anda menjalankan Windows Management Framework 3.0 atau yang lebih baru. Kerangka kerja ini termasuk Windows PowerShell. Untuk informasi lebih lanjut, lihat Windows Management Framework 3.0.

> **Note**
>
> Prosedur ini berlaku untuk menginstal atau menginstal ulang SSM Agent pada instans EC2 untuk. Windows Server Jika Anda perlu menginstal agen di server lokal atau mesin virtual (VM) agar dapat digunakan dengan Systems Manager, lihat Menginstal SSM Agent untuk lingkungan hybrid (Windows).

Untuk menginstal versi terbaru dari instans EC2 secara manual untuk SSM AgentWindows Server

1. Connect ke instans Anda dengan menggunakan Remote Desktop atau Windows PowerShell. Untuk informasi selengkapnya, lihat Connect ke instans Anda di Panduan Pengguna Amazon EC2 untuk Instans Windows.

2. Unduh versi terbaru SSM Agent ke instans Anda. Anda dapat mengunduh menggunakan PowerShell perintah atau tautan unduhan langsung.

> **Note**
>
> URL dalam langkah ini memungkinkan Anda mengunduh SSM Agent dari mana pun Wilayah AWS. Jika Anda ingin mengunduh agen dari Wilayah tertentu, gunakan URL khusus Wilayah sebagai gantinya: `https://amazon-ssm-region.s3.region.amazonaws.com/latest/windows_amd64/AmazonSSMAgentSetup.exe`
> *wilayah* mewakili pengenal untuk Wilayah AWS didukung oleh AWS Systems Manager, seperti `us-east-2` untuk Wilayah Timur AS (Ohio). Untuk daftar nilai *wilayah* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

PowerShell

Jalankan tiga PowerShell perintah berikut secara berurutan. Perintah ini memungkinkan Anda untuk mengunduh SSM Agent tanpa menyesuaikan pengaturan Internet Explorer (IE) Enhanced Security, dan kemudian menginstal agen dan menghapus file instalasi.

64-bit

```
[System.Net.ServicePointManager]::SecurityProtocol = 'TLS12'
$progressPreference = 'silentlyContinue'
Invoke-WebRequest `
    https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/windows_amd64/AmazonSSMAgentSetup.exe `
    -OutFile $env:USERPROFILE\Desktop\SSMAgent_latest.exe
```

32-bit

```
[System.Net.ServicePointManager]::SecurityProtocol = 'TLS12'
$progressPreference = 'silentlyContinue'
Invoke-WebRequest `
    https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/windows_386/AmazonSSMAgentSetup.exe `
    -OutFile $env:USERPROFILE\Desktop\SSMAgent_latest.exe
```

```
Start-Process `
    -FilePath $env:USERPROFILE\Desktop\SSMAgent_latest.exe `
    -ArgumentList "/S"
```

```
rm -Force $env:USERPROFILE\Desktop\SSMAgent_latest.exe
```

Unduh langsung

Unduh versi terbaru SSM Agent ke instans Anda dengan menggunakan tautan berikut. Jika Anda mau, perbarui URL ini dengan URL Wilayah AWS-spesifik.

https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/windows_amd64/AmazonSSM AgentSetup .exe

Jalankan `AmazonSSMAgentSetup.exe` file yang diunduh untuk menginstalSSM Agent.

3. Mulai atau restart SSM Agent dengan mengirimkan perintah berikut di PowerShell:

```
Restart-Service AmazonSSMAgent
```

Copot pemasangan SSM Agent dari instans EC2 untuk Windows Server

Untuk menghapus instalan SSM Agent dari sebuah Windows Server instance, buka Control Panel, Programs. Pilih opsi Uninstall a program. Buka menu konteks (klik kanan) untuk Amazon SSM Agent dan pilih Uninstall.

SSM AgentKonfigurasikan untuk menggunakan proxy untuk Windows Server instance

Informasi dalam topik ini berlaku untuk Windows Server instance yang dibuat pada atau setelah November 2016 yang tidak menggunakan opsi instalasi Nano. Jika Anda ingin menggunakanSession Manager, perhatikan bahwa server proxy HTTPS tidak didukung.

Jika instans Anda adalah instans Windows Server 2008-2012 R2 yang dibuat sebelum November 2016, maka EC2Config memproses permintaan AWS Systems Manager pada instans Anda. Kami sarankan Anda meningkatkan instans Anda yang sudah ada untuk menggunakan versi terbaru EC2Config. Dengan menggunakan installer EC2config terbaru, Anda menginstal AWS Systems Manager Agent () SSM Agent dengan EC2config. side-by-side side-by-sideVersi ini SSM Agent

kompatibel dengan instans yang dibuat dari Windows sebelumnya Amazon Machine Images (AMIs) dan memungkinkan Anda menggunakan fitur Systems Manager yang diterbitkan setelah November 2016. Untuk informasi tentang cara menginstal versi terbaru layanan EC2config, lihat Menginstal versi terbaru EC2config di Panduan Pengguna Amazon EC2 untuk Instans Windows. Jika Anda tidak meningkatkan ke EC2Config versi terbaru dan menggunakan EC2Config untuk memproses permintaan Systems Manager, konfigurasi pengaturan proxy untuk EC2Config. Untuk informasi tentang mengonfigurasi EC2config untuk menggunakan proxy, lihat Mengonfigurasi setelan proxy untuk layanan EC2config di Panduan Pengguna Amazon EC2 untuk Instans Windows.

> **Note**
>
> Per 14 Januari 2020, Windows Server 2008 tidak lagi didukung untuk pembaruan fitur atau keamanan dari Microsoft. Legacy Amazon Machine Images (AMIs) untuk Windows Server 2008 dan 2008 R2 masih menyertakan versi 2 dari SSM Agent prainstal, tetapi Systems Manager tidak lagi secara resmi mendukung versi 2008 dan tidak lagi memperbarui agen untuk versi ini. Windows Server Selain itu, SSM Agent versi 3 mungkin tidak kompatibel dengan semua operasi pada Windows Server 2008 dan 2008 R2. Versi final yang didukung secara resmi SSM Agent untuk versi Windows Server 2008 adalah 2.3.1644.0.

Untuk mengkonfigurasi SSM Agent untuk menggunakan proxy

1. Menggunakan Remote Desktop atau Windows PowerShell, sambungkan ke instance yang ingin Anda konfigurasikan untuk menggunakan proxy.
2. Jalankan blok perintah berikut di PowerShell. Ganti *`nama host`* dan *`port`* dengan informasi tentang proxy Anda.

```
$serviceKey = "HKLM:\SYSTEM\CurrentControlSet\Services\AmazonSSMAgent"
$keyInfo = (Get-Item -Path $serviceKey).GetValue("Environment")
$proxyVariables = @("http_proxy=hostname:port", "https_proxy=hostname:port",
 "no_proxy=169.254.169.254")

if ($keyInfo -eq $null) {
    New-ItemProperty -Path $serviceKey -Name Environment -Value $proxyVariables -
PropertyType MultiString -Force
}
else {
    Set-ItemProperty -Path $serviceKey -Name Environment -Value $proxyVariables
}
```

```
Restart-Service AmazonSSMAgent
```

Setelah menjalankan perintah sebelumnya, Anda dapat meninjau SSM Agent log untuk mengonfirmasi pengaturan proxy diterapkan. Entri di log terlihat serupa dengan yang berikut ini. Untuk informasi selengkapnya tentang SSM Agent log, lihatMelihat SSM Agent log.

```
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: 169.254.169.254
2020-02-24 15:31:54 INFO Starting Agent: amazon-ssm-agent - v2.3.871.0
2020-02-24 15:31:54 INFO OS: windows, Arch: amd64
```

Untuk mengatur ulang konfigurasi SSM Agent proxy

1. Menggunakan Remote Desktop atau Windows PowerShell, sambungkan ke instance untuk mengkonfigurasi.
2. Jika Anda terhubung menggunakan Remote Desktop, luncurkan PowerShell sebagai administrator.
3. Jalankan blok perintah berikut di PowerShell.

```
Remove-ItemProperty -Path HKLM:\SYSTEM\CurrentControlSet\Services\AmazonSSMAgent -
Name Environment
Restart-Service AmazonSSMAgent
```

SSM Agentprioritas pengaturan proxy

Saat mengonfigurasi pengaturan proxy untuk Windows Server instance aktif, penting untuk memahami pengaturan ini dievaluasi dan diterapkan ke konfigurasi agen saat dimulai. SSM Agent SSM Agent Cara Anda mengonfigurasi pengaturan proxy untuk sebuah Windows Server instans dapat menentukan apakah pengaturan lain mungkin menggantikan pengaturan yang Anda inginkan.

> **Important**
>
> SSM Agentberkomunikasi menggunakan protokol HTTPS. Untuk alasan ini, Anda harus mengonfigurasi `HTTPS proxy` parameter dengan menggunakan salah satu opsi pengaturan berikut.

SSM Agentpengaturan proxy dievaluasi dalam urutan berikut.

1. Pengaturan Registri AmazonSSMAgent (`HKLM:\SYSTEM\CurrentControlSet\Services\AmazonSSMAgent`)
2. Variabel lingkungan sistem (`http_proxy`,`https_proxy`,`no_proxy`)
3. LocalSystem variabel lingkungan akun penggunahttp_proxy,https_proxy,no_proxy)
4. Pengaturan Internet Explorer (HTTP,`secure`,`exceptions`)
5. Pengaturan proxy WinHTTP (`http=`,,`https=`) `bypass-list=`

SSM Agentpengaturan proxy dan layanan Systems Manager

Jika Anda mengonfigurasi SSM Agent untuk menggunakan proxy dan menggunakan AWS Systems Manager kemampuan, seperti Run Command danPatch Manager, yang menggunakan PowerShell atau klien Pembaruan Windows selama eksekusi mereka pada Windows Server instance, konfigurasikan pengaturan proxy tambahan. Jika tidak, operasi mungkin gagal karena pengaturan proxy yang digunakan oleh PowerShell dan klien Pembaruan Windows tidak diwarisi dari konfigurasi SSM Agent proxy.

UntukRun Command, konfigurasikan pengaturan `WinINet` proxy pada Windows Server instans Anda. Perintah `[System.Net.WebRequest]` yang diberikan adalah per sesi. Untuk menerapkan konfigurasi ini ke perintah jaringan berikutnya yang dijalankanRun Command, perintah ini harus mendahului PowerShell perintah lain dalam input plugin yang sama`aws:runPowershellScript`.

PowerShell Perintah berikut mengembalikan pengaturan `WinINet` proxy saat ini, dan menerapkan pengaturan proxy Anda ke`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
```

UntukPatch Manager, konfigurasikan pengaturan proxy seluruh sistem sehingga klien Pembaruan Windows dapat memindai dan mengunduh pembaruan. Kami menyarankan Anda menggunakan Run Command untuk menjalankan perintah berikut karena mereka berjalan di akun SYSTEM, dan pengaturannya berlaku di seluruh sistem. Perintah `netsh` berikut ini mengembalikan pengaturan proxy saat ini, dan menerapkan pengaturan proxy Anda ke sistem lokal.

```
netsh winhttp show proxy

netsh winhttp set proxy proxy-server="hostname:port" bypass-list="169.254.169.254"
```

Untuk informasi selengkapnya tentang penggunaan Run Command, lihat AWS Systems Manager Run Command.

Bekerja denganSSM Agentpada perangkat edge

Systems Manager mendukung instalasi dan menjalankanSSM Agentdi atasAWS IoT Greengrassperangkat inti,AWS IoT, dan non-AWSPerangkat IoT. ParameterSSM Agentproses instalasi dan konfigurasi untukAWS IoT Greengrassperangkat inti berbeda dariAWS IoTdan bukanAWSPerangkat IoT. Untuk informasi selengkapnya, lihat AWS Systems ManagerMenyiapkan perangkat edge.

> **Note**
>
> Untuk informasi tentang menghapus instalasiSSM Agentdari perangkat edge, lihatMenghapus instalasiAWS Systems ManagerAgendi dalamAWS IoT Greengrass Version 2Panduan Pengembang.

Memeriksa SSM Agent status dan memulai agen

Topik ini mencantumkan perintah untuk memeriksa apakah AWS Systems Manager Agent (SSM Agent) berjalan pada setiap sistem operasi yang didukung. Hal ini juga menyediakan perintah untuk memulai agen jika tidak berjalan.

Sistem operasi	Perintah untuk memeriksa SSM Agent status	Perintah untuk memulai SSM Agent
Amazon Linux 1	`sudo status amazon-ssm-agent`	`sudo start amazon-ssm-agent`
Amazon Linux 2 dan 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 dan CentOS 8.x	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`
Debian Server8, 9, dan 10	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`
macOS	Periksa file log agen di `/var/log/amazon/ssm/amazon-ssm-agent.log`	`sudo launchctl load -w /Library/LaunchDaemons/com.amazon.aws.ssm.plist` `sudo launchctl start com.amazon.aws.ssm`
Oracle Linux	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent`

Sistem operasi	Perintah untuk memeriksa SSM Agent status	Perintah untuk memulai 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 dan 8.x	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`
SUSE Linux Enterprise Server (SLES)	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`
Ubuntu Server14.04 (semua) dan 16.04 (32-bit)	`sudo status amazon-ssm-agent`	`sudo start amazon-ssm-agent`
Ubuntu Server16.04 instance 64-bit (instalasi paket deb)	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`
Ubuntu Server16.04, 18.04, dan 20.04 LTS, 20.10 STR 64-bit, dan 22.04 LTS (instalasi paket Snap)	`sudo systemctl status snap.amazon-ssm-agent.amazon-ssm-agent.service`	`sudo snap start amazon-ssm-agent`
Windows Server	Jalankan di PowerShell: `Get-Service AmazonSSMAgent`	Jalankan dalam mode PowerShell Administrator: `Start-Service AmazonSSMAgent`

Info lebih lanjut

- Bekerja denganSSM Agent instans EC2 untuk Linux
- Bekerja dengan SSM Agent instans EC2 untuk Windows Server
- Memeriksa nomor SSM Agent versi

Memeriksa nomor SSM Agent versi

AWS Systems Manager Fungsionalitas tertentu memiliki prasyarat yang menyertakan versi Systems Manager Agent (SSM Agent) minimum yang diinstal pada node terkelola Anda. Anda bisa mendapatkan SSM Agent versi yang saat ini diinstal pada node terkelola menggunakan konsol Systems Manager, atau dengan masuk ke node terkelola Anda.

Prosedur berikut menjelaskan cara mendapatkan SSM Agent versi yang saat ini diinstal pada node terkelola Anda.

Untuk memeriksa nomor versi yang SSM Agent diinstal pada node terkelola

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Di kolom SSM Agentversi, perhatikan nomor versi Agen.

Untuk mendapatkan SSM Agent versi yang saat ini diinstal dari dalam sistem operasi

Pilih dari tab berikut untuk mendapatkan SSM Agent versi yang saat ini diinstal dari dalam sistem operasi.

Amazon Linux 1, Amazon Linux 2, and Amazon Linux 2023

> **Note**
>
> Perintah ini bervariasi tergantung pada pengelola paket untuk sistem operasi Anda.

1. Masuk ke node terkelola Anda.
2. Jalankan perintah berikut.

```
yum info amazon-ssm-agent
```

Perintah ini mengembalikan output yang serupa dengan yang berikut ini.

```
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Installed Packages
Name        : amazon-ssm-agent
Arch        : x86_64
Version     : 3.0.655.0
```

CentOS

1. Masuk ke node terkelola Anda.
2. Jalankan perintah berikut untuk CentOS 6 dan 7.

```
yum info amazon-ssm-agent
```

Perintah ini mengembalikan output yang serupa dengan yang berikut ini.

```
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Installed Packages
Name        : amazon-ssm-agent
Arch        : x86_64
Version     : 3.0.655.0
```

Debian Server

1. Masuk ke node terkelola Anda.
2. Jalankan perintah berikut.

```
apt list amazon-ssm-agent
```

Perintah ini mengembalikan output yang serupa dengan yang berikut.

```
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. Masuk ke node terkelola Anda.
2. Jalankan perintah berikut.

```
pkgutil --pkg-info com.amazon.aws.ssm
```

RHEL

1. Masuk ke node terkelola Anda.
2. Jalankan perintah berikut untuk RHEL 6, 7, 8, dan 9.

```
yum info amazon-ssm-agent
```

Perintah ini mengembalikan output yang serupa dengan yang berikut ini.

```
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Installed Packages
Name        : amazon-ssm-agent
Arch        : x86_64
Version     : 3.0.655.0
```

Jalankan perintah berikut untuk utilitas paket DNF.

```
dnf info amazon-ssm-agent
```

SLES

1. Masuk ke node terkelola Anda.

2. Jalankan perintah berikut untuk SLES 12 dan 15.

```
zypper info amazon-ssm-agent
```

Perintah ini mengembalikan output yang serupa dengan yang berikut ini.

```
Loading repository data...
Reading installed packages...
Information for package amazon-ssm-agent:
----------------------------------------
Repository : @System
Name : amazon-ssm-agent
Version : 3.0.655.0-1
```

Ubuntu Server

> **Note**
>
> Untuk memeriksa apakah instans Ubuntu Server 16.04 Anda menggunakan paket deb atau Snap, lihat. Menginstal secara manual SSM Agent pada Ubuntu Server instance

1. Masuk ke node terkelola Anda.
2. Jalankan perintah berikut untuk Ubuntu Server 16.04 dan 14.04 64-bit (dengan paket installer deb).

```
apt list amazon-ssm-agent
```

Perintah ini mengembalikan output yang serupa dengan yang berikut ini.

```
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
```

Jalankan perintah berikut untuk instance 64-bit Ubuntu Server 22.04 LTS, 20.10 STR dan 20.04, 18.04, dan 16.04 LTS 64-bit (dengan paket Snap).

```
sudo snap list amazon-ssm-agent
```

Perintah ini mengembalikan output yang serupa dengan yang berikut ini.

```
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. Masuk ke node terkelola Anda.
2. Jalankan PowerShell perintah berikut.

```
& "C:\Program Files\Amazon\SSM\amazon-ssm-agent.exe" -version
```

Perintah ini mengembalikan output yang serupa dengan yang berikut ini.

```
SSM Agent version: 3.1.804.0
```

Sebaiknya gunakan versi terbaru SSM Agent sehingga Anda bisa mendapatkan keuntungan dari kemampuan baru atau yang diperbarui. Untuk memastikan instans terkelola Anda selalu menjalankan sebagian besar up-to-date versiSSM Agent, Anda dapat mengotomatiskan proses pembaruan. SSM Agent Lihat informasi yang lebih lengkap di Mengotomatiskan pembaruan ke SSM Agent.

Melihat SSM Agent log

AWS Systems Manager Agent (SSM Agent) menulis informasi tentang eksekusi, perintah, tindakan terjadwal, kesalahan, dan status kesehatan untuk mencatat file di setiap node terkelola. Anda dapat melihat file log dengan menghubungkan secara manual ke node terkelola, atau Anda dapat secara otomatis mengirim log ke Amazon CloudWatch Logs. Untuk informasi selengkapnya tentang mengirim CloudWatch log ke Log, lihatPemantauan AWS Systems Manager.

Anda dapat melihat SSM Agent log pada node terkelola di lokasi berikut.

Linux and macOS

```
/var/log/amazon/ssm/
```

Windows

```
%PROGRAMDATA%\Amazon\SSM\Logs\
```

Untuk node yang dikelola Linux, `stdout` file SSM Agent `stderr` dan ditulis ke direktori berikut:`/var/lib/amazon/ssm/`.

Untuk node yang dikelola Windows, `stdout` file SSM Agent `stderr` dan ditulis ke direktori berikut:`%PROGRAMDATA%\Amazon\SSM\InstanceData\`.

Untuk informasi tentang mengizinkan logging SSM Agent debug, lihatMengizinkan SSM Agent logging debug.

Untuk informasi selengkapnya tentang `cihub/seelog` konfigurasi, lihat Wiki Seelog di. GitHub
Untuk contoh `cihub/seelog` konfigurasi, lihat repositori contoh cihub/seelog di. GitHub

Mengizinkan SSM Agent logging debug

Gunakan prosedur berikut untuk mengizinkan logging SSM Agent debug pada node terkelola Anda.

Linux and macOS

Untuk memungkinkan logging SSM Agent debug di Linux dan node macOS terkelola

1. Baik gunakanSession Manager, kemampuan AWS Systems Manager, untuk terhubung ke node terkelola tempat Anda ingin mengizinkan logging debug, atau masuk ke node terkelola. Untuk informasi selengkapnya, lihat Bekerja dengan Session Manager.
2. Temukan file seelog.xml.template.

   Linux:

   Pada sebagian besar jenis node yang dikelola Linux, file tersebut terletak di direktori`/etc/amazon/ssm/seelog.xml.template`.

   Pada Ubuntu Server 20.10 STR & 20.04, 18.04, dan 16.04 LTS, file tersebut terletak di direktori. `/snap/amazon-ssm-agent/current/seelog.xml.template` Salin file ini

dari direktori `/snap/amazon-ssm-agent/current/` ke direktori `/etc/amazon/ssm/` sebelum membuat perubahan apapun.

macOS:

Pada tipe instans macOS, file terletak di direktori `/opt/aws/ssm/seelog.xml.template`.

3. Ubah nama file dari `seelog.xml.template` ke `seelog.xml`.

> **Note**
>
> Pada Ubuntu Server 20.10 STR & 20.04, 18.04, dan 16.04 LTS, file `seelog.xml` harus dibuat dalam direktori. `/etc/amazon/ssm/` Anda dapat membuat direktori dan file ini dengan menjalankan perintah berikut.
>
> ```
> sudo mkdir -p /etc/amazon/ssm
> ```
>
> ```
> sudo cp -p /snap/amazon-ssm-agent/current/seelog.xml.template /etc/
> amazon/ssm/seelog.xml
> ```

4. Edit file `seelog.xml` untuk mengubah perilaku pencatatan default. Mengubah nilai minlevel dari info ke debug, seperti yang ditunjukkan dalam contoh berikut.

```
<seelog type="adaptive" mininterval="2000000" maxinterval="100000000"
critmsgcount="500" minlevel="debug">
```

5. (Opsional) Mulai ulang SSM Agent menggunakan perintah berikut.

Linux:

```
sudo service amazon-ssm-agent restart
```

macOS:

```
sudo /opt/aws/ssm/bin/amazon-ssm-agent restart
```

Windows

Untuk mengizinkan logging SSM Agent debug pada node Windows Server terkelola

1. Gunakan Session Manager untuk terhubung ke node terkelola tempat Anda ingin mengizinkan logging debug, atau masuk ke node terkelola. Untuk informasi selengkapnya, lihat Bekerja dengan Session Manager.
2. Membuat salinan file seelog.xml.template. Ubah nama salinan menjadi seelog.xml. file terletak di direktori berikut.

   ```
   %PROGRAMFILES%\Amazon\SSM\seelog.xml.template
   ```
3. Edit file `seelog.xml` untuk mengubah perilaku pencatatan default. Mengubah nilai minlevel dari info ke debug, seperti yang ditunjukkan dalam contoh berikut.

   ```
   <seelog type="adaptive" mininterval="2000000" maxinterval="100000000"
   critmsgcount="500" minlevel="debug">
   ```
4. Temukan entri berikut.

   ```
   filename="{{LOCALAPPDATA}}\Amazon\SSM\Logs\{{EXECUTABLENAME}}.log"
   ```

   Ubah entri ini untuk menggunakan jalur berikut.

   ```
   filename="C:\ProgramData\Amazon\SSM\Logs\amazon-ssm-agent.log"
   ```
5. Temukan entri berikut.

   ```
   filename="{{LOCALAPPDATA}}\Amazon\SSM\Logs\errors.log"
   ```

   Ubah entri ini untuk menggunakan jalur berikut.

   ```
   filename="C:\ProgramData\Amazon\SSM\Logs\errors.log"
   ```
6. Mulai ulang SSM Agent menggunakan PowerShell perintah berikut dalam mode Administrator.

   ```
   Restart-Service AmazonSSMAgent
   ```

Membatasi akses ke perintah tingkat root melaluiSSM Agent

AWS Systems ManagerAgen (SSM Agent) berjalan pada instans Amazon Elastic Compute Cloud (Amazon EC2) dan jenis alat berat lainnya dihybrid dan multicloudlingkungan yang menggunakan izin root (Linux) atau izin SISTEM (Windows Server). Karena ini adalah tingkat izin akses sistem tertinggi, entitas tepercaya apa pun yang telah diberikan izin untuk mengirim perintahSSM Agentmemiliki izin root atau SYSTEM. (DalamAWS, entitas tepercaya yang dapat melakukan tindakan dan mengakses sumber daya diAWSdisebut autama. Seorang kepala sekolah bisa menjadiPengguna root akun AWS, pengguna, atau peran.)

Tingkat akses ini diperlukan bagi kepala sekolah untuk mengirim perintah Manajer Sistem yang berwenang keSSM Agent, tetapi juga memungkinkan prinsipal untuk menjalankan kode berbahaya dengan mengeksploitasi potensi kerentanan diSSM Agent.

Secara khusus, izin untuk menjalankan perintahSendCommanddanStartSessionharus dibatasi dengan hati-hati. Langkah pertama yang baik adalah memberikan izin untuk setiap perintah untuk hanya memilih principal di organisasi Anda. Namun, kami menyarankan untuk memperketat postur keamanan Anda lebih jauh dengan membatasi node terkelola mana yang dapat menjalankan perintah ini. Ini dapat dilakukan dalam kebijakan IAM yang diberikan kepada kepala sekolah. Dalam kebijakan IAM, Anda dapat menyertakan kondisi yang membatasi pengguna untuk menjalankan perintah hanya pada node terkelola yang ditandai dengan tag tertentu atau kombinasi tag.

Misalnya, Anda memiliki dua armada server, satu untuk pengujian, satu untuk produksi. Dalam kebijakan IAM yang diterapkan untuk insinyur junior, Anda menentukan bahwa mereka dapat menjalankan perintah hanya pada instans yang ditandai dengan `ssm:resourceTag/testServer`. Namun, untuk grup insinyur utama yang lebih kecil, yang seharusnya memiliki akses ke semua instans, Anda memberikan akses ke instans yang ditandai dengan `ssm:resourceTag/testServer` dan `ssm:resourceTag/productionServer`.

Menggunakan pendekatan ini, jika insinyur junior mencoba untuk menjalankan perintah pada instans produksi, akses mereka akan ditolak karena kebijakan IAM mereka yang ditugaskan tidak menyediakan akses eksplisit ke instans yang ditandai dengan `ssm:resourceTag/productionServer`.

Untuk informasi selengkapnya dan contoh, lihat topik berikut ini:

- MembatasiRun Command akses akses berdasarkan tag
- Batasi akses sesi berdasarkan tag instance

Mengotomatiskan pembaruan ke SSM Agent

AWS merilis versi baru AWS Systems Manager Agent (SSM Agent) ketika kita menambahkan atau memperbarui kemampuan Systems Manager. Jika node terkelola Anda menggunakan versi agen yang lebih lama, maka Anda tidak dapat menggunakan kemampuan baru atau memanfaatkan kemampuan yang diperbarui. Untuk alasan ini, kami menyarankan Anda mengotomatiskan proses pembaruan SSM Agent pada node terkelola Anda menggunakan salah satu metode berikut.

Pembaruan agen pada sistem operasi Bottlerocket

SSM Agentpada sistem operasi Bottlerocket tidak dapat diperbarui menggunakan dokumen Command Systems Manager. `AWS-UpdateSSMAgent` Pembaruan dikelola dalam wadah kontrol Bottlerocket. Untuk informasi selengkapnya, lihat Infrastruktur pembaruan Kontainer Kontrol Bottlerocket dan Bottlerocket. GitHub

macOSpersyaratan versi

Jika sebuah instance menjalankan macOS versi 11.0 (Big Sur) atau yang lebih baru, instance harus memiliki SSM Agent versi 3.1.941.0 atau lebih tinggi untuk menjalankan dokumen. AWS-UpdateSSMAgent Jika instance menjalankan versi SSM Agent rilis sebelum 3.1.941.0, perbarui SSM Agent untuk menjalankan AWS-UpdateSSMAgent by running dan perintah. `brew update` `brew upgrade amazon-ssm-agent`

Metode	Detail
Pembaruan otomatis sekali klik pada semua node terkelola (Disarankan)	Anda dapat mengonfigurasi semua node terkelola di Anda Akun AWS untuk secara otomatis memeriksa dan mengunduh versi baruSSM Agent. Untuk melakukan ini, pilih Pembaruan otomatis SSM Agent pada tab Pengaturan diFleet Manager, seperti yang dijelaskan nanti dalam topik ini.
Pembaruan global atau selektif	Anda dapat menggunakanState Manager, kemampuan AWS Systems Manager, untuk membuat asosiasi yang secara otomatis mengunduh dan menginstal SSM Agent pada node terkelola Anda. Jika Anda ingin

Metode	Detail
	membatasi gangguan beban kerja Anda, Anda dapat membuat jendela pemeliharaan Systems Manager untuk melakukan instalasi selama periode waktu yang ditentukan. Kedua metode memungkinkan Anda untuk membuat konfigurasi pembaruan global untuk semua node terkelola Anda atau secara selektif memilih instance mana yang diperbarui. Untuk informasi tentang membuat State Manager asosiasi, lihatWalkthrough: Perbarui secara otomatis (SSM AgentCLI). Untuk informasi tentang menggunakan jendela pemeliharaan, lihat Panduan: Membuat jendela pemeliharaan untuk memperbaruiSSM Agent (AWS CLI) danPanduan: Membuat jendela pemeliharaan untuk memperbarui secara otomatisSSM Agent (konsol).
Pembaruan global atau selektif untuk lingkungan baru	Jika Anda memulai dengan Systems Manager, kami sarankan Anda menggunakan Agen Update Systems Manager (SSM) setiap dua mingguQuick Setup, dengan kemampuan. AWS Systems ManagerQuick Setupmemungkinkan Anda membuat konfigurasi pembaruan global untuk semua node terkelola Anda atau secara selektif memilih node terkelola mana yang diperbarui. Untuk informasi selengkapnya, lihat Manajemen host Amazon EC2.

Jika Anda lebih suka memperbarui SSM Agent pada node terkelola secara manual, Anda dapat berlangganan notifikasi yang AWS diterbitkan saat versi baru agen dirilis. Untuk informasi, lihat Berlangganan notifikasi SSM Agent. Setelah berlangganan notifikasi, Anda dapat menggunakan Run

Command untuk memperbarui satu atau lebih node terkelola secara manual dengan versi terbaru. Untuk informasi selengkapnya, lihat Memperbarui SSM Agent penggunaan Run Command.

Memperbarui secara otomatis SSM Agent

Anda dapat mengonfigurasi Systems Manager untuk memperbarui secara otomatis SSM Agent pada semua node terkelola berbasis Linux dan berbasis Windows di Anda. Akun AWS Jika Anda mengaktifkan opsi ini, maka Systems Manager secara otomatis memeriksa setiap dua minggu untuk versi baru agen. Jika ada versi baru, maka Systems Manager secara otomatis memperbarui agen ke versi terbaru yang dirilis menggunakan dokumen SSM `AWS-UpdateSSMAgent`. Kami mendorong Anda untuk memilih opsi ini untuk memastikan bahwa node terkelola Anda selalu menjalankan sebagian besar up-to-date versiSSM Agent.

> **Note**
>
> Jika Anda menggunakan `yum` perintah untuk memperbarui SSM Agent pada node terkelola setelah agen diinstal atau diperbarui menggunakan dokumen SSM`AWS-UpdateSSMAgent`, Anda mungkin melihat pesan berikut: “Peringatan: RPMDB diubah di luar yum.” Pesan ini diharapkan dan dapat diabaikan dengan aman.

Untuk memperbarui secara otomatis SSM Agent

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tab Pengaturan.
4. Di area pembaruan otomatis Agen, pilih Pembaruan otomatis SSM Agent.

Untuk mengubah versi pembaruan armada SSM Agent Anda, pilih Edit di bawah Pembaruan otomatis agen di tab Pengaturan. Kemudian masukkan nomor versi yang ingin SSM Agent Anda perbarui di Versi di bawah Parameter. Jika tidak ditentukan, agen memperbarui ke versi terbaru.

Untuk menghentikan penerapan versi terbaru secara otomatis SSM Agent ke semua node terkelola di akun Anda, pilih Hapus di bawah Pembaruan otomatis Agen pada tab Pengaturan. Tindakan ini menghapus State Manager asosiasi yang secara otomatis memperbarui SSM Agent pada node terkelola Anda.

Berlangganan notifikasi SSM Agent

Amazon Simple Notification Service (Amazon SNS) dapat memberi tahu Anda saat versi baru Agen SSM Agent () AWS Systems Manager dirilis. Gunakan prosedur berikut untuk berlangganan notifikasi ini.

> **Tip**
>
> Anda juga dapat berlangganan notifikasi dengan menonton halaman Catatan SSM Agent Rilis diGitHub.

Untuk berlanggananSSM Agent notifikasi

1. Buka konsol Amazon SNS di https://console.aws.amazon.com/sns/v3/home.
2. Dari pemilih Wilayah di bar navigasi, pilih US East (N. Virginia), jika belum dipilih. Anda harus memilih ini Wilayah AWS karena notifikasi Amazon SNS untuk SSM Agent langganan Anda dibuat dari Wilayah ini saja.
3. Di panel navigasi, pilih Langganan.
4. Pilih Buat langganan.
5. Untuk Buat langganan, lakukan hal berikut:
   a. Untuk Topik ARN, gunakan Amazon Resource Name (ARN) berikut:

      ```
      arn:aws:sns:us-east-1:720620558202:SSM-Agent-Update
      ```

   b. Untuk Protokol, pilih `Email` atau `SMS`.
   c. Untuk Endpoint, tergantung pada apakah Anda memilih `Email` atau `SMS` pada langkah sebelumnya, masukkan alamat email atau kode area dan nomor untuk menerima pemberitahuan.
   d. Pilih Buat langganan.

6. Jika Anda memilih `Email`, Anda akan menerima email yang meminta Anda untuk mengkonfirmasi langganan Anda. Buka pesan dan ikuti petunjuk untuk menyelesaikan langganan Anda.

Setiap kali versi baru dirilis, kami mengirim pemberitahuan ke pelanggan. SSM Agent Jika Anda tidak ingin lagi menerima notifikasi ini, gunakan prosedur berikut untuk berhenti berlangganan.

Untuk berhenti berlangganan SSM Agent notifikasi

1. Buka konsol Amazon SNS.
2. Di panel navigasi, pilih Langganan.
3. Pilih langganan, lalu pilih Hapus. Saat diminta konfirmasi, pilih Hapus.

SSM Agent komunikasi dengan bucket S3 AWS terkelola

Dalam menjalankan berbagai operasi Systems Manager, AWS Systems Manager Agent (SSM Agent) mengakses sejumlah bucket Amazon Simple Storage Service (Amazon S3). Bucket S3 ini dapat diakses publik, dan secara default, SSM Agent terhubung ke mereka menggunakan panggilan. `HTTP`

Namun, jika Anda menggunakan titik akhir virtual private cloud (VPC) dalam operasi Systems Manager, Anda harus memberikan izin eksplisit di profil instans Amazon Elastic Compute Cloud (Amazon EC2) untuk Systems Manager, atau dalam peran layanan untuk mesin non-EC2 di lingkungan hybrid dan multicloud. Jika tidak, sumber daya Anda tidak dapat mengakses bucket publik ini.

Untuk memberikan akses node terkelola ke bucket ini saat Anda menggunakan titik akhir VPC, Anda membuat kebijakan izin Amazon S3 khusus, lalu melampirkannya ke profil instans Anda (untuk instans EC2) atau peran layanan Anda (untuk node terkelola non-EC2).

Untuk informasi tentang penggunaan titik akhir virtual private cloud (VPC) dalam operasi Systems Manager, lihat Membuat titik akhir VPC.

> **Note**
>
> Izin ini hanya menyediakan akses ke bucket AWS terkelola yang diperlukan oleh. SSM Agent Mereka tidak memberikan izin yang diperlukan untuk operasi Amazon S3 lainnya. Mereka juga tidak memberikan izin untuk bucket S3 Anda sendiri.

Untuk informasi selengkapnya, lihat topik berikut:

- Konfigurasikan izin instans untuk Systems Manager
- Buat peran layanan IAM untuk lingkungan hybrid

Daftar Isi

Izin bucket yang diperlukan

Tabel berikut menjelaskan setiap bucket S3 yang SSM Agent mungkin perlu diakses untuk operasi Systems Manager.

> **Note**
>
> *wilayah* mewakili pengenal untuk Wilayah AWS didukung oleh AWS Systems Manager, seperti us-east-2 untuk Wilayah Timur AS (Ohio). Untuk daftar nilai *wilayah* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

Izin Amazon S3 diperlukan oleh SSM Agent

Bucket S3 ARN	Deskripsi
arn:aws:s3:::aws-windows-downloads- *region*/*	Diperlukan untuk beberapa dokumen SSM yang hanya mendukung sistem Windows Server operasi, ditambah beberapa untuk dukungan lintas platform, seperti. AWSEC2-ConfigureSTIG
arn:aws:s3:::amazon-ssm- *region*/*	Diperlukan untuk memperbarui SSM Agent instalasi. Ember ini berisi paket SSM Agent instalasi, dan manifes instalasi yang direferen sikan oleh AWS-UpdateSSMAgent dokumen

Bucket S3 ARN	Deskripsi
	dan plugin. Jika izin ini tidak diberikan, maka akan SSM Agent membuat panggilan HTTP untuk mengunduh pembaruan.
`arn:aws:s3:::amazon-ssm-packages-` *region*`/*`	Diperlukan untuk menggunakan versi SSM Agent sebelum 2.2.45.0 untuk menjalankan dokumen SSM. `AWS-ConfigureAWSPackage`
`arn:aws:s3::: ` *region*`-birdwatcher-prod/*`	Menyediakan akses ke layanan distribusi yang digunakan oleh versi 2.2.45.0 dan yang lebih baru. SSM Agent Layanan ini digunakan untuk menjalankan dokumen `AWS-ConfigureAWSPackage` . Izin ini diperlukan untuk semua Wilayah AWS kecuali Wilayah Afrika (Cape Town) (af-selatan-1) dan Wilayah Eropa (Milan) (eu-selatan-1).
`arn:aws:s3:::aws-ssm-distributor-file-` *region*`/*`	Menyediakan akses ke layanan distribusi yang digunakan oleh versi 2.2.45.0 dan yang lebih baru. SSM Agent Layanan ini digunakan untuk menjalankan dokumen SSM `AWS-ConfigureAWSPackage` . Izin ini diperlukan hanya untuk Wilayah Africa (Cape Town) (af-south-1) dan Wilayah Europe (Milan) (eu-south-1).
`arn:aws:s3:::aws-ssm-document-attachments-` *region*`/*`	Menyediakan akses ke bucket S3 yang berisi paket untukDistributor, kemampuan AWS Systems Manager, yang dimiliki oleh AWS.

Bucket S3 ARN	Deskripsi
`arn:aws:s3:::patch-baseline-snapshot-` *region*`/*`	Menyediakan akses ke bucket S3 yang berisi snapshot dasar patch. Hal ini diperlukan jika Anda menggunakan salah satu dokumen SSM berikut: • `AWS-RunPatchBaseline` • `AWS-RunPatchBaselineAssociation` • `AWS-RunPatchBaselineWithHooks` • `AWS-ApplyPatchBaseline` (dokumen SSM warisan) **Note** Hanya di Wilayah Middle East (Bahrain) (me-south-1), bucket S3 ini menggunakan konvensi penamaan yang berbeda. Untuk Wilayah AWS saja, gunakan bucket berikut sebagai gantinya. • `patch-baseline-snapshot-me-south-1-uduvl7q8` Di Wilayah Afrika (Cape Town) (af-selatan-1) saja, bucket S3 ini menggunakan konvensi penamaan yang berbeda. Untuk Wilayah AWS saja, gunakan bucket berikut sebagai gantinya. • `patch-baseline-snapshot-af-south-1-tbxdb5b9`

Bucket S3 ARN	Deskripsi
Untuk Linux dan node Windows Server terkelola: `arn:aws:s3:::aws-ssm-` *region*`/*` Untuk instans Amazon EC2 untuk macOS: `arn:aws:s3:::aws-patchmanager-macos-` *region*`/*`	Menyediakan akses ke bucket S3 yang berisi modul yang diperlukan untuk digunakan dengan dokumen Systems Manager tertentu (dokumen SSM). Sebagai contoh: • `arn:aws:s3:::aws-ssm-us-east-2/*` • `aws-patchmanager-macos-us-east-2/*` Pengecualian Nama bucket S3 dalam beberapa Wilayah AWS menggunakan konvensi penamaan yang diperluas, seperti yang ditunjukkan oleh ARN mereka. Untuk Wilayah ini, gunakan ARN berikut sebagai gantinya: • Wilayah Middle East (Bahrain) (me-south-1)): `aws-patch-manager-me-south-1-a53fc9dce` • Wilayah Africa (Cape Town) (af-south-1): `aws-patch-manager-af-south-1-bdd5f65a9` • Wilayah Europe (Milan) (eu-south-1): `aws-patch-manager-eu-south-1-c52f3f594` • Wilayah Asia Pacific (Osaka) (ap-northeast-3): `aws-patch-manager-ap-northeast-3-67373598a` Dokumen SSM

Bucket S3 ARN	Deskripsi
	Berikut ini adalah beberapa dokumen SSM yang umum digunakan disimpan dalam bucket ini. Dalam `arn:aws:s3:::aws-ssm-` *region*`/`: • `AWS-RunPatchBaseline` • `AWS-RunPatchBaselineAssociation` • `AWS-RunPatchBaselineWithHooks` • `AWS-InstanceRebootWithHooks` • `AWS-ConfigureWindowsUpdate` • `AWS-FindWindowsUpdates` • `AWS-PatchAsgInstance` • `AWS-PatchInstanceWithRollback` • `AWS-UpdateSSMAgent` • `AWS-UpdateEC2Config` Dalam `arn:aws:s3:::aws-patchmanager-macos-` *region*`/`: • `AWS-RunPatchBaseline` • `AWS-RunPatchBaselineAssociation` • `AWS-RunPatchBaselineWithHooks` • `AWS-InstanceRebootWithHooks` • `AWS-PatchAsgInstance` • `AWS-PatchInstanceWithRollback`

Contoh

Contoh berikut menggambarkan cara menyediakan akses ke bucket S3 yang diperlukan untuk operasi Systems Manager di Wilayah US East (Ohio) Region (us-east-2). Dalam kebanyakan kasus, Anda perlu memberikan izin ini secara eksplisit dalam profil instans atau peran layanan hanya saat menggunakan titik akhir VPC.

> **Important**
>
> Kami menyarankan Anda untuk menghindari menggunakan karakter wildcard (*) di tempat Wilayah tertentu dalam kebijakan ini. Misalnya, gunakan `arn:aws:s3:::aws-ssm-us-east-2/*` dan jangan gunakan `arn:aws:s3:::aws-ssm-*/*`. Menggunakan wildcard dapat menyediakan akses ke bucket S3 yang tidak ingin Anda berikan akses. Jika Anda ingin menggunakan profil instans untuk lebih dari satu Wilayah, kami sarankan Anda mengulangi blok `Statement` pertama untuk setiap Wilayah.

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

Pemecahan Masalah SSM Agent

Jika Anda mengalami masalah saat menjalankan operasi pada node terkelola, mungkin ada masalah dengan AWS Systems Manager Agent (SSM Agent). Gunakan informasi berikut untuk membantu Anda melihat file SSM Agent log dan memecahkan masalah agen.

Topik

SSM Agentkedaluwarsa

Versi terbaru dirilis setiap kali kemampuan baru ditambahkan ke Systems Manager atau pembaruan dibuat untuk kemampuan yang ada. SSM Agent Gagal menggunakan agen versi terbaru dapat mencegah node terkelola Anda menggunakan berbagai kemampuan dan fitur Systems Manager. Untuk alasan itu, kami menyarankan Anda mengotomatiskan proses menjaga agar tetap SSM Agent up to date pada mesin Anda. Untuk informasi, lihat Mengotomatiskan pembaruan ke SSM Agent. Berlangganan halaman Catatan SSM Agent Rilis GitHub untuk mendapatkan pemberitahuan tentang SSM Agent pembaruan.

Memecahkan masalah menggunakan file log SSM Agent

SSM Agentlog informasi dalam file-file berikut. Informasi dalam file ini juga dapat membantu Anda memecahkan masalah. Untuk informasi selengkapnya tentang file SSM Agent log, termasuk cara mengaktifkan logging debug, lihatMelihat SSM Agent log.

> **Note**
>
> Jika Anda memilih untuk melihat log ini menggunakan Windows File Explorer, pastikan untuk mengizinkan tampilan file tersembunyi dan file sistem dalam Opsi Folder.

Di Windows

- `%PROGRAMDATA%\Amazon\SSM\Logs\amazon-ssm-agent.log`
- `%PROGRAMDATA%\Amazon\SSM\Logs\errors.log`

Di Linux dan macOS

- `/var/log/amazon/ssm/amazon-ssm-agent.log`
- `/var/log/amazon/ssm/errors.log`

Untuk node terkelola Linux, Anda mungkin menemukan informasi lebih lanjut dalam `messages` file yang ditulis ke direktori berikut: `/var/log`

Untuk informasi tambahan tentang pemecahan masalah menggunakan log agen, lihat Bagaimana cara menggunakan SSM Agent log untuk memecahkan masalah SSM Agent dalam instance terkelola saya? di Pusat Pengetahuan AWS RE: Post.

File log agen tidak berputar (Windows)

Jika Anda menentukan rotasi file log berbasis tanggal dalam file seelog.xml (pada node Windows Server terkelola) dan log tidak berputar, tentukan parameternya`fullname=true`. Berikut adalah contoh dari file konfigurasi seelog.xml dengan parameter `fullname=true` yang ditentukan.

```
<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>
```

Tidak dapat terhubung ke titik akhir SSM

SSM Agentharus mengizinkan lalu lintas keluar HTTPS (port 443) ke titik akhir berikut:

- `ssm.region.amazonaws.com`
- `ssmmessages.region.amazonaws.com`
- `ec2messages.region.amazonaws.com`

> **Note**
>
> *wilayah* mewakili pengenal untuk Wilayah AWS didukung oleh AWS Systems Manager, seperti `us-east-2` untuk Wilayah Timur AS (Ohio). Untuk daftar nilai *wilayah* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

SSM Agenttidak akan berfungsi jika tidak dapat berkomunikasi dengan titik akhir sebelumnya, bahkan jika Anda menggunakan AWS provided Amazon Machine Images (AMIs) seperti Amazon Linux 2 atau Amazon Linux 2023. Konfigurasi jaringan Anda harus memiliki akses internet terbuka atau Anda harus memiliki titik akhir virtual private cloud (VPC) khusus yang dikonfigurasi. Jika Anda tidak berencana untuk membuat VPC endpoint khusus, periksa gateway internet atau gateway NAT Anda. Untuk informasi lebih lanjut tentang cara mengelola VPC endpoint, lihat Langkah 2: Buat titik akhir VPC.

Gunakan **`ssm-cli`** untuk memecahkan masalah ketersediaan node terkelola

Dimulai dengan SSM Agent versi 3.1.501.0, Anda dapat menggunakan `ssm-cli` untuk menentukan apakah node terkelola memenuhi persyaratan utama yang akan dikelola oleh Systems Manager, dan

muncul dalam daftar node terkelola di. Fleet Manager `ssm-cli`Ini adalah alat baris perintah mandiri yang termasuk dalam SSM Agent instalasi. Perintah yang telah dikonfigurasi sebelumnya disertakan yang mengumpulkan informasi yang diperlukan untuk membantu Anda mendiagnosis mengapa instans Amazon EC2 atau mesin non-EC2 yang telah Anda konfirmasi berjalan tidak disertakan dalam daftar node terkelola di Systems Manager. Perintah ini dijalankan ketika Anda menentukan `get-diagnostics` opsi.

Untuk informasi selengkapnya, lihat Memecahkan masalah ketersediaan node terkelola menggunakan `ssm-cli`.

AWS Systems Manager Quick Setup

GunakanQuick Setup, kemampuanAWS Systems Manager, untuk mengonfigurasi layanan dan fitur Amazon Web Services yang sering digunakan dengan cepat dengan praktik terbaik yang direkomendasikan. Quick Setupmenyederhanakan pengaturan layanan, termasuk Systems Manager, dengan mengotomatiskan tugas umum atau yang direkomendasikan. Tugas-tugas ini termasuk, misalnya, membuat peran profil instans (IAM) AWS Identity and Access Management dan mengatur praktik terbaik operasional, seperti pemindaian patch berkala dan pengumpulan inventaris. Tidak ada biaya untuk menggunakan Quick Setup. Namun, biaya dapat dikeluarkan berdasarkan jenis layanan yang Anda tetapkan dan batas penggunaan tanpa biaya untuk layanan yang digunakan untuk mengatur layanan Anda. Untuk memulaiQuick Setup, buka konsol Systems Manager. Di panel navigasi, pilih Quick Setup.

> **Note**
>
> Jika Anda diarahkan Quick Setup untuk membantu mengonfigurasi instans agar dikelola oleh Systems Manager, selesaikan prosedurnya. Manajemen host Amazon EC2

Apa manfaatnyaQuick Setup?

Manfaat Quick Setup antara lain sebagai berikut:

- Sederhanakan konfigurasi layanan dan fitur

  Quick Setupmemandu Anda melalui konfigurasi praktik terbaik operasional dan secara otomatis menerapkan konfigurasi tersebut. Quick SetupDasbor menampilkan tampilan real-time dari status penerapan konfigurasi Anda.

- Menyebarkan konfigurasi secara otomatis di beberapa akun

  Anda dapat menggunakan Quick Setup secara individu Akun AWS atau di beberapa Akun AWS dan Wilayah AWS dengan mengintegrasikan denganAWS Organizations. Menggunakan Quick Setup di beberapa akun membantu memastikan bahwa organisasi Anda mempertahankan konfigurasi yang konsisten.

- Hilangkan penyimpangan konfigurasi

Konfigurasi drift terjadi setiap kali pengguna membuat perubahan pada layanan atau fitur yang bertentangan dengan pilihan yang dibuat melalui. Quick Setup Quick Setupsecara berkala memeriksa penyimpangan konfigurasi dan mencoba untuk memperbaikinya.

Siapa yang harus menggunakanQuick Setup?

Quick Setuppaling bermanfaat bagi pelanggan yang sudah memiliki pengalaman dengan layanan dan fitur yang mereka siapkan, dan ingin menyederhanakan proses penyiapan mereka. Jika Anda tidak terbiasa dengan konfigurasi yang Layanan AWS Anda gunakanQuick Setup, sebaiknya Anda mempelajari lebih lanjut tentang layanan ini. Tinjau konten dalam Panduan Pengguna yang relevan sebelum Anda membuat konfigurasi denganQuick Setup.

Ketersediaan Quick Setup di Wilayah AWS

Berikut iniWilayah AWS, Anda dapat menggunakan semua jenis Quick Setup konfigurasi untuk seluruh organisasi, seperti yang dikonfigurasiAWS Organizations, atau hanya untuk akun organisasi dan Wilayah yang Anda pilih. Anda juga dapat menggunakan hanya Quick Setup dengan satu akun di Wilayah ini.

- AS Timur (Ohio)
- US East (N. Virginia)
- US West (N. California)
- US West (Oregon)
- Asia Pacific (Mumbai)
- Asia Pacific (Seoul)
- Asia Pacific (Singapore)
- Asia Pacific (Sydney)
- Asia Pacific (Tokyo)
- Canada (Central)
- Europe (Frankfurt)
- Europe (Stockholm)
- Europe (Ireland)
- Europe (London)

- Eropa (Paris)
- Amerika Selatan (São Paulo)

Di Wilayah berikut, hanya tipe konfigurasi Manajemen Host yang tersedia untuk akun individual:

- Europe (Milan)
- Asia Pacific (Hong Kong)
- Middle East (Bahrain) (Middle East (Bahrain))
- China (Beijing)
- China (Ningxia)
- AWS GovCloud (AS-Timur)
- AWS GovCloud (AS-Barat)

Untuk daftar semua Wilayah yang didukung untuk Systems Manager, lihat kolom Region di titik akhir layanan Systems Manager di bagian Referensi Umum Amazon Web Services.

Memulai dengan Quick Setup

Gunakan informasi dalam topik ini untuk membantu Anda bersiap untuk menggunakannyaQuick Setup.

Topik

- Konfigurasikan rumah Wilayah AWS
- Peran dan izin IAM untuk orientasi Quick Setup

Konfigurasikan rumah Wilayah AWS

Untuk memulaiQuick Setup, kemampuanAWS Systems Manager, Anda harus memilih rumah Wilayah AWS dan kemudian onboard denganQuick Setup. Wilayah rumah adalah tempat Quick Setup membuat AWS sumber daya yang digunakan untuk menerapkan konfigurasi Anda. Wilayah beranda tidak dapat diubah setelah Anda memilihnya.

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Quick Setup.

-atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Quick Setupdi panel navigasi.

3. Untuk Pilih Wilayah beranda, pilih Wilayah AWS tempat Anda Quick Setupingin membuat AWS sumber daya yang digunakan untuk menerapkan konfigurasi Anda.
4. Pilih Mulai.

Untuk mulai menggunakanQuick Setup, pilih layanan atau fitur dalam daftar jenis konfigurasi yang tersedia. Tipe konfigurasi khusus untuk fitur Layanan AWS atau. Quick Setup Ketika Anda memilih jenis konfigurasi, Anda memilih opsi yang ingin dikonfigurasi untuk layanan atau fitur tersebut. Secara default, jenis konfigurasi membantu Anda mengatur layanan atau fitur untuk menggunakan praktik terbaik yang direkomendasikan.

Setelah menyiapkan konfigurasi, Anda dapat melihat detail tentang hal itu dan status deployment di seluruh unit organisasi (OU) dan Region. Anda juga dapat melihat status State Manager asosiasi untuk konfigurasi. State Manageradalah kemampuanAWS Systems Manager. Di panel Detail konfigurasi, Anda dapat melihat ringkasan Quick Setup konfigurasi. Ringkasan ini mencakup detail dari semua akun dan penyimpangan konfigurasi yang terdeteksi.

Peran dan izin IAM untuk orientasi Quick Setup

Selama orientasi, Quick Setup buat peran AWS Identity and Access Management (IAM) berikut atas nama Anda:

- `AWS-QuickSetup-StackSet-Local-ExecutionRole`— Memberikan AWS CloudFormation izin untuk menggunakan template apa pun.
- `AWS-QuickSetup-StackSet-Local-AdministrationRole`— Memberikan izin untuk AWS CloudFormation berasumsi. `AWS-QuickSetup-StackSet-Local-ExecutionRole`

Jika Anda melakukan onboarding akun pengelolaan—akun yang Anda gunakan untuk membuat organisasi AWS Organizations — Quick Setup juga membuat peran berikut atas nama Anda:

- `AWS-QuickSetup-SSM-RoleForEnablingExplorer`— Memberikan izin ke runbook `AWS-EnableExplorer` otomatisasi. `AWS-EnableExplorer`Runbook mengkonfigurasiExplorer,

kemampuan Systems Manager, untuk menampilkan informasi untuk beberapa Akun AWS dan. Wilayah AWS

- `AWSServiceRoleForAmazonSSM`— Peran terkait layanan yang memberikan akses ke AWS sumber daya yang dikelola dan digunakan oleh Systems Manager.
- `AWSServiceRoleForAmazonSSM_AccountDiscovery`— Peran terkait layanan yang memberikan izin kepada Systems Manager untuk menelepon untuk menemukan Akun AWS informasi saat Layanan AWS menyinkronkan data. Untuk informasi selengkapnya, lihat Tentang peran `AWSServiceRoleForAmazonSSM_AccountDiscovery`.

Saat melakukan onboarding akun manajemen, Quick Setup aktifkan akses tepercaya antara AWS Organizations dan CloudFormation untuk menerapkan Quick Setup konfigurasi di seluruh organisasi Anda. Untuk mengaktifkan akses tepercaya, akun manajemen Anda harus memiliki izin administrator. Setelah onboarding, Anda tidak lagi memerlukan izin administrator. Untuk informasi selengkapnya, lihat Mengaktifkan akses tepercaya dengan Organizations.

Untuk informasi tentang jenis AWS Organizations akun, lihat AWS Organizationsterminologi dan konsep di Panduan AWS Organizations Pengguna.

> **Note**
>
> Quick Setupdigunakan AWS CloudFormation StackSets untuk menyebarkan konfigurasi Anda di seluruh Akun AWS dan Wilayah. Jika jumlah akun target dikalikan dengan jumlah Wilayah melebihi 10.000, konfigurasi gagal diterapkan. Sebaiknya tinjau kasus penggunaan Anda dan buat konfigurasi yang menggunakan lebih sedikit target untuk mengakomodasi pertumbuhan organisasi Anda. Instans tumpukan tidak diterapkan ke akun manajemen organisasi Anda. Untuk informasi selengkapnya, lihat Pertimbangan saat membuat kumpulan tumpukan dengan izin yang dikelola layanan.

Jika pengguna, grup, atau peran Anda memiliki akses ke operasi API yang tercantum dalam tabel berikut, Anda dapat menggunakan semua fiturQuick Setup. Ada dua tab operasi API, tab pertama adalah izin yang diperlukan oleh semua akun dan tab kedua berisi izin tambahan yang Anda perlukan untuk akun manajemen organisasi Anda.

Non-management account

```
"iam:CreateRole",
```

```
"iam:AttachRolePolicy",
"iam:PutRolePolicy",
"iam:GetRole",
"iam:ListRoles",
"iam:PassRole"
"ssm:ListAssociations",
"ssm:ListDocuments",
"ssm:GetDocument",
"ssm:DescribeAssociation",
"ssm:DescribeAutomationExecutions",
"cloudformation:DescribeStackSet",
"cloudformation:DescribeStackInstance",
"cloudformation:DescribeStacks",
"cloudformation:DescribeStackResources",
"cloudformation:ListStackSetOperations",
"cloudformation:ListStackSets",
"cloudformation:ListStacks",
"cloudformation:ListStackInstances",
"cloudformation:ListStackSetOperationResults",
"cloudformation:TagResource",
"cloudformation:CreateStack",
"cloudformation:DeleteStackSet",
"cloudformation:UpdateStackSet",
"cloudformation:CreateStackSet",
"cloudformation:DeleteStackInstances",
"cloudformation:CreateStackInstances"
```

Management account

```
"ssm:createResourceDataSync",
"ssm:listResourceDataSync",
"ssm:getOpsSummary",
"ssm:createAssociation",
"ssm:createDocument",
"ssm:startAssociationsOnce",
"ssm:startAutomationExecution",
"ssm:updateAssociation",
"ssm:listAssociations",
"ssm:listDocuments",
"ssm:getDocument",
"ssm:describeAssociation",
```

```
"ssm:describeAutomationExecutions",
"organizations:ListRoots",
"organizations:DescribeOrganization",
"organizations:ListOrganizationalUnitsForParent"
"organizations:EnableAWSServiceAccess",
"cloudformation:describe*"
```

Menggunakan Quick Setup

Quick Setup, kemampuanAWS Systems Manager, menampilkan hasil dari setiap konfigurasi dalam tabel Konfigurasi di Quick Setup halaman beranda. Dari halaman ini, Anda dapat Melihat detail setiap konfigurasi, menghapus konfigurasi dari drop-down Tindakan, atau Buat konfigurasi. Tabel Konfigurasi berisi informasi berikut:

- Jenis konfigurasi — Jenis konfigurasi yang dipilih saat membuat konfigurasi.
- Jenis penyebaran — Menunjukkan apakah penerapan berlaku untuk seluruh organisasi (`Organizational`) atau hanya akun Anda (`Local`).
- Unit organisasi — Menampilkan unit organisasi (OU) tempat konfigurasi digunakan jika Anda memilih kumpulan target khusus. Unit organisasi dan target khusus hanya tersedia untuk akun manajemen organisasi Anda. Akun manajemen adalah akun yang Anda gunakan untuk membuat organisasiAWS Organizations.
- Wilayah — Wilayah tempat konfigurasi digunakan jika Anda memilih kumpulan target atau target khusus dalam akun Anda Saat Ini.
- Status penyebaran — Status penerapan menunjukkan apakah AWS CloudFormation berhasil menerapkan target atau instance tumpukan. Instance target dan stack berisi opsi konfigurasi yang Anda pilih selama pembuatan konfigurasi.
- Status asosiasi - Status asosiasi adalah status semua asosiasi yang dibuat oleh konfigurasi yang Anda buat. Asosiasi untuk semua target harus berjalan dengan sukses; jika tidak, statusnya Gagal.

  Quick Setupmembuat dan menjalankan State Manager asosiasi untuk setiap target konfigurasi. State Manageradalah kemampuanAWS Systems Manager.

Detail konfigurasi

Halaman detail Konfigurasi menampilkan informasi tentang penerapan konfigurasi dan asosiasi terkait. Dari halaman ini, Anda dapat mengedit opsi konfigurasi, memperbarui target, atau menghapus konfigurasi. Anda juga dapat melihat detail dari setiap penerapan konfigurasi untuk mendapatkan informasi lebih lanjut tentang asosiasi.

Bergantung pada jenis konfigurasi, satu atau lebih grafik status berikut ditampilkan:

Status penerapan konfigurasi

Menampilkan jumlah penerapan yang berhasil, gagal, atau sedang berjalan atau tertunda. Deployment terjadi di akun target tertentu dan Wilayah yang berisi node yang dipengaruhi oleh konfigurasi.

Status asosiasi konfigurasi

Menampilkan jumlah State Manager asosiasi yang telah berhasil, gagal, atau tertunda. Quick Setupmembuat asosiasi di setiap penerapan untuk opsi konfigurasi yang dipilih.

Status penyiapan

Menampilkan jumlah tindakan yang dilakukan oleh jenis konfigurasi dan statusnya saat ini.

Kepatuhan sumber daya

Menampilkan jumlah sumber daya yang sesuai dengan kebijakan konfigurasi yang ditentukan.

Tabel detail Konfigurasi menampilkan informasi tentang penerapan konfigurasi Anda. Anda dapat melihat detail selengkapnya tentang setiap penerapan dengan memilih penerapan dan kemudian memilih Lihat detail. Halaman detail dari setiap penerapan menampilkan asosiasi yang diterapkan ke node dalam penerapan itu.

Mengedit dan menghapus konfigurasi Anda

Anda dapat mengedit opsi konfigurasi konfigurasi dari halaman detail Konfigurasi dengan memilih Tindakan dan kemudian Edit opsi konfigurasi. Saat Anda menambahkan opsi baru ke konfigurasi, Quick Setup jalankan penerapan Anda dan buat asosiasi baru. Saat Anda menghapus opsi dari konfigurasi, Quick Setup jalankan penerapan Anda dan hapus asosiasi terkait apa pun.

> **Note**
>
> Anda dapat mengedit Quick Setup konfigurasi untuk akun Anda kapan saja. Untuk mengedit konfigurasi Organisasi, status Konfigurasi harus Sukses atau Gagal.

Anda juga dapat memperbarui target yang disertakan dalam konfigurasi Anda dengan memilih Tindakan dan Tambahkan OU, Tambah Wilayah, Hapus OU, atau Hapus Wilayah. Jika akun Anda tidak dikonfigurasi sebagai akun manajemen atau Anda membuat konfigurasi hanya untuk akun saat ini, Anda tidak dapat memperbarui unit organisasi target (OU). Menghapus Wilayah atau OU menghapus asosiasi dari Wilayah atau OU tersebut.

Anda dapat menghapus konfigurasi Quick Setup dengan memilih konfigurasi, lalu Tindakan, dan kemudian Hapus konfigurasi. Atau, Anda dapat menghapus konfigurasi dari halaman detail Konfigurasi di bawah dropdown Tindakan dan kemudian Hapus konfigurasi. Quick Setupkemudian meminta Anda untuk Hapus semua OU dan Wilayah yang mungkin membutuhkan waktu untuk menyelesaikannya. Menghapus konfigurasi juga menghapus semua asosiasi terkait. Proses penghapusan dua langkah ini menghapus semua sumber daya yang digunakan dari semua akun dan Wilayah dan kemudian menghapus konfigurasi.

Kepatuhan konfigurasi

Anda dapat melihat apakah instans Anda sesuai dengan asosiasi yang dibuat oleh konfigurasi Anda di salah satu Explorer atau Kepatuhan, yang keduanya merupakan kemampuan. AWS Systems Manager Untuk mempelajari lebih lanjut tentang kepatuhan, lihatBekerja dengan Kepatuhan. Untuk mempelajari selengkapnya tentang melihat kepatuhan diExplorer, lihatAWS Systems Manager Explorer.

Jenis Quick Setup konfigurasi yang didukung

Jenis konfigurasi yang didukung

Quick Setupmemberikan dukungan untuk jenis konfigurasi berikut.

- Manajemen host Amazon EC2
- Manajemen Host default untuk organisasi
- AWS Configperekam konfigurasi

- AWS Configpenyebaran paket kesesuaian
- Patch Managerkonfigurasi penambalan organisasi
- Change Managerpengaturan organisasi
- DevOpsKonfigurasi guru
- Distributorpenyebaran paket
- Penjadwalan sumber daya instans Amazon EC2
- OpsCenterpengaturan organisasi
- Penjelajah Sumber Daya AWS konfigurasi

Manajemen host Amazon EC2

GunakanQuick Setup, kemampuanAWS Systems Manager, untuk mengonfigurasi peran keamanan yang diperlukan dengan cepat dan kemampuan Systems Manager yang umum digunakan di instans Amazon Elastic Compute Cloud (Amazon EC2). Anda dapat menggunakan Quick Setup di akun individual atau di beberapa akun dan Wilayah AWS dengan mengintegrasikan denganAWS Organizations. Kemampuan ini membantu Anda mengelola dan memantau kesehatan instans Anda sambil memberikan izin minimum yang diperlukan untuk memulai.

Jika Anda tidak terbiasa dengan layanan dan fitur Systems Manager, kami sarankan Anda meninjau Panduan AWS Systems Manager Pengguna sebelum membuat konfigurasi denganQuick Setup. Untuk informasi selengkapnya tentang Systems Manager, lihat Apakah AWS Systems Manager itu?.

Important

Quick Setupmungkin bukan alat yang tepat untuk digunakan untuk manajemen EC2 jika salah satu dari berikut ini berlaku untuk Anda:

- Anda mencoba membuat instans EC2 untuk pertama kalinya untuk mencoba AWS kemampuan.
- Anda masih baru mengenal manajemen instans EC2.

Sebagai gantinya, kami menyarankan Anda menjelajahi konten berikut:

- Memulai dengan Amazon EC2
- Luncurkan instans menggunakan wizard instans peluncuran baru di Panduan Pengguna Amazon EC2 untuk Instans Linux

- Luncurkan instans menggunakan wizard instans peluncuran baru di Panduan Pengguna Amazon EC2 untuk Instans Windows
- Tutorial: Memulai instans Amazon EC2 Linux di Panduan Pengguna Amazon EC2 untuk Instans Linux

Jika Anda sudah terbiasa dengan manajemen instans EC2 dan ingin merampingkan konfigurasi dan manajemen untuk beberapa instans EC2, gunakan. Quick Setup Baik organisasi Anda memiliki lusinan, ribuan, atau jutaan instans EC2, gunakan Quick Setup prosedur berikut untuk mengonfigurasi beberapa opsi untuk instans tersebut, sekaligus.

Prasyarat

Wilayah rumah untuk Quick Setup harus sudah ditentukan sebelum Anda menyelesaikan tugas-tugas berikut. Untuk informasi, lihat Konfigurasikan rumah Wilayah AWS.

> **Note**
>
> Jenis konfigurasi ini memungkinkan Anda mengatur beberapa opsi untuk seluruh organisasi yang ditentukanAWS Organizations, hanya beberapa akun organisasi dan Wilayah, atau satu akun. Salah satu opsi ini adalah memeriksa dan menerapkan pembaruan SSM Agent setiap dua minggu. Jika Anda seorang administrator organisasi, Anda juga dapat memilih untuk memperbarui semua instans EC2 di organisasi Anda dengan pembaruan agen setiap dua minggu menggunakan jenis Konfigurasi Manajemen Host Default. Untuk informasi, lihat Manajemen Host default untuk organisasi.

Mengkonfigurasi opsi manajemen host untuk instans EC2

Untuk mengatur manajemen host, lakukan tugas-tugas berikut di AWS Systems Manager Quick Setup konsol.

Untuk membuka halaman konfigurasi Manajemen Host

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Quick Setup.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Quick Setupdi panel navigasi.

3. Pada kartu Manajemen Host, pilih Buat.

> **Tip**
>
> Jika Anda sudah memiliki satu atau beberapa konfigurasi di akun Anda, pertama-tama pilih tab Perpustakaan atau tombol Buat di bagian Konfigurasi untuk melihat kartu.

Untuk mengonfigurasi opsi manajemen host Systems Manager

- Untuk mengonfigurasi fungsionalitas Systems Manager, di bagian Opsi konfigurasi, pilih opsi di grup Systems Manager yang ingin Anda aktifkan untuk konfigurasi Anda:

  Perbarui Agen Systems Manager (SSM) setiap dua minggu

  Memungkinkan Systems Manager untuk memeriksa setiap dua minggu untuk versi baru agen. Jika ada versi baru, maka Systems Manager secara otomatis memperbarui agen pada node terkelola Anda ke versi terbaru yang dirilis. Quick Setuptidak menginstal agen pada contoh yang belum ada. Untuk informasi tentang yang AMIs telah SSM Agent diinstal sebelumnya, lihatAmazon Machine Images(AMIs) dengan SSM Agent prainstal.

  Kami mendorong Anda untuk memilih opsi ini untuk memastikan bahwa node Anda selalu menjalankan sebagian besar up-to-date versiSSM Agent. Untuk informasi selengkapnyaSSM Agent, termasuk informasi tentang cara menginstal agen secara manual, lihatBekerja dengan SSM Agent.

  Kumpulkan inventaris dari instans Anda setiap 30 menit

  Memungkinkan Quick Setup untuk mengkonfigurasi koleksi jenis metadata berikut:

  - AWSkomponen — driver EC2, agen, versi, dan banyak lagi.
  - Aplikasi — Nama aplikasi, penerbit, versi, dan lainnya.
  - Detail node — Nama sistem, nama sistem operasi (OS), versi OS, boot terakhir, DNS, domain, kelompok kerja, arsitektur OS, dan banyak lagi.

- Konfigurasi jaringan — alamat IP, alamat MAC, DNS, gateway, subnet mask, dan banyak lagi.
- Layanan — Nama, nama tampilan, status, layanan dependen, jenis layanan, jenis awal, dan lainnya (hanya Windows Server node).
- Peran Windows — Nama, nama tampilan, jalur, jenis fitur, status terinstal, dan lainnya (hanya Windows Server node).
- Pembaruan Windows — Hotfix ID, diinstal oleh, tanggal terinstal, dan banyak lagi (hanya Windows Server node).

Untuk informasi lebih lanjut tentang Inventaris, suatu kemampuan AWS Systems Manager, lihat AWS Systems ManagerInventaris.

> **Note**
>
> Opsi pengumpulan Inventaris dapat memakan waktu hingga 10 menit untuk diselesaikan, bahkan jika Anda hanya memilih beberapa node.

Pindai instance untuk patch yang hilang setiap hari

MemungkinkanPatch Manager, kemampuan Systems Manager, untuk memindai node Anda setiap hari dan menghasilkan laporan di halaman Kepatuhan. Laporan menunjukkan berapa banyak node yang patch-compliant sesuai dengan baseline patch default. Laporan tersebut mencakup daftar setiap node dan status kepatuhannya.

Untuk informasi tentang operasi patching dan patch baseline, lihat. AWS Systems Manager Patch Manager

Untuk informasi tentang kepatuhan tambalan, lihat halaman Kepatuhan Systems Manager.

Untuk informasi tentang menambal node terkelola di beberapa akun dan Wilayah dalam satu konfigurasi, lihat Menggunakan kebijakan Quick Setup tambalan danPatch Managerkonfigurasi penambalan organisasi.

> **Important**
>
> Systems Manager mendukung beberapa metode untuk memindai node terkelola untuk kepatuhan patch. Jika Anda menerapkan lebih dari satu metode ini sekaligus,

informasi kepatuhan patch yang Anda lihat selalu merupakan hasil dari pemindaian terbaru. Hasil dari pemindaian sebelumnya ditimpa. Jika metode pemindaian menggunakan baseline patch yang berbeda, dengan aturan persetujuan yang berbeda, informasi kepatuhan patch dapat berubah secara tak terduga. Untuk informasi selengkapnya, lihat Menghindari penimpaan data kepatuhan patch yang tidak disengaja.

Untuk mengonfigurasi opsi manajemen CloudWatch host Amazon

- Untuk mengonfigurasi CloudWatch fungsionalitas, di bagian Opsi konfigurasi, pilih opsi di CloudWatch grup Amazon yang ingin Anda aktifkan untuk konfigurasi Anda:

  Instal dan konfigurasikan CloudWatch agen

  Menginstal konfigurasi dasar CloudWatch agen terpadu di instans Amazon EC2 Anda. Agen mengumpulkan metrik dan file log dari instans Anda untuk Amazon. CloudWatch Informasi ini dikonsolidasikan sehingga Anda dapat dengan cepat menentukan kesehatan instans Anda. Untuk informasi selengkapnya tentang konfigurasi dasar CloudWatch agen, lihat set metrik CloudWatch agen yang telah ditentukan sebelumnya. Mungkin ada biaya tambahan. Untuk informasi lebih lanjut, lihat harga Amazon CloudWatch.

  Perbarui CloudWatch agen setiap 30 hari sekali

  Memungkinkan Systems Manager untuk memeriksa setiap 30 hari untuk versi baru CloudWatch agen. Jika ada versi baru, Systems Manager memperbarui agen pada instans Anda. Kami mendorong Anda untuk memilih opsi ini untuk memastikan bahwa instans Anda selalu menjalankan sebagian besar up-to-date versi CloudWatch agen.

Untuk mengonfigurasi opsi manajemen host Agen Peluncuran Amazon EC2

- Untuk mengonfigurasi fungsionalitas Agen Peluncuran Amazon EC2, di bagian Opsi konfigurasi, pilih opsi di grup Agen Peluncuran Amazon EC2 yang ingin Anda aktifkan untuk konfigurasi Anda:

Perbarui agen peluncuran EC2 setiap 30 hari sekali

Memungkinkan Systems Manager untuk memeriksa setiap 30 hari untuk versi baru agen peluncuran yang diinstal pada instans Anda. Jika versi baru tersedia, Systems Manager memperbarui agen pada instans Anda. Kami mendorong Anda untuk memilih opsi ini untuk memastikan bahwa instans Anda selalu menjalankan sebagian besar up-to-date versi agen peluncuran yang berlaku. Untuk instans Windows Amazon EC2, opsi ini mendukung EC2launch, EC2launch v2, dan EC2config. Untuk instans Amazon EC2 Linux, opsi ini mendukung. `cloud-init` Untuk instans Amazon EC2 Mac, opsi ini mendukung. `ec2-macos-init` Quick Setuptidak mendukung pembaruan agen peluncuran yang diinstal pada sistem operasi yang tidak didukung oleh agen peluncuran, atau pada AL2023.

Untuk informasi lebih lanjut tentang agen inisialisasi ini, lihat topik berikut:

- Konfigurasikan instance Windows menggunakan EC2launch v2
- Konfigurasikan instance Windows menggunakan EC2launch
- Konfigurasikan instance Windows menggunakan layanan EC2config
- Dokumentasi cloud-init
- ec2-macos-init

Untuk memilih instans EC2 yang akan diperbarui oleh konfigurasi manajemen host

- Di bagian Target, pilih metode untuk menentukan akun dan Wilayah tempat konfigurasi akan digunakan:

> **Note**
>
> Anda tidak dapat membuat beberapa konfigurasi Manajemen Quick Setup Host yang menargetkan hal yang samaWilayah AWS.

Entire organization

Konfigurasi Anda diterapkan ke semua unit organisasi (OU) dan Wilayah AWS di organisasi Anda.

> **Note**
>
> Opsi Seluruh organisasi hanya tersedia jika Anda mengonfigurasi manajemen host dari akun manajemen organisasi Anda.

Custom

1. Di bagian Target OU, pilih OU tempat Anda ingin menerapkan konfigurasi manajemen host ini.
2. Di bagian Wilayah Target, pilih Wilayah tempat Anda ingin menerapkan konfigurasi manajemen host ini.

Current account

Pilih salah satu opsi Wilayah dan ikuti langkah-langkah untuk opsi itu.

Wilayah Saat Ini

Pilih cara menargetkan instans di Wilayah saat ini saja:

- Semua instans — Konfigurasi manajemen host secara otomatis menargetkan setiap EC2 di Wilayah saat ini.
- Tag - Pilih Tambah dan masukkan kunci dan nilai opsional yang ditambahkan ke instance yang akan ditargetkan.
- Grup sumber daya - Untuk grup Sumber Daya, pilih grup sumber daya yang ada yang berisi instans EC2 yang akan ditargetkan.
- Manual - Di bagian Instans, pilih kotak centang setiap instans EC2 yang akan ditargetkan.

Pilih Wilayah

Pilih cara menargetkan instance di Wilayah yang Anda tentukan dengan memilih salah satu dari berikut ini:

- Semua instance - Semua instance di Wilayah yang Anda tentukan ditargetkan.

- Tag — Pilih Tambah dan masukkan kunci dan nilai opsional yang telah ditambahkan ke instance yang akan ditargetkan.

Di bagian Wilayah Target, pilih Wilayah tempat Anda ingin menerapkan konfigurasi manajemen host ini.

Untuk menentukan opsi profil contoh

- Seluruh organisasi dan target Custom saja.

  Di bagian Opsi profil Instans, pilih apakah Anda ingin menambahkan kebijakan IAM yang diperlukan ke profil instans yang ada yang dilampirkan pada instans Anda, atau Quick Setup untuk mengizinkan membuat kebijakan IAM dan profil instans dengan izin yang diperlukan untuk konfigurasi yang Anda pilih.

Setelah menentukan semua pilihan konfigurasi Anda, pilih Buat.

Manajemen Host default untuk organisasi

Dengan kemampuan Quick SetupAWS Systems Manager, Anda dapat mengaktifkan Konfigurasi Manajemen Host Default untuk semua akun dan Wilayah yang telah ditambahkan ke organisasi AndaAWS Organizations. Hal ini memastikan bahwa SSM Agent instans Amazon Elastic Compute Cloud (EC2) tetap up to date di organisasi, dan instans tersebut dapat terhubung ke Systems Manager.

Sebelum Anda memulai

Pastikan bahwa persyaratan berikut terpenuhi sebelum mengaktifkan pengaturan ini.

- Wilayah rumah untuk Quick Setup harus sudah ditentukan sebelum Anda menyelesaikan tugas berikut. Untuk informasi, lihat Konfigurasikan rumah Wilayah AWS.
- Versi terbaru SSM Agent sudah diinstal pada semua instans EC2 untuk dikelola di organisasi Anda.
- Instans EC2 Anda yang akan dikelola menggunakan Layanan Metadata Instans Versi 2 (IMDSv2).
- Anda masuk ke akun manajemen untuk organisasi Anda, sebagaimana ditentukan dalamAWS Organizations, menggunakan identitas AWS Identity and Access Management (IAM) (pengguna, peran, atau grup) dengan izin administrator.

Menggunakan peran manajemen instans EC2 default

Konfigurasi Manajemen Host default menggunakan pengaturan `default-ec2-instance-management-role` layanan untuk Systems Manager. Ini adalah peran dengan izin yang ingin Anda sediakan untuk semua akun di organisasi Anda untuk memungkinkan komunikasi antara SSM Agent instans dan layanan Systems Manager di cloud.

Jika Anda telah mengatur peran ini menggunakan perintah update-service-settingCLI, Konfigurasi Manajemen Host Default menggunakan peran itu. Jika Anda belum menetapkan peran ini, Quick Setup akan membuat dan menerapkan peran untuk Anda.

Untuk memeriksa apakah peran ini telah ditentukan untuk organisasi Anda, gunakan get-service-settingperintah.

Aktifkan pembaruan otomatis SSM Agent setiap dua minggu

Gunakan prosedur berikut untuk mengaktifkan opsi Konfigurasi Manajemen Host Default untuk seluruh AWS Organizations organisasi Anda.

Untuk mengaktifkan pembaruan otomatis SSM Agent setiap dua minggu

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Quick Setup.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Quick Setupdi panel navigasi.
3. Pada kartu Konfigurasi Manajemen Host Default, pilih Buat.

   > **Tip**
   >
   > Jika Anda sudah memiliki satu atau beberapa konfigurasi di akun Anda, pertama-tama pilih tab Perpustakaan atau tombol Buat di bagian Konfigurasi untuk melihat kartu.
4. Di bagian Opsi konfigurasi, pilih Aktifkan pembaruan otomatis SSM Agent setiap dua minggu.
5. Pilih Buat

AWS Configperekam konfigurasi

denganQuick SetupSebuah kemampuan dariAWS Systems Manager, Anda dapat dengan cepat membuat perekam konfigurasi yang didukung olehAWS Config. Gunakan perekam konfigurasi untuk mendeteksi perubahan dalam konfigurasi sumber daya Anda dan menangkap perubahan sebagai item konfigurasi. Jika Anda tidak terbiasa denganAWS Config, kami sarankan untuk mempelajari lebih lanjut tentang layanan dengan meninjau konten diAWS ConfigPanduan Pengembangsebelum membuat konfigurasi denganQuick Setup. Untuk informasi lebih lanjut tentang AWS Config, lihat Apa yang dimaksud AWS Config? dalam Panduan DeveloperAWS Config.

Secara default, perekam konfigurasi mencatat semua sumber daya yang didukung di Wilayah AWS tempat AWS Config sedang berjalan. Anda dapat menyesuaikan konfigurasi sehingga hanya jenis sumber daya yang Anda tentukan yang dicatat. Untuk informasi lebih lanjut, lihat Memilih sumber daya yang direkam AWS Config di Panduan Developer AWS Config.

Anda dikenakan biaya penggunaan layanan saat AWS Config mulai merekam konfigurasi. Untuk informasi harga, lihat harga AWS Config.

> **Note**
>
> Jika Anda sudah membuat perekam konfigurasi,Quick Setuptidak berhenti merekam atau membuat perubahan apa pun pada jenis sumber daya yang sudah Anda rekam. Jika Anda memilih untuk merekam jenis sumber daya tambahan menggunakanQuick Setup, layanan menambahkannya ke grup perekam Anda yang ada. MenghapusQuick Setup Rekaman konfigurasijenis konfigurasi tidak menghentikan perekam konfigurasi. Perubahan terus dicatat, dan biaya penggunaan layanan berlaku sampai Anda menghentikan perekam konfigurasi. Untuk mempelajari lebih lanjut tentang mengelola perekam konfigurasi, lihat Mengelola Perekam Konfigurasi di Panduan Developer AWS Config.

Prasyarat

Rumah Wilayah untukQuick Setupharus sudah ditentukan sebelum Anda menyelesaikan tugas-tugas berikut. Untuk informasi, lihat Konfigurasikan rumah Wilayah AWS.

Untuk mengatur perekaman AWS Config, lakukan tugas-tugas berikut di konsol AWS Systems Manager.

Untuk mengaturAWS Configmerekam denganQuick Setup

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Quick Setup.

   -atau-

   JikaAWS Systems Managerhalaman rumah terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilihQuick Setupdi panel navigasi.
3. PadaPerekaman Konfigurasikartu, pilihBuat.

   > **Tip**
   >
   > Jika Anda sudah memiliki satu atau lebih konfigurasi di akun Anda, pertama-tama pilihPerpustakaantab atauBuattombol diKonfigurasibagian untuk melihat kartu.
4. Di bagian Opsi konfigurasi, pilih jenis sumber daya AWS yang ingin Anda rekam dan apakah Anda ingin menyertakan sumber daya global.
5. Pilih Region yang Anda ingin digunakan AWS Config saat merekam perubahan yang dibuat pada sumber daya global. Nilai yang Anda tentukan menentukan asal panggilan API ketika AWS Config mengumpulkan informasi tentang sumber daya global dalam konfigurasi Anda. Region yang Anda pilih harus Region yang Anda tentukan nanti di Target.
6. Buat bucket Amazon Simple Storage Service (Amazon S3) baru, atau pilih bucket yang sudah ada yang ingin Anda kirimkan snapshot konfigurasi.
7. Pilih opsi notifikasi yang Anda inginkan. AWS Config menggunakan Amazon Simple Notification Service (Amazon SNS) untuk memberitahu Anda tentang kejadian AWS Config penting yang berkaitan dengan sumber daya Anda. Jika Anda memilih opsi Gunakan topik SNS yang ada, Anda harus memberikan ID Akun AWS dan nama topik Amazon SNS yang ada di akun yang ingin Anda gunakan. Jika Anda menargetkan beberapa Wilayah AWS, nama topik harus identik di setiap Region.
8. DiJadwalbagian, pilih seberapa sering Anda inginkanQuick Setupuntuk memulihkan perubahan yang dibuat pada sumber daya yang berbeda dari konfigurasi Anda. Opsi Default berjalan sekali. Jika Anda tidak mauQuick Setupuntuk memulihkan perubahan yang dibuat pada sumber daya yang berbeda dari konfigurasi Anda, pilihNonaktifkan remediasidi bawahKustom.

9. Di bagian Target, pilih apakah akan mengizinkan perekaman AWS Config untuk seluruh organisasi Anda, beberapa unit organisasi (OU), atau akun tempat Anda log in.

   Jika Anda memilih Seluruh organisasi, lanjutkan ke langkah 12.

   Jika Anda memilih Kustom, lanjutkan ke langkah 11.

10. Di bagian OU target, pilih kotak centang OU dan Region tempat Anda ingin menggunakan perekaman AWS Config.
11. Pilih Buat.

AWS Configpenyebaran paket kesesuaian

Paket kesesuaian adalah kumpulanAWS Configaturan dan tindakan remediasi. denganQuick Setup, Anda dapat menerapkan paket kesesuaian sebagai entitas tunggal di akun danWilayah AWSatau di seluruh organisasi diAWS Organizations. Ini membantu Anda mengelola kepatuhan konfigurasiAWSsumber daya dalam skala besar, dari definisi kebijakan hingga audit dan pelaporan agregat, dengan menggunakan kerangka kerja umum dan model pengemasan.

Prasyarat

Rumah Wilayah untukQuick Setupharus sudah ditentukan sebelum Anda menyelesaikan tugas-tugas berikut. Untuk informasi, lihat Konfigurasikan rumah Wilayah AWS.

Untuk menerapkan paket kesesuaian, lakukan tugas-tugas berikut diAWS Systems Manager Quick Setupkonsol.

> **Note**
>
> Anda harus mengaktifkanAWS Configmerekam sebelum menerapkan konfigurasi ini. Untuk informasi lebih lanjut, lihatPaket kesesuaiandiAWS ConfigPanduan Pengembang.

Untuk menyebarkan paket kesesuaian denganQuick Setup

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Quick Setup.

   -atau-

JikaAWS Systems Managerhalaman rumah terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilihQuick Setupdi panel navigasi.

3. PadaPaket Kesesuaiankartu, pilihBuat.

> **Tip**
>
> Jika Anda sudah memiliki satu atau lebih konfigurasi di akun Anda, pertama-tama pilihPerpustakaantab atauBuattombol diKonfigurasibagian untuk melihat kartu.

4. DiPilih paket kesesuaianbagian, pilih paket kesesuaian yang ingin Anda terapkan.
5. DiJadwalbagian, pilih seberapa sering Anda inginkanQuick Setupuntuk memulihkan perubahan yang dibuat pada sumber daya yang berbeda dari konfigurasi Anda. Opsi Default berjalan sekali. Jika Anda tidak mauQuick Setupuntuk memulihkan perubahan yang dibuat pada sumber daya yang berbeda dari konfigurasi Anda, pilihDinonaktifkandi bawahKustom.
6. DiTargetbagian, pilih apakah akan menerapkan paket kesesuaian ke seluruh organisasi Anda, beberapaWilayah AWS, atau akun yang saat ini Anda masuki.

   Jika Anda memilih Seluruh organisasi, lanjutkan ke langkah 8.

   Jika Anda memilih Kustom, lanjutkan ke langkah 7.
7. DiWilayah Targetbagian, pilih kotak centang Wilayah yang ingin Anda gunakan paket kesesuaian.
8. Pilih Buat.

Patch Managerkonfigurasi penambalan organisasi

Dengan kemampuan Quick Setup AWS Systems Manager, Anda dapat membuat kebijakan tambalan yang didukung olehPatch Manager. Kebijakan tambalan menentukan jadwal dan garis dasar yang akan digunakan saat menambal instans Amazon Elastic Compute Cloud (Amazon EC2) dan node terkelola lainnya secara otomatis. Dengan menggunakan konfigurasi kebijakan tambalan tunggal, Anda dapat menentukan penambalan untuk semua akun dalam beberapa akun Wilayah AWS di organisasi Anda, hanya untuk akun dan Wilayah yang Anda pilih, atau untuk satu pasangan Account-region. Untuk informasi selengkapnya tentang kebijakan tambalan, lihatMenggunakan kebijakan Quick Setup tambalan.

Prasyarat

Untuk menentukan kebijakan tambalan untuk node yang menggunakanQuick Setup, node harus berupa node terkelola. Untuk informasi selengkapnya tentang mengelola node Anda, lihatMenyiapkan AWS Systems Manager.

> **Important**
>
> Metode pemindaian kepatuhan patch — Systems Manager mendukung beberapa metode untuk memindai node terkelola untuk kepatuhan patch. Jika Anda menerapkan lebih dari satu metode ini sekaligus, informasi kepatuhan patch yang Anda lihat selalu merupakan hasil dari pemindaian terbaru. Hasil dari pemindaian sebelumnya ditimpa. Jika metode pemindaian menggunakan baseline patch yang berbeda, dengan aturan persetujuan yang berbeda, informasi kepatuhan patch dapat berubah secara tak terduga. Untuk informasi selengkapnya, lihat Menghindari penimpaan data kepatuhan patch yang tidak disengaja.
> Status kepatuhan asosiasi dan kebijakan tambalan — Status patching untuk node terkelola yang berada di bawah kebijakan Quick Setup tambalan cocok dengan status eksekusi State Manager asosiasi untuk node tersebut. Jika status eksekusi asosiasi adalah`Compliant`, status patching untuk node terkelola juga ditandai`Compliant`. Jika status eksekusi asosiasi adalah`Non-Compliant`, status patching untuk node terkelola juga ditandai`Non-Compliant`.

Wilayah yang Didukung untuk konfigurasi kebijakan tambalan

Konfigurasi kebijakan tambalan di Quick Setup saat ini didukung di Wilayah berikut:

- AS Timur (Ohio) (us-east-2)
- AS Timur (Virginia Utara) (us-east-1)
- AS Barat (California Utara) (us-west-1)
- AS Barat (Oregon) (us-west-2)
- Asia Pacific (Mumbai) (ap-south-1)
- Asia Pacific (Seoul) (ap-northeast-2)
- Asia Pasifik (Singapura) (ap-southeast-1)
- Asia Pacific (Sydney) (ap-southeast-2)
- Asia Pacific (Tokyo) (ap-northeast-1)
- Kanada (Pusat) (ca-central-1)

- Eropa (Frankfurt) (eu-central-1)
- Eropa (Irlandia) (eu-west-1)
- Eropa (London) (eu-west-2)
- Eropa (Paris) (eu-west-3)
- Eropa (Stockholm) (eu-north-1)
- Amerika Selatan (São Paulo) (sa-east-1)

Izin untuk bucket S3 kebijakan patch

Saat Anda membuat kebijakan tambalan, Quick Setup buat bucket Amazon S3 yang berisi file bernama. `baseline_overrides.json` File ini menyimpan informasi tentang garis dasar tambalan yang Anda tentukan untuk kebijakan tambalan Anda.

Bucket S3 dinamai dalam format`aws-quicksetup-patchpolicy-`*`account-id`*`-`*`quick-setup-configuration-id`*.

Sebagai contoh: `aws-quicksetup-patchpolicy-123456789012-abcde`

Jika Anda membuat kebijakan tambalan untuk organisasi, bucket akan dibuat di akun manajemen organisasi Anda.

Ada dua kasus penggunaan ketika Anda harus memberikan izin kepada AWS sumber daya lain untuk mengakses kebijakan bucket S3 menggunakan AWS Identity and Access Management (IAM) ini:

- Kasus 1: Gunakan profil instans atau peran layanan Anda sendiri dengan node terkelola, bukan yang disediakan oleh Quick Setup
- Kasus 2: Gunakan titik akhir VPC untuk terhubung ke Systems Manager

Kebijakan izin yang Anda perlukan dalam kedua kasus terletak di bagian di bawah ini. Izin kebijakan untuk bucket Quick Setup S3

Kasus 1: Gunakan profil instans atau peran layanan Anda sendiri dengan node terkelola, bukan yang disediakan oleh Quick Setup

Konfigurasi kebijakan tambalan menyertakan opsi untuk Menambahkan kebijakan IAM yang diperlukan ke profil instans yang ada yang dilampirkan ke instance Anda.

Jika Anda tidak memilih opsi ini tetapi Quick Setup ingin menambal node terkelola menggunakan kebijakan tambalan ini, Anda harus memastikan bahwa berikut ini diterapkan:

- Kebijakan terkelola IAM `AmazonSSMManagedInstanceCore` harus dilampirkan ke profil instans IAM atau peran layanan IAM yang digunakan untuk memberikan izin Systems Manager ke node terkelola Anda.
- Anda harus menambahkan izin untuk mengakses keranjang kebijakan tambalan sebagai kebijakan inline ke profil instans IAM atau peran layanan IAM. Anda dapat memberikan akses wildcard ke semua `aws-quicksetup-patchpolicy` bucket atau hanya bucket khusus yang dibuat untuk organisasi atau akun Anda, seperti yang ditunjukkan pada contoh kode sebelumnya.
- Anda harus menandai profil instans IAM atau peran layanan IAM Anda dengan pasangan nilai kunci berikut.

  `Key: QSConfigId-`*`quick-setup-configuration-id`*`, Value:` *`quick-setup-configuration-id`*

  *`quick-setup-configuration-id`*mewakili nilai parameter yang diterapkan ke AWS CloudFormation tumpukan yang digunakan dalam membuat konfigurasi kebijakan tambalan Anda. Untuk mengambil ID ini, lakukan hal berikut:

  1. Buka AWS CloudFormation konsol di https://console.aws.amazon.com/cloudformation.
  2. Pilih nama tumpukan yang digunakan untuk membuat kebijakan tambalan Anda. Namanya dalam format seperti`StackSet-AWS-QuickSetup-PatchPolicy-LA-q4bkg-52cd2f06-d0f9-499e-9818-d887cEXAMPLE`.
  3. Pilih tab Parameter.
  4. Dalam daftar Parameter, di kolom Kunci, cari kunci QS. ConfigurationId Di kolom Nilai untuk barisnya, cari ID konfigurasi, seperti`abcde`.

     Dalam contoh ini, agar tag diterapkan ke profil instans atau peran layanan Anda, kuncinya adalah`QSConfigId-abcde`, dan nilainya adalah`abcde`.

Untuk informasi tentang menambahkan tag ke peran IAM, lihat Menandai peran IAM dan Mengelola tag pada profil instans (AWS CLI atau AWS API) di Panduan Pengguna IAM.

Kasus 2: Gunakan titik akhir VPC untuk terhubung ke Systems Manager

Jika Anda menggunakan titik akhir VPC untuk terhubung ke Systems Manager, kebijakan titik akhir VPC Anda untuk S3 harus mengizinkan akses ke bucket S3 kebijakan patch Anda. Quick Setup

Untuk informasi tentang menambahkan izin ke kebijakan titik akhir VPC untuk S3, lihat Mengontrol akses dari titik akhir VPC dengan kebijakan bucket di Panduan Pengguna Amazon S3.

Izin kebijakan untuk bucket Quick Setup S3

Anda dapat memberikan akses wildcard ke semua `aws-quicksetup-patchpolicy` bucket atau hanya bucket khusus yang dibuat untuk organisasi atau akun Anda. Untuk memberikan izin yang diperlukan untuk dua kasus yang dijelaskan di bawah ini, gunakan salah satu format.

All patch policy buckets

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AccessToAllPatchPolicyRelatedBuckets",
      "Effect": "Allow",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::aws-quicksetup-patchpolicy-*"
    }
  ]
}
```

Specific patch policy bucket

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AccessToMyPatchPolicyRelatedBucket",
      "Effect": "Allow",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::aws-quicksetup-patchpolicy-account-id-quick-setup-
configuration-
id"[1]
    }
  ]
}
```

[1] Setelah konfigurasi kebijakan tambalan dibuat, Anda dapat menemukan nama lengkap bucket Anda di konsol S3. Sebagai contoh: `aws-quicksetup-patchpolicy-123456789012-abcde`

ID dasar patch acak dalam operasi kebijakan tambalan

Operasi patching untuk kebijakan patch menggunakan `BaselineOverride` parameter dalam dokumen `AWS-RunPatchBaseline` SSM Command.

Saat Anda menggunakan `AWS-RunPatchBaseline` untuk menambal di luar kebijakan tambalan, Anda dapat menggunakan `BaselineOverride` untuk menentukan daftar garis dasar tambalan yang akan digunakan selama operasi yang berbeda dari default yang ditentukan. Anda membuat daftar ini dalam file bernama `baseline_overrides.json` dan menambahkannya secara manual ke bucket Amazon S3 yang Anda miliki, seperti yang dijelaskan di. Menggunakan BaselineOverride parameter

Namun, untuk menambal operasi berdasarkan kebijakan tambalan, Systems Manager secara otomatis membuat bucket S3 dan menambahkan `baseline_overrides.json` file ke dalamnya. Kemudian, setiap kali Quick Setup menjalankan operasi patching (menggunakanRun Command) kemampuan, sistem menghasilkan ID acak untuk setiap baseline patch. ID ini berbeda untuk setiap operasi patching kebijakan patch, dan baseline patch yang diwakilinya tidak disimpan atau dapat diakses oleh Anda di akun Anda.

Akibatnya, Anda tidak akan melihat ID dari baseline patch yang dipilih dalam konfigurasi Anda dalam menambal log. Ini berlaku untuk baseline patch AWS terkelola dan baseline patch kustom yang mungkin telah Anda pilih. ID dasar yang dilaporkan dalam log adalah ID yang dihasilkan untuk operasi penambalan tertentu.

Selain itu, jika Anda mencoba melihat detail Patch Manager tentang baseline patch yang dihasilkan dengan ID acak, sistem melaporkan bahwa baseline patch tidak ada. Ini adalah perilaku yang diharapkan dan dapat diabaikan.

Membuat kebijakan tambalan

Prasyarat

Wilayah rumah untuk Quick Setup harus sudah ditentukan sebelum Anda menyelesaikan tugas-tugas berikut. Untuk informasi, lihat Konfigurasikan rumah Wilayah AWS.

Untuk membuat kebijakan tambalan, lakukan tugas berikut di konsol Systems Manager.

Untuk membuat kebijakan tambalan dengan Quick Setup

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.

Jika Anda menyiapkan penambalan untuk organisasi, pastikan Anda masuk ke akun manajemen untuk organisasi tersebut. Anda tidak dapat menyiapkan kebijakan menggunakan akun administrator yang didelegasikan atau akun anggota.

2. Di panel navigasi, pilih Quick Setup.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Quick Setupdi panel navigasi.

3. Pada kartu Patch Manager, pilih Buat.

   > **Tip**
   >
   > Jika Anda sudah memiliki satu atau beberapa konfigurasi di akun Anda, pertama-tama pilih tab Perpustakaan atau tombol Buat di bagian Konfigurasi untuk melihat kartu.

4. Untuk nama Konfigurasi, masukkan nama untuk membantu mengidentifikasi kebijakan tambalan.
5. Di bagian Pemindaian dan instalasi, di bawah operasi Patch, pilih apakah kebijakan tambalan akan Memindai target yang ditentukan atau Pindai dan instal tambalan pada target yang ditentukan.
6. Di bawah Jadwal pemindaian, pilih Gunakan default yang disarankan atau Jadwal pemindaian khusus. Jadwal pemindaian default akan memindai target Anda setiap hari pada pukul 1:00 UTC.

   - Jika Anda memilih Jadwal pemindaian khusus, pilih Frekuensi pemindaian.
   - Jika Anda memilih Harian, masukkan waktu, di UTC, yang ingin Anda pindai target Anda.
   - Jika Anda memilih Ekspresi CRON Kustom, masukkan jadwal sebagai ekspresi CRON. Untuk informasi selengkapnya tentang memformat ekspresi CRON untuk Systems Manager, lihat. Referensi: Ekspresi cron dan rate untuk Systems Manager

     Juga, pilih Tunggu untuk memindai target hingga interval CRON pertama. Secara default, Patch Manager segera memindai node saat mereka menjadi target.

7. Jika Anda memilih Pindai dan menginstal, pilih jadwal Instalasi yang akan digunakan saat menginstal tambalan ke target yang ditentukan. Jika Anda memilih Gunakan default yang direkomendasikan, Patch Manager akan menginstal patch mingguan pada pukul 2:00 UTC pada hari Minggu.

- Jika Anda memilih Jadwal pemasangan kustom, pilih Frekuensi Instalasi.
- Jika Anda memilih Harian, masukkan waktu, di UTC, yang ingin Anda instal pembaruan pada target Anda.
- Jika Anda memilih ekspresi CRON Kustom, masukkan jadwal sebagai ekspresi CRON. Untuk informasi selengkapnya tentang memformat ekspresi CRON untuk Systems Manager, lihat. Referensi: Ekspresi cron dan rate untuk Systems Manager

  Juga, hapus Tunggu untuk menginstal pembaruan hingga interval CRON pertama untuk segera menginstal pembaruan pada node saat menjadi target. Secara default, Patch Manager menunggu hingga interval CRON pertama untuk menginstal pembaruan.
- Pilih Reboot jika diperlukan untuk me-reboot node setelah instalasi patch. Reboot setelah instalasi dianjurkan tetapi dapat menyebabkan masalah ketersediaan.

8. Di bagian dasar Patch, pilih garis dasar patch yang akan digunakan saat memindai dan memperbarui target Anda.

   Secara default, Patch Manager menggunakan baseline patch yang telah ditentukan. Untuk informasi selengkapnya, lihat Tentang baseline yang telah ditetapkan.

   Jika Anda memilih Custom patch baseline, ubah baseline patch yang dipilih untuk sistem operasi yang Anda tidak ingin menggunakan baseline patch yang telah ditentukan. AWS

   Garis dasar tambalan yang tersediaQuick Setup, baik Anda menggunakan garis dasar tambalan yang AWS telah ditentukan sebelumnya atau garis dasar tambalan khusus, adalah garis dasar dari Wilayah Beranda yang Anda pilih.

   > **Note**
   >
   > Jika Anda menggunakan titik akhir VPC untuk terhubung ke Systems Manager, pastikan kebijakan titik akhir VPC Anda untuk S3 memungkinkan akses ke bucket S3 ini. Untuk informasi selengkapnya, lihat Izin untuk bucket S3 kebijakan patch.

   > **Important**
   >
   > Jika Anda menggunakan konfigurasi kebijakan tambalanQuick Setup, pembaruan yang Anda buat ke baseline patch kustom disinkronkan dengan Quick Setup satu jam sekali.

Jika baseline patch kustom yang direferensikan dalam kebijakan tambalan dihapus, spanduk akan ditampilkan di halaman Detail Quick Setup konfigurasi untuk kebijakan tambalan Anda. Spanduk memberi tahu Anda bahwa kebijakan tambalan mereferensikan baseline tambalan yang tidak ada lagi, dan operasi penambalan berikutnya akan gagal. Dalam hal ini, kembali ke halaman Quick Setup Konfigurasi, pilih Patch Manager konfigurasi, dan pilih Tindakan, Edit konfigurasi. Nama dasar patch yang dihapus disorot, dan Anda harus memilih baseline patch baru untuk sistem operasi yang terpengaruh.

9. (Opsional) Di bagian penyimpanan log Patching, pilih Tulis output ke bucket S3 untuk menyimpan log operasi penambalan di bucket Amazon S3.

> **Note**
>
> Jika Anda menyiapkan kebijakan tambalan untuk organisasi, akun manajemen untuk organisasi Anda harus memiliki setidaknya izin hanya-baca untuk bucket ini. Semua unit organisasi yang termasuk dalam kebijakan harus memiliki akses tulis ke bucket. Untuk informasi tentang pemberian akses bucket ke akun yang berbeda, lihat Contoh 2: Pemilik bucket yang memberikan izin bucket lintas akun di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

10. Pilih Browse S3 untuk memilih bucket tempat Anda ingin menyimpan keluaran log tambalan. Akun manajemen harus memiliki akses baca ke bucket ini. Semua akun dan target non-manajemen yang dikonfigurasi di bagian Target harus memiliki akses tulis ke bucket S3 yang disediakan untuk pencatatan.
11. Di bagian Target, pilih salah satu dari berikut ini untuk mengidentifikasi akun dan Wilayah untuk operasi kebijakan tambalan ini.

> **Note**
>
> Jika Anda bekerja dalam satu akun, opsi untuk bekerja dengan organisasi dan unit organisasi (OU) tidak tersedia. Anda dapat memilih apakah akan menerapkan konfigurasi ini ke semua Wilayah AWS di akun Anda atau hanya Wilayah yang Anda pilih.

- Seluruh organisasi — Semua akun dan Wilayah di organisasi Anda.

- Kustom - Hanya OU dan Wilayah yang Anda tentukan.
  - Di bagian Target OU, pilih OU tempat Anda ingin mengatur kebijakan tambalan.
  - Di bagian Wilayah Target, pilih Wilayah tempat Anda ingin menerapkan kebijakan tambalan.
- Akun saat ini — Hanya Wilayah yang Anda tentukan di akun yang saat ini Anda masuki yang ditargetkan. Pilih salah satu cara berikut:
  - Wilayah Saat Ini — Hanya node terkelola di Wilayah yang dipilih di konsol yang ditargetkan.
  - Pilih Wilayah — Pilih masing-masing Wilayah untuk menerapkan kebijakan tambalan.

12. Untuk Pilih bagaimana Anda ingin menargetkan instance, pilih salah satu dari berikut ini untuk mengidentifikasi node yang akan ditambal:

- Semua node terkelola — Semua node terkelola di OU dan Wilayah yang dipilih.
- Tentukan grup sumber daya — Pilih nama grup sumber daya dari daftar untuk menargetkan sumber daya terkait.

> **Note**
>
> Saat ini, memilih grup sumber daya hanya didukung untuk konfigurasi akun tunggal. Untuk menambal sumber daya di beberapa akun, pilih opsi penargetan yang berbeda.

- Tentukan tag node — Hanya node yang ditandai dengan pasangan nilai kunci yang Anda tentukan yang ditambal di semua akun dan Wilayah yang telah Anda targetkan.
- Manual — Pilih node terkelola dari semua akun dan Wilayah yang ditentukan secara manual dari daftar.

> **Note**
>
> Opsi ini saat ini hanya mendukung instans Amazon EC2.

13. Di bagian Kontrol tarif, lakukan hal berikut:

- Untuk Konkurensi, masukkan sejumlah atau persentase node untuk menjalankan kebijakan tambalan secara bersamaan.
- Untuk ambang kesalahan, masukkan jumlah atau persentase node yang dapat mengalami kesalahan sebelum kebijakan patch gagal.

14. (Opsional) Pilih kotak centang Tambahkan kebijakan IAM yang diperlukan ke profil instans yang ada yang dilampirkan ke instance Anda.

Pilihan ini menerapkan kebijakan IAM yang dibuat oleh Quick Setup konfigurasi ini ke node yang sudah memiliki profil instance terpasang (instans EC2) atau peran layanan yang dilampirkan (node yang diaktifkan hibrida). Kami merekomendasikan pilihan ini ketika node terkelola Anda sudah memiliki profil instans atau peran layanan yang dilampirkan, tetapi tidak berisi semua izin yang diperlukan untuk bekerja dengan Systems Manager.

Pilihan Anda di sini diterapkan ke node terkelola yang dibuat nanti di akun dan Wilayah tempat konfigurasi kebijakan tambalan ini berlaku.

> **Important**
>
> Jika Anda tidak memilih kotak centang ini tetapi Quick Setup ingin menambal node terkelola menggunakan kebijakan tambalan ini, Anda harus melakukan hal berikut:
> Tambahkan izin ke profil instans IAM atau peran layanan IAM Anda untuk mengakses bucket S3 yang dibuat untuk kebijakan tambalan Anda
> Tandai profil instans IAM atau peran layanan IAM Anda dengan pasangan nilai kunci tertentu.
> Untuk informasi, lihat Kasus 1: Gunakan profil instans atau peran layanan Anda sendiri dengan node terkelola, bukan yang disediakan oleh Quick Setup.

15. Pilih Buat.

Untuk meninjau status penambalan setelah kebijakan tambalan dibuat, Anda dapat mengakses konfigurasi dari Quick Setuphalaman.

DevOpsKonfigurasi guru

Anda dapat dengan cepat mengkonfigurasi DevOpsOpsi Guru dengan menggunakanQuick Setup. Amazon DevOpsGuru adalah layanan pembelajaran mesin (ML) yang memudahkan untuk meningkatkan kinerja dan ketersediaan operasional aplikasi. DevOpsGuru mendeteksi perilaku yang berbeda dari pola operasi normal sehingga Anda dapat mengidentifikasi masalah operasional jauh sebelum berdampak pada pelanggan Anda. DevOpsGuru secara otomatis menyerap data operasional dari AndaAWSaplikasi dan menyediakan dasbor tunggal untuk memvisualisasikan masalah dalam data operasional Anda. Anda bisa memulai dengan DevOpsGuru untuk meningkatkan ketersediaan dan keandalan aplikasi tanpa pengaturan manual atau keahlian pembelajaran mesin.

Mengkonfigurasi DevOpsGuru denganQuick Setuptersedia dalam hal berikutWilayah AWS:

- AS Timur (N. Virginia)
- US East (Ohio)
- US West (Oregon)
- Europe (Frankfurt)
- Europe (Ireland)
- Europe (Stockholm)
- Asia Pacific (Singapore)
- Asia Pacific (Sydney)
- Asia Pacific (Tokyo)

Untuk informasi harga, lihatAmazon DevOps Harga Guru.

Prasyarat

Rumah Wilayah untukQuick Setupharus sudah ditentukan sebelum Anda menyelesaikan tugas-tugas berikut. Untuk informasi, lihat Konfigurasikan rumah Wilayah AWS.

Untuk mengatur DevOpsGuru, lakukan tugas-tugas berikut diAWS Systems Manager Quick Setupkonsol.

Untuk mengatur DevOpsGuru denganQuick Setup

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Quick Setup.

   -atau-

   JikaAWS Systems Managerhalaman rumah terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilihQuick Setupdi panel navigasi.
3. PadaDevOps Gurukartu, pilihBuat.

> **Tip**
>
> Jika Anda sudah memiliki satu atau lebih konfigurasi di akun Anda, pertama-tama pilihPerpustakaantab atauBuattombol diKonfigurasibagian untuk melihat kartu.

4. DiOpsi konfigurasibagian, pilihAWSjenis sumber daya yang ingin Anda analisis dan preferensi notifikasi Anda.

   Jika Anda tidak memilihMenganalisis semuaAWSsumber daya di semua akun di organisasi sayapilihan, Anda dapat memilihAWSsumber daya untuk dianalisis nanti di DevOpsKonsol guru. DevOpsGuru menganalisis berbedaAWSjenis sumber daya (seperti bucket Amazon Simple Storage Service (Amazon S3) dan instans Amazon Elastic Compute Cloud (Amazon EC2)), yang dikategorikan ke dalam dua grup harga. Anda membayar untukAWSjam sumber daya dianalisis, untuk setiap sumber daya aktif. Sumber daya hanya aktif jika menghasilkan metrik, peristiwa, atau entri log dalam waktu satu jam. Tarif yang Anda kenakan untuk yang spesifikAWSjenis sumber daya tergantung pada kelompok harga.

   Jika Anda memilihAktifkan notifikasi SNSopsi, topik Amazon Simple Notification Service (Amazon SNS) dibuat di masing-masingAkun AWSdi unit organisasi (OU) yang Anda targetkan dengan konfigurasi Anda. DevOpsGuru menggunakan topik untuk memberi tahu Anda tentang hal penting DevOpsPeristiwa guru, seperti penciptaan wawasan baru. Jika Anda tidak mengaktifkan opsi ini, Anda dapat menambahkan topik nanti di DevOpsKonsol guru.

   Jika Anda memilihAktifkanAWS Systems Manager OpsItemsopsi, item pekerjaan operasional (OpsItems) akan dibuat untuk Amazon terkait EventBridgeacara dan Amazon CloudWatch alarm.

5. DiJadwalbagian, pilih seberapa sering Anda inginkanQuick Setupuntuk memulihkan perubahan yang dibuat pada sumber daya yang berbeda dari konfigurasi Anda. Opsi Default berjalan sekali. Jika Anda tidak mauQuick Setupuntuk memulihkan perubahan yang dibuat pada sumber daya yang berbeda dari konfigurasi Anda, pilihDinonaktifkandi bawahKustom.

6. DiTargetbagian, pilih apakah akan mengizinkan DevOpsGuru untuk menganalisis sumber daya di beberapa unit organisasi (OU) Anda, atau akun yang saat ini Anda masuki.

   Jika Anda memilih Kustom, lanjutkan ke langkah 8.

   Jika Anda memilihAkun saat ini, lanjutkan ke langkah 9.

7. DiTarget OUdanWilayah Targetbagian, pilih kotak centang OU dan Wilayah tempat Anda ingin menggunakan DevOpsGuru.

8. Pilih Wilayah tempat Anda ingin menggunakan DevOpsGuru di akun saat ini.
9. Pilih Buat.

Distributorpenyebaran paket

Distributoradalah kemampuan dariAWS Systems Manager. SEBUAHDistributorpaket adalah kumpulan perangkat lunak atau aset yang dapat diinstal yang dapat digunakan sebagai satu kesatuan. denganQuick Setup, Anda dapat menerapkanDistributorpaket dalamAkun AWSdan sebuahWilayah AWSatau di seluruh organisasi diAWS Organizations. Saat ini, hanya agen EC2launch v2, paket utilitas Amazon Elastic File System (Amazon EFS) dan Amazon CloudWatch Agen dapat dikerahkan denganQuick Setup. Untuk informasi selengkapnya tentang Distributor, lihat AWS Systems Manager Distributor.

Prasyarat

Rumah Wilayah untukQuick Setupharus sudah ditentukan sebelum Anda menyelesaikan tugas-tugas berikut. Untuk informasi, lihat Konfigurasikan rumah Wilayah AWS.

Untuk menyebarkanDistributorpaket, melakukan tugas-tugas berikut diAWS Systems Manager Quick Setupkonsol.

Untuk menyebarkanDistributorpaket denganQuick Setup

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Quick Setup.

   -atau-

   JikaAWS Systems Managerhalaman rumah terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilihQuick Setupdi panel navigasi.
3. PadaDistributorkartu, pilihBuat.

   > **Tip**
   >
   > Jika Anda sudah memiliki satu atau lebih konfigurasi di akun Anda, pertama-tama pilihPerpustakaantab atauBuattombol diKonfigurasibagian untuk melihat kartu.

4. Di bagian Opsi konfigurasi, pilih paket yang ingin Anda deploy.

5. Di bagian Target, pilih apakah akan men-deploy paket untuk seluruh organisasi Anda, beberapa unit organisasi (OU), atau akun tempat Anda log in.

   Jika Anda memilih Seluruh organisasi, lanjutkan ke langkah 8.

   Jika Anda memilih Kustom, lanjutkan ke langkah 7.
6. Di bagian OU target, pilih kotak centang OU dan Region tempat Anda ingin men-deploy paket.
7. Pilih Buat.

Penjadwalan sumber daya instans Amazon EC2

Dengan kemampuan Quick Setup AWS Systems Manager, Anda dapat mengonfigurasi Resource Scheduler untuk mengotomatiskan memulai dan menghentikan instans Amazon Elastic Compute Cloud (Amazon EC2).

Quick SetupKonfigurasi ini membantu Anda mengurangi biaya operasional dengan memulai dan menghentikan instans sesuai dengan jadwal yang Anda tentukan. Kemampuan ini membantu Anda menghindari biaya yang tidak perlu untuk menjalankan instance ketika tidak diperlukan. Misalnya, saat ini Anda mungkin membiarkan instance Anda berjalan terus-menerus, meskipun hanya digunakan 10 jam sehari, 5 hari seminggu. Sebagai gantinya, Anda dapat menjadwalkan instans Anda untuk berhenti setiap hari setelah jam kerja. Akibatnya, akan ada penghematan 70 persen untuk instans tersebut karena waktu berjalan berkurang dari 168 jam menjadi 50 jam. Tidak ada biaya untuk menggunakan Quick Setup. Namun, biaya dapat dikeluarkan oleh sumber daya yang Anda atur dan batas penggunaan tanpa biaya untuk layanan yang digunakan untuk mengatur konfigurasi Anda.

Dengan Resource Scheduler, Anda dapat memilih untuk secara otomatis menghentikan dan memulai instance di beberapa Wilayah AWS dan Akun AWS sesuai dengan jadwal yang Anda tentukan. Quick SetupKonfigurasi menargetkan instans Amazon EC2 menggunakan kunci tag dan nilai yang Anda tentukan. Hanya instance dengan tag yang cocok dengan nilai yang Anda tentukan dalam konfigurasi yang dihentikan atau dimulai oleh Resource Scheduler.

Konfigurasi individual mendukung penjadwalan hingga 5.000 instans per Wilayah. Jika kasus Anda memerlukan lebih dari 5.000 instans untuk dijadwalkan di Wilayah tertentu, Anda harus membuat beberapa konfigurasi. Tandai instans Anda sehingga setiap konfigurasi mengelola hingga 5.000 instance. Saat membuat beberapa Quick Setup konfigurasi Resource Scheduler, Anda harus menentukan nilai kunci tag yang berbeda. Misalnya, satu konfigurasi dapat menggunakan kunci tag “Env” dengan nilai “Prod”, sementara yang lain menggunakan “Env” dan “Dev”.

Jika Anda menghapus konfigurasi, instance tidak lagi dihentikan dan dimulai sesuai dengan jadwal yang ditentukan sebelumnya. Dalam kasus yang jarang terjadi, instance mungkin tidak berhasil dihentikan atau dimulai karena kegagalan operasi API.

Resource Scheduler memulai instance yang ditandai hanya jika mereka berada dalam status. `stopped` Demikian pula, contoh hanya dihentikan jika mereka berada di `running` negara bagian. Resource Scheduler beroperasi pada model yang digerakkan oleh peristiwa dan hanya memulai atau menghentikan instance pada waktu yang Anda tentukan. Misalnya, Anda membuat jadwal yang memulai instance pada pukul 9 pagi. Resource Scheduler memulai semua instance yang terkait dengan tag yang Anda tentukan yang berada dalam `stopped` status pada pukul 9 pagi. Jika instance dihentikan secara manual di lain waktu, Resource Scheduler tidak akan memulainya lagi untuk mempertahankan status. `running` Demikian pula, jika sebuah instance dimulai secara manual setelah dihentikan sesuai dengan jadwal Anda, Resource Scheduler tidak akan menghentikan instance lagi.

Jika Anda membuat jadwal dengan waktu mulai yang lebih lambat dari waktu berhenti, Resource Scheduler mengasumsikan instance Anda berjalan dalam semalam. Misalnya, Anda membuat jadwal yang memulai instance pada pukul 9 malam, dan menghentikan instance pada pukul 7 pagi. Resource Scheduler memulai semua instance yang terkait dengan tag yang Anda tentukan yang berada dalam `stopped` status pada pukul 9 malam, dan menghentikannya pada pukul 7 pagi keesokan harinya. Untuk jadwal semalam, waktu mulai berlaku untuk hari-hari yang Anda pilih untuk jadwal Anda. Namun, waktu berhenti berlaku untuk hari berikutnya dalam jadwal Anda.

Prasyarat

Wilayah rumah untuk Quick Setup harus sudah ditentukan sebelum Anda menyelesaikan tugas berikut. Untuk informasi, lihat Konfigurasikan rumah Wilayah AWS.

Untuk mengatur penjadwalan instans Amazon EC2, lakukan tugas berikut di konsol. AWS Systems Manager Quick Setup

Untuk mengatur penjadwalan instance dengan Quick Setup

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Quick Setup.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Quick Setupdi panel navigasi.

3. Pada kartu Resource Scheduler, pilih Create.

> **Tip**
>
> Jika Anda sudah memiliki satu atau beberapa konfigurasi di akun Anda, pertama-tama pilih tab Perpustakaan atau tombol Buat di bagian Konfigurasi untuk melihat kartu.

4. Di bagian tag Instance, tentukan kunci tag dan nilai yang diterapkan pada instance yang ingin Anda kaitkan dengan jadwal Anda.
5. Di bagian Opsi jadwal, tentukan zona waktu, hari, dan waktu yang ingin Anda mulai dan hentikan instance Anda.
6. Di bagian Target, pilih apakah akan mengatur penjadwalan untuk grup kustom unit organisasi (OU), atau akun saat ini yang Anda masuki:
   - Kustom - Di bagian Target OU, pilih OU tempat Anda ingin mengatur penjadwalan. Selanjutnya, di bagian Wilayah Target, pilih Wilayah tempat Anda ingin mengatur penjadwalan.
   - Akun saat ini — Pilih Wilayah Saat Ini atau Pilih Wilayah. Jika Anda memilih Pilih Wilayah, pilih Wilayah Target tempat Anda ingin mengatur penjadwalan.
7. Verifikasi informasi jadwal di bagian Ringkasan.
8. Pilih Buat.

Penjelajah Sumber Daya AWS konfigurasi

Dengan kemampuan Quick Setup AWS Systems Manager, Anda dapat dengan cepat mengkonfigurasi Penjelajah Sumber Daya AWS untuk mencari dan menemukan sumber daya di Anda Akun AWS atau di seluruh AWS organisasi. Anda dapat mencari sumber daya menggunakan metadata seperti nama, tag, dan ID. Penjelajah Sumber Daya AWS memberikan respons cepat untuk permintaan pencarian Anda dengan menggunakan indeks. Resource Explorer membuat dan memelihara indeks menggunakan berbagai sumber data untuk mengumpulkan informasi tentang sumber daya di Anda Akun AWS.

Quick Setupuntuk Resource Explorer mengotomatiskan proses konfigurasi indeks. Untuk informasi lebih lanjut tentang Penjelajah Sumber Daya AWS, lihat Apa itu Penjelajah Sumber Daya AWS? dalam Penjelajah Sumber Daya AWS User Guide.

SelamaQuick Setup, Resource Explorer melakukan hal berikut:

- Membuat indeks di setiap Wilayah AWS di Anda Akun AWS.
- Memperbarui indeks di Wilayah yang Anda tentukan sebagai indeks agregator untuk akun tersebut.
- Membuat tampilan default di Region indeks agregator. Tampilan ini tidak memiliki filter sehingga mengembalikan semua sumber daya yang ditemukan dalam indeks.

Izin minimum

Untuk melakukan langkah-langkah dalam prosedur berikut, Anda harus memiliki izin berikut:

- Tindakan: `resource-explorer-2:*` — Sumber daya: tidak ada sumber daya tertentu (*)
- Tindakan: `iam:CreateServiceLinkedRole` — Sumber daya: tidak ada sumber daya tertentu (*)

Untuk mengkonfigurasi Resource Explorer

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Quick Setup.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Quick Setupdi panel navigasi.
3. Pilih wilayah rumah dan kemudian pilih Memulai.
4. Pada kartu Resource Explorer, pilih Buat.
5. Di bagian Wilayah Indeks Agregator, pilih Wilayah mana yang ingin Anda isi indeks agregator. Anda harus memilih Wilayah yang sesuai untuk lokasi geografis untuk pengguna Anda.
6. (Opsional) Pilih kotak centang Ganti indeks agregator yang ada di Wilayah selain yang dipilih di atas.

7. Di bagian Target, pilih organisasi target atau Unit Organisasi (OU) tertentu yang berisi sumber daya yang ingin Anda temukan.
8. Di bagian Wilayah, pilih Wilayah mana yang akan disertakan dalam konfigurasi.
9. Tinjau ringkasan konfigurasi, lalu pilih Buat.

Pada halaman Resource Explorer, Anda dapat memantau status konfigurasi.

Pemecahan masalahQuick Setuphasil

Penerapan gagal

Penerapan gagal jika CloudFormation set tumpukan gagal selama pembuatan. Gunakan langkah-langkah berikut untuk menyelidiki kegagalan penerapan.

1. Navigasikan ke konsol AWS CloudFormation tersebut.
2. Pilih tumpukan yang dibuat oleh AndaQuick Setupkonfigurasi. TheNama tumpukantermasuk`QuickSetup`diikuti oleh jenis konfigurasi yang Anda pilih, seperti`SSMHostMgmt`.

   > **Note**
   >
   > CloudFormation terkadang menghapus penerapan tumpukan yang gagal. Jika tumpukan tidak tersedia diTumpukanmeja, pilihDihapusdari daftar filter.

3. LihatStatusdanAlasan status. Untuk informasi selengkapnya tentang status tumpukan, lihatKode status tumpukandiAWS CloudFormationPanduan Pengguna.
4. Untuk memahami langkah tepat yang gagal, lihatEventtab dan tinjau setiap acaraStatus.
5. UlasanPemecahan masalahdiAWS CloudFormationPanduan Pengguna.
6. Jika Anda tidak dapat menyelesaikan kegagalan penerapan menggunakan CloudFormation langkah-langkah pemecahan masalah, hapus konfigurasi dan konfigurasi ulang.

Asosiasi yang gagal

TheDetail konfigurasimeja diDetail konfigurasihalaman konfigurasi Anda menunjukkanStatus konfigurasidariGagaljika salah satu asosiasi gagal selama pengaturan. Gunakan langkah-langkah berikut untuk memecahkan masalah asosiasi yang gagal.

1. DiDetail konfigurasitabel, pilih konfigurasi yang gagal dan kemudian pilihLihat Detail.
2. SalinNama asosiasi.
3. Arahkan keState Managerdan tempelkan nama asosiasi ke bidang pencarian.
4. Pilih asosiasi dan pilihRiwayat eksekusitab.
5. Di bawah ID eksekusi, pilih eksekusi asosiasi yang gagal.
6. TheTarget eksekusi asosiasipage mencantumkan semua node tempat asosiasi berjalan. Pilih tombol Output untuk eksekusi yang gagal dijalankan.
7. Di halaman Output, pilih Langkah - Output untuk melihat pesan kesalahan untuk langkah tersebut dalam eksekusi perintah. Setiap langkah dapat menampilkan pesan kesalahan yang berbeda. Tinjau pesan kesalahan untuk semua langkah untuk membantu memecahkan masalah.

Jika melihat output langkah tidak menyelesaikan masalah, maka Anda dapat mencoba membuat ulang asosiasi. Untuk membuat ulang asosiasi, pertama-tama hapus asosiasi yang gagal diState Manager. Setelah menghapus asosiasi, edit konfigurasi dan pilih opsi yang Anda hapus dan pilihPerbarui.

> **Note**
>
> Untuk menyelidikiGagalasosiasi untukOrganisasikonfigurasi, Anda harus masuk ke akun dengan asosiasi gagal dan menggunakan prosedur asosiasi gagal berikut, yang dijelaskan sebelumnya. ID Asosiasi bukanlah hyperlink ke akun target saat melihat hasil dari akun manajemen.

Status melayang

Saat melihat halaman detail konfigurasi, Anda dapat melihat status drift dari setiap penerapan. Konfigurasi drift terjadi setiap kali pengguna membuat perubahan pada layanan atau fitur yang bertentangan dengan pilihan yang dibuat melaluiQuick Setup. Jika asosiasi telah berubah setelah konfigurasi awal, tabel menampilkan ikon peringatan yang menunjukkan jumlah item yang telah hanyut. Anda dapat menentukan apa yang menyebabkan penyimpangan dengan mengarahkan kursor ke ikon.

Saat asosiasi dihapus diState Manager, penerapan terkait menampilkan peringatan drift. Untuk memperbaikinya, edit konfigurasi dan pilih opsi yang dihapus saat asosiasi dihapus. PilihPerbaruidan menunggu penyebaran selesai.

Manajemen Operasi

Manajemen Operasi adalah rangkaian kemampuan yang membantu mengelola sumber daya AWS Anda.

Topik

- AWS Systems ManagerInsiden Incident Manager
- AWS Systems Manager Explorer
- AWS Systems Manager OpsCenter
- CloudWatchDasbor Amazon dasbor Amazon dasbor Amazon dasbor Amazon

AWS Systems ManagerInsiden Incident Manager

Gunakan Incident Manager, kemampuan dariAWS Systems Manager, untuk mengelola insiden yang terjadi di aplikasi yangAWS dihosting Anda. Incident Manager menggabungkan keterlibatan pengguna, eskalasi, runbook, rencana respons, saluran obrolan, dan analisis pasca-insiden untuk membantu tim Anda menangani insiden lebih cepat dan mengembalikan aplikasi Anda menjadi normal. Untuk mempelajari selengkapnya tentang Incident Manager, lihat Panduan Pengguna Incident Manager.

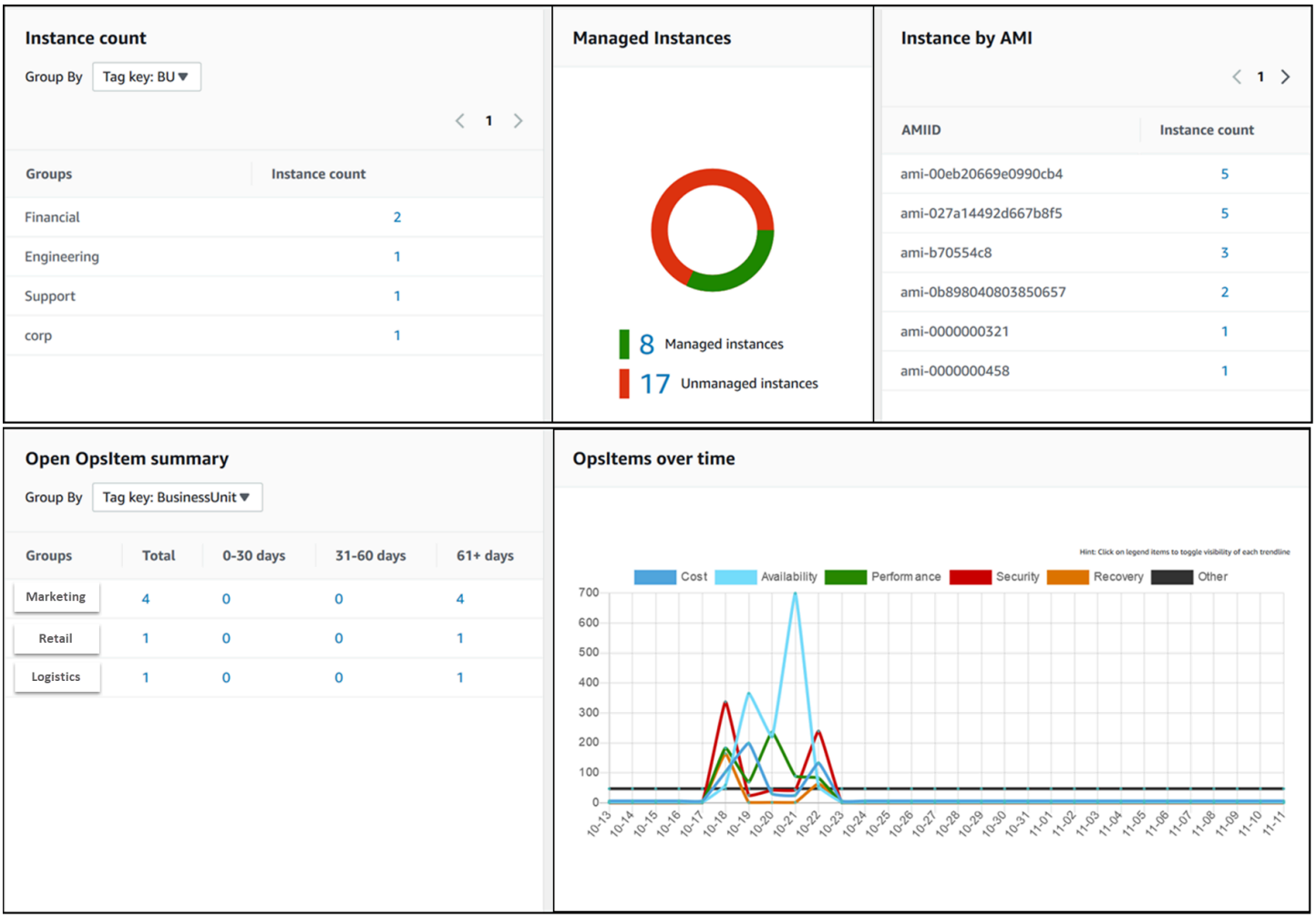
AWS Systems Manager Explorer

AWS Systems ManagerExplorer adalah dasbor operasi yang dapat disesuaikan yang melaporkan informasi tentangAWS sumber daya Anda. Explorermenampilkan tampilan gabungan data operasi (OpsData) untuk AndaAkun AWS dan di seluruhWilayah AWS. DiExplorer,OpsData termasuk metadata tentang node yang dikelola di lingkungan hybrid dan multicloud Anda. OpsDatajuga mencakup informasi yang diberikan oleh kemampuan Systems Manager lainnya, termasuk kepatuhanPatch Manager patch dan detail kepatuhanState Manager asosiasi. Untuk lebih menyederhanakan cara Anda mengaksesOpsData,Explorer menampilkan informasi dariAWS layanan pendukung sepertiAWS ConfigAWS Trusted Advisor,AWS Compute Optimizer,, danAWS Support (kasus dukungan).

Untuk meningkatkan kesadaran operasional,Explorer juga menampilkan item pekerjaan operasional (OpsItems). Explorermenyediakan konteks tentang bagaimanaOpsItems didistribusikan di seluruh unit bisnis atau aplikasi Anda, bagaimana trennya dari waktu ke waktu, dan bagaimana trennya

menurut kategori. Anda dapat mengelompokkan dan memfilter informasiExplorer untuk fokus pada item yang relevan dengan Anda dan yang memerlukan tindakan. Ketika Anda mengidentifikasi masalah prioritas tinggi, Anda dapat menggunakan Systems ManagerOpsCenter untuk menjalankan runbook Otomatisasi dan menyelesaikan masalah tersebut dengan cepat. Untuk memulaiExplorer, buka konsol Systems Manager. Di panel navigasi, pilih Explorer.

Gambar berikut menunjukkan beberapa kotak laporan individu, yang disebut widget, yang tersedia diExplorer.



Apa saja fitur dariExplorer?

Explorermencakup fitur berikut:

- Tampilan informasi yang dapat ditindaklanjuti yang dapat disesuaikan:Explorer mencakupdrag-and-drop widget yang secara otomatis menampilkan informasi yang dapat ditindaklanjuti tentangAWS sumber daya Anda. Explorermenampilkan informasi dalam dua jenis widget.

  - Widget informasi: Widget ini meringkas data dari Amazon EC2,,,,Patch Manager,State Manager, dan mendukungLayanan AWS sepertiAWS Trusted Advisor,AWS Compute Optimizer, danAWS Support. Widget ini memberikan konteks penting untuk membantu Anda memahami status dan risiko operasional sumber daya AWS Anda. Contoh widget informasi termasuk jumlah Instance, Instance by AMI, Total node noncompliant (patch), asosiasi Noncompliant, dan kasus Pusat Support.
  - OpsItemwidget: Systems Manager OpsItemadalah item pekerjaan operasional yang terkait dengan satu atau beberapaAWS sumber daya. OpsItemsadalah fitur dari Systems ManagerOpsCenter. OpsItemsmungkin memerlukanDevOps teknisi untuk menyelidiki dan berpotensi mengatasi masalah. Contoh kemungkinanOpsItems mencakup penggunaan CPU instans EC2 yang tinggi, volume Amazon Elastic Block Store (Amazon EBS) yang terlepas, kegagalanAWS CodeDeploy deployment, atau kegagalan eksekusi Otomatisasi Systems Manager. ContohOpsItem widget termasuk OpsItemRingkasan terbuka, OpsItemberdasarkan status, dan OpsItemsseiring waktu.
- Filter: Setiap widget menawarkan kemampuan untuk memfilter informasi berdasarkan Akun AWS, Wilayah AWS, dan tag. Filter membantu Anda dengan cepat menyaring informasi yang ditampilkan diExplorer.
- Tautan langsung ke layar layanan: Untuk membantu Anda menyelidiki masalah denganAWS sumber daya,Explorer widget berisi tautan langsung ke layar layanan terkait. Filter yang diterapkan pada widget tetap berlaku jika Anda menavigasi ke layar layanan terkait.
- Grup: Untuk membantu Anda memahami jenis masalah operasional di seluruh organisasi, beberapa widget memungkinkan Anda mengelompokkan data berdasarkan akun, Wilayah, dan tag.
- Kunci tag pelaporan: Saat menyiapkanExplorer, Anda dapat menentukan hingga lima kunci tag. Kunci ini membantu Anda mengelompokkan dan memfilter dataExplorer. Jika kunci yang ditentukan cocok dengan kunci pada sumber daya yang menghasilkanOpsItem, kunci dan nilai tersebut akan disertakan dalamOpsItems.
- Tiga modeAkun AWS danWilayah AWS tampilan:Explorer termasuk mode tampilan berikut untukOpsData danOpsItems masukAkun AWS danWilayah AWS:
  - Akun tunggal/Wilayah tunggal: Ini adalah tampilan default. Mode ini memungkinkan pengguna untuk melihat data danOpsItems dari akun mereka sendiri dan Wilayah saat ini.
  - Akun tunggal/Beberapa Wilayah: Mode ini mengharuskan Anda membuat satu atau beberapa sinkronisasi data sumber daya dengan menggunakan halamanExplorer Pengaturan. Sinkronisasi data sumber daya menggabungkanOpsData dari satu atau beberapa Wilayah. Setelah Anda membuat sinkronisasi data sumber daya, Anda dapat mengaktifkan sinkronisasi mana yang

akan digunakan diExplorer dasbor. Anda kemudian dapat memfilter dan mengelompokkan data berdasarkan Wilayah.

- Beberapa akun/beberapa Wilayah: Mode ini mengharuskan organisasi atau perusahaan Anda menggunakan AWS Organizations dengan Semua fitur diaktifkan. Setelah Anda mengonfigurasi AWS Organizations di lingkungan komputasi, Anda dapat menggabungkan semua data akun di akun manajemen. Anda kemudian dapat membuat sinkronisasi data sumber daya sehingga Anda dapat memfilter dan mengelompokkan data berdasarkan Wilayah. Untuk informasi selengkapnya tentang mode Semua fitur Organizations, lihat Mengaktifkan Semua Fitur di Organisasi Anda.

- Pelaporan: Anda dapat mengeksporExplorer laporan sebagai file nilai yang dipisahkan koma (.csv) ke bucket Amazon Simple Storage Service (Amazon S3). Anda menerima peringatan dari Amazon Simple Notification Service (Amazon SNS) ketika ekspor selesai.

BagaimanaExplorer berhubungan denganOpsCenter?

Systems ManagerOpsCenter menyediakan lokasi pusat dimana para teknisi dan profesional operasi dan IT dapat melihat, menyelidiki, dan menyelesaikanOpsItems terkait denganAWS sumber daya. Exploreradalah hub laporan tempatDevOps manajer melihat ringkasan gabungan dari data operasi mereka, termasukOpsItems, diWilayah AWS dan akun. Explorermembantu pengguna menemukan tren dan pola dan, jika perlu, dengan cepat menyelesaikan masalah menggunakan runbook Otomatisasi Systems Manager.

OpsCenterPenyiapan setup terintegrasi denganExplorer Penyiapan Penyiapan. Jika Anda sudah menyiapkanOpsCenter, makaExplorer secara otomatis menampilkan data operasi, termasuk informasi agregat tentangOpsItems. Jika Anda belum menyiapkanOpsCenter, Anda dapat menggunakanExplorer Penyiapan untuk memulai dengan kedua kemampuan. Untuk informasi selengkapnya, lihat Memulai dengan Systems Manager Explorer danOpsCenter.

Apa OpsData?

OpsDataadalah data operasi yang ditampilkan di dasbor Systems Manager Explorer. ExplorermengambilOpsData dari sumber-sumber berikut:

- Amazon Elastic Compute Cloud (Amazon EC2)

  Data yang ditampilkan diExplorer mencakup: jumlah total node, jumlah total node yang dikelola dan tidak terkelola, jumlah total node, jumlah total node, jumlah total node yang dikelola dan tidak

terkelola, jumlah total node yang dikelola dan tidak terkelola, dan jumlah node menggunakan spesifikAmazon Machine Image (AMI).

- Systems ManagerOpsCenter

  Data yang ditampilkanExplorer mencakup: jumlah status, jumlahOpsItems tingkat kepelikan, jumlahOpsItems terbukaOpsItems di seluruh grup dan di periode waktu 30 hari, dan data historis dari waktuOpsItems ke waktu.

- Systems ManagerPatch Manager

  Data yang ditampilkanExplorer termasuk jumlah node noncompliant dan kritis noncompliant.

- AWS Trusted Advisor

  Data yang ditampilkanExplorer mencakup: status pemeriksaan praktik terbaik untuk Instans Cadangan EC2 di bidang pengoptimalan biaya, keamanan, toleransi kesalahan, kinerja, dan batas layanan.

- AWS Compute Optimizer

  Data yang ditampilkan dalamExplorer mencakup: jumlah instans EC2 Kurang dari yang ditentukan dan Lebih dari yang ditentukan, temuan optimasi, detail harga sesuai permintaan, dan rekomendasi untuk tipe instans dan harga instans.

- AWS SupportKasus pusat

  Data yang ditampilkanExplorer mencakup: ID kasus, tingkat kepelikan, status, waktu pembuatan, subjek, layanan, dan kategori.

- AWS Config

  Data yang ditampilkan diExplorer mencakup: ringkasan keseluruhanAWS Config aturan yang sesuai dan tidak sesuai, jumlah sumber daya yang sesuai dan tidak sesuai, dan detail spesifik tentang masing-masing (saat Anda menelusuri aturan atau sumber daya yang tidak sesuai).

- AWS Security Hub

  Data yang ditampilkanExplorer mencakup: ringkasan keseluruhan temuan Security Hub, jumlah setiap temuan yang dikelompokkan berdasarkan tingkat kepelikan, dan detail spesifik tentang temuan.

> **Note**
>
> Untuk melihat kasusAWS Support PusatAWS Trusted Advisor dan diExplorer, Anda harus memiliki akun Enterprise atau Business yang disiapkan denganAWS Support.

Anda dapat melihat dan mengelolaOpsData sumber dari halamanExplorer Pengaturan. Untuk informasi tentang mengatur dan mengonfigurasi layanan yang mengisiExplorer widgetOpsData, lihatMenyiapkan layanan terkait.

Apakah ada biaya untuk digunakanExplorer?

Ya. Saat Anda mengaktifkan aturan default untuk membuatOpsItems selama Penyiapan Terintegrasi, Anda memulai proses yang dibuat secara otomatisOpsItems. Akun Anda dikenakan biaya berdasarkan jumlah yangOpsItems dibuat per bulan. Akun Anda juga dikenakan biaya berdasarkan jumlah panggilan API `GetOpsItem`, `DescribeOpsItem`, `UpdateOpsItem`, dan `GetOpsSummary` yang dilakukan per bulan. Selain itu, Anda dapat dikenakan biaya untuk panggilan API publik ke layanan lain yang mengekspos informasi diagnostik yang relevan. Untuk informasi selengkapnya, lihat Harga AWS Systems Manager.

Topik

- Memulai dengan Systems Manager Explorer danOpsCenter
- Menggunakan Systems Manager Explorer
- MengeksporOpsData dari Systems ManagerExplorer
- Memecahkan Masalah Systems Manager Explorer

Memulai dengan Systems Manager Explorer danOpsCenter

AWS Systems Managermenggunakan pengalaman penyiapan terintegrasi untuk membantu Anda memulai dengan Systems ManagerExplorer dan Systems ManagerOpsCenter. Dalam dokumentasi ini,Explorer danOpsCenter Setup disebut Integrated Setup. Jika Anda sudah menyiapkanOpsCenter, Anda masih perlu melengkapi Penyiapan Terintegrasi untuk memverifikasi pengaturan dan opsi. Jika Anda belum menyiapkanOpsCenter, Anda dapat menggunakan Penyiapan Terintegrasi untuk memulai dengan kedua kemampuan.

> **Note**
>
> Penyiapan Terintegrasi hanya tersedia di konsol Systems Manager. Anda tidak dapat mengaturExplorer atauOpsCenter secara terprogram.

Penyiapan Terintegrasi melakukan tugas-tugas berikut:

- Mengonfigurasi peran dan izin: Penyiapan Terintegrasi membuatAWS Identity and Access Management (IAM) role yang EventBridge memungkinkan Amazon membuat secara otomatisOpsItems berdasarkan aturan default. Setelah menyiapkan, Anda harus mengonfigurasi izin pengguna, grup, atau peran untukOpsCenter, seperti yang dijelaskan di bagian ini.
- Mengizinkan aturan default untukOpsItem pembuatan: Penyiapan Terintegrasi membuat aturan default di EventBridge. Aturan ini secara otomatis dibuatOpsItems sebagai respons terhadap peristiwa. Contoh peristiwa ini adalah: perubahan status sumber daya AWS, perubahan pengaturan keamanan, atau layanan menjadi tidak tersedia.
- Mengizinkan OpsData sumber: Penyiapan Terintegrasi memungkinkan sumber data yang mengisiExplorer widget.
- Memungkinkan Anda menentukan kunci tag pelaporan: Penyiapan Terintegrasi memungkinkan Anda menentukan hingga lima kunci tag pelaporan untuk secara otomatis ditetapkan ke baruOpsItems yang memenuhi kriteria tertentu.

Setelah Anda menyelesaikan Penyiapan Terintegrasi, kami menyarankan Anda MenyiapkanExplorer untuk menampilkan data dari beberapa Wilayah dan akun. ExplorerdanOpsCenter secara otomatis menyinkronkan OpsData danOpsItems untukAkun AWS danWilayah AWS Anda gunakan ketika Anda menyelesaikan Integrated Setup. Anda dapat menggabungkan OpsData danOpsItems dari akun dan Wilayah lain dengan membuat sinkronisasi data sumber daya.

> **Note**
>
> Anda dapat mengubah konfigurasi pengaturan kapan saja di halaman Pengaturan.

Menyiapkan layanan terkait

AWS Systems ManagerExplorer danAWS Systems ManagerOpsCenter mengumpulkan informasi dari, atau berinteraksi dengan, kemampuan Systems ManagerLayanan AWS dan. Kami

menyarankan Anda menyiapkan dan mengonfigurasi layanan atau kemampuan lain ini sebelum menggunakan Penyiapan Terintegrasi.

Tabel berikut mencakup tugas yangOpsCenter memungkinkanExplorer dan mengumpulkan informasi dari, atau berinteraksi dengan, kemampuan Systems Manager.Layanan AWS

Tugas	Informasi
Memverifikasi izin di Otomatisasi Systems Manager	ExplorerdanOpsCenter memungkinkan Anda untuk memperbaiki masalah denganAWS sumber daya dengan menggunakan runbook Otomatisasi Systems Manager. Untuk menggunakan kemampuan perbaikan ini, Anda harus memiliki izin untuk menjalankan runbook Otomatisasi Systems Manager. Untuk informasi selengkapnya, lihat Menyiapkan Otomatisasi.
Menyiapkan dan mengonfigurasi Systems ManagerPatch Manager	Explorermencakup widget yang menyediak an informasi tentang kepatuhan patch. Untuk melihat data iniExplorer, Anda harus mengonfig urasi patching. Untuk informasi selengkapnya, lihat AWS Systems Manager Patch Manager.
Menyiapkan dan mengonfigurasi Systems ManagerState Manager	Explorermencakup widget yang menyediakan informasi tentang kepatuhanState Manager asosiasi Systems Manager. Untuk melihat data iniExplorer, Anda harus mengonfigurasiState Manager. Untuk informasi selengkapnya, lihat AWS Systems Manager State Manager.
Aktifkan Perekam Konfigurasi AWS Config	Explorermenggunakan data yang disediakan oleh perekamAWS Config konfigurasi untuk mengisi widget dengan informasi tentang instans EC2 Anda. Untuk melihat data iniExplor er, aktifkan perekamAWS Config konfigurasi. Untuk informasi selengkapnya, lihat Mengelola Perekam Konfigurasi.

Tugas	Informasi
	Note Setelah Anda mengizinkan perekam konfigurasi, Systems Manager dapat memakan waktu hingga enam jam untuk menampilkan data dalamExplorer widget yang menampilkan informasi tentang instans EC2 Anda.
Mengaktifkan AWS Trusted Advisor	Explorermenggunakan data yang disediaka n olehTrusted Advisor untuk menampilkan status pemeriksaan praktik terbaik untuk Instans Cadangan Amazon EC2 di bidang pengoptimalan biaya, keamanan, toleransi kesalahan, kinerja, dan batas layanan. Untuk melihat data iniExplorer, Anda harus memiliki paket dukungan bisnis atau korporasi. Untuk informasi selengkapnya, lihat AWS Support.
Mengaktifkan AWS Compute Optimizer	Explorermenggunakan data yang disediaka n oleh Compute Optimizer untuk menampilk an detail jumlah instans EC2 Kurang dari yang ditentukan dan Lebih dari yang ditentukan, temuan optimasi, detail harga sesuai permintaa n, dan rekomendasi untuk tipe instans dan harga instans. Untuk melihat data iniExplorer, aktifkan Compute Optimizer. Untuk informasi selengkapnya, lihat Memulai dengan Manajer Sesi AWS Compute Optimizer.

Tugas	Informasi
Mengaktifkan AWS Security Hub	Explorermenggunakan data yang disediaka n oleh Security Hub untuk mengisi widget dengan informasi tentang temuan keamanan Anda. Untuk melihat data iniExplorer, aktifkan integrasi Security Hub. Untuk informasi lebih lanjut, lihat Apa yang dimaksud dengan AWS Security Hub.

Mengonfigurasi peran dan izin untuk Systems Manager Explorer

Penyiapan Terintegrasi secara otomatis membuat dan mengonfigurasiAWS Identity and Access Management (IAM) role untukAWS Systems Manager Explorer danAWS Systems ManagerOpsCenter. Jika Anda menyelesaikan Penyiapan Terintegrasi, Anda tidak perlu melakukan tugas tambahan apa pun untuk mengonfigurasi peran dan izin untukExplorer. Namun, Anda harus mengonfigurasi izin untukOpsCenter, seperti yang dijelaskan nanti dalam topik ini.

Daftar Isi

Tentang peran yang dibuat oleh penyiapan terpadu

Penyiapan Terintegrasi membuat dan mengonfigurasi peran berikut untuk bekerja denganExplorer danOpsCenter.

- `AWSServiceRoleForAmazonSSM`: Menyediakan akses keAWS sumber daya yang dikelola atau digunakan oleh Systems Manager.
- `OpsItem-CWE-Role`: Memungkinkan CloudWatch Acara dan EventBridge untuk membuatOpsItems dalam menanggapi peristiwa umum.
- `AWSServiceRoleForAmazonSSM_AccountDiscovery`: Memungkinkan Systems Manager memanggil lainnyaLayanan AWS untuk menemukanAkun AWS informasi saat menyinkronkan data. Untuk informasi selengkapnya tentang peran ini, lihat Tentang peran `AWSServiceRoleForAmazonSSM_AccountDiscovery`.

- `AmazonSSMExplorerExport`: MemungkinkanExplorer untuk mengekspor OpsData ke file yang dipisahkan dengan koma (CSV).

Tentang peran **`AWSServiceRoleForAmazonSSM_AccountDiscovery`**

Jika Anda mengonfigurasiExplorer untuk menampilkan data dari beberapa akun dan Wilayah dengan menggunakanAWS Organizations dan sinkronisasi data sumber daya, lalu Systems Manager membuat peran terkait layanan. Systems Manager menggunakan peran ini untuk mendapatkan informasi tentang Akun AWS Anda dalam AWS Organizations. Peran tersebut menggunakan kebijakan izin berikut.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "organizations:DescribeAccount",
            "organizations:DescribeOrganization",
            "organizations:ListAccounts",
            "organizations:ListAWSServiceAccessForOrganization",
            "organizations:ListChildren",
            "organizations:ListParents"
         ],
         "Resource":"*"
      }
   ]
}
```

Untuk informasi selengkapnya tentang peran `AWSServiceRoleForAmazonSSM_AccountDiscovery` ini, lihat Menggunakan peran untuk mengumpulkan Akun AWS informasi untuk OpsCenter dan Explorer.

Mengonfigurasi izin untuk Systems ManagerOpsCenter

Setelah Anda menyelesaikan Penyiapan Terintegrasi, Anda harus mengonfigurasi pengguna, grup, atau izin peran untuk mengonfigurasi pengguna, grup, atau izin peranOpsCenter.

Sebelum Anda memulai

Anda dapat mengonfigurasi AndaOpsCenter untuk membuat dan mengelolaOpsItems di beberapa akun atau hanya satu akun. Jika Anda mengonfigurasiOpsCenter untuk membuat dan mengelolaOpsItems di beberapa akun, akunAWS Organizations manajemen dapat membuat, melihat, atau mengeditOpsItems di akun lain secara manual. Jika diperlukan, Anda juga dapat memilih akun administrator yang didelegasikan oleh Systems Manager untuk dibuat dan dikelolaOpsItems di akun anggota. Namun, jika Anda mengkonfigurasiOpsCenter untuk satu akun, Anda hanya dapat melihat atau mengeditOpsItems di akun tempatOpsItems dibuat. Anda tidak dapat membagikan atau mentransferOpsItems di seluruhAkun AWS. Untuk alasan ini, kami menyarankan Anda mengonfigurasi izin untukOpsCenter diAkun AWS yang digunakan untuk menjalankanAWS beban kerja Anda. Anda kemudian dapat membuat pengguna atau grup di akun tersebut. Dengan cara ini, beberapa insinyur operasi atau profesional IT dapat membuat, melihat, dan mengeditOpsItems dalam yang samaAkun AWS.

ExplorerdanOpsCenter gunakan operasi API berikut. Anda dapat menggunakan semua fiturExplorer danOpsCenter jika pengguna, grup, atau peran Anda memiliki akses ke tindakan ini. Anda juga dapat membuat akses yang lebih ketat, seperti yang dijelaskan nanti di bagian ini.

- CreateOpsItem
- CreateResourceDataSync
- DescribeOpsItems
- DeleteResourceDataSync
- GetOpsItem
- GetOpsSummary
- ListResourceDataSync
- UpdateOpsItem
- UpdateResourceDataSync

Jika ingin, Anda dapat menentukan izin hanya-baca dengan menambahkan kebijakan inline berikut ke akun, grup, atau peran Anda.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:GetOpsItem",
```

```
        "ssm:GetOpsSummary",
        "ssm:DescribeOpsItems",
        "ssm:GetServiceSetting",
        "ssm:ListResourceDataSync"
      ],
      "Resource": "*"
    }
  ]
}
```

Untuk informasi selengkapnya tentang membuat dan mengedit kebijakan IAM, lihat Membuat Kebijakan IAM dalam Panduan Pengguna IAM. Untuk informasi tentang cara menetapkan kebijakan ini ke grup IAM, lihatMelampirkan Kebijakan ke Grup IAM.

Buat izin menggunakan yang berikut dan tambahkan ke pengguna, grup, atau peran Anda:

```
{
  "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": "*"
    }
  ]
}
```

Tergantung pada aplikasi identitas yang Anda gunakan di organisasi Anda, Anda dapat memilih salah satu opsi berikut untuk mengonfigurasi akses pengguna.

Untuk menyediakan akses, tambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup diAWS IAM Identity Center:

Buat set izin. Ikuti petunjuk di Buat set izin di PanduanAWS IAM Identity Center Pengguna.

- Pengguna yang dikelola dalam IAM melalui penyedia identitas:

  Membuat peran untuk federasi identitas. Ikuti petunjuk dalam Membuat peran untuk penyedia identitas pihak ketiga (federasi) di Panduan Pengguna IAM.

- Pengguna IAM:
  - Buat peran yang dapat diasumsikan pengguna Anda. Ikuti petunjuk dalam Membuat peran untuk pengguna IAM di Panduan Pengguna IAM.
  - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk dalam Menambahkan izin ke pengguna (konsol) di Panduan Pengguna IAM.

Membatasi akses keOpsItems dengan menggunakan tag

Anda juga dapat membatasi aksesOpsItems dengan menggunakan kebijakan inline IAM yang menentukan tag. Berikut adalah contoh yang menentukan kunci tag Departemen dan nilai tag Keuangan. Dengan kebijakan ini, pengguna hanya dapat memanggil operasi GetOpsItemAPI untuk melihatOpsItems yang sebelumnya ditandai dengan Kunci=Departemen dan Nilai=Keuangan. Pengguna tidak dapat melihat yang lainOpsItems.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:GetOpsItem"
             ],
      "Resource": "*"
      ,
      "Condition": { "StringEquals": { "ssm:resourceTag/Department": "Finance" } }
    }
  ]
}
```

Berikut adalah contoh yang menentukan operasi API untuk melihat dan memperbaruiOpsItems. Kebijakan ini juga menentukan dua set pasangan nilai kunci tag: Departemen-Keuangan dan Project-Unity.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "ssm:GetOpsItem",
            "ssm:UpdateOpsItem"
         ],
         "Resource":"*",
         "Condition":{
            "StringEquals":{
               "ssm:resourceTag/Department":"Finance",
               "ssm:resourceTag/Project":"Unity"
            }
         }
      }
   ]
}
```

Untuk informasi tentang menambahkan tag ke sebuahOpsItem, lihatBuat OpsItems secara manual.

Mengaktifkan aturan default

Penyiapan Terintegrasi secara otomatis mengonfigurasi aturan default berikut di Amazon EventBridge. Peraturan iniOpsItemsdiAWS Systems Manager OpsCenter. Jika Anda tidak mau EventBridge untuk membuatOpsItemsuntuk peristiwa berikut, hapus opsi ini di Penyiapan Terintegrasi. Jika mau, Anda dapat menentukanOpsCentersebagai target spesifik EventBridge peristiwa. Untuk informasi selengkapnya, lihat Konfigurasikan EventBridge aturan untuk dibuat OpsItems. Anda juga dapat menonaktifkan aturan default kapan saja di halaman Pengaturan.

> **Important**
>
> Anda tidak dapat mengeditKategoridanKepelikannilai untuk aturan default tetapi Anda dapat mengedit nilai-nilai iniOpsItemsdibuat dari aturan default.

Rule	Category	Severity
CWE rules (11)		
SSMOpsItems-Autoscaling-instance-launch-failure	Availability	2-High
SSMOpsItems-Autoscaling-instance-termination-failure	Availability	2-High
SSMOpsItems-EBS-snapshot-copy-failed	Availability	2-High
SSMOpsItems-EBS-snapshot-creation-failed	Availability	2-High
SSMOpsItems-EBS-volume-performance-issue	Performance	3-Medium
SSMOpsItems-EC2-issue	Availability	2-High
SSMOpsItems-EC2-scheduled-change	Availability	3-Medium
SSMOpsItems-RDS-issue	Availability	2-High
SSMOpsItems-RDS-scheduled-change	Availability	3-Medium
SSMOpsItems-SSM-maintenance-window-execution-failed	Availability	3-Medium
SSMOpsItems-SSM-maintenance-window-execution-timedout	Availability	2-High

Mengkonfigurasi OpsData sumber

Penyiapan Terintegrasi mengaktifkan sumber data berikut yang mengisiExplorer widget.

- AWS SupportCenter (Anda harus memiliki paket Support Business atau Enterprise untuk mengaktifkan sumber ini.)
- AWS Compute Optimizer(Anda harus memiliki paket Support Business atau Enterprise untuk mengaktifkan sumber ini.)
- KepatuhanState Manager asosiasi Systems Manager
- Kepatuhan AWS Config
- Systems ManagerOpsCenter
- KepatuhanPatch Manager patch Systems Manager
- Amazon Elastic Compute Cloud (Amazon EC2)
- Inventaris Systems Manager
- AWS Trusted Advisor(Anda harus memiliki paket Support Business atau Enterprise untuk mengaktifkan sumber ini.)
- AWS Security Hub

Menentukan kunci tag

Saat menyiapkan AWS Systems Manager Explorer, Anda dapat menentukan hingga lima kunci tag pelaporan. Kunci tag ini seharusnya sudah ada di sumber daya AWS. Ini bukan kunci tag baru. Setelah menambahkan kunci ke sistem, Anda kemudian dapat memfilterOpsItemsdiExplorerdengan menggunakan kunci tanda ini.

> **Note**
>
> Anda juga dapat menentukan kunci tag pelaporan di halaman Pengaturan.

Menyiapkan Systems Manager Explorer untuk menampilkan data dari beberapa akun dan Wilayah

AWS Systems Managermenggunakan pengalaman penyiapan terintegrasi untuk membantu Anda memulaiAWS Systems ManagerPenjelajahdan AWS Systems Manager OpsCenter. Setelah menyelesaikan Pengaturan Terpadu,ExplorerdanOpsCentersecara otomatis menyinkronkan data. Lebih khusus lagi, kemampuan ini menyinkronkan OpsData danOpsItemsuntukAkun AWSdanWilayah AWSAnda gunakan saat Anda menyelesaikan Pengaturan Terpadu. Jika Anda ingin mengumpulkan OpsData danOpsItemsdari akun dan Wilayah lain, Anda harus membuat sinkronisasi data sumber daya, seperti yang dijelaskan dalam topik ini.

> **Note**
>
> Untuk informasi selengkapnya tentang Penyiapan Terintegrasi, lihat Memulai dengan Systems Manager Explorer danOpsCenter.

Tentang sinkronisasi data sumber daya untukExplorer

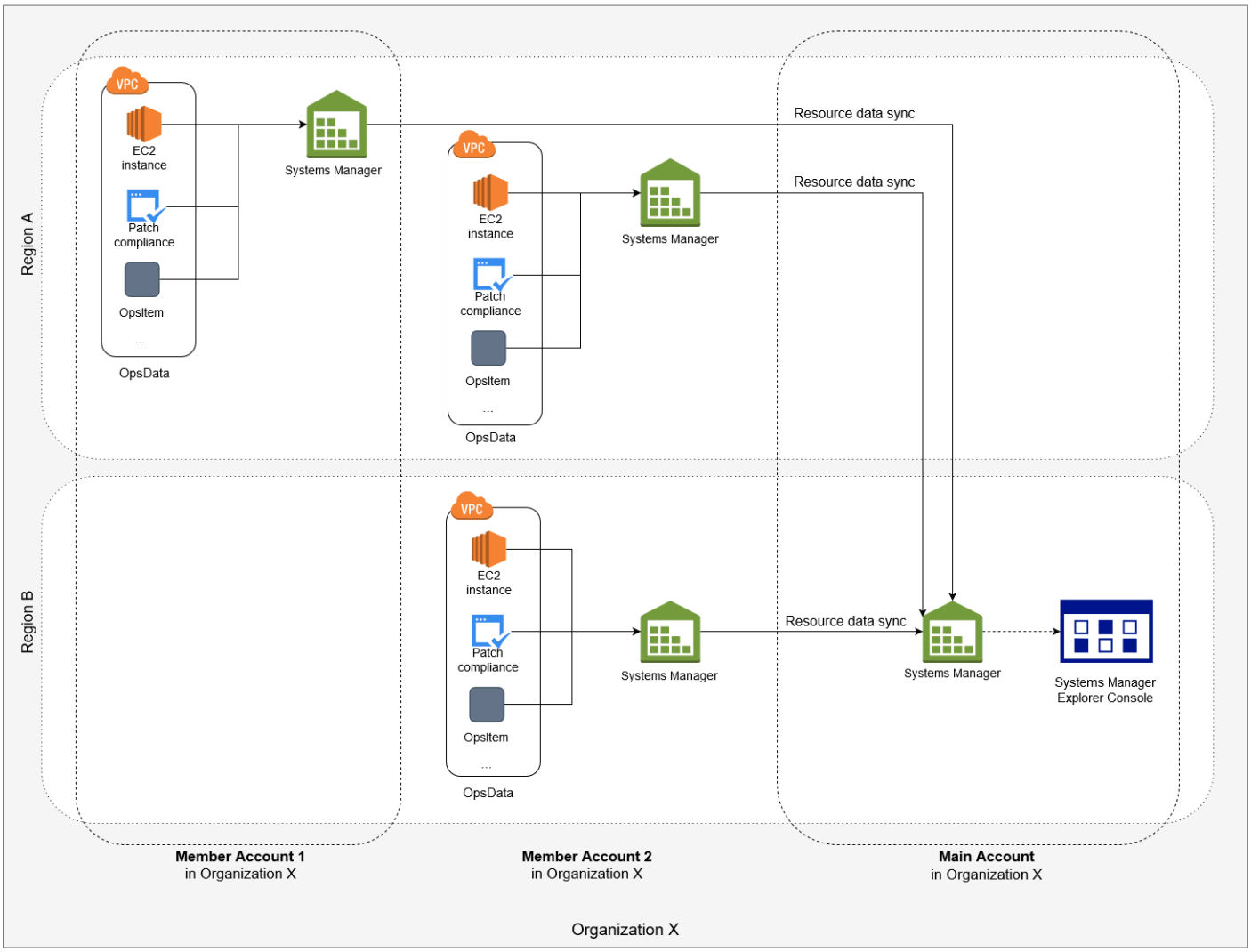
Sinkronisasi data sumber daya untukExplorermenawarkan dua opsi agregasi:

- Akun tunggal/beberapa wilayah:Anda dapat mengkonfigurasiExploreruntuk agregatOpsItemsdan OpsData data dari beberapaWilayah AWS, tetapi kumpulan data terbatas pada saat iniAkun AWS.
- Multiple-akun/Multiple-Region:Anda dapat mengkonfigurasiExploreruntuk mengumpulkan data dari beberapaWilayah AWSdan akun. Opsi ini mengharuskan Anda menyiapkan dan mengonfigurasi AWS Organizations. Setelah Anda mengatur dan mengkonfigurasiAWS Organizations, Anda

dapat mengumpulkan data diExploreroleh unit organisasi (OU) atau untuk seluruh organisasi. Manajer Sistem mengumpulkan data ke dalamAWS Organizationsakun manajemen sebelum menampilkannya diExplorer. Untuk informasi lebih lanjut, lihat Apa AWS Organizations? dalam AWS Organizations Panduan Pengguna.

> **Warning**
>
> Jika Anda mengkonfigurasiExploreruntuk mengumpulkan data dari suatu organisasi diAWS Organizations, sistem memungkinkan OpsData di semua akun anggota dalam organisasi. Mengaktifkan OpsData sumber di semua akun anggota meningkatkan jumlah panggilan keOpsCenterAPI seperti CreateOpsItem dan GetOpsSummary. Anda dikenakan biaya untuk panggilan ke tindakan API ini.

Diagram berikut menunjukkan sinkronisasi data sumber daya yang dikonfigurasi untuk bekerja dengan AWS Organizations. Dalam skenario ini, pengguna memiliki dua akun yang ditentukan dalam AWS Organizations. Sinkronisasi data sumber daya mengumpulkan data dari akun dan beberapaWilayah AWSke dalamAWS Organizationsakun manajemen tempat itu kemudian ditampilkanExplorer.



Tentang sinkronisasi data sumber daya beberapa akun dan Wilayah

Bagian ini menjelaskan detail penting tentang beberapa akun dan beberapa sinkronisasi data sumber daya Wilayah yang menggunakan AWS Organizations. Secara khusus, informasi di bagian ini berlaku jika Anda memilih salah satu opsi berikut di halaman Membuat sinkronisasi data sumber daya:

- Memasukkan semua akun dari konfigurasi AWS Organizations saya
- Pilih unit organisasi di AWS Organizations

Jika Anda tidak berencana untuk menggunakan salah satu opsi berikut, Anda dapat melewati bagian ini.

Saat Anda membuat sinkronisasi data sumber daya di konsol SSM, jika Anda memilih salah satuAWS Organizationsopsi, maka Manajer Sistem secara otomatis memungkinkan semua

OpsData sumber di Wilayah yang dipilih untuk semuaAkun AWSdi organisasi Anda (atau di unit organisasi yang dipilih). Misalnya, bahkan jika Anda belum berbalikExplorerpada suatu Wilayah, jika Anda memilihAWS Organizationsopsi untuk sinkronisasi data sumber daya Anda, lalu Manajer Sistem secara otomatis mengumpulkan OpsData dari wilayah tersebut. Untuk membuat sinkronisasi data sumber daya tanpa mengizinkan OpsData sumber, tentukanEnableAllOpsDataSourcessebagai false saat membuat sinkronisasi data. Untuk informasi lebih lanjut, lihat EnableAllOpsDataSourcesdiReferensi API Manajer Sistem Amazon EC2.

Jika Anda tidak memilih salah satuAWS Organizationspilihan untuk sinkronisasi data sumber daya, maka Anda harus menyelesaikan Pengaturan Terpadu di setiap akun dan Wilayah di mana Anda inginkanExploreruntuk mengakses data. Jika Anda tidak,Explorertidak akan ditampilkan OpsData danOpsItemsuntuk akun dan Wilayah di mana Anda tidak menyelesaikan Penyiapan Terpadu.

Jika Anda menambahkan akun anak ke organisasi Anda,Explorersecara otomatis memungkinkan semua OpsData sumber untuk akun tersebut. Jika, di lain waktu, Anda menghapus akun anak dari organisasi Anda,Explorerterus mengumpulkan OpsData dari akun.

Jika Anda memperbarui sinkronisasi data sumber daya yang ada yang menggunakan salah satuAWS Organizationspilihan, sistem meminta Anda untuk menyetujui koleksi semua OpsData sumber untuk semua akun dan Wilayah yang terpengaruh oleh perubahan.

Jika Anda menambahkan layanan baru keAkun AWS, dan jikaExplorermengumpulkan OpsData untuk layanan itu, Manajer Sistem secara otomatis mengonfigurasiExploreruntuk mengumpulkan itu OpsData. Misalnya, jika organisasi Anda tidak menggunakanAWS Trusted Advisorketika Anda sebelumnya membuat sinkronisasi data sumber daya, tetapi organisasi Anda mendaftar untuk layanan ini,Explorersecara otomatis memperbarui sinkronisasi data sumber daya Anda untuk mengumpulkan ini OpsData.

Important

Perhatikan informasi penting berikut tentang beberapa akun dan sinkronisasi data sumber daya Wilayah:

- Menghapus sinkronisasi data sumber daya tidak mematikan OpsData sumber diExplorer.
- Untuk melihat OpsData danOpsItemsdari beberapa akun, Anda harus memilikiAWS Organizations Semua fiturmode dihidupkan dan Anda harus masuk keAWS Organizationsakun manajemen.

Membuat sinkronisasi data sumber daya

Sebelum Anda mengonfigurasi sinkronisasi data sumber daya untukExplorer, perhatikan detail berikut.

- Explorermendukung maksimal lima sinkronisasi data sumber daya.
- Setelah Anda membuat sinkronisasi data sumber daya untuk suatu Wilayah, Anda tidak dapat mengubah opsi akun untuk sinkronisasi tersebut. Misalnya, jika Anda membuat sinkronisasi di Wilayah us-east-2 (Ohio) dan memilih opsi Sertakan hanya akun saat ini, Anda tidak dapat mengedit sinkronisasi itu nanti dan memilih opsi Sertakan semua akun dari konfigurasi AWS Organizations saya. Sebaliknya, Anda harus menghapus sinkronisasi data sumber daya pertama, lalu membuat sinkronisasi baru. Untuk informasi selengkapnya, lihat Menghapus sinkronisasi data sumber daya untuk Systems Manager Explorer
- OpsData dilihat diExploreradalah read-only.

Gunakan prosedur berikut untuk membuat sinkronisasi data sumber dayaExplorer.

Untuk membuat sinkronisasi data sumber daya

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Explorer.
3. Pilih Pengaturan.
4. Di bagian Mengonfigurasi sinkronisasi data sumber daya, pilih Membuat sinkronisasi data sumber daya.
5. Untuk Nama sinkronisasi data sumber daya, masukkan nama.
6. Di bagian Tambahkan akun, pilih opsi.

> **Note**
>
> Untuk menggunakan salah satuAWS Organizationspilihan, Anda harus login keAWS Organizationsakun manajemen atau Anda harus masuk keExplorerakun administrator yang didelegasikan. Untuk informasi selengkapnya tentang akun administrator yang didelegasikan, lihat Mengonfigurasi administrator yang didelegasikan.

7. Di bagian Wilayah yang akan disertakan, pilih salah satu opsi berikut.

- Pilih Semua wilayah saat ini dan yang akan datang untuk menyinkronkan data secara otomatis dari semua Wilayah AWS saat ini dan setiap Wilayah baru yang online di masa mendatang.
- Pilih Semua wilayah untuk menyinkronkan data secara otomatis dari semua Wilayah AWS saat ini.
- Pilih satu per satu Wilayah yang ingin Anda sertakan.

8. Pilih Membuat sinkronisasi data sumber daya.

Sistem dapat memakan waktu beberapa menit untuk mengisiExplorerdengan data setelah Anda membuat sinkronisasi data sumber daya. Anda dapat melihat sinkronisasi dengan memilihnya dariPilih sinkronisasi data sumber dayadaftar diExplorer.

Mengonfigurasi administrator yang didelegasikan

Jika Anda menggabungkan dataAWS Systems Manager Explorer dari beberapaWilayah AWS dan dengan menggunakan sinkronisasi data sumber daya denganAWS Organizations, kami menyarankan agar Anda mengonfigurasi administrator yang didelegasikan untukExplorer.

Administrator yang didelegasikan dapat menggunakan API sinkronisasi dataExplorer sumber daya berikut menggunakan konsol, SDK,AWS Command Line Interface (AWS CLI), atauAWS Tools for Windows PowerShell:

- CreateResourceDataSync
- DeleteResourceDataSync
- ListResourceDataSync
- UpdateResourceDataSync

Administrator yang didelegasikan dapat membuat maksimum lima sinkronisasi data sumber daya untuk seluruh organisasi. Sinkronisasi data sumber daya yang dibuat oleh administrator yang didelegasikan hanya tersedia di akun administrator yang didelegasikan. Anda tidak dapat melihat sinkronisasi atau data gabungan di akun manajemen AWS Organizations.

Untuk informasi selengkapnya tentang sinkronisasi data sumber daya, lihat Menyiapkan Systems Manager Explorer untuk menampilkan data dari beberapa akun dan Wilayah. Untuk informasi lebih lanjut tentang AWS Organizations, lihat Apa itu AWS Organizations? di Panduan Pengguna AWS Organizations.

Topik

Mengonfigurasi administratorExplorer yang didelegasikan

Gunakan prosedur berikut untuk mendaftarkanExplorer administrator.

Untuk mendaftarkan administrator yangExplorer didelegasikan

1. Masuk ke akun manajemen AWS Organizations Anda.
2. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
3. Di panel navigasi, pilih Explorer.
4. Pilih Pengaturan.
5. DiExplorer bagian Administrator yang diperlukan dan opsi akses layanan. Jika perlu, pilih tombol Buat peran dan Aktifkan akses untuk mengonfigurasi opsi ini.
6. Untuk ID Akun, masukkan ID Akun AWS. Akun ini harus menjadi akun anggota di AWS Organizations.
7. Pilih Daftar administrator yang didelegasikan.

Administrator yang didelegasikan sekarang memiliki akses ke Sertakan semua akun dari konfigurasi AWS Organizations saya dan Pilih unit organisasi di opsi AWS Organizations di halaman Membuat sinkronisasi data sumber daya.

Membatalkan pendaftaran administrator yangExplorer didelegasikan

Gunakan prosedur berikut untuk membatalkan pendaftaran administrator yangExplorer didelegasikan. Akun administrator yang didelegasikan hanya dapat dibatalkan pendaftarannya oleh akun manajemen AWS Organizations. Saat akun administrator yang didelegasikan dibatalkan pendaftarannya, sistem akan menghapus semua sinkronisasi data sumber daya AWS Organizations yang dibuat oleh administrator yang didelegasikan.

Untuk membatalkan pendaftaran administrator yangExplorer didelegasikan

1. Masuk ke akun manajemen AWS Organizations Anda.
2. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
3. Di panel navigasi, pilih Explorer.

4. Pilih Pengaturan.
5. DiExplorer bagian Administrator yang didelegasikan untuk, pilih Deregister. Sistem menampilkan peringatan.
6. Masukkan ID akun dan pilih Hapus.

Akun tidak lagi memiliki akses ke operasi API sinkronisasi data sumber daya AWS Organizations. Sistem ini menghapus semua sinkronisasi data sumber daya AWS Organizations yang dibuat oleh akun.

Menggunakan Systems Manager Explorer

Bagian ini berisi informasi tentang cara menyesuaikan AWS Systems Manager Explorer dengan mengubah tata letak widget dan dengan mengubah data yang ditampilkan di dasbor.

Konten

Mengedit aturan default untukOpsItems

Saat Anda menyelesaikan Penyiapan Terintegrasi, sistem mengizinkan banyak aturan di Amazon EventBridge. Aturan-aturan ini secara otomatis membuatOpsItemsdiAWS Systems Manager OpsCenter.AWS Systems Manager Explorerkemudian menampilkan informasi agregat tentangOpsItems.

Setiap aturan menyertakan nilai Kategori dan Kepelikan yang telah ditetapkan sebelumnya. Saat sistemnya membuatOpsItemsdari suatu peristiwa, secara otomatis menetapkan presetKategoridanKepelikan.

> **Important**
>
> Anda tidak dapat mengeditKategoridanKepelikannilai untuk aturan default tetapi Anda dapat mengedit nilai-nilai iniOpsItemsdibuat dari aturan default.

Rule	Category	Severity
CWE rules (11)		
SSMOpsItems-Autoscaling-instance-launch-failure	Availability	2-High
SSMOpsItems-Autoscaling-instance-termination-failure	Availability	2-High
SSMOpsItems-EBS-snapshot-copy-failed	Availability	2-High
SSMOpsItems-EBS-snapshot-creation-failed	Availability	2-High
SSMOpsItems-EBS-volume-performance-issue	Performance	3-Medium
SSMOpsItems-EC2-issue	Availability	2-High
SSMOpsItems-EC2-scheduled-change	Availability	3-Medium
SSMOpsItems-RDS-issue	Availability	2-High
SSMOpsItems-RDS-scheduled-change	Availability	3-Medium
SSMOpsItems-SSM-maintenance-window-execution-failed	Availability	3-Medium
SSMOpsItems-SSM-maintenance-window-execution-timedout	Availability	2-High

Untuk mengedit aturan default untuk membuatOpsItems

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Explorer.
3. Pilih Pengaturan.
4. DiOpsItemsaturanbagian, pilihedit.
5. Perluas Aturan CWE.
6. Kosongkan kotak centang di samping aturan yang tidak ingin Anda gunakan.
7. Gunakan daftar Kategori dan Kepelikan untuk mengubah informasi ini untuk suatu aturan.
8. Pilih Simpan.

Perubahan Anda berlaku saat berikutnya sistem membuatOpsItem.

Mengedit sumber data Systems Manager Explorer

AWS Systems ManagerExplorer menampilkan data dari sumber berikut. Anda dapat mengeditExplorerpengaturan untuk menambah atau menghapus sumber data:

- Amazon Elastic Compute Cloud (Amazon EC2)

- AWS Systems Manager OpsCenter
- AWS Systems Manager Patch Managerkepatuhan patch
- AWS Systems Manager State Managerkepatuhan asosiasi
- AWS Trusted Advisor
- AWS Compute Optimizer
- Kasus pusat AWS Support
- Kepatuhan aturan dan sumber daya AWS Config
- Temuan AWS Security Hub

> **Note**
>
> - Untuk melihatAWS SupportKasus pusat diExplorerAnda harus memiliki akun Enterprise atau Business yang diatur denganAWS Support.
> - Anda tidak dapat mengonfigurasiExploreruntuk berhenti menampilkanOpsCenter OpsItemdata.

Sebelum Anda memulai

Verifikasi bahwa Anda telah menyiapkan dan mengonfigurasi layanan yang mengisiExplorerwidget dengan data. Untuk informasi selengkapnya, lihat Menyiapkan layanan terkait.

Untuk mengedit sumber data

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Explorer.
3. Pilih Pengaturan.
4. DiOpsData narasumberbagian, pilihedit.
5. PerluasOpsData narasumber.
6. Tambahkan atau hapus satu atau beberapa sumber.
7. Pilih Simpan.

Menyesuaikan tampilan dan menggunakan filter

Anda dapat menyesuaikan tata letak widget diAWS Systems Manager Explorer dengan menggunakan drag-and-drop kemampuan. Anda juga dapat menyesuaikan OpsData danOpsItems ditampilkan di bawahExplorer dengan menggunakan filter, seperti yang dijelaskan dalam topik ini.

Sebelum Anda memulai

Sebelum Anda menyesuaikan tata letak widget, verifikasi bahwa widget yang ingin Anda lihat saat ini ditampilkanExplorer. Untuk melihat beberapa widget diExplorer (seperti widgetAWS Config kepatuhan), Anda harus mengaktifkannya di halaman dasbor Konfigurasi.

Untuk mengaktifkan widget untuk ditampilkan diExplorer

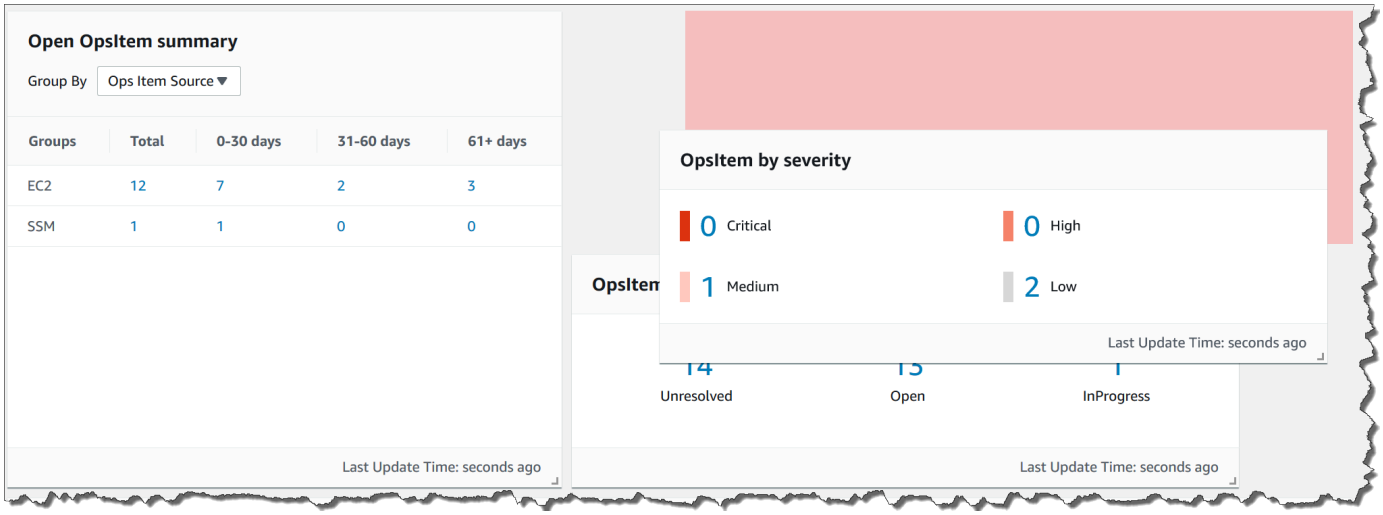
1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Explorer.
3. Pilih Tindakan dasbor, Konfigurasi dasbor.
4. Pilih tab Konfigurasi Dasbor.
5. Pilih Aktifkan semua atau aktifkan widget atau sumber data individual.
6. Pilih Exploreruntuk melihat perubahan Anda.

Menyesuaikan tata letak widget

Gunakan prosedur berikut untuk menyesuaikan tata letak widget diExplorer.

Untuk menyesuaikan tata letak widget

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Explorer.
3. Pilih widget yang ingin Anda pindahkan.
4. Klik dan tahan nama widget, lalu seret ke lokasi barunya.



5. Ulangi proses ini untuk setiap widget yang ingin Anda ubah posisinya.

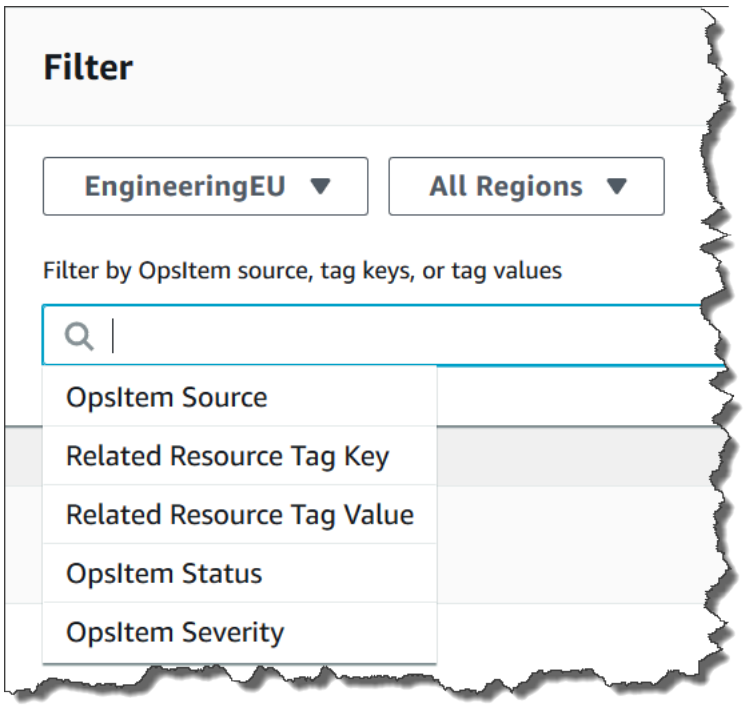
Jika Anda memutuskan tidak menyukai tata letak barunya, pilih Atur ulang tata letak untuk memindahkan semua widget kembali ke lokasi aslinya.

Menggunakan filter untuk mengubah data yang ditampilkan diExplorer

Secara default,Explorer menampilkan data untuk saat iniAkun AWS dan Wilayah saat ini. Jika Anda membuat satu atau beberapa sinkronisasi data sumber daya, Anda dapat menggunakan filter untuk mengubah sinkronisasi yang aktif. Anda kemudian dapat memilih untuk menampilkan data untuk Wilayah tertentu atau semua Wilayah. Anda juga dapat menggunakan bilah Pencarian untuk memfilter Op-tag kunci yang berbedaOpsItem.

Untuk mengubah data yang ditampilkan di dalamnyaExplorer dengan menggunakan filter

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Explorer.
3. Di bagian Filter, gunakan daftar Pilih sinkronisasi data sumber daya untuk memilih sinkronisasi.
4. Gunakan daftar Wilayah untuk memilih Wilayah AWS tertentu atau pilih Semua Wilayah.
5. Pilih bilah Pencarian, kemudian pilih kriteria untuk memfilter data.



6. Tekan Enter.

Explorermempertahankan opsi filter yang Anda pilih jika Anda menutup dan membuka kembali halaman.

Menghapus sinkronisasi data sumber daya untuk Systems Manager Explorer

MasukAWS Systems ManagerExplorer, Anda dapat agregat OpsData danOpsItemsdari akun dan Wilayah lain dengan membuat sinkronisasi data sumber daya.

Anda tidak dapat mengubah opsi akun untuk sinkronisasi data sumber daya. Misalnya, jika Anda membuat sinkronisasi di Wilayah us-east-2 (Ohio) dan memilih opsi Sertakan hanya akun saat ini, Anda tidak dapat mengedit sinkronisasi itu nanti dan memilih opsi Sertakan semua akun dari konfigurasi AWS Organizations saya. Sebaliknya, Anda harus menghapus sinkronisasi data sumber daya, dan membuat yang baru, seperti dijelaskan dalam prosedur berikut.

Untuk menghapus sinkronisasi data sumber daya

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Explorer.
3. Pilih Pengaturan.

4. Di bagian Mengonfigurasi sinkronisasi data sumber daya, pilih sinkronisasi data sumber daya yang ingin Anda hapus.
5. Pilih Hapus.

Menerima temuan dariAWS Security HubdiExplorer

AWS Security Hubmemberikan pandangan komprehensif tentang keadaan keamanan Anda diAWS. Layanan ini mengumpulkan data keamanan, yang disebuttemuan, dari seberangAkun AWS, layanan, dan produk pihak ketiga yang didukung. Temuan Security Hub dapat membantu Anda memeriksa lingkungan Anda terhadap standar industri keamanan dan praktik terbaik, menganalisis tren keamanan Anda, dan mengidentifikasi masalah keamanan prioritas tertinggi.

Security Hub mengirimkan temuan ke Amazon EventBridge, yang menggunakan aturan acara untuk mengirim temuan keExplorer. Setelah mengaktifkan integrasi, seperti yang dijelaskan di sini, Anda dapat melihat temuan Security Hub diExplorerwidget dan lihat detail temuan diOpsCenter OpsItems. Widget menyediakan ringkasan semua temuan Security Hub berdasarkan tingkat keparahan. Temuan baru di Security Hub biasanya terlihat diExplorerdalam hitungan detik setelah diciptakan.

Warning

Perhatikan informasi penting berikut:

- Explorerterintegrasi denganOpsCenter, kemampuan Manajer Sistem. Setelah Anda mengaktifkanExplorerIntegrasi dengan Security Hub,OpsCentersecara otomatis menciptakanOpsItemsuntuk temuan Security Hub. Tergantung pada AndaAWSlingkungan, memungkinkan integrasi dapat menghasilkan sejumlah besarOpsItems, dengan biaya.

  Sebelum melanjutkan, baca tentangOpsCenterIntegrasi dengan Security Hub Topik ini mencakup detail spesifik tentang bagaimana perubahan dan pembaruan temuan danOpsItemsdibebankan ke akun Anda. Untuk informasi selengkapnya, lihat AWS Security Hub. UntukOpsCenterinformasi harga, lihatAWS Systems ManagerHarga.
- Jika Anda membuat sinkronisasi data sumber daya diExplorersaat masuk ke akun administrator, integrasi Security Hub diaktifkan secara otomatis untuk administrator dan semua akun anggota dalam sinkronisasi. Setelah diaktifkan,OpsCentersecara otomatis menciptakanOpsItemsuntuk temuan Security Hub, dengan biaya tertentu. Untuk informasi selengkapnya tentang membuat sinkronisasi data sumber daya, lihatMenyiapkan Systems Manager Explorer untuk menampilkan data dari beberapa akun dan Wilayah.

Jenis-jenis temuan yangExplorermenerima

Explorermenerimasemua temuandari Security Hub. Anda dapat melihat semua temuan berdasarkan tingkat keparahan diExplorerwidget saat Anda mengaktifkan pengaturan default Security Hub. Secara default,ExplorermenghasilkanOpsItemsuntuk temuan kritis dan tingkat keparahan tinggi. Anda dapat mengkonfigurasi secara manualExploreruntuk membuatOpsItemsuntuk temuan tingkat keparahan sedang dan rendah.

PadahalExplorertidak membuatOpsItemsuntuk temuan informasi, Anda dapat melihat data operasi informasi (OpsData) di widget ringkasan temuan Security Hub.Explorermenghasilkan OpsData untuk semua temuan terlepas dari tingkat keparahannya. Untuk informasi selengkapnya tentang tingkat keparahan Security Hub, lihatKeparahandiAWS Security HubReferensi API.

Mengaktifkan integrasi

Bagian ini menjelaskan cara mengaktifkan dan mengkonfigurasiExploreruntuk mulai menerima temuan Security Hub.

Sebelum Anda memulai

Selesaikan tugas-tugas berikut sebelum Anda mengkonfigurasiExploreruntuk mulai menerima temuan Security Hub.

- Aktifkan dan konfigurasikan Hub Keamanan. Untuk informasi selengkapnya, lihat Menyiapkan Security Hub di Panduan Pengguna AWS Security Hub.
- Masuk ke akun manajemen AWS Organizations. Manajer Sistem membutuhkan akses keAWS Organizationsuntuk membuatOpsItemsdari temuan Security Hub. Setelah masuk ke akun manajemen, Anda diminta untuk memilihAktifkan aksestombol padaExplorer Konfigurasikan dasbortab, seperti yang dijelaskan dalam prosedur berikut. Jika Anda tidak masuk keAWS Organizationsakun manajemen, Anda tidak dapat mengizinkan akses danExplorertidak bisa membuatOpsItemsdari temuan Security Hub.

Untuk mulai menerima temuan Security Hub

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Explorer.
3. Pilih Pengaturan.
4. Pilih tab Konfigurasi dasbor.
5. Pilih AWS Security Hub.

6. Pilih penggeser Dinonaktifkan untuk mengaktifkan AWS Security Hub.

   Temuan kritis dan tingkat keparahan tinggi ditampilkan secara default. Untuk menampilkan temuan tingkat keparahan sedang dan rendah, pilihDinonaktifkanslider di sebelahSedang, Rendah.
7. DiOpsItemsdibuat oleh temuan Security Hubbagian, pilihAktifkan akses. Jika Anda tidak melihat tombol ini, masuk ke akun manajemen AWS Organizations dan kembali ke halaman ini untuk memilih tombol.

Cara melihat temuan dari Security Hub

Prosedur berikut menjelaskan cara melihat temuan Security Hub.

Untuk melihat temuan Security Hub

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Explorer.
3. Temukan widget Ringkasan temuan AWS Security Hub. Ini akan menampilkan temuan Security Hub Anda. Anda dapat memilih tingkat keparahan untuk melihat deskripsi rinci tentang yang sesuaiOpsItem.

Cara berhenti menerima temuan

Prosedur berikut menjelaskan cara berhenti menerima temuan Security Hub.

Untuk berhenti menerima temuan Security Hub

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Explorer.
3. Pilih Pengaturan.
4. Pilih tab Konfigurasi dasbor.
5. Pilih penggeser Diaktifkan mematikan AWS Security Hub.

MengeksporOpsData dari Systems ManagerExplorer

Anda dapat mengekspor 5.000OpsData item sebagai file nilai yang dipisahkan koma (.csv) ke bucket Amazon Simple Storage Service (Amazon S3) dariAWS Systems Manager Explorer. Explorer

menggunakan Runbook Ops Data Ekspor ke S3 otomatisasi untuk mengeksporOpsData. Saat Anda mengeksporOpsData, sistem menampilkan halaman runbook otomatisasi tempat Anda dapat menentukan detail, seperti AssumeRole, nama bucket Amazon S3, SNS topic Arn, dan kolom yang akan diekspor.

Untuk informasi tentang runbook`AWS-ExportOpsDataToS3` otomatisasi, lihat AWS-ExportOpsDataToS3.

Untuk mengeksporOpsData:

- Langkah 1: Menentukan topik SNS
- Langkah 2: (Opsional) Mengonfigurasi ekspor data
- Langkah 3: MengeksporOpsData

Langkah 1: Menentukan topik SNS

Ketika Anda mengonfigurasi ekspor data, Anda harus menentukan topik Amazon Simple Notification Service (Amazon SNS) yang ada di Wilayah AWS yang sama tempat Anda ingin mengekspor data. Systems Manager mengirimkan notifikasi ke topik Amazon SNS ketika ekspor selesai. Untuk informasi tentang cara membuat topik Amazon SNS, lihat Membuat topik Amazon SNS.

Langkah 2: (Opsional) Mengonfigurasi ekspor data

Anda dapat mengonfigurasi pengaturan ekspor data dari halaman Pengaturan atau Ekspor Data Operasi ke Bucket S3.

Untuk mengkonfigurasi ekspor data dariExplorer

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Explorer.
3. Pilih Pengaturan.
4. Di bagian Mengonfigurasi ekspor data, pilih Edit.
5. Untuk mengunggah file ekspor data ke bucket Amazon S3 yang ada, pilih Pilih bucket S3 yang ada dan pilih bucket dari daftar.

   Untuk mengunggah file ekspor data ke bucket Amazon S3 baru, pilih Buat bucket S3 baru dan masukkan nama yang ingin Anda gunakan untuk bucket baru.

> **Note**
>
> Anda hanya dapat mengedit nama bucket Amazon S3 dan ARN topik Amazon SNS dari halaman tempat Anda mengonfigurasi pengaturan tersebut untuk pertama kalinyaExplorer. Jika Anda menyiapkan bucket Amazon S3 dan ARN topik Amazon SNS dari halaman Pengaturan, maka Anda hanya dapat mengubah pengaturan tersebut dari halaman Pengaturan.

6. Untuk Pilih topik Amazon SNS, pilih topik yang ingin Anda beri tahu saat ekspornya selesai.
7. Pilih Create (Buat).

Langkah 3: MengeksporOpsData

Saat Anda mengeksporExplorer data, Systems Manager membuatAWS Identity and Access Management (IAM) role bernama`AmazonSSMExplorerExportRole`. Peran ini menggunakan kebijakan IAM berikut.

```
{
      "Version": "2012-10-17",
      "Statement": [
          {
              "Effect": "Allow",
              "Action": [
                  "s3:PutObject"
              ],
              "Resource": [
                 "arn:aws:s3:::{{ExportDestinationS3BucketName}}/*"
              ]
          },
          {
              "Effect": "Allow",
              "Action": [
                  "s3:GetBucketAcl"
              ],
              "Resource": [
                 "arn:aws:s3:::{{ExportDestinationS3BucketName}}"
              ]
          },
          {
              "Effect": "Allow",
```

```
            "Action": [
                "sns:Publish"
            ],
            "Resource": [
                "{{SnsTopicArn}}"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:DescribeLogGroups",
                "logs:DescribeLogStreams"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogGroup",
                "logs:PutLogEvents",
                "logs:CreateLogStream"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetOpsSummary"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

Peran tersebut mencakup entitas kepercayaan berikut.

```
{
```

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

Untuk mengeksporOpsData dariExplorer

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Explorer.
3. Pilih Tabel Ekspor.

> **Note**
>
> Saat Anda mengeksporOpsData untuk pertama kalinya, sistem akan membuat peran asumsikan untuk ekspor. Anda tidak dapat memodifikasi peran asumsi default.

4. Untuk Amazon S3 Bucket Name, pilih bucket yang ada. Anda dapat memilih Buat untuk membuat bucket Amazon S3 jika diperlukan. Jika Anda tidak dapat mengubah nama bucket S3, itu berarti Anda mengonfigurasi nama bucket dari halaman Pengaturan. Anda hanya dapat mengubah nama bucket dari halaman Pengaturan.

> **Note**
>
> Anda hanya dapat mengedit nama bucket Amazon S3 dan ARN topik Amazon SNS dari halaman tempat Anda mengonfigurasi pengaturan tersebut untuk pertama kalinyaExplorer.

5. Untuk SNS Topic Arn, pilih ARN topik Amazon SNS yang ada untuk memberi tahu saat unduhan selesai.

Jika Anda tidak dapat mengubah topik ARN Amazon SNS, itu berarti Anda mengonfigurasi ARN topik Amazon SNS dari halaman Pengaturan. Anda hanya dapat mengubah topik ARN dari halaman Pengaturan.

6. (Opsional) Untuk Pesan Sukses SNS, tentukan pesan sukses yang ingin ditampilkan saat ekspor berhasil diselesaikan.
7. Pilih Submit (Kirim). Sistem menavigasi ke halaman sebelumnya dan menampilkan pesan Klik untuk melihat status proses ekspor. Lihat detail.

   Anda dapat memilih Lihat detail untuk melihat status runbook dan kemajuan dalam Otomatisasi Systems Manager.

Anda sekarang dapat mengeksporOpsData dariExplorer bucket Amazon S3 tertentu.

Jika Anda tidak dapat mengekspor data dengan menggunakan prosedur ini, verifikasi bahwa pengguna, atau peran, atau peran, verifikasi bahwa pengguna, atau peran, atau`iam:DeletePolicyVersion` peran.`iam:CreatePolicyVersion` Untuk informasi tentang menambahkan tindakan ini ke pengguna, atau peran, lihat Mengedit Kebijakan IAM dalam Panduan Pengguna IAM.

Memecahkan Masalah Systems Manager Explorer

Topik ini mencakup informasi tentang cara memecahkan masalah umum dengan AWS Systems Manager Explorer.

Tidak dapat memfilterAWS sumber dayaExplorer setelah memperbarui tag pada halaman Pengaturan

Jika Anda memperbarui kunci tag atau pengaturan data lainnya di Explorer, sistem dapat memakan waktu hingga enam jam untuk menyinkronkan data berdasarkan perubahan Anda.

AWS OrganizationsOpsi di halaman Buat sinkronisasi data sumber daya berwarna abu-abu

Opsi Sertakan semua akun dari konfigurasi AWS Organizations sayadan Pilih unit organisasi di AWS Organizations di halaman Buat sinkronisasi data sumber daya hanya tersedia jika Anda telah menyiapkan dan mengonfigurasi AWS Organizations. Jika Anda menyiapkan dan mengonfigurasiAWS Organizations, baik akunAWS Organizations manajemen atau administrator yangExplorer didelegasikan dapat membuat sinkronisasi data sumber daya yang menggunakan opsi ini.

Untuk informasi selengkapnya, lihat Menyiapkan Systems Manager Explorer untuk menampilkan data dari beberapa akun dan Wilayah dan Mengonfigurasi administrator yang didelegasikan.

Explorertidak menampilkan data sama sekali

- Verifikasi bahwa Anda menyelesaikan Penyiapan Terintegrasi di setiap akun dan Wilayah di mana AndaExplorer ingin mengakses dan menampilkan data. Jika tidak, tidakExplorer akan ditampilkan OpsData danOpsItems untuk akun dan Wilayah tersebut tempat Anda tidak menyelesaikan Penyiapan Terintegrasi. Untuk informasi selengkapnya, lihat Memulai dengan Systems Manager Explorer danOpsCenter.
- Saat menggunakanExplorer untuk melihat data dari beberapa akun dan Wilayah, verifikasi bahwa Anda telah masuk ke akunAWS Organizations manajemen. Untuk melihat OpsData danOpsItems dari beberapa akun dan Wilayah, Anda harus masuk ke akun ini.

Widget tentang instans Amazon EC2 tidak menampilkan data

Jika widget tentang instans Amazon Elastic Compute Cloud (Amazon EC2), seperti instans Jumlah instans,Instans terkelola, danInstans oleh AMI tidak menampilkan data, verifikasi hal berikut:

- Verifikasi bahwa Anda menunggu beberapa menit. OpsData memerlukan waktu beberapa menit untuk ditampilkanExplorer setelah Anda menyelesaikan Penyiapan Terintegrasi.
- Verifikasi bahwa Anda telahAWS Config mengonfigurasi perekam konfigurasi. Explorermenggunakan data yang disediakan oleh perekamAWS Config konfigurasi untuk mengisi widget dengan informasi tentang instans EC2 Anda. Untuk informasi selengkapnya, lihat Mengelola Perekam Konfigurasi.
- Verifikasi bahwa OpsData sumber Amazon EC2 aktif di halaman Pengaturan. Juga, verifikasi bahwa lebih dari 6 jam telah berlalu sejak Anda mengaktifkan perekam konfigurasi atau sejak Anda membuat perubahan pada instans. Systems Manager dapat memakan waktu hingga enam jam untuk menampilkan data dariAWS Config di widgetExplorer EC2 setelah Anda mengaktifkan perekam konfigurasi atau membuat perubahan pada instans Anda.
- Ketahui bahwa jika sebuah instans dihentikan atau diakhiri,Explorer berhenti menampilkan instans tersebut setelah 24 jam.
- Verifikasi bahwa Anda berada di yang benarWilayah AWS tempat Anda mengonfigurasi instans Amazon EC2 Anda. Explorertidak menampilkan data tentang instans on-premise.
- Jika Anda mengonfigurasi sinkronisasi data sumber daya untuk beberapa akun dan Wilayah, pastikan Anda masuk ke akun manajemen Organizations.

Widget patch tidak menampilkan data

Widget Instans yang tidak sesuai untuk patching hanya menampilkan data tentang instans patch yang tidak sesuai. Widget ini tidak menampilkan data jika instans Anda sesuai. Jika Anda mencurigai bahwa Anda memiliki instans yang tidak sesuai, verifikasi bahwa Anda telah menyiapkan dan mengonfigurasi patching Systems Manager dan menggunakanAWS Systems ManagerPatch Manager untuk memeriksa kepatuhan patch Anda. Untuk informasi selengkapnya, lihat AWS Systems Manager Patch Manager.

Masalah Miscellaneous

Explorertidak mengizinkan Anda untuk mengedit atau memulihkanOpsItems:OpsItems dilihat di seluruh akun atau Wilayah bersifat hanya-baca. OpsItem tersebut hanya dapat diperbarui dan dipulihkan dari akun atau Wilayah asalnya.

AWS Systems Manager OpsCenter

OpsCenter, kemampuanAWS Systems Manager, menyediakan lokasi pusat di mana insinyur operasi dan profesional TI dapat mengelola item kerja operasional (OpsItems) yang terkait dengan AWS sumber daya. An OpsItemadalah masalah operasional atau gangguan yang membutuhkan investigasi dan remediasi. Dengan menggunakanOpsCenter, Anda dapat melihat data investigasi kontekstual tentang masing-masingOpsItem, termasuk sumber daya terkait OpsItems dan terkait. Anda juga dapat menjalankan runbook Systems Manager Automation untuk menyelesaikannyaOpsItems.

Masing-masing OpsItem mencakup informasi yang relevan, seperti nama dan ID AWS sumber daya yang dihasilkanOpsItem, yang diperlukan untuk menyelesaikan suatu peristiwa. Ketika Anda mengatur OpsCenter dan mengintegrasikannya dengan yang lainLayanan AWS, itu dapat membuat OpsItems secara otomatis. Jika terintegrasi dengan layanan ini, OpsCenter menampilkan informasi dari AWS ConfigAWS CloudTrail, dan Amazon EventBridge untuk membantu Anda menyelidikiOpsItem. Akibatnya, Anda tidak perlu menavigasi antar halaman konsol untuk penyelidikan Anda.

Anda dapat menggunakannya OpsCenter untuk menyelidiki dan memulihkan masalah dengan node terkelola lokal yang dikonfigurasi untuk Systems Manager. Untuk informasi selengkapnya tentang pengaturan dan konfigurasi server on-premise dan mesin virtual untuk Systems Manager, lihat Menyiapkan Manajer Sistem untuk lingkungan hybrid dan multicloud.

Anda dapat bekerja OpsCenter dengan menggunakan konsol Systems Manager, AWS Command Line Interface (AWS CLI)AWS Tools for PowerShell, atau AWS SDK pilihan Anda. Dengan menggunakan kebijakan AWS Identity and Access Management (IAM), Anda dapat memutuskan anggota organisasi mana yang dapat membuat, melihat, membuat daftar, dan memperbaruiOpsItems. Anda dapat menetapkan tag OpsItems dan kemudian membuat kebijakan IAM yang memberikan akses ke pengguna dan grup berdasarkan tag.

> **Note**
>
> Ada biaya untuk digunakanOpsCenter. Untuk selengkapnya, lihat AWS Systems ManagerHarga.
> Anda dapat melihat kuota untuk semua kemampuan Systems Manager dalam kuota layanan Systems Manager di. Referensi Umum Amazon Web Services Kecuali dinyatakan lain, setiap kuota bersifat khusus per Wilayah.

OpsCenteralur kerja

Untuk mengatur dan bekerja dengan OpsCenter untuk memulihkanOpsItems, lakukan langkah-langkah berikut:

1. Mengatur OpsCenter. Anda juga dapat mengatur OpsCenter untuk mengelola secara terpusat OpsItems di seluruh akun.
2. Integrasikan OpsCenter dengan yang lain Layanan AWS. OpsCenterdapat berintegrasi dengan Amazon CloudWatch, Amazon CloudWatch Application Insights, Amazon EventBridge, Amazon DevOps Guru,AWS Config,AWS Security Hub, danAWS Systems Manager Incident Manager.
3. Buat OpsItems. Anda dapat membuat OpsItems secara otomatis dan manual.
4. Kelola OpsItems dengan menambahkan konteks pada sumber daya terkait, terkaitOpsItems, dan data operasional, dan menghapus duplikat. OpsItems
5. Remediasi OpsItems menggunakan runbook Automation Systems Manager.

Mengatur OpsCenter

AWS Systems Managermenggunakan pengalaman penyiapan terintegrasi untuk membantu Anda memulai OpsCenter danExplorer, yang merupakan kemampuan Systems Manager. Exploreradalah dasbor operasi yang dapat disesuaikan yang melaporkan informasi tentang sumber daya AndaAWS. Dalam dokumentasi ini, Explorer dan OpsCenter setup disebut Integrated Setup.

Anda harus menggunakan Integrated Setup untuk mengatur OpsCenter denganExplorer. Pengaturan Terintegrasi hanya tersedia di konsol AWS Systems Manager. Anda tidak dapat mengatur Explorer dan secara OpsCenter terprogram. Untuk informasi selengkapnya, lihat Memulai dengan Systems Manager Explorer danOpsCenter.

Aturan default diaktifkan oleh penyiapan

Saat menyiapkanOpsCenter, Anda mengaktifkan aturan default di Amazon EventBridge yang dibuat secara otomatisOpsItems. Tabel berikut menjelaskan EventBridge aturan default yang secara otomatis membuatOpsItems. Anda dapat menonaktifkan EventBridge aturan di halaman OpsCenter Pengaturan di bawah OpsItematuran.

> **Important**
>
> Akun Anda dikenakan biaya untuk OpsItems dibuat berdasarkan aturan default. Untuk informasi selengkapnya, lihat AWS Systems Manager Harga.

Nama aturan	Deskripsi
SSMOpsItems-Autoscaling-instance-launch-failure	Aturan ini dibuat OpsItems saat peluncuran instans penskalaan otomatis EC2 gagal.
SSMOpsItems-Autoscaling-instance-termination-failure	Aturan ini dibuat OpsItems saat penghentian instans penskalaan otomatis EC2 gagal.
SSMOpsItems-EBS-snapshot-copy-failed	Aturan ini dibuat OpsItems ketika sistem gagal menyalin snapshot Amazon Elastic Block Store (Amazon EBS).
SSMOpsItems-EBS-snapshot-creation-failed	Aturan ini dibuat OpsItems ketika sistem gagal membuat snapshot Amazon EBS.
SSMOpsItems-EBS-volume-performance-issue	Aturan ini sesuai dengan aturan AWS Health pelacakan. Aturan dibuat setiap OpsItems kali ada masalah kinerja dengan volume Amazon EBS (event kesehatan =`AWS_EBS_DEGRADED_EBS_VOLUME_PERFORMANCE`).

Nama aturan	Deskripsi
SSMOpsItems-EC2-issue	Aturan ini sesuai dengan aturan AWS Health pelacakan untuk kejadian tak terduga yang memengaruhi AWS layanan atau sumber daya. Aturan dibuat OpsItems ketika, misalnya, layanan mengirimkan komunikasi tentang masalah operasional yang menyebabkan degradasi layanan atau untuk meningkatkan kesadaran tentang masalah tingkat sumber daya lokal. Misalnya, aturan ini membuat OpsItem untuk acara berikut:`AWS_EC2_OPERATIONAL_ISSUE` .
SSMOpsItems-EC2-scheduled-change	Aturan ini sesuai dengan aturan AWS Health pelacakan. AWSdapat menjadwalkan acara untuk instans Anda, seperti me-reboot, menghentikan, atau memulai instance. Aturan dibuat OpsItems untuk acara terjadwal EC2. Untuk informasi selengkapnya tentang peristiwa terjadwal, lihat Acara terjadwal untuk instans Anda di Panduan Pengguna Amazon EC2 untuk Instans Linux.

Nama aturan	Deskripsi
SSMOpsItems-RDS-issue	Aturan ini sesuai dengan aturan AWS Health pelacakan untuk kejadian tak terduga yang memengaruhi AWS layanan atau sumber daya. Aturan dibuat OpsItems ketika, misalnya, layanan mengirimkan komunikasi tentang masalah operasional yang menyebabkan degradasi layanan atau untuk meningkat kan kesadaran tentang masalah tingkat sumber daya lokal. Misalnya, aturan ini menciptakan sebuah OpsItem untuk peristiwa berikut:`AWS_RDS_MYSQL_DATABASE_CRAS HING_REPEATEDLY` ,`AWS_RDS_E XPORT_TASK_FAILED` , dan`AWS_RDS_C ONNECTIVITY_ISSUE` .
SSMOpsItems-RDS-scheduled-change	Aturan ini sesuai dengan aturan AWS Health pelacakan. Aturan dibuat OpsItems untuk acara terjadwal Amazon RDS. Acara terjadwal memberikan informasi tentang perubahan mendatang pada sumber daya Amazon RDS Anda. Beberapa acara mungkin menyarankan Anda mengambil tindakan untuk menghinda ri gangguan layanan. Peristiwa lain terjadi secara otomatis tanpa tindakan apa pun di pihak Anda. Sumber daya Anda mungkin tidak tersedia untuk sementara selama aktivitas perubahan terjadwal. Misalnya, aturan ini membuat OpsItem untuk peristiwa berikut: `AWS_RDS_SYSTEM_UPGRADE_SCHEDULED` dan`AWS_RDS_MAINTENANCE_SCHEDULED` . Untuk informasi selengkapnya tentang acara terjadwal, lihat Kategori jenis acara di Panduan AWS Health Pengguna.

Nama aturan	Deskripsi
SSMOpsItems-SSM-maintenance-window-execution-failed	Aturan ini dibuat OpsItems ketika pemrosesan jendela pemeliharaan Systems Manager gagal.
SSMOpsItems-SSM-maintenance-window-execution-timedout	Aturan ini dibuat OpsItems saat peluncuran jendela pemeliharaan Systems Manager habis waktu.

Menyiapkan OpsCenter

Gunakan prosedur berikut untuk mengaturOpsCenter.

Untuk mengatur OpsCenter

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter.
3. Di OpsCenter halaman beranda, pilih Memulai.
4. Pada halaman OpsCenter penyiapan, pilih Aktifkan opsi ini untuk Explorer mengonfigurasi AWS Config dan CloudWatch peristiwa Amazon untuk dibuat secara otomatis OpsItems berdasarkan aturan dan peristiwa yang umum digunakan. Jika Anda tidak memilih opsi ini, OpsCenter tetap dinonaktifkan.

   > **Note**
   >
   > Amazon EventBridge (sebelumnya Amazon CloudWatch Events) menyediakan semua fungsionalitas CloudWatch Acara dan beberapa fitur baru, seperti bus acara khusus, sumber acara pihak ketiga, dan registri skema.

5. Pilih AktifkanOpsCenter.

Setelah Anda mengaktifkanOpsCenter, Anda dapat melakukan hal berikut dari Pengaturan:

- Buat CloudWatch alarm menggunakan tombol Buka CloudWatch konsol. Untuk informasi selengkapnya, lihat Konfigurasikan CloudWatch alarm untuk dibuatOpsItems.
- Aktifkan wawasan operasional. Untuk informasi selengkapnya, lihat Menganalisis wawasan operasional untuk mengurangi OpsItems.

- Aktifkan alarm AWS Security Hub temuan. Untuk informasi selengkapnya, lihat AWS Security Hub.

Konten

(Opsional) Menyiapkan OpsCenter untuk mengelola secara terpusat OpsItems di seluruh akun

Anda dapat menggunakan Manajer Sistem OpsCenter untuk mengelola secara terpusat OpsItems di beberapa Akun AWS dalam yang dipilihWilayah AWS. Fitur ini tersedia setelah Anda menyiapkan organisasiAWS Organizations. AWS Organizationsadalah layanan manajemen akun yang memungkinkan Anda untuk mengkonsolidasikan beberapa AWS akun ke dalam organisasi yang Anda buat dan kelola secara terpusat. AWS Organizationsmencakup manajemen akun dan kemampuan penagihan konsolidasi yang memungkinkan Anda untuk lebih memenuhi kebutuhan anggaran, keamanan, dan kepatuhan bisnis Anda. Untuk informasi lebih lanjut, lihat Apa ituAWS Organizations? di AWS OrganizationsUser Guide

Pengguna yang termasuk dalam akun AWS Organizations manajemen dapat mengatur akun administrator yang didelegasikan untuk Manajer Sistem. Dalam konteksOpsCenter, administrator yang didelegasikan dapat membuat, mengedit, dan melihat OpsItems di akun anggota. Administrator yang didelegasikan juga dapat menggunakan runbook Otomasi Manajer Sistem untuk menyelesaikan secara massal OpsItems atau memperbaiki masalah dengan AWS sumber daya yang dihasilkan. OpsItems

> **Note**
>
> Anda hanya dapat menetapkan satu akun sebagai administrator yang didelegasikan untuk Manajer Sistem. Untuk informasi selengkapnya, lihat Menyiapkan administrator yang didelegasikan untuk Manajer Sistem.

Manajer Sistem menawarkan metode berikut untuk menyiapkan OpsCenter untuk mengelola secara terpusat OpsItems di beberapaAkun AWS.

- Pengaturan Cepat: Pengaturan Cepat, kemampuan Manajer Sistem, menyederhanakan pengaturan dan tugas konfigurasi untuk kemampuan Manajer Sistem. Untuk informasi selengkapnya, lihat AWS Systems Manager Quick Setup.

  Pengaturan Cepat untuk OpsCenter membantu Anda menyelesaikan tugas-tugas berikut untuk mengelola OpsItems di seluruh akun:

  - Mendaftarkan akun sebagai administrator yang didelegasikan (jika administrator yang didelegasikan belum ditunjuk)
  - Membuat kebijakan dan peran yang diperlukan AWS Identity and Access Management (IAM)
  - Menentukan organisasi atau unit AWS Organizations organisasi (oU) tempat administrator yang didelegasikan dapat mengelola seluruh akun OpsItems

  Untuk informasi selengkapnya, lihat (Opsional) Konfigurasi OpsCenter untuk mengelola OpsItems seluruh akun dengan menggunakan Quick Setup.

  > **Note**
  >
  > Pengaturan Cepat tidak tersedia di semua Wilayah AWS tempat Manajer Sistem saat ini tersedia. Jika Pengaturan Cepat tidak tersedia di Wilayah tempat Anda ingin menggunakannya untuk mengonfigurasi OpsCenter untuk mengelola secara terpusat OpsItems di beberapa akun, maka Anda harus menggunakan metode manual. Untuk melihat daftar Wilayah AWS di mana Quick Setup tersedia, lihatKetersediaan Quick Setup di Wilayah AWS.

- Pengaturan manual: Jika Pengaturan Cepat tidak tersedia di Wilayah tempat Anda ingin mengonfigurasi OpsCenter untuk mengelola secara terpusat OpsItems di seluruh akun, Anda dapat menggunakan prosedur manual untuk melakukannya. Untuk informasi selengkapnya, lihat (Opsional) Menyiapkan OpsCenter untuk mengelola secara terpusat OpsItems di seluruh akun.

(Opsional) Konfigurasi OpsCenter untuk mengelola OpsItems seluruh akun dengan menggunakan Quick Setup

Quick Setup, kemampuanAWS Systems Manager, menyederhanakan mengatur dan konfigurasi tugas untuk kemampuan Systems Manager. Quick Setupuntuk OpsCenter membantu Anda menyelesaikan tugas-tugas berikut untuk mengelola OpsItems seluruh akun:

- Menentukan akun administrator yang didelegasikan

- Membuat kebijakan dan peran yang diperlukan AWS Identity and Access Management (IAM)
- Menentukan AWS Organizations organisasi, atau subset akun anggota, tempat administrator yang didelegasikan dapat mengelola seluruh akun OpsItems

Saat Anda mengonfigurasi OpsCenter untuk mengelola OpsItems seluruh akun dengan menggunakan Pengaturan Cepat, Quick Setup buat sumber daya berikut di akun yang ditentukan. Sumber daya ini memberikan izin akun yang ditentukan untuk bekerja dengan OpsItems dan menggunakan runbook Otomasi untuk memperbaiki masalah dengan menghasilkan AWS OpsItems sumber daya.

Sumber daya	Akun
`AWSServiceRoleForAmazonSSM_AccountDiscovery` AWS Identity and Access Management(IAM) peran terkait layanan Untuk informasi selengkapnya tentang peran ini, lihat Menggunakan peran untuk mengumpulkan Akun AWS informasi untuk OpsCenter dan Explorer.	AWS Organizationsakun manajemen dan akun administrator yang didelegasikan
`OpsItem-CrossAccountManagementRole` Peran IAM `AWS-SystemsManager-AutomationAdministrationRole` Peran IAM	Akun administrator yang didelegasikan
`OpsItem-CrossAccountExecutionRole` Peran IAM `AWS-SystemsManager-AutomationExecutionRole` Peran IAM `AWS::SSM::ResourcePolicy` Kebijakan sumber daya Manajer Sistem untuk OpsItem grup default (`OpsItemGroup`)	Semua akun AWS Organizations anggota

> **Note**
>
> Jika sebelumnya Anda mengonfigurasi OpsCenter untuk mengelola OpsItems seluruh akun menggunakan metode manual, Anda harus menghapus tumpukan atau kumpulan tumpukan yang dibuat selama Langkah 4 dan 5 dari proses tersebut. AWS CloudFormation Jika sumber daya tersebut ada di akun Anda saat Anda menyelesaikan prosedur berikut, Quick Setup gagal mengonfigurasi OpsItem manajemen lintas akun dengan benar.

Mengkonfigurasi OpsCenter untuk mengelola OpsItems seluruh akun dengan menggunakan Quick Setup

1. Masuk ke AWS Management Console menggunakan akun AWS Organizations manajemen.
2. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
3. Di panel navigasi, pilih Quick Setup.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Quick Setupdi panel navigasi.
4. Pilih tab Library.
5. Gulir ke bawah dan cari ubin OpsCenterkonfigurasi. Pilih Create (Buat).
6. Pada Quick Setup OpsCenter halaman, di bagian Administrator yang didelegasikan, masukkan ID akun. Jika Anda tidak dapat mengedit bidang ini, maka akun administrator yang didelegasikan telah ditentukan untuk Manajer Sistem.
7. Di bagian Target, pilih opsi. Jika Anda memilih Kustom, pilih unit organisasi (OU) yang ingin Anda kelola OpsItems di seluruh akun.
8. Pilih Create (Buat).

Quick Setupmembuat OpsCenter konfigurasi dan menyebarkan AWS sumber daya yang diperlukan ke oU yang ditunjuk.

> **Note**
>
> Jika Anda tidak ingin mengelola OpsItems di beberapa akun, Anda dapat menghapus konfigurasi dariQuick Setup. Saat Anda menghapus konfigurasi, Quick Setup menghapus kebijakan dan peran IAM berikut yang dibuat saat konfigurasi awalnya diterapkan:
>
> - `OpsItem-CrossAccountManagementRole`dari akun administrator yang didelegasikan
> - `OpsItem-CrossAccountExecutionRole`dan `SSM::ResourcePolicy` dari semua akun anggota Organisasi
>
> Quick Setupmenghapus konfigurasi dari semua unit organisasi dan Wilayah AWS di mana konfigurasi awalnya digunakan.

Memecahkan masalah dengan Quick Setup konfigurasi untuk OpsCenter

Bagian ini mencakup informasi untuk membantu Anda memecahkan masalah saat mengonfigurasi pengelolaan lintas akun OpsItem menggunakan. Quick Setup

Topik

- Deployment ke ini StackSets gagal: DelegatedAdmin
- Quick Setupstatus konfigurasi menunjukkan Gagal

Deployment ke ini StackSets gagal: DelegatedAdmin

Saat membuat OpsCenter konfigurasi, Quick Setup menyebarkan dua set AWS CloudFormation tumpukan di akun manajemen Organisasi. Set stack menggunakan awalan berikut:`AWS-QuickSetup-SSMOpsCenter`. Jika Quick Setup menampilkan galat berikut: `Deployment to these StackSets failed: delegatedAdmin` gunakan prosedur berikut untuk memperbaiki masalah ini.

Untuk memecahkan masalah galat StackSets failed:DelegatedAdmin

1. Jika Anda menerima `Deployment to these StackSets failed: delegatedAdmin` kesalahan dalam spanduk merah di Quick Setup konsol, masuk ke akun administrator yang didelegasikan dan Wilayah AWS yang ditunjuk sebagai Wilayah Quick Setup rumah.
2. Buka AWS CloudFormation konsol di https://console.aws.amazon.com/cloudformation.

3. Pilih tumpukan yang dibuat oleh Quick Setup konfigurasi Anda. Nama stack mencakup yang berikut: AWS- QuickSetup OpsCenter -SSM.

   > **Note**
   >
   > Terkadang CloudFormation menghapus penyebaran tumpukan yang gagal. Jika tumpukan tidak tersedia di tabel Stacks, pilih Dihapus dari daftar filter.

4. Lihat alasan Status dan Status. Untuk informasi selengkapnya tentang status tumpukan, lihat Menumpuk kode status di Panduan AWS CloudFormation Pengguna.
5. Untuk memahami langkah pasti yang gagal, lihat tab Peristiwa dan tinjau Status setiap peristiwa. Untuk informasi selengkapnya, lihat Pemecahan Masalah di Panduan AWS CloudFormation Pengguna.

> **Note**
>
> Jika Anda tidak dapat menyelesaikan kegagalan penyebaran menggunakan langkah-langkah CloudFormation pemecahan masalah, hapus konfigurasi dan coba lagi.

Quick Setupstatus konfigurasi menunjukkan Gagal

Jika tabel Rincian konfigurasi pada halaman Rincian konfigurasi menunjukkan status konfigurasi`Failed`, masuk ke Akun AWS dan Wilayah tempat gagal.

Untuk memecahkan masalah Quick Setup kegagalan untuk membuat konfigurasi OpsCenter

1. Masuk ke Akun AWS dan di Wilayah AWS mana kegagalan terjadi.
2. Buka AWS CloudFormation konsol di https://console.aws.amazon.com/cloudformation.
3. Pilih tumpukan yang dibuat oleh Quick Setup konfigurasi Anda. Nama stack mencakup yang berikut: AWS- QuickSetup OpsCenter -SSM.

   > **Note**
   >
   > Terkadang CloudFormation menghapus penyebaran tumpukan yang gagal. Jika tumpukan tidak tersedia di tabel Stacks, pilih Dihapus dari daftar filter.

4. Lihat alasan Status dan Status. Untuk informasi selengkapnya tentang status tumpukan, lihat Menumpuk kode status di Panduan AWS CloudFormation Pengguna.
5. Untuk memahami langkah pasti yang gagal, lihat tab Peristiwa dan tinjau Status setiap peristiwa. Untuk informasi selengkapnya, lihat Pemecahan Masalah di Panduan AWS CloudFormation Pengguna.

Konfigurasi akun anggota menunjukkan ResourcePolicyLimitExceededException

Jika status stack menunjukkan`ResourcePolicyLimitExceededException`, akun sebelumnya telah onboarded ke manajemen OpsCenter lintas akun dengan menggunakan metode manual. Untuk mengatasi masalah ini, Anda harus menghapus AWS CloudFormation tumpukan atau set tumpukan yang dibuat selama Langkah 4 dan 5 proses orientasi manual. Untuk informasi selengkapnya, lihat Menghapus kumpulan tumpukan dan Menghapus tumpukan di AWS CloudFormation konsol di Panduan AWS CloudFormation Pengguna.

(Opsional) Menyiapkan OpsCenter untuk mengelola secara terpusat OpsItems di seluruh akun

Bagian ini menjelaskan cara mengonfigurasi OpsCenter OpsItem manajemen lintas akun secara manual. Meskipun proses ini masih didukung, itu telah digantikan oleh proses yang lebih baru yang menggunakan Systems ManagerQuick Setup. Untuk informasi selengkapnya, lihat (Opsional) Konfigurasi OpsCenter untuk mengelola OpsItems seluruh akun dengan menggunakan Quick Setup.

Anda dapat mengatur akun pusat untuk membuat manual OpsItems untuk akun anggota, dan mengelola serta memulihkannya. OpsItems Akun pusat dapat berupa akun AWS Organizations manajemen, atau akun manajemen dan akun administrator yang didelegasikan Systems Manager. AWS Organizations Kami menyarankan Anda menggunakan akun administrator yang didelegasikan Systems Manager sebagai akun pusat. Anda hanya dapat menggunakan fitur ini setelah Anda mengkonfigurasiAWS Organizations.

DenganAWS Organizations, Anda dapat mengkonsolidasikan beberapa Akun AWS ke dalam organisasi yang Anda buat dan kelola secara terpusat. Pengguna akun pusat dapat membuat OpsItems untuk semua akun anggota yang dipilih secara bersamaan, dan mengelolanyaOpsItems.

Gunakan proses di bagian ini untuk mengaktifkan prinsip layanan Systems Manager di Organizations and configure AWS Identity and Access Management (IAM) izin untuk bekerja dengan OpsItems lintas akun.

Topik

- Sebelum Anda memulai

- Langkah 1: Membuat sinkronisasi data sumber daya
- Langkah 2: Mengaktifkan prinsip layanan Systems Manager di AWS Organizations
- Langkah 3: Membuat peran AWSServiceRoleForAmazonSSM_AccountDiscovery terkait layanan
- Langkah 4: Mengonfigurasi izin untuk bekerja dengan seluruh akun OpsItems
- Langkah 5: Mengonfigurasi izin untuk bekerja dengan sumber daya terkait di seluruh akun

> **Note**
>
> Hanya OpsItems jenis `/aws/issue` yang didukung saat bekerja di OpsCenter seluruh akun.

Sebelum Anda memulai

Sebelum Anda mengatur OpsCenter untuk bekerja dengan OpsItems seluruh akun, pastikan bahwa Anda telah menyiapkan hal-hal berikut:

- Akun administrator yang didelegasikan oleh Systems Manager. Untuk informasi selengkapnya, lihat Mengonfigurasi administrator yang didelegasikan.
- Satu organisasi diatur dan dikonfigurasi dalam Organizations. Untuk informasi selengkapnya, lihat Membuat dan mengelola organisasi di Panduan AWS Organizations Pengguna.
- Anda mengonfigurasi Otomasi Systems Manager untuk menjalankan runbook otomatisasi di beberapa Wilayah AWS dan AWS akun. Untuk informasi selengkapnya, lihat Menjalankan otomatisasi dalam beberapa Wilayah AWS dan akun.

Langkah 1: Membuat sinkronisasi data sumber daya

Setelah menyiapkan dan mengonfigurasiAWS Organizations, Anda dapat menggabungkan OpsItems seluruh organisasi dengan membuat sinkronisasi data sumber daya. OpsCenter Untuk informasi selengkapnya, lihat Membuat sinkronisasi data sumber daya. Saat Anda membuat sinkronisasi, di bagian Tambah akun, pastikan untuk memilih opsi Sertakan semua akun dari AWS Organizations konfigurasi saya.

Langkah 2: Mengaktifkan prinsip layanan Systems Manager di AWS Organizations

Agar pengguna dapat bekerja dengan OpsItems seluruh akun, prinsipal layanan Systems Manager harus diaktifkanAWS Organizations. Jika sebelumnya Anda mengonfigurasi Systems Manager untuk

skenario multi-akun menggunakan kemampuan lain, prinsipal layanan Systems Manager mungkin sudah dikonfigurasi di Organizations. Jalankan perintah berikut dari AWS Command Line Interface (AWS CLI) untuk memverifikasi. Jika Anda belum mengonfigurasi Systems Manager untuk skenario multi-akun lainnya, lewati ke prosedur berikutnya, Untuk mengaktifkan prinsip layanan Systems Manager. AWS Organizations

Untuk memverifikasi, prinsipal layanan Systems Manager diaktifkan AWS Organizations

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya. Untuk selengkapnya, lihat Menginstal CLI dan Mengkonfigurasi CLI.
2. Unduh versi terbaru dari AWS CLI ke mesin lokal Anda.
3. BukaAWS CLI, dan jalankan perintah berikut untuk menentukan kredensional Anda dan file. Wilayah AWS

   ```
   aws configure
   ```

   Sistem meminta Anda untuk menentukan yang berikut ini. Dalam contoh berikut, ganti setiap *placeholder input pengguna* dengan informasi Anda sendiri.

   ```
   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. Jalankan perintah berikut untuk memverifikasi bahwa prinsipal layanan Systems Manager diaktifkanAWS Organizations.

   ```
   aws organizations list-aws-service-access-for-organization
   ```

   Perintah mengembalikan informasi yang mirip dengan yang ditunjukkan dalam contoh berikut.

   ```
   {
       "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"
        }
    ]
}
```

Untuk mengaktifkan prinsip layanan Systems Manager di AWS Organizations

Jika sebelumnya Anda belum mengonfigurasi prinsip layanan Systems Manager untuk Organizations, gunakan prosedur berikut untuk melakukannya. Untuk informasi selengkapnya tentang perintah ini, lihat enable-aws-service-accessdi Referensi AWS CLI Perintah.

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya. Untuk selengkapnya, lihat Menginstal CLI dan Mengkonfigurasi CLI.
2. Unduh versi terbaru dari AWS CLI ke mesin lokal Anda.
3. BukaAWS CLI, dan jalankan perintah berikut untuk menentukan kredensional Anda dan file. Wilayah AWS

   ```
   aws configure
   ```

   Sistem meminta Anda untuk menentukan yang berikut ini. Dalam contoh berikut, ganti setiap *placeholder input pengguna* dengan informasi Anda sendiri.

   ```
   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. Jalankan perintah berikut untuk mengaktifkan prinsip layanan Systems Manager untukAWS Organizations.

   ```
   aws organizations enable-aws-service-access --service-principal "ssm.amazonaws.com"
   ```

Langkah 3: Membuat peran **`AWSServiceRoleForAmazonSSM_AccountDiscovery`** terkait layanan

Peran terkait layanan seperti peran adalah jenis unik `AWSServiceRoleForAmazonSSM_AccountDiscovery` peran IAM yang ditautkan langsung keLayanan AWS, seperti Systems Manager. Peran terkait layanan telah ditentukan sebelumnya oleh layanan dan mencakup semua izin yang diperlukan layanan untuk memanggil orang lain Layanan AWS atas nama Anda. Untuk informasi selengkapnya tentang peran `AWSServiceRoleForAmazonSSM_AccountDiscovery` terkait layanan, lihat. Izin peran terkait layanan untuk penemuan akun Systems Manager

Gunakan prosedur berikut untuk membuat peran `AWSServiceRoleForAmazonSSM_AccountDiscovery` terkait layanan dengan menggunakan. AWS CLI Untuk informasi selengkapnya tentang perintah yang digunakan dalam prosedur ini, lihat create-service-linked-roledi AWS CLICommand Reference.

Untuk membuat peran **`AWSServiceRoleForAmazonSSM_AccountDiscovery`** terkait layanan

1. Masuk ke akun AWS Organizations manajemen.
2. Saat masuk ke akun manajemen Organizations, jalankan perintah berikut.

```
aws iam create-service-linked-role \
    --aws-service-name accountdiscovery.ssm.amazonaws.com \
    --description "Systems Manager account discovery for AWS Organizations service-
linked role"
```

Langkah 4: Mengonfigurasi izin untuk bekerja dengan seluruh akun OpsItems

Gunakan AWS CloudFormation stacksets untuk membuat kebijakan `OpsItemGroup` sumber daya dan peran eksekusi IAM yang memberikan izin kepada pengguna untuk bekerja dengan OpsItems seluruh akun. Untuk memulai, unduh dan unzip `OpsCenterCrossAccountMembers.zip`file. File ini mencakup `OpsCenterCrossAccountMembers.yaml` AWS CloudFormation file templat. Saat Anda membuat kumpulan tumpukan menggunakan templat ini, CloudFormation secara otomatis membuat kebijakan `OpsItemCrossAccountResourcePolicy` sumber daya dan peran `OpsItemCrossAccountExecutionRole` eksekusi di akun. Untuk informasi selengkapnya tentang membuat kumpulan tumpukan, lihat Membuat kumpulan tumpukan di Panduan AWS CloudFormation Pengguna.

Important

Perhatikan informasi penting berikut tentang tugas ini:

- Anda harus menerapkan stackset saat masuk ke akun manajemen. AWS Organizations
- Anda harus mengulangi prosedur ini saat masuk ke setiap akun yang ingin Anda targetkan untuk digunakan OpsItems di seluruh akun, termasuk akun administrator yang didelegasikan.
- Jika Anda ingin mengaktifkan OpsItems administrasi lintas akun secara berbedaWilayah AWS, pilih Tambahkan semua wilayah di bagian Tentukan wilayah pada templat. OpsItemAdministrasi lintas akun tidak didukung untuk Wilayah keikutsertaan.

Langkah 5: Mengonfigurasi izin untuk bekerja dengan sumber daya terkait di seluruh akun

OpsItemDapat menyertakan informasi terperinci tentang sumber daya yang terkena dampak seperti instance Amazon Elastic Compute Cloud (Amazon EC2) atau bucket Amazon Simple Storage Service (Amazon S3). Peran `OpsItemCrossAccountExecutionRole` eksekusi, yang Anda buat di Langkah 4 sebelumnya, menyediakan izin OpsCenter hanya-baca bagi akun anggota untuk melihat sumber daya terkait. Anda juga harus membuat peran IAM untuk memberikan izin kepada akun manajemen untuk melihat dan berinteraksi dengan sumber daya terkait, yang akan Anda selesaikan dalam tugas ini.

Untuk memulai, unduh dan unzip `OpsCenterCrossAccountManagementRole.zip`file. File ini mencakup `OpsCenterCrossAccountManagementRole.yaml` AWS CloudFormation file templat. Saat Anda membuat tumpukan dengan menggunakan templat ini, CloudFormation secara otomatis membuat peran `OpsCenterCrossAccountManagementRole` IAM di akun. Untuk informasi selengkapnya tentang membuat tumpukan, lihat Membuat tumpukan di AWS CloudFormation konsol di Panduan AWS CloudFormation Pengguna.

Important

Perhatikan informasi penting berikut tentang tugas ini:

- Jika Anda berencana untuk menentukan akun sebagai administrator yang didelegasikanOpsCenter, pastikan untuk menentukannya Akun AWS saat Anda membuat tumpukan.

- Anda harus melakukan prosedur ini saat masuk ke akun AWS Organizations manajemen dan lagi saat masuk ke akun administrator yang didelegasikan.

(Opsional) Siapkan Amazon SNS untuk menerima pemberitahuan tentang OpsItems

Anda dapat mengonfigurasi OpsCenter untuk mengirim notifikasi ke topik Amazon Simple Notification Service (Amazon SNS) saat sistem membuat atau memperbarui OpsItem yang sudah ada. OpsItem

Selesaikan langkah-langkah berikut untuk menerima pemberitahuan untukOpsItems.

- Langkah 1: Membuat dan berlangganan topik Amazon SNS
- Langkah 2: Memperbarui kebijakan akses Amazon SNS
- Langkah 3: Memperbarui kebijakan AWS KMS akses

> **Note**
>
> Jika Anda mengaktifkan enkripsi sisi server AWS Key Management Service (AWS KMS) di Langkah 2, maka Anda harus menyelesaikan Langkah 3. Jika tidak, Anda dapat melewati Langkah 3.

- Langkah 4: Mengaktifkan OpsItems aturan default untuk mengirim pemberitahuan untuk yang baru OpsItems

Langkah 1: Membuat dan berlangganan topik Amazon SNS

Untuk menerima notifikasi, Anda harus membuat dan berlangganan ke topik Amazon SNS. Untuk informasi selengkapnya, lihat Membuat topik Amazon SNS dan Berlangganan topik Amazon SNS di Panduan Pengembang Layanan Pemberitahuan Sederhana Amazon.

> **Note**
>
> Jika Anda menggunakan OpsCenter beberapa Wilayah AWS atau akun, Anda harus membuat dan berlangganan topik Amazon SNS di setiap Wilayah atau akun tempat Anda ingin menerima OpsItem notifikasi.

Langkah 2: Memperbarui kebijakan akses Amazon SNS

Anda harus mengaitkan topik Amazon SNS dengan. OpsItems Gunakan prosedur berikut untuk menyiapkan kebijakan akses Amazon SNS sehingga Systems Manager dapat mempublikasikan OpsItems notifikasi ke topik Amazon SNS yang Anda buat di Langkah 1.

1. Masuk ke AWS Management Console dan buka konsol Amazon SNS di https://console.aws.amazon.com/sns/v3/home.
2. Di panel navigasi, pilih Pengguna.
3. Pilih topik yang Anda buat di Langkah 1, lalu pilih Edit.
4. Perluas Kebijakan akses.
5. Tambahkan blok `Sid` berikut untuk kebijakan yang ada. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri.

```
{
      "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**
>
> Kunci kondisi `aws:SourceAccount` global melindungi dari skenario wakil yang membingungkan. Untuk menggunakan kunci kondisi ini, atur nilainya ke ID akun OpsItem pemilik. Untuk informasi selengkapnya, lihat Deputi Bingung di Panduan Pengguna IAM.

6. Pilih Simpan perubahan.

Sistem sekarang mengirimkan pemberitahuan ke topik Amazon SNS saat OpsItems dibuat atau diperbarui.

> **Important**
>
> Jika Anda mengonfigurasi topik Amazon SNS dengan kunci enkripsi sisi server AWS Key Management Service (AWS KMS) di Langkah 2, selesaikan Langkah 3. Jika tidak, Anda dapat melewati Langkah 3.

Langkah 3: Memperbarui kebijakan AWS KMS akses

Jika Anda mengaktifkan enkripsi AWS KMS sisi server untuk topik Amazon SNS Anda, Anda juga harus memperbarui kebijakan akses AWS KMS key yang Anda pilih saat mengonfigurasi topik. Gunakan prosedur berikut untuk memperbarui kebijakan akses sehingga Systems Manager dapat mempublikasikan OpsItem notifikasi ke topik Amazon SNS yang Anda buat di Langkah 1.

> **Note**
>
> OpsCentertidak mendukung penerbitan OpsItems ke topik Amazon SNS yang dikonfigurasi dengan file. Kunci yang dikelola AWS

1. Buka AWS KMS konsol di https://console.aws.amazon.com/kms.
2. Untuk mengubah Wilayah AWS, gunakan pemilih Wilayah di sudut kanan atas halaman.
3. Di panel navigasi, pilih Kunci yang dikelola pelanggan.
4. Pilih ID kunci KMS yang Anda pilih saat Anda membuat topik.
5. Di bagian Kebijakan Kunci, pilih Beralih ke tampilan kebijakan.
6. Pilih Edit.
7. Tambahkan blok `Sid` berikut untuk kebijakan yang ada. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri.

```
{
      "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"
    }
```

Pada contoh berikut, blok baru dimasukkan pada baris 14.

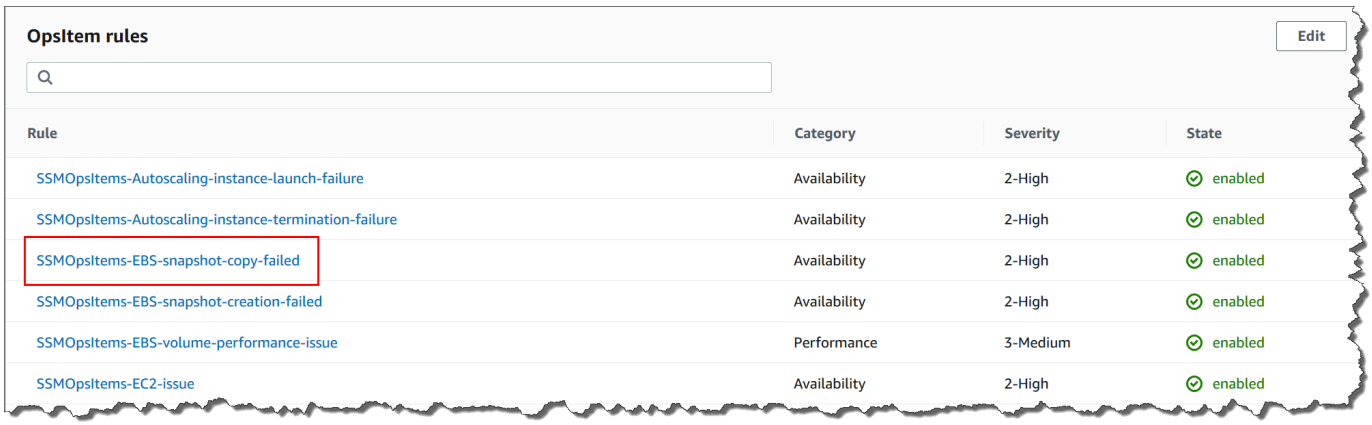
8. Pilih Simpan perubahan.

Langkah 4: Mengaktifkan OpsItems aturan default untuk mengirim pemberitahuan untuk yang baru OpsItems

OpsItemsAturan default di Amazon EventBridge tidak dikonfigurasi dengan Nama Sumber Daya Amazon (ARN) untuk notifikasi Amazon SNS. Gunakan prosedur berikut untuk mengedit aturan EventBridge dan memasukkan `notifications` blok.

Untuk menambahkan blok notifikasi ke OpsItem aturan default

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter.
3. Pilih OpsItemstab, lalu pilih Konfigurasikan sumber.
4. Pilih nama aturan sumber yang ingin Anda konfigurasikan dengan `notifications` blok, seperti yang ditunjukkan pada contoh berikut.



Aturan terbuka di Amazon EventBridge.

5. Pada halaman detail aturan, pada tab Target, pilih Edit.
6. Di bagian Pengaturan tambahan, pilih Konfigurasi transformator input.
7. Dalam kotak Template, tambahkan `notifications` blok dalam format berikut.

```
"notifications":[{"arn":"arn:aws:sns:region:account ID:topic name"}],
```

Inilah contohnya.

```
"notifications":[{"arn":"arn:aws:sns:us-west-2:1234567890:MySNSTopic"}],
```

Masukkan blok notifikasi sebelum `resources` pemblokiran, seperti yang ditunjukkan pada contoh berikut untuk Wilayah AS Barat (Oregon) (us-barat-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. Pilih Konfirmasi.
9. Pilih Selanjutnya.
10. Pilih Selanjutnya.
11. Pilih Perbarui aturan.

Lain kali sistem membuat OpsItem aturan default, ia menerbitkan pemberitahuan ke topik Amazon SNS.

Integrasikan OpsCenter dengan yang lain Layanan AWS

OpsCenter, kemampuanAWS Systems Manager, terintegrasi dengan beberapa Layanan AWS untuk mendiagnosis dan memulihkan masalah dengan AWS sumber daya. Anda harus mengatur Layanan AWS sebelum Anda mengintegrasikannya denganOpsCenter.

Secara default, berikut ini Layanan AWS terintegrasi dengan OpsCenter dan dapat membuat OpsItems secara otomatis:

- Amazon CloudWatch
- Wawasan CloudWatch Aplikasi Amazon

- Amazon EventBridge
- AWS Config
- AWS Systems Manager Incident Manager

Anda harus mengintegrasikan layanan berikut OpsCenter untuk membuat OpsItems secara otomatis:

- DevOpsGuru Amazon
- AWS Security Hub

Ketika salah satu layanan ini membuatOpsItem, Anda dapat mengelola dan memulihkan OpsItem dariOpsCenter. Lihat informasi yang lebih lengkap di MengelolaOpsItems dan MemperbaikiOpsItem masalah.

Untuk informasi lebih lanjut tentang masing-masing Layanan AWS dan bagaimana hal itu terintegrasi denganOpsCenter, lihat topik berikut.

Topik

- Amazon CloudWatch
- Wawasan CloudWatch Aplikasi Amazon
- DevOpsGuru Amazon
- Amazon EventBridge
- AWS Config
- AWS Security Hub
- Incident Manager

Amazon CloudWatch

Amazon CloudWatch memantau AWS sumber daya dan layanan Anda, dan menampilkan metrik pada setiap Layanan AWS yang Anda gunakan. CloudWatch menciptakan OpsItem ketika alarm memasuki status alarm. Misalnya, Anda dapat mengonfigurasi alarm untuk secara otomatis membuat OpsItem jika ada lonjakan kesalahan HTTP yang dihasilkan oleh Application Load Balancer Anda.

Beberapa alarm yang dapat Anda konfigurasi CloudWatch untuk membuat OpsItems ditampilkan dalam daftar berikut:

- Amazon DynamoDB: tindakan baca dan tulis database mencapai ambang batas
- Amazon EC2: pemanfaatan CPU mencapai ambang batas
- AWSpenagihan: perkiraan biaya mencapai ambang batas
- Amazon EC2: sebuah instans gagal pemeriksaan status
- Amazon Elastic Block Store (EBS): pemanfaatan ruang disk mencapai ambang batas

Anda dapat membuat alarm atau mengedit alarm yang ada untuk membuat alarmOpsItem. Untuk informasi selengkapnya, lihat Konfigurasikan CloudWatch alarm untuk dibuatOpsItems.

Ketika Anda mengaktifkan OpsCenter menggunakan Pengaturan Terpadu, itu terintegrasi CloudWatch denganOpsCenter.

Wawasan CloudWatch Aplikasi Amazon

Menggunakan Amazon CloudWatch Application Insights, Anda dapat mengatur monitor yang paling tepat untuk sumber daya aplikasi Anda untuk terus menganalisis data untuk tanda-tanda masalah dengan aplikasi Anda. Saat Anda mengonfigurasi sumber daya CloudWatch aplikasi di Application Insights, Anda dapat memilih untuk membuat OpsItems sistem. OpsCenter An OpsItem dibuat di OpsCenter konsol untuk setiap masalah yang terdeteksi dengan aplikasi. Untuk selengkapnya, lihat Mengatur, mengonfigurasi, dan mengelola aplikasi untuk pemantauan di Panduan CloudWatch Pengguna Amazon.

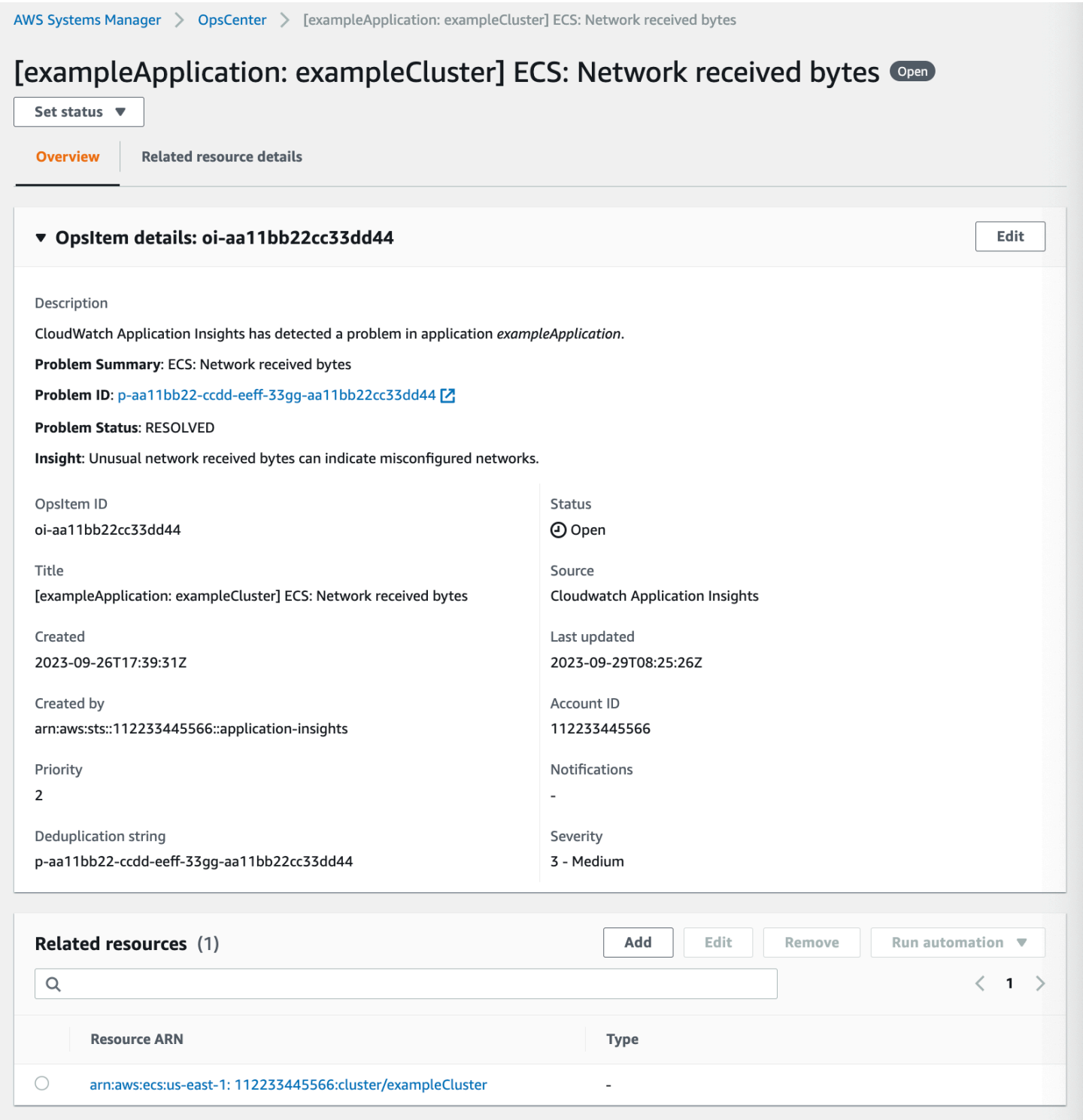
> **Note**
>
> Mulai 16 Oktober 2023, judul dan deskripsi untuk OpsItems dibuat oleh CloudWatch Application Insights sekarang menggunakan format yang ditingkatkan berikut:

```
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>
```

Inilah contohnya:



DevOpsGuru Amazon

Amazon DevOps Guru menerapkan pembelajaran mesin untuk menganalisis data operasional, metrik aplikasi, dan peristiwa aplikasi Anda untuk mengidentifikasi perilaku yang menyimpang dari pola

operasi normal. Jika Anda mengaktifkan DevOps Guru untuk menghasilkan OpsItem inOpsCenter, setiap wawasan menghasilkan yang baruOpsItem. Anda dapat menggunakan OpsCenter untuk mengelola AndaOpsItems.

DevOpsGuru secara otomatis menciptakanOpsItems. Anda dapat mengaktifkan Amazon DevOps Guru untuk membuat OpsItems dengan menggunakanQuick Setup, yang merupakan kemampuan Systems Manager. Sistem membuat OpsItems dengan menggunakan peran terkait layanan AWSServiceRoleForDevOpsGuruAWS Identity and Access Management(IAM).

Berintegrasi OpsCenter dengan DevOps Guru

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Quick Setup.
3. Pada halaman opsi konfigurasi Customize DevOps Guru, pilih tab Library.
4. Di panel DevOpsGuru, pilih Buat.
5. Untuk opsi Konfigurasi, pilih Aktifkan AWS Systems ManagerOpsItems.
6. Pilih Buat setelah Anda menyelesaikan pengaturan.

Amazon EventBridge

Amazon EventBridge memberikan aliran peristiwa yang menggambarkan perubahan AWS sumber daya. Bila Anda mengaktifkan OpsCenter menggunakan Integrated Setup, itu terintegrasi EventBridge denganOpsCenter, dan mengaktifkan EventBridge aturan default. Berdasarkan aturan-aturan ini, EventBridge buatOpsItems. Dengan menggunakan aturan, Anda dapat memfilter dan merutekan acara OpsCenter untuk penyelidikan dan remediasi.

> **Note**
>
> Amazon EventBridge (sebelumnya Amazon CloudWatch Events) menyediakan semua fungsionalitas CloudWatch Acara dan beberapa fitur baru, seperti bus acara khusus, sumber acara pihak ketiga, dan registri skema.

Berikut ini adalah beberapa aturan yang dapat Anda konfigurasikan EventBridge untuk membuatOpsItem:

- Security Hub: peringatan keamanan dikeluarkan

- Amazon DynamoDB adalah peristiwa pelambatan
- Amazon Elastic Compute Cloud Auto Scaling: kegagalan meluncurkan instance
- Systems Manager: gagal untuk menjalankan otomatisasi
- AWS Health: pemberitahuan untuk pemeliharaan terjadwal
- Amazon EC2: status instans diubah dari berjalan menjadi berhenti

Berdasarkan kebutuhan Anda, Anda dapat membuat aturan atau mengedit aturan yang ada untuk membuat aturanOpsItems. Untuk petunjuk tentang cara mengedit aturan untuk membuatOpsItem, lihatKonfigurasikan EventBridge aturan untuk dibuat OpsItems.

AWS Config

AWS Config menyediakan tampilan detail dari konfigurasi sumber daya AWS dalam Akun AWS. Anda.

AWS Configtidak terintegrasi langsung denganOpsCenter. Sebagai gantinya, Anda membuat AWS Config aturan yang mengirimkan peristiwa ke Amazon EventBridge, seperti saat AWS Config mendeteksi instance yang tidak sesuai. Kemudian EventBridge evaluasi peristiwa itu terhadap EventBridge aturan yang telah Anda buat. Jika aturan cocok, EventBridge ubah acara menjadi OpsItem dan mengirimkannya OpsCenter sebagai target tujuan.

Dengan menggunakan iniOpsItem, Anda dapat melacak detail sumber daya yang tidak sesuai, merekam tindakan investigasi, dan menyediakan akses ke tindakan remediasi yang konsisten.

Info terkait

Konfigurasikan EventBridge aturan untuk dibuat OpsItems

Menggunakan AWS Systems ManagerOpsCenter dan AWS Config untuk pemantauan kepatuhan

AWS Security Hub

AWS Security Hubmengumpulkan data keamanan, yang disebut temuan, dari seluruh Akun AWS dan layanan. Menggunakan seperangkat aturan untuk mendeteksi dan menghasilkan temuan, Security Hub membantu Anda mengidentifikasi, memprioritaskan, dan memulihkan masalah keamanan untuk sumber daya yang Anda kelola. Setelah Anda mengonfigurasi integrasi, seperti yang dijelaskan dalam topik ini, Systems Manager membuat OpsItems temuan Security Hub diOpsCenter.

> **Note**
>
> OpsCentermemiliki integrasi dua arah dengan Security Hub. Ini berarti bahwa jika Anda memperbarui bidang Status atau Tingkat Keparahan untuk yang OpsItem terkait dengan temuan keamanan, sistem akan menyinkronkan perubahan dengan Security Hub. Demikian juga, setiap perubahan pada temuan diperbarui secara otomatis di bagian yang sesuai OpsItemsOpsCenter.

Secara default, Systems Manager membuat OpsItems temuan kritis dan tingkat keparahan tinggi. Anda dapat mengonfigurasi secara manual OpsCenter OpsItems untuk membuat temuan tingkat keparahan sedang dan rendah. OpsCentertidak menciptakan temuan informasi OpsItems karena tidak memerlukan remediasi. Untuk informasi selengkapnya tentang tingkat keparahan Security Hub, lihat Keparahan di Referensi AWS Security Hub API.

Sebelum Anda memulai

Sebelum mengonfigurasi OpsCenter untuk membuat OpsItems berdasarkan temuan Security Hub, verifikasi bahwa Anda telah menyelesaikan tugas penyiapan Security Hub. Untuk informasi selengkapnya, lihat Menyiapkan Security Hub di Panduan Pengguna AWS Security Hub.

Saat Anda mengintegrasikan Security Hub denganOpsCenter, sistem akan membuat OpsItems dengan menggunakan peran `AWSServiceRoleForSystemsManagerOpsDataSync` terkait layanan IAM. Untuk informasi selengkapnya tentang peran ini, silakan lihat Menggunakan peran untuk membuat OpsData danOpsItemsuntukExplorer.

> **Warning**
>
> Perhatikan informasi penting berikut tentang harga untuk OpsCenter integrasi dengan Security Hub:
>
> - Jika Anda masuk ke akun administrator Security Hub saat mengonfigurasi OpsCenter dan integrasi Security Hub, sistem akan membuat OpsItems temuan di administrator dan semua akun anggota. Semuanya OpsItems dibuat di akun administrator. Bergantung pada berbagai faktor, ini dapat menyebabkan tagihan besar yang tak terduga dari. AWS
>
>   Jika Anda masuk ke akun anggota saat mengonfigurasi integrasi, sistem hanya membuat OpsItems temuan di akun individu tersebut. Untuk informasi selengkapnya tentang akun administrator Security Hub, akun anggota, dan hubungannya dengan feed EventBridge

peristiwa untuk temuan, lihat Jenis integrasi Security Hub dengan EventBridge dalam Panduan AWS Security Hub Pengguna.

- Untuk setiap temuan yang menghasilkanOpsItem, Anda dikenakan biaya reguler untuk membuatOpsItem. Anda juga dikenakan biaya jika Anda mengedit OpsItem atau jika temuan terkait diperbarui di Security Hub (yang memicu OpsItem pembaruan).

Mengkonfigurasi OpsCenterOpsItems untuk membuat temuan Security Hub

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter.
3. Pilih Pengaturan.
4. Di bagian temuan Security Hub, pilih Edit.
5. Pilih slider untuk mengubah Dinonaktifkan ke Diaktifkan.
6. Jika Anda ingin sistem membuat temuan tingkat keparahan OpsItems sedang atau rendah, alihkan opsi ini.
7. Pilih Simpan untuk menyimpan konfigurasi Anda.

Gunakan prosedur berikut jika Anda tidak lagi ingin sistem membuat OpsItems temuan Security Hub.

Untuk berhenti menerima OpsItems temuan Security Hub

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter.
3. Pilih Pengaturan.
4. Di bagian temuan Security Hub, pilih Edit.
5. Pilih slider untuk mengubah Diaktifkan ke Dinonaktifkan. Jika Anda tidak dapat mengaktifkan slider, Security Hub belum diaktifkan untuk Anda. Akun AWS
6. Pilih Simpan untuk menyimpan konfigurasi Anda. OpsCentertidak lagi dibuat OpsItems berdasarkan temuan Security Hub.

Important

Administrator yang didelegasikan Systems Manager atau akun AWS Organizations manajemen dapat mengaktifkan temuan Security Hub OpsCenter untuk beberapa akun

dan Wilayah AWS dengan membuat sinkronisasi data sumber daya. Explorer Jika sumber Security Hub diaktifkan Explorer dan ada sinkronisasi data sumber daya yang menargetkan akun anggota tempat Anda menonaktifkan integrasi Security Hub, maka pengaturan yang dipilih oleh administrator akan diutamakan. OpsCenterterus membuat OpsItems untuk temuan Security Hub. Untuk berhenti membuat OpsItems temuan Security Hub di akun anggota yang ditargetkan oleh sinkronisasi data sumber daya, hubungi administrator Anda dan minta mereka untuk menghapus akun Anda dari sinkronisasi data sumber daya atau menonaktifkan sumber Security Hub diExplorer. Untuk informasi tentang mengubah setelanExplorer, lihatMengedit sumber data Systems Manager Explorer.

Incident Manager

Incident Manager, kemampuanAWS Systems Manager, menyediakan konsol manajemen insiden yang membantu Anda mengurangi dan memulihkan dari insiden yang memengaruhi aplikasi yang Anda AWS host. insiden adalah gangguan yang tidak direncanakan atau penurunan kualitas layanan. Setelah Anda mengatur dan mengkonfigurasi Manajer Insiden, sistem secara otomatis membuat OpsItemsOpsCenter.

Ketika sistem membuat insiden di Manajer Insiden, itu juga membuat OpsItem inOpsCenter, dan menampilkan insiden sebagai item terkait. Jika OpsItem sudah ada, Manajer Insiden tidak membuat fileOpsItem. Yang pertama OpsItem dikenal sebagai orangtuaOpsItem. Jika insiden tumbuh dalam skala dan ruang lingkup, Anda dapat menambahkan insiden ke yang sudah adaOpsItem. Jika diperlukan, Anda dapat secara manual membuat insiden untuk sebuahOpsItem. Setelah insiden ditutup, Anda dapat membuat analisis di Manajer Insiden untuk meninjau dan meningkatkan proses remediasi untuk masalah serupa.

Secara default, OpsCenter terintegrasi dengan Manajer Insiden. Jika Manajer Insiden tidak disiapkan, OpsCenter halaman akan menampilkan pesan untuk mengatur Manajer Insiden. Ketika Manajer Insiden membuatOpsItem, Anda dapat mengelola dan memulihkan OpsItem dariOpsCenter. Untuk petunjuk tentang membuat insiden untuk sebuahOpsItem, lihatMembuat insiden untukOpsItem.

Buat OpsItems

Setelah Anda mengaturOpsCenter, kemampuanAWS Systems Manager, dan mengintegrasikannya dengan AndaLayanan AWS, Anda Layanan AWS secara otomatis membuat OpsItems berdasarkan aturan default, peristiwa, atau alarm.

Anda dapat melihat status dan tingkat keparahan EventBridge aturan Amazon default. Jika diperlukan, Anda dapat membuat atau mengedit aturan ini dari AmazonEventBridge. Anda juga dapat melihat alarm dari AmazonCloudWatch, dan membuat atau mengedit alarm. Dengan menggunakan aturan dan alarm, Anda dapat mengonfigurasi peristiwa yang ingin Anda hasilkan OpsItems secara otomatis.

Ketika sistem membuatOpsItem, itu dalam status Open. Anda dapat mengubah status ke Dalam proses ketika Anda memulai penyelidikan OpsItem dan Terselesaikan setelah Anda memulihkan. OpsItem Untuk informasi lebih lanjut tentang cara mengonfigurasi alarm dan aturan Layanan AWS untuk membuat OpsItems dan cara membuat OpsItems secara manual, lihat hal berikut.

Topik

Konfigurasikan EventBridge aturan untuk dibuat OpsItems

Ketika Amazon EventBridge menerima sebuah kejadian, ia menciptakan sebuah kejadian baru OpsItem berdasarkan aturan default. Anda dapat membuat aturan atau mengedit aturan yang ada untuk ditetapkan OpsCenter sebagai target suatu EventBridge peristiwa. Untuk informasi tentang cara membuat aturan kejadian, lihat Membuat aturan untuk Layanan AWS di Panduan EventBridge Pengguna Amazon.

Untuk mengkonfigurasi EventBridge aturan yang akan dibuat OpsItemsOpsCenter

1. Buka konsol Amazon EventBridge di https://console.aws.amazon.com/events/.
2. Di panel navigasi, pilih Aturan.
3. Pada halaman Aturan, untuk Event bus, pilih default.
4. Untuk Aturan, pilih aturan dengan memilih kotak centang di samping namanya.
5. Pilih sebuah nama untuk membuka halaman detailnya untuk membuka halaman detailnya. Dalam rincian Aturan, verifikasi bahwa Status diatur ke Diaktifkan.

> **Note**
>
> Jika diperlukan, Anda dapat memperbarui status menggunakan Edit di sudut kanan atas halaman.

6. Pilih tabTarget.
7. Di tabTargets, pilih Edit.
8. Untuk jenis Target, pilih Layanan AWS.
9. Untuk Pilih target, pilih Systems Manager OpsItem.
10. Untuk banyak jenis target, EventBridge membutuhkan izin untuk mengirim kejadian ke target. Dalam kasus ini, EventBridge dapat membuat AWS Identity and Access Management (IAM) role yang diperlukan bagi aturan Anda untuk menjalankan:

    - Untuk membuat IAM role secara otomatis, pilih Buat peran baru untuk sumber daya khusus ini.
    - Untuk menggunakan peran IAM yang Anda buat untuk memberikan EventBridge izin untuk membuat OpsItemsOpsCenter, pilih Gunakan peran yang ada.
11. Di Pengaturan tambahan, untuk Konfigurasikan input target, pilih Input Transformer.

    Anda dapat menggunakan opsi transformator input untuk menentukan string deduplikasi dan informasi penting lainnya untukOpsItems, seperti judul dan tingkat keparahan.
12. Memilih Konfigurasikan transformator input.
13. Di Trafo input target, untuk Jalur input, tentukan nilai yang akan diuraikan dari peristiwa pemicu. Misalnya, untuk menguraikan waktu mulai, waktu akhir, dan detail lainnya dari kejadian yang memicu aturan, gunakan JSON berikut.

    ```
    {
        "end-time": "$.detail.EndTime",
        "failure-cause": "$.detail.cause",
        "resources": "$.resources",
        "source": "$.detail.source",
        "start-time": "$.detail.StartTime"
    }
    ```
14. Untuk Template, tentukan informasi yang akan dikirim ke target. Misalnya, gunakan JSON berikut untuk meneruskan informasi. OpsCenter Informasi ini digunakan untuk membuatOpsItem.

> **Note**
>
> Jika template input dalam format JSON, maka nilai objek dalam template tidak dapat menyertakan tanda kutip. Misalnya, nilai untuk sumber daya, penyebab kegagalan, sumber, waktu mulai, dan waktu akhir tidak dapat dalam tanda kutip.

```
{
    "title": "EBS snapshot copy failed",
    "description": "CloudWatch Event Rule SSMOpsItems-EBS-snapshot-copy-failed was
 triggered. Your EBS snapshot copy has failed. See below for more details.",
    "category": "Availability",
    "severity": "2",
    "source": "EC2",
    "resources": <resources>,
    "operationalData": {
        "/aws/dedup": {
            "type": "SearchableString",
            "value": "{\"dedupString\":\"SSMOpsItems-EBS-snapshot-copy-failed\"}"
        },
        "/aws/automations": {
            "value": "[ { \"automationType\": \"AWS:SSM:Automation\",
 \"automationId\": \"AWS-CopySnapshot\" } ]"
        },
        "failure-cause": {
            "value": <failure-cause>
        },
        "source": {
            "value": <source>
        },
        "start-time": {
            "value": <start-time>
        },
        "end-time": {
            "value": <end-time>
        }
    }
}
```

Untuk informasi selengkapnya tentang bidang ini, lihat Mengubah input target di Panduan EventBridge Pengguna Amazon.

15. Pilih Konfirmasi.
16. Pilih Selanjutnya.
17. Pilih Selanjutnya.
18. Pilih Perbarui aturan.

Setelah OpsItem dibuat dari sebuah kejadian, Anda dapat melihat detail acara dengan membuka OpsItem dan gulir ke bagian data operasional privat. Untuk informasi tentang cara mengkonfigurasi opsi dalam sebuahOpsItem, lihatMengelolaOpsItems.

Konfigurasikan CloudWatch alarm untuk dibuatOpsItems

Selama pengaturan terintegrasiOpsCenter, kemampuanAWS Systems Manager, Anda memungkinkan Amazon CloudWatch untuk secara otomatis membuatOpsItems berdasarkan alarm umum. Anda dapat membuat alarm atau mengedit alarm yang ada untuk dibuatOpsItemsOpsCenter.

CloudWatch membuat peran tertaut layanan baru diAWS Identity and Access Management (IAM) saat Anda mengkonfigurasi alarm untuk membuatOpsItems. Peran baru ini bernama `AWSServiceRoleForCloudWatchAlarms_ActionSSM`. Untuk informasi selengkapnya tentang peran CloudWatch terkait layanan, lihat Menggunakan peran terkait layanan untuk CloudWatch dalam Panduan CloudWatch Pengguna Amazon.

Ketika CloudWatch alarm menghasilkanOpsItem,OpsItem menampilkan CloudWatch alarm - '***alarm_name***' dalam keadaan ALARM.

Untuk melihat detail tentang spesifikOpsItem, pilihOpsItem dan kemudian pilih Detail sumber daya terkait. Anda dapat mengedit secara manualOpsItems untuk mengubah detail, seperti kepelikan atau kategori. Namun, saat Anda mengedit tingkat keparahan atau kategori alarm, Systems Manager tidak dapat memperbarui tingkat keparahan atau kategoriOpsItems yang sudah dibuat dari alarm. Jika alarm membuatOpsItem dan jika Anda menentukan string deduplikasi, alarm tidak akan membuat tambahanOpsItems meskipun Anda mengedit alarm CloudWatch. JikaOpsItem diselesaikan diOpsCenter, CloudWatch akan membuat yang baruOpsItem.

Untuk informasi lebih lanjut tentang konfigurasi CloudWatch alarm, lihat topik berikut.

Topik

- Mengkonfigurasi CloudWatch alarm untuk membuatOpsItems (konsol)
- Mengkonfigurasi CloudWatch alarm yang ada untuk membuatOpsItems (pemrograman)

Mengkonfigurasi CloudWatch alarm untuk membuatOpsItems (konsol)

Anda dapat membuat alarm secara manual atau memperbarui alarm yang ada untuk dibuatOpsItems dari Amazon CloudWatch.

Untuk membuat CloudWatch alarm dan mengkonfigurasi Systems Manager sebagai target alarm tersebut

1. Selesaikan langkah 1—9 seperti yang ditentukan dalam Buat CloudWatch alarm berdasarkan ambang batas statis di Panduan CloudWatch Pengguna Amazon.
2. Di bagian tindakan Systems Manager, pilih OpsCentertindakan Tambahkan Systems Manager.
3. Pilih OpsItems.
4. Untuk Kepelikan, pilih dari 1 sampai 4.
5. (Opsional) Untuk Kategori, pilih kategori untukOpsItem.
6. Selesaikan langkah 11-13 seperti yang ditentukan dalam Buat CloudWatch alarm berdasarkan ambang batas statis di Panduan CloudWatch Pengguna Amazon.
7. Pilih Selanjutnya dan selesaikan wizard.

Untuk mengedit alarm yang ada dan mengkonfigurasi Systems Manager sebagai target alarm tersebut

1. Buka CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/.
2. Di panel navigasi, pilih Alarm.
3. Pilih alarm, dan kemudian pilih Tindakan, Edit.
4. (Opsional) Ubah pengaturan di bagian Metrik dan Kondisi, dan kemudian pilih Selanjutnya.
5. Di bagian Systems Manager, pilih OpsCentertindakan Add Systems Manager.
6. Untuk Kepelikan, pilih sebuah angka.

> **Note**
>
> Kepelikan adalah nilai yang ditentukan pengguna. Anda atau organisasi Anda menentukan arti setiap nilai kepelikan dan perjanjian tingkat layanan yang terkait dengan setiap kepelikan.

7. (Opsional) Untuk Kategori, pilih sebuah opsi.
8. Pilih Selanjutnya dan selesaikan wizard.

Mengkonfigurasi CloudWatch alarm yang ada untuk membuatOpsItems (pemrograman)

Anda dapat mengonfigurasi CloudWatch alarm Amazon untuk membuatOpsItems pemrograman dengan menggunakanAWS Command Line Interface (AWS CLI),AWS CloudFormation templat, atau cuplikanJava kode.

Topik

- Sebelum Anda memulai
- Mengkonfigurasi CloudWatch alarm untuk membuatOpsItems (AWS CLI)
- Mengkonfigurasi CloudWatch alarm untuk membuat atau memperbaruiOpsItems (CloudFormation)
- Mengkonfigurasi CloudWatch alarm untuk membuat atau memperbaruiOpsItems (Java)

Sebelum Anda memulai

Jika Anda secara terprogram mengedit alarm yang ada atau membuat alarm yang dibuatOpsItems, Anda harus menentukan Amazon Resource Name (ARN). ARN ini mengidentifikasi Systems ManagerOpsCenter sebagai target untukOpsItems dibuat dari alarm. Anda dapat menyesuaikan ARN sehingga yangOpsItems dibuat dari alarm menyertakan informasi spesifik seperti kepelikan atau kategori. Setiap ARN menyertakan informasi yang dijelaskan dalam tabel berikut.

Parameter	Detail
`Region`(diperlukan)	Wilayah AWS tempat alarm berada. Misalnya: `us-west-2` . Untuk informasi tentangWilayah AWS tempat yang dapat Anda gunakanOp

Parameter	Detail
	sCenter, lihat AWS Systems Managertitik akhir dan kuota.
`account_ID` (diperlukan)	ID Akun AWS yang sama yang digunakan untuk membuat alarm. Misalnya: `123456789012` . ID akun harus diikuti dengan titik dua (`:`) dan parameter`opsitem` seperti yang ditunjukkan dalam contoh berikut.
`severity`(diperlukan)	Tingkat kepelikan yang ditetapkan pengguna untukOpsItems dibuat dari alarm. Nilai valid: `1`, `2`, `3`, `4`
`Category`(opsional)	Kategori untukOpsItems dibuat dari alarm. Nilai yang valid:`Availability` ,`Cost`,`Performance` ,`Recovery`, dan`Security`.

Buat ARN dengan menggunakan sintaks berikut. ARN ini tidak termasuk parameter `Category` opsional.

```
arn:aws:ssm:Region:account_ID:opsitem:severity
```

Berikut adalah contohnya.

```
arn:aws:ssm:us-west-2:123456789012:opsitem:3
```

Untuk membuat ARN yang menggunakan parameter `Category` opsional, gunakan sintaks berikut.

```
arn:aws:ssm:Region:account_ID:opsitem:severity#CATEGORY=category_name
```

Berikut adalah contohnya.

```
arn:aws:ssm:us-west-2:123456789012:opsitem:3#CATEGORY=Security
```

Mengkonfigurasi CloudWatch alarm untuk membuatOpsItems (AWS CLI)

Perintah ini mengharuskan Anda menentukan ARN untuk`alarm-actions` parameter. Untuk informasi tentang cara membuat ARN, lihat Sebelum Anda memulai.

Untuk mengkonfigurasi CloudWatch alarm untuk membuatOpsItems (AWS CLI)

1. Instal dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaruAWS CLI.

2. Jalankan perintah berikut untuk mengumpulkan informasi tentang alarm yang ingin Anda konfigurasi.

```
aws cloudwatch describe-alarms --alarm-names "alarm name"
```

3. Jalankan perintah berikut untuk memperbarui alarm. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

```
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
```

   Inilah contohnya.

   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
```

Mengkonfigurasi CloudWatch alarm untuk membuat atau memperbaruiOpsItems (CloudFormation)

Bagian ini mencakupAWS CloudFormation templat yang dapat Anda gunakan untuk mengkonfigurasi CloudWatch alarm untuk membuat atau memperbarui secara otomatisOpsItems. Setiap templat mengharuskan Anda menentukan ARN untuk`AlarmActions` parameter. Untuk informasi tentang cara membuat ARN, lihat Sebelum Anda memulai.

alarm metrik - Gunakan CloudFormation templat berikut untuk membuat atau memperbarui alarm CloudWatch metrik. alarm yang ditentukan dalam templat ini memantau pemeriksaan status instans Amazon Elastic Compute Cloud (Amazon EC2). Jika alarm memasuki`ALARM` negara, itu menciptakanOpsItem inOpsCenter.

```
   {
     "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"}}]
          }
        }
      }
    }
```

alarm komposit - Gunakan CloudFormation templat berikut untuk membuat atau memperbarui alarm komposit. Alarm komposit terdiri dari beberapa alarm metrik. Jika alarm memasukiALARM negara, itu menciptakanOpsItem inOpsCenter.

```
"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"
         }
      }
   }
```

Mengkonfigurasi CloudWatch alarm untuk membuat atau memperbaruiOpsItems (Java)

Bagian ini mencakup cuplikanJava kode yang dapat Anda gunakan untuk mengkonfigurasi CloudWatch alarm untuk membuat atau memperbarui secara otomatisOpsItems. Setiap cuplikan mengharuskan Anda menentukan ARN untuk`validSsmActionStr` parameter. Untuk informasi tentang cara membuat ARN, lihat Sebelum Anda memulai.

alarm khusus - Gunakan cuplikanJava kode berikut untuk membuat atau memperbarui CloudWatch alarm. alarm yang ditentukan dalam templat ini memantau pemeriksaan status instans Amazon EC2. Jika alarm memasukiALARM negara, itu menciptakanOpsItem inOpsCenter.

```
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);
    }
```

Perbarui semua alarm - Gunakan cuplikanJava kode berikut untuk memperbarui semua CloudWatch alarm yang ada di AndaAkun AWS yang dibuatOpsItems saat alarm memasukiALARM status.

```
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;
            }
        }
    }
```

Buat OpsItems secara manual

Ketika Anda menemukan masalah operasional, Anda dapat secara manual membuatOpsItem dariOpsCenter, kemampuanAWS Systems Manager, untuk mengelola dan menyelesaikan masalah.

Jika Anda menyiapkanOpsCenter administrasi lintas akun, administrator atau akunAWS Organizations manajemen yang didelegasikan oleh Systems Manager dapat membuatOpsItems akun

anggota. Untuk informasi selengkapnya, lihat (Opsional) Menyiapkan OpsCenter untuk mengelola secara terpusat OpsItems di seluruh akun.

Anda dapat membuatOpsItems dengan menggunakanAWS Systems Manager konsol,AWS Command Line Interface (AWS CLI), atauAWS Tools for Windows PowerShell.

Topik

- MenciptakanOpsItemssecara manual (konsol)
- MembuatOpsItems secara manual (AWS CLI)
- MembuatOpsItems secara manual (PowerShell)

MenciptakanOpsItemssecara manual (konsol)

Anda dapat secara manual membuatOpsItemsmenggunakanAWS Systems Managerkonsol. Saat Anda membuatOpsItem, itu ditampilkan diOpsCenterakun. Jika Anda mengaturOpsCenteruntuk administrasi lintas akun,OpsCentermenyediakan administrator atau akun manajemen yang didelegasikan dengan opsi untuk membuatOpsItemsuntuk akun anggota yang dipilih. Untuk informasi selengkapnya, lihat (Opsional) Menyiapkan OpsCenter untuk mengelola secara terpusat OpsItems di seluruh akun.

Untuk membuatOpsItemmenggunakanAWS Systems Managerkonsol

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter.
3. Pilih CreateOpsItem (Buat). Jika Anda tidak melihat tombol ini, pilihOpsItemstab, lalu pilihBuatOpsItem.
4. (Opsional) PilihAkun lainnya, lalu pilih akun tempat Anda ingin membuatOpsItem.

   > **Note**
   >
   > Langkah ini diperlukan jika Anda membuatOpsItemsuntuk akun anggota.

5. UntukJudul, masukkan nama deskriptif untuk membantu Anda memahami tujuanOpsItem.
6. UntukSumber, masukkan jenis yang terkena dampakAWSsumber daya atau informasi sumber lainnya untuk membantu pengguna memahami asalOpsItem.

> **Note**
>
> Anda tidak dapat mengeditSumberbidang setelah Anda membuatOpsItem.

7. (Opsional) Untuk Prioritas, pilih tingkat prioritas.
8. (Opsional) Untuk Kepelikan, pilih tingkat kepelikan.
9. (Opsional) Untuk Kategori, pilih kategori.
10. UntukDeskripsi, masukkan informasi tentang iniOpsItemtermasuk (jika ada) langkah-langkah untuk mereproduksi masalah.

> **Note**
>
> Konsol mendukung sebagian besar pemformatan penurunan harga di OpsItembidang deskripsi. Untuk informasi lebih lanjut, lihatMenggunakan Markdown di KonsoldiMemulai denganAWS Management ConsolePanduan Memulai.

11. UntukString deduplikasi, masukkan kata-kata yang dapat digunakan sistem untuk memeriksa duplikatOpsItems. Untuk informasi lebih lanjut tentang string deduplikasi, lihat Mengelola duplikatOpsItems.
12. (Opsional) UntukPemberitahuan, tentukan Nama Sumber Daya Amazon (ARN) dari topik Amazon SNS tempat Anda ingin pemberitahuan dikirim saat iniOpsItemdiperbarui. Anda harus menentukan Amazon SNS ARN yang samaWilayah AWSsebagaiOpsItem.
13. (Opsional) UntukSumber daya terkait, pilihMenambahkanuntuk menentukan ID atau ARN dari sumber daya yang terkena dampak dan sumber daya terkait.
14. Pilih CreateOpsItem (Buat).

Jika berhasil, halaman akan menampilkanOpsItem. Ketika administrator atau akun manajemen yang didelegasikan membuatOpsItemuntuk akun anggota yang dipilih, yang baruOpsItemsditampilkan diOpsCenterdari akun administrator dan anggota. Untuk informasi tentang cara mengkonfigurasi opsi dalamOpsItem, lihatMengelolaOpsItems.

MembuatOpsItems secara manual (AWS CLI)

Prosedur berikut menjelaskan cara membuatOpsItem dengan menggunakanAWS Command Line Interface (AWS CLI).

Untuk membuatOpsItem menggunakanAWS CLI

1. Instal dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaru dariAWS CLI.

2. BukaAWS CLI dan jalankan perintah berikut untuk membuatOpsItem. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

```
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"
```

   Tentukan data operasional dari sebuah file

   Ketika Anda membuatOpsItem, Anda dapat menentukan data operasional dari file. File harus berupaJSON file, dan isi file harus menggunakan format berikut.

```
{
  "key_name": {
    "Type": "SearchableString",
    "Value": "Up to 20 KB of data"
  }
}
```

   Inilah contohnya.

```
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**
>
> Untuk informasi tentang cara memasukkan parameter yang berformat JSON pada baris perintah pada sistem operasi lokal yang berbeda, lihat Menggunakan tanda kutip dengan stringAWS CLI di PanduanAWS Command Line Interface Pengguna.

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "OpsItemId": "oi-1a2b3c4d5e6f"
}
```

3. Jalankan perintah berikut untuk menampilkan detail tentangOpsItem yang Anda buat.

```
aws ssm get-ops-item --ops-item-id ID
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "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. Jalankan perintah berikut untuk memperbaruiOpsItem. Perintah ini mengubah status dari `Open` (default) ke `InProgress`.

```
aws ssm update-ops-item --ops-item-id ID --status InProgress
```

   Perintah tidak memiliki output.

5. Jalankan lagi perintah berikut untuk memverifikasi bahwa status diubah ke`InProgress`.

```
aws ssm get-ops-item --ops-item-id ID
```

Contoh pembuatanOpsItem

Contoh kode berikut menunjukkan cara membuatOpsItem dengan menggunakan portalLinux manajemen,macOS, atauWindows.

Linuxportal manajemen ataumacOS

Perintah berikut membuat diskOpsItem instans Amazon Elastic Compute Cloud (Amazon EC2) penuh.

```
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"
```

Perintah berikut menggunakan`/aws/resources` kunci`OperationalData` untuk membuat sumber dayaOpsItem terkait 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"
```

Perintah berikut menggunakan/aws/automations kunci0perationalData untuk membuatOpsItem yang menentukanAWS-ASGEnterStandby dokumen sebagai runbook Otomasi terkait.

```
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

Perintah berikut membuatOpsItem instans Amazon Relational Database Service (Amazon RDS) tidak merespons.

```
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"
```

Perintah berikut menggunakan/aws/resources kunci0perationalData untuk membuatOpsItem sumber daya terkait instans 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\"}}
```

Perintah berikut menggunakan`/aws/automations` kunci`OperationalData` untuk membuat`OpsItem`AWS-RestartEC2Instance runbook sebagai runbook Otomatisasi terkait.

```
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\"}}
```

MembuatOpsItems secara manual (PowerShell)

Prosedur di bawah menjelaskan cara membuatOpsItem dengan menggunakanAWS Tools for Windows PowerShell.

Untuk membuatOpsItem menggunakanAWS Tools for Windows PowerShell

1. Buka AWS Tools for Windows PowerShell dan jalankan perintah berikut untuk menentukan kredensial Anda.

   ```
   Set-AWSCredentials –AccessKey key-name –SecretKey key-name
   ```

2. Jalankan perintah Anda untuk mengatur untuk mengaturWilayah AWS untukPowerShell sesi Anda.

   ```
   Set-DefaultAWSRegion -Region Region
   ```

3. Jalankan perintah`OpsItem`. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri. Perintah ini menentukan runbook Systems Manager Automation untuk remediasi ini`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
```

Jika berhasil, perintah menghasilkan ID dari yang baruOpsItem.

Contoh di bawah menentukan Amazon Resource Name (ARN) dari instans Amazon Elastic Compute Cloud (Amazon EC2).

```
$opsItem = New-Object Amazon.SimpleSystemsManagement.Model.OpsItemDataValue
$opsItem.Type = [Amazon.SimpleSystemsManagement.OpsItemDataType]::SearchableString
$opsItem.Value = '[{\"arn\":\"arn:aws:ec2:us-east-1:123456789012:instance/
i-1234567890abcdef0\"}]'
$newHash = @{" /aws/
resources"=[Amazon.SimpleSystemsManagement.Model.OpsItemDataValue]$opsItem}
New-SSMOpsItem -Title "EC2 instance disk full still" -Description "Log clean up may
 have failed which caused the disk to be full" -Priority 2 -Source ec2 -OperationalData
 $newHash
```

MengelolaOpsItems

OpsCenter, kemampuanAWS Systems Manager, melacakOpsItems dari penciptaan mereka untuk resolusi. Jika Anda menyiapkanOpsCenter administrasi lintas akun, administrator atau akun manajemen yang didelegasikan dapat mengelolaOpsItems dari akun mereka. Untuk informasi selengkapnya, lihat (Opsional) Menyiapkan OpsCenter untuk mengelola secara terpusat OpsItems di seluruh akun.

Anda dapat melihat dan mengelolaOpsItems dengan menggunakan halaman-halaman berikut di konsol Systems Manager:

- Ringkasan - Menampilkan hitungan terbuka dan sedang berlangsungOpsItems, hitunganOpsItems berdasarkan sumber dan usia, dan wawasan operasional. Anda dapat memfilterOpsItems berdasarkan sumber danOpsItems status.

- OpsItems- Menampilkan daftarOpsItems dengan beberapa bidang informasi, seperti judul, ID, prioritas, deskripsi, sumber, sumberOpsItem, dan tanggal dan waktu pembaruan terakhir. Dengan menggunakan halaman ini, Anda dapat secara manual membuatOpsItems, mengonfigurasi sumber, mengubah statusOpsItem, dan memfilterOpsItems berdasarkan insiden baru. Anda dapat memilihOpsItem untuk menampilkan halaman OpsItemsdetailnya.

- OpsItemrincian - Menyediakan wawasan rinci dan alat yang dapat Anda gunakan untuk mengelolaOpsItem. HalamanOpsItems rincian memiliki tab berikut:
  - Ikhtisar - Menampilkan sumber daya terkait, runbook yang berjalan dalam 30 hari terakhir, dan daftar runbook yang tersedia yang dapat Anda jalankan. Anda juga dapat melihat serupaOpsItems, menambahkan data operasional, dan menambahkan terkaitOpsItems.
  - Rincian sumber daya terkait - Menampilkan informasi tentang sumber daya dari beberapaAWS layanan. Perluas bagian Detail sumber daya untuk melihat informasi tentang sumber daya ini sebagaimana disediakan oleh layanan AWS yang menjadi host. Anda juga dapat beralih melalui sumber daya terkait lainnya yang terkait dengan iniOpsItem dengan menggunakan daftar Sumber daya terkait.

Untuk informasi selengkapnya tentang cara mengelolaOpsItems, lihat topik berikut.

Topik

MelihatOpsItem

Untuk mendapatkan tampilan komprehensifOpsItem, gunakan halaman OpsItemdetail diOpsCenter konsol. Halaman Ikhtisar menampilkan informasi berikut:

- OpsItemsrincian - Menampilkan informasi umum untuk yang dipilihOpsItem.
- Sumber daya Terkait - Sumber daya terkait adalah sumber daya yang terkena dampak atau sumber daya yang memulai kejadian yang menciptakanOpsItem.
- Eksekusi otomatisasi dalam 30 hari terakhir - Daftar runbook yang berjalan dalam 30 hari terakhir.
- Runbooks - Anda dapat memilih runbook dari daftar runbook yang tersedia.
- Serupa OpsItems - Ini adalah daftar yang dihasilkan sistemOpsItems yang mungkin terkait atau menarik bagi Anda. Untuk menghasilkan daftar, sistem memindai judul dan deskripsi dari semuaOpsItems dan mengembalikanOpsItems yang menggunakan kata-kata yang serupa.
- Data operasional - Data operasional adalah data kustom yang menyediakan detail referensi berguna tentangOpsItem. Misalnya, Anda dapat menentukan berkas log, string kesalahan, kunci lisensi, kiat pemecahan masalah, atau data lain yang relevan.
- Terkait OpsItems - Anda dapat menentukan IDOpsItems yang memiliki sejumlah keterkaitan dengan arusOpsItem.
- Detail Sumber Daya Terkait - Menampilkan penyedia data, termasuk CloudWatch metrik dan alarm Amazon,AWS CloudTrail log, dan detailnyaAWS Config.

Untuk melihat detail OpsItem

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter.
3. PilihOpsItem untuk melihat

MengeditOpsItem

TheOpsItemdetailbagian mencakup informasi tentangOpsItem, termasuk deskripsi, judul, sumber,OpsItemID, dan statusnya.
Anda dapat mengedit satuOpsItematau Anda dapat memilih beberapaOpsItemsdan edit bidang berikut:Status,Prioritas,Keparahan,Kategori.

Saat Amazon EventBridge menciptakan sebuahOpsItem, itu mengisiJudul,Sumber, danDeskripsiladang. Anda dapat mengeditJuduldanDeskripsibidang, tetapi Anda tidak dapat mengeditSumberlapangan.

> **Note**
>
> Konsol mendukung sebagian besar pemformatan penurunan harga di OpsItem bidang deskripsi. Untuk informasi lebih lanjut, lihatMenggunakan Markdown di KonsoldiMemulai denganAWS Management ConsolePanduan Memulai.

Umumnya, Anda dapat mengedit data yang dapat dikonfigurasi berikut untukOpsItem:

- Judul— NamaOpsItem. Sumber menciptakan judulOpsItem.
- Deskripsi— Informasi tentang iniOpsItemtermasuk (jika ada) langkah-langkah untuk mereproduksi masalah.
- Status— Status dariOpsItembisa Terbuka, Dalam proses, atau Terselesaikan.
- Prioritas— Prioritas dariOpsItembisa antara 1 dan 5. Kami menyarankan agar organisasi Anda menentukan arti setiap tingkat prioritas dan perjanjian tingkat layanan yang sesuai untuk setiap level.
- Keparahan— Keparahan dariOpsItembisa antara 1 hingga 4, di mana 1 kritis, 2 tinggi, 3 sedang, dan 4 rendah.
- Kategori— Kategori dariOpsItemdapat berupa ketersediaan, biaya, kinerja, pemulihan, atau keamanan.
- Pemberitahuan— Saat Anda mengeditOpsItem, Anda dapat menentukan Amazon Resource Name (ARN) dari topik Amazon Simple Notification Service diPemberitahuanlapangan. Dengan menentukan ARN, Anda memastikan bahwa semua pemangku kepentingan menerima pemberitahuan ketikaOpsItemdiedit, termasuk perubahan status. Untuk informasi lebih lanjut, lihat Panduan Developer Layanan Notifikasi Sederhana Amazon.

> **Important**
>
> Topik Amazon SNS harus ada dalam hal yang samaWilayah AWSsebagaiOpsItem. Jika topik danOpsItemberada di Wilayah yang berbeda, sistem mengembalikan kesalahan.

OpsCentermemiliki integrasi dua arah denganAWS Security Hub. Saat Anda memperbaruiOpsItemstatus dan tingkat keparahan yang terkait dengan temuan keamanan, perubahan tersebut secara otomatis dikirim ke Security Hub untuk memastikan Anda selalu melihat informasi terbaru dan benar.

Untuk mengeditOpsItemdetail

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter.
3. PilihOpsItemID untuk membuka halaman detail atau memilih beberapaOpsItems. Jika Anda memilih beberapaOpsItems, Anda hanya dapat mengedit status, prioritas, tingkat keparahan, atau kategori. Jika Anda mengedit beberapaOpsItems,OpsCentermemperbarui dan menyimpan perubahan Anda segera setelah Anda memilih status, prioritas, tingkat keparahan, atau kategori baru.
4. DiOpsItem detailbagian, pilihSunting.
5. Edit detailOpsItemsesuai dengan persyaratan dan pedoman yang ditentukan oleh organisasi Anda.
6. Setelah selesai, pilih Simpan.

Menambahkan sumber daya terkait ke OpsItem

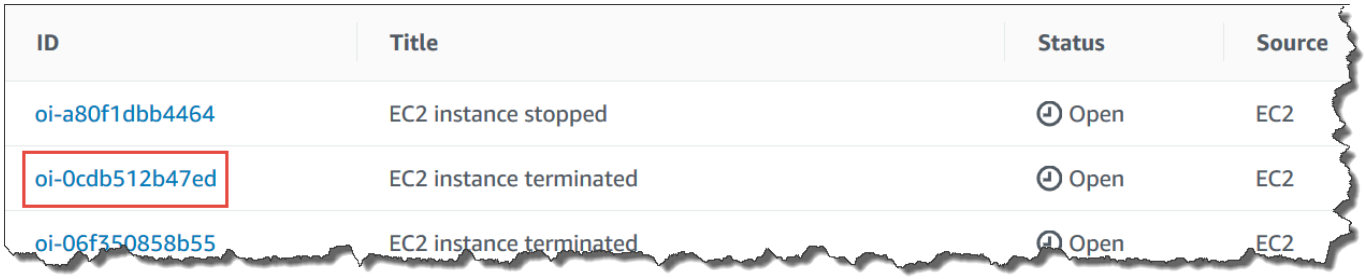
Masing-masing OpsItem mencakup bagian Sumber daya terkait yang mencantumkan Amazon Resource Name (ARN) dari sumber daya terkait. Sumber daya terkait adalah sumber daya yang terkena dampak AWS yang perlu diselidiki.

Jika Amazon EventBridge membuatOpsItem, sistem secara otomatis mengisi OpsItem dengan ARN sumber daya. Anda dapat secara manual menentukan ARN sumber daya terkait. Untuk tipe ARN tertentu, OpsCenter secara otomatis membuat tautan mendalam yang menampilkan detail tentang sumber daya secara langsung di OpsCenter konsol. Misalnya, jika Anda menentukan ARN instans Amazon Amazon EC2 (Amazon EC2) instans Amazon EC2 (Amazon EC2) sebagai sumber daya terkait, maka OpsCenter menarik detail tentang instans EC2 tersebut. Hal ini mengizinkan Anda untuk melihat informasi detail tentang AWS sumber daya Anda yang terdampak tanpa harus meninggalkanOpsCenter.

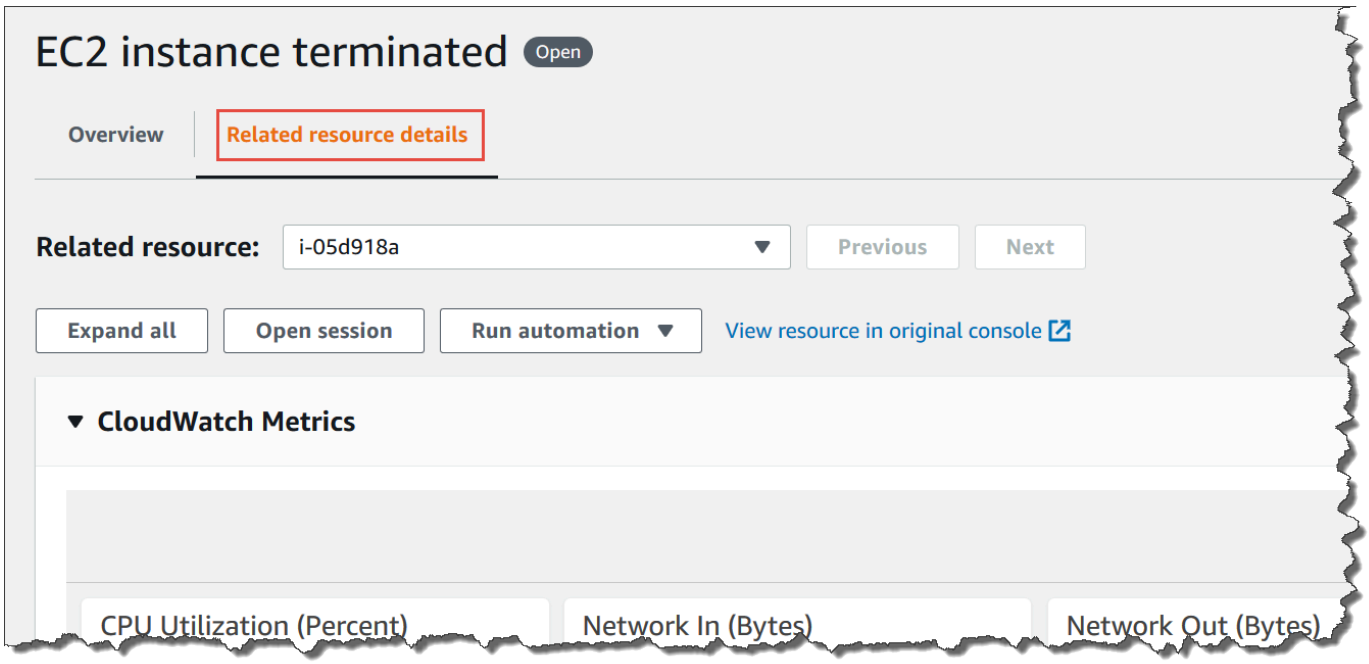
Untuk melihat dan menambahkan sumber daya terkait untuk OpsItem

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.

2. Di panel navigasi, pilih OpsCenter.
3. Pilih OpsItemstab.
4. Pilih OpsItem ID.



5. Untuk melihat informasi tentang sumber daya yang terdampak, pilih tab Detail sumber daya terkait.



Tab ini menampilkan informasi tentang sumber daya dari beberapaLayanan AWS. Perluas bagian Detail sumber daya ini sebagaimana disediakan oleh Layanan AWS yang menjadi host. Anda juga dapat beralih melalui sumber daya terkait lainnya OpsItem dengan menggunakan daftar Sumber daya terkait.

6. Untuk menambahkan sumber daya terkait tambahan, pilih tab Gambaran Umum.
7. Di bagian Sumber daya terkait, pilih Tambahkan.
8. Untuk Jenis sumber daya, pilih sumber daya dari daftar.

9. Untuk ID Sumber Daya masukkan ID atau Amazon Resource Name (ARN). Jenis informasi yang Anda pilih tergantung pada sumber daya yang Anda pilih pada langkah sebelumnya.

> **Note**
>
> Anda dapat secara manual menambahkan ARN sumber daya terkait tambahan. Masing-masing OpsItem dapat mencantumkan maksimum 100 ARN terkait.

Tabel berikut mencantumkan tipe sumber daya yang secara otomatis membuat tautan mendalam untuk sumber daya terkait.

Jenis sumber daya yang mendukung

Nama sumber daya	Format ARN
Sertifikat AWS Certificate Manager	`arn:aws:acm: region:account-id :certificate/ certificate-id`
Grup Amazon EC2 Auto Scaling	`arn:aws:autoscaling: region:account-id :autoScalingGroup: groupid:autoScalingGroupName/ groupfriendlyname`
CloudFrontDistribusi Amazon	`arn:aws:cloudfront:: account-id :*`
Tumpukan AWS CloudFormation	`arn:aws:cloudformation: region:account-id :stack/stackname /additionalidentifier`
CloudWatchAlarm Amazon	`arn:aws:cloudwatch: region:account-id :alarm:alarm-name`
Jejak AWS CloudTrail	`arn:aws:cloudtrail: region:account-id :trail/trailname`

Nama sumber daya	Format ARN
Proyek AWS CodeBuild	`arn:aws:codebuild:region:account-id:resourcetype/resource`
AWS CodePipeline	`arn:aws:codepipeline:region:account-id:resource-specifier`
Wawasan Amazon DevOps Guru	`arn:aws:devops-guru:region:account-id:insight/proactive or reactive/resource-id`
Tabel Amazon DynamoDB	`arn:aws:dynamodb:region:account-id:table/tablename`
Gateway pelanggan Amazon Elastic Compute Cloud (Amazon EC2)	`arn:aws:ec2:region:account-id:customer-gateway/cgw-id`
IP elastis Amazon EC2	`arn:aws:ec2:region:account-id:eip/eipalloc-id`
Host Khusus Amazon EC2	`arn:aws:ec2:region:account-id:dedicated-host/host-id`
Instans Amazon EC2	`arn:aws:ec2:region:account-id:instance/instance-id`
Gateway internet Amazon EC2	`arn:aws:ec2:region:account-id:internet-gateway/igw-id`
Daftar kontrol akses (ACL jaringan jaringan) jaringan Amazon Amazon EC2	`arn:aws:ec2:region:account-id:network-acl/nacl-id`

Nama sumber daya	Format ARN
Antarmuka jaringan Amazon EC2	`arn:aws:ec2: region:account-id :network-interface/ eni-id`
Tabel rute Amazon EC2	`arn:aws:ec2: region:account-id :route-table/ route-table-id`
Grup keamanan Amazon EC2	`arn:aws:ec2: region:account-id :security-group/ security-group-id`
subnet Amazon EC2	`arn:aws:ec2: region:account-id :subnet/subnet-id`
volume Amazon EC2	`arn:aws:ec2: region:account-id :volume/volume-id`
VPC Amazon EC2	`arn:aws:ec2: region:account-id :vpc/vpc-id`
Koneksi VPN Amazon EC2	`arn:aws:ec2: region:account-id :vpn-connection/ vpn-id`
Gateway VPN Amazon EC2	`arn:aws:ec2: region:account-id :vpn-gateway/ vgw-id`
Aplikasi 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`

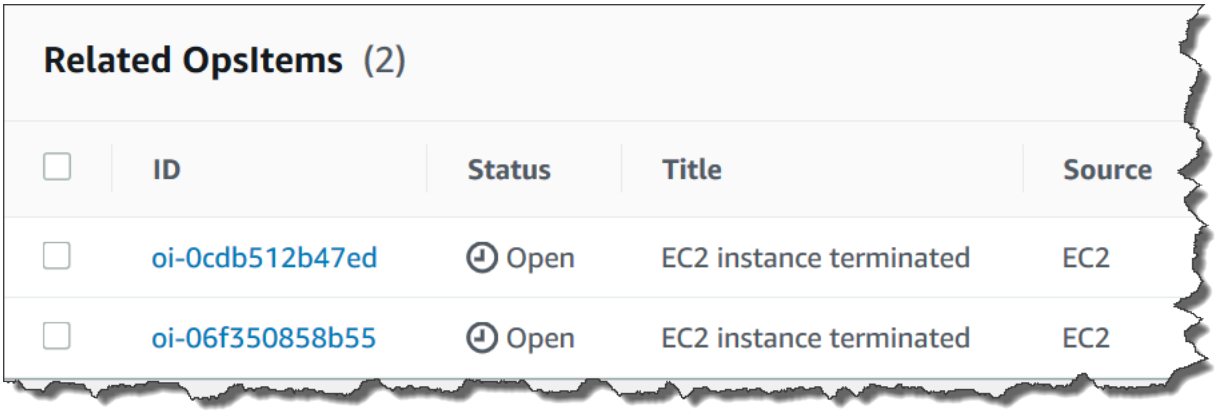
Nama sumber daya	Format ARN
Elastic Load Balancing (Application Load Balancer)	`arn:aws:elasticloadbalancing: region:account-id :loadbalancer/app/ load-balancer-name /load-balancer-id`
Elastic Load Balancing (Network Load Balancer)	`arn:aws:elasticloadbalancing: region:account-id :loadbalancer/net/ load-balancer-name /load-balancer-id`
Grup AWS Identity and Access Management (IAM)	`arn:aws:iam:: account-id :group/group-name`
Kebijakan IAM	`arn:aws:iam:: account-id :policy/policy-name`
IAM role	`arn:aws:iam:: account-id :role/role-name`
Pengguna IAM	`arn:aws:iam:: account-id :user/user-name`
Fungsi AWS Lambda	`arn:aws:lambda: region:account-id :function: function-name`
Klaster Amazon Relational Database Service (Amazon RDS)	`arn:aws:rds: region:account-id :cluster: db-cluster-name`
Instans basis data Amazon RDS	`arn:aws:rds: region:account-id :db:db-instance-name`

Nama sumber daya	Format ARN
Langganan Amazon RDS	`arn:aws:rds: region:account-id :es:subscription-name`
Grup keamanan Amazon RDS	`arn:aws:rds: region:account-id :secgrp:security-group-name`
snapshot klaster Amazon RDS	`arn:aws:rds: region:account-id :cluster-snapshot: cluster-snapshot-name`
Grup subnet Amazon RDS	`arn:aws:rds: region:account-id :subgrp:subnet-group-name`
Klaster Amazon Redshift	`arn:aws:redshift: region:account-id :cluster: cluster-name`
Grup parameter Amazon Redshift	`arn:aws:redshift: region:account-id :parametergroup: parameter-group-name`
Grup keamanan Amazon Redshift	`arn:aws:redshift: region:account-id :securitygroup: security-group-name`
snapshot klaster Amazon Redshift	`arn:aws:redshift: region:account-id :snapshot: cluster-name /snapshot-name`
Grup subnet Amazon Redshift	`arn:aws:redshift: region:account-id :subnetgroup: subnet-group-name`
bucket Amazon Simple Storage Service (Amazon S3)	`arn:aws:s3::: bucket_name`

Nama sumber daya	Format ARN
AWS Configpencatatan inventaris node AWS Systems Manager terkelola	`arn:aws:ssm: region:account-id :managed-instance-inventory/ node_id`
State ManagerAsosiasi Systems Manager	`arn:aws:ssm: region:account-id :association/ association_ID`

MenambahkanOpsItems terkait denganOpsItem

Dengan menggunakan halaman TerkaitOpsItemsOpsItems Detail, Anda dapat menyelidiki masalah operasi dan menyediakan konteks untuk suatu masalah. OpsItemsdapat dikaitkan dengan cara yang berbeda, termasuk hubungan induk-anak antaraOpsItems, akar masalah, atau duplikat. Anda dapat mengasosiasikan satuOpsItem dengan yang lain untukOpsItem menampilkannya di bagian Terkait. Anda dapat menentukan maksimum 10 ID untuk lainOpsItems yang berkaitan dengan arusOpsItem.

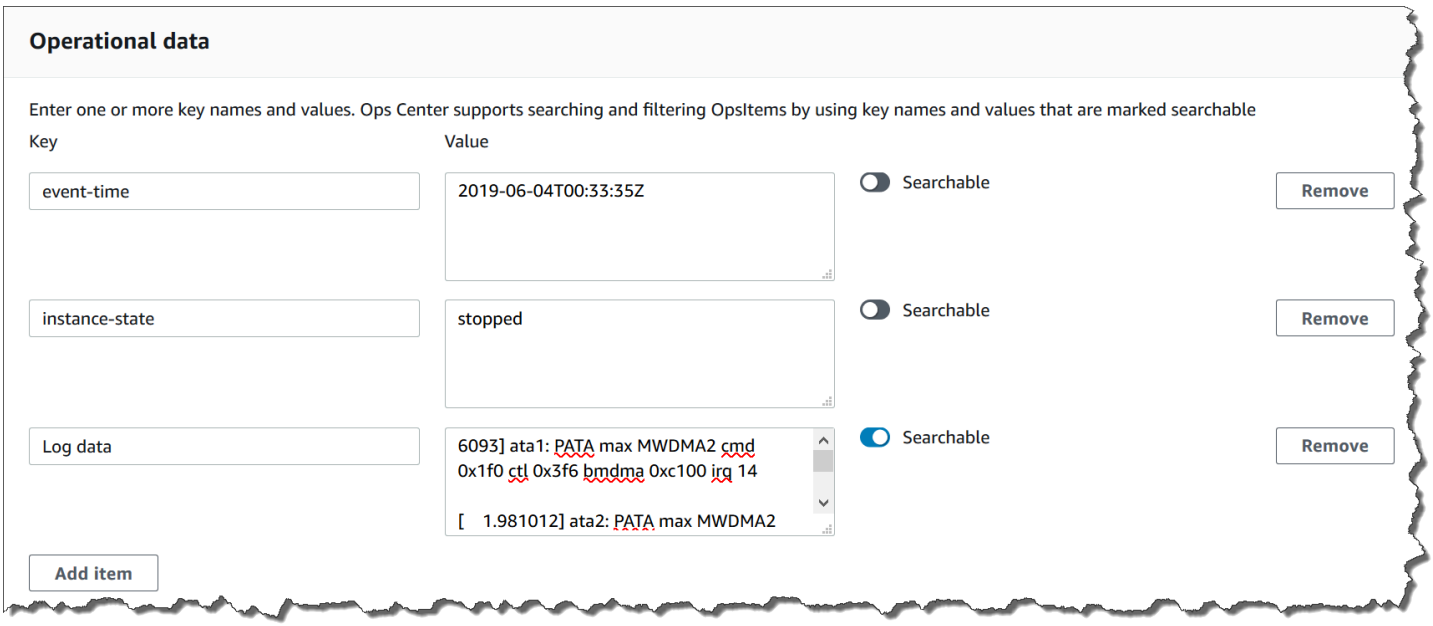


Untuk menambahkan terkaitOpsItem

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter.
3. PilihOpsItem ID untuk membuka halaman detail.
4. DiOpsItem bagian Terkait, pilih Tambah.
5. Untuk OpsItemID, tentukan ID.
6. Pilih Tambahkan.

Menambahkan data operasional keOpsItem

Data operasional adalah data kustom yang menyediakan data kustom yang menyediakan detail referensi berguna tentang data operasional adalah data kustom yang menyediakan detail referensi berguna tentangOpsItem. Anda dapat memasukkan data operasional sebagai pasangan nilai kunci. Misalnya, Anda dapat menentukan berkas log, string kesalahan, kunci lisensi, kiat pemecahan masalah, atau data lain yang relevan. Panjang maksimum kunci bisa 128 karakter, dan ukuran maksimum nilainya bisa 20 KB.



Anda dapat membuat data dapat dicari dapat dicari dapat dicari dapat dicari dapat dicari dapat dicari dengan pengguna lain di akun, atau Anda dapat membatasi akses pencarian. Data yang dapat dicari artinya semua pengguna dengan akses ke halaman Gambaran Umum ke halamanOpsItemGambaran Umum ke halaman Gambaran Umum (seperti yang disediakan oleh operasiOpsItems API Descricricris) dapat melihat dan mencari pada data yang ditentukan. data operasional yang tidak dapat dicari tidak dapat dicari hanya dapat dicari hanya dapat dicari hanya dapat dicari hanya dapat dapat dicari hanya dapat dapat dicari hanya dapat dapat dapat dicari hanya dapat dapat dapat dapat dicari hanya dapat dapat dapat dicari GetOpsItemhanya dapat dapat dicari hanya dapatOpsItem

Untuk menambahkan data operasional keOpsItem

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter.
3. PilihOpsItem ID untuk membuka halaman detailnya untuk membuka halaman detailnya.

4. Perluas data operasional.
5. Jika tidak ada data operasional untukOpsItem, pilih Tambahkan. Jika ada data operasional untukOpsItem, pilih Kelola.

   Setelah Anda membuat data operasional, Anda dapat mengedit kunci dan nilai, menghapus data operasional, atau menambahkan pasangan kunci-nilai tambahan dengan memilih Kelola.
6. Untuk Kunci, tentukan satu atau beberapa kata untuk membantu pengguna memahami tujuan data.

   > **Important**
   >
   > Kunci data operasional tidak dapat diawali dengan yang berikut ini: `amazon`, `aws`, `amzn`, `ssm`, `/amazon`, `/aws`, `/amzn`, `/ssm`.
7. Untuk Nilai, tentukan data.
8. Pilih Simpan.

> **Note**
>
> Anda dapat memfilterOpsItems dengan menggunakan operator data operasional pada OpsItemshalaman. Di kotak Pencarian, pilih data operasional, pilih data operasional, dan kemudian masukkan pasangan nilai kunci di JSON. Anda harus memasukkan pasangan kunci-nilai dengan menggunakan format berikut: `{"key":"key_name","value":"a_value"}`

Membuat insiden untukOpsItem

Gunakan prosedur berikut untuk secara manual membuat insiden untuk untukOpsItem untuk untuk untuk melacak dan mengelola dalamAWS Systems Manager Incident Manager, yang merupakan kemampuanAWS Systems Manager. insiden adalah gangguan yang tidak direncanakan atau penurunan kualitas layanan. Untuk informasi selengkapnya tentang Incident Manager, lihatthe section called “Integrasikan OpsCenter dengan yang lain Layanan AWS”.

Untuk secara manual membuat insiden untukOpsItem

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.

2. Di panel navigasi, pilih OpsCenter.
3. Jika Incident ManagerOpsItem membuatkan untuk Anda, pilih dan lanjutkan ke langkah 5. Jika tidak, pilih BuatOpsItem dan lengkapi formulirnya. Jika Anda tidak melihat tombol OpsItemsini, pilih BuatOpsItem.
4. Jika Anda membuatOpsItem, buka.
5. Pilih Mulai insiden.
6. Untuk Rencana tanggapan, pilih rencana tanggapan Incident Manager yang ingin Anda tetapkan pada insiden ini.
7. (Opsional) Untuk Judul, masukkan nama deskriptif untuk membantu anggota tim lain memahami sifat insiden tersebut. Jika Anda tidak memasukkan judul baru, pilihOpsCenterOpsItem dan insiden yang sesuai di Incident Manager akan menggunakan judul tersebut dalam rencana tanggapan.
8. (Opsional) Untuk Dampak insiden, pilih tingkat dampak untuk insiden ini. Jika Anda tidak memilih tingkat dampakOpsCenter,OpsItem pilih tingkat dampak, pilih tingkat dampak tersebut dalam rencana tanggapan.
9. Pilih Mulai.

Mengelola duplikatOpsItems

OpsCenterdapat menerima beberapa duplikatOpsItems untuk satu sumber dari beberapaLayanan AWS. OpsCentermenggunakan kombinasi logika tertanam dan string deduplikasi yang dapat dikonfigurasi untuk menghindari membuat duplikat duplikatOpsItems. AWS Systems Managermenerapkan logika tertanam deduplikasi ketika operasi CreateOpsItem API dipanggil.

AWS Systems Managermenggunakan logika deduplikasi berikut:

1. Saat membuatOpsItem, Systems Manager membuat dan menyimpan hash berdasarkan string deduplikasi dan sumber daya yang memulaiOpsItem.
2. Ketika permintaan lain dibuat untuk membuatOpsItem, sistem memeriksa string deduplikasi permintaan baru.
3. Jika ada hash yang cocok untuk string deduplikasi ini, Systems Manager memeriksa status yang adaOpsItem. Jika status yangOpsItem ada terbuka atau sedang berlangsung,OpsItem tidak dibuat. Jika yangOpsItem ada diselesaikan, Systems Manager menciptakan yang baruOpsItem.

Setelah Anda membuatOpsItem, Anda tidak dapat mengedit atau mengubah string deduplikasi dalam hal ituOpsItem.

Untuk mengelola duplikatOpsItems, Anda dapat melakukan hal berikut:

- Edit string deduplikasi untuk EventBridge aturan Amazon yang ditargetkan OpsCenter. Untuk informasi selengkapnya, lihat Mengedit string deduplikasi dalam EventBridge aturan default.
- Tentukan string deduplikasi ketika Anda secara manual membuatOpsItem. Untuk informasi selengkapnya, lihat Menentukan string deduplikasi menggunakanAWS CLI.
- Tinjau dan selesaikan duplikatOpsItems menggunakan wawasan operasional. Anda dapat menggunakan runbook untuk menyelesaikan duplikatOpsItems.

  Untuk membantu Anda menyelesaikan duplikatOpsItems dan mengurangi jumlah yangOpsItems dibuat oleh sumber, Systems Manager menyediakan runbook otomatisasi. Untuk informasi, lihat Menyelesaikan duplikat OpsItems berdasarkan wawasan.

Mengedit string deduplikasi dalam EventBridge aturan default

Gunakan prosedur berikut untuk menentukan string deduplikasi untuk EventBridge aturan yang menargetkanOpsCenter.

Untuk mengedit string deduplikasi untuk EventBridge aturan

1. Masuk keAWS Management Console dan buka EventBridge konsol Amazon di https://console.aws.amazon.com/events/.
2. Di panel navigasi, pilih Aturan.
3. Pilih aturan, dan kemudian pilih Edit.
4. Pergi ke halaman Pilih target.
5. Di bagian Pengaturan tambahan, pilih Konfigurasikan transformator input.
6. Di kotak Template, temukan entri`"operationalData": { "/aws/dedup"` JSON dan string deduplikasi yang ingin Anda edit.

   Entri string deduplikasi dalam EventBridge aturan menggunakan format JSON berikut ini.

```
"operationalData": { "/aws/dedup": {"type": "SearchableString","value":
 "{\"dedupString\":\"Words the system should use to check for duplicate
 OpsItems\"}"}}
```

Inilah contohnya.

```
"operationalData": { "/aws/dedup": {"type": "SearchableString","value":
 "{\"dedupString\":\"SSMOpsCenter-EBS-volume-performance-issue\"}"}}
```

7. Edit string deduplikasi, lalu pilih Konfirmasi.
8. Pilih Selanjutnya.
9. Pilih Selanjutnya.
10. Pilih Perbarui aturan.

Menentukan string deduplikasi menggunakanAWS CLI

Anda dapat menentukan string deduplikasi ketika Anda secara manual membuat baruOpsItem dengan menggunakan baikAWS Systems Manager konsol atauAWS CLI. Untuk informasi tentang memasukkan string deduplikasi ketika Anda secara manual membuatOpsItem di konsol, lihatBuat OpsItems secara manual. Jika Anda menggunakanAWS CLI, Anda dapat memasukkan string deduplikasi untuk`OperationalData` parameter. Sintaks parameter menggunakanJSON, seperti yang ditunjukkan dalam contoh berikut ini.

```
--operational-data '{"/aws/dedup":{"Value":"{\"dedupString\": \"Words the system should
 use to check for duplicate OpsItems\"}","Type":"SearchableString"}}'
```

Berikut adalah contoh perintah yang menentukan string deduplikasi `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"
```

Menganalisis wawasan operasional untuk mengurangi OpsItems

OpsCenter wawasan operasional menampilkan informasi tentang duplikatOpsItems. OpsCentersecara otomatis menganalisis OpsItems di akun Anda dan menghasilkan tiga jenis wawasan. Anda dapat melihat informasi ini di bagian Wawasan operasional pada tab OpsCenter Ringkasan.

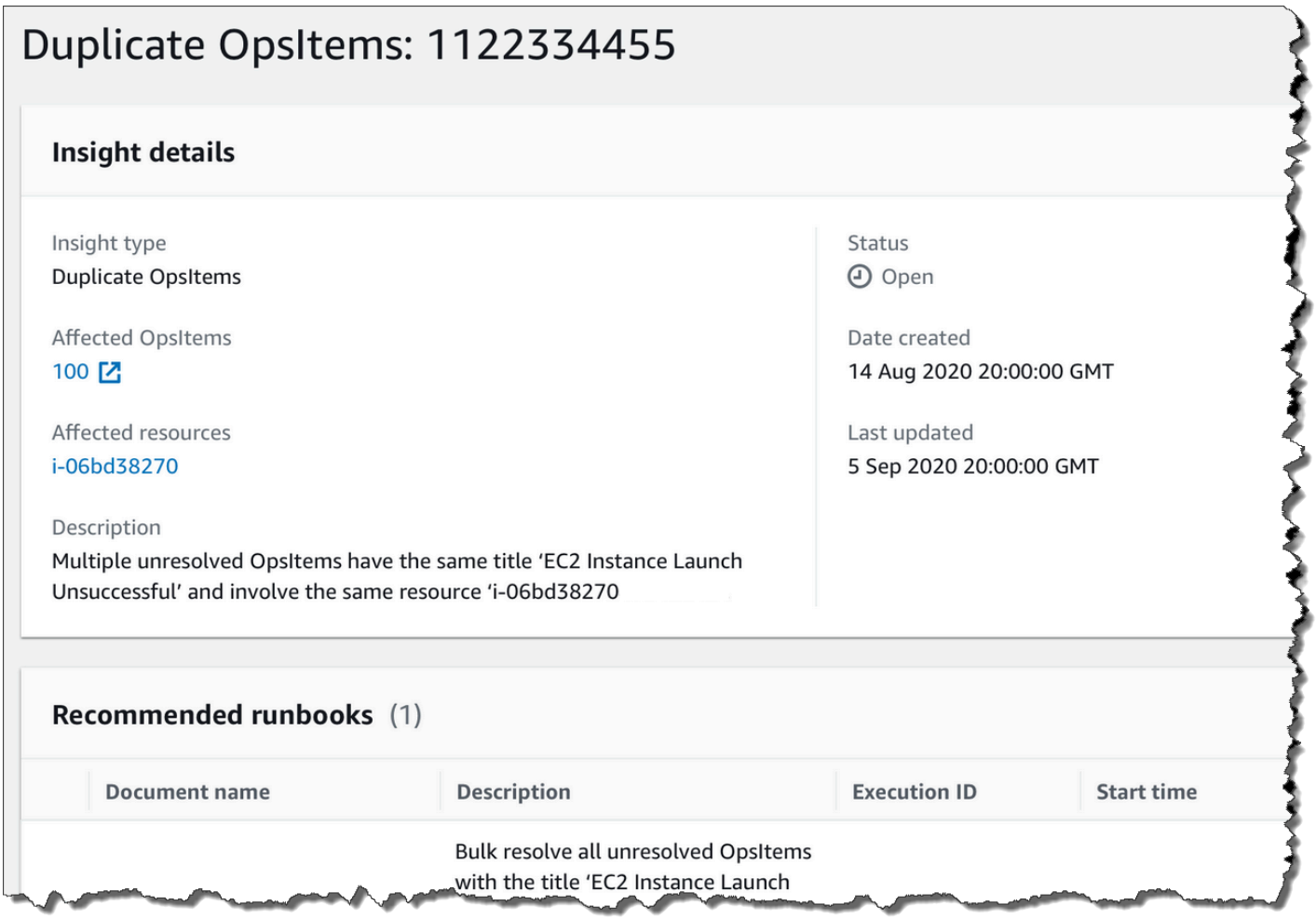
- Duplikat OpsItems — Wawasan dihasilkan ketika delapan atau lebih OpsItems memiliki judul yang sama untuk sumber daya yang sama.
- Judul yang paling umum — Wawasan dihasilkan ketika lebih dari 50 OpsItems memiliki judul yang sama.
- Sumber daya menghasilkan paling banyak OpsItems — Wawasan dihasilkan ketika AWS sumber daya memiliki lebih dari 10 terbukaOpsItems. Wawasan ini dan sumber daya yang sesuai ditampilkan di OpsItems tabel Resources yang menghasilkan paling banyak di tab OpsCenter Ringkasan. Sumber daya terdaftar dalam urutan penghitungan yang menurun. OpsItem

> **Note**
>
> OpsCentermenciptakan Sumber daya yang menghasilkan OpsItems wawasan terbanyak untuk jenis sumber daya berikut:
>
> - Instans Amazon Elastic Compute Cloud (Amazon EC2)
> - Grup keamanan Amazon EC2
> - Grup Amazon EC2 Auto Scaling
> - Basis data Amazon Relational Database Service (Amazon RDS)
> - Kluster Amazon RDS
> - Fungsi AWS Lambda

- Tabel Amazon DynamoDB
- Penyeimbang beban Elastic Load Balancing
- Klaster Amazon Redshift
- Sertifikat AWS Certificate Manager
- Volume Toko Blok Elastis Amazon

OpsCentermemberlakukan batas 15 wawasan per jenis. Jika suatu tipe mencapai batas ini, OpsCenter berhenti menampilkan lebih banyak wawasan untuk tipe tersebut. Untuk melihat wawasan tambahan, Anda harus menyelesaikan semua OpsItems yang terkait dengan OpsInsight jenis itu. Jika insight yang tertunda dicegah ditampilkan di konsol karena batas 15 wawasan, wawasan tersebut menjadi terlihat setelah wawasan lain ditutup.

Ketika Anda memilih wawasan, OpsCenter menampilkan informasi tentang yang terpengaruh OpsItems dan sumber daya. Tangkapan layar berikut menunjukkan contoh dengan detail OpsItem wawasan duplikat.



Wawasan operasional dimatikan secara default. Untuk informasi selengkapnya tentang bekerja dengan wawasan operasional, lihat topik berikut.

Topik

- Mengaktifkan wawasan operasional
- Menyelesaikan duplikat OpsItems berdasarkan wawasan
- Menonaktifkan wawasan operasional

Mengaktifkan wawasan operasional

Anda dapat mengaktifkan wawasan operasional pada OpsCenterhalaman di konsol Systems Manager. Saat Anda mengaktifkan wawasan operasional, Systems Manager membuat peran terkait layanan AWS Identity and Access Management (IAM) yang disebut. `AWSServiceRoleForAmazonSSM_OpsInsights` Peran tertaut layanan adalah tipe IAM role unik yang ditautkan langsung ke Systems Manager. Peran terkait layanan telah

ditentukan sebelumnya dan mencakup semua izin yang diperlukan layanan untuk memanggil orang lain Layanan AWS atas nama Anda. Untuk informasi selengkapnya tentang peran `AWSServiceRoleForAmazonSSM_OpsInsights` terkait layanan, lihat. Menggunakan peran untuk menciptakan wawasan operasional OpsItems di Manajer SistemOpsCenter

> **Note**
>
> Perhatikan informasi penting berikut:
>
> - Anda Akun AWS dikenakan biaya untuk wawasan operasional. Untuk informasi selengkapnya, lihat AWS Systems Manager Harga.
> - OpsCentermenyegarkan wawasan secara berkala menggunakan proses batch. Ini berarti daftar wawasan yang ditampilkan OpsCenter mungkin tidak sinkron.

Gunakan prosedur berikut untuk mengaktifkan dan melihat wawasan operasional diOpsCenter.

Untuk mengaktifkan dan melihat wawasan operasional

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter.
3. Di kotak pesan Wawasan operasional tersedia, pilih Aktifkan. Jika Anda tidak melihat pesan ini, gulir ke bawah ke bagian Wawasan operasional dan pilih Aktifkan.
4. Setelah Anda mengaktifkan fitur ini, pada tab Ringkasan, gulir ke bawah ke bagian Wawasan operasional.
5. Untuk melihat daftar wawasan yang difilter, pilih tautan di samping Duplikat OpsItems, Judul paling umum, atau Sumber daya yang paling banyak menghasilkan. OpsItems Untuk melihat semua wawasan, pilih Lihat semua wawasan operasional.
6. Pilih ID wawasan untuk melihat informasi selengkapnya.

Menyelesaikan duplikat OpsItems berdasarkan wawasan

Untuk menyelesaikan wawasan, Anda harus terlebih dahulu menyelesaikan semua OpsItems yang terkait dengan wawasan. Anda dapat menggunakan `AWS-BulkResolveOpsItemsForInsight` runbook untuk menyelesaikan OpsItems terkait dengan wawasan.

Untuk membantu Anda menyelesaikan duplikat OpsItems dan mengurangi jumlah yang OpsItems dibuat oleh sumber, Systems Manager menyediakan runbook otomatisasi berikut:

- `AWS-BulkResolveOpsItems`Runbook menyelesaikan OpsItems yang cocok dengan filter tertentu.
- `AWS-AddOpsItemDedupStringToEventBridgeRule`Runbook menambahkan string deduplikasi untuk semua OpsItem target yang terkait dengan aturan Amazon tertentu. EventBridge Runbook ini tidak menambahkan string deduplikasi jika aturan sudah memilikinya.
- `AWS-DisableEventBridgeRule`Mematikan aturan EventBridge jika aturan menghasilkan lusinan atau ratusanOpsItems.

Untuk menyelesaikan wawasan operasional

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter.
3. Pada tab Ikhtisar, gulir ke bawah ke wawasan Operasional.
4. Pilih Lihat semua wawasan operasional.
5. Pilih ID wawasan untuk melihat informasi selengkapnya.
6. Pilih runbook dan pilih Execute.

Menonaktifkan wawasan operasional

Saat Anda mematikan wawasan operasional, sistem berhenti membuat wawasan baru dan berhenti menampilkan wawasan di konsol. Wawasan aktif apa pun tetap tidak berubah di sistem, meskipun Anda tidak akan melihatnya ditampilkan di konsol. Jika Anda mengaktifkan fitur ini lagi, sistem akan menampilkan wawasan yang sebelumnya belum terselesaikan dan mulai membuat wawasan baru. Gunakan prosedur berikut untuk mematikan wawasan operasional.

Untuk mematikan wawasan operasional

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter.
3. Pilih Pengaturan.
4. Di bagian Wawasan operasional, pilih Edit lalu alihkan opsi Nonaktifkan.
5. Pilih Simpan.

MelihatOpsCenter log dan laporan

AWS CloudTrailmencatat panggilanAWS Systems ManagerOpsCenter API ke konsol,AWS Command Line Interface (AWS CLI), dan SDK. Anda dapat melihat informasi tersebut di CloudTrail konsol atau di bucket Amazon Simple Storage Service (Amazon S3). Amazon S3 menggunakan satu bucket untuk menyimpan semua CloudTrail log untuk akun Anda.

LogOpsCenter tindakan menunjukkanOpsItem aktivitas membuat, memperbarui, mendapatkan, dan menggambarkan. Untuk informasi selengkapnya tentang melihat dan menggunakan CloudTrail log aktivitas Systems Manager, lihatPencatatan panggilan AWS Systems Manager API dengan AWS CloudTrail.

AWS Systems ManagerOpsCentermemberi Anda informasi berikut tentangOpsItems:

- OpsItemOpsItemsRingkasan OOO
- SumberOpsItems dengan paling banyak OOLayanan AWSOpsItems
- OpsItemsberdasarkan sumber dan usia - Menyediakan hitunganOpsItems, dikelompokkan berdasarkan sumber dan hari sejak pembuatan.

Untuk melihatOpsCenter

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter.
3. Pada halamanOpsItems Ikhtisar, pilih Ringkasan.
4. Di bawah OpsItemsberdasarkan sumber dan usia, pilih bilah Pencarian untuk mem-filterOpsItems menurut Sumber. Gunakan daftar untuk mem-filter menurut Status.

Hapus OpsItems

Anda dapat menghapus individu OpsItem dengan memanggil operasi Delete OpsItem API menggunakan AWS Command Line Interface atau AWS SDK. Anda tidak dapat menghapus file OpsItem diAWS Management Console. Untuk menghapusOpsItem, pengguna, grup, atau peran AWS Identity and Access Management (IAM) Anda harus memiliki izin administrator atau Anda harus diberi izin untuk memanggil operasi `DeleteOpsItem` API.

Important

Perhatikan informasi penting berikut tentang operasi ini.

- Menghapus tidak dapat OpsItem diubah. Anda tidak dapat mengembalikan yang dihapusOpsItem.
- Operasi ini menggunakan model konsistensi akhirnya, yang berarti sistem dapat memakan waktu beberapa menit untuk menyelesaikan operasi ini. Jika Anda menghapus panggilan OpsItem dan segera, misalnya, Dapatkan OpsItem, yang dihapus OpsItem mungkin masih muncul dalam respons.
- Operasi ini bersifat idempoten. Sistem tidak memberikan pengecualian jika Anda berulang kali memanggil operasi ini untuk hal yang samaOpsItem. Jika panggilan pertama berhasil, semua panggilan tambahan mengembalikan respons sukses yang sama dengan panggilan pertama.
- Operasi ini tidak mendukung panggilan lintas akun. Administrator atau akun manajemen yang didelegasikan tidak dapat menghapus OpsItems di akun lain, meskipun OpsCenter telah disiapkan untuk administrasi lintas akun. Untuk informasi selengkapnya tentang administrasi lintas akun, lihat(Opsional) Menyiapkan OpsCenter untuk mengelola secara terpusat OpsItems di seluruh akun.
- Jika Anda menerima`OpsItemLimitExceededException`, Anda dapat menghapus satu atau lebih OpsItems untuk mengurangi jumlah total Anda OpsItems di bawah batas kuota. Untuk informasi lebih lanjut tentang pengecualian ini, lihatMemecahkan masalah dengan OpsCenter.

Menghapus sebuah OpsItem

Gunakan prosedur berikut untuk menghapus fileOpsItem.

Untuk menghapus sebuah OpsItem

1. Instal dan konfigurasikan AWS CLI, jika Anda belum melakukannya. Untuk informasi selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI.
2. Jalankan perintah berikut. Ganti *ID* dengan ID yang ingin OpsItem Anda hapus.

```
aws ssm delete-ops-item --OpsItemId ID
```

Jika berhasil, perintah tidak mengembalikan data.

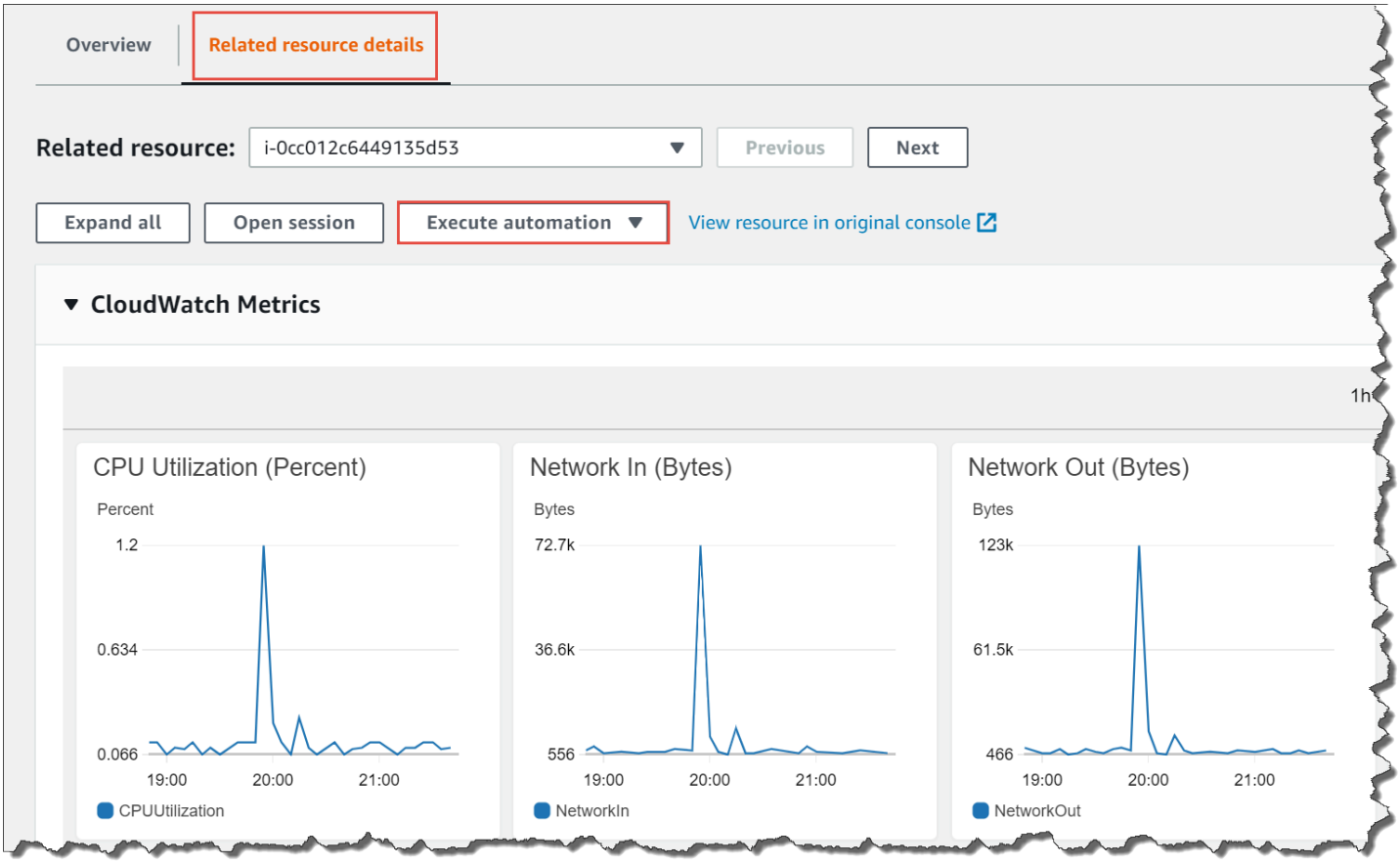
MemperbaikiOpsItem masalah

Menggunakan runbookAWS Systems Manager Otomasi, Anda dapat memulihkan masalah denganAWS sumber daya yang diidentifikasi dalamOpsItem. Otomatisasi menggunakan runbook yang telah ditentukan untuk memperbaiki masalah umum denganAWS sumber daya.

Masing-masingOpsItem termasuk bagian Runbooks yang menyediakan daftar runbook yang dapat Anda gunakan untuk remediasi. Ketika Anda memilih runbook Otomatisasi dari daftar,OpsCenter secara otomatis menampilkan beberapa bidang yang diperlukan untuk menjalankan dokumen. Ketika Anda menjalankan runbook Otomatisasi, sistem mengaitkan runbook dengan sumber daya terkaitOpsItem. Jika Amazon EventBridge membuatOpsItem, ia mengaitkan runbook denganOpsItem. OpsCentermenyimpan catatan 30-hari untuk runbook Otomatisasi untuk OpsItem.

Anda dapat memilih status untuk melihat detail penting tentang runbook, seperti alasan mengapa otomatisasi gagal dan langkah apa dari runbook Otomatisasi yang berjalan ketika kegagalan terjadi, seperti yang ditunjukkan dalam contoh berikut.

Halaman rincian sumber daya terkait untuk dipilihOpsItem termasuk daftar Jalankan otomatisasi. Anda dapat memilih runbook Otomatisasi terbaru atau khusus sumber daya dan menjalankannya untuk memperbaiki masalah. Halaman ini juga mencakup penyedia data, termasuk CloudWatch metrik dan alarm Amazon,AWS CloudTrail log, dan detail dariAWS Config.



Anda dapat melihat informasi tentang runbook Otomatisasi dengan memilih namanya di konsol atau dengan menggunakan Referensi runbook Otomatisasi Systems Manager.

RemediasiOpsItem menggunakan runbook

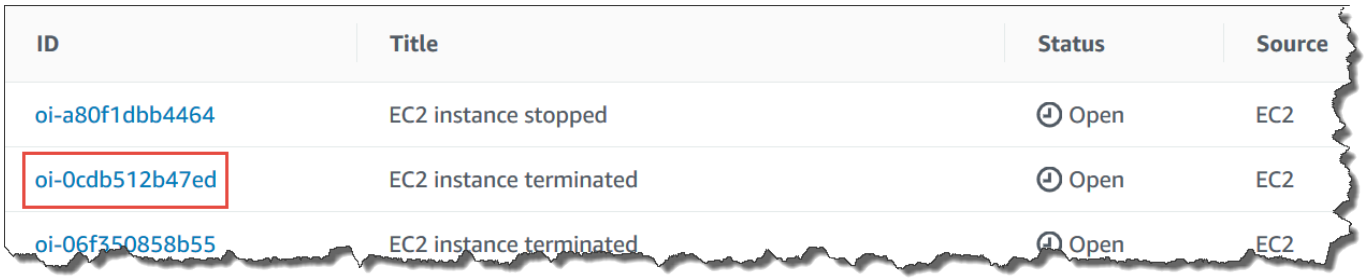
Sebelum Anda menggunakan runbook Otomatisasi untuk memperbaikiOpsItem masalah, lakukan hal berikut:

- Verifikasi bahwa Anda memiliki izin untuk menjalankan runbook Otomatisasi Systems Manager. Untuk informasi selengkapnya, lihat Menyiapkan Otomatisasi.
- Mengumpulkan informasi ID khusus sumber daya untuk otomatisasi yang ingin Anda jalankan. Misalnya, jika Anda ingin menjalankan otomatisasi yang memulai ulang instans EC2, maka Anda harus menentukan ID instans EC2 untuk memulai ulang.

Untuk menjalankan runbook Otomatisasi untuk memperbaikiOpsItem masalah

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter.

3. PilihOpsItem ID untuk membuka halaman detail.



4. Gulir ke bagian Runbook.
5. Gunakan bilah pencarian atau nomor di kanan atas untuk menemukan runbook Otomatisasi yang ingin Anda jalankan.
6. Pilih runbook, dan kemudian pilih Eksekusi.
7. Masukkan informasi yang diperlukan untuk runbook, dan kemudian pilih Kirim.

   Setelah Anda memulai runbook, sistem kembali ke layar sebelumnya dan menampilkan statusnya.
8. Di eksekusi Otomatisasi di bagian 30 hari terakhir, pilih tautan ID eksekusi untuk melihat langkah-langkah dan status eksekusi.

RemediasiOpsItem menggunakan runbook terkait

Setelah Anda menjalankan runbook Otomatisasi dariOpsItem,OpsCenter runbook dengan runbook dengan runbookOpsItem. Runbook terkait diberi peringkat lebih tinggi daripada runbook lain dalam daftar Runbooks.

Gunakan prosedur berikut ini untuk menjalankan runbook Otomatisasi yang telah dikaitkan dengan sumber daya terkait diOpsItem. Untuk informasi tentang menambahkan sumber daya terkait, lihat MengelolaOpsItems.

Untuk menjalankan runbook terkait sumber daya untuk memperbaikiOpsItem masalah

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter.
3. Buka OpsItem.
4. Di bagian Sumber daya terkait, pilih sumber daya tempat Anda ingin menjalankan runbook Otomatisasi.

5. Pilih Jalankan otomatisasi, dan kemudian pilih runbook Otomatisasi terkait yang ingin Anda jalankan.
6. Masukkan informasi yang diperlukan untuk runbook, dan kemudian pilih Eksekusi.

   Setelah Anda memulai runbook, sistem kembali ke layar sebelumnya dan menampilkan statusnya.
7. Di eksekusi Otomatisasi di bagian 30 hari terakhir, pilih tautan ID eksekusi untuk melihat langkah-langkah dan status eksekusi.

Melihat laporan ringkasan Melihat laporan OpsCenter ringkasan Melihat

AWS Systems ManagerOpsCentermencakup halaman ringkasan yang secara otomatis menampilkan informasi:

- OpsItemringkasan status - Ringkasan OpsItems berdasarkan status, seperti `Open` dan`In progress`.
- Sumber dengan paling terbuka OpsItems - Rincian bagian atas Layanan AWS yang memiliki terbukaOpsItems.
- OpsItemsberdasarkan sumber dan usia - hitunganOpsItems, dikelompokkan berdasarkan sumber dan hitungan hari sejak pembuatan.

Untuk melihat laporan OpsCenter ringkasan

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter, lalu pilih tab Ringkasan.
3. Di bagian OpsItemsberdasarkan sumber dan usia, lakukan hal berikut:

   1. (Opsional) Di bidang filter, pilih Sumber`Equal`, pilih`Begin With`,`Not Equal`, atau, lalu masukkan parameter pencarian.
   2. Dalam daftar berdekatan, pilih salah satu nilai status berikut:
      - `Open`
      - `In progress`
      - `Resolved`
      - `Open and in progress`
      - `All`

Memecahkan masalah dengan OpsCenter

Topik ini mencakup informasi untuk membantu Anda memecahkan masalah kesalahan dan masalah umum. OpsCenter

Anda menerima OpsItemLimitExceededException

Jika Anda Akun AWS telah mencapai jumlah maksimum yang OpsItems diizinkan saat Anda memanggil operasi CreateOpsItem API, Anda menerima file`OpsItemLimitExceededException`. OpsCentermengembalikan pengecualian jika panggilan Anda akan melebihi jumlah maksimum OpsItems untuk salah satu dari kuota berikut:

- Jumlah total OpsItems Akun AWS per Wilayah (termasuk `Open` dan `Resolved` OpsItems): 500.000
- Jumlah maksimum OpsItems Akun AWS per bulan: 10.000

Kuota ini berlaku untuk OpsItems dibuat dari sumber apa pun kecuali yang berikut:

- OpsItemsdiciptakan oleh AWS Security Hub temuan
- OpsItemsyang dibuat secara otomatis ketika insiden Manajer Insiden dibuka

OpsItemsdibuat dari sumber-sumber ini tidak dihitung terhadap OpsItem kuota Anda, tetapi Anda dikenakan biaya untuk masing-masingOpsItem.

Jika Anda menerima`OpsItemLimitExceededException`, Anda dapat menghapus secara manual OpsItems hingga Anda berada di bawah kuota yang mencegah Anda membuat yang baruOpsItem. Sekali lagi, menghapus yang OpsItems dibuat untuk temuan Security Hub atau insiden Manajer Insiden tidak akan mengurangi jumlah total yang OpsItems diberlakukan oleh kuota. Anda harus menghapus OpsItems dari sumber lain. Untuk informasi tentang cara menghapusOpsItem, lihatHapus OpsItems.

Anda menerima tagihan besar dari AWS untuk sejumlah besar yang dibuat secara otomatis OpsItems

Jika Anda mengonfigurasi integrasi denganAWS Security Hub, OpsCenter OpsItems buat temuan Security Hub. Bergantung pada jumlah pencarian yang dihasilkan oleh Security Hub dan akun yang Anda masuki saat mengonfigurasi integrasi, OpsCenter dapat menghasilkan sejumlah

besarOpsItems, dengan biaya tertentu. Berikut adalah rincian yang lebih spesifik terkait dengan yang OpsItems dihasilkan oleh temuan Security Hub:

- Jika Anda masuk ke akun administrator Security Hub saat mengonfigurasi OpsCenter dan integrasi Security Hub, sistem akan membuat OpsItems temuan di administrator dan semua akun anggota. Semuanya OpsItems dibuat di akun administrator. Bergantung pada berbagai faktor, ini dapat menyebabkan tagihan besar yang tak terduga dari. AWS

  Jika Anda masuk ke akun anggota saat mengonfigurasi integrasi, sistem hanya membuat OpsItems temuan di akun individu tersebut. Untuk informasi selengkapnya tentang akun administrator Security Hub, akun anggota, dan hubungannya dengan feed EventBridge peristiwa untuk temuan, lihat Jenis integrasi Security Hub dengan EventBridge dalam Panduan AWS Security Hub Pengguna.
- Untuk setiap temuan yang menghasilkanOpsItem, Anda dikenakan biaya reguler untuk membuatOpsItem. Anda juga dikenakan biaya jika Anda mengedit OpsItem atau jika temuan terkait diperbarui di Security Hub (yang memicu OpsItem pembaruan).

> **Important**
>
> Jika Anda yakin sejumlah besar OpsItems dibuat karena kesalahan dan AWS tagihan Anda tidak beralasan, hubungi. AWS Support

Gunakan prosedur berikut jika Anda tidak lagi ingin sistem membuat OpsItems temuan Security Hub.

Untuk berhenti menerima OpsItems temuan Security Hub

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih OpsCenter.
3. Pilih Pengaturan.
4. Di bagian temuan Security Hub, pilih Edit.
5. Pilih slider untuk mengubah Diaktifkan ke Dinonaktifkan. Jika Anda tidak dapat mengaktifkan slider, Security Hub belum diaktifkan untuk Anda. Akun AWS
6. Pilih Simpan untuk menyimpan konfigurasi Anda. OpsCentertidak lagi dibuat OpsItems berdasarkan temuan Security Hub.

> **Important**
>
> Jika OpsCenter mengubah pengaturan kembali ke Diaktifkan dan terus membuat OpsItems temuan, masuk ke akun administrator yang didelegasikan Systems Manager atau akun AWS Organizations manajemen dan ulangi prosedur ini. Jika Anda tidak memiliki izin untuk masuk ke salah satu akun tersebut, hubungi administrator Anda dan minta mereka mengulangi prosedur ini untuk menonaktifkan integrasi akun Anda.

CloudWatchDasbor Amazon dasbor Amazon dasbor Amazon dasbor Amazon

CloudWatch Dasbor Amazon dasbor adalah halaman beranda yang dapat disesuaikan di CloudWatch konsol yang dapat Anda gunakan untuk memantau sumber daya Anda dalam satu tampilan, bahkan sumber daya yang tersebar di berbagaiWilayah AWS. Anda dapat menggunakan CloudWatch dasbor untuk membuat tampilan metrik dan alarm yang disesuaikan untukAWS sumber daya Anda. Dengan dasbor, Anda dapat membuat hal berikut:

- Tampilan tunggal untuk metrik dan alarm yang dipilih untuk membantu Anda menilai kesehatan sumber daya dan aplikasi Anda di satu atau beberapa Wilayah AWS. Anda dapat memilih warna yang digunakan untuk setiap metrik pada setiap grafik, sehingga Anda dapat melacak metrik yang sama di beberapa grafik.
- Buku pedoman operasi yang memberikan panduan bagi anggota tim selama peristiwa operasi tentang cara merespons kejadian tertentu.
- Tampilan umum dari pengukuran sumber daya dan aplikasi penting yang dapat dibagikan oleh anggota tim untuk alur komunikasi yang lebih cepat selama peristiwa operasi.

Anda dapat membuat dasbor dengan menggunakan konsol,AWS Command Line Interface (AWS CLI), atau dengan menggunakan CloudWatch `PutDashboard` API. Untuk informasi selengkapnya, lihat Menggunakan CloudWatch Dasbor Amazon di Panduan CloudWatch Pengguna Amazon.

AWS Systems Manager Manajemen Aplikasi

Manajemen Aplikasi adalah rangkaian kemampuan yang membantu Anda mengelola aplikasi Anda berjalan di AWS.

Topik

AWS Systems Manager Application Manager

Application Manager, kemampuanAWS Systems Manager, membantu DevOps teknisi menyelidiki dan mengatasi masalah denganAWS sumber daya dalam konteks aplikasi dan klaster mereka. Application Managermengumpulkan informasi operasi dari beberapaLayanan AWS dan kemampuan Systems Manager ke satuAWS Management Console.

DiApplication Manager, aplikasi adalah grup logis dariAWS sumber daya yang ingin Anda operasikan sebagai satu unit. Kelompok logis ini dapat mewakili aplikasi, batas kepemilikan untuk operator, atau lingkungan developer dengan versi yang berbeda, untuk beberapa nama. Application Managerdukungan untuk klaster kontainer termasuk kluster Amazon Elastic Kubernetes Service (Amazon EKS) dan Amazon Elastic Container Service (Amazon ECS).

Bila Anda memilih Mulailah denganApplication Manager halaman beranda,Application Manager secara otomatis mengimpor metadata tentang sumber daya Anda yang dibuat di kemampuan Systems ManagerLayanan AWS atau lainnya. Untuk aplikasi,Application Manager impor metadata tentang semuaAWS sumber daya yang diatur dalam grup sumber daya. Setiap grup sumber daya terdaftar di kategori Aplikasi khusus sebagai aplikasi yang unik. Application Managerjuga secara otomatis mengimpor metadata tentang sumber daya yang dibuat olehAWS CloudFormation,,AWS Launch Wizard, Amazon ECS, dan Amazon EKS. Application Managerkemudian menampilkan sumber daya tersebut dalam kategori yang telah ditetapkan.

Untuk Aplikasi, daftar tersebut mencakup hal berikut:

- Aplikasi khusus
- Launch Wizard
- CloudFormation tumpukan

- AppRegistry aplikasi

Untuk Klaster kontainer, daftar ini mencakup hal berikut:

- Klaster Amazon ECS
- Klaster Amazon EKS

Setelah impor selesai, Anda dapat melihat informasi operasi tentang sumber daya Anda dalam kategori yang telah ditetapkan ini. Atau, jika Anda ingin memberikan lebih banyak konteks tentang koleksi sumber daya, Anda dapat secara manual membuat aplikasiApplication Manager dan memindahkan sumber daya atau grup sumber daya ke aplikasi tersebut. Hal ini mengizinkan Anda untuk menampilkan informasi operasi dalam konteks aplikasi.

Setelah Anda menyiapkan dan mengonfigurasiLayanan AWS dan kemampuan Systems Manager,Application Manager tampilkan beberapa jenis informasi tentang sumber daya Anda:

- Informasi tentang status, status, dan kondisi Amazon EC2 Auto Scaling dari instans Amazon Elastic Compute Cloud (Amazon EC2) di aplikasi Anda
- Alarm yang disediakan oleh Amazon CloudWatch
- Informasi kepatuhan yang diberikan olehAWS Config danState Manager (komponen Systems Manager)
- Informasi klaster Kubernetas yang disediakan oleh Amazon EKS
- Data log log log CloudWatch log log logAWS CloudTrail
- OpsItemsdisediakan oleh Systems ManagerOpsCenter
- Rincian sumber dayaLayanan AWS yang disediakan oleh host tersebut.
- Informasi klaster kontainer yang disediakan oleh Amazon ECS.

Untuk membantu Anda mengatasi masalah dengan komponen atau sumber daya,Application Manager juga menyediakan runbook yang dapat Anda kaitkan dengan aplikasi Anda. Untuk memulaiApplication Manager, buka konsol Systems Manager. Di panel navigasi, pilih Application Manager.

Apa saja manfaat menggunakanApplication Manager?

Application Managermengurangi waktu yang diperlukan oleh DevOps teknisi untuk mendeteksi dan menyelidiki masalah denganAWS sumber daya. Untuk melakukan ini,Application Manager

menampilkan beberapa jenis informasi operasi dalam konteks aplikasi dalam satu konsol. Application Managerjuga mengurangi waktu yang diperlukan untuk mengatasi masalah dengan menyediakan runbook yang melakukan tugas perbaikan umum padaAWS sumber daya.

Apa saja fitur dariApplication Manager?

Application Managermencakup fitur berikut:

- ImporAWS sumber daya Anda secara otomatis

  Selama penyiapan awal, Anda dapat memilih agarApplication Manager secara otomatis mengimpor dan menampilkan sumber daya di AndaAkun AWS yang didasarkan pada CloudFormation tumpukanAWS Resource Groups, penerapan Launch Wizard, serta klaster Amazon ECS dan Amazon EKS. AppRegistry Sistem ini menampilkan sumber daya ini dalam aplikasi atau klaster kategori yang telah ditetapkan. Setelah itu, setiap kali sumber daya baru dari jenis ini ditambahkan keAkun AWS,Application Manager secara otomatis menampilkan sumber daya baru dalam aplikasi dan kategori klaster yang telah ditetapkan.
- Membuat atau mengedit CloudFormation tumpukan dan templat

  Application Managermembantu Anda menyediakan dan mengelola sumber daya untuk aplikasi Anda dengan mengintegrasikan dengan CloudFormation. Anda dapat membuat, mengedit, dan menghapusAWS CloudFormation templat dan tumpukanApplication Manager. Application Managerjuga mencakup pustaka template tempat Anda dapat mengkloning, membuat, dan menyimpan template. Application Managerdan CloudFormation menampilkan informasi yang sama tentang status tumpukan saat ini. Pembaruan templat dan templat disimpan di Systems Manager hingga Anda menyediakan tumpukan, pada saat itu perubahan juga ditampilkan CloudFormation.
- Tampilkan informasi tentang instans Anda dalam konteks aplikasi

  Application Managerterintegrasi dengan Amazon Elastic Compute Cloud (Amazon EC2) untuk menampilkan informasi tentang instans Anda dalam konteks aplikasi. Application Managermenampilkan status instans, status, dan kesehatan Auto Scaling Amazon EC2 untuk aplikasi yang dipilih dalam format grafis. Tab Instances juga menyertakan tabel dengan informasi berikut untuk setiap instans dalam aplikasi Anda.
  - Status instans (Pending, Stopping, Running, Stopped)
  - Status ping untukSSM Agent
  - Status dan nama runbook Systems Manager Automation terakhir diproses pada instance
  - Jumlah alarm Amazon CloudWatch Logs per status.

    - `ALARM` – Metrik atau pernyataan berada di luar ambang batas yang ditetapkan.
    - `OK` – Metrik atau pernyataan berada dalam ambang batas yang ditetapkan.
    - `INSUFFICIENT_DATA` – Alarm baru saja dimulai, metrik tidak tersedia, atau tidak cukup data yang tersedia bagi metrik untuk menentukan status alarm.
  - Kesehatan grup Auto Scaling untuk kelompok penskalaan otomatis orang tua dan individu
- Tampilkan metrik operasional dan alarm

  Application Managerterintegrasi dengan Amazon CloudWatch untuk menyediakan metrik operasional real-time dan alarm untuk aplikasi atau klaster. Anda dapat menelusuri ke pohon aplikasi Anda untuk menampilkan alarm pada setiap tingkat komponen, atau menampilkan alarm untuk klaster individu.
- Tampilkan data log log log log

  Application Managerterintegrasi dengan Amazon CloudWatch Logs untuk menyediakan data log log dalam konteks aplikasi Anda tanpa harus meninggalkan Systems Manager.
- Tampilkan dan kelola dan kelolaOpsItems aplikasi atau klaster

  Application Managerterintegrasi denganAWS Systems Manager OpsCenter untuk menyediakan daftar item pekerjaan operasional (OpsItems) untuk aplikasi dan klaster Anda. Daftar tersebut mencerminkan secara otomatis dan dibuat secara manualOpsItems. Anda dapat melihat rincian tentang sumber daya yang membuatOpsItem danOpsItem status, sumber, dan tingkat keparahan.
- Tampilkan data kepatuhan sumber daya untuk aplikasi atau klaster

  Application Managerterintegrasi dengan AWS Configuntuk memberikan rincian kepatuhan dan riwayat tentangAWS sumber daya sesuai dengan aturan yang Anda tentukan. Application Managerjuga terintegrasi denganAWS Systems Manager State Manager untuk memberikan informasi kepatuhan tentang status yang ingin Anda pertahankan untuk instans Amazon Elastic Compute Cloud (Amazon EC2).
- Tampilkan informasi infrastruktur klaster Amazon ECS dan Amazon EKS

  Application Managerterintegrasi dengan Amazon ECS dan Amazon EKS untuk memberikan informasi tentang kesehatan infrastruktur klaster dan tampilan waktu aktif komponen komputasi, jaringan, dan penyimpanan sumber daya dalam sebuah klaster.

  Namun, Anda tidak dapat mengelola atau menampilkan informasi operasi tentang pod atau kontainer Amazon EKS AndaApplication Manager. Anda hanya dapat mengelola dan melihat informasi operasi tentang infrastruktur yang meng-host sumber daya Amazon EKS Anda.

- Melihat rincian biaya sumber daya untuk aplikasi

  Application Managerterintegrasi denganAWS Cost Explorer, fiturAWS Billing and Cost Management, melalui widget Biaya. Setelah Anda mengaktifkan Cost Explorer di konsol Billing and Cost Management, widget BiayaApplication Manager menunjukkan data biaya untuk aplikasi atau komponen aplikasi non-kontainer tertentu. Anda dapat menggunakan filter di widget untuk melihat data biaya sesuai dengan periode waktu, granularitas, dan jenis biaya yang berbeda baik dalam grafik batang atau garis.

- Tampilkan informasi sumber daya terperinci dalam satu konsol

  Pilih nama sumber daya yang tercantum dalamApplication Manager dan tampilkan informasi kontekstual dan operasi informasi tentang sumber daya tersebut tanpa harus meninggalkan Systems Manager.

- Terima pembaruan sumber daya otomatis untuk aplikasi

  Jika Anda membuat perubahan ke sumber daya di konsol layanan, dan sumber daya tersebut merupakan bagian dari aplikasiApplication Manager, maka Systems Manager secara otomatis menampilkan perubahan tersebut. Sebagai contoh, jika Anda memperbarui tumpukan diAWS CloudFormation konsol, dan jika tumpukan tersebut merupakan bagian dariApplication Manager aplikasi, maka pembaruan tumpukan secara otomatis tercermin dalamApplication Manager.

- Cari aplikasi Launch Wizard secara otomatis

  Application Manager terintegrasi dengan AWS Launch Wizard. Jika Anda menggunakan Launch Wizard untuk men-deploy sumber daya untuk aplikasi,Application Manager dapat secara otomatis mengimpor dan menampilkannya di bagian Launch Wizard.

- Memantau sumber daya aplikasiApplication Manager dengan menggunakan CloudWatch Application Insights

  Application Managerterintegrasi dengan Amazon CloudWatch Application Insights. Wawasan Aplikasi mengidentifikasi dan menyiapkan metrik kunci, log, dan alarm di seluruh sumber daya aplikasi dan tumpukan teknologi Anda. Wawasan Aplikasi secara terus menerus memantau metrik dan log untuk mendeteksi dan menghubungkan anomali dan kesalahan. Ketika sistem mendeteksi kesalahan atau anomali, Wawasan Aplikasi menghasilkan CloudWatch Events yang dapat Anda gunakan untuk mengatur notifikasi atau mengambil tindakan. Anda dapat mengaktifkan dan melihat Wawasan Aplikasi pada tab Ikhtisar dan Pemantauan diApplication Manager. Untuk informasi selengkapnya tentang Wawasan Aplikasi, lihat Apa itu Amazon CloudWatch Application Insights di Panduan CloudWatch Pengguna Amazon.

- Pulihkan masalah dengan runbook

  Application Managermencakup runbook Systems Manager yang telah ditetapkan sebelumnya untuk mengatasi masalah umum denganAWS sumber daya. Anda dapat menjalankan runbook terhadap semua sumber daya yang berlaku dalam aplikasi tanpa harus pergiApplication Manager.

Apakah ada biaya untuk digunakanApplication Manager?

Application Managertersedia tanpa biaya tambahan.

Untuk apa kuota sumber dayaApplication Manager?

Anda dapat melohat kuota untuk semua kemampuan Systems Manager di service quota Referensi Umum Amazon Web Services. Kecuali dinyatakan lain, masing-masing kuota bersifat khusus per Wilayah.

Topik

- Memulai dengan Systems ManagerApplication Manager
- Bekerja dengan Application Manager

Memulai dengan Systems ManagerApplication Manager

Gunakan informasi di bagian ini untuk membantu Anda mengatur dan mengonfigurasiApplication Manager, suatu kemampuanAWS Systems Manager, untuk menampilkan informasi operasi dari yang berbedaLayanan AWSKemampuan Systems Manager. Bagian ini juga mencakup informasi tentang menambahkan aplikasi dan klaster untukApplication Manager.

Topik

- Menyiapkan layanan terkait
- Mengonfigurasi izin untuk Systems Manager Application Manager
- Menambahkan aplikasi dan klaster untukApplication Manager

Menyiapkan layanan terkait

Application Manager, kemampuanAWS Systems Manager, menampilkan sumber daya dan informasi dari orang lainLayanan AWSdan kemampuan Manajer Sistem. Untuk memaksimalkan jumlah

informasi operasi yang ditampilkan diApplication Manager, kami menyarankan Anda mengatur dan mengonfigurasi layanan atau kemampuan lain inisebelumnyaAnda menggunakanApplication Manager.

Topik

- Menyiapkan tugas untuk mengimpor sumber daya
- Atur tugas untuk melihat informasi operasi tentang sumber daya

Menyiapkan tugas untuk mengimpor sumber daya

Tugas penyiapan berikut membantu Anda melihatAWSsumber daya diApplication Manager. Setelah masing-masing tugas ini selesai, Manajer Sistem dapat secara otomatis mengimpor sumber daya keApplication Manager. Setelah sumber daya Anda diimpor, Anda dapat membuat aplikasi diApplication Managerdan pindahkan sumber daya impor Anda ke dalamnya. Hal ini membantu Anda untuk menampilkan informasi operasi dalam konteks aplikasi.

(Opsional) Atur AWS sumber daya dengan menggunakan tag

Anda dapat menetapkan metadata ke sumber daya AWS Anda dalam bentuk tag. Setiap tag adalah label yang terdiri dari kunci dan nilai yang ditentukan pengguna. Tag membantu Anda mengelola, mengidentifikasi, mengatur, dan memfilter sumber daya. Anda dapat menggunakan tag untuk mengategorikan sumber daya berdasarkan tujuan, pemilik, lingkungan, atau kriteria lainnya.

(Opsional) Atur AWS sumber daya Anda dengan menggunakan AWS Resource Groups

Anda dapat menggunakan grup sumber daya untuk mengatur AWS sumber daya. Resource groups mempermudah Anda untuk mengelola, memantau dan mengotomatiskan tugas pada sejumlah besar sumber daya secara sekaligus.

Application Managersecara otomatis mengimpor semua grup sumber daya Anda dan mencantumkannya diAplikasi kustomkategori.

(Opsional) Atur dan deploy AWS sumber daya Anda dengan menggunakan AWS CloudFormation

AWS CloudFormation mengizinkan Anda untuk membuat dan menyediakan AWS deployment infrastruktur dengan cara yang dapat diprediksi dan berulang kali. Ini membantu Anda menggunakanLayanan AWSseperti Amazon EC2, Amazon Elastic Block Store (Amazon EBS), Amazon Simple Notification Service (Amazon SNS), Elastic Load Balancing, danAWSPenskalaan Otomatis. dengan CloudFormation, Anda dapat membangun aplikasi yang

andal, terukur, dan hemat biaya di cloud tanpa khawatir membuat dan mengonfigurasi yang mendasarinyaAWSinfrastruktur.

Application Managersecara otomatis mengimpor semuaAWS CloudFormationsumber daya dan daftar mereka diAWS CloudFormationtumpukankategori. Anda dapat membuat CloudFormation tumpukan dan templat diApplication Manager. Perubahan tumpukan dan template secara otomatis disinkronkan antaraApplication Managerdan CloudFormation. Anda juga dapat membuat aplikasi diApplication Managerdan memindahkan tumpukan ke dalamnya. Hal ini membantu Anda untuk menampilkan informasi operasi sumber daya di tumpukan dalam konteks aplikasi. Untuk informasi harga, lihat AWS CloudFormation Harga.

(Opsional) Atur dan deploy aplikasi Anda dengan menggunakan AWS Launch Wizard

Launch Wizard memandu Anda melewati proses pengukuran, konfigurasi, dan depoyment AWS sumber daya untuk aplikasi pihak ketiga tanpa perlu mengidentifikasi secara manual dan menyediakan individu AWS sumber daya.

Application Managersecara otomatis mengimpor semua sumber daya Wisaya Peluncuran Anda dan mencantumkannya diLuncurkan Wizardkategori. Untuk informasi selengkapnya tentang AWS Launch Wizard, lihat Memulai dengan AWS Launch Wizard untuk SQL Server. Launch Wizard tersedia tanpa biaya tambahan. Anda hanya membayar untuk AWS sumber daya yang Anda sediakan untuk menjalankan solusi.

(Opsional) Atur dan deploy aplikasi kontainer Anda dengan menggunakan Amazon ECS dan Amazon EKS

Amazon Elastic Container Service (Amazon ECS) adalah layanan manajemen kontainer yang sangat dapat diskalakan dan cepat sehingga memudahkan untuk menjalankan, menghentikan, dan mengelola kontainer di sebuah klaster. Kontainer Anda didefinisikan dalam definisi tugas yang Anda gunakan untuk menjalankan tugas individu atau tugas dalam layanan.

Amazon EKS adalah sebuah layanan terkelola yang membantu Anda menjalankan Kubernetes pada AWS tanpa perlu menginstal, mengoperasikan, dan mengelola bidang kontrol atau simpul Kubernetes Anda sendiri. Kubernetes adalah sistem sumber terbuka untuk mengotomatiskan deployment, penskalaan, dan pengelolaan aplikasi dalam kontainer.

Application Managersecara otomatis mengimpor semua sumber daya infrastruktur Amazon ECS dan Amazon EKS Anda dan mencantumkannya diCluster kontainertab. Namun, Anda tidak dapat mengelola atau melihat informasi operasi tentang pod atau kontainer Amazon EKS Anda diApplication Manager. Anda hanya dapat mengelola dan melihat informasi operasi tentang

infrastruktur yang meng-host sumber daya Amazon EKS Anda. Untuk informasi harga, lihat Harga Amazon ECS dan Harga Amazon EKS.

Atur tugas untuk melihat informasi operasi tentang sumber daya

Tugas penyiapan berikut membantu Anda melihat informasi operasi tentangAWSsumber daya diApplication Manager.

(Disarankan) Verifikasi Izin Runbook

Anda dapat memperbaiki masalah denganAWSsumber daya dariApplication Managerdengan menggunakan runbook Otomasi Manajer Sistem. Untuk menggunakan kemampuan perbaikan ini, Anda harus mengonfigurasi atau memverifikasi izin. Untuk informasi harga, lihat AWS Systems Manager Harga.

(Opsional) AktifkanPenjelajah Biaya

AWS Cost Exploreradalah fitur dariAWS Cost Managementyang dapat Anda gunakan untuk memvisualisasikan data biaya Anda untuk analisis lebih lanjut. Saat mengaktifkan Cost Explorer, Anda dapat melihat informasi biaya, riwayat biaya, dan pengoptimalan biaya untuk sumber daya aplikasi Anda diApplication Managerkonsol.

(Opsional) Siapkan dan konfigurasikan Amazon CloudWatch logdanalarm

CloudWatch adalah layanan pemantauan dan manajemen yang menyediakan data dan wawasan yang dapat ditindaklanjuti untukAWS, aplikasi hybrid, dan multicloud dan sumber daya infrastruktur. dengan CloudWatch, Anda dapat mengumpulkan dan mengakses semua data kinerja dan operasional Anda dalam bentuk log dan metrik dari satu platform. Untuk melihat CloudWatch log dan alarm untuk sumber daya Anda diApplication Manager, Anda harus mengatur dan mengkonfigurasi CloudWatch. Untuk informasi harga, lihat Harga CloudWatch.

> **Note**
>
> CloudWatch Dukungan log hanya berlaku untuk aplikasi, bukan untuk cluster.

(Opsional) Atur dan konfigurasikan AWS Config

AWS Config menyediakan tampilan mendetail tentang sumber daya yang terkait dengan Akun AWS, termasuk cara mengonfigurasi, bagaimana mereka terkait satu sama lain, dan bagaimana

konfigurasi dan hubungan mereka berubah seiring waktu. Anda dapat menggunakan AWS Config untuk mengevaluasi pengaturan konfigurasi AWS sumber daya. Anda melakukan ini dengan membuat AWS Config aturan, yang mewakili pengaturan konfigurasi ideal Anda. Sementara AWS Config terus melacak perubahan konfigurasi yang terjadi di antara sumber daya Anda, memeriksa apakah perubahan ini melanggar salah satu kondisi dalam aturan Anda. Jika sumber daya melanggar aturan,AWS Configmenandai sumber daya dan aturan sebagaitidak patuh.Application Managermenampilkan informasi kepatuhan tentangAWS Configaturan. Untuk melihat data ini diApplication Manager, Anda harus mengatur dan mengkonfigurasiAWS Config. Untuk informasi harga, lihat Harga AWS Config.

(Opsional) BuatState Manager asosiasi

Anda dapat menggunakan Manajer SistemState Manageruntuk membuat konfigurasi yang Anda tetapkan ke node terkelola Anda. Konfigurasi, yang disebutasosiasi, mendefinisikan status yang ingin Anda pertahankan di node Anda. Untuk melihat data kepatuhan asosiasi diApplication Manager, Anda harus mengkonfigurasi satu atau lebihState Managerasosiasi.State Managerditawarkan tanpa biaya tambahan.

(Opsional) Atur dan konfigurasikan OpsCenter

Anda dapat melihat item pekerjaan operasional (OpsItems) tentang sumber daya Anda diApplication Managerdengan menggunakanOpsCenter. Anda dapat mengonfigurasi Amazon CloudWatch dan Amazon EventBridge untuk mengirim secara otomatisOpsItemskepadaOpsCenterberdasarkan alarm dan peristiwa. Anda juga bisa masukOpsItemssecara manual. Untuk informasi harga, lihat Harga AWS Systems Manager.

Mengonfigurasi izin untuk Systems Manager Application Manager

Anda dapat menggunakan semua fiturApplication Manager, kemampuanAWS Systems Manager, jika entitas AWS Identity and Access Management (IAM) memiliki akses ke operasi API yang tercantum dalam topik ini. Operasi API dipisahkan menjadi dua tabel untuk membantu Anda memahami fungsi yang berbeda yang mereka lakukan.

Tabel berikut mencantumkan operasi API yang dipanggil Systems Manager jika Anda memilih sumber daya Application Manager karena ingin melihat rincian sumber daya. Sebagai contoh, jika Application Manager mendata grup Amazon EC2 Auto Scaling, dan jika Anda memilih grup untuk melihat rincian, maka Systems Manager akan memanggil operasi `autoscaling:DescribeAutoScalingGroups` API. Jika Anda tidak memiliki grup Auto Scaling di akun Anda, operasi API ini tidak dipanggil dariApplication Manager.

Hanya rincian sumber daya

```
acm:DescribeCertificate
acm:ListTagsForCertificate
autoscaling:DescribeAutoScalingGroups
cloudfront:GetDistribution
cloudfront:ListTagsForResource
cloudtrail:DescribeTrails
cloudtrail:ListTags
cloudtrail:LookupEvents
codebuild:BatchGetProjects
codepipeline:GetPipeline
codepipeline:ListTagsForResource
dynamodb:DescribeTable
dynamodb:ListTagsOfResource
ec2:DescribeAddresses
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
```

Hanya rincian sumber daya

```
rds:DescribeDBSubnetGroups
rds:DescribeEventSubscriptions
rds:ListTagsForResource
redshift:DescribeClusterParameters
redshift:DescribeClusterSecurityGroups
redshift:DescribeClusterSnapshots
redshift:DescribeClusterSubnetGroups
redshift:DescribeClusters
s3:GetBucketTagging
```

Tabel berikut mencantumkan operasi API yang digunakan Systems Manager untuk membuat perubahan pada aplikasi dan sumber daya yang tercantum di Application Manager atau untuk melihat informasi operasi untuk aplikasi atau sumber daya yang dipilih.

Tindakan dan detail aplikasi

```
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
```

Tindakan dan detail aplikasi

```
cloudwatch:ListTagsForResource
cloudwatch:PutMetricAlarm
config:DescribeComplianceByConfigRule
config:DescribeComplianceByResource
config:DescribeConfigRules
config:DescribeRemediationConfigurations
config:GetComplianceDetailsByConfigRule
config:GetComplianceDetailsByResource
config:GetResourceConfigHistory
config:ListDiscoveredResources
config:PutRemediationConfigurations
config:SelectResourceConfig
config:StartConfigRulesEvaluation
config:StartRemediationExecution
ec2:DescribeInstances
ecs:DescribeCapacityProviders
ecs:DescribeClusters
ecs:DescribeContainerInstances
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
```

Tindakan dan detail aplikasi

```
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:UpdateOpsItem
ssm:UpdateOpsMetadata
ssm:UpdateServiceSetting
tag:GetTagKeys
tag:GetTagValues
tag:TagResources
```

Tindakan dan detail aplikasi

```
tag:UntagResources
```

Mengonfigurasi izin

Untuk mengonfigurasi Application Manager izin untuk entitas IAM (seperti pengguna, grup, atau peran), buat kebijakan IAM menggunakan contoh berikut. Contoh kebijakan ini mencakup semua operasi API yang digunakan olehApplication Manager.

```
                {
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "acm:DescribeCertificate",
                "acm:ListTagsForCertificate",
                "applicationinsights:CreateApplication",
                "applicationinsights:DescribeApplication",
                "applicationinsights:ListProblems",
                "autoscaling:DescribeAutoScalingGroups",
                "ce:GetCostAndUsage",
                "ce:GetTags",
                "ce:ListCostAllocationTags",
                "ce:UpdateCostAllocationTagsStatus",
                "cloudformation:CreateStack",
                "cloudformation:DeleteStack",
                "cloudformation:DescribeStackDriftDetectionStatus",
                "cloudformation:DescribeStackEvents",
                "cloudformation:DescribeStacks",
                "cloudformation:DetectStackDrift",
                "cloudformation:GetTemplate",
                "cloudformation:GetTemplateSummary",
                "cloudformation:ListStacks",
                "cloudformation:ListStackResources",
                "cloudformation:UpdateStack",
                "cloudfront:GetDistribution",
                "cloudfront:ListTagsForResource",
                "cloudtrail:DescribeTrails",
                "cloudtrail:ListTags",
```

```
                "cloudtrail:LookupEvents",
                "cloudwatch:DescribeAlarms",
                "cloudwatch:DescribeInsightRules",
                "cloudwatch:DisableAlarmActions",
                "cloudwatch:EnableAlarmActions",
                "cloudwatch:GetMetricData",
                "cloudwatch:ListTagsForResource",
                "cloudwatch:PutMetricAlarm",
                "codebuild:BatchGetProjects",
                "codepipeline:GetPipeline",
                "codepipeline:ListTagsForResource",
                "config:DescribeComplianceByConfigRule",
                "config:DescribeComplianceByResource",
                "config:DescribeConfigRules",
                "config:DescribeRemediationConfigurations",
                "config:GetComplianceDetailsByConfigRule",
                "config:GetComplianceDetailsByResource",
                "config:GetResourceConfigHistory",
                "config:ListDiscoveredResources",
                "config:PutRemediationConfigurations",
                "config:SelectResourceConfig",
                "config:StartConfigRulesEvaluation",
                "config:StartRemediationExecution",
                "dynamodb:DescribeTable",
                "dynamodb:ListTagsOfResource",
                "ec2:DescribeAddresses",
                "ec2:DescribeCustomerGateways",
                "ec2:DescribeHosts",
                "ec2:DescribeInstances",
                "ec2:DescribeInternetGateways",
                "ec2:DescribeNetworkAcls",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVolumes",
                "ec2:DescribeVpcs",
                "ec2:DescribeVpnConnections",
                "ec2:DescribeVpnGateways",
                "ecs:DescribeCapacityProviders",
                "ecs:DescribeClusters",
                "ecs:DescribeContainerInstances",
                "ecs:ListClusters",
                "ecs:ListContainerInstances",
```

```
                "ecs:TagResource",
                "eks:DescribeCluster",
                "eks:DescribeFargateProfile",
                "eks:DescribeNodegroup",
                "eks:ListClusters",
                "eks:ListFargateProfiles",
                "eks:ListNodegroups",
                "eks:TagResource",
                "elasticbeanstalk:DescribeApplications",
                "elasticbeanstalk:ListTagsForResource",
                "elasticloadbalancing:DescribeInstanceHealth",
                "elasticloadbalancing:DescribeListeners",
                "elasticloadbalancing:DescribeLoadBalancers",
                "elasticloadbalancing:DescribeTags",
                "iam:CreateServiceLinkedRole",
                "iam:GetGroup",
                "iam:GetPolicy",
                "iam:GetRole",
                "iam:GetUser",
                "iam:ListRoles",
                "lambda:GetFunction",
                "logs:DescribeLogGroups",
                "rds:DescribeDBClusters",
                "rds:DescribeDBInstances",
                "rds:DescribeDBSecurityGroups",
                "rds:DescribeDBSnapshots",
                "rds:DescribeDBSubnetGroups",
                "rds:DescribeEventSubscriptions",
                "rds:ListTagsForResource",
                "redshift:DescribeClusterParameters",
                "redshift:DescribeClusters",
                "redshift:DescribeClusterSecurityGroups",
                "redshift:DescribeClusterSnapshots",
                "redshift:DescribeClusterSubnetGroups",
                "resource-groups:CreateGroup",
                "resource-groups:DeleteGroup",
                "resource-groups:GetGroup",
                "resource-groups:GetGroupQuery",
                "resource-groups:GetTags",
                "resource-groups:ListGroupResources",
                "resource-groups:ListGroups",
                "resource-groups:Tag",
                "resource-groups:Untag",
                "resource-groups:UpdateGroup",
```

```
                "s3:GetBucketTagging",
                "s3:ListAllMyBuckets",
                "s3:ListBucket",
                "s3:ListBucketVersions",
                "servicecatalog:GetApplication",
                "servicecatalog:ListApplications",
                "sns:CreateTopic",
                "sns:ListSubscriptionsByTopic",
                "sns:ListTopics",
                "sns:Subscribe",
                "ssm:AddTagsToResource",
                "ssm:CreateDocument",
                "ssm:CreateOpsMetadata",
                "ssm:DeleteDocument",
                "ssm:DeleteOpsMetadata",
                "ssm:DescribeAssociation",
                "ssm:DescribeAutomationExecutions",
                "ssm:DescribeDocument",
                "ssm:DescribeDocumentPermission",
                "ssm:GetDocument",
                "ssm:GetInventory",
                "ssm:GetOpsMetadata",
                "ssm:GetOpsSummary",
                "ssm:GetServiceSetting",
                "ssm:ListAssociations",
                "ssm:ListComplianceItems",
                "ssm:ListDocuments",
                "ssm:ListDocumentVersions",
                "ssm:ListOpsMetadata",
                "ssm:ListResourceComplianceSummaries",
                "ssm:ListTagsForResource",
                "ssm:ModifyDocumentPermission",
                "ssm:RemoveTagsFromResource",
                "ssm:StartAssociationsOnce",
                "ssm:StartAutomationExecution",
                "ssm:UpdateDocument",
                "ssm:UpdateDocumentDefaultVersion",
                "ssm:UpdateOpsMetadata",
                "ssm:UpdateOpsItem",
                "ssm:UpdateServiceSetting",
                "tag:GetResources",
                "tag:GetTagKeys",
                "tag:GetTagValues",
                "tag:TagResources",
```

```
                "tag:UntagResources"
            ],
            "Resource": "*"
        }
    ]
}
```

Note

Anda dapat membatasi kemampuan pengguna untuk membuat perubahan ke aplikasi dan sumber daya Application Manager dengan menghapus operasi API berikut dari kebijakan izin IAM yang melekat pada pengguna, grup, atau peran mereka. Menghapus tindakan ini menciptakan pengalaman baca saja di. Application Manager Berikut ini adalah semua API yang memungkinkan pengguna membuat perubahan pada aplikasi atau sumber daya terkait lainnya.

```
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
```

Untuk informasi selengkapnya tentang membuat dan mengubah kebijakan IAM, lihat Membuat Kebijakan IAM dalam Panduan Pengguna IAM. Untuk informasi tentang cara menetapkan kebijakan ini ke entitas IAM (seperti pengguna, grup, atau peran), lihat Menambahkan dan menghapus izin identitas IAM.

Menambahkan aplikasi dan klaster untukApplication Manager

Application Manageradalah komponenAWS Systems Manager. MasukApplication Manager, sebuahpenerapanadalah kelompok logisAWSsumber daya yang ingin Anda operasikan sebagai unit. Kelompok logis ini dapat mewakili aplikasi, batas kepemilikan untuk operator, atau lingkungan developer dengan versi yang berbeda, untuk beberapa nama.

Bila Anda memilihMemulaipadaApplication Managerhalaman rumah,Application Managersecara otomatis mengimpor metadata tentang sumber daya Anda yang dibuat di lainLayanan AWSatau kemampuan Systems Manager. Untuk aplikasi,Application Managermengimpor metadata tentang semuaAWSsumber daya yang diatur dalam kelompok sumber daya. Setiap grup sumber daya tercantum dalamAplikasi khususkategori sebagai aplikasi yang unik.Application Managerjuga secara otomatis mengimpor metadata tentang sumber daya yang dibuat olehAWS CloudFormation,AWS Launch Wizard, Amazon Elastic Container Service (Amazon ECS), dan Amazon Elastic Kubernetes Service (Amazon ECS), Amazon Elastic Container ServiceApplication Managerkemudian menampilkan sumber daya tersebut dalam kategori yang telah ditetapkan.

Untuk Aplikasi, daftar tersebut mencakup hal berikut:

- Aplikasi khusus
- Launch Wizard

- CloudFormation tumpukan
- AppRegistry aplikasi

Untuk Klaster kontainer, daftar ini mencakup hal berikut:

- Klaster Amazon ECS
- Klaster Amazon EKS

Setelah impor selesai, Anda dapat melihat informasi operasi untuk aplikasi atau sumber daya tertentu dalam kategori yang telah ditetapkan. Atau, jika Anda ingin memberikan lebih banyak konteks tentang koleksi sumber daya, Anda dapat secara manual membuat aplikasi diApplication Manager. Anda kemudian dapat menambahkan sumber daya atau grup sumber daya ke dalam aplikasi itu. Setelah Anda membuat aplikasi diApplication Manager, Anda dapat melihat informasi operasi tentang sumber daya Anda dalam konteks aplikasi.

Membuat aplikasi dalamApplication Manager

Gunakan prosedur berikut untuk membuat aplikasi diApplication Managerdan menambahkan sumber daya ke aplikasi itu.

Untuk membuat aplikasi diApplication Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Pilih tab Aplikasi, dan kemudian pilih Buat aplikasi.
4. Untuk Nama aplikasi, masukkan nama untuk membantu Anda memahami tujuan sumber daya yang akan ditambahkan ke aplikasi ini.
5. Untuk Deskripsi aplikasi, masukkan informasi tentang aplikasi ini.
6. Di bagian Pilih komponen aplikasi, gunakan opsi yang disediakan untuk memilih sumber daya untuk aplikasi ini. Anda dapat menambahkan kombinasi dari tag sumber daya, grup sumber daya, dan tumpukan untuk aplikasi. Anda harus memilih minimal dua komponen dan maksimal 15. Jika Anda memilih sumber daya dengan menggunakan tag, maka semua sumber daya yang ditetapkan tag tersebut akan tercantum pada tab Sumber Daya setelah Anda menambahkan aplikasi baru. Hal ini juga berlaku untuk sumber daya yang disertakan dalam grup sumber daya atau dalam tumpukan.

Jika Anda tidak melihat sumber daya yang ingin ditambahkan ke aplikasi, verifikasi sumber daya telah ditandai dengan benar, ditambahkan ke AWS Resource Groups grup, atau ditambahkan ke AWS CloudFormation tumpukan.

7. Untuk Tag aplikasi - opsional, tentukan tag untuk aplikasi ini.
8. Pilih Buat.

Application Managermembuat aplikasi dan membukanya. Pohon Komponen mencantumkan aplikasi baru sebagai komponen tingkat atas dan sumber daya, grup, atau tumpukan yang Anda pilih sebagai subkomponen. Lain kali Anda membukaApplication Manager, Anda dapat menemukan aplikasi baru diAplikasi khususkategori.

Bekerja dengan Application Manager

Application Manageradalah komponen dariAWS Systems Manager. Bagian ini mencakup topik untuk membantu Anda bekerja dengan Application Manager aplikasi dan melihat informasi operasi tentang AWS sumber daya.

Daftar Isi

Bekerja dengan aplikasi

Application Manageradalah komponen dariAWS Systems Manager. Bagian ini mencakup topik untuk membantu Anda bekerja dengan Application Manager aplikasi dan melihat informasi operasi tentang AWS sumber daya.

Daftar Isi

- Melihat OpsItemsuntuk aplikasi
- Menampilkan grup log dan data log
- Bekerja dengan runbook diApplication Manager
- Cara menggunakan tag di Application Manager

Melihat informasi gambaran umum tentang aplikasi

DiApplication Manager, tab Gambaran umum menampilkan ringkasan CloudWatch alarm Amazon, item pekerjaan operasional (OpsItems), Wawasan CloudWatch Aplikasi, dan riwayat runbook. AWS Systems Manager Pilih Lihat semua untuk setiap kartu untuk membuka tab yang sesuai di mana Anda dapat melihat semua wawasan aplikasi, alarmOpsItems, atau riwayat runbook.

Tentang Wawasan Aplikasi

CloudWatchApplication Insights mengidentifikasi dan menyiapkan metrik kunci, log, dan alarm di seluruh sumber daya aplikasi dan tumpukan teknologi. Wawasan Aplikasi secara terus menerus memantau metrik dan log untuk mendeteksi dan menghubungkan anomali dan kesalahan. Ketika sistem mendeteksi kesalahan atau anomali, Wawasan Aplikasi menghasilkan CloudWatch Peristiwa yang dapat Anda gunakan untuk mengatur notifikasi atau mengambil tindakan. Jika Anda memilih tombol Edit konfigurasi pada tab Monitoring, sistem akan membuka konsol CloudWatch Application Insights. Untuk informasi selengkapnya tentang Wawasan Aplikasi, lihat Apa itu Amazon CloudWatch Application Insights di CloudWatchPanduan Pengguna Amazon.

Tentang Cost Explorer

Application Managerterintegrasi denganAWS Cost Explorer, fitur Manajemen AWS Biaya, melalui widget Biaya dan tab Biaya. Setelah Anda mengaktifkan Cost Explorer di konsol Manajemen Biaya, widget Biaya dan tab Biaya Application Manager menunjukkan data biaya untuk aplikasi non-kontainer atau komponen aplikasi tertentu. Anda dapat menggunakan filter di widget, atau tab, untuk melihat data biaya sesuai dengan periode waktu yang berbeda, tingkat perincian, dan jenis biaya baik dalam grafik batang atau garis.

Anda dapat mengaktifkan fitur ini dengan memilih tombol Konsol Go to AWS Cost Management. Secara default, data disaring hingga tiga bulan terakhir. Untuk aplikasi non-kontainer, jika Anda memilih tombol Lihat semua, Application Manager buka tab Sumber Daya. Untuk aplikasi kontainer, tombol Lihat semua membuka AWS Cost Explorer konsol.

Tindakan yang dapat Anda lakukan di halaman ini

Anda dapat mengaktifkan dan mengakses informasi tentang widget berikut di tab Ikhtisar di halaman ini. Saat widget diaktifkan, pilih View all untuk melihat detail aplikasi yang relevan untuk area tersebut.

- Di bagian Wawasan dan Alarm, pilih nomor keparahan untuk membuka tab Pemantauan, di mana Anda dapat melihat rincian lebih lanjut tentang alarm keparahan yang dipilih.
- Di bagian Biaya, pilih Lihat semua untuk membuka tab Sumber Daya, di mana Anda dapat melihat data biaya untuk aplikasi atau komponen aplikasi tertentu.
- Di bagian Kepatuhan, pilih Lihat semua untuk membuka tab Kepatuhan, tempat Anda dapat melihat informasi kepatuhan dari AWS Config dan State Manager asosiasi.

> **Note**
>
> Untuk melihat detail kepatuhan patch, pilih tab Kepatuhan secara langsung. Kemudian Anda dapat melihat detail kepatuhan patch untuk node terkelola yang digunakan oleh aplikasi yang dipilih.

- Di bagian Runbook, pilih runbook untuk membukanya di halaman Dokumen Systems Manager agar Anda dapat melihat rincian lebih lanjut tentang dokumen.
- Di OpsItemsbagian, pilih tingkat keparahan untuk membuka OpsItemstab di mana Anda dapat melihat semua OpsItems keparahan yang dipilih.
- Pilih tombol Lihat semua untuk membuka tab yang sesuai. Anda dapat melihat semua alarmOpsItems, atau entri riwayat runbook untuk aplikasi.

Untuk membuka tab Ikhtisar

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih kategori. Jika anda ingin membuka aplikasi yang anda buat secara manualApplication Manager, pilih Aplikasi khusus.
4. Pilih aplikasi dalam daftar. Application Managermembuka tab Ikhtisar.

Bekerja dengan instans aplikasi Anda

Application Managerterintegrasi dengan Amazon Elastic Compute Cloud (Amazon EC2) untuk menampilkan informasi tentang instans Anda dalam konteks aplikasi. Application Managermenampilkan status instans, status, dan kesehatan Auto Scaling Amazon EC2 untuk aplikasi

yang dipilih dalam format grafis. Tab Instances juga menyertakan tabel dengan informasi berikut untuk setiap instance dalam aplikasi Anda:

- Status instans (Tertunda, Berhenti, Berlari, Berhenti)
- Status ping untuk SSM Agent
- Status dan nama runbook Systems Manager Automation terbaru yang diproses pada instans
- Hitungan alarm Amazon CloudWatch Logs per negara bagian.
  - `ALARM` – Metrik atau ekspresi berada di luar ambang batas yang telah ditetapkan sebelumnya.
  - `OK` – Metrik atau ekspresi berada dalam ambang batas yang telah ditetapkan sebelumnya.
  - `INSUFFICIENT_DATA` – Alarm baru saja dimulai, metrik tidak tersedia, atau tidak ada data yang memadai yang tersedia bagi metrik untuk menentukan status alarm.
- Kesehatan grup Auto Scaling untuk grup penskalaan otomatis induk dan individu

Jika Anda memilih instance di tabel Semua instance, Application Manager menampilkan informasi tentang instance tersebut pada empat tab:

- Detail — Semua detail instans dari Amazon EC2, termasuk Amazon Machine Image (AMI), informasi DNS, informasi alamat IP, dan banyak lagi.
- Kesehatan — Status saat ini seperti yang disediakan oleh sistem EC2 dan pemeriksaan status instans.
- Riwayat eksekusi — Log eksekusi untuk runbook Systems Manager Automation dan panggilan API yang diproses oleh instance.
- CloudWatch alarm — Nama, status, dan lainnya untuk setiap CloudWatch alarm yang diangkat oleh instance.

Tindakan yang dapat Anda lakukan di halaman ini

Anda dapat melakukan salah satu tindakan berikut di halaman ini:

- Mulai, hentikan, dan akhiri instance.
- Oleskan Chef resep.
- Lampirkan instance ke, atau lepaskan instance dari, grup Auto Scaling.
- Aktifkan pembaruan otomatis untukSSM Agent.

Untuk membuka tab Instances

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih kategori. Jika Anda ingin membuka aplikasi yang Anda buat secara manualApplication Manager, pilih Aplikasi khusus.
4. Pilih aplikasi dalam daftar. Application Managermembuka tab Ikhtisar.
5. Pilih tab Instances.

Untuk melihat detail untuk instance aplikasi Anda

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih kategori. Jika Anda ingin membuka aplikasi yang Anda buat secara manualApplication Manager, pilih Aplikasi khusus.
4. Pilih aplikasi dalam daftar. Application Managermembuka tab Ikhtisar.
5. Pilih tab Instances.
6. Pilih tombol di sebelah instance yang detailnya ingin Anda lihat.
7. Tinjau detail instance di bagian bawah halaman.

Untuk memperbarui secara otomatis SSM Agent

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih kategori. Jika Anda ingin membuka aplikasi yang Anda buat secara manualApplication Manager, pilih Aplikasi khusus.
4. Pilih aplikasi dalam daftar. Application Managermembuka tab Ikhtisar.
5. Pilih tab Instances.
6. Di dropdown Tindakan agen, pilih Konfigurasi SSM Agent pembaruan.
7. Pilih Semua instance untuk mengonfigurasi SSM Agent pembaruan otomatis untuk semua instans terkelola. Atau, pilih Instance untuk mengonfigurasi SSM Agent pembaruan otomatisasi untuk satu instance dalam aplikasi Anda.

8. Pilih sakelar Aktifkan pembaruan otomatis.
9. Dalam menu tarik-turun Tentukan jadwal, pilih jadwal yang ingin Anda gunakan untuk SSM Agent pembaruan.
10. Pilih Konfigurasi.

Menampilkan sumber daya aplikasi

DalamApplication Manager, komponenAWS Systems Manager, tab Sumber Daya menampilkanAWS sumber daya di aplikasi Anda. Jika Anda memilih komponen tingkat atas, halaman ini menampilkan semua sumber daya untuk komponen tersebut dan subkomponen apa pun. Jika Anda memilih subkomponen, halaman ini hanya menampilkan sumber daya yang ditetapkan ke subkomponen tersebut.

Tindakan yang dapat Anda lakukan di halaman ini

Anda dapat melakukan salah satu tindakan berikut di halaman ini:

- Pilih nama sumber daya untuk melihat informasi tentang hal itu, termasuk rincian yang disediakan oleh konsol tempat ia dibuat, tag, CloudWatch alarm Amazon,AWS Config detail, dan informasiAWS CloudTrail log.
- Pilih tombol pilihan di samping nama sumber daya. Kemudian, pilih tombol Garis waktu sumber daya untuk membuka AWS Config konsol agar Anda dapat melihat informasi kepatuhan tentang sumber daya yang dipilih.
- Jika Anda mengaktifkanAWS Cost Explorer, bagian Cost Explorer menampilkan data biaya untuk aplikasi non-kontainer atau komponen aplikasi tertentu. Anda dapat mengaktifkan fitur ini dengan memilih tombol Konsol Go toAWS Cost Management. Gunakan filter di bagian ini untuk melihat informasi biaya tentang aplikasi Anda.

Untuk membuka tab Resources

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih kategori. Jika anda ingin membuka aplikasi yang anda buat secara manualApplication Manager, pilih Aplikasi khusus.
4. Pilih aplikasi dalam daftar. Application Managermembuka tab Ikhtisar.
5. Pilih tab Sumber Daya.

Menampilkan informasi kepatuhan

Dalam komponenApplication Manager, halaman Konfigurasi menampilka informasi kepatuhan aturan AWS Configsumber daya dan konfigurasi.AWS Systems Manager Halaman ini juga menampilkan informasi kepatuhanAWS Systems Manager State Managerasosiasi. Anda dapat memilih sumber daya, aturan, atau asosiasi untuk membuka konsol yang sesuai untuk informasi lebih lanjut. Halaman ini menampilkan informasi kepatuhan dari 90 hari terakhir.

Tindakan yang dapat Anda lakukan di halaman ini

Anda dapat melakukan salah satu tindakan berikut di halaman ini:

- Pilih nama sumber daya untuk membuka AWS Config konsol di mana Anda dapat melihat informasi kepatuhan tentang sumber daya yang dipilih.
- Pilih tombol pilihan di samping nama sumber daya. Kemudian, pilih tombol Garis waktu sumber daya untuk membuka AWS Config konsol agar Anda dapat melihat informasi kepatuhan tentang sumber daya yang dipilih.
- Di bagian Kepatuhan aturan Config, Anda dapat melakukan hal berikut:
  - Pilih nama aturan untuk membuka AWS Config konsol agar Anda dapat melihat informasi tentang aturan tersebut.
  - Pilih Tambahkan aturan untuk membuka AWS Config konsol agar Anda dapat membuat aturan.
  - Pilih tombol opsi di samping nama aturan, pilih Tindakan, lalu pilih Kelola remediasi untuk mengubah tindakan perbaikan untuk aturan.
  - Pilih tombol opsi di samping nama aturan, pilih Tindakan, lalu pilih Evaluasi ulang untuk bisa AWS Config menjalankan pemeriksaan kepatuhan pada aturan yang dipilih.
- Di bagian Kepatuhan asosiasi, Anda dapat melakukan hal berikut:
  - Pilih nama asosiasi untuk membuka halaman Asosiasi agar Anda dapat melihat informasi tentang asosiasi tersebut.
  - Pilih Buat asosiasi untuk membuka Systems ManagerState Manager tempat Anda dapat membuat asosiasi.
  - Pilih tombol opsi di samping nama asosiasi dan pilih Terapkan asosiasi untuk segera memulai semua tindakan yang ditentukan dalam asosiasi.

Untuk membuka tab Kepatuhan

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.

2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih kategori. Jika Anda ingin membuka aplikasi yang Anda buat secara manualApplication Manager, pilih Aplikasi khusus.
4. Pilih aplikasi dalam daftar. Application Managermembuka tab Ikhtisar.
5. Pilih tab Kepatuhan.

Menampilkan informasi pemantauan

MasukApplication Manager, komponen dariAWS Systems Manager, yangPemantauantab menampilkan Amazon CloudWatch Wawasan Aplikasi dan detail alarm untuk sumber daya dalam aplikasi.

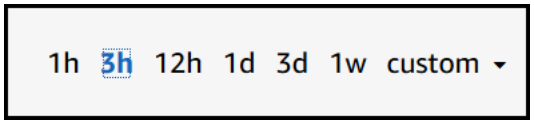
Tentang Application Insights

CloudWatch Application Insights mengidentifikasi dan menyiapkan metrik kunci, log, dan alarm di seluruh sumber daya aplikasi dan tumpukan teknologi Anda. Application Insights secara terus menerus memantau metrik dan log untuk mendeteksi dan menghubungkan anomali dan kesalahan. Ketika sistem mendeteksi kesalahan atau anomali, Application Insights menghasilkan CloudWatch Acara yang dapat Anda gunakan untuk mengatur notifikasi atau mengambil tindakan. Jika Anda memilihEdit konfigurasitombol padaPemantauantab, sistem membuka CloudWatch Konsol Application Insights. Untuk informasi selengkapnya tentang Application InsightsApa itu Amazon CloudWatch Wawasan Aplikasidi dalamAmazon CloudWatch Panduan Pengguna.

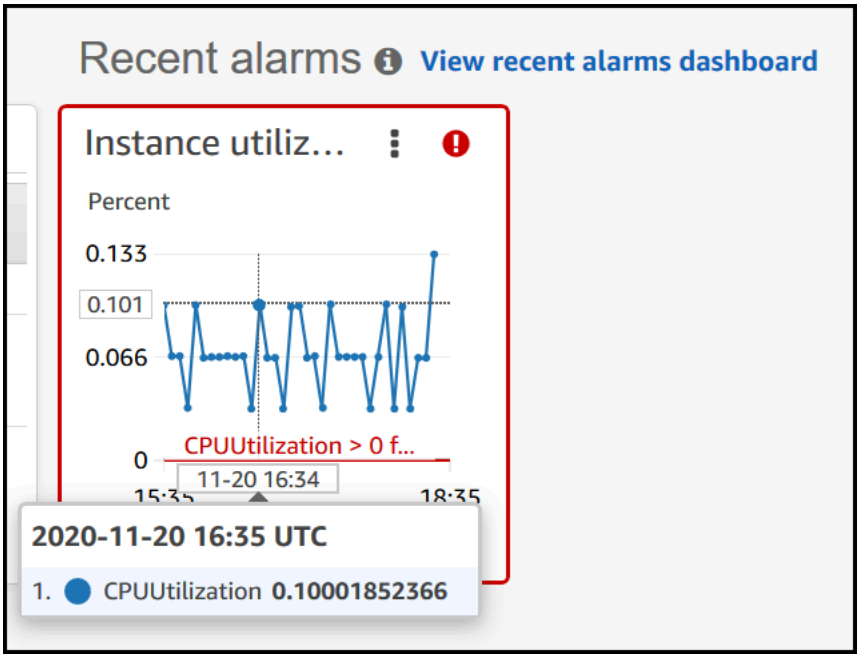
Tindakan yang dapat Anda lakukan di halaman ini

Anda dapat melakukan salah satu tindakan berikut di halaman ini:

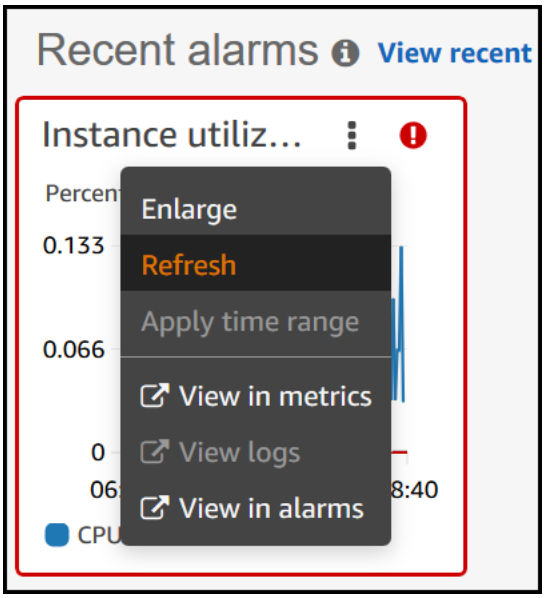
- Pilih nama layanan diAlarm olehAWSlayananbagian untuk membuka CloudWatch ke layanan dan alarm yang dipilih.
- Sesuaikan periode waktu untuk data yang ditampilkan dalam widget di bagian Alarm terbaru dengan memilih salah satu nilai periode waktu yang telah ditetapkan. Anda dapat memilih khusus untuk menentukan periode waktu Anda sendiri.

- Arahkan kursor ke widget di bagian Alarm terbaru untuk melihat pop-up data untuk waktu tertentu.



- Pilih menu opsi di widget untuk menampilkan opsi tampilan. Pilih Perbesar untuk memperluas widget. Pilih Refresh untuk memperbarui data dalam widget. Klik dan seret kursor Anda dalam tampilan data widget untuk memilih rentang tertentu. Anda kemudian dapat memilih Terapkan rentang waktu.

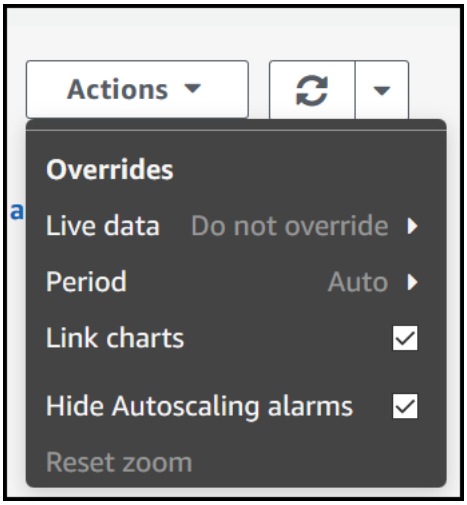


- Pilih menu Tindakan untuk melihat opsi Override data alarm, yang mencakup hal berikut:
  - Pilih apakah widget Anda menampilkan data langsung. Data langsung adalah data yang dipublikasikan dalam menit terakhir yang belum dikumpulkan sepenuhnya. Jika data langsung ditutup, hanya titik data dengan periode gabungan setidaknya satu menit di masa lalu yang

ditampilkan. Misalnya, ketika menggunakan periode 5 menit, titik data untuk 12:35 akan digabungkan dari 12:35 hingga 12:40, dan ditampilkan pada 12:41.

Jika data langsung dibuka, titik data terbaru ditampilkan segera setelah data apa pun dipublikasikan dalam rentang penggabungan terkait. Setiap kali Anda menyegarkan tampilan, titik data terbaru dapat berubah ketika data baru dalam periode pengumpulan tersebut dipublikasikan.

- Tentukan periode waktu untuk data langsung.
- Tautkan bagan di Alarm terbaru, sehingga ketika Anda memperbesar atau memperkecil satu grafik, grafik lainnya akan memperbesar atau memperkecil dalam waktu bersamaan. Anda dapat membatalkan tautan grafik untuk membatasi perbesaran ke satu grafik.
- Sembunyikan alarm Auto Scaling.



Untuk membukaPemantauantab

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih kategori. Jika Anda ingin membuka aplikasi yang Anda buat secara manualApplication Manager, memilihAplikasi khusus.
4. Pilih aplikasi dalam daftar.Application ManagermembukaIkhtisarTab.
5. Pilih tab Pemantauan.

Melihat OpsItemsuntuk aplikasi

MasukApplication Manager, komponen dariAWS Systems Manager, yangOpsItemstab menampilkan item pekerjaan operasional (OpsItems) untuk sumber daya dalam aplikasi yang dipilih. Anda dapat mengkonfigurasi Systems ManagerOpsCenteruntuk membuat secara otomatisOpsItemsdari Amazon CloudWatch alarm dan Amazon EventBridge peristiwa. Anda juga dapat membuat secara manualOpsItems.

Tindakan yang dapat Anda lakukan di tab ini

Anda dapat melakukan salah satu tindakan berikut di halaman ini:

- Saring daftarOpsItemsdengan menggunakan kolom pencarian. Anda dapat memfilterOpsItemnama, ID, sumber ID, atau tingkat keparahan. Anda juga dapat memfilter daftar berdasarkan status.OpsItemsmendukung status berikut: Buka, Sedang berlangsung, Open dan Sedang berlangsung, Resolved, atau All.
- Mengubah statusOpsItemdengan memilih tombol opsi di Tetapkannya dan kemudian memilih di Tetingkap di Tetingkap dan kemudian memilih di TetapTetapkan statusMenu.
- Systems Manager TerbukaOpsCenteruntuk membuatOpsItemdengan memilihBuatOpsItem.

Untuk membuka OpsItems tab

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih kategori. Jika anda ingin membuka aplikasi yang anda buat secara manualApplication Manager, pilihAplikasi khusus.
4. Pilih aplikasi dalam daftar.Application ManagermembukaIkhtisaraTab.
5. PilihOpsItemsaTab.

Menampilkan grup log dan data log

MasukApplication Manager, komponenAWS Systems Manager, yangBeberapa catatanTab menampilkan daftar grup log dari Amazon CloudWatch Log.

Tindakan yang dapat Anda lakukan di tab ini

Anda dapat melakukan salah satu tindakan berikut di halaman ini:

- Pilih nama grup log untuk membukanya CloudWatch Log. Anda kemudian dapat memilih aliran log untuk melihat log untuk sumber daya dalam konteks aplikasi.
- PilihMembuat grup loguntuk membuat grup log di CloudWatch Log.

Untuk membukaBeberapa catatantab

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih kategori. Jika anda ingin membuka aplikasi yang anda buat secara manualApplication Manager, pilihAplikasi khusus.
4. Pilih aplikasi dalam daftar.Application ManagermembukaIkhtisarTab.
5. Pilih tab Log.

Bekerja dengan runbook diApplication Manager

Anda dapat mengatasi masalahAWS sumber daya dari kemampuanApplication ManagerAWS Systems Manager, dengan menggunakan runbook otomatisasi. Sebuah runbook otomatisasi mendefinisikan tindakan yang dilakukan Systems Manager pada instans terkelola danAWS sumber daya lain ketika otomatisasi berjalan. Otomatisasi adalah kemampuan AWS Systems Manager. Runbook berisi satu langkah atau lebih yang berjalan dalam urutan yang tepat. Setiap langkah dibangun di sekitar satu tindakan. Output satu langkah dapat digunakan sebagai masukan dalam langkah selanjutnya.

Bila Anda memilih Mulai runbook dariApplication Manager aplikasi runbook dari aplikasi runbook yang difilter berdasarkan jenis sumber daya dalam aplikasi atau klaster. Ketika Anda memilih runbook yang ingin Anda mulai, Systems Manager akan membuka halaman Eksekusi dokumen otomatisasi.

Application Managertermasuk perangkat tambahan berikut untuk bekerja dengan runbook.

- Jika Anda memilih nama sumber daya di dalamnyaApplication Manager dan memilih Eksekusi runbook, sistem menampilkan daftar runbook yang difilter untuk jenis sumber daya tersebut.
- Anda dapat memulai otomatisasi pada semua sumber daya dari jenis yang sama dengan memilih runbook dalam daftar dan kemudian memilih Jalankan untuk sumber daya dari jenis yang sama.

Sebelum Anda memulai

Sebelum Anda memulai runbook dariApplication Manager, lakukan hal berikut:

- Verifikasi bahwa Anda memiliki izin yang benar untuk memulai runbook. Untuk informasi selengkapnya, lihat Menyiapkan Otomatisasi.
- Tinjau dokumentasi prosedur Otomatisasi tentang memulai runbook. Untuk informasi selengkapnya, lihat Menjalankan Otomatisasi.

Untuk memulai runbook dariApplication Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih kategori. Jika anda ingin membuka aplikasi yang anda buat secara manual dalamApplication Manager, pilih Aplikasi khusus.
4. Pilih aplikasi dalam daftar. Application Managermembuka tab Ikhtisar.
5. Pilih Mulai runbook. Application Managermembuka widget Automation muncul. Untuk informasi tentang opsi di widget otomatisasi, lihatMenjalankan Otomatisasi.

Cara menggunakan tag di Application Manager

Anda dapat dengan cepat menambahkan atau menghapus tag pada aplikasi danAWSsumber daya dalamApplication Manager. Untuk informasi selengkapnya tentang tanda, lihat Penandaan sumber daya Systems Manager.

Gunakan prosedur berikut untuk menambahkan tag ke atau menghapus tag dari aplikasi dan semuaAWSsumber daya dalam aplikasi itu.

Untuk menambahkan tag ke atau menghapus tag dari aplikasi dan semua sumber daya dalam aplikasi

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih kategori. Jika anda ingin membuka aplikasi yang anda buat secara manual dalamApplication Manager, pilihAplikasi khusus.
4. Pilih aplikasi dalam daftar.Application ManagermembukaIkhtisartab.
5. DiInformasi aplikasibagian, pilih nomor di bawahTag aplikasi. Jika tidak ada tag yang ditugaskan ke aplikasi, jumlahnya nol.
6. Untuk menambahkan tanda, pilih Tambahkan tanda baru. Tentukan kunci dan nilai opsional. Untuk menghapus sebuah tanda, pilih Hapus.

7. Pilih Save (Simpan).

Gunakan prosedur berikut untuk menambahkan tag ke atau menghapus tag dari sumber daya tertentu diApplication Manager.

Untuk menambahkan tag ke atau menghapus tag dari sumber daya

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih kategori. Jika anda ingin membuka aplikasi yang anda buat secara manual dalamApplication Manager, pilihAplikasi khusus.
4. Pilih aplikasi dalam daftar.Application ManagermembukaIkhtisartab.
5. Pilih tab Sumber Daya.
6. Pilih nama sumber daya.
7. DiTandabagian memilihedit.
8. Untuk menambahkan tanda, pilih Tambahkan tanda baru. Tentukan kunci dan nilai opsional. Untuk menghapus sebuah tanda, pilih Hapus.
9. Pilih Simpan.

Bekerja denganAWS CloudFormationtemplate dan tumpukan diApplication Manager

Application ManagerSebuah kemampuan dariAWS Systems Manager, membantu Anda menyediakan dan mengelola sumber daya untuk aplikasi Anda dengan mengintegrasikanAWS CloudFormation. Anda dapat membuat, mengedit, dan menghapusAWS CloudFormationtemplate dan tumpukan diApplication Manager. SEBUAHtumpukanadalah koleksiAWSsumber daya yang dapat Anda kelola sebagai satu unit. Ini berarti Anda dapat membuat, memperbarui, atau menghapus koleksiAWSsumber daya dengan menggunakan CloudFormationtumpukan. SEBUAHtemplatadalah file teks yang diformat di JSON atau YAMAL yang menentukan sumber daya yang ingin Anda sediakan di tumpukan Anda.

Application Managerjuga mencakup pustaka templat tempat Anda dapat mengkloning, membuat, dan menyimpan templat.Application Managerdan CloudFormation menampilkan informasi yang sama tentang status tumpukan saat ini. Template dan pembaruan template disimpan di Manajer Sistem sampai Anda menyediakan tumpukan, pada saat itu perubahan juga ditampilkan di CloudFormation.

Setelah Anda membuat tumpukan diApplication Manager,CloudFormation tumpukanhalaman menampilkan informasi bermanfaat tentang hal itu. Ini termasuk template yang digunakan untuk membuatnya, hitunganOpsItemsuntuk sumber daya di tumpukan Anda,status tumpukan, danstatus drift.

Tentang Cost Explorer

Application Managerterintegrasi denganAWS Cost Explorer, fitur dariAWSManajemen Biaya, melaluiBiayawidget. Setelah Anda mengaktifkan Cost Explorer di konsol Manajemen Biaya,Biayawidget diApplication Managermenunjukkan data biaya untuk aplikasi non-kontainer atau komponen aplikasi tertentu. Anda dapat menggunakan filter di widget untuk melihat data biaya sesuai dengan periode waktu, perincian, dan jenis biaya yang berbeda baik dalam bagan batang atau garis.

Anda dapat mengaktifkan fitur ini dengan memilihPergi keAWSKonsol Manajemen Biayatombol. Secara default, data disaring hingga tiga bulan terakhir. Untuk aplikasi non-kontainer, jika Anda memilihLihat semuatombol,Application ManagermembukaSumber Dayatab. Untuk aplikasi kontainer,Lihat semuatombol membukaAWS Cost Explorerkonsol.

> **Note**
>
> Cost Explorer menggunakan tag untuk melacak biaya aplikasi Anda. Jika AndaAWS CloudFormationaplikasi berbasis tumpukan tidak dikonfigurasi denganAppManagerCFNStackKeykunci tag, Cost Explorer gagal menyajikan data biaya yang akurat diApplication Manager. KetikaAppManagerCFNStackKeykunci tag tidak terdeteksi, Anda akan diminta di konsol untuk menambahkan tag ke CloudFormation tumpukan untuk mengaktifkan pelacakan biaya. Menambahkannya memetakan kunci tag ke Amazon Resource Name (ARN) tumpukan Anda dan mengaktifkanBiayawidget untuk menampilkan data biaya yang akurat.

> **Important**
>
> MenambahkanAppManagerCFNStackKeytag akan memicu pembaruan tumpukan. Konfigurasi manual apa pun yang dilakukan setelah tumpukan awalnya digunakan tidak akan tercermin setelah tag pengguna ditambahkan. Untuk informasi selengkapnya tentang perilaku pembaruan sumber daya, lihatPerbarui perilaku sumber daya tumpukandiAWS CloudFormationPanduan Pengguna

Sebelum Anda memulai

Gunakan tautan berikut untuk mempelajari CloudFormation konsep sebelum Anda membuat, mengedit, atau menghapus CloudFormation template dan tumpukan dengan menggunakanApplication Manager.

- Apa itu AWS CloudFormation?
- AWS CloudFormationpraktik terbaik
- Pelajari dasar-dasar template
- Bekerja denganAWS CloudFormationtumpukan
- Bekerja denganAWS CloudFormationtemplat
- Contoh template

Topik

- Bekerja dengan templat CloudFormation
- bekerja denganCloudFormation tumpukan

Bekerja dengan templat CloudFormation

Application Manager, kemampuanAWS Systems Manager, termasuk perpustakaan template dan alat lain untuk membantu Anda mengelolaAWS CloudFormation template. Bagian ini mencakup informasi berikut.

Topik

- Bekerja dengan pustaka template
- Membuat templat
- Mengedit templat

Bekerja dengan pustaka template

PustakaApplication Manager template menyediakan alat untuk membantu Anda melihat, membuat, mengedit, menghapus, dan mengkloning template. Anda juga dapat menyediakan tumpukan langsung dari pustaka template. templat disimpan sebagai dokumen Systems Manager (SSM) jenis dokumen Systems Manager (SSM)`CloudFormation`. Dengan menyimpan template sebagai dokumen SSM, Anda dapat menggunakan kontrol versi untuk bekerja dengan versi template

yang berbeda. Anda juga dapat mengatur templat dan berbagi templat. Setelah Anda berhasil menyediakan stack, stack dan template tersedia diApplication Manager danCloudFormation.

Sebelum Anda memulai

Kami menyarankan Anda membaca topik berikut untuk mempelajari lebih lanjut tentang dokumen SSM sebelum Anda mulai bekerja denganCloudFormation template diApplication Manager.

- AWS Systems ManagerDokumen
- Membagikan dokumen SSM
- Praktik terbaik untuk dokumen SSM bersama

Untuk melihat pustakaApplication Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih CloudFormationtumpukan.
4. Pilih Pustaka template.

Membuat templat

Prosedur berikut menjelaskan cara membuatCloudFormation templatApplication Manager Saat Anda membuat template, Anda memasukkan detail tumpukan template di JSON atau YAKL. Jika Anda tidak terbiasa dengan JSON atau YAL, Anda dapat menggunakanAWS CloudFormation Designer, alat untuk membuat dan memodifikasi templat secara visual. Untuk informasi selengkapnya, lihat Apa yang dimaksud dengan AWS CloudFormation Designer? dalam Panduan Pengguna AWS CloudFormation. Untuk informasi selengkapnya tentang struktur dan sintaks dari templat, lihat Anatomi templat.

Anda juga dapat membuat template dari beberapa cuplikan template. Cuplikan templat. adalah contoh yang menunjukkan cara menulis templat untuk sumber daya tertentu. Misalnya, Anda dapat melihat cuplikan untuk Instans Amazon Elastic Compute Cloud (Amazon EC2), domain Amazon Simple Storage Service (Amazon S3),AWS CloudFormation pemetaan, dan banyak lagi. Cuplikan dikelompokkan berdasarkan sumber daya. Anda dapat menemukanAWS CloudFormation cuplikan tujuan umum di bagian cuplikan template Umum di PanduanAWS CloudFormation Pengguna.

MembuatCloudFormation template diApplication Manager (konsol)

Gunakan prosedur berikut untuk membuatCloudFormation templateApplication Manager dengan menggunakanAWS Management Console.

Untuk membuatCloudFormation template diApplication Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih CloudFormationtumpukan.
4. Pilih Template library, dan kemudian pilih Create template atau pilih template yang ada dan kemudian pilih Actions, Clone.
5. Untuk Nama, masukkan nama untuk template yang membantu Anda mengidentifikasi sumber daya yang dibuatnya atau tujuan tumpukan.
6. (Opsional) Untuk nama Versi, masukkan nama atau nomor untuk identitas versi template.
7. (Opsional) Untuk Deskripsi, masukkan informasi tentang template ini.
8. Di bagian Editor kode, pilih YAKL atau JSON dan kemudian masukkan atau salin dan tempel kode template Anda.
9. (Opsional) Dalam bagian Tag, terapkan satu pasangan nilai kunci tag atau lebih ke templat.

   Tag adalah metadata opsional yang Anda tetapkan ke sumber daya. Dengan menggunakan tag, Anda dapat mengategorikan sumber daya dengan cara yang berbeda, seperti berdasarkan tujuan, pemilik, atau lingkungan. Untuk informasi selengkapnya tentang penandaan sumber daya Systems Manager, lihat Penandaan sumber daya Systems Manager.
10. (Opsional) Di bagian Izin, masukkanAkun AWS ID dan pilih Tambahkan akun. Tindakan ini memberikan izin baca untuk template. Pemilik akun dapat menyediakan dan mengkloning template, tetapi mereka tidak dapat mengedit atau menghapusnya.
11. Pilih Create (Buat). templat disimpan dalam layanan Systems Manager (SSM) dokumen Systems Manager (SSM).

MembuatCloudFormation template diApplication Manager (baris perintah)

Setelah Anda membuat kontenCloudFormation template Anda di JSON atau YAKL, Anda dapat menggunakanAWS Command Line Interface (AWS CLI) atauAWS Tools for PowerShell untuk menyimpan template sebagai dokumen SSM. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri.

Sebelum Anda memulai

Instal dan konfigurasikanAWS CLI atauAWS Tools for PowerShell, jika belum. Untuk informasi, lihat Menginstal atau memperbarui versi terbaruAWS CLI dan MemasangAWS Tools for PowerShell.

Linux & macOS

```
aws ssm create-document \
    --content file://path/to/template_in_json_or_yaml \
    --name "a_name_for_the_template" \
    --document-type "CloudFormation" \
    --document-format "JSON_or_YAML" \
    --tags "Key=tag-key,Value=tag-value"
```

Windows

```
aws ssm create-document ^
    --content file://C:\path\to\template_in_json_or_yaml ^
    --name "a_name_for_the_template" ^
    --document-type "CloudFormation" ^
    --document-format "JSON_or_YAML" ^
    --tags "Key=tag-key,Value=tag-value"
```

PowerShell

```
$json = Get-Content -Path "C:\path\to\template_in_json_or_yaml | Out-String
New-SSMDocument `
    -Content $json `
    -Name "a_name_for_the_template" `
    -DocumentType "CloudFormation" `
    -DocumentFormat "JSON_or_YAML" `
    -Tags "Key=tag-key,Value=tag-value"
```

Jika berhasil, sistem menampilkan respon seperti berikut ini.

```
{
    "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"
            }
        ]
    }
```

Mengedit templat

Gunakan prosedur berikut untuk mengeditCloudFormation template diApplication Manager. Perubahan template tersediaCloudFormation setelah Anda menyediakan tumpukan yang menggunakan template yang diperbarui.

Untuk mengeditCloudFormation template diApplication Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih CloudFormationtumpukan.
4. Pilih Pustaka template.
5. Pilih template, lalu pilih Tindakan, Edit. Anda tidak dapat mengubah nama templat, tetapi Anda dapat mengubah semua detail lainnya.
6. Pilih Save (Simpan). Template disimpan dalam layanan Dokumen Systems Manager.

bekerja denganCloudFormation tumpukan

Application Manager, kemampuanAWS Systems Manager, membantu Anda menyediakan dan mengelola sumber daya untuk aplikasi Anda dengan mengintegrasikan denganAWS CloudFormation. Anda dapat membuat, mengedit, dan menghapusCloudFormation templat dan tumpukanApplication

Manager. Tumpukan adalah kumpulanAWS sumber daya yang dapat Anda kelola sebagai unit tunggal. Ini berarti Anda dapat membuat, memperbarui, atau menghapus kumpulanAWS sumber daya denganCloudFormation tumpukan. Template adalah file teks yang diformat di JSON atau YAKL yang menentukan sumber daya yang ingin Anda sediakan di tumpukan Anda. Bagian ini mencakup informasi berikut.

Topik

Membuat tumpukan

Prosedur berikut menjelaskan cara membuatCloudFormation tumpukan dengan menggunakanApplication Manager. Tumpukan didasarkan pada template. Saat Anda membuat tumpukan, Anda dapat memilih template yang sudah ada atau menciptakan tumpukan baru. Setelah Anda membuat tumpukan, sistem segera mencoba untuk membuat sumber daya yang diidentifikasi dalam tumpukan. Setelah sistem berhasil menyediakan sumber daya, template dan tumpukan tersedia untuk melihat dan mengeditApplication Manager danCloudFormation.

> **Note**
>
> Tidak ada biaya untuk digunakanApplication Manager untuk membuat tumpukan, tetapi Anda dikenakan biaya untukAWS sumber daya yang dibuat di tumpukan.

MembuatCloudFormation tumpukan dengan menggunakanApplication Manager (konsol)

Gunakan prosedur berikut untuk membuat tumpukan dengan menggunakanApplication Manager dalamAWS Management Console.

Untuk membuatCloudFormation tumpukan

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih CloudFormationtumpukan.
4. Di bagian Siapkan template, pilih opsi. Jika Anda memilih Gunakan template yang ada, Anda dapat menggunakan tab di bagian Pilih template untuk menemukan template yang Anda inginkan. Jika Anda memilih salah satu opsi lain, lengkapi wizard untuk menyiapkan templat.

5. Pada halaman Tentukan detail template, verifikasi detail template untuk memastikan proses membuat sumber daya yang Anda inginkan.

   - (Opsional) Dalam bagian Tag, terapkan satu pasangan nilai kunci tag atau lebih ke templat.
   - Tag adalah metadata opsional yang Anda tetapkan ke sumber daya. Dengan menggunakan tag, Anda dapat mengategorikan sumber daya dengan cara yang berbeda, seperti berdasarkan tujuan, pemilik, atau lingkungan. Untuk informasi selengkapnya tentang penandaan sumber daya Systems Manager, lihat Penandaan sumber daya Systems Manager.
   - Pilih Selanjutnya.

6. Pada halaman Edit detail tumpukan, untuk nama Stack, masukkan nama yang membantu Anda mengidentifikasi sumber daya yang dibuat oleh stack atau tujuannya.

   - Bagian Parameter mencakup semua parameter opsional dan diperlukan yang ditentukan dalam template. Masukkan satu atau lebih parameter di setiap bidang.
   - (Opsional) Dalam bagian Tag, terapkan satu pasangan nilai kunci tag atau lebih ke tumpukan.
   - (Opsional) Dalam bagian Izin, tentukan nama peranAWS Identity and Access Management (IAM) atau API Amazon Resource Name (ARN). Sistem menggunakan peran layanan yang ditentukan untuk membuat semua sumber daya yang ditentukan dalam tumpukan Anda. Jika Anda tidak menetapkan peran IAM, makaAWS CloudFormation gunakan sesi sementara yang dihasilkan sistem dari kredenal pengguna Anda. Untuk informasi selengkapnya tentang peran IAM ini, lihat peranAWS CloudFormation layanan di PanduanAWS CloudFormation Pengguna.
   - Pilih Selanjutnya.

7. Pada halaman Ulasan dan ketentuan, tinjau semua detail tumpukan. Pilih tombol Edit di halaman ini untuk membuat perubahan.

8. Pilih Provision stack.

Application Managermenampilkan halaman CloudFormationtumpukan dan status pembuatan dan penyebaran tumpukan. JikaCloudFormation gagal membuat dan menyediakan tumpukan, lihat topik berikut di PanduanAWS CloudFormation Pengguna.

- Kode status tumpukan
- Pemecahan MasalahAWS CloudFormation

Setelah sumber daya tumpukan Anda disediakan dan berjalan, pengguna dapat mengedit sumber daya secara langsung menggunakan layanan pokok yang membuat sumber daya. Misalnya,

pengguna dapat menggunakan konsol Amazon Elastic Compute Cloud (Amazon EC2) untuk memperbarui instans server yang dibuat sebagai bagian dariCloudFormation tumpukan. Beberapa perubahan mungkin tidak disengaja, dan beberapa mungkin dibuat dengan sengaja untuk menanggapi peristiwa operasional yang sensitif terhadap waktu. Terlepas dari itu, perubahan yang dibuat di luar CloudFormation dapat memperumit operasi pembaruan atau penghapusan tumpukan. Anda dapat menggunakan deteksi penyimpangan atau status penyimpangan untuk mengidentifikasi sumber daya tumpukan yang perubahan konfigurasinya telah dibuat di luarCloudFormation manajemen. Untuk informasi tentang status penyimpangan, lihat Mendeteksi perubahan konfigurasi tidak terkelola ke tumpukan dan sumber daya.

MembuatCloudFormation tumpukan dengan menggunakanApplication Manager (baris perintah)

Gunakan prosedurAWS Command Line Interface (AWS CLI) berikut untuk menyediakan tumpukan dengan menggunakanCloudFormation template yang disimpan sebagai dokumen SSM di Systems Manager. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri. Untuk informasi tentangAWS CLI prosedur lain untuk membuat tumpukan, lihat Membuat tumpukan di PanduanAWS CloudFormation Pengguna.

Sebelum Anda memulai

Instal dan konfigurasikanAWS CLI atauAWS Tools for PowerShell, jika Anda belum melakukannya. Untuk informasi, lihat Menginstal atau memperbarui versi terbaruAWS CLI dan MemasangAWS Tools for PowerShell.

Linux & macOS

```
aws cloudformation create-stack \
    --stack-name a_name_for_the_stack \
    --template-url "ssm-doc://arn:aws:ssm:Region:account_ID:document/template_name"
 \
```

Windows

```
aws cloudformation create-stack ^
     --stack-name a_name_for_the_stack ^
     --template-url "ssm-doc://arn:aws:ssm:Region:account_ID:document/template_name"
 ^
```

PowerShell

```
New-CFNStack `
```

```
    -StackName "a_name_for_the_stack" `
    -TemplateURL "ssm-doc://arn:aws:ssm:Region:account_ID:document/template_name" `
```

Memperbarui tumpukan

Anda dapat menyebarkan pembaruan keCloudFormation tumpukan dengan mengedit tumpukan secara langsungApplication Manager. Dengan pembaruan langsung, Anda menentukan pembaruan ke templat atau parameter input. Setelah Anda menyimpan dan menyebarkan perubahan,CloudFormation perbaruiAWS sumber daya sesuai dengan perubahan yang Anda tentukan.

Anda dapat melihat perubahan yangCloudFormation akan membuat tumpukan Anda sebelum Anda memperbaruinya dengan menggunakan set perubahan. Untuk informasi selengkapnya, lihat Memperbarui tumpukan menggunakan set perubahan di PanduanAWS CloudFormation Pengguna.

Untuk memperbaruiCloudFormation tumpukanApplication Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Aplikasi, pilih CloudFormationtumpukan.
4. Pilih tumpukan dalam daftar dan kemudian pilih Tindakan, Perbarui tumpukan.
5. Pada halaman Tentukan sumber template, pilih salah satu opsi berikut, lalu pilih Berikutnya.
   - Pilih Gunakan kode template yang saat ini disediakan di tumpukan untuk melihat template. Pilih versi template dalam daftar Versi, lalu pilih Berikutnya.
   - Pilih Beralih ke template yang berbeda untuk memilih atau membuat template baru untuk tumpukan.
6. Setelah Anda selesai membuat perubahan pada template, pilih Berikutnya.
7. Pada halaman Edit detail tumpukan, Anda dapat mengedit parameter, tag, dan izin. Anda tidak dapat mengubah nama tumpukan. Buat perubahan dan pilih Berikutnya.
8. Pada halaman Tinjauan dan ketentuan, tinjau semua detail tumpukan, lalu pilih Provision stack.

Bekerja dengan cluster di Application Manager

Bagian ini mencakup topik untuk membantu Anda bekerja dengan klaster kontainer Amazon Elastic Container Service (Amazon ECS) dan Amazon Elastic Kubernetes Service (Amazon EKS) di Application Manager komponen. AWS Systems Manager

Konten

Bekerja dengan Amazon ECS di Application Manager

Dengan kemampuan Application Manager AWS Systems Manager, Anda dapat melihat dan mengelola infrastruktur klaster Amazon Elastic Container Service (Amazon ECS). Application Managermenerapkan tag ke cluster Amazon ECS Anda menggunakan Amazon Resource Name (ARN) klaster sebagai nilai tag. Application Managermenyediakan tampilan runtime komponen dari sumber daya komputasi, jaringan, dan penyimpanan dalam sebuah cluster.

> **Note**
>
> Anda tidak dapat mengelola atau melihat informasi operasi tentang kontainer Anda diApplication Manager. Anda hanya dapat mengelola dan melihat informasi operasi tentang infrastruktur yang menghosting sumber daya Amazon ECS Anda.

Tindakan yang dapat Anda lakukan di halaman ini

Anda dapat melakukan salah satu tindakan berikut di halaman ini:

- Pilih Kelola klaster untuk membuka klaster di Amazon ECS.
- Pilih Tampilkan semua untuk melihat daftar sumber daya di klaster Anda.
- Pilih Lihat CloudWatch untuk melihat alarm sumber daya di Amazon CloudWatch.
- Pilih Kelola node atau Kelola profil Fargate untuk melihat sumber daya ini di Amazon ECS.
- Pilih ID sumber daya untuk melihat informasi rinci tentang hal itu di konsol tempat ia dibuat.
- Lihat daftar yang OpsItems terkait dengan cluster Anda.

- Tampilkan riwayat runbook yang telah berjalan di klaster Anda.

Untuk membuka klaster ECS

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Klaster kontainer, pilih klaster ECS.
4. Pilih cluster dalam daftar. Application Managermembuka tab Ikhtisar.

Bekerja dengan Amazon EKS di Application Manager

Application Manager, kemampuan AWS Systems Manager, terintegrasi dengan Amazon Elastic Kubernetes Service (Amazon EKS) untuk memberikan informasi tentang kesehatan infrastruktur cluster Amazon EKS Anda. Application Managermenerapkan tag ke kluster Amazon EKS Anda menggunakan Amazon Resource Name (ARN) cluster sebagai nilai tag. Application Managermenyediakan tampilan runtime komponen dari sumber daya komputasi, jaringan, dan penyimpanan dalam sebuah cluster.

> **Note**
>
> Anda tidak dapat mengelola atau melihat informasi operasi tentang pod atau kontainer Amazon EKS AndaApplication Manager. Anda hanya dapat mengelola dan melihat informasi operasi tentang infrastruktur yang meng-host sumber daya Amazon EKS Anda.

Tindakan yang dapat Anda lakukan di halaman ini

Anda dapat melakukan salah satu tindakan berikut di halaman ini:

- Pilih Kelola klaster untuk membuka klaster di Amazon EKS.
- Pilih Tampilkan semua untuk melihat daftar sumber daya di klaster Anda.
- Pilih Lihat CloudWatch untuk melihat alarm sumber daya di Amazon CloudWatch.
- Pilih Kelola node atau Kelola profil Fargate untuk melihat sumber daya ini di Amazon EKS.
- Pilih ID sumber daya untuk melihat informasi rinci tentang hal itu di konsol tempat ia dibuat.
- Lihat daftar yang OpsItems terkait dengan cluster Anda.

- Tampilkan riwayat runbook yang telah berjalan di klaster Anda.

Untuk membuka aplikasi Klaster EKS

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Klaster kontainer, pilih Klaster EKS.
4. Pilih cluster dalam daftar. Application Managermembuka tab Ikhtisar.

Bekerja dengan runbook untuk klaster

Anda dapat mengatasi masalahAWS dengan menggunakan runbook Otomatisasi Systems Manager.Application ManagerAWS Systems Manager Bila Anda memilih Mulai runbook dariApplication Manager klaster, sistem menampilkan daftar runbook yang difilter berdasarkan jenis sumber daya di klaster Anda. Ketika Anda memilih runbook yang ingin Anda mulai, Systems Manager akan membuka halaman Eksekusi dokumen otomatisasi.

Sebelum Anda memulai

Sebelum Anda memulai runbook dariApplication Manager, lakukan hal berikut:

- Verifikasi bahwa Anda memiliki izin yang benar untuk memulai runbook. Untuk informasi selengkapnya, lihat Menyiapkan Otomatisasi.
- Tinjau dokumentasi prosedur Otomatisasi tentang memulai runbook. Untuk informasi selengkapnya, lihat Menjalankan Otomatisasi.
- Jika Anda berniat untuk memulai runbook pada beberapa sumber daya pada satu waktu, tinjau dokumentasi tentang menggunakan target dan kontrol tingkat. Untuk informasi selengkapnya, lihat Jalankan otomatisasi dalam skala besar.

Untuk memulai runbook untuk klaster dariApplication Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Application Manager.
3. Di bagian Klaster kontainer, pilih jenis kontainer.
4. Pilih klaster di daftar. Application Managermembuka tab Ikhtisar.

5. Pada tab Runbooks, pilih Mulai runbook. Application Managermembuka halaman Eksekusi dokumen otomatisasi di tab baru. Untuk informasi tentang opsi di halaman Eksekusi dokumen otomatisasi, lihat Menjalankan Otomatisasi.

AWS AppConfig

Gunakan AWS AppConfig, kemampuan AWS Systems Manager, untuk membuat, mengelola, dan men-deploy konfigurasi aplikasi dengan cepat. AWS AppConfig men-support deployment terkendali ke aplikasi dalam berbagai ukuran dan mencakup pemeriksaan dan pemantauan validasi bawaan. Anda dapat menggunakan AWS AppConfig dengan aplikasi yang di-host di instans Amazon Elastic Compute Cloud (Amazon EC2), AWS Lambda, kontainer , aplikasi mobile, atau perangkat IoT.

Informasi tentang AWS AppConfig telah dipindahkan ke panduan pengguna yang terpisah. Untuk informasi lebih lanjut, lihat Apa yang Dimaksud Dengan AWS AppConfig?

AWS Systems Manager Parameter Store

Parameter Store, kemampuan AWS Systems Manager, menyediakan penyimpanan hierarkis yang aman untuk manajemen data konfigurasi dan manajemen rahasia. Anda dapat menyimpan data seperti kata sandi, string basis data, ID Amazon Machine Image (AMI), dan kode lisensi sebagai nilai parameter. Anda dapat menyimpan nilai-nilai sebagai teks biasa atau data terenkripsi. Anda dapat mereferensi parameter Systems Manager dalam skrip, perintah, dokumen SSM, serta konfigurasi dan alur kerja otomatisasi Anda dengan menggunakan nama unik yang Anda tentukan ketika Anda membuat parameter. Untuk memulaiParameter Store, buka konsol Systems Manager. Di panel navigasi, pilih Parameter Store.

Parameter Storejuga terintegrasi dengan Secrets Manager. Anda dapat mengambil rahasia Secrets Manager saat menggunakan rahasia lain Layanan AWS yang sudah mendukung referensi ke Parameter Store parameter. Untuk informasi selengkapnya, lihat MerujukAWS Secrets Manager rahasia dariParameter Store parameter.

> **Note**
>
> Untuk menerapkan siklus hidup rotasi kata sandi, gunakan. AWS Secrets Manager Anda dapat memutar, mengelola, dan mengambil kredensial basis data, kunci API, dan rahasia lainnya sepanjang siklus hidupnya menggunakan Secrets Manager. Untuk informasi lebih lanjut, lihat Apa itu AWS Secrets Manager? dalam AWS Secrets Manager User Guide.

Bagaimana bisa Parameter Store menguntungkan organisasi saya?

Parameter Storemenawarkan manfaat ini:

- Menggunakan layanan pengelolaan rahasia yang aman, dapat diskalakan, dan di-host tanpa server untuk dikelola.
- Meningkatkan postur keamanan Anda dengan memisahkan data Anda dari kode Anda.
- Menyimpan data konfigurasi dan string terenkripsi dalam hierarki dan melacak versi.
- Mengendalikan dan meng-audit akses pada tingkat terperinci.
- Simpan parameter dengan andal karena Parameter Store di-host di beberapa Availability Zone di file Wilayah AWS.

Siapa yang harus menggunakanParameter Store?

- Setiap AWS pelanggan yang ingin memiliki cara terpusat untuk mengelola data konfigurasi.
- Developer perangkat lunak yang ingin menyimpan login dan mereferensi stream yang berbeda.
- Administrator yang ingin menerima notifikasi ketika rahasia dan kata sandi mereka diubah atau tidak diubah.

Apa saja fitur-fiturnyaParameter Store?

- Ubah pemberitahuan

  Anda dapat mengkonfigurasi notifikasi perubahan dan meminta tindakan otomatis untuk parameter dan kebijakan parameter. Untuk informasi selengkapnya, lihat Menyiapkan notifikasi atau memicu tindakan berdasarkan Parameter Store peristiwa.
- Atur parameter

  Anda dapat menandai parameter secara individual untuk membantu Anda mengidentifikasi satu atau beberapa parameter berdasarkan tag yang telah Anda tetapkan. Misalnya, Anda dapat menandai parameter untuk lingkungan atau departemen tertentu. Untuk informasi selengkapnya, lihat Menandai parameter Systems Manager.
- Versi label

  Anda dapat mengaitkan alias untuk versi parameter Anda dengan membuat label. Label dapat membantu Anda mengingat tujuan dari versi parameter ketika ada beberapa versi.

- Validasi data

  Anda dapat membuat parameter yang mengarah ke instans Amazon Elastic Compute Cloud (Amazon EC2) Parameter Store dan memvalidasi parameter ini untuk memastikan bahwa parameter tersebut mereferensikan jenis sumber daya yang diharapkan, bahwa sumber daya itu ada, dan bahwa pelanggan memiliki izin untuk menggunakan sumber daya. Misalnya, Anda dapat membuat parameter dengan Amazon Machine Image (AMI) ID sebagai nilai dengan tipe `aws:ec2:image` data, dan Parameter Store melakukan operasi validasi asinkron untuk memastikan bahwa nilai parameter memenuhi persyaratan pemformatan untuk AMI ID, dan bahwa yang ditentukan AMI tersedia di Anda. Akun AWS

- Rahasia referensi

  Parameter Storeterintegrasi dengan AWS Secrets Manager sehingga Anda dapat mengambil rahasia Secrets Manager saat menggunakan rahasia lain Layanan AWS yang sudah mendukung referensi ke Parameter Store parameter.

- Bagikan parameter dengan akun lain

  Anda dapat secara opsional memusatkan data konfigurasi dalam satu Akun AWS dan berbagi parameter dengan akun lain yang perlu mengaksesnya.

- Dapat diakses dari yang lain Layanan AWS

  Anda dapat menggunakan Parameter Store parameter dengan kemampuan Systems Manager lainnya dan Layanan AWS untuk mengambil rahasia dan data konfigurasi dari toko pusat. Parameter bekerja dengan kemampuan Systems Manager sepertiRun Command, OtomasiState Manager, dan, kemampuan AWS Systems Manager. Anda juga dapat mereferensikan parameter di sejumlah parameter lainnya Layanan AWS, termasuk yang berikut ini:

  - 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

- Integrasikan dengan yang lain Layanan AWS

Konfigurasikan integrasi dengan yang berikut ini Layanan AWS untuk enkripsi, pemberitahuan, pemantauan, dan audit:

- AWS Key Management Service (AWS KMS)
- Amazon Simple Notification Service (Amazon SNS)
- Amazon CloudWatch: Untuk informasi lebih lanjut, lihatMengkonfigurasi EventBridge aturan untuk parameter dan kebijakan parameter.
- Amazon EventBridge: Untuk informasi lebih lanjut, lihat Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS danReferensi: Pola dan jenis EventBridge acara Amazon untuk Systems Manager.
- AWS CloudTrail: Untuk informasi selengkapnya, lihat Pencatatan panggilan AWS Systems Manager API dengan AWS CloudTrail.

Apa itu parameter?

Parameter StoreParameter adalah setiap bagian data yang disimpanParameter Store, seperti blok teks, daftar nama, kata sandi, AMI ID, kunci lisensi, dan sebagainya. Anda dapat secara terpusat dan aman mereferensi data ini dalam skrip, perintah, dan dokumen SSM Anda.

Ketika Anda mereferensi sebuah parameter, Anda menentukan nama parameter dengan menggunakan konvensi berikut.

{{ssm:*parameter-name*}}

> **Note**
>
> Parameter tidak dapat direferensikan atau di-nest dalam nilai-nilai parameter lainnya. Anda tidak dapat menyertakan {{}} atau {{ssm:*parameter-name*}} dalam nilai parameter.

Parameter Storememberikan dukungan untuk tiga jenis parameter:`String`,`StringList`, dan`SecureString`.

Dengan satu pengecualian, saat Anda membuat atau memperbarui parameter, Anda memasukkan nilai parameter sebagai teks biasa, dan tidak Parameter Store melakukan validasi pada teks yang Anda masukkan. Untuk `String` parameter, bagaimanapun, Anda dapat menentukan tipe data sebagai`aws:ec2:image`, dan Parameter Store memvalidasi bahwa nilai yang Anda masukkan adalah format yang tepat untuk Amazon AMI EC2; misalnya:. `ami-12345abcdeEXAMPLE`

Jenis parameter: String

Secara default, parameter `String` terdiri dari setiap blok teks yang Anda masukkan. Sebagai contoh:

- `abc123`
- `Example Corp`
- `<img src="images/bannerImage1.png"/>`

Jenis parameter: StringList

Parameter `StringList` berisi daftar nilai yang dipisahkan koma, seperti yang ditunjukkan dalam contoh berikut.

```
Monday,Wednesday,Friday
```

```
CSV,TSV,CLF,ELF,JSON
```

Jenis parameter: SecureString

parameter `SecureString` adalah data sensitif yang perlu disimpan dan direferensikan dengan cara yang aman. Jika Anda memiliki data yang Anda tidak ingin pengguna untuk ubah atau referensikan dalam teks biasa, seperti kata sandi atau kunci lisensi, buatlah parameter tersebut menggunakan tipe data `SecureString`.

> **Important**
>
> Jangan simpan data sensitif dalam parameter `String` atau `StringList`. Untuk semua data sensitif yang harus tetap dienkripsi, gunakan hanya tipe parameter `SecureString`.
> Untuk informasi selengkapnya, lihat Membuat parameter SecureString (AWS CLI).

Kami merekomendasikan penggunaan parameter `SecureString` untuk skenario berikut:

- Anda ingin menggunakan data/parameter Layanan AWS tanpa mengekspos nilai sebagai plaintext dalam perintah, fungsi, log agen, atau log. CloudTrail
- Anda ingin mengendalikan siapa yang memiliki akses ke data sensitif.
- Anda ingin dapat mengaudit ketika data sensitif diakses (CloudTrail).

- Anda ingin mengenkripsi data sensitif Anda, dan Anda ingin membawa kunci enkripsi Anda sendiri untuk mengelola akses.

> **Important**
>
> Hanya nilai dari parameter `SecureString` yang dienkripsi. Nama parameter, deskripsi, dan properti lainnya tidak dienkripsi.

Anda dapat menggunakan tipe `SecureString` parameter untuk data tekstual yang ingin Anda enkripsi, seperti kata sandi, rahasia aplikasi, data konfigurasi rahasia, atau jenis data lain yang ingin Anda lindungi. `SecureString`data dienkripsi dan didekripsi menggunakan kunci. AWS KMS Anda dapat menggunakan kunci KMS default yang disediakan oleh AWS atau membuat dan menggunakan kunci Anda sendiri AWS KMS key. (Gunakan AWS KMS key Anda sendiri jika Anda ingin membatasi akses pengguna ke parameter `SecureString`. Untuk informasi lebih lanjut, lihat Izin IAM untuk menggunakan kunci AWS default dan kunci yang dikelola pelanggan.)

Anda juga dapat menggunakan `SecureString` parameter dengan yang lain Layanan AWS. Dalam contoh berikut, fungsi Lambda mengambil `SecureString` parameter dengan menggunakan API. GetParameters

```
from __future__ import print_function

import json
import boto3
ssm = boto3.client('ssm', 'us-east-2')
def get_parameters():
    response = ssm.get_parameters(
        Names=['LambdaSecureString'],WithDecryption=True
    )
    for parameter in response['Parameters']:
        return parameter['Value']

def lambda_handler(event, context):
    value = get_parameters()
    print("value1 = " + value)
    return value  # Echo back the first key value
```

AWS KMS enkripsi dan harga

Jika Anda memilih jenis `SecureString` parameter saat membuat parameter, Systems Manager menggunakan AWS KMS untuk mengenkripsi nilai parameter.

> **Important**
>
> Parameter Storehanya mendukung kunci KMS enkripsi simetris. Anda tidak dapat menggunakan kunci KMS enkripsi asimetris untuk mengenkripsi parameter Anda. Untuk bantuan menentukan apakah kunci KMS simetris atau asimetris, lihat Mengidentifikasi kunci KMS simetris dan asimetris dalam Panduan Developer AWS Key Management Service

Tidak ada biaya Parameter Store untuk membuat `SecureString` parameter, tetapi biaya untuk penggunaan AWS KMS enkripsi berlaku. Untuk informasi, lihat hargaAWS Key Management Service.

Untuk informasi selengkapnya tentang Kunci yang dikelola AWS dan kunci yang dikelola pelanggan, lihat AWS Key Management Service Konsep di Panduan AWS Key Management Service Pengembang. Untuk informasi selengkapnya tentang Parameter Store dan AWS KMS enkripsi, lihat Cara AWS Systems ManagerParameter Store Penggunaan AWS KMS.

> **Note**
>
> Untuk melihat Kunci yang dikelola AWS, gunakan AWS KMS `DescribeKey` operasi. Contoh ini AWS Command Line Interface (AWS CLI) digunakan `DescribeKey` untuk melihat dan Kunci yang dikelola AWS.
>
> ```
> aws kms describe-key --key-id alias/aws/ssm
> ```

Info lebih lanjut

- Buat SecureString parameter dan bergabung dengan node ke Domain (PowerShell)
- Gunakan Parameter Store untuk Mengakses Rahasia dan Config Data dengan Aman di CodeDeploy
- Artikel Menarik tentang Amazon EC2 Systems Manager Parameter Store

Menyiapkan Parameter Store

Sebelum mengatur parameter diParameter Store, kemampuanAWS Systems Manager, konfigurasikan duluAWS Identity and Access ManagementKebijakan (IAM) yang memberikan izin kepada pengguna di akun Anda untuk melakukan tindakan yang Anda tentukan. Bagian ini mencakup informasi tentang cara mengkonfigurasi kebijakan ini secara manual menggunakan konsol IAM, dan bagaimana menetapkan mereka untuk pengguna dan grup pengguna. Anda juga dapat membuat dan menetapkan kebijakan untuk mengendalikan tindakan parameter apa yang dapat dijalankan pada node yang dikelola. Bagian ini juga mencakup informasi tentang cara membuat Amazon EventBridge aturan yang memungkinkan Anda menerima notifikasi tentang perubahan pada parameter Systems Manager. Anda juga dapat menggunakan EventBridge aturan untuk memohon tindakan lain diAWSberdasarkan perubahanParameter Store.

Konten

Membatasi akses ke parameter Systems Manager menggunakan kebijakan IAM

Anda membatasi akses ke parameter AWS Systems Manager dengan menggunakan AWS Identity and Access Management (IAM). Secara lebih spesifik, Anda membuat kebijakan IAM yang membatasi akses ke operasi API berikut:

- DeleteParameter
- DeleteParameters
- DescribeParameters
- GetParameter
- GetParameters
- GetParameterHistory
- GetParametersByPath
- PutParameter

Saat menggunakan kebijakan IAM untuk membatasi akses ke parameter Systems Manager, kami sarankan Anda membuat dan menggunakan kebijakan IAM restriktif. Misalnya, kebijakan berikut memungkinkan pengguna untuk memanggil operasi API `DescribeParameters` dan `GetParameters` untuk satu set sumber daya terbatas. Ini berarti pengguna bisa mendapatkan informasi tentang dan menggunakan semua parameter yang diawali dengan `prod-*`.

```
{
    "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

Jika pengguna memiliki akses ke sebuah jalur, maka pengguna dapat mengakses semua tingkat pada jalur tersebut. Misalnya, jika pengguna memiliki izin untuk mengakses jalur `/a`, maka pengguna juga bisa mengakses `/a/b`. Bahkan jika pengguna telah secara eksplisit ditolak akses di IAM untuk parameter`/a/b`, mereka masih dapat memanggil operasi`GetParametersByPath` API API secara rekursif untuk`/a` dan melihat dan melihat`/a/b`.

Untuk administrator tepercaya, Anda dapat memberikan akses ke semua operasi API parameter Systems Manager dengan menggunakan kebijakan yang mirip dengan contoh berikut. Kebijakan ini memberikan pengguna akses penuh ke semua parameter produksi yang diawali dengan `dbserver-prod-*`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:PutParameter",
                "ssm:DeleteParameter",
                "ssm:GetParameterHistory",
                "ssm:GetParametersByPath",
                "ssm:GetParameters",
                "ssm:GetParameter",
                "ssm:DeleteParameters"
            ],
            "Resource": "arn:aws:ssm:us-east-2:123456789012:parameter/dbserver-prod-*"
        },
        {
            "Effect": "Allow",
            "Action": "ssm:DescribeParameters",
            "Resource": "*"
        }
    ]
}
```

Menolak izin

Setiap API bersifat unik dan memiliki operasi dan izin yang berbeda yang dapat Anda izinkan atau tolak secara individual. Penolakan eksplisit dalam kebijakan apa pun akan menimpa izin yang diberikan.

> **Note**
>
> Kunci AWS Key Management Service (AWS KMS) default memiliki izin `Decrypt` untuk semua prinsip IAM dalam Akun AWS. Jika Anda ingin memiliki tingkat akses yang berbeda ke parameter `SecureString` di akun Anda, kami tidak menyarankan Anda menggunakan kunci default.

Jika Anda ingin semua operasi API yang mengambil nilai parameter untuk memiliki perilaku yang sama, maka Anda dapat menggunakan pola seperti `GetParameter*` dalam sebuah kebijakan. Contoh berikut menunjukkan cara menolak `GetParameter`, `GetParameters`,

`GetParameterHistory`, dan `GetParametersByPath` untuk semua parameter yang diawali dengan `prod-*`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "ssm:GetParameter*"
            ],
            "Resource": "arn:aws:ssm:us-east-2:123456789012:parameter/prod-*"
        }
    ]
}
```

Contoh berikut menunjukkan cara menolak beberapa perintah sambil mengizinkan pengguna untuk melakukan perintah lain pada semua parameter yang diawali dengan `prod-*`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "ssm:PutParameter",
                "ssm:DeleteParameter",
                "ssm:DeleteParameters",
                "ssm:DescribeParameters"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetParametersByPath",
                "ssm:GetParameters",
                "ssm:GetParameter",
                "ssm:GetParameterHistory"
            ],
            "Resource": "arn:aws:ssm:us-east-2:123456789012:parameter/prod-*"
        }
    ]
```

```
}
```

> **Note**
>
> Riwayat parameter mencakup semua versi parameter, termasuk yang saat ini digunakan. Oleh karena itu, jika pengguna ditolak izin untuk `GetParameter`, `GetParameters`, dan `GetParameterByPath` namun diberikan izin untuk `GetParameterHistory`, mereka dapat melihat parameter saat ini, termasuk parameter `SecureString`, menggunakan `GetParameterHistory`.

Mengizinkan hanya parameter tertentu tertentu untuk berjalan pada node node

Anda dapat mengendalikan akses sehingga node yang dikelola node yang dapat menjalankan parameter yang Anda tentukan hanya parameter yang Anda tentukan parameter yang Anda tentukan parameter yang Anda tentukan.

Jika Anda memilih tipe`SecureString` parameter parameter parameter ketika Anda membuat parameter parameter Anda, Systems Manager Manager Systems Manager Systems Manager menggunakanAWS KMS untuk mengenkripsi parameter parameter parameter Anda ketika Anda membuat parameter Anda, Systems Manager Manager Systems Manager Systems AWS KMSmengenkripsi nilai dengan menggunakan salah satuKunci yang dikelola AWS atau kunci yang dikelola pelanggan. Untuk informasi selengkapnya tentang AWS KMS dan AWS KMS key, lihat Panduan Developer AWS Key Management Service.

Anda dapat melihatKunci yang dikelola AWS dengan menjalankan perintah berikut berikut dariAWS CLI.

```
aws kms describe-key --key-id alias/aws/ssm
```

Contoh berikut memungkinkan node untuk mendapatkan nilai parameterprod- Jika parameter adalah`SecureString` parameter parameter parameter, maka node mendekripsi string menggunakanAWS KMS.

> **Note**
>
> Kebijakan instans, seperti dalam contoh berikut, ditugaskan untuk peran instans tersebut dalam IAM. Untuk informasi lebih lanjut tentang mengkonfigurasi akses ke fitur Systems Manager, termasuk cara menetapkan kebijakan untuk pengguna dan instans, lihat Menyiapkan Systems Manager untuk instans EC2.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetParameters"
            ],
            "Resource": [
                "arn:aws:ssm:us-east-2:123456789012:parameter/prod-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:us-east-2:123456789012:key/4914ec06-e888-4ea5-
a371-5b88eEXAMPLE"
            ]
        }
    ]
}
```

Izin IAM untuk menggunakan kunci AWS default dan kunci yang dikelola pelanggan

Parameter Store`SecureString`parameter dienkripsi dan didekripsi menggunakanAWS KMS kunci. Anda dapat memilih untuk mengenkripsi`SecureString` parameter Anda menggunakanAWS KMS key atau kunci KMS default default default yang disediakan olehAWS.

Saat menggunakan kunci yang dikelola pelanggan, kebijakan IAM yang memberikan akses pengguna ke sebuah parameter atau jalur parameter harus menyediakan izin `kms:Encrypt`

eksplisit untuk kunci tersebut. Misalnya, kebijakan berikut memungkinkan pengguna untuk membuat, membuat, melihat, melihat parameter berikut memungkinkan pengguna untuk membuat, melihat parameter berikut memungkinkan pengguna untuk membuat, melihat, melihat, melihat, melihat, melihat, melihat`SecureString` parameter berikut ini`prod-` dalamWilayah AWS danAkun AWS melihat parameter yang diawali parameter berikut

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:PutParameter",
                "ssm:GetParameter",
                "ssm:GetParameters"
            ],
            "Resource": [
                "arn:aws:ssm:us-east-2:111122223333:parameter/prod-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "kms:Encrypt",
                "kms:GenerateDataKey"
  1
            ],
            "Resource": [
                "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-12345EXAMPLE"
            ]
        }
    ]
}
```

[1]Izin `kms:GenerateDataKey` diperlukan untuk membuat parameter lanjutan terenkripsi menggunakan kunci yang dikelola pelanggan tertentu.

Sebaliknya, semua pengguna dalam akun pelanggan memiliki akses ke kunci yang dikelola AWS secara default. Jika Anda menggunakan kunci default ini untuk mengenkripsi parameter

`SecureString` dan tidak ingin pengguna bekerja dengan parameter `SecureString`, kebijakan IAM mereka harus secara eksplisit menolak akses ke kunci default, seperti yang ditunjukkan dalam contoh kebijakan berikut.

> **Note**
>
> Anda dapat menemukan Amazon Resource Name (ARN) dari kunci default di konsol AWS KMS pada halaman kunci yang dikelola AWS. Kunci default adalah kunci yang diidentifikasi dengan `aws/ssm` dalam kolom Alias.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "kms:Decrypt",
                "kms:GenerateDataKey"
            ],
            "Resource": [
                "arn:aws:kms:us-east-2:111122223333:key/abcd1234-ab12-cd34-ef56-
abcdeEXAMPLE"
            ]
        }
    ]
}
```

Jika Anda memerlukan kontrol akses yang sangat baik atas parameter `SecureString` di akun Anda, Anda harus menggunakan kunci yang dikelola pelanggan untuk melindungi dan membatasi akses ke parameter ini. Kami juga merekomendasikan menggunakan AWS CloudTrail untuk memantau aktivitas parameter `SecureString`.

Untuk informasi selengkapnya, lihat topik berikut:

- Logika evaluasi kebijakan dalam Panduan Pengguna IAM
- Menggunakan kebijakan kunci dalam AWS KMS dalam Panduan Developer AWS Key Management Service
- Melihat peristiwa dengan riwayat CloudTrail peristiwa di PanduanAWS CloudTrail Pengguna

Mengelola tingkatan parameter

Parameter Store, kemampuan AWS Systems Manager, termasuk parameter standar dan parameter lanjutan. Anda mengkonfigurasi masing-masing parameter untuk menggunakan tingkat parameter standar (tingkat default) atau tingkat parameter lanjutan.

Anda dapat mengubah parameter standar menjadi parameter lanjutan kapan saja, tetapi Anda tidak dapat mengembalikan parameter lanjutan ke parameter standar. Ini karena mengembalikan parameter lanjutan ke parameter standar akan menyebabkan sistem memotong ukuran parameter dari 8 KB menjadi 4 KB, yang mengakibatkan kehilangan data. Mengembalikan juga akan menghapus setiap kebijakan yang melekat pada parameter. Selain itu, parameter lanjutan menggunakan bentuk enkripsi yang berbeda dari parameter standar. Untuk informasi selengkapnya, lihat Cara AWS Systems ManagerParameter Store penggunaan AWS KMS dalam Panduan AWS Key Management Service Pengembang.

Jika Anda tidak lagi memerlukan parameter lanjutan, atau jika Anda tidak lagi ingin dikenakan biaya untuk parameter lanjutan, hapus dan buat ulang sebagai parameter standar yang baru.

Tabel berikut menjelaskan perbedaan antara tingkatan.

	Standar	Lanjutan
Jumlah total parameter yang diizinkan (per Akun AWS dan Wilayah AWS)	10.000	100.000
Ukuran maksimum dari nilai parameter	4 KB	8 KB
Kebijakan parameter tersedia	Tidak	Ya Untuk informasi selengkapnya, lihat Menetapkan kebijakan parameter.
Biaya	Tanpa biaya tambahan	Biaya berlaku

	Standar	Lanjutan
		Untuk informasi selengkapnya, lihat AWS Systems Manager Harga untuk Parameter Store.

Topik

- Menentukan tingkat parameter default
- Mengubah parameter standar ke parameter lanjutan

Menentukan tingkat parameter default

Dalam permintaan untuk membuat atau memperbarui parameter (yaitu, operasi `PutParameter`), Anda dapat menentukan tingkat parameter untuk digunakan dalam permintaan tersebut. Berikut ini adalah sebuah contoh, menggunakan 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"
```

Setiap kali Anda menentukan tingkat dalam permintaan, Parameter Store membuat atau memperbarui parameter sesuai dengan permintaan Anda. Namun, jika Anda tidak secara eksplisit menentukan tingkat dalam permintaan, pengaturan tingkat Parameter Store default menentukan tingkat mana parameter dibuat.

Tingkat default saat Anda mulai menggunakan Parameter Store adalah tingkat parameter standar. Jika Anda menggunakan tingkat parameter lanjutan, Anda dapat menentukan salah satu dari hal berikut sebagai default:

- Lanjutan: Dengan opsi ini, Parameter Store mengevaluasi semua permintaan sebagai parameter lanjutan.
- Intelligent-Tiering: Dengan opsi ini, Parameter Store evaluasi setiap permintaan untuk menentukan apakah parameternya standar atau lanjutan.

  Jika permintaan tidak menyertakan opsi apa pun yang memerlukan parameter lanjutan, parameter akan dibuat di tingkat parameter standar. Jika satu atau beberapa opsi yang memerlukan parameter lanjutan disertakan dalam permintaan, Parameter Store buat parameter di tingkat parameter lanjutan.

Manfaat Intelligent-Tiering

Berikut ini adalah alasan Anda mungkin memilih Intelligent-Tiering sebagai tingkat default.

Kontrol biaya— Inteligent-Tiering membantu mengendalikan biaya terkait parameter Anda dengan selalu membuat parameter standar kecuali parameter lanjutan mutlak diperlukan.

Pemutakhiran otomatis ke tingkat parameter lanjutan — Saat Anda membuat perubahan pada kode Anda yang memerlukan peningkatan parameter standar ke parameter lanjutan, Intelligent-Tiering menangani konversi untuk Anda. Anda tidak perlu mengubah kode Anda untuk menangani pemutakhiran.

Berikut ini beberapa contoh pemutakhiran otomatis:

- AWS CloudFormation Template Anda menyediakan banyak parameter saat dijalankan. Ketika proses ini menyebabkan Anda mencapai kuota parameter 10.000 di tingkat parameter standar, Intelligent-Tiering secara otomatis memutakhirkan Anda ke tingkat parameter lanjutan, dan proses Anda tidak terganggu. AWS CloudFormation
- Anda menyimpan nilai sertifikat dalam parameter, memutar nilai sertifikat secara teratur, dan konten kurang dari kuota 4 KB dari tingkat parameter standar. Jika nilai sertifikat pengganti melebihi 4 KB, Intelligent-Tiering secara otomatis memutakhirkan parameter ke tingkat parameter lanjutan.
- Anda ingin mengasosiasikan banyak parameter standar yang ada untuk sebuah kebijakan parameter, yang memerlukan tingkat parameter lanjutan. Daripada harus menyertakan opsi --

`tier Advanced` dalam semua panggilan untuk memperbarui parameter, Intelligent-Tiering secara otomatis memutakhirkan parameter ke tingkat parameter lanjutan. Opsi Intelligent-Tiering memutakhirkan parameter dari standar ke lanjutan setiap kali kriteria untuk tingkat parameter lanjutan diperkenalkan.

Opsi yang memerlukan parameter lanjutan meliputi berikut ini:

- Ukuran isi parameter lebih dari 4 KB.
- Parameter menggunakan sebuah kebijakan parameter.
- Lebih dari 10.000 parameter sudah ada Akun AWS di Anda saat ini Wilayah AWS.

Opsi Tingkat Default

Opsi tingkat yang dapat Anda tentukan sebagai default meliputi berikut ini.

- Standar - Tingkat parameter standar adalah tingkat default saat Anda mulai menggunakan. Parameter Store Dengan menggunakan tingkat parameter standar, Anda dapat membuat 10.000 parameter untuk masing-masing Wilayah AWS parameter dalam file. Akun AWS Ukuran konten dari setiap parameter dapat sama dengan maksimum 4 KB. Parameter standar tidak mendukung kebijakan parameter. Tidak ada biaya tambahan untuk menggunakan tingkat parameter standar. Memilih Standar sebagai tingkat default berarti Parameter Store selalu mencoba membuat parameter standar untuk permintaan yang tidak menentukan tingkat.
- Advanced - Gunakan tingkat parameter lanjutan untuk membuat maksimum 100.000 parameter untuk masing-masing Wilayah AWS parameter dalam file. Akun AWS Ukuran konten setiap parameter dapat sama dengan maksimum 8 KB. Parameter lanjutan mendukung kebijakan parameter. Ada biaya untuk menggunakan tingkat parameter lanjutan. Untuk informasi selengkapnya, lihat AWS Systems Manager Harga untuk Parameter Store. Memilih Tingkat Lanjut sebagai tingkat default berarti Parameter Store selalu mencoba membuat parameter lanjutan untuk permintaan yang tidak menentukan tingkat.

  > **Note**
  >
  > Saat Anda memilih tingkat parameter lanjutan, beri otorisasi AWS secara eksplisit untuk menagih akun Anda untuk parameter lanjutan yang Anda buat.

- Intelligent-Tiering — Dengan opsi Intelligent-Tiering, Parameter Store menentukan apakah akan menggunakan tingkat parameter standar atau tingkat parameter lanjutan berdasarkan konten

permintaan. Misalnya, jika Anda menjalankan perintah untuk membuat parameter dengan konten di bawah 4 KB, dan ada kurang dari 10.000 parameter saat ini Wilayah AWS di Anda Akun AWS, dan Anda tidak menentukan kebijakan parameter, parameter standar dibuat. Jika Anda menjalankan perintah untuk membuat parameter dengan konten lebih dari 4 KB, Anda sudah memiliki lebih dari 10.000 parameter saat ini Wilayah AWS di Anda Akun AWS, atau Anda menentukan kebijakan parameter, parameter lanjutan dibuat.

> **Note**
>
> Ketika Anda memilih Intelligent-Tiering, secara eksplisit memberi wewenang AWS untuk menagih akun Anda untuk parameter lanjutan apa pun yang Anda buat.

Anda dapat mengubah pengaturan tingkat Parameter Store default kapan saja.

Mengkonfigurasi izin untuk menentukan tingkat default Parameter Store

Verifikasi bahwa Anda memiliki izin di AWS Identity and Access Management (IAM) untuk mengubah tingkat parameter default Parameter Store dengan melakukan salah satu hal berikut:

- Pastikan Anda melampirkan `AdministratorAccess` kebijakan ke entitas IAM Anda (seperti pengguna, grup, atau peran).
- Pastikan bahwa Anda memiliki izin untuk mengubah pengaturan tingkat default dengan menggunakan operasi API berikut:
  - GetServiceSetting
  - UpdateServiceSetting
  - ResetServiceSetting

Berikan izin berikut ke entitas IAM untuk memungkinkan pengguna melihat dan mengubah setelan tingkat default untuk parameter tertentu Wilayah AWS di file. Akun AWS

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"
```

```
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "arn:aws:ssm:region:account-id:servicesetting/ssm/parameter-
store/default-parameter-tier"
        }
    ]
}
```

Administrator dapat menentukan izin hanya-baca dengan menetapkan izin berikut.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "*"
        }
    ]
}
```

Untuk memberikan akses, tambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

  Buat rangkaian izin. Ikuti petunjuk dalam Buat set izin dalam Panduan Pengguna AWS IAM Identity Center .

- Pengguna yang dikelola di IAM melalui penyedia identitas:

  Buat peran untuk federasi identitas. Ikuti petunjuk dalam Membuat peran untuk penyedia identitas pihak ketiga (federasi) di Panduan Pengguna IAM.
- Pengguna IAM:
  - Buat peran yang dapat diambil pengguna Anda. Ikuti petunjuk dalam Membuat peran untuk pengguna IAM dalam Panduan Pengguna IAM.
  - (Tidak disarankan) Pasang kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk di Menambahkan izin ke pengguna (konsol) dalam Panduan Pengguna IAM.

Menentukan atau mengubah tingkat Parameter Store default (konsol)

Prosedur berikut menunjukkan cara menggunakan konsol Systems Manager untuk menentukan atau mengubah tingkat parameter default untuk saat ini Akun AWS dan Wilayah AWS.

> **Tip**
>
> Jika Anda belum membuat parameter, Anda dapat menggunakan AWS Command Line Interface (AWS CLI) atau AWS Tools for Windows PowerShell untuk mengubah tingkat parameter default. Untuk informasi selengkapnya, lihat Menentukan atau mengubah tingkat Parameter Store default () AWS CLI dan Menentukan atau mengubah tingkat Parameter Store default () PowerShell.

Untuk menentukan atau mengubah tingkat Parameter Store default

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Parameter Store.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Parameter Store.
3. Pilih tab Pengaturan.
4. Pilih Ubah tingkat default.

5. Pilih salah satu opsi berikut.

   - Standar
   - Advanced
   - Tingkat Cerdas

   Untuk informasi tentang opsi ini, lihat Menentukan tingkat parameter default.

6. Tinjau pesan, dan pilih Konfirmasi.

Jika Anda ingin mengubah pengaturan tingkatan default nanti, ulangi prosedur ini dan tentukan opsi tingkat default yang berbeda.

Menentukan atau mengubah tingkat Parameter Store default () AWS CLI

Prosedur berikut menunjukkan cara menggunakan AWS CLI untuk mengubah pengaturan tingkat parameter default untuk saat ini Akun AWS dan Wilayah AWS.

Untuk menentukan atau mengubah tingkat Parameter Store default menggunakan AWS CLI

1. Buka AWS CLI dan jalankan perintah berikut untuk mengubah pengaturan tingkat parameter default untuk spesifik Wilayah AWS dalam file Akun 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
   ```

   wilayah mewakili pengenal untuk Wilayah AWS didukung oleh AWS Systems Manager, seperti `us-east-2` untuk Wilayah Timur AS (Ohio). Untuk daftar nilai *wilayah* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

   Nilai *opsi tingkatan* mencakup `Standard`, `Advanced`, dan `Intelligent-Tiering`. Untuk informasi tentang opsi ini, lihat Menentukan tingkat parameter default.

   Tidak ada output jika perintah berhasil.

2. Jalankan perintah berikut untuk melihat pengaturan layanan tingkat parameter default saat ini untuk Parameter Store saat ini Akun AWS dan Wilayah AWS.

```
aws ssm get-service-setting --setting-id arn:aws:ssm:region:account-
id:servicesetting/ssm/parameter-store/default-parameter-tier
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "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"
    }
}
```

Jika Anda ingin mengubah pengaturan tingkatan default lagi, ulangi prosedur ini dan tentukan opsi `SettingValue` yang berbeda.

Menentukan atau mengubah tingkat Parameter Store default () PowerShell

Prosedur berikut menunjukkan cara menggunakan Alat untuk Windows PowerShell untuk mengubah pengaturan tingkat parameter default untuk spesifik Wilayah AWS di akun Amazon Web Services.

Untuk menentukan atau mengubah tingkat Parameter Store default menggunakan PowerShell

1. Ubah tingkat Parameter Store default saat ini Akun AWS dan Wilayah AWS gunakan AWS Tools for PowerShell (Alat untuk PowerShell).

   ```
   Update-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/
   ssm/parameter-store/default-parameter-tier" -SettingValue "tier-option" -
   Region region
   ```

   wilayah mewakili pengenal untuk Wilayah AWS didukung oleh AWS Systems Manager, seperti `us-east-2` untuk Wilayah Timur AS (Ohio). Untuk daftar nilai *wilayah* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

Nilai *opsi tingkatan* mencakup `Standard`, `Advanced`, dan `Intelligent-Tiering`. Untuk informasi tentang opsi ini, lihat Menentukan tingkat parameter default.

Tidak ada output jika perintah berhasil.

2. Jalankan perintah berikut untuk melihat pengaturan layanan tingkat parameter default saat ini untuk Parameter Store saat ini Akun AWS dan Wilayah AWS.

```
Get-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/
parameter-store/default-parameter-tier" -Region region
```

wilayah mewakili pengenal untuk Wilayah AWS didukung oleh AWS Systems Manager, seperti `us-east-2` untuk Wilayah Timur AS (Ohio). Untuk daftar nilai *wilayah* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

Sistem mengembalikan informasi seperti berikut ini.

```
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
```

Jika Anda ingin mengubah pengaturan tingkatan default lagi, ulangi prosedur ini dan tentukan opsi `SettingValue` yang berbeda.

Mengubah parameter standar ke parameter lanjutan

Gunakan prosedur berikut untuk mengubah parameter standar yang ada ke parameter lanjutan. Untuk informasi tentang cara membuat parameter lanjutan baru, lihat Menandai parameter Systems Manager.

Untuk mengubah parameter standar ke parameter lanjutan

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Parameter Store.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Parameter Store.

3. Pilih parameter, dan kemudian pilih Edit.
4. Untuk Deskripsi, masukkan informasi tentang parameter ini.
5. Pilih Lanjutan.
6. Untuk Nilai, masukkan nilai parameter ini. Parameter lanjutan memiliki batas nilai maksimum 8 KB.
7. Pilih Simpan perubahan.

Meningkatkan atau mengatur ulang throughput Parameter Store

Peningkatan Parameter Store throughput meningkatkan jumlah maksimum transaksi per detik (TPS) yangParameter Store, kemampuanAWS Systems Manager, dapat memproses. Peningkatan throughput memungkinkan Anda beroperasi Parameter Store pada volume yang lebih tinggi untuk mendukung aplikasi dan beban kerja yang memerlukan akses bersamaan ke beberapa parameter. Anda dapat meningkatkan kuota hingga throughput maksimal pada tab Pengaturan.

Untuk informasi selengkapnya tentang batas default dan maksimum throughput maksimal, lihat AWS Systems Managertitik akhir dan kuota.

Meningkatkan kuota throughput menimbulkan biaya pada Anda. Akun AWS Untuk informasi selengkapnya, silakan lihat Harga AWS Systems Manager.

> **Note**
>
> Pengaturan Parameter Store throughput berlaku untuk semua transaksi yang dibuat oleh semua pengguna IAM saat ini Akun AWS dan. Wilayah AWS Pengaturan throughput berlaku untuk parameter standar dan lanjutan.

Topik

- Mengkonfigurasi izin untuk mengubah throughput Parameter Store
- Meningkatkan atau mengatur ulang throughput (konsol)
- Meningkatkan atau mengatur ulang throughput () AWS CLI

- Meningkatkan atau mengatur ulang throughput () PowerShell

Mengkonfigurasi izin untuk mengubah throughput Parameter Store

Verifikasi bahwa Anda memiliki izin di IAM untuk mengubah Parameter Store throughput dengan melakukan salah satu hal berikut:

- Pastikan `AdministratorAccess` kebijakan tersebut dilampirkan ke entitas IAM Anda (pengguna, grup, atau peran).
- Pastikan bahwa Anda memiliki izin untuk mengubah pengaturan layanan throughput dengan menggunakan operasi API berikut:
  - GetServiceSetting
  - UpdateServiceSetting
  - ResetServiceSetting

Berikan izin berikut ke entitas IAM untuk memungkinkan pengguna melihat dan mengubah setelan parameter-throughput untuk parameter tertentu di file. Wilayah AWS Akun AWS

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "arn:aws:ssm:region:account-id:servicesetting/ssm/parameter-
store/high-throughput-enabled"
        }
    ]
}
```

Administrator dapat menentukan izin hanya-baca dengan menetapkan izin berikut.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "*"
        }
    ]
}
```

Untuk memberikan akses, tambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

  Buat rangkaian izin. Ikuti petunjuk dalam Buat set izin dalam Panduan Pengguna AWS IAM Identity Center.

- Pengguna yang dikelola di IAM melalui penyedia identitas:

  Buat peran untuk federasi identitas. Ikuti petunjuk dalam Membuat peran untuk penyedia identitas pihak ketiga (federasi) di Panduan Pengguna IAM.

- Pengguna IAM:
  - Buat peran yang dapat diambil pengguna Anda. Ikuti petunjuk dalam Membuat peran untuk pengguna IAM dalam Panduan Pengguna IAM.
  - (Tidak disarankan) Pasang kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk di Menambahkan izin ke pengguna (konsol) dalam Panduan Pengguna IAM.

Meningkatkan atau mengatur ulang throughput (konsol)

Prosedur berikut menunjukkan cara menggunakan konsol Systems Manager untuk meningkatkan jumlah transaksi per detik yang Parameter Store dapat diproses untuk saat ini Akun AWS danWilayah AWS. Ini juga menunjukkan cara untuk kembali ke pengaturan standar jika Anda tidak lagi memerlukan peningkatan throughput atau tidak lagi ingin mengeluarkan biaya.

> **Tip**
>
> Jika Anda belum membuat parameter, Anda dapat menggunakan AWS Command Line Interface (AWS CLI) atau AWS Tools for Windows PowerShell untuk meningkatkan throughput. Untuk informasi selengkapnya, lihat Meningkatkan atau mengatur ulang throughput () AWS CLI dan Meningkatkan atau mengatur ulang throughput () PowerShell.

Untuk menambah atau mengatur ulang Parameter Store throughput

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Parameter Store.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Parameter Store.
3. Pilih tab Pengaturan.
4. Untuk meningkatkan throughput, pilih Tetapkan batas.

   -atau-

   Untuk kembali ke batas default, pilih Reset limit.
5. Jika Anda meningkatkan batas, lakukan hal berikut:

   - Pilih kotak centang untuk Saya menerima bahwa mengubah pengaturan ini menimbulkan biaya pada pengaturan saya. Akun AWS
   - Pilih Atur batas.

   -atau-

Jika Anda mengatur ulang batas ke default, lakukan hal berikut:

- Pilih kotak centang untuk Saya menerima bahwa mengatur ulang ke batas throughput default Parameter Store menyebabkan proses transaksi lebih sedikit per detik.
- Pilih Reset limit.

Meningkatkan atau mengatur ulang throughput () AWS CLI

Prosedur berikut menunjukkan bagaimana menggunakan AWS CLI untuk meningkatkan jumlah transaksi per detik yang Parameter Store dapat memproses untuk saat ini Akun AWS danWilayah AWS. Anda juga dapat kembali ke batas default.

Untuk meningkatkan Parameter Store throughput menggunakan AWS CLI

1. Buka AWS CLI dan jalankan perintah berikut untuk meningkatkan transaksi per detik yang Parameter Store dapat memproses saat ini Akun AWS danWilayah AWS.

   ```
   aws ssm update-service-setting --setting-id arn:aws:ssm:region:account-
   id:servicesetting/ssm/parameter-store/high-throughput-enabled --setting-value true
   ```

   Tidak ada output jika perintah berhasil.
2. Jalankan perintah berikut untuk melihat pengaturan layanan throughput saat ini untuk Parameter Store saat ini Akun AWS danWilayah AWS.

   ```
   aws ssm get-service-setting --setting-id arn:aws:ssm:region:account-
   id:servicesetting/ssm/parameter-store/high-throughput-enabled
   ```

   Sistem mengembalikan informasi seperti berikut ini:

   ```
   {
       "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"
   ```

```
    }
}
```

Jika Anda tidak lagi membutuhkan peningkatan throughput, atau jika Anda tidak lagi ingin dikenakan biaya, Anda dapat kembali ke pengaturan standar. Untuk mengembalikan pengaturan Anda, jalankan perintah berikut.

```
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"
    }
}
```

Meningkatkan atau mengatur ulang throughput () PowerShell

Prosedur berikut menunjukkan cara menggunakan Alat untuk Windows PowerShell untuk meningkatkan jumlah transaksi per detik yang Parameter Store dapat memproses untuk saat ini Akun AWS danWilayah AWS. Anda juga dapat kembali ke batas default.

Untuk meningkatkan Parameter Store throughput menggunakan PowerShell

1. Tingkatkan Parameter Store throughput saat ini Akun AWS dan Wilayah AWS gunakan AWS Tools for PowerShell (Alat untuk PowerShell).

   ```
   Update-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/
   ssm/parameter-store/high-throughput-enabled" -SettingValue "true" -Region region
   ```

   Tidak ada output jika perintah berhasil.
2. Jalankan perintah berikut untuk melihat pengaturan layanan throughput saat ini untuk Parameter Store saat ini Akun AWS danWilayah AWS.

```
Get-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/
parameter-store/high-throughput-enabled" -Region region
```

Sistem mengembalikan informasi yang mirip dengan yang berikut ini:

```
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
```

Jika Anda tidak lagi membutuhkan peningkatan throughput, atau jika Anda tidak lagi ingin dikenakan biaya, Anda dapat kembali ke pengaturan standar. Untuk mengembalikan pengaturan Anda, jalankan perintah berikut.

```
Reset-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/
parameter-store/high-throughput-enabled" -Region region
```

Sistem mengembalikan informasi seperti berikut ini:

```
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
```

Menyiapkan notifikasi atau memicu tindakan berdasarkan Parameter Store peristiwa

Topik di bagian ini menjelaskan cara menggunakan Amazon EventBridge dan Amazon Simple Notification Service (Amazon SNS) untuk memberi tahu Anda tentang perubahan parameter. AWS Systems Manager Anda dapat membuat EventBridge aturan untuk memberi tahu Anda saat parameter atau versi label parameter dibuat, diperbarui, atau dihapus. Peristiwa dikeluarkan atas dasar upaya terbaik. Anda dapat diberitahu tentang perubahan atau status terkait kebijakan

parameter, seperti saat parameter berakhir, akan kedaluwarsa, atau belum berubah selama jangka waktu tertentu.

> **Note**
>
> Kebijakan parameter tersedia untuk parameter yang menggunakan tingkat parameter lanjutan. Biaya berlaku. Untuk informasi lebih lanjut, lihat Menetapkan kebijakan parameter dan Mengelola tingkatan parameter.

Topik dalam bagian ini juga menjelaskan cara memulai tindakan lain pada target untuk peristiwa parameter tertentu. Misalnya, Anda dapat menjalankan fungsi AWS Lambda untuk membuat parameter secara otomatis ketika parameter tersebut kedaluwarsa atau dihapus. Anda dapat mengatur notifikasi untuk memanggil fungsi Lambda ketika kata sandi basis data Anda diperbarui. Fungsi Lambda dapat memaksa koneksi database Anda untuk mengatur ulang atau menyambung kembali dengan kata sandi baru. EventBridge juga mendukung menjalankan Run Command perintah dan eksekusi Otomasi, dan tindakan di banyak lainnyaLayanan AWS. Run Commanddan Otomasi adalah kemampuan keduanyaAWS Systems Manager. Untuk informasi selengkapnya, lihat Panduan EventBridge Pengguna Amazon.

Sebelum Anda Memulai

Buat sumber daya apa pun yang Anda butuhkan untuk menentukan tindakan target untuk aturan yang Anda buat. Misalnya, jika aturan yang Anda buat adalah untuk mengirim notifikasi, pertama buatlah topik Amazon SNS. Untuk informasi lebih lanjut, lihat Memulai dengan Amazon SNS di Panduan Developer Amazon Simple Notification Service.

Mengkonfigurasi EventBridge aturan untuk parameter dan kebijakan parameter

Topik ini menjelaskan hal berikut:

- Cara membuat EventBridge aturan yang memanggil target berdasarkan peristiwa yang terjadi pada satu atau lebih parameter di AndaAkun AWS.
- Cara membuat EventBridge aturan yang memanggil target berdasarkan peristiwa yang terjadi pada satu atau beberapa kebijakan parameter di AndaAkun AWS. Ketika Anda membuat parameter lanjutan, Anda menentukan kapan parameter kedaluwarsa, kapan harus menerima notifikasi sebelum parameter kedaluwarsa, dan berapa lama menunggu sebelum notifikasi harus dikirim bahwa parameter belum berubah. Anda mengatur notifikasi untuk peristiwa ini menggunakan

prosedur berikut. Untuk informasi lebih lanjut, lihat Menetapkan kebijakan parameter dan Mengelola tingkatan parameter.

Untuk mengonfigurasi EventBridge aturan untuk parameter Systems Manager atau kebijakan parameter

1. Buka EventBridge konsol Amazon di https://console.aws.amazon.com/events/.
2. Di panel navigasi, pilih Aturan, lalu pilih Buat aturan.

   -atau-

   Jika EventBridge halaman beranda terbuka terlebih dahulu, pilih Buat aturan.
3. Masukkan nama dan deskripsi untuk aturan.

   Aturan tidak boleh memiliki nama yang sama dengan aturan lain di Wilayah yang sama dan di bus kejadian yang sama.
4. Untuk bus acara, pilih bus acara yang ingin Anda kaitkan dengan aturan ini. Jika Anda ingin aturan ini dimulai pada acara pencocokan yang berasal dari Anda sendiriAkun AWS, pilih default. Ketika Layanan AWS di akun Anda memancarkan acara, itu selalu masuk ke bus acara default akun Anda.
5. Untuk jenis Aturan, biarkan Aturan default dengan pola acara yang dipilih.
6. Pilih Berikutnya.
7. Untuk sumber Acara, biarkan AWSacara default atau acara EventBridge mitra dipilih. Anda dapat melewati bagian Contoh acara.
8. Untuk Pola peristiwa, lakukan hal berikut:

   - Pilih pola kustom (editor JSON).
   - Untuk pola Peristiwa, tempelkan salah satu konten berikut di dalam kotak, tergantung apakah Anda membuat aturan untuk parameter atau kebijakan parameter:

     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"
        ]
    }
}
```

- Ubah konten untuk parameter dan operasi yang ingin Anda tindaklanjuti, seperti yang ditunjukkan pada sampel berikut.

Parameter

Dengan contoh ini, tindakan diambil ketika salah satu parameter bernama/`Oncall`dan `/Project/Teamlead` diperbarui:

```
{
    "source": [
        "aws.ssm"
    ],
    "detail-type": [
        "Parameter Store Change"
    ],
    "detail": {
        "name": [
            "/Oncall",
            "/Project/Teamlead"
        ],
        "operation": [
            "Update"
        ]
    }
}
```

Parameter policy

Dengan contoh ini, tindakan diambil setiap kali parameter bernama/`OncallDuties`kedaluwarsa dan dihapus:

```
{
    "source": [
        "aws.ssm"
    ],
    "detail-type": [
        "Parameter Store Policy Action"
    ],
    "detail": {
        "parameter-name": [
            "/OncallDuties"
        ],
        "policy-type": [
            "Expiration"
        ]
```

```
    }
}
```

9. Pilih Berikutnya.
10. Untuk Target 1, pilih jenis target dan sumber daya yang didukung. Misalnya, jika Anda memilih Topik SNS, buat pilihan untuk Topik. Jika Anda memilih CodePipeline, masukkan ARN pipeline untuk ARN Pipeline. Berikan nilai konfigurasi tambahan sesuai kebutuhan.

> **Tip**
>
> Pilih Tambahkan target lain jika Anda memerlukan target tambahan untuk aturan tersebut.

11. Pilih Berikutnya.
12. (Opsional) Masukkan satu atau lebih tanda untuk aturan. Untuk informasi selengkapnya, lihat EventBridgetag Amazon di Panduan EventBridge Pengguna Amazon.
13. Pilih Berikutnya.
14. Pilih Buat aturan.

Info lebih lanjut

- Gunakan label parameter untuk pembaruan konfigurasi yang mudah di seluruh lingkungan
- Tutorial: Gunakan EventBridge untuk menyampaikan peristiwa ke AWS Systems ManagerRun Command dalam Panduan EventBridge Pengguna Amazon
- Tutorial: Tetapkan AWS Systems Manager Otomasi sebagai EventBridge target di Panduan EventBridge Pengguna Amazon

Bekerja dengan Parameter Store

Bagian ini menjelaskan cara menata dan membuat parameter tag, dan cara membuat versi parameter yang berbeda. Anda dapat menggunakan AWS Systems Manager konsol, konsol Amazon Elastic Compute Cloud (Amazon EC2), atau AWS CLI () untuk membuat dan AWS Command Line Interface bekerja dengan parameter. Untuk informasi selengkapnya tentang parameter, lihat Apa itu parameter?

Topik

- Menandai parameter Systems Manager

- Mencari parameter Systems Manager
- Menetapkan kebijakan parameter
- Bekerja dengan hierarki parameter
- Bekerja dengan label parameter
- Bekerja dengan versi parameter
- Bekerja dengan parameter bersama
- Bekerja dengan parameter menggunakan Run Command perintah
- Dukungan parameter native untuk ID Amazon Machine Image
- Menghapus parameter Systems Manager

Menandai parameter Systems Manager

Menggunakan informasi dalam topik berikut untuk membantu Anda membuat parameter Systems Manager menggunakanAWS Systems Managerkonsol,AWS Command Line Interface(AWS CLI), atauAWS Tools for Windows PowerShell(Alat untuk Windows PowerShell).

Bagian ini menunjukkan cara membuat, menyimpan, dan menjalankan parameter denganParameter Storedalam lingkungan pengujian. Bagian ini juga menunjukkan cara menggunakanParameter Storedengan kemampuan Systems Manager dan lainnyaLayanan AWS. Untuk informasi selengkapnya, lihatApa itu parameter?

Tentang persyaratan dan kendala untuk nama parameter

Gunakan informasi dalam topik ini untuk membantu Anda menentukan nilai yang valid untuk nama parameter saat Anda membuat parameter.

Informasi ini melengkapi detail dalam topikPutParameterdi dalamAWS Systems ManagerReferensi API, yang juga menyediakan informasi tentang nilai-nilaiAllowedPattern,Deskripsi,KeyId,Timpa,Jenis, danNilai.

Persyaratan dan kendala untuk nama parameter mencakup hal berikut:

- Sensitivitas kasus: Nama parameter peka huruf besar atau kecil.
- Spasi: Nama parameter tidak boleh mengandung spasi.
- Karakter valid: Nama parameter hanya dapat terdiri dari simbol dan huruf berikut:`a-zA-Z0-9_.-`

Selain itu, karakter garis miring (/) digunakan untuk menggambarkan hierarki dalam nama parameter. Sebagai contoh: `/Dev/Production/East/Project-ABC/MyParameter`

- ValiditasAMIformat: Bila Anda memilih`aws:ec2:image`sebagai tipe data untuk`String`parameter, ID yang Anda masukkan harus memvalidasi untukAMIFormat ID`ami-12345abcdeEXAMPLE`.
- Berkualifikasi penuh: Saat Anda membuat atau mereferensi sebuah parameter dalam hierarki, sertakan karakter garis miring ke depan (/). Saat Anda mereferensi parameter yang merupakan bagian dari sebuah hierarki, tentukan seluruh jalur hierarki termasuk garis miring awal (/).
  - Nama parameter yang memenuhi syarat: `MyParameter1`, `/MyParameter2`, `/Dev/Production/East/Project-ABC/MyParameter`
  - Nama parameter yang tidak memenuhi syarat: `MyParameter3/L1`
- Panjang: Panjang maksimum untuk nama parameter yang Anda buat adalah 1011 karakter. Ini termasuk karakter dalam ARN yang mendahului nama yang Anda tentukan, seperti`arn:aws:ssm:us-east-2:111122223333:parameter/`.
- Awalan: Nama parameter tidak dapat diawali dengan”`aws`“atau”`ssm`“(peka huruf besar kecil). Sebagai contoh, upaya untuk membuat parameter dengan nama berikut gagal dengan pengecualian:
  - `awsTestParameter`
  - `SSM-testparameter`
  - `/aws/testparam1`

  > **Note**
  >
  > Saat Anda menentukan parameter dalam dokumen SSM, perintah, atau skrip, sertakan `ssm` sebagai bagian dari sintaks. Sebagai contoh, {{ssm:*nama-parameter*}} dan {{ ssm:*nama-parameter*}}, seperti `{{ssm:MyParameter}}`, dan `{{ ssm:MyParameter }}`.

- Keunikan: Sebuah nama parameter harus unik dalamWilayah AWS. Sebagai contoh, Systems Manager memperlakukan yang berikut ini sebagai parameter terpisah, jika mereka ada di Region yang sama:
  - `/Test/TestParam1`
  - `/TestParam1`

  Contoh berikut ini juga unik:

- `/Test/TestParam1/Logpath1`
- `/Test/TestParam1`

  Namun, contoh-contoh berikut, jika di Region yang sama, tidak unik:

- `/TestParam1`
- `TestParam1`

- Kedalaman hierarki: Jika Anda menentukan hierarki parameter, hierarki tersebut dapat memiliki kedalaman maksimum lima belas tingkat. Anda dapat menentukan parameter pada setiap tingkat hierarki. Kedua contoh berikut ini secara struktural valid:
  - `/Level-1/L2/L3/L4/L5/L6/L7/L8/L9/L10/L11/L12/L13/L14/parameter-name`
  - `parameter-name`

  Berupaya untuk membuat parameter berikut akan gagal dengan pengecualian `HierarchyLevelLimitExceededException`:
  - `/Level-1/L2/L3/L4/L5/L6/L7/L8/L9/L10/L11/L12/L13/L14/L15/L16/parameter-name`

> **Important**
>
> Jika pengguna memiliki akses ke sebuah jalur, maka pengguna dapat mengakses semua tingkat pada jalur tersebut. Misalnya, jika pengguna memiliki izin untuk mengakses jalur `/a`, maka pengguna juga bisa mengakses `/a/b`. Bahkan jika pengguna secara eksplisit telah ditolak aksesnyaAWS Identity and Access Management(IAM) untuk parameter`/a/b`, mereka masih bisa memanggilGetParametersByPathOperasi API secara rekursif untuk`/a`dan pandangan`/a/b`.

Topik

- Membuat parameter Systems Manager (konsol)
- Membuat parameter Systems Manager (AWS CLI)
- Membuat parameter Systems Manager (Tools for WindowsPowerShell)

Membuat parameter Systems Manager (konsol)

Anda dapat menggunakan konsol AWS Systems Manager untuk membuat dan menjalankan tipe parameter `String`, `StringList`, dan `SecureString`. Setelah menghapus parameter, tunggu setidaknya 30 detik untuk membuat parameter dengan nama yang sama.

> **Note**
>
> Parameter hanya tersedia di Wilayah AWS tempat mereka dibuat.

Prosedur berikut memandu Anda melalui prosesParameter Store membuat. Anda dapat membuat`String`,`StringList`, dan jenis`SecureString` parameter dari konsol.

Untuk membuat parameter

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Parameter Store.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon (☰) untuk membuka panel, lalu pilih Parameter Store.
3. Pilih Buat parameter.
4. Di kotak Nama, masukkan hierarki dan nama. Misalnya, enter **`/Test/helloWorld`**.

   Untuk informasi selengkapnya tentang hierarki parameter, lihat Bekerja dengan hierarki parameter.
5. Di kotak Deskripsi, ketik deskripsi yang mengidentifikasi parameter ini sebagai parameter uji.
6. Untuk Tingkat parameter pilih Standard atau Lanjutan. Untuk informasi selengkapnya tentang parameter lanjutan, lihat Mengelola tingkatan parameter.
7. Untuk Type, pilih String, StringList, atau SecureString.

   - Jika Anda memilih String, bidang Tipe data akan ditampilkan. Jika Anda membuat parameter untuk menahan ID sumber daya untuk Amazon Machine Image (AMI), pilih `aws:ec2:image`. Jika tidak, biarkan default `text` dipilih.

- Jika Anda memilih SecureString, bidang ID Kunci KMS ditampilkan. Jika Anda tidak memberikanAWS Key Management ServiceAWS KMS key ID,AWS KMS key Amazon Resource Name (ARN), nama alias, atau ARN alias, maka sistem menggunakan`alias/aws/ssm`, yang merupakanKunci yang dikelola AWS untuk Systems Manager. Jika Anda tidak ingin menggunakan kunci ini, maka Anda dapat menggunakan kunci yang dikelola pelanggan. Untuk informasi lebih lanjut tentangKunci yang dikelola AWS dan kunci yang dikelola pelanggan, lihat AWS Key Management ServiceKonsep dalam PanduanAWS Key Management Service Developer. Untuk informasi selengkapnya tentangParameter Store danAWS KMS enkripsi, lihat CaraAWS Systems ManagerParameter Store PenggunaanAWS KMS.

> **Important**
>
> Parameter Storehanya mendukung kunci KMS enkripsi simetris. Anda tidak dapat menggunakan kunci KMS enkripsi asimetris untuk mengenkripsi. Untuk bantuan menentukan apakah kunci KMS simetris atau asimetris, lihat Mengidentifikasi kunci KMS simetris dan asimetris dalam Panduan Developer AWS Key Management Service

- Saat membuat parameter `SecureString` di konsol dengan menggunakan parameter `key-id`dengan nama alias atau ARN alias kunci yang dikelola pelanggan, tentukan prefiks `alias/` sebelum alias. Berikut ini adalah contoh ARN.

```
arn:aws:kms:us-east-2:123456789012:alias/abcd1234-ab12-cd34-ef56-abcdeEXAMPLE
```

  Berikut ini adalah contoh nama alias.

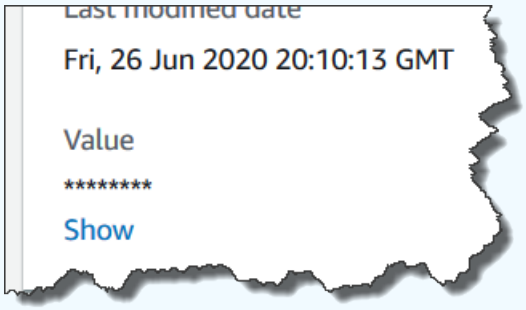
```
alias/MyAliasName
```

8. Di kotak Nilai, ketik sebuah nilai. Sebagai contoh, ketik **`This is my first parameter`** atau **`ami-0dbf5ea29aEXAMPLE`**.

> **Note**
>
> Parameter tidak dapat direferensikan atau di-nest dalam nilai-nilai parameter lainnya. Anda tidak dapat menyertakan `{{}}` atau `{{ssm:parameter-name}}` dalam nilai parameter.
> Jika Anda memilih SecureString, nilai dari parameter ditutupi secara default ("****** ****

**** ****

**** **** **** **** **** **** **** **** Pilih Tampilkan untuk menampilkan nilai parameter.



9. (Opsional) Dalam area Tag, terapkan satu atau lebih pasangan nilai kunci tag ke parameter.

   Tag adalah metadata opsional yang Anda tetapkan ke sumber daya. Tag memungkinkan Anda untuk mengkategorikan sumber daya dengan berbagai cara, seperti berdasarkan tujuan, pemilik, atau lingkungan. Misalnya, Anda mungkin ingin menandai parameter Systems Manager untuk mengidentifikasi tipe sumber daya yang berlaku, lingkungan, atau jenis data konfigurasi yang direferensikan oleh parameter tersebut. Dalam kasus ini, Anda bisa menentukan pasangan nilai kunci berikut:

   - `Key=Resource,Value=S3bucket`
   - `Key=OS,Value=Windows`
   - `Key=ParameterType,Value=LicenseKey`

10. Pilih Buat parameter.

11. Di daftar parameter, pilih nama parameter yang baru saja Anda buat. Verifikasi detailnya pada tab Gambaran Umum. Jika Anda membuat sebuah parameter `SecureString`, pilih Tampilkan untuk melihat nilai yang tidak terenkripsi.

> **Note**
>
> Anda tidak dapat mengubah parameter lanjutan ke parameter standar. Jika Anda tidak lagi memerlukan parameter lanjutan, atau jika Anda tidak lagi ingin dikenakan biaya untuk parameter lanjutan, hapus dan buat ulang sebagai parameter standar yang baru.

Membuat parameter Systems Manager (AWS CLI)

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk membuat tipe parameter `String`, `StringList`, dan `SecureString`. Setelah menghapus parameter, tunggu setidaknya 30 detik untuk membuat parameter dengan nama yang sama.

Parameter tidak dapat direferensikan atau di-nest dalam nilai-nilai parameter lainnya. Anda tidak dapat menyertakan `{{}}` atau `{{ssm:parameter-name}}` dalam nilai parameter.

> **Note**
>
> Parameter hanya tersedia di Wilayah AWS tempat mereka dibuat.

Topik

- Membuat parameter String (AWS CLI)
- Membuat parameter StringList (AWS CLI)
- Membuat parameter SecureString (AWS CLI)
- Membuat parameter multi-baris (AWS CLI)

Membuat parameter **`String`** (AWS CLI)

1. Instal dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaruAWS CLI.

2. Jalankan perintah berikut ini untuk membuat parameter tipe-`String`. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri.

   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"
```

-atau-

Jalankan perintah berikut ini untuk membuat parameter yang berisi ID Amazon Machine Image (AMI) sebagai nilai parameter.

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"
```

Opsi `--name` mendukung hierarki. Untuk informasi tentang hierarki, lihat Bekerja dengan hierarki parameter.

Opsi `--data-type` harus ditentukan hanya jika Anda membuat parameter yang berisi ID AMI. Opsi ini memvalidasi bahwa nilai parameter yang Anda masukkan memiliki format yang benar untuk ID AMI Amazon Elastic Compute Cloud (Amazon EC2). Untuk semua parameter

lainnya, tipe data default adalah `text` dan tidak wajib untuk menentukan nilai. Untuk informasi selengkapnya, lihat Dukungan parameter native untuk ID Amazon Machine Image.

> **Important**
>
> Jika berhasil, perintah mengembalikan nomor versi parameter. Pengecualian: Jika Anda telah menentukan `aws:ec2:image` sebagai tipe data, nomor versi baru dalam respons tidak berarti bahwa nilai parameter telah divalidasi. Untuk informasi selengkapnya, lihat Dukungan parameter native untuk ID Amazon Machine Image.

Contoh berikut ini menambahkan dua tag pasangan nilai kunci ke sebuah parameter.

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\"}]
```

Contoh berikut ini menggunakan hierarki parameter dalam nama untuk membuat parameter `String` teks biasa. Perintah tersebut mengembalikan nomor versi parameter. Untuk informasi selengkapnya tentang hierarki parameter, lihat Bekerja dengan hierarki parameter.

Linux & macOS

Parameter tidak dalam sebuah hierarki

```
aws ssm put-parameter \
    --name "golden-ami" \
    --type "String" \
    --value "ami-12345abcdeEXAMPLE"
```

Parameter dalam sebuah hierarki

```
aws ssm put-parameter \
    --name "/amis/linux/golden-ami" \
    --type "String" \
    --value "ami-12345abcdeEXAMPLE"
```

Windows

Parameter tidak dalam sebuah hierarki

```
aws ssm put-parameter ^
    --name "golden-ami" ^
    --type "String" ^
    --value "ami-12345abcdeEXAMPLE"
```

Parameter dalam sebuah hierarki

```
aws ssm put-parameter ^
    --name "/amis/windows/golden-ami" ^
    --type "String" ^
    --value "ami-12345abcdeEXAMPLE"
```

3. Jalankan perintah berikut untuk menampilkan nilai parameter terbaru dan memverifikasi detail parameter baru Anda.

```
aws ssm get-parameters --names "/Test/IAD/helloWorld"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "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"
        }
    ]
}
```

Jalankan perintah berikut ini untuk mengubah nilai parameter . Perintah tersebut mengembalikan nomor versi parameter.

```
aws ssm put-parameter --name "/Test/IAD/helloWorld" --value "My updated 1st parameter"
 --type String --overwrite
```

Jalankan perintah berikut ini untuk melihat riwayat nilai parameter.

```
aws ssm get-parameter-history --name "/Test/IAD/helloWorld"
```

Jalankan perintah berikut untuk menggunakan parameter ini dalam sebuah perintah.

```
aws ssm send-command --document-name "AWS-RunShellScript" --parameters '{"commands":
["echo {{ssm:/Test/IAD/helloWorld}}"]}' --targets "Key=instanceids,Values=instance-ids"
```

Jalankan perintah berikut jika Anda hanya ingin mengambil nilai parameter.

```
aws ssm get-parameter --name testDataTypeParameter --query "Parameter.Value"
```

Jalankan perintah berikut jika Anda hanya ingin mengambil nilai parameter menggunakan `get-parameters`.

```
aws ssm get-parameters --names "testDataTypeParameter" --query "Parameters[*].Value"
```

Jalankan perintah berikut ini untuk melihat metadata parameter.

```
aws ssm describe-parameters --filters "Key=Name,Values=/Test/IAD/helloWorld"
```

> **Note**
>
> Nama harus dikapitalisasi.

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "Parameters": [
        {
            "Name": "helloworld",
            "Type": "String",
            "LastModifiedUser": "arn:aws:iam::123456789012:user/JohnDoe",
            "LastModifiedDate": 1494529763.156,
            "Version": 1,
            "Tier": "Standard",
            "Policies": []
        }
    ]
}
```

Membuat parameter **`StringList`** (AWS CLI)

1. Instal dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaruAWS CLI.
2. Jalankan perintah berikut ini untuk membuat sebuah parameter. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri.

   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**
>
> Jika berhasil, perintah mengembalikan nomor versi parameter.

Contoh berikut ini menambahkan dua tag pasangan nilai kunci ke sebuah parameter. (Tergantung pada jenis sistem operasi pada mesin lokal Anda, jalankan salah satu perintah berikut. Versi untuk dijalankan dari mesin Windows lokal termasuk karakter escape ("\") yang Anda butuhkan untuk menjalankan perintah dari alat baris perintah Anda.)

Berikut adalah contoh `StringList` yang menggunakan hierarki parameter.

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**
>
> Item dalam `StringList` harus dipisahkan dengan koma (,). Anda tidak dapat menggunakan tanda baca lain atau karakter khusus untuk item escape dalam daftar. Jika Anda memiliki nilai parameter yang memerlukan koma, maka gunakan tipe `String`.

3. Jalankan perintah `get-parameters` untuk memverifikasi detail parameter. Misalnya:

```
aws ssm get-parameters --name "/IAD/ERP/Oracle/addUsers"
```

Membuat parameter SecureString (AWS CLI)

Gunakan prosedur berikut untuk membuat sebuah parameter `SecureString`. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri.

> **Important**
>
> Hanya nilai dari parameter `SecureString` yang dienkripsi. Nama parameter, deskripsi, dan properti lainnya tidak dienkripsi.

> **Important**
>
> Parameter Storehanya mendukung kunci KMS enkripsi simetris. Anda tidak dapat menggunakan kunci enkripsi asimetris untuk mengenkripsi parameter Anda. Untuk bantuan menentukan apakah kunci KMS simetris atau asimetris, lihat Mengidentifikasi kunci KMS simetris dan asimetris dalam Panduan Developer AWS Key Management Service

1. Instal dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaruAWS CLI.

2. Jalankan salah satu perintah berikut untuk membuat parameter yang menggunakan tipe data `SecureString`.

   Linux & macOS

   Buat**`SecureString`** parameter menggunakan defaultKunci yang dikelola AWS

   ```
   aws ssm put-parameter \
       --name "parameter-name" \
       --value "parameter-value" \
       --type "SecureString"
   ```

Membuat**SecureString** parameter yang menggunakan kunci yang dikelola pelanggan

```
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"
```

Membuat**SecureString** parameter yang menggunakanAWS KMS kunci kustom

```
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

Buat**SecureString** parameter menggunakan defaultKunci yang dikelola AWS

```
aws ssm put-parameter ^
    --name "parameter-name" ^
    --value "parameter-value" ^
    --type "SecureString"
```

Membuat**SecureString** parameter yang menggunakan kunci yang dikelola pelanggan

```
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"
```

Membuat**SecureString** parameter yang menggunakanAWS KMS kunci kustom

```
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"
```

Jika Anda membuat`SecureString` parameter dengan menggunakanKunci yang dikelola AWS kunci di akun dan Region Anda, maka Anda tidak harus memberikan nilai untuk`--key-id` parameter.

> **Note**
>
> Untuk menggunakan AWS KMS key yang ditugaskan ke Akun AWS dan Wilayah AWS, hapus parameter `key-id` dari perintah. Untuk informasi selengkapnya tentang AWS KMS keys, lihat Konsep AWS Key Management Service dalam Panduan Developer AWS Key Management Service.

Untuk menggunakan kunci yang dikelola pelanggan dan bukan kunci yangKunci yang dikelola AWS ditetapkan ke akun Anda, tentukan kunci dengan menggunakan`--key-id` parameter. Parameter mendukung format parameter KMS berikut.

- Contoh Amazon Resource Name (ARN) Kunci:

  `arn:aws:kms:us-east-2:123456789012:key/key-id`

- Contoh ARN Alias:

  `arn:aws:kms:us-east-2:123456789012:alias/alias-name`

- Contoh ID Kunci:

  `12345678-1234-1234-1234-123456789012`

- Contoh Nama Alias:

  `alias/MyAliasName`

Anda dapat membuat kunci yang dikelola pelanggan dengan menggunakan AWS Management Console atau API AWS KMS. Perintah AWS CLI berikut ini membuat kunci yang dikelola pelanggan di Wilayah AWS pada Akun AWS Anda.

```
aws kms create-key
```

Gunakan perintah dalam format berikut untuk membuat parameter `SecureString` menggunakan kunci yang baru Anda buat.

Contoh berikut menggunakan nama yang dikaburkan (`3l3vat3131`) untuk parameter kata sandi danAWS 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. Jalankan perintah berikut ini untuk memverifikasi detail parameter.

Jika Anda tidak menentukan parameter `with-decryption`, atau jika Anda menentukan parameter `no-with-decryption`, perintah tersebut mengembalikan GUID yang terenkripsi.

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. Jalankan perintah berikut ini untuk melihat metadata parameter.

   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. Jalankan perintah berikut ini untuk mengubah nilai parameter jika Anda tidak menggunakan AWS KMS key yang dikelola pelanggan.

   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
   ```

   -atau-

   Jalankan perintah berikut ini untuk mengubah nilai parameter jika Anda menggunakan AWS KMS key yang dikelola pelanggan.

   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. Jalankan perintah berikut ini untuk melihat nilai parameter terbaru.

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. Jalankan perintah berikut ini untuk melihat riwayat nilai parameter.

   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**
>
> Anda dapat membuat parameter secara manual dengan suatu nilai terenkripsi. Dalam hal ini, karena nilai sudah dienkripsi, Anda tidak perlu memilih tipe parameter `SecureString`. Jika Anda memilih `SecureString`, parameter Anda akan terenkripsi ganda.

Secara default, semua nilai `SecureString` ditampilkan sebagai cipher-text. Untuk mendekripsi sebuah nilai `SecureString`, pengguna harus memiliki izin untuk memanggil operasi API AWS KMS Dekripsi. Untuk informasi tentang mengkonfigurasi kontrol akses AWS KMS, lihat Autentikasi dan Kontrol Akses untuk AWS KMS di Panduan Developer AWS Key Management Service.

> **Important**
>
> Jika Anda mengubah alias kunci KMS untuk kunci KMS yang digunakan untuk mengenkripsi parameter, maka Anda juga harus memperbarui alias kunci yang digunakan parameter untuk referensiAWS KMS. Ini hanya berlaku untuk alias kunci KMS; ID kunci yang dilampirkan alias tetap sama kecuali Anda menghapus seluruh kunci.

Membuat parameter multi-baris (AWS CLI)

Anda dapat menggunakan AWS CLI untuk membuat parameter dengan jeda baris. Gunakan jeda baris untuk memecah teks dalam nilai parameter yang lebih panjang untuk keterbacaan yang lebih

baik atau, misalnya, memperbarui konten parameter multi-paragraf untuk halaman web. Anda dapat menyertakan konten dalam file JSON dan menggunakan opsi `--cli-input-json`, menggunakan karakter jeda baris seperti `\n`, seperti yang ditunjukkan dalam contoh berikut.

1. Instal dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaruAWS CLI.

2. Jalankan perintah berikut ini untuk membuat parameter multi baris.

   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
   ```

   Contoh berikut ini menunjukkan konten dari file `MultiLineParameter.json`.

   ```
   {
       "Value": "<para>Paragraph One</para>\n<para>Paragraph Two</para>
   \n<para>Paragraph Three</para>"
   }
   ```

Nilai parameter yang tersimpan akan disimpan sebagai berikut.

```
<para>Paragraph One</para>
<para>Paragraph Two</para>
<para>Paragraph Three</para>
```

Membuat parameter Systems Manager (Tools for WindowsPowerShell)

Anda dapat menggunakan AWS Tools for Windows PowerShell untuk membuat tipe parameter `String`, `StringList`, dan `SecureString`. Setelah menghapus parameter, tunggu setidaknya 30 detik untuk membuat parameter dengan nama yang sama.

Parameter tidak dapat direferensikan atau di-nest dalam nilai-nilai parameter lainnya. Anda tidak dapat menyertakan `{{}}` atau `{{ssm:parameter-name}}` dalam nilai parameter.

> **Note**
>
> Parameter hanya tersedia di Wilayah AWS tempat mereka dibuat.

Topik

- BuatString parameter (Alat untuk WindowsPowerShell)
- BuatStringList parameter (Alat untuk WindowsPowerShell)
- BuatSecureString parameter (Alat untuk WindowsPowerShell)

Buat**`String`** parameter (Alat untuk WindowsPowerShell)

1. Instal dan konfigurasiAWS Tools for PowerShell (Tools for WindowsPowerShell), jika Anda belum melakukannya.

   Untuk informasi, lihat MenginstalAWS Tools for PowerShell.

2. Jalankan perintah berikut ini untuk membuat parameter yang berisi nilai teks biasa. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri.

   ```
   Write-SSMParameter `
       -Name "parameter-name" `
       -Value "parameter-value" `
       -Type "String"
   ```

   -atau-

   Jalankan perintah berikut ini untuk membuat parameter yang berisi ID Amazon Machine Image (AMI) sebagai nilai parameter.

> **Note**
>
> Untuk membuat parameter dengan tag, buat service.model.tag sebelum tangan sebagai variabel. Inilah contohnya.
>
> ```
> $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
```

Opsi `-DataType` harus ditentukan hanya jika Anda membuat parameter yang berisi ID AMI. Untuk semua parameter lainnya, tipe data default adalah `text`. Untuk informasi selengkapnya, lihat Dukungan parameter native untuk ID Amazon Machine Image.

Berikut ini adalah contoh yang menggunakan hierarki parameter.

```
Write-SSMParameter `
    -Name "/IAD/Web/SQL/IPaddress" `
    -Value "99.99.99.999" `
    -Type "String" `
    -Tags $tag
```

3. Jalankan perintah berikut ini untuk memverifikasi detail parameter.

```
(Get-SSMParameterValue -Name "the-parameter-name-you-specified").Parameters
```

Buat**`StringList`** parameter (Alat untuk WindowsPowerShell)

1. Instal dan konfigurasiAWS Tools for PowerShell (Tools for WindowsPowerShell), jika Anda belum melakukannya.

Untuk informasi, lihat MenginstalAWS Tools for PowerShell.

2. Jalankan perintah berikut ini untuk membuat sebuahStringList parameter. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

> **Note**
>
> Untuk membuat parameter dengan tag, buat service.model.tag sebelum tangan sebagai variabel. Inilah contohnya.
>
> ```
> $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
```

Jika berhasil, perintah mengembalikan nomor versi parameter.

Ini contohnya.

```
Write-SSMParameter `
    -Name "stringlist-parameter" `
    -Value "Milana,Mariana,Mark,Miguel" `
    -Type "StringList" `
    -Tags $tag
```

> **Note**
>
> Item dalam `StringList` harus dipisahkan dengan koma (,). Anda tidak dapat menggunakan tanda baca lain atau karakter khusus untuk item escape dalam daftar. Jika Anda memiliki nilai parameter yang memerlukan koma, maka gunakan tipe `String`.

3. Jalankan perintah berikut ini untuk memverifikasi detail parameter.

```
(Get-SSMParameterValue -Name "the-parameter-name-you-specified").Parameters
```

BuatSecureString parameter (Alat untuk WindowsPowerShell)

Sebelum Anda membuat parameter `SecureString`, baca tentang persyaratan untuk tipe parameter ini. Untuk informasi selengkapnya, lihat Membuat parameter SecureString (AWS CLI).

> **Important**
>
> Hanya nilai dari parameter `SecureString` yang dienkripsi. Nama parameter, deskripsi, dan properti lainnya tidak dienkripsi.

> **Important**
>
> Parameter Storehanya mendukung kunci KMS enkripsi simetris. Anda tidak dapat menggunakan kunci KMS enkripsi asimetris untuk mengenkripsi parameter Anda. Untuk bantuan menentukan apakah kunci KMS simetris atau asimetris, lihat Mengidentifikasi kunci KMS simetris dan asimetris dalam Panduan Developer AWS Key Management Service

1. Instal dan konfigurasiAWS Tools for PowerShell (Tools for WindowsPowerShell), jika Anda belum melakukannya.

   Untuk informasi, lihat MenginstalAWS Tools for PowerShell.
2. Jalankan perintah berikut ini untuk membuat sebuah parameter. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri.

   > **Note**
   >
   > Untuk membuat parameter dengan tag, pertama buat service.model.tag sebagai variabel. Inilah contohnya.
   >
   > ```
   > $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
```

Jika berhasil, perintah mengembalikan nomor versi parameter.

> **Note**
>
> Untuk menggunakan yangKunci yang dikelola AWS ditetapkan ke akun Anda, hapus`-KeyId` parameter dari perintah.

Berikut ini adalah contoh yang menggunakan nama yang dikaburkan (3l3vat3131) untuk parameter kata sandi danKunci yang dikelola AWS -.

```
Write-SSMParameter `
    -Name "/Finance/Payroll/3l3vat3131" `
    -Value "P@sSwW)rd" `
    -Type "SecureString"`
    -Tags $tag
```

3. Jalankan perintah berikut ini untuk memverifikasi detail parameter.

```
(Get-SSMParameterValue -Name "the-parameter-name-you-specified" -WithDecryption
 $true).Parameters
```

Secara default, semua nilai `SecureString` ditampilkan sebagai cipher-text. Untuk mendekripsi sebuah nilai `SecureString`, pengguna harus memiliki izin untuk memanggil operasi API AWS KMS Dekripsi. Untuk informasi tentang mengkonfigurasi kontrol akses AWS KMS, lihat Autentikasi dan Kontrol Akses untuk AWS KMS di Panduan Developer AWS Key Management Service.

> **Important**
>
> Jika Anda mengubah alias kunci KMS untuk kunci KMS yang digunakan untuk mengenkripsi parameter, maka Anda juga harus memperbarui alias kunci yang digunakan parameter untuk referensiAWS KMS. Ini hanya berlaku untuk alias kunci KMS; ID kunci yang dilampirkan alias tetap sama kecuali Anda menghapus seluruh kunci.

Mencari parameter Systems Manager

Saat Anda memiliki banyak parameter di akun Anda, mungkin sulit untuk menemukan informasi tentang satu atau beberapa parameter sekaligus. Dalam hal ini, Anda dapat menggunakan alat filter untuk mencari informasi yang Anda butuhkan, sesuai dengan kriteria pencarian yang Anda tentukan. Anda dapat menggunakan AWS Systems Manager konsol, AWS Command Line Interface (AWS CLI),AWS Tools for PowerShell, atau DescribeParametersAPI untuk mencari parameter.

Topik

- Mencari parameter (konsol)
- Mencari parameter (AWS CLI)

Mencari parameter (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Parameter Store.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Parameter Store.
3. Pilih di kotak pencarian dan pilih cara Anda ingin mencari. Misalnya, `Type` atau `Name`.
4. Berikan informasi untuk tipe pencarian yang Anda pilih. Sebagai contoh:
   - Jika Anda mencari dengan `Type`, pilih dari `String`, `StringList`, atau `SecureString`.
   - Jika Anda mencari dengan `Name`, pilih `contains`, `equals`, atau `begins-with`, lalu masukkan semua atau sebagian nama parameter.

> **Note**
>
> Di konsol, tipe pencarian default untuk `Name` adalah `contains`.

5. Tekan Enter.

Daftar parameter diperbarui dengan hasil pencarian Anda.

Mencari parameter (AWS CLI)

Gunakan perintah `describe-parameters` untuk melihat informasi tentang satu atau lebih parameter dalam AWS CLI.

Contoh berikut ini menunjukkan berbagai opsi yang dapat Anda gunakan untuk melihat informasi tentang parameter di Akun AWS Anda. Untuk informasi selengkapnya tentang opsi ini, lihat describe-parametersdi Panduan AWS Command Line Interface Pengguna.

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaru dari AWS CLI.
2. Ganti nilai sampel dalam perintah berikut dengan nilai yang mencerminkan parameter yang telah dibuat di akun Anda.

   Linux & macOS

   ```
   aws ssm describe-parameters \
       --parameter-filters "Key=Name,Values=MyParameterName"
   ```

   Windows

   ```
   aws ssm describe-parameters ^
       --parameter-filters "Key=Name,Values=MyParameterName"
   ```

   > **Note**
   >
   > Untuk `describe-parameters`, tipe pencarian default untuk `Name` adalah `Equals`. Dalam filter parameter Anda, menentukan

"Key=Name,Values=*MyParameterName*" sama saja dengan menentukan "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

Pada contoh terakhir, *key-id* mewakili ID dari kunci AWS Key Management Service (AWS KMS) yang digunakan untuk mengenkripsi suatu parameter `SecureString` yang dibuat di akun Anda. Sebagai alternatif, Anda bisa masukkan **`alias/aws/ssm`** untuk menggunakan kunci AWS KMS default untuk akun Anda. Untuk informasi selengkapnya, lihat Membuat parameter SecureString (AWS CLI).

Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut ini.

```
{
    "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": []
        }
    ]
}
```


Menetapkan kebijakan parameter

Kebijakan parameter membantu Anda mengelola serangkaian parameter yang berkembang dengan memungkinkan Anda menetapkan kriteria tertentu ke parameter seperti tanggal kedaluwarsa atau waktu untuk hidup. Kebijakan parameter sangat membantu dalam memaksa Anda untuk memperbarui atau menghapus kata sandi dan data konfigurasi yang disimpan diParameter Store, suatu kemampuanAWS Systems Manager. Parameter Storemenawarkan jenis kebijakan berikut:`Expiration`,`ExpirationNotification`, dan`NoChangeNotification`.

> **Note**
>
> Untuk menerapkan siklus hidup rotasi kata sandi, gunakan AWS Secrets Manager. Anda dapat memutar, mengelola, dan mengambil kredensial basis data, kunci API, dan rahasia lainnya sepanjang siklus hidupnya menggunakan Secrets Manager. Untuk informasi lebih lanjut, lihat Apa itu AWS Secrets Manager? di Panduan Pengguna AWS Secrets Manager.

Parameter Storememberlakukan kebijakan parameter dengan menggunakan pemindaian periodik yang asinkron. Setelah membuat kebijakan, Anda tidak perlu melakukan tindakan tambahan untuk menerapkan kebijakan tersebut. Parameter Storesecara mandiri melakukan tindakan yang ditentukan oleh kebijakan tersebut sesuai dengan kriteria yang Anda tentukan.

> **Note**
>
> Kebijakan parameter tersedia untuk parameter yang menggunakan tingkat parameter lanjutan. Untuk informasi selengkapnya, lihat Mengelola tingkatan parameter.

Kebijakan parameter adalah suatu array JSON, seperti yang ditunjukkan dalam tabel berikut. Anda dapat menetapkan kebijakan saat membuat parameter lanjutan baru, atau Anda dapat menerapkan kebijakan dengan memperbarui parameter. Parameter Storemendukung tipe kebijakan parameter berikut ini.

Kebijakan	Detail	Contoh
Kedaluwarsa	Kebijakan ini menghapus parameter. Anda dapat menentukan tanggal dan waktu tertentu dengan menggunakan format `ISO_INSTANT` atau format `ISO_OFFSET_DATE_TIME` . Untuk mengubah kapan Anda ingin parameter dihapus, perbarui kebijakan . Memperbarui sebuah	`{ "Type": "Expiration", "Version": "1.0", "Attributes": { "Timestamp": "2018-12-02T21:34:33.000Z" } }`

Kebijakan	Detail	Contoh
	parameter tidak mempengaruhi waktu atau tanggal kedaluwarsa kebijakan yang dilampirkan padanya. Ketika waktu dan tanggal kedaluwarsa tercapai,Parameter Store hapus parameter. **Note** Contoh ini menggunakan format `ISO_INSTANT` . Anda juga dapat menentukan tanggal dan waktu dengan menggunakan format `ISO_OFFSET_DATE_TIME` . Ini contohnya : `2019-11-01T22:13:48.87+10:30:00` .	
ExpirationNotification	Kebijakan ini memulai peristiwa di AmazonEventBridge (EventBridge) yang memberitahu Anda tentang kedaluwarsa. Dengan menggunakan kebijakan ini, Anda dapat menerima notifikasi sebelum waktu kedaluwarsa tiba, dalam satuan hari atau jam.	`{ "Type": "ExpirationNotification", "Version": "1.0", "Attributes": { "Before": "15", "Unit": "Days" } }`

Kebijakan	Detail	Contoh
NoChangeNotification	Kebijakan ini memulai peristiwaEventBridge jika sebuah parameter belum dimodifikasi untuk jangka waktu tertentu. Jenis kebijakan ini berguna ketika, misalnya, kata sandi perlu diubah dalam jangka waktu tertentu. Kebijakan ini menentuka n kapan harus mengirim notifikasi dengan membaca atribut `LastModifiedTime` dari parameter. Jika Anda mengubah atau mengedit parameter, sistem akan mengatur ulang periode waktu notifikasi berdasarkan nilai baru `LastModifiedTime` .	`{ "Type": "NoChange Notification", "Version": "1.0", "Attributes": { "After": "20", "Unit": "Days" } }`

Anda dapat menetapkan beberapa kebijakan ke sebuah parameter. Misalnya, Anda dapat menetapkan`Expiration` dan`ExpirationNotification` kebijakan agar sistem memulaiEventBridge peristiwa untuk memberitahu Anda tentang penghapusan parameter yang akan datang. Anda dapat menetapkan maksimum sepuluh (10) kebijakan ke sebuah parameter.

Contoh berikut menunjukkan sintaks permintaan untuk permintaan PutParameterAPI yang menetapkan empat kebijakan ke`SecureString` parameter baru bernama`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"
            }
        }
    ]
}
```

Menambahkan tag ke parameter yang sudah ada

Bagian ini mencakup informasi tentang cara menambahkan kebijakan ke sebuah parameter yang ada dengan menggunakan konsol AWS Systems Manager, AWS Command Line Interface (AWS CLI), dan AWS Tools for Windows PowerShell . Untuk informasi tentang cara membuat parameter baru yang menyertakan kebijakan, lihat Menandai parameter Systems Manager.

Topik

- Menambahkan kebijakan ke parameter yang sudah ada (konsol)

- Menambahkan kebijakan ke parameter yang sudah ada (AWS CLI)
- Menambahkan kebijakan ke parameter yang ada (Tools for WindowsPowerShell)

Menambahkan kebijakan ke parameter yang sudah ada (konsol)

Gunakan prosedur berikut untuk menambahkan kebijakan ke sebuah parameter yang ada dengan menggunakan konsol Systems Manager.

Untuk menambahkan kebijakan ke parameter yang sudah ada

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Parameter Store.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Parameter Store.
3. Pilih opsi di sebelah parameter yang ingin Anda perbarui untuk menyertakan kebijakan, dan kemudian pilih Edit.
4. Pilih Lanjutan.
5. (Opsional) Dalam bagian Kebijakan parameter, pilih Diaktifkan. Anda dapat menentukan tanggal kedaluwarsa dan satu atau lebih kebijakan notifikasi untuk parameter ini.
6. Pilih Simpan perubahan.

⚠ Important

- Parameter Storemempertahankan kebijakan pada suatu parameter sampai Anda menimpa kebijakan tersebut dengan kebijakan baru atau menghapus kebijakan tersebut.
- Untuk menghapus semua kebijakan dari parameter yang ada, edit parameter dan terapkan kebijakan kosong dengan menggunakan kurung dan kurung kurawal, sebagai berikut: `[{}]`
- Jika Anda menambahkan kebijakan baru ke parameter yang sudah memiliki kebijakan, maka Systems Manager akan menimpa kebijakan yang melekat pada parameter tersebut. Kebijakan yang ada dihapus. Jika Anda ingin menambahkan kebijakan baru untuk

parameter yang sudah memiliki satu atau lebih kebijakan, salin dan tempel kebijakan asli, ketik kebijakan baru, dan kemudian simpan perubahan Anda.

Menambahkan kebijakan ke parameter yang sudah ada (AWS CLI)

Gunakan prosedur berikut untuk menambahkan kebijakan ke sebuah parameter yang ada dengan menggunakan AWS CLI.

Untuk menambahkan kebijakan ke parameter yang sudah ada

1. Instal dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaru dariAWS CLI.

2. Jalankan perintah berikut untuk menambahkan kebijakan ke sebuah parameter yang ada. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

   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}]"
   ```

   Berikut ini adalah contoh yang mencakup kebijakan kedaluwarsa yang menghapus parameter setelah 15 hari. Contoh tersebut juga mencakup kebijakan notifikasi yang

menghasilkanEventBridge peristiwa lima (5) hari sebelum parameter dihapus. Terakhir, contoh ini menyertakan kebijakan `NoChangeNotification` jika tidak ada perubahan yang dibuat untuk parameter ini setelah 60 hari. Contoh menggunakan nama yang dikaburkan (`3l3vat3131`) untuk parameter kata sandi dan AWS Key Management Service AWS KMS key. Untuk informasi selengkapnya tentang AWS KMS keys, lihat Konsep AWS Key Management Service dalam Panduan Developer 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. Jalankan perintah berikut ini untuk memverifikasi detail parameter. Ganti *`nama parameter`* dengan informasi Anda sendiri.

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 Storemempertahankan kebijakan untuk suatu parameter sampai Anda menimpa kebijakan tersebut dengan kebijakan baru atau menghapus kebijakan tersebut.
- Untuk menghapus semua kebijakan dari parameter yang ada, edit parameter dan terapkan kebijakan kosong dengan menggunakan kurung dan kurung kurawal. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri. Misalnya:

  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 "[{}]"
  ```

- Jika Anda menambahkan kebijakan baru ke parameter yang sudah memiliki kebijakan, maka Systems Manager akan menimpa kebijakan yang melekat pada parameter tersebut. Kebijakan yang ada dihapus. Jika Anda ingin menambahkan kebijakan baru untuk parameter yang sudah memiliki satu atau lebih kebijakan, salin dan tempel kebijakan asli, ketik kebijakan baru, dan kemudian simpan perubahan Anda.

Menambahkan kebijakan ke parameter yang ada (Tools for WindowsPowerShell)

Gunakan prosedur berikut untuk menambahkan kebijakan ke sebuah parameter yang ada dengan menggunakan Tools for WindowsPowerShell. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

Untuk menambahkan kebijakan ke parameter yang sudah ada

1. Buka Tools for WindowsPowerShell dan jalankan perintah berikut untuk menentukan kredensial Anda. Anda harus memiliki hak istimewa administrator di Amazon Elastic Compute Cloud (Amazon EC2), atau telah diberikan izin yang sesuai di AWS Identity and Access Management (IAM).

   ```
   Set-AWSCredentials `
       -AccessKey access-key-name `
       -SecretKey secret-key-name
   ```

2. Jalankan perintah berikut untuk mengatur Area untuk PowerShell yang tepat. Contoh ini menggunakan Region US East (Ohio) (us-east-2).

   ```
   Set-DefaultAWSRegion `
       -Region us-east-2
   ```

3. Jalankan perintah berikut untuk menambahkan kebijakan ke sebuah parameter yang ada. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

   ```
   Write-SSMParameter `
       -Name "parameter name" `
       -Value "parameter value" `
       -Type "parameter type" `
       -Policies "[{policies-enclosed-in-brackets-and-curly-braces}]" `
       -Overwrite
   ```

   Berikut ini adalah contoh yang mencakup kebijakan kedaluwarsa yang menghapus parameter pada tengah malam (GMT) tanggal 13 Mei 2020. Contoh tersebut juga mencakup kebijakan notifikasi yang menghasilkanEventBridge peristiwa lima (5) hari sebelum parameter dihapus. Terakhir, contoh ini menyertakan kebijakan `NoChangeNotification` jika tidak ada perubahan yang dibuat untuk parameter ini setelah 60 hari. Contoh menggunakan nama yang dikaburkan (`3l3vat3131`) untuk parameter kata sandi danKunci yang dikelola 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. Jalankan perintah berikut ini untuk memverifikasi detail parameter. Ganti *nama parameter* dengan informasi Anda sendiri.

```
(Get-SSMParameterValue -Name "parameter name").Parameters
```

Important

- Parameter Storemempertahankan kebijakan pada suatu parameter sampai Anda menimpa kebijakan tersebut dengan kebijakan baru atau menghapus kebijakan tersebut.
- Untuk menghapus semua kebijakan dari parameter yang ada, edit parameter dan terapkan kebijakan kosong dengan menggunakan kurung dan kurung kurawal. Sebagai contoh:

  ```
  Write-SSMParameter `
      -Name "parameter name" `
      -Value "parameter value" `
      -Type "parameter type" `
      -Policies "[{}]"
  ```

- Jika Anda menambahkan kebijakan baru ke parameter yang sudah memiliki kebijakan, maka Systems Manager akan menimpa kebijakan yang melekat pada parameter tersebut. Kebijakan yang ada dihapus. Jika Anda ingin menambahkan kebijakan baru untuk parameter yang sudah memiliki satu atau lebih kebijakan, salin dan tempel kebijakan asli, ketik kebijakan baru, dan kemudian simpan perubahan Anda.

Bekerja dengan hierarki parameter

Mengelola puluhan atau ratusan parameter sebagai daftar yang datar memakan waktu dan rentan terhadap kesalahan. Akan sulit juga untuk mengidentifikasi parameter yang benar untuk suatu tugas. Ini berarti Anda mungkin secara tidak sengaja menggunakan parameter yang salah, atau Anda dapat membuat beberapa parameter yang menggunakan data konfigurasi yang sama.

Anda dapat menggunakan hierarki parameter untuk membantu Anda menata dan mengelola parameter. Hierarki adalah nama parameter yang menyertakan jalur yang Anda tentukan dengan menggunakan garis miring (/).

Topik

Contoh hierarki parameter

Contoh berikut ini menggunakan tiga tingkat hierarki dalam nama untuk mengidentifikasi hal berikut:

```
/Environment/Type of computer/Application/Data
```

```
/Dev/DBServer/MySQL/db-string13
```

Anda dapat membuat hierarki dengan maksimal 15 tingkat. Kami menyarankan bahwa Anda membuat hierarki yang mencerminkan struktur hierarkis yang ada di lingkungan Anda, seperti yang ditunjukkan dalam contoh berikut:

- Lingkungan Continuous Integration dan Continuous Delivery (alur kerja CI/CD)

  ```
  /Dev/DBServer/MySQL/db-string
  ```

  ```
  /Staging/DBServer/MySQL/db-string
  ```

  ```
  /Prod/DBServer/MySQL/db-string
  ```

- Aplikasi Anda yang menggunakan kontainer

  ```
  /MyApp/.NET/Libraries/my-password
  ```

- Organisasi bisnis Anda

  `/Finance/Accountants/UserList`

  `/Finance/Analysts/UserList`

  `/HR/Employees/EU/UserList`

Hierarki parameter menstandardisasi cara Anda membuat parameter dan memudahkan untuk mengelola parameter dari waktu ke waktu. Hierarki parameter juga dapat membantu Anda mengidentifikasi parameter yang benar untuk suatu tugas konfigurasi. Ini membantu Anda untuk menghindari membuat beberapa parameter dengan data konfigurasi yang sama.

Anda dapat membuat suatu hierarki yang memungkinkan Anda untuk berbagi parameter di lingkungan yang berbeda, seperti yang ditunjukkan dalam contoh berikut yang menggunakan kata sandi dalam lingkungan pengembangan dan staging.

`/DevTest/MyApp/database/my-password`

Anda lalu dapat membuat kata sandi unik untuk lingkungan produksi Anda, seperti yang ditunjukkan dalam contoh berikut:

`/prod/MyApp/database/my-password`

Anda tidak diharuskan untuk menentukan hierarki parameter. Anda dapat membuat parameter pada tingkat satu. Ini disebut parameter root. Untuk kompatibilitas mundur, semua parameter yang dibuatParameter Store sebelum hierarki dirilis adalah parameter root. Sistem memperlakukan kedua parameter berikut sebagai parameter root.

`/parameter-name`

`parameter-name`

Melakukan kueri parameter dalam hierarki

Manfaat lain menggunakan hierarki adalah kemampuan untuk melakukan kueri untuk semua parameter dalam sebuah hierarki dengan menggunakan operasi GetParametersByPathAPI. Misalnya, jika Anda menjalankan perintah berikut dari AWS Command Line Interface (AWS CLI), sistem mengembalikan semua parameter di tingkat IIS.

```
aws ssm get-parameters-by-path --path /Dev/Web/IIS
```

Untuk melihat parameter `SecureString` terdekripsi dalam sebuah hierarki, Anda menentukan jalur dan parameter `--with-decryption`, seperti yang ditunjukkan dalam contoh berikut.

```
aws ssm get-parameters-by-path --path /Prod/ERP/SAP --with-decryption
```

Pembatasan akses ke operasiParameter Store API

Dengan kebijakanAWS Identity and Access Management (IAM), Anda dapat menyediakan atau membatasi akses pengguna ke operasiParameter Store API dan konten (IAM).

Dalam contoh kebijakan berikut ini, pengguna pertama-tama diberikan akses untuk menjalankan operasi API `PutParameter` pada semua parameter dalam Akun AWS 123456789012 di Region US East (Ohio) (us-east-2). Tapi kemudian pengguna dibatasi dari mengubah nilai-nilai paramater yang sudah ada karena opsi `Overwrite` ditolak secara eksplisit untuk operasi `PutParameter`. Dengan kata lain, pengguna yang ditetapkan dengan kebijakan ini dapat membuat parameter, tetapi tidak dapat membuat perubahan pada parameter yang ada.

```
{
    "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/*"
        }
    ]
```

```
}
```

Mengelola parameter menggunakan hierarki (AWS CLI)

Prosedur ini menunjukkan cara bekerja dengan parameter dan hierarki parameter dengan menggunakan AWS CLI.

Untuk mengelola parameter menggunakan hierarki

1. Instal dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaru dariAWS CLI.

2. Jalankan perintah berikut untuk membuat parameter yang menggunakan parameter `allowedPattern` dan tipe parameter `String`. Pola yang diperbolehkan dalam contoh ini berarti nilai untuk parameter harus antara 1 dan 4 digit panjangnya.

   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
   ```

   Perintah mengembalikan nomor versi parameter.

3. Jalankan perintah berikut untuk mencoba menimpa parameter yang baru saja Anda buat dengan nilai yang baru.

   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
```

Sistem mengembalikan kesalahan berikut karena nilai baru tidak memenuhi persyaratan pola yang diperbolehkan sebagaimana yang Anda tentukan pada langkah sebelumnya.

```
An error occurred (ParameterPatternMismatchException) when calling the PutParameter
 operation: Parameter value, cannot be validated against allowedPattern: \d{1,4}
```

4. Jalankan perintah berikut untuk membuat`SecureString` parameter yang menggunakanKunci yang dikelola AWS. Pola yang diperbolehkan dalam contoh ini berarti pengguna dapat menentukan karakter apa pun, dan nilai harus antara 8 dan 20 karakter.

   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. Jalankan perintah berikut untuk membuat lebih banyak parameter yang menggunakan struktur hierarki dari langkah sebelumnya.

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. Jalankan perintah berikut ini untuk mendapatkan nilai dua parameter.

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. Jalankan perintah berikut untuk melakukan kueri untuk semua parameter dalam satu tingkat.

   Linux & macOS

   ```
   aws ssm get-parameters-by-path \
       --path "/MyService/Test"
   ```

   Windows

   ```
   aws ssm get-parameters-by-path ^
       --path "/MyService/Test"
   ```

8. Jalankan perintah berikut ini untuk menghapus dua parameter.

   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"
   ```

Bekerja dengan label parameter

Label parameter adalah alias yang ditetapkan pengguna untuk membantu Anda mengelola versi parameter yang berbeda. Saat Anda memodifikasi sebuah parameter, AWS Systems Manager secara otomatis menyimpan versi baru dan menambah nomor versi dengan satu. Label dapat membantu Anda mengingat tujuan dari sebuah versi parameter ketika ada beberapa versi.

Sebagai contoh, katakanlah Anda memiliki parameter yang bernama`/MyApp/DB/ConnectionString`. Nilai parameter tersebut adalah string koneksi ke server MySQL dalam sebuah basis data lokal di lingkungan pengujian. Setelah Anda selesai memperbarui aplikasi, Anda ingin parameter tersebut menggunakan string koneksi untuk basis data produksi. Anda mengubah nilai`/MyApp/DB/ConnectionString`. Systems Manager secara otomatis membuat versi dua dengan string koneksi baru. Untuk membantu Anda mengingat tujuan setiap versi, Anda melampirkan label ke setiap parameter. Untuk versi satu, Anda melampirkan label Pengujian dan untuk versi dua Anda melampirkan label Produksi.

Anda dapat memindahkan label dari satu versi parameter ke versi lainnya. Sebagai contoh, jika Anda membuat parameter `/MyApp/DB/ConnectionString` versi 3 dengan string koneksi untuk basis data produksi yang baru, maka Anda dapat memindahkan label Produksi dari parameter versi 2 ke parameter versi 3.

Label parameter adalah alternatif ringan untuk tag parameter. Organisasi Anda mungkin memiliki pedoman ketat untuk tag yang harus diterapkan ke berbagai sumber daya AWS berbeda. Sebaliknya, label hanyalah sebuah asosiasi teks untuk versi parameter tertentu.

Mirip dengan tag, Anda dapat melakukan kueri untuk parameter dengan menggunakan label. Anda dapat melihat daftar versi parameter tertentu yang semuanya menggunakan label yang sama jika Anda melakukan kueri untuk parameter yang ditetapkan dengan menggunakan operasi GetParametersByPathAPI, seperti yang dijelaskan nanti dalam bagian ini.

> **Note**
>
> Jika Anda menjalankan perintah yang menentukan versi parameter yang tidak ada, perintah gagal. Itu tidak jatuh kembali ke nilai terbaru atau default parameter.

Persyaratan dan batasan label label label dan pembatasan label label

Label parameter memiliki persyaratan dan pembatasan berikut:

- Versi parameter dapat memiliki maksimum 10 label.
- Anda tidak dapat melampirkan label yang sama ke versi yang berbeda dari parameter yang sama. Sebagai contoh, jika versi 1 parameter mempunyai label Produksi, maka Anda tidak dapat melampirkan Produksi ke versi 2.
- Anda dapat memindahkan label dari satu versi parameter ke yang lainnya.

- Anda tidak dapat membuat label saat membuat parameter. Anda harus melampirkan label ke versi parameter tertentu.
- Jika Anda sudah tidak ingin menggunakan label parameter, Anda dapat memindahkannya ke versi parameter yang berbeda atau menghapusnya.
- Label dapat memiliki maksimum 100 karakter.
- Label dapat berisi huruf (peka terhadap huruf besar-kecil), angka, titik (.), tanda hubung (-), atau garis bawah (_).
- Label tidak dapat dimulai dengan angka, "aws", atau "ssm" (tidak peka terhadap huruf besar-kecil). Jika label tidak memenuhi persyaratan ini, maka label tidak dilampirkan ke versi parameter dan sistem menampilkannya dalam daftar `InvalidLabels`.

Topik

Bekerja dengan label parameter (konsol)

Bagian ini menjelaskan cara untuk melakukan tugas-tugas berikut dengan menggunakan konsol Systems Manager.

Membuat label parameter (konsol)

Prosedur berikut ini menjelaskan cara melampirkan label ke versi tertentu dari parameter yang sudah ada dengan menggunakan konsol Systems Manager. Anda tidak dapat melampirkan label saat membuat parameter.

Untuk melampirkan label ke versi parameter

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Parameter Store.

-atau-

JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Parameter Store.

3. Pilih nama parameter untuk membuka halaman detail untuk parameter tersebut.
4. Pilih tab Riwayat.
5. Pilih versi parameter yang ingin Anda lampirkan labelnya.
6. Pilih Kelola label.
7. Pilih Tambah label baru.
8. Dalam kotak teks, masukkan nama label. Untuk menambahkan label lainnya, pilih Tambah label baru. Anda dapat melampirkan maksimal sepuluh label.
9. Setelah Anda selesai, pilih Simpan perubahan.

Melihat label yang dilampirkan pada parameter (konsol)

Sebuah versi parameter dapat memiliki maksimum sepuluh label. Prosedur berikut ini menjelaskan cara melihat semua label yang dilampirkan ke suatu versi parameter dengan menggunakan konsol Systems Manager.

Untuk melihat label yang dilampirkan ke versi parameter

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Parameter Store.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Parameter Store.
3. Pilih nama parameter untuk membuka halaman detail untuk parameter tersebut.
4. Pilih tab Riwayat.
5. Cari versi parameter yang Anda ingin lihat semua label terlampir. Kolom Label menunjukkan semua label yang terlampir pada versi parameter.

Memindahkan label parameter (konsol)

Prosedur berikut ini menjelaskan cara memindahkan sebuah label parameter ke versi yang berbeda dari parameter yang sama dengan menggunakan konsol Systems Manager.

Untuk memindahkan label ke versi parameter yang berbeda

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Parameter Store.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Parameter Store.
3. Pilih nama parameter untuk membuka halaman detail untuk parameter tersebut.
4. Pilih tab Riwayat.
5. Pilih versi parameter yang ingin Anda pindahkan labelnya.
6. Pilih Kelola label.
7. Pilih Tambah label baru.
8. Dalam kotak teks, masukkan nama label.
9. Setelah Anda selesai, pilih Simpan perubahan.

Menghapus label parameter (konsol)

Prosedur berikut ini menjelaskan cara menghapus satu atau beberapa label parameter menggunakan konsol Systems Manager.

Untuk menghapus label dari parameter

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Parameter Store.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Parameter Store.

3. Pilih nama parameter untuk membuka halaman detail untuk parameter tersebut.
4. Pilih tab Riwayat.
5. Pilih versi parameter yang ingin Anda hapus labelnya.
6. Pilih Kelola label.
7. Pilih Hapus. di sebelah setiap label yang ingin Anda hapus.
8. Setelah Anda selesai, pilih Simpan perubahan.
9. Konfirmasikan bahwa perubahan Anda sudah benar, masukkan `Confirm` di kotak teks, lalu pilih Konfirmasi.

Bekerja dengan label parameter (AWS CLI)

Bagian ini menjelaskan cara melakukan tugas berikut ini dengan menggunakan AWS Command Line Interface (AWS CLI).

- Membuat label parameter baru (AWS CLI)
- Melihat label untuk sebuah parameter (AWS CLI)
- Melihat daftar parameter yang memiliki suatu label (AWS CLI)
- Memindahkan label parameter (AWS CLI)
- Menghapus label parameter (AWS CLI)

Membuat label parameter baru (AWS CLI)

Prosedur berikut ini menjelaskan cara melampirkan label ke versi tertentu dari parameter yang sudah ada dengan menggunakan AWS CLI. Anda tidak dapat melampirkan label saat membuat parameter.

Untuk membuat label parameter

1. Instal dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaru dariAWS CLI.
2. Jalankan perintah berikut untuk melihat daftar parameter yang Anda memiliki izin untuk melampirkan label.

> **Note**
>
> Parameter hanya tersedia di Wilayah AWS tempat mereka dibuat. Jika Anda tidak melihat parameter yang ingin Anda lampirkan labelnya, maka verifikasi Region Anda.

```
aws ssm describe-parameters
```

Perhatikan nama parameter yang Anda ingin melampirkan label.

3. Jalankan perintah berikut ini untuk melihat semua versi dari parameter tersebut.

```
aws ssm get-parameter-history --name "parameter-name"
```

Perhatikan versi parameter yang Anda ingin melampirkan label.

4. Jalankan perintah berikut ini untuk mengambil informasi tentang parameter berdasarkan nomor versinya.

```
aws ssm get-parameters --names "parameter-name:version-number"
```

Ini contohnya.

```
aws ssm get-parameters --names "/Production/SQLConnectionString:3"
```

5. Jalankan salah satu perintah berikut untuk melampirkan label ke sebuah versi parameter. Jika Anda melampirkan beberapa label, pisahkan nama label dengan spasi.

Melampirkan label ke parameter terbaru

```
aws ssm label-parameter-version --name parameter-name  --labels label-name
```

Melampirkan label ke parameter tertentu

```
aws ssm label-parameter-version --name parameter-name --parameter-version version-
number --labels label-name
```

Berikut ini adalah beberapa contoh.

```
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**
>
> Jika output menunjukkan label yang Anda buat di daftar `InvalidLabels`, maka label tidak memenuhi persyaratan yang dijelaskan sebelumnya dalam topik ini. Tinjau persyaratan dan coba lagi. Jika daftar `InvalidLabels` kosong, maka label Anda berhasil diterapkan ke versi parameter.

6. Anda dapat melihat detail parameter dengan menggunakan nomor versi atau nama label. Jalankan perintah berikut ini dan tentukan label yang Anda buat dalam langkah sebelumnya.

```
aws ssm get-parameter --name parameter-name:label-name --with-decryption
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
{
    "Parameter": {
        "Version": version-number,
        "Type": "parameter-type",
        "Name": "parameter-name",
        "Value": "parameter-value",
        "Selector": ":label-name"
    }
}
```

> **Note**
>
> Pemilih dalam output adalah nomor versi atau label yang Anda tentukan dalam bidang input `Name`.

Melihat label untuk sebuah parameter (AWS CLI)

Anda dapat menggunakan operasi GetParameterHistoryAPI untuk melihat riwayat lengkap dan semua label yang terlampir pada parameter tertentu. Atau, Anda dapat menggunakan operasi GetParametersByPathAPI untuk melihat daftar semua parameter yang memiliki label tertentu.

Untuk melihat label untuk parameter dengan menggunakan operasi GetParameterHistory API

1. Jalankan perintah berikut ini untuk melihat daftar parameter yang dapat Anda lihat labelnya.

   > **Note**
   >
   > Parameter hanya tersedia di Region tempat mereka dibuat. Jika Anda tidak melihat parameter yang ingin Anda pindahkan labelnya, maka verifikasi Region Anda.

   ```
   aws ssm describe-parameters
   ```

   Perhatikan nama parameter yang ingin Anda lihat labelnya.

2. Jalankan perintah berikut ini untuk melihat semua versi dari parameter tersebut.

   ```
   aws ssm get-parameter-history --name parameter-name --with-decryption
   ```

   Sistem mengembalikan informasi seperti berikut ini.

   ```
   {
       "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"
        }
    ]
}
```

Melihat daftar parameter yang memiliki suatu label (AWS CLI)

Anda dapat menggunakan operasi GetParametersByPathAPI untuk melihat daftar semua parameter di suatu jalur yang memiliki label tertentu.

Jalankan perintah berikut ini untuk melihat daftar parameter di suatu jalur yang memiliki label tertentu. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

```
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
```

Sistem mengembalikan informasi seperti berikut ini. Untuk contoh ini, pengguna mencari pada`/Config` jalur.

```
{
    "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"
        }
    ]
}
```

Memindahkan label parameter (AWS CLI)

Prosedur berikut ini menjelaskan cara memindahkan sebuah label parameter ke versi yang berbeda dari parameter yang sama.

Untuk memindahkan label parameter

1. Jalankan perintah berikut ini untuk melihat semua versi dari parameter tersebut. Ganti *nama parameter* dengan informasi Anda sendiri.

   ```
   aws ssm get-parameter-history \
       --name "parameter name"
   ```

   Perhatikan versi parameter yang ingin Anda pindahkan labelnya yaitu sumber dan target.

2. Jalankan perintah berikut ini untuk menetapkan label yang ada untuk versi yang berbeda dari sebuah parameter. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

   ```
   aws ssm label-parameter-version \
       --name parameter name \
       --parameter-version version number \
       --labels name-of-existing-label
   ```

   > **Note**
   >
   > Jika Anda ingin memindahkan label yang ada ke versi parameter terbaru, maka hapus `--parameter-version` dari perintah.

Menghapus label parameter (AWS CLI)

Prosedur berikut ini menjelaskan cara menghapus label parameter menggunakan AWS CLI.

Untuk menghapus label parameter

1. Jalankan perintah berikut ini untuk melihat semua versi dari parameter tersebut. Ganti *nama parameter* dengan informasi Anda sendiri.

```
aws ssm get-parameter-history \
    --name "parameter name"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "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"
        }
    ]
}
```

Perhatikan versi parameter yang Anda ingin hapus labelnya.

2. Jalankan perintah berikut ini untuk menghapus label yang Anda pilih dari parameter tersebut. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri.

```
aws ssm unlabel-parameter-version \
    --name parameter name \
    --parameter-version version \
    --labels label 1,label 2,label 3
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "InvalidLabels": ["invalid"],
    "DeletedLabels" : ["Prod"]
 }
```

Bekerja dengan versi parameter

Setiap kali Anda mengubah nilai parameterParameter Store, suatu kemampuan akanAWS Systems Manager membuat versi baru dari suatu parameter dan mempertahankan versi sebelumnya. Ketika Anda awalnya membuat parameter,Parameter Store menetapkan versi1 ke parameter itu. Ketika Anda mengubah nilai parameter tersebut,Parameter Store secara otomatis menambah nomor versi dengan satu. Anda dapat melihat detail, termasuk nilai-nilai, dari semua versi dalam riwayat parameter.

Anda juga dapat menentukan versi parameter untuk digunakan dalam perintah API dan dokumen SSM; misalnya: `ssm:MyParameter:3`. Anda dapat menentukan nama parameter dan nomor versi tertentu dalam panggilan API dan dokumen SSM. Jika Anda tidak menentukan nomor versi, sistem akan secara otomatis menggunakan versi terbaru. Jika Anda menentukan nomor untuk versi yang tidak ada, sistem mengembalikan kesalahan daripada jatuh kembali ke versi terbaru atau default parameter.

Anda dapat menggunakan versi parameter untuk melihat berapa kali parameter berubah selama periode waktu tertentu. Versi parameter juga menyediakan lapisan perlindungan jika nilai parameter secara tidak sengaja berubah.

Anda dapat membuat dan mempertahankan hingga 100 versi parameter. Setelah Anda membuat 100 versi parameter, setiap kali Anda membuat versi baru, versi tertua dari parameter dihapus dari riwayat untuk membuat ruang untuk versi baru.

Pengecualian untuk ini adalah ketika sudah ada 100 versi parameter dalam riwayat, dan label parameter ditetapkan ke versi tertua dari parameter. Dalam kasus ini, versi tersebut tidak dihapus dari riwayat, dan permintaan untuk membuat versi parameter baru gagal. Perlindungan ini adalah untuk mencegah terhapusnya versi parameter dengan label bermisi kritis yang ditetapkan kepada mereka. Untuk terus membuat parameter baru, pertama-tama pindahkan label dari versi tertua parameter ke parameter yang lebih baru untuk digunakan dalam operasi Anda. Untuk informasi tentang memindahkan label parameter, lihat Memindahkan label parameter (konsol) dan Memindahkan label parameter (AWS CLI).

Prosedur berikut menunjukkan cara mengubah parameter dan kemudian memverifikasi bahwa Anda membuat versi baru. Anda dapat menggunakan perintah `get-parameter` dan `get-parameters` untuk melihat versi parameter. Untuk contoh penggunaan perintah ini, lihat GetParameterdan GetParametersdi ReferensiAWS Systems Manager API

Topik

- Membuat versi baru dari suatu parameter (konsol)
- Mereferensikan versi parameter

Membuat versi baru dari suatu parameter (konsol)

Anda dapat menggunakan konsol Systems Manager untuk membuat versi baru dari parameter dan melihat riwayat versi sebuah parameter.

Membuat versi baru dari suatu parameter

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Parameter Store.

   -atau-

JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Parameter Store.

3. Pilih nama parameter yang Anda buat sebelumnya. Untuk informasi lebih lanjut tentang membuat parameter baru, lihat Menandai parameter Systems Manager.
4. Pilih Edit.
5. Di kotak Nilai, masukkan nilai baru, lalu pilih Simpan perubahan.
6. Pilih nama parameter yang baru saja Anda perbarui. Pada tab Gambaran Umum, verifikasi bahwa nomor versi bertambah 1, dan verifikasi nilai yang baru.
7. Untuk melihat riwayat semua versi parameter, pilih tab Riwayat.

Mereferensikan versi parameter

Anda dapat mereferensi versi parameter tertentu dalam perintah, panggilan API, dan dokumen SSM dengan menggunakan format berikut: ssm:*parameter-name*:*version-number*.

Pada contoh berikut, `run-instances command` Amazon Elastic Compute Cloud (Amazon EC2) menggunakan versi 3 dari parameter `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**
>
> Menggunakan `resolve` dan nilai parameter hanya bekerja dengan opsi `--image-id` dan parameter yang berisi Amazon Machine Image (AMI) sebagai nilainya. Untuk informasi selengkapnya, lihat Dukungan parameter native untuk ID Amazon Machine Image.

Berikut ini adalah contoh untuk menentukan versi 2 dari parameter bernama `MyRunCommandParameter` dalam dokumen 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}}"
                ]
            }
        }
    ]
}
```

Bekerja dengan parameter bersama

Berbagi parameter lanjutan menyederhanakan manajemen data konfigurasi dalam lingkungan multi-akun. Anda dapat menyimpan dan mengelola parameter Anda secara terpusat dan membagikannya dengan orang lain Akun AWS yang perlu mereferensikannya.

Parameter Storeterintegrasi dengan AWS Resource Access Manager (AWS RAM) untuk mengaktifkan berbagi parameter lanjutan. AWS RAM adalah layanan yang memungkinkan Anda untuk berbagi sumber daya dengan orang lain Akun AWS atau melalui AWS Organizations.

Dengan AWS RAM, Anda berbagi sumber daya yang Anda miliki dengan membuat pembagian sumber daya. Pembagian sumber daya menentukan sumber daya untuk dibagikan, izin untuk diberikan, dan konsumen yang akan dibagikan. Konsumen dapat mencakup:

- Khusus Akun AWS di dalam atau di luar organisasinya di AWS Organizations
- Unit organisasi di dalam organisasinya di AWS Organizations
- Seluruh organisasinya di AWS Organizations

Untuk informasi selengkapnya AWS RAM, lihat Panduan AWS RAM Pengguna.

Topik ini menjelaskan cara berbagi parameter yang Anda miliki, dan cara menggunakan parameter yang dibagikan dengan Anda.

Daftar Isi

- Berhenti berbagi parameter bersama
- Mengidentifikasi parameter bersama
- Mengakses parameter bersama
- Set izin untuk berbagi parameter
- Throughput maksimum untuk parameter bersama
- Harga untuk parameter bersama
- Akses lintas akun untuk ditutup Akun AWS

Prasyarat untuk berbagi parameter

Prasyarat berikut harus dipenuhi sebelum Anda dapat berbagi parameter dari akun Anda:

- Untuk berbagi parameter, Anda harus memilikinya di Akun AWS. Anda tidak dapat membagikan parameter yang telah dibagikan dengan Anda.
- Untuk berbagi parameter, itu harus dalam tingkat parameter lanjutan. Untuk informasi tentang tingkatan parameter, lihatMengelola tingkatan parameter. Untuk informasi tentang mengubah parameter standar yang ada ke parameter lanjutan, lihatMengubah parameter standar ke parameter lanjutan.
- Untuk berbagi `SecureString` parameter, itu harus dienkripsi dengan kunci yang dikelola pelanggan, dan Anda harus membagikan kunci secara terpisah. AWS Key Management Service Kunci yang dikelola AWS tidak dapat dibagikan. Parameter yang dienkripsi dengan default Kunci yang dikelola AWS dapat diperbarui untuk menggunakan kunci yang dikelola pelanggan sebagai gantinya. Untuk definisi AWS KMS kunci, lihat AWS KMS konsep di Panduan AWS Key Management Service Pengembang.
- Untuk berbagi parameter dengan organisasi Anda atau unit organisasi di AWS Organizations, Anda harus mengaktifkan berbagi dengan AWS Organizations. Untuk informasi selengkapnya, lihat Aktifkan Berbagi dengan AWS Organizations dalam Panduan Pengguna AWS RAM .

Berbagi parameter

Untuk berbagi parameter, Anda harus menambahkannya ke berbagi sumber daya. Pembagian sumber daya adalah AWS RAM sumber daya yang memungkinkan Anda berbagi sumber daya Anda Akun AWS. Pembagian sumber daya menentukan sumber daya yang akan dibagikan, dan konsumen yang akan berbagi dengan mereka.

Ketika Anda berbagi parameter yang Anda miliki dengan yang lain Akun AWS, Anda dapat memilih dari dua izin AWS terkelola untuk diberikan kepada konsumen. Untuk informasi selengkapnya, lihat Set izin untuk berbagi parameter.

Jika Anda adalah bagian dari organisasi AWS Organizations dan berbagi dalam organisasi Anda diaktifkan, Anda dapat memberikan konsumen di organisasi Anda akses dari AWS RAM konsol ke parameter bersama. Jika tidak, konsumen menerima undangan untuk bergabung dengan pembagian sumber daya dan diberikan akses ke parameter bersama setelah menerima undangan.

Anda dapat membagikan parameter yang Anda miliki menggunakan AWS RAM konsol, atau AWS CLI.

> **Note**
>
> Meskipun Anda dapat membagikan parameter menggunakan operasi Systems Manager PutResourcePolicyAPI, sebaiknya gunakan AWS Resource Access Manager (AWS RAM) sebagai gantinya. Ini karena penggunaan `PutResourcePolicy` memerlukan langkah ekstra untuk mempromosikan parameter ke Resource Share standar menggunakan operasi AWS RAM PromoteResourceShareCreatedFromPolicyAPI. Jika tidak, parameter tidak akan dikembalikan oleh operasi Systems Manager DescribeParametersAPI menggunakan `--shared` opsi.

Untuk berbagi parameter yang Anda miliki menggunakan AWS RAM konsol

Lihat Membuat Sumber Daya Bersama di Panduan Pengguna AWS RAM .

Untuk berbagi parameter yang Anda miliki menggunakan AWS CLI

Gunakan create-resource-shareperintah untuk menambahkan parameter ke berbagi sumber daya baru.

Gunakan associate-resource-shareperintah untuk menambahkan parameter ke pembagian sumber daya yang ada.

Contoh berikut membuat pembagian sumber daya baru untuk berbagi parameter dengan konsumen dalam suatu organisasi dan dalam akun individu.

```
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"
```

Berhenti berbagi parameter bersama

Ketika Anda berhenti berbagi parameter bersama, akun konsumen tidak dapat lagi mengakses parameter.

Untuk berhenti berbagi parameter yang Anda miliki, Anda harus menghapusnya dari pembagian sumber daya. Anda dapat melakukan ini menggunakan Systems Manager konsol, AWS RAM konsol, atau AWS CLI.

Untuk berhenti berbagi parameter yang Anda miliki menggunakan AWS RAM konsol

Lihat Memperbarui bagian sumber daya AWS RAM di Panduan AWS RAM Pengguna.

Untuk berhenti berbagi parameter yang Anda miliki menggunakan AWS CLI

Gunakan perintah disassociate-resource-share.

Mengidentifikasi parameter bersama

Pemilik dan konsumen dapat mengidentifikasi parameter bersama menggunakan AWS CLI.

Untuk mengidentifikasi parameter bersama menggunakan AWS CLI

Untuk mengidentifikasi parameter bersama menggunakan AWS CLI, Anda dapat memilih dari `describe-parameters` perintah Systems Manager dan AWS RAM `list-resources` perintah.

Saat Anda menggunakan `--shared` opsi dengan`describe-parameters`, perintah mengembalikan parameter yang dibagikan dengan Anda.

Berikut ini adalah contohnya:

```
aws ssm describe-parameters --shared
```

Mengakses parameter bersama

Konsumen dapat mengakses parameter bersama menggunakan alat baris AWS perintah, dan AWS SDK. Untuk akun konsumen, parameter yang dibagikan dengan akun tersebut tidak disertakan dalam halaman Parameter saya.

Contoh CLI: Mengakses detail parameter bersama menggunakan AWS CLI

Untuk mengakses detail parameter bersama menggunakan AWS CLI, Anda dapat menggunakan get-parametersperintah get-parameteratau. Anda harus menentukan parameter penuh ARN sebagai untuk mengambil parameter dari akun lain. `--name`

Berikut sebuah contoh.

```
aws ssm get-parameter \
    --name arn:aws:ssm:us-east-2:123456789012:parameter/MySharedParameter
```

Integrasi yang didukung dan tidak didukung untuk parameter bersama

Saat ini, Anda dapat menggunakan parameter bersama dalam skenario integrasi berikut:

- AWS CloudFormation parameter template
- AWS Parameter dan Rahasia ekstensi Lambda
- Templat peluncuran Amazon Elastic Compute Cloud (EC2)
- Nilai `ImageID` dengan RunInstances perintah EC2 untuk membuat instance dari Amazon Machine Image (AMI)
- Parameter dalam perintah diRun Command, kemampuan Systems Manager
- Mengambil nilai parameter dalam runbook untuk Otomasi, kemampuan Systems Manager

Skenario berikut dan layanan terintegrasi saat ini tidak mendukung penggunaan parameter bersama:

- AWS CloudFormation referensi dinamis
- Nilai variabel lingkungan di AWS CodeBuild
- Nilai variabel lingkungan di AWS App Runner
- Nilai rahasia di Amazon Elastic Container Service

Set izin untuk berbagi parameter

Akun konsumen Anda menerima akses hanya-baca ke parameter bersama Anda. Konsumen tidak dapat memperbarui atau menghapus parameter. Konsumen tidak dapat berbagi parameter dengan akun ketiga. Anda dapat memilih dari dua set izin AWS terkelola untuk memberikan akses hanya-baca ini:

AWSRAMDefaultPermissionSSMParameterReadOnly

Tindakan yang diizinkan:`DescribeParameters`,`GetParameter`, `GetParameters`

AWSRAMPermissionSSMParameterReadOnlyWithHistory

Tindakan yang diizinkan:`DescribeParameters`,`GetParameter`,`GetParameters`, `GetParameterHistory`

Throughput maksimum untuk parameter bersama

Systems Manager membatasi throughput maksimum (transaksi per detik) untuk GetParameterdan GetParameters. operasi. Throughput diberlakukan pada tingkat akun individu. Oleh karena itu, setiap akun yang menggunakan parameter bersama dapat menggunakan throughput maksimum yang diizinkan tanpa terpengaruh oleh akun lain. Untuk informasi selengkapnya tentang throughput maksimum untuk parameter, lihat topik berikut:

- Meningkatkan Parameter Store throughput
- Kuota Layanan Systems Manager di. Referensi Umum Amazon Web

Harga untuk parameter bersama

Berbagi lintas akun hanya tersedia di tingkat parameter lanjutan. Untuk parameter lanjutan, biaya dikenakan pada harga saat ini untuk penyimpanan dan penggunaan API untuk setiap parameter lanjutan. Akun pemilik dibebankan untuk penyimpanan parameter lanjutan. Akun konsumsi apa pun yang melakukan panggilan API ke parameter lanjutan bersama dikenakan biaya untuk penggunaan parameter.

Misalnya, jika Akun A membuat parameter lanjutan`MyAdvancedParameter`, akun tersebut dikenakan biaya USD 0,05 per bulan untuk menyimpan parameter tersebut.

Akun A kemudian dibagikan `MyAdvancedParameter` dengan Akun B dan Akun C. Selama sebulan, ketiga akun melakukan panggilan ke`MyAdvancedParameter`. Tabel berikut menggambarkan biaya yang akan mereka keluarkan untuk jumlah panggilan yang dilakukan masing-masing.

> **Note**
>
> Biaya dalam tabel berikut hanya untuk ilustrasi. Untuk memverifikasi harga saat ini, lihat AWS Systems Manager Harga untuk Parameter Store.

Akun	Jumlah panggilan	Biaya
Akun A (memiliki akun)	10.000 panggilan	• Penyimpanan parameter lanjutan satu bulan: USD 0,05 • 10.000 panggilan ke`MyAdvancedParameter` : USD 0,05 • Total: USD 0,10
Akun B (akun konsumsi)	20.000 panggilan	• 20.000 panggilan ke`MyAdvancedParameter` : USD 0,10 • Total: USD 0,10
Akun C (akun konsumsi)	30.000 panggilan	• 30.000 panggilan ke`MyAdvancedParameter` : USD 0,15 • Total: USD 0,15

Akses lintas akun untuk ditutup Akun AWS

Jika Akun AWS yang memiliki parameter bersama ditutup, semua akun yang mengkonsumsi kehilangan akses ke parameter bersama. Jika akun pemilik dibuka kembali dalam waktu 90 hari setelah akun ditutup, akun yang dikonsumsi mendapatkan kembali akses ke parameter yang dibagikan sebelumnya. Untuk informasi selengkapnya tentang membuka kembali akun selama Periode Pasca-Penutupan, lihat Mengakses akun Anda Akun AWS setelah Anda menutupnya di Panduan Referensi. AWS Account Management

Bekerja dengan parameter menggunakan Run Command perintah

Anda dapat bekerja dengan parameter diRun Command, kemampuanAWS Systems Manager. Untuk informasi selengkapnya, lihat AWS Systems Manager Run Command.

Menjalankan parameter String (konsol)

Prosedur berikut memandu Anda melalui proses menjalankan perintah yang menggunakan parameter `String`.

Untuk menjalankan parameter String menggunakan Parameter Store

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Run Command.
3. Pilih Jalankan perintah.
4. Di daftar Dokumen perintah, pilih `AWS-RunPowerShellScript` (Windows) atau `AWS-RunShellScript` (Linux).
5. Untuk Parameter perintah, masukkan **`echo {{ssm:parameter-name}}`**. Sebagai contoh: **`echo {{ssm:/Test/helloWorld}}`**.
6. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

   > **Tip**
   >
   > Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.
7. Untuk Parameter lainnya:

   - Untuk Komentar, ketik informasi tentang perintah ini.
   - Untuk Waktu habis (detik), tentukan jumlah detik untuk menunggu sistem sebelum gagal menjalankan perintah keseluruhan.
8. Untuk Pengendalian rate:

   - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

> **Note**
>
> Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

- Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

9. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan milik pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbedaAkun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

10. Di bagian Notifikasi SNS, jika Anda ingin notifikasi dikirim tentang status eksekusi perintah, pilih kotak centang Aktifkan notifikasi SNS.

    Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Run Command Amazon SNS, lihat. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

11. Pilih Jalankan.

12. Di halaman Command ID, di area Target dan output, pilih tombol di sebelah ID node tempat Anda menjalankan perintah, lalu pilih Lihat output. Verifikasi bahwa output dari perintah tersebut adalah nilai yang Anda berikan untuk parameter, seperti **`This is my first parameter`**.

Menjalankan parameter (AWS CLI)

Contoh 1: Perintah sederhana

Contoh perintah berikut mencakup parameter Systems Manager bernama `DNS-IP`. Nilai parameter ini hanyalah alamat IP dari sebuah node. Contoh ini menggunakan perintah AWS Command Line Interface (AWS CLI) untuk menggemakan nilai parameter.

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
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
{
    "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": []
    }
}
```

Setelah eksekusi perintah selesai, Anda dapat melihat informasi lebih lanjut tentang hal itu menggunakan perintah berikut:

- get-command-invocation— Lihat informasi rinci tentang eksekusi perintah.
- list-command-invocations— Lihat status eksekusi perintah pada node terkelola tertentu.
- list-commands— Lihat status eksekusi perintah di seluruh node terkelola.

Contoh 2: Dekripsi nilai parameter **`SecureString`**

Contoh perintah berikutnya menggunakan `SecureString` parameter bernama SecurePassword. Perintah yang digunakan di `parameters` lapangan mengambil dan mendekripsi nilai `SecureString` parameter, dan kemudian mengatur ulang kata sandi administrator lokal tanpa harus melewatkan kata sandi dalam teks yang jelas.

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
```

Contoh 3: Referensi parameter dalam dokumen SSM

Anda juga dapat mereferensi parameter Systems Manager di bagian Parameter pada dokumen SSM, seperti yang ditunjukkan dalam contoh berikut.

```
{
   "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}}"
          }
       }
    ]
}
```

Jangan bingung sintaks serupa untuk parameter lokal yang digunakan di `runtimeConfig` bagian dokumen SSM dengan parameter. Parameter Store Parameter lokal tidak sama dengan parameter Systems Manager. Anda dapat membedakan parameter lokal dari parameter Systems Manager dengan tidak adanya prefiks `ssm:`.

```
"runtimeConfig":{
        "aws:runShellScript":{
            "properties":[
                {
                    "id":"0.aws:runShellScript",
                    "runCommand":"{{ commands }}",
                    "workingDirectory":"{{ workingDirectory }}",
                    "timeoutSeconds":"{{ executionTimeout }}"
```

Note

Dokumen SSM tidak mendukung referensi ke parameter `SecureString`. Ini berarti bahwa untuk menggunakan `SecureString` parameter dengan, misalnyaRun Command, Anda harus mengambil nilai parameter sebelum meneruskannyaRun Command, seperti yang ditunjukkan pada contoh berikut.

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
```

Dukungan parameter native untuk ID Amazon Machine Image

Saat Anda membuat parameter `String`, Anda dapat menentukan tipe data sebagai `aws:ec2:image` untuk memastikan bahwa nilai parameter yang Anda masukkan adalah format ID Amazon Machine Image (AMI) yang valid.

Support untuk format ID AMI memungkinkan Anda untuk menghindari memperbarui semua script dan templat Anda dengan ID baru setiap kali AMI yang ingin Anda gunakan dalam proses berubah. Anda dapat membuat parameter dengan tipe data `aws:ec2:image`, dan untuk nilainya, masukkan ID AMI. Ini adalah AMI yang ingin Anda gunakan untuk membuat instans baru. Anda kemudian mereferensi parameter ini dalam templat, perintah, dan script Anda.

Misalnya, Anda dapat menentukan parameter yang berisi pilihan ID AMI Anda ketika menjalankan perintah `run-instances` Amazon Elastic Compute Cloud (Amazon EC2).

> **Note**
>
> Pengguna yang menjalankan perintah ini harus memiliki izin AWS Identity and Access Management (IAM) yang menyertakan operasi `ssm:GetParameters` API agar nilai parameter divalidasi. Jika tidak, proses pembuatan parameter gagal.

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
```

Anda juga dapat memilih AMI yang Anda sukai ketika membuat instans menggunakan konsol Amazon EC2. Untuk informasi lebih lanjut, lihat Menggunakan parameter Systems Manager untuk menemukanAMI di Panduan Pengguna Amazon EC2 untuk Instans Windows.

Ketika saatnya untuk menggunakan yang berbeda AMI dalam alur kerja pembuatan instans Anda, Anda hanya perlu memperbarui parameter dengan AMI nilai baru, dan Parameter Store sekali lagi memvalidasi bahwa Anda telah memasukkan ID dalam format yang tepat.

Berikan izin untuk membuat parameter tipe **`aws:ec2:image`** data

Dengan menggunakan kebijakan AWS Identity and Access Management (IAM), Anda dapat menyediakan atau membatasi akses pengguna ke operasi dan Parameter Store konten API.

Untuk membuat parameter tipe `aws:ec2:image` data, pengguna harus memiliki keduanya `ssm:PutParameter` dan `ec2:DescribeImages` izin.

Contoh kebijakan berikut ini memberikan izin pengguna untuk memanggil operasi API `PutParameter` untuk `aws:ec2:image`. Ini berarti bahwa pengguna dapat menambahkan parameter tipe data `aws:ec2:image` ke sistem.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ssm:PutParameter",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DescribeImages",
            "Resource": "*"
        }
    ]
}
```

Cara kerja validasi format AMI

Saat Anda menentukan `aws:ec2:image` sebagai tipe data untuk sebuah parameter, Systems Manager tidak segera membuat parameter. Sebagai gantinya, akan dilakukan operasi validasi asinkron untuk memastikan bahwa nilai parameter memenuhi persyaratan format untuk ID AMI, dan bahwa AMI yang ditentukan tersedia di Akun AWS Anda.

Nomor versi parameter mungkin dihasilkan sebelum operasi validasi selesai. Operasi ini mungkin belum selesai bahkan jika nomor versi parameter telah dihasilkan.

Untuk memantau apakah parameter Anda berhasil dibuat, sebaiknya gunakan Amazon EventBridge untuk mengirimi Anda pemberitahuan tentang operasi Anda `create` dan `update` parameter. Notifikasi ini melaporkan apakah operasi parameter berhasil atau tidak. Jika operasi gagal, notifikasi menyertakan pesan kesalahan yang menjelaskan alasan kegagalan.

```
{
    "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"
    }
}
```

Untuk informasi tentang berlangganan Parameter Store acara di EventBridge, lihatMenyiapkan notifikasi atau memicu tindakan berdasarkan Parameter Store peristiwa.

Menghapus parameter Systems Manager

Topik ini menjelaskan cara menghapus parameter yang telah Anda buatParameter Store, kemampuan AWS Systems Manager.

Untuk menghapus parameter (konsol)

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Parameter Store.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Parameter Store.

3. Pada tab Parameter saya, pilih kotak centang di sebelah setiap parameter yang akan dihapus.
4. Pilih Hapus.
5. Pada dialog konfirmasi, pilih Hapus parameter.

Untuk menghapus parameter (AWS CLI)

- Jalankan perintah berikut:

```
aws ssm delete-parameter --name "my-parameter"
```

Ganti parameter saya dengan nama parameter Anda yang akan dihapus.

Untuk informasi tentang semua opsi yang tersedia untuk digunakan dengan `delete-parameter` perintah, lihat delete-parameterdi AWS Systems Manager bagian Referensi AWS CLI Perintah.

Menggunakan dengan parameter publik

Beberapa Layanan AWS mempublikasikan informasi tentang artefak umum sebagai parameter AWS Systems Manager publik. Misalnya, layanan Amazon Elastic Compute Cloud (Amazon EC2) menerbitkan informasi tentang Amazon Machine Images (AMIs) sebagai parameter publik.

Anda dapat memanggil informasi ini dari skrip dan kode Anda dengan menggunakan operasi DescribeParameters, GetParametersByPath, GetParameter, dan GetParametersAPI.

Info lebih lanjut

- Kueri untuk Wilayah AWS, Titik Akhir, dan Lainnya Menggunakan AWS Systems ManagerParameter Store
- Kueri untuk AMI ID Amazon Linux terbaru yang menggunakan AWS Systems ManagerParameter Store
- Kueri untuk Windows Terbaru AMI Menggunakan AWS Systems ManagerParameter Store

Topik

Menemukan parameter publik

Anda dapat mencari parameter publik menggunakan Parameter Store konsol atau AWS Command Line Interface. Nama parameter publik diawali dengan `aws/service/list`. Bagian selanjutnya dari nama sesuai dengan layanan yang memiliki parameter tersebut.

Berikut ini adalah daftar beberapa layanan yang menyediakan parameter publik:

- `ami-amazon-linux-latest`
- `ami-windows-latest`
- `appmesh`
- `aws-for-fluent-bit`
- `bottlerocket`
- `canonical`
- `cloud9`
- `datasync`
- `debian`
- `ecs`
- `eks`
- `freebsd`
- `global-infrastructure`
- `marketplace`
- `storagegateway`

Semua parameter publik tidak dipublikasikan ke semua Wilayah AWS.

Menemukan parameter publik menggunakan AWS CLI

Gunakan `describe-parameters` untuk penemuan parameter publik. Gunakan `get-parameters-by-path` untuk mendapatkan jalur yang sebenarnya untuk layanan yang tercantum di bawah `/aws/service/list`. Untuk mendapatkan jalur layanan, hapus `/list` dari jalur tersebut. Misalnya, `/aws/service/list/ecs` menjadi `/aws/service/ecs`.

Untuk mengambil daftar parameter publik yang dimiliki oleh layanan yang berbeda diParameter Store, jalankan perintah berikut.

```
aws ssm get-parameters-by-path --path /aws/service/list
```

Perintah tersebut mengembalikan informasi seperti berikut. Contoh ini telah dipotong untuk ruang.

```
{
    "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"
        }
    ]
}
```

Jika Anda ingin melihat parameter yang dimiliki oleh layanan tertentu, pilih layanan dari daftar yang dihasilkan setelah menjalankan perintah sebelumnya. Kemudian, lakukan `get-parameters-by-path` panggilan menggunakan nama layanan yang Anda inginkan. Misalnya, `/aws/service/global-infrastructure`. Jalur mungkin satu tingkat (hanya memanggil parameter yang cocok dengan nilai persis yang diberikan) atau rekursif (berisi elemen di jalur di luar apa yang telah Anda berikan). Jika tidak ada hasil yang dikembalikan untuk layanan yang Anda tentukan, tambahkan `--recursive` bendera dan jalankan perintah lagi.

```
aws ssm get-parameters-by-path --path /aws/service/global-infrastructure
```

Hal ini mengembalikan semua parameter yang dimiliki oleh `global-infrastructure`.

```
{
    "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"
        }
    ]
}
```

Anda juga dapat melihat parameter yang dimiliki oleh layanan tertentu dengan menggunakan filter `Option:BeginsWith`.

```
aws ssm describe-parameters --parameter-filters "Key=Name, Option=BeginsWith, Values=/
aws/service/ami-amazon-linux-latest"
```

Perintah tersebut mengembalikan informasi seperti berikut. Contoh output ini telah dipotong untuk ruang.

```
{
    "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**
>
> Parameter yang dikembalikan mungkin berbeda ketika Anda menggunakan `Option=BeginsWith` karena menggunakan pola pencarian yang berbeda.

Menemukan parameter publik menggunakan Parameter Store konsol

Anda harus memiliki setidaknya satu parameter di Anda Akun AWS dan Wilayah AWS sebelum Anda dapat mencari parameter publik menggunakan konsol.

Untuk menemukan parameter publik menggunakan konsol

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Parameter Store.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Parameter Store.
3. Pilih tab Parameter publik.
4. Pilih dropdown Pilih layanan. Pilih layanan yang parameternya ingin Anda gunakan.
5. Filter parameter yang dimiliki oleh layanan yang Anda pilih dengan memasukkan lebih banyak informasi ke dalam bilah pencarian.
6. Pilih parameter publik yang ingin Anda gunakan.

Memanggil parameter publik AMI

Parameter publik Amazon Elastic Compute Cloud (Amazon Amazon Machine Image EC2) AMI () tersedia untuk Amazon Linux 1, Amazon Linux 2, Windows Server dan dari jalur berikut:

- Amazon Linux 1 dan Amazon Linux 2: `/aws/service/ami-amazon-linux-latest`
- Windows Server: `/aws/service/ami-windows-latest`

Memanggil parameter AMI publik untuk Amazon Linux 1, Amazon Linux 2, dan Amazon Linux 2023

Anda dapat melihat daftar semua Amazon Linux 1, Amazon Linux 2, dan Amazon Linux 2023 (AL2023) AMIs saat ini Wilayah AWS dengan menggunakan perintah berikut di (). 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
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
[
    "/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"
]
```

Anda dapat melihat detail tentang AMIs ini, termasuk ID AMI dan Amazon Resource Name (ARN), dengan menggunakan perintah berikut.

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
```

wilayah mewakili pengenal untuk Wilayah AWS didukung oleh AWS Systems Manager, seperti `us-east-2` untuk Wilayah Timur AS (Ohio). Untuk daftar nilai *wilayah* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

Perintah tersebut mengembalikan informasi seperti berikut. Contoh output ini telah dipotong untuk ruang.

```
{
    "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"
        }
    ]
}
```

Anda dapat melihat detail spesifik AMI dengan menggunakan operasi GetParametersAPI dengan AMI nama lengkap, termasuk jalurnya. Berikut ini adalah contoh perintah.

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
```

Perintah tersebut mengembalikan informasi berikut ini.

```
{
    "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": []
}
```

Memanggil parameter publik AMI untuk Windows Server

Anda dapat melihat daftar semua Windows Server AMIs di saat ini Wilayah AWS dengan menggunakan perintah berikut di 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
```

Perintah tersebut mengembalikan informasi seperti berikut. Contoh output ini telah dipotong untuk ruang.

```
[
    "/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
]
```

Anda dapat melihat detail tentang AMIs ini, termasuk ID AMI dan Amazon Resource Name (ARN), dengan menggunakan perintah berikut.

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
```

wilayah mewakili pengenal untuk Wilayah AWS didukung oleh AWS Systems Manager, seperti us-east-2 untuk Wilayah Timur AS (Ohio). Untuk daftar nilai *wilayah* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum Amazon Web Services.

Perintah tersebut mengembalikan informasi seperti berikut. Contoh output ini telah dipotong untuk ruang.

```
{
    "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"
        }
    ]
}
```

Anda dapat melihat detail spesifik AMI dengan menggunakan operasi GetParametersAPI dengan AMI nama lengkap, termasuk jalurnya. Berikut ini adalah contoh perintah.

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
```

Perintah tersebut mengembalikan informasi berikut ini.

```
{
    "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": []
}
```

Memanggil parameter publik AMI ECS yang dioptimalkan

Layanan Amazon Elastic Container Service (Amazon ECS) menerbitkan nama Amazon ECS terbaru yang Amazon Machine Images dioptimalkan AMIs () sebagai parameter publik. Pengguna dianjurkan untuk menggunakan ini AMI saat membuat cluster Amazon Elastic Compute Cloud (Amazon EC2) baru untuk Amazon ECS karena yang dioptimalkan mencakup perbaikan bug dan pembaruan AMIs fitur.

Gunakan perintah berikut untuk melihat nama Amazon ECS terbaru yang dioptimalkan AMI untuk Amazon Linux 2. Untuk melihat perintah untuk sistem operasi lain, lihat Mengambil metadata AMI Amazon ECS yang dioptimalkan dalam Panduan Developer 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
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
{
    "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": []
}
```

Memanggil parameter publik AMI EKS yang dioptimalkan

Layanan Amazon Elastic Kubernetes Service (Amazon EKS) menerbitkan nama Amazon Machine Image (AMI) Amazon EKS terbaru yang dioptimalkan sebagai parameter publik. Kami menganjurkan Anda untuk menggunakan AMI ini saat menambahkan simpul ke klaster Amazon EKS, karena rilis terbaru mencakup patch Kubernetes dan pembaruan keamanan. Sebelumnya, untuk menjamin Anda menggunakan AMI yang terbaru artinya memeriksa dokumentasi Amazon EKS dan secara manual memperbarui templat deployment atau sumber daya dengan ID AMI baru.

Gunakan perintah berikut untuk melihat nama Amazon EKS terbaru yang dioptimalkan AMI untuk 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
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
{
```

```
    "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": []
}
```

Memanggil parameter publik untuk Layanan AWS, Wilayah, titik akhir, Availability Zone, zona lokal, dan Wavelength Zones

Anda dapat memanggil Wilayah AWS, service, endpoint, Availability, dan Wavelength Zones dari parameter publik dengan menggunakan jalur berikut.

`/aws/service/global-infrastructure`

Lihat aktif Wilayah AWS

Anda dapat melihat daftar semua aktif Wilayah AWS dengan menggunakan perintah berikut di 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
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
[
    "/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"
]
```

Lihat tersedia Layanan AWS

Anda dapat melihat daftar lengkap semua yang tersedia Layanan AWS dan mengurutkannya ke dalam urutan abjad dengan menggunakan perintah berikut. Contoh output ini telah dipotong untuk ruang.

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(@)"
```

Perintah tersebut mengembalikan informasi seperti berikut. Contoh ini telah dipotong untuk ruang.

```
[
     "/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",
```

Melihat Wilayah yang didukung untuk Layanan AWS

Anda dapat melihat daftar Wilayah AWS di mana layanan tersedia. Contoh ini menggunakan 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
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
[
    "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"
]
```

Melihat titik akhir Regional untuk suatu layanan

Anda dapat melihat titik akhir Regional untuk suatu layanan dengan menggunakan perintah berikut. Perintah ini menanyakan Wilayah Timur AS (Ohio) (us-timur-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
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
"ssm.us-east-2.amazonaws.com"
```

Melihat detail lengkap Availability Zone

Anda dapat melihat Availability Zone dengan menggunakan perintah berikut.

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/
```

Perintah tersebut mengembalikan informasi seperti berikut. Contoh ini telah dipotong untuk ruang.

```
{
    "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"
        }
    ]
}
```

Melihat nama Availability Zone saja

Anda dapat melihat nama Availability Zone saja dengan menggunakan perintah berikut.

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(@)"
```

Perintah tersebut mengembalikan informasi seperti berikut. Contoh ini telah dipotong untuk ruang.

```
[
    "/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"
```

Melihat nama Availability Zones di satu Region

Anda dapat melihat nama-nama Availability Zone dalam satu Region (`us-east-2`, dalam contoh ini) menggunakan perintah berikut.

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(@)"
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
[
    "/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"
```

Melihat ARN Availability Zone saja

Anda dapat melihat nama Amazon Resource Names (ARN) dari Availability Zone saja dengan menggunakan perintah berikut.

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(@)"
```

Perintah tersebut mengembalikan informasi seperti berikut. Contoh ini telah dipotong untuk ruang.

```
[
    "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",
```

Melihat detail Local Zone

Anda dapat melihat Local Zone dengan menggunakan perintah berikut.

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
```

Perintah tersebut mengembalikan informasi seperti berikut. Contoh ini telah dipotong untuk ruang.

```
{
    "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"
        }
    ]
}
```

Melihat detail Wavelength Zone

Anda dapat melihat Wavelength Zone dengan menggunakan perintah berikut.

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
```

Perintah tersebut mengembalikan informasi seperti berikut. Contoh ini telah dipotong untuk ruang.


```
{
    "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"
        }
    ]
}
```


Melihat semua parameter dan nilai pada local zone

Anda dapat melihat semua data parameter untuk suatu Local Zone dengan menggunakan perintah berikut.

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"
```

Perintah tersebut mengembalikan informasi seperti berikut. Contoh ini telah dipotong untuk ruang.

```
{
    "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"
        }
    ]
}
```

Melihat nama parameter local zone saja

Anda dapat melihat nama parameter local zone saja dengan menggunakan perintah berikut.

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(@)"
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
[
    "/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"
]
```

Parameter StorePanduan

Panduan dalam bagian ini menunjukkan cara untuk membuat, menyimpan, dan menjalankan parameter denganParameter Store, suatu kemampuanAWS Systems Manager, dalam lingkungan pengujian. Panduan ini menunjukkan cara menggunakanParameter Storedengan kemampuan Systems Manager lainnya. Anda juga dapat menggunakanParameter Storedengan lainnyaLayanan AWS. Untuk informasi selengkapnya, lihat Apa itu parameter?.

Konten

Buat SecureString parameter dan bergabung dengan node ke Domain (PowerShell)

Panduan ini menunjukkan cara menggabungkan Windows Server node ke domain menggunakan AWS Systems Manager `SecureString` parameter dan. Run Command Panduan ini menggunakan parameter domain biasa, seperti nama domain dan nama pengguna domain. Nilai-nilai ini diteruskan sebagai nilai string tidak terenkripsi. Kata sandi domain dienkripsi menggunakan Kunci yang dikelola AWS dan diteruskan sebagai string terenkripsi.

Prasyarat

Panduan ini mengasumsikan bahwa Anda sudah menentukan nama domain dan alamat IP server DNS di opsi DHCP yang terkait dengan Amazon VPC Anda. Untuk informasi, lihat Bekerja dengan Rangkaian Opsi DHCP dalam Panduan Pengguna Amazon VPC.

Untuk membuat **`SecureString`** parameter dan bergabung dengan node ke domain

1. Masukkan parameter ke dalam sistem menggunakanAWS Tools for Windows PowerShell.

   Dalam perintah berikut, ganti setiap *`placeholder input pengguna dengan informasi`* Anda sendiri.

   ```
   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**
>
> Hanya nilai dari parameter `SecureString` yang dienkripsi. Nama parameter, deskripsi, dan properti lainnya tidak dienkripsi.

2. Lampirkan kebijakan AWS Identity and Access Management (IAM) berikut ini ke izin IAM role untuk node Anda:

   - AmazonSSM ManagedInstanceCore - Diperlukan. Kebijakan AWS terkelola ini memungkinkan node untuk menggunakan fungsi inti layanan Systems Manager.
   - AmazonSSM DirectoryServiceAccess - Diperlukan. Kebijakan AWS terkelola ini memungkinkan SSM Agent untuk mengakses AWS Directory Service atas nama Anda untuk meminta bergabung dengan domain oleh node terkelola.
   - untuk bucket S3 — Wajib. SSM Agent, yang ada di node Anda dan melakukan tugas Systems Manager, memerlukan akses ke bucket Amazon Simple Storage Service (Amazon S3) tertentu yang dimiliki Amazon. Dalam kebijakan kustom bucket S3 yang Anda buat, Anda juga menyediakan akses ke bucket S3 Anda sendiri yang diperlukan untuk operasi Systems Manager.

     Contoh: Anda dapat menulis untuk Run Command perintah atau Session Manager sesi ke bucket S3, lalu menggunakan output ini nanti untuk audit atau pemecahan masalah. Anda menyimpan script akses atau daftar dasar patch kustom dalam bucket S3, lalu mereferensi script atau daftar tersebut ketika Anda menjalankan sebuah perintah, atau ketika dasar patch diterapkan.

     Untuk informasi tentang membuat kebijakan kustom untuk akses bucket Amazon S3, lihat Membuat kebijakan bucket S3 kustom untuk profil instans

     > **Note**
     >
     > Menyimpan data log output dalam bucket S3 bersifat opsional, namun kami merekomendasikan untuk mengaturnya di awal proses konfigurasi Systems Manager jika Anda memutuskan untuk menggunakannya. Untuk informasi selengkapnya, lihat Membuat Bucket di Panduan Pengguna Amazon Simple Storage Service.

   - CloudWatchAgentServerPolicy – Opsional. Kebijakan AWS terkelola ini memungkinkan Anda untuk menjalankan CloudWatch agen pada node terkelola. Kebijakan ini memungkinkan

untuk membaca informasi pada sebuah node dan menuliskannya ke AmazonCloudWatch. Profil instans Anda memerlukan kebijakan ini hanya jika Anda menggunakan layanan seperti Amazon EventBridge atau CloudWatch Logs.

> **Note**
>
> Menggunakan CloudWatch dan EventBridge fitur bersifat opsional, namun kami merekomendasikan untuk mengaturnya di awal proses konfigurasi Systems Manager jika Anda memutuskan untuk menggunakannya. Untuk informasi selengkapnya, lihat Panduan EventBridge Pengguna Amazon dan Panduan Pengguna Amazon CloudWatch Logs.

3. Ubah IAM role yang dilampirkan ke node dan tambahkan kebijakan berikut. Kebijakan untuk memanggil `kms:Decrypt` dan `ssm:CreateDocument` API.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "ssm:CreateDocument"
            ],
            "Resource": [
                "arn:aws:kms:region:account-id:key/kms-key-id"
            ]
        }
    ]
}
```

4. Salin dan tempel teks json berikut ini ke editor teks dan simpan file sebagai `JoinInstanceToDomain.json` di lokasi berikut: `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. Jalankan perintah berikut ini di Tools for Windows PowerShell untuk membuat dokumen SSM baru.

```
$json = Get-Content C:\temp\JoinInstanceToDomain | Out-String
New-SSMDocument -Name JoinInstanceToDomain -Content $json -DocumentType Command
```

6. Jalankan perintah berikut ini di Tools for Windows PowerShell untuk menggabungkan node ke domain.

```
Send-SSMCommand -InstanceId instance-id -DocumentName JoinInstanceToDomain
```

Jika perintah berhasil, sistem mengembalikan informasi seperti berikut ini.

```
WARNING: The changes will take effect after you restart the computer EC2ABCD-
EXAMPLE.
Domain join succeeded, restarting
Computer is part of example.local, exiting
```

Jika perintah gagal, sistem mengembalikan informasi seperti berikut ini.

```
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.
```

Gunakan Parameter Store parameter di Amazon Elastic Kubernetes Service

Untuk menampilkan rahasia dari Secrets Manager dan parameter dari Parameter Store as file yang dipasang di pod Amazon EKS, Anda dapat menggunakan AWS Secrets and Configuration Provider (ASCP) untuk Kubernetes Secrets Store CSI Driver. (Parameter Storeadalah kemampuan AWS Systems Manager.) ASCP bekerja dengan Amazon Elastic Kubernetes Service (Amazon EKS) 1.17+. AWS Fargate (Fargate) grup node tidak didukung.

Dengan ASCP, Anda dapat mengambil parameter yang disimpan dan dikelola. Parameter Store Kemudian Anda dapat menggunakan parameter dalam beban kerja Anda yang berjalan di Amazon EKS. Jika parameter Anda berisi beberapa pasangan nilai kunci dalam format JSON, Anda dapat memilih untuk memasangnya di Amazon EKS. ASCP dapat menggunakan sintaks JMESPath untuk menanyakan pasangan nilai kunci dalam parameter Anda.

Anda dapat menggunakan peran dan kebijakan AWS Identity and Access Management (IAM) untuk membatasi akses ke parameter ke pod Amazon EKS tertentu dalam sebuah klaster. ASCP mengambil identitas pod dan menukar identitas untuk peran IAM. ASCP mengasumsikan peran IAM dari pod. Kemudian dapat mengambil parameter dari Parameter Store yang diizinkan untuk peran itu.

Untuk mempelajari cara mengintegrasikan Secrets Manager dengan Amazon EKS, lihat Menggunakan rahasia Secrets Manager di Amazon Elastic Kubernetes Service.

Menginstal ASCP

ASCP tersedia GitHub di repositori secrets-store-csi-driver-provider-aws. Repositori juga berisi contoh file YAMAL untuk membuat dan memasang rahasia. Anda pertama kali menginstal Kubernetes Secrets Store CSI Driver, dan kemudian Anda menginstal ASCP.

Untuk menginstal Kubernetes Secrets Store CSI Driver dan ASCP

1. Untuk menginstal Kubernetes Secrets Store CSI Driver, jalankan perintah berikut. Untuk petunjuk penginstalan lengkap, lihat Instalasi di Kubernetes Secrets Store CSI Driver Book. Untuk informasi tentang menginstal Helm, lihat Menggunakan Helm dengan Amazon EKS.

   ```
   helm repo add secrets-store-csi-driver https://kubernetes-sigs.github.io/secrets-store-csi-driver/charts
   helm install -n kube-system csi-secrets-store secrets-store-csi-driver/secrets-store-csi-driver
   ```

2. Untuk menginstal ASCP, gunakan file YAMM di direktori penyebaran GitHub repositori. Untuk informasi tentang menginstal`kubectl`, lihat Menginstal `kubectl`.

   ```
   kubectl apply -f https://raw.githubusercontent.com/aws/secrets-store-csi-driver-provider-aws/main/deployment/aws-provider-installer.yaml
   ```

Langkah 1: Siapkan kontrol akses

Untuk memberikan akses pod Amazon EKS ke parameterParameter Store, pertama-tama Anda membuat kebijakan yang membatasi akses ke parameter yang perlu diakses oleh pod. Kemudian Anda membuat peran IAM untuk akun layanan dan melampirkan kebijakan ke dalamnya. Untuk informasi selengkapnya tentang membatasi akses ke parameter Systems Manager menggunakan kebijakan IAM, lihat. Membatasi akses ke parameter Systems Manager menggunakan kebijakan IAM

> **Note**
>
> Saat menggunakan Parameter Store parameter, izin `ssm:GetParameters` diperlukan dalam kebijakan.

ASCP mengambil identitas pod dan menukarnya dengan peran IAM. ASCP mengasumsikan peran IAM dari pod, yang memberinya akses ke parameter yang Anda otorisasi. Kontainer lain tidak dapat mengakses parameter kecuali Anda juga mengaitkannya dengan peran IAM.

Langkah 2: Pasang parameter di Amazon EKS

Untuk menampilkan parameter di Amazon EKS seolah-olah mereka adalah file di sistem file, Anda membuat file `SecretProviderClass` YAMM yang berisi informasi tentang parameter Anda dan cara memasangnya di pod Amazon EKS.

`SecretProviderClass`Harus berada di namespace yang sama dengan pod Amazon EKS yang direferensikannya.

`SecretProviderClass`

`SecretProviderClass`YAMAL memiliki format berikut.

```
apiVersion: secrets-store.csi.x-k8s.io/v1alpha1
kind: SecretProviderClass
metadata:
   name: <NAME>
spec:
  provider: aws
  parameters:
```

parameter

Berisi detail permintaan pemasangan.

objek

Sebuah string yang berisi deklarasi YAMAL dari parameter yang akan dipasang. Sebaiknya gunakan karakter string atau pipe (|) multi-line YAMAL.

objectName

Nama parameter yang ramah. Ini menjadi nama file parameter di pod Amazon EKS kecuali Anda menentukan`objectAlias`. Untuk Parameter Store ini harus menjadi parameter, dan tidak bisa menjadi Nama Sumber Daya Amazon (ARN) lengkap. `Name`

JMESPath

(Opsional) Peta kunci dalam parameter yang dikodekan JSON ke file yang akan dipasang di Amazon EKS. Contoh berikut menunjukkan seperti apa parameter yang dikodekan JSON.

```
{
    "username" : "myusername",
    "password" : "mypassword"
}
```

Kuncinya adalah `username` dan`password`. Nilai yang terkait dengan `username` is`myusername`, dan nilai yang terkait dengannya `password` adalah`mypassword`.

path

Kunci dalam parameter.

Objectalias

Nama file yang akan dipasang di pod Amazon EKS.

objectType

UntukParameter Store, bidang ini diperlukan. Gunakan `ssmparameter`.

Objectalias

(Opsional) Nama file parameter di pod Amazon EKS. Jika Anda tidak menentukan bidang ini, `objectName` muncul sebagai nama file.

ObjectVersion

(Opsional) Nomor versi parameter. Kami menyarankan Anda untuk tidak menggunakan bidang ini karena Anda harus memperbaruinya setiap kali Anda memperbarui parameter. Secara default, versi terbaru digunakan. Untuk Parameter Store parameter, Anda dapat menggunakan `objectVersion` atau `objectVersionLabel` tetapi tidak keduanya.

objectVersionLabel

(Opsional) Label parameter untuk versi. Defaultnya adalah versi terbaru. Untuk Parameter Store parameter, Anda dapat menggunakan `objectVersion` atau `objectVersionLabel` tetapi tidak keduanya.

region

(Opsional) Parameter. Wilayah AWS Jika Anda tidak menggunakan bidang ini, ASCP mencari Region dari anotasi pada node. Pencarian ini menambahkan overhead ke permintaan mount, jadi sebaiknya Anda menyediakan Region untuk cluster yang menggunakan pod dalam jumlah besar.

PathTranslation

(Opsional) Karakter substitusi tunggal untuk digunakan jika nama file (salah satu `objectName` atau`objectAlias`) berisi karakter pemisah jalur, seperti garis miring (/) di Linux. Jika nama parameter berisi pemisah jalur, ASCP tidak dapat membuat file yang dipasang dengan nama itu. Sebagai gantinya, Anda dapat mengganti karakter pemisah jalur dengan karakter yang berbeda dengan memasukkannya di bidang ini. Jika Anda tidak menggunakan bidang ini, defaultnya adalah underscore (_), jadi misalnya, `My/Path/Parameter` mount as. `My_Path_Parameter`

Untuk mencegah substitusi karakter, masukkan string`False`.

Contoh

Contoh konfigurasi berikut menunjukkan `SecretProviderClass` dengan sumber daya Parameter Store parameter.

```
apiVersion: secrets-store.csi.x-k8s.io/v1alpha1
kind: SecretProviderClass
metadata:
    name: aws-secrets
spec:
    provider: aws
    parameters:
        objects: |
            - objectName: "MyParameter"
              objectType: "ssmparameter"
```

Langkah 3: Perbarui penerapan YAMAL

Perbarui penerapan YAMM Anda untuk menggunakan `secrets-store.csi.k8s.io` driver dan referensi `SecretProviderClass` sumber daya yang dibuat pada langkah sebelumnya. Ini memastikan cluster Anda menggunakan driver Secrets Store CSI.

Di bawah ini adalah contoh penerapan YAMAL menggunakan nama. `SecretProviderClass` `aws-secrets`

```
volumes:
  - name: secrets-store-inline
    csi:
      driver: secrets-store.csi.k8s.io
      readOnly: true
      volumeAttributes:
        secretProviderClass: "aws-secrets"
```

Tutorial: Membuat dan memasang parameter di pod Amazon EKS

Dalam tutorial ini, Anda membuat parameter contoh diParameter Store, dan kemudian Anda me-mount parameter di pod Amazon EKS dan menerapkannya.

Sebelum Anda mulai, instal ASCP. Untuk informasi selengkapnya, lihat the section called “Menginstal ASCP”.

Untuk membuat dan memasang rahasia

1. Atur Wilayah AWS dan nama cluster Anda sebagai variabel shell sehingga Anda dapat menggunakannya dalam `bash` perintah. Untuk *`wilayah`*, masukkan Wilayah AWS tempat cluster Amazon EKS Anda berjalan. Untuk *`clustername`*, masukkan nama cluster Anda.

   ```
   REGION=region
   CLUSTERNAME=clustername
   ```

2. Buat parameter uji.

   ```
   aws ssm put-parameter --name "MyParameter" --value "EKS parameter" --type String --
   region "$Region"
   ```

3. Buat kebijakan sumber daya untuk pod yang membatasi aksesnya ke parameter yang Anda buat pada langkah sebelumnya. Untuk<PARAMETERARN>, gunakan ARN parameter. Simpan ARN kebijakan dalam variabel shell. Untuk mengambil parameter ARN, gunakan. `get-parameter`

   ```
   POLICY_ARN=$(aws --region "$REGION" --query Policy.Arn --output text iam create-
   policy --policy-name nginx-parameter-deployment-policy --policy-document '{
       "Version": "2012-10-17",
       "Statement": [ {
           "Effect": "Allow",
           "Action": ["ssm:GetParameter", "ssm:GetParameters"],
           "Resource": ["parameter-arn"]
   ```

```
    } ]
}')
```

4. Buat penyedia OpenID Connect (OIDC) IAM untuk cluster jika Anda belum memilikinya. Untuk informasi selengkapnya, lihat Membuat penyedia IAM OIDC untuk klaster Anda.

```
eksctl utils associate-iam-oidc-provider --region="$REGION" --
cluster="$CLUSTERNAME" --approve # Only run this once
```

5. Buat akun layanan yang digunakan pod, dan kaitkan kebijakan sumber daya yang Anda buat di langkah 3 dengan akun layanan tersebut. Untuk tutorial ini, untuk nama akun layanan, Anda gunakan nginx-deployment-sa. Untuk informasi selengkapnya, lihat Membuat peran IAM untuk akun layanan.

```
eksctl create iamserviceaccount --name nginx-deployment-sa --region="$REGION" --
cluster "$CLUSTERNAME" --attach-policy-arn "$POLICY_ARN" --approve --override-
existing-serviceaccounts
```

6. Buat parameter `SecretProviderClass` untuk menentukan parameter mana yang akan dipasang di pod. Perintah berikut menggunakan lokasi file dari `SecretProviderClass` file bernama`ExampleSecretProviderClass.yaml`. Untuk informasi tentang membuat sendiri`SecretProviderClass`, lihatthe section called “`SecretProviderClass`”.

```
kubectl apply -f ./ExampleSecretProviderClass.yaml
```

7. Terapkan pod Anda. Perintah berikut menggunakan file deployment bernama`ExampleDeployment.yaml`. Untuk informasi tentang membuat sendiri`SecretProviderClass`, lihatthe section called “Langkah 3: Perbarui penerapan YAMAL”.

```
kubectl apply -f ./ExampleDeployment.yaml
```

8. Untuk memverifikasi parameter telah dipasang dengan benar, gunakan perintah berikut dan konfirmasikan bahwa nilai parameter Anda muncul.

```
kubectl exec -it $(kubectl get pods | awk '/nginx-deployment/{print $1}' | head -1)
 cat /mnt/secrets-store/MyParameter; echo
```

   Nilai parameter muncul.

```
"EKS parameter"
```

Pemecahan Masalah

Anda dapat melihat sebagian besar kesalahan dengan menjelaskan penerapan pod.

Untuk melihat pesan kesalahan untuk penampung Anda

1. Dapatkan daftar nama pod dengan perintah berikut. Jika Anda tidak menggunakan namespace default, gunakan. `-n <NAMESPACE>`

   ```
   kubectl get pods
   ```

2. Untuk mendeskripsikan pod, pada perintah berikut, untuk *pod-id* gunakan ID pod dari pod yang Anda temukan di langkah sebelumnya. Jika Anda tidak menggunakan namespace default, gunakan. `-n <NAMESPACE>`

   ```
   kubectl describe pod/pod-id
   ```

Untuk melihat kesalahan untuk ASCP

- Untuk menemukan informasi selengkapnya di log penyedia, pada perintah berikut, untuk *pod-id* gunakan ID pod csi-secrets-store-provider-aws.

  ```
  kubectl -n kube-system get pods
  kubectl -n kube-system logs pod/pod-id
  ```

Pengauditan dan pencatatanParameter Storeaktivitas

AWS CloudTrail menangkap panggilan API yang dibuat dalam konsol AWS Systems Manager, AWS Command Line Interface (AWS CLI), dan SDK Systems Manager. Anda dapat melihat informasi di CloudTrail konsol atau bucket Amazon Simple Storage Service (Amazon S3). Semua CloudTrail log untuk akun Anda menggunakan satu bucket. Untuk informasi selengkapnya tentang melihat dan menggunakan CloudTrail log aktivitas Systems Manager, lihatPencatatan panggilan AWS Systems Manager API dengan AWS CloudTrail. Untuk informasi selengkapnya tentang opsi pengauditan dan pencatatan untuk Systems Manager, lihat Pemantauan AWS Systems Manager.

Pemecahan Masalah Parameter Store

Gunakan informasi berikut untuk membantu Anda memecahkan masalah denganParameter Store, kemampuan. AWS Systems Manager

Pemecahan masalah pembuatan parameter **aws:ec2:image**

Gunakan informasi berikut untuk membantu memecahkan masalah dengan membuat parameter tipe data `aws:ec2:image`.

Tidak ada izin untuk membuat instance

Masalah: Anda mencoba membuat instance menggunakan `aws:ec2:image` parameter tetapi menerima pesan kesalahan seperti “Anda tidak berwenang untuk melakukan operasi ini.”

- Solusi: Anda tidak memiliki semua izin yang diperlukan untuk membuat instans EC2 menggunakan nilai parameter, seperti izin untuk,, dan `ec2:RunInstances` `ec2:DescribeImagesssm:GetParameter`, antara lain. Hubungi pengguna dengan izin administrator di organisasi Anda untuk meminta izin yang diperlukan.

EventBridge melaporkan pesan kegagalan “Tidak Dapat Menjelaskan Sumber Daya”

Masalah: Anda menjalankan perintah untuk membuat parameter `aws:ec2:image`, tetapi pembuatan parameter gagal. Anda menerima pemberitahuan dari Amazon EventBridge yang melaporkan pengecualian “Tidak Dapat Menjelaskan Sumber Daya”.

Solusi: Pesan ini dapat menunjukkan hal berikut ini:

- Anda tidak memiliki semua izin yang diperlukan untuk operasi `ec2:DescribeImages` API, atau Anda tidak memiliki izin untuk mengakses gambar tertentu yang direferensikan dalam parameter. Hubungi pengguna dengan izin administrator di organisasi Anda untuk meminta izin yang diperlukan.
- ID Amazon Machine Image (AMI) yang Anda masukkan sebagai nilai parameter tidak valid. Pastikan Anda memasukkan ID AMI yang tersedia di saat ini Wilayah AWS dan akun tempat Anda bekerja.

parameter **aws:ec2:image** baru tidak tersedia

Masalah: Anda baru saja menjalankan perintah untuk membuat parameter `aws:ec2:image` dan nomor versi dilaporkan, namun parameternya tidak tersedia.

- Solusi: Ketika Anda menjalankan perintah untuk membuat parameter yang menggunakan tipe data `aws:ec2:image`, nomor versi segera dihasilkan untuk parameter tersebut, tetapi format parameter harus divalidasi sebelum parameter tersedia. Proses ini dapat memakan waktu hingga beberapa menit. Untuk memantau proses pembuatan dan validasi parameter, Anda dapat melakukan hal berikut:
  - Gunakan EventBridge untuk mengirimi Anda pemberitahuan tentang operasi Anda `create` dan `update` parameter. Notifikasi ini melaporkan apakah operasi parameter berhasil atau tidak. Untuk informasi tentang berlangganan Parameter Store acara di EventBridge, lihatMenyiapkan notifikasi atau memicu tindakan berdasarkan Parameter Store peristiwa.
  - Di Parameter Store bagian konsol Systems Manager, segarkan daftar parameter secara berkala untuk mencari detail parameter baru atau yang diperbarui.
  - Gunakan perintah GetParameter untuk memeriksa parameter yang baru atau telah diperbarui. Sebagai contoh, menggunakan AWS Command Line Interface (AWS CLI):

    ```
    aws ssm get-parameter name MyParameter
    ```

    Untuk parameter baru, pesan `ParameterNotFound` akan dikembalikan sampai parameter divalidasi. Untuk parameter yang ada yang Anda perbarui, informasi tentang versi baru tidak disertakan hingga parameter tersebut divalidasi.

  Jika Anda mencoba untuk membuat atau memperbarui parameter lagi sebelum proses validasi selesai, sistem akan melaporkan bahwa validasi masih dalam proses. Jika parameter tidak dibuat atau diperbarui, Anda dapat mencoba lagi setelah 5 menit berlalu dari percobaan awal.

AWS Systems Manager Manajemen Perubahan

AWS Systems Manager menyediakan kemampuan berikut untuk membuat perubahan pada AWS sumber daya.

Topik

AWS Systems Manager Change Manager

Change Manager, kemampuan dari, kemampuan dariAWS Systems Manager, adalah kerangka manajemen perubahan korporasi untuk meminta, menyetujui, menerapkan, menerapkan, dan melaporkan perubahan operasional pada konfigurasi dan infrastruktur aplikasi Anda. Dari satu akun administrator yang didelegasikan, jika Anda menggunakan AWS Organizations, Anda dapat mengelola perubahan di beberapa Akun AWS dan di seluruh Wilayah AWS. Atau, dengan menggunakan akun lokal, Anda dapat mengelola perubahan untuk satu Akun AWS. Gunakan Change Manager untuk mengelola perubahan untuk kedua AWS sumber daya dan on-premise. Untuk memulaiChange Manager, buka konsol Systems Manager. Di panel navigasi, pilih Change Manager.

DenganChange Manager, Anda dapat menggunakan templat perubahan yang telah disetujui sebelumnya untuk membantu mengotomatiskan proses perubahan untuk sumber daya Anda dan membantu menghindari hasil yang tidak disengaja saat membuat perubahan operasional. Setiap templat perubahan menentukan hal berikut:

- Satu atau beberapa runbook Otomatisasi untuk dipilih pengguna saat membuat permintaan perubahan. Perubahan yang dibuat pada sumber daya Anda ditentukan dalam runbook otomatisasi. Anda dapat menyertakan runbook kustom atau runbook terkelola AWS dalam templat perubahan yang Anda buat. Ketika pengguna membuat permintaan perubahan, mereka dapat memilih salah satu runbook yang tersedia untuk disertakan dalam permintaan. Selain itu, Anda dapat membuat templat perubahan yang memungkinkan pengguna yang membuat permintaan untuk menentukan runbook apa pun dalam permintaan perubahan.

- Pengguna di akun yang harus meninjau permintaan perubahan yang dibuat menggunakan templat perubahan tersebut.
- Topik Amazon Simple Notification Service (Amazon SNS) yang digunakan untuk memberi tahu pemberi persetujuan yang ditetapkan bahwa permintaan perubahan siap untuk ditinjau.
- CloudWatchAlarm Amazon yang digunakan untuk memantau alur kerja runbook.
- Topik Amazon SNS yang digunakan untuk mengirim notifikasi tentang perubahan status untuk permintaan perubahan yang dibuat menggunakan templat perubahan.
- Tag yang akan diterapkan ke templat perubahan untuk digunakan dalam mengategorikan dan memfilter templat perubahan Anda.
- Apakah permintaan perubahan yang dibuat dari template perubahan dapat dijalankan tanpa langkah persetujuan (permintaan yang disetujui otomatis).

Melalui integrasinya denganChange Calendar, yang merupakan kemampuan lain dari Systems Manager, Change Manager juga membantu Anda menerapkan perubahan dengan aman sambil menghindari konflik jadwal dengan peristiwa bisnis penting. Change ManagerIntegrasi dengan AWS Organizations dan AWS IAM Identity Center membantu Anda mengelola perubahan di seluruh organisasi Anda dari satu akun menggunakan sistem manajemen identitas yang ada. Anda dapat memantau kemajuan perubahan dari Change Manager dan mengaudit perubahan operasional di seluruh organisasi Anda, memberikan peningkatan visibilitas dan akuntabilitas.

Change Managermelengkapi kontrol keamanan praktik Integrasi Berkelanjutan (CI) dan metodologi Pengiriman Berkelanjutan (CD) Anda. Change Managertidak dimaksudkan untuk perubahan yang dibuat sebagai bagian dari proses rilis otomatis, seperti pipeline CI/CD, kecuali ada pengecualian atau persetujuan yang diperlukan.

Cara kerja Change Manager

Ketika kebutuhan untuk perubahan operasional standar atau darurat diidentifikasi, seseorang di organisasi membuat permintaan perubahan yang didasarkan pada salah satu templat perubahan yang dibuat untuk digunakan di organisasi atau akun Anda.

Jika perubahan yang diminta memerlukan persetujuan manual, Change Manager beri tahu pemberi persetujuan yang ditunjuk melalui notifikasi Amazon SNS bahwa permintaan perubahan siap untuk ditinjau. Anda dapat menunjuk pemberi persetujuan untuk permintaan perubahan di templat perubahan, atau membiarkan pengguna menunjuk pemberi persetujuan dalam permintaan perubahan itu sendiri. Anda dapat menetapkan peninjau yang berbeda untuk templat yang berbeda.

Misalnya, tetapkan satu peran pengguna, grup pengguna, atau peran AWS Identity and Access Management (IAM) yang harus menyetujui permintaan perubahan ke node terkelola, dan peran pengguna, grup, atau IAM lainnya untuk perubahan database. Jika template perubahan mengizinkan persetujuan otomatis, dan kebijakan pengguna pemohon tidak melarangnya, pengguna juga dapat memilih untuk menjalankan runbook Otomasi untuk permintaan mereka tanpa langkah peninjauan (dengan pengecualian peristiwa pembekuan perubahan).

Untuk setiap templat perubahan, Anda dapat menambahkan hingga lima tingkat pemberi persetujuan. Misalnya, Anda mungkin memerlukan peninjau teknis untuk menyetujui permintaan perubahan yang dibuat dari templat perubahan terlebih dahulu, dan kemudian memerlukan persetujuan tingkat kedua dari satu pengelola atau lebih.

Change Manager terintegrasi dengan AWS Systems Manager Change Calendar. Ketika perubahan yang diminta disetujui, sistem pertama-tama akan menentukan apakah permintaan tersebut bertentangan dengan aktivitas bisnis terjadwal lainnya. Jika konflik terdeteksi, Change Manager dapat memblokir perubahan atau memerlukan persetujuan tambahan sebelum memulai alur kerja runbook. Misalnya, Anda mungkin mengizinkan perubahan hanya selama jam kerja untuk memastikan bahwa tim tersedia untuk mengelola masalah yang tidak terduga. Untuk setiap perubahan yang diminta untuk dijalankan di luar jam tersebut, Anda dapat meminta persetujuan manajemen tingkat yang lebih tinggi dalam bentuk pemberi persetujuan pembekuan perubahan. Untuk perubahan darurat, Change Manager dapat melewati langkah pemeriksaan Change Calendar konflik atau peristiwa pemblokiran setelah permintaan perubahan disetujui.

Saat tiba waktunya untuk menerapkan perubahan yang disetujui, Change Manager jalankan runbook Otomatisasi yang ditentukan dalam permintaan perubahan terkait. Hanya operasi yang ditentukan dalam permintaan perubahan yang disetujui yang diizinkan saat alur kerja runbook berjalan. Pendekatan ini membantu Anda menghindari hasil yang tidak disengaja saat perubahan sedang dilaksanakan.

Selain membatasi perubahan yang dapat dibuat saat alur kerja runbook berjalan, Change Manager juga membantu Anda mengontrol ambang batas konkurensi dan kesalahan. Anda memilih berapa banyak sumber daya yang dapat dijalankan oleh alur kerja runbook sekaligus, berapa banyak akun yang dapat menjalankan perubahan sekaligus, dan berapa banyak kegagalan yang diizinkan sebelum proses dihentikan dan (jika runbook menyertakan skrip rollback) dibatalkan. Anda juga dapat memantau kemajuan perubahan yang dibuat dengan menggunakan CloudWatch alarm.

Setelah alur kerja runbook selesai, Anda dapat meninjau detail tentang perubahan yang dibuat. Detail ini mencakup alasan permintaan perubahan, templat perubahan mana yang digunakan, siapa yang meminta dan menyetujui perubahan, dan bagaimana perubahan diterapkan.

Info lebih lanjut

Memperkenalkan AWS Systems ManagerChange Manager di Blog AWSBerita

Bagaimana dapat Change Manager menguntungkan operasi saya?

Manfaat Change Manager antara lain sebagai berikut:

- Mengurangi risiko gangguan layanan dan waktu henti

  Change Managerdapat membuat perubahan operasional lebih aman dengan memastikan bahwa hanya perubahan yang disetujui yang diterapkan saat alur kerja runbook berjalan. Anda dapat memblokir perubahan yang tidak direncanakan dan belum ditinjau. Change Managermembantu Anda menghindari jenis hasil yang tidak disengaja yang disebabkan oleh kesalahan manusia yang memerlukan berjam-jam penelitian dan backtrack.
- Dapatkan audit dan pelaporan mendetail tentang riwayat perubahan

  Change Managermemberikan akuntabilitas dengan cara yang konsisten untuk melaporkan dan mengaudit perubahan yang dibuat di seluruh organisasi Anda, dan detail tentang siapa yang menyetujui dan menerapkannya.
- Menghindari konflik atau pelanggaran jadwal

  Change Managerdapat mendeteksi konflik jadwal atau peluncuran produk baru, berdasarkan kalender perubahan aktif untuk organisasi Anda. Anda dapat mengizinkan alur kerja runbook untuk berjalan hanya selama jam kerja, atau mengizinkannya hanya dengan persetujuan tambahan.
- Menyesuaikan persyaratan perubahan untuk bisnis Anda yang berubah

  Selama periode bisnis yang berbeda, Anda dapat menerapkan persyaratan manajemen perubahan yang berbeda. Misalnya, selama nd-of-month pelaporan, musim pajak, atau periode bisnis penting lainnya, Anda dapat memblokir perubahan atau meminta persetujuan tingkat direktur untuk perubahan yang dapat menimbulkan risiko operasional yang tidak perlu.
- Mengelola perubahan di seluruh akun secara terpusat

  Melalui integrasinya dengan Organizations, Change Manager memungkinkan Anda untuk mengelola perubahan di seluruh unit organisasi (OU) Anda dari satu akun administrator yang didelegasikan. Anda dapat mengaktifkan Change Manager untuk digunakan dengan seluruh organisasi atau hanya dengan beberapa OU Anda.

Siapa yang harus menggunakanChange Manager?

Change Managersesuai untuk AWS pelanggan dan organisasi berikut:

- Setiap pelanggan AWS yang ingin meningkatkan keamanan dan tata kelola perubahan operasional yang dibuat pada cloud atau lingkungan on-premise mereka.
- Organizations yang ingin meningkatkan kolaborasi dan visibilitas di seluruh tim, meningkatkan ketersediaan aplikasi dengan menghindari waktu henti, dan mengurangi risiko yang terkait dengan tugas manual dan berulang.
- Organizations yang harus mematuhi praktik terbaik untuk manajemen perubahan.
- Pelanggan yang membutuhkan riwayat perubahan yang dapat diaudit sepenuhnya pada konfigurasi dan infrastruktur aplikasi mereka.

Apa saja fitur utama dariChange Manager?

Fitur utama dari Change Manager meliputi sebagai berikut:

- Dukungan terintegrasi untuk praktik terbaik manajemen perubahan

  DenganChange Manager, Anda dapat menerapkan praktik terbaik manajemen perubahan tertentu untuk operasi Anda. Anda dapat memilih untuk mengaktifkan opsi berikut:

  - Memeriksa Change Calendar untuk melihat apakah peristiwa saat ini sedang dibatasi sehingga perubahan dilakukan hanya selama periode kalender terbuka.
  - Mengizinkan perubahan selama peristiwa terbatas dengan persetujuan ekstra dari pemberi persetujuan pembekuan perubahan.
  - Mengharuskan CloudWatch alarm yang akan ditentukan untuk semua templat perubahan.
  - Mengharuskan semua templat perubahan yang dibuat di akun Anda untuk ditinjau dan disetujui sebelum dapat digunakan untuk membuat permintaan perubahan.
- Jalur persetujuan yang berbeda untuk periode kalender tertutup dan permintaan perubahan darurat

  Anda dapat mengizinkan opsi Change Calendar untuk mencentang peristiwa terbatas dan memblokir permintaan perubahan yang disetujui hingga peristiwa selesai. Namun, Anda juga dapat menunjuk grup pemberi persetujuan kedua, pemberi persetujuan pembekuan perubahan, yang dapat mengizinkan perubahan dilakukan meskipun kalender ditutup. Anda juga dapat membuat templat perubahan darurat. Permintaan perubahan yang dibuat dari templat perubahan darurat

masih memerlukan persetujuan reguler tetapi tidak tunduk pada batasan kalender dan tidak memerlukan persetujuan pembekuan perubahan.

- Mengontrol bagaimana dan kapan alur kerja runbook dimulai

  Alur kerja runbook dapat dimulai sesuai dengan jadwal, atau segera setelah persetujuan selesai (tunduk pada aturan pembatasan kalender).

- Dukungan notifikasi bawaan

  Tentukan siapa di organisasi Anda yang harus meninjau dan menyetujui templat perubahan dan permintaan perubahan. Tetapkan topik Amazon SNS ke templat perubahan untuk mengirim notifikasi ke pelanggan topik tentang perubahan status untuk permintaan perubahan yang dibuat dengan templat perubahan tersebut.

- Integrasi dengan AWS Systems Manager Change Calendar

  Change Managermemungkinkan administrator untuk membatasi perubahan penjadwalan selama periode waktu tertentu. Misalnya, Anda dapat membuat kebijakan yang mengizinkan perubahan hanya selama jam kerja untuk memastikan bahwa tim tersedia untuk menangani masalah apa pun. Anda juga dapat membatasi perubahan selama peristiwa bisnis penting. Misalnya, bisnis ritel mungkin membatasi perubahan selama peristiwa penjualan besar. Anda juga dapat meminta persetujuan tambahan selama periode terbatas.

- Integrasi dengan dukungan AWS IAM Identity Center dan Direktori Aktif

  Dengan integrasi IAM Identity Center, anggota organisasi Anda dapat mengakses Akun AWS dan mengelola sumber daya mereka menggunakan Systems Manager berdasarkan identitas pengguna yang sama. Dengan IAM Identity Center, Anda dapat menetapkan akses pengguna ke akun di seluruhAWS.

  Integrasi dengan Direktori Aktif memungkinkan untuk menetapkan pengguna di akun Direktori Aktif Anda sebagai pemberi persetujuan untuk templat perubahan yang dibuat untuk Change Manager operasi Anda.

- Integrasi dengan CloudWatch alarm Amazon

  Change Managerterintegrasi dengan CloudWatch alarm. Change Managermendengarkan CloudWatch alarm selama alur kerja runbook dan mengambil tindakan apa pun, termasuk mengirim notifikasi, yang ditentukan untuk alarm.

- Integrasi dengan AWS CloudTrail Danau

Dengan membuat penyimpanan data peristiwa di AWS CloudTrail Lake, Anda dapat melihat informasi yang dapat diaudit tentang perubahan yang dilakukan oleh permintaan perubahan yang berjalan di akun atau organisasi Anda. Informasi acara yang disimpan mencakup rincian seperti berikut ini:

- Tindakan API yang dijalankan
- Parameter permintaan Tthe disertakan untuk tindakan tersebut
- Pengguna yang menjalankan tindakan
- Sumber daya yang diperbarui selama proses

- Integrasi dengan AWS Organizations

  Dengan menggunakan kemampuan lintas-akun yang disediakan oleh Organizations, Anda dapat menggunakan akun administrator yang didelegasikan untuk mengelola Change Manager operasi di OU di organisasi Anda. Di akun manajemen Organizations, Anda dapat menentukan akun mana yang menjadi akun administrator yang didelegasikan. Anda juga dapat mengontrol OU mana yang Change Manager dapat digunakan.

Apakah ada biaya untuk digunakanChange Manager?

Ya. Change Managerdibanderol pay-per-use berdasarkan harga. Anda hanya membayar untuk apa yang Anda gunakan. Untuk informasi selengkapnya, lihat Harga AWS Systems Manager.

Apa komponen utama dariChange Manager?

Change Managerkomponen yang Anda gunakan untuk mengelola proses perubahan di organisasi atau akun Anda meliputi:

Akun administrator yang didelegasikan

Jika Anda menggunakan Change Manager seluruh organisasi, Anda menggunakan akun administrator yang didelegasikan. Ini adalah yang Akun AWS ditetapkan sebagai akun untuk mengelola aktivitas operasi di seluruh Systems Manager, termasukChange Manager. Akun administrator yang didelegasikan mengelola aktivitas perubahan di seluruh organisasi Anda. Saat menyiapkan organisasi untuk digunakanChange Manager, Anda menentukan akun mana yang berfungsi dalam peran ini. Akun administrator yang didelegasikan harus menjadi satu-satunya anggota unit organisasi (OU) yang ditetapkan. Akun administrator yang didelegasikan tidak diperlukan jika Anda menggunakannya Change Manager dengan satu Akun AWS saja.

> **Important**
>
> Jika Anda menggunakan Change Manager seluruh organisasi, kami sarankan untuk selalu membuat perubahan dari akun administrator yang didelegasikan. Meskipun Anda dapat membuat perubahan dari akun lain di organisasi, perubahan tersebut tidak akan dilaporkan atau dapat dilihat dari akun administrator yang didelegasikan.

Mengubah templat

Templat perubahan adalah kumpulan pengaturan konfigurasi Change Manager yang menentukan hal-hal seperti persetujuan yang diperlukan, dan opsi notifikasi untuk permintaan perubahan.

Anda dapat mengharuskan templat perubahan yang dibuat oleh pengguna di organisasi atau akun Anda untuk melalui proses persetujuan sebelum dapat digunakan.

Change Managermendukung dua jenis perubahan templat. Untuk permintaan perubahan yang disetujui yang didasarkan pada templat perubahan darurat, perubahan yang diminta dapat dilakukan meskipun ada peristiwa pemblokiran diChange Calendar. Untuk permintaan perubahan yang disetujui yang didasarkan pada templat perubahan standar, perubahan yang diminta tidak dapat dilakukan jika ada peristiwa pemblokiran Change Calendar kecuali persetujuan tambahan diterima dari pemberi persetujuan peristiwa pembekuan perubahan yang ditunjuk.

Permintaan perubahan

Permintaan perubahan adalah permintaan Change Manager untuk menjalankan runbook Otomatisasi yang memperbarui satu atau beberapa sumber daya di lingkungan AWS atau on-premise Anda. Permintaan perubahan dibuat menggunakan templat perubahan.

Saat Anda membuat permintaan perubahan, satu atau beberapa pemberi persetujuan di organisasi atau akun Anda harus meninjau dan menyetujui permintaan tersebut. Tanpa persetujuan yang diperlukan, alur kerja runbook, yang menerapkan perubahan yang Anda minta, tidak diizinkan untuk berjalan.

Dalam sistem, permintaan perubahan adalah jenis OpsItem di AWS Systems ManagerOpsCenter. Namun, OpsItems dari jenisnya `/aws/changerequest` tidak ditampilkan diOpsCenter. KarenaOpsItems, permintaan perubahan tunduk pada kuota yang diberlakukan sama seperti jenis lain dari. OpsItems

Selain itu, untuk membuat permintaan perubahan secara terprogram, Anda tidak memanggil operasi API `CreateOpsItem`. Sebaliknya, Anda menggunakan operasi API `StartChangeRequestExecution`. Namun, alih-alih langsung berjalan, permintaan perubahan harus disetujui, dan tidak boleh ada peristiwa pemblokiran Change Calendar untuk mencegah berjalannya alur kerja. Ketika persetujuan telah diterima dan kalender tidak diblokir (atau izin telah diberikan untuk mengabaikan peristiwa pemblokiran kalender), tindakan `StartChangeRequestExecution` dapat diselesaikan.

Alur kerja runbook

Alur kerja runbook adalah proses perubahan yang diminta yang dibuat di sumber daya yang ditargetkan di lingkungan cloud atau on-premise Anda. Setiap permintaan perubahan menunjuk satu runbook Otomatisasi yang akan digunakan untuk membuat perubahan yang diminta. Alur kerja runbook terjadi setelah semua persetujuan yang diperlukan telah diberikan dan tidak ada peristiwa pemblokiran di. Change Calendar Jika perubahan telah dijadwalkan untuk tanggal dan waktu tertentu, alur kerja runbook tidak dimulai hingga dijadwalkan, meskipun semua persetujuan telah diterima dan kalender tidak diblokir.

Topik

- Menyiapkan Change Manager
- Bekerja dengan Change Manager
- Audit dan loggingChange Manageraktivitas
- Pemecahan Masalah Change Manager

Menyiapkan Change Manager

Anda dapat menggunakanChange Manager, kemampuan dariAWS Systems Manager, untuk mengelola perubahan untuk seluruh organisasi, seperti yang dikonfigurasi diAWS Organizations, atau untuk satuAkun AWS.

Jika Anda menggunakanChange Manager dengan organisasi, mulailah dengan topikMenyiapkan Change Manager untuk organisasi (akun manajemen), dan kemudian lanjutkan keMengonfigurasiChange Manager opsi dan praktik terbaik dan praktik terbaik.

Jika Anda menggunakanChange Manager dengan satu akun, lanjutkan langsung keMengonfigurasiChange Manager opsi dan praktik terbaik dan praktik terbaik.

> **Note**
>
> Jika Anda mulai menggunakanChange Manager dengan satu akun, tetapi akun tersebut kemudian ditambahkan ke unit organisasi yangChange Manager diizinkan, pengaturan akun tunggal Anda akan diabaikan.

Topik

Menyiapkan Change Manager untuk organisasi (akun manajemen)

Tugas dalam topik ini berlaku jika Anda menggunakan Change Manager AWS Systems Manager, kemampuan, dengan organisasi yang diatur AWS Organizations. Jika Anda ingin menggunakan Change Manager hanya dengan satu Akun AWS, lompat ke topikMengonfigurasiChange Manager opsi dan praktik terbaik dan praktik terbaik.

Lakukan tugas-tugas di bagian ini di Akun AWS yang berfungsi sebagai akun manajemen di Organizations. Untuk informasi tentang akun manajemen dan konsep Organizations lainnya, lihat terminologi dan konsep AWS Organizations.

Jika Anda perlu mengaktifkan Organizations dan menentukan akun Anda sebagai akun manajemen sebelum melanjutkan, lihat Membuat dan mengelola organisasidi Panduan Pengguna AWS Organizations .

> **Note**
>
> Proses penyiapan ini tidak dapat dilakukan dalam hal berikut Wilayah AWS:
>
> - Europe (Milan) (eu-south-1)
> - Middle East (Bahrain) (me-south-1)
> - Africa (Cape Town) (af-south-1)
> - Asia Pacific (Hong Kong) (ap-east-1)

Pastikan Anda bekerja di Wilayah yang berbeda di akun manajemen Anda untuk prosedur ini.

Selama prosedur penyiapan, Anda melakukan tugas-tugas utama berikut diQuick Setup, kemampuan AWS Systems Manager.

- Tugas 1: Daftarkan akun administrator yang didelegasikan untuk organisasi Anda

  Tugas terkait perubahan yang dilakukan menggunakan Change Manager dikelola di salah satu akun anggota Anda, yang Anda tentukan sebagai akun administrator yang didelegasikan. Akun administrator yang didelegasikan yang Anda daftarkan Change Manager menjadi akun administrator yang didelegasikan untuk semua operasi Systems Manager Anda. (Anda mungkin telah mendelegasikan akun administrator untuk lainnya Layanan AWS). Akun administrator yang didelegasikanChange Manager, yang tidak sama dengan akun manajemen Anda, mengelola aktivitas perubahan di seluruh organisasi Anda, termasuk templat perubahan, permintaan perubahan, dan persetujuan untuk masing-masing. Di akun administrator yang didelegasikan, Anda juga menentukan opsi konfigurasi lain untuk Change Manager operasi Anda.

  Important

  Akun administrator yang didelegasikan harus menjadi satu-satunya anggota unit organisasi (OU) yang ditetapkan di Organizations.

- Tugas 2: Tentukan dan tentukan kebijakan akses buku runbook untuk peran pemohon perubahan, atau fungsi pekerjaan khusus, yang ingin Anda gunakan untuk operasi Change Manager

  Untuk membuat permintaan perubahanChange Manager, pengguna di akun anggota Anda harus diberikan izin AWS Identity and Access Management (IAM) yang memungkinkan mereka mengakses hanya runbook Otomasi dan mengubah templat yang Anda pilih untuk disediakan bagi mereka.

  Note

  Ketika pengguna membuat permintaan perubahan, mereka terlebih dahulu memilih templat perubahan. Templat perubahan ini mungkin membuat beberapa runbook tersedia, tetapi pengguna hanya dapat memilih satu runbook untuk setiap permintaan perubahan. Templat

perubahan juga dapat dikonfigurasi untuk memungkinkan pengguna untuk menyertakan buku runbook yang tersedia dalam permintaan mereka.

Untuk memberikan izin yang diperlukan, Change Manager gunakan konsep fungsi pekerjaan, yang juga digunakan oleh IAM. Namun, tidak seperti kebijakan AWS terkelola untuk fungsi pekerjaan di IAM, Anda menentukan nama fungsi Change Manager pekerjaan dan izin IAM untuk fungsi pekerjaan tersebut.

Saat Anda mengonfigurasi fungsi tugas, sebaiknya buat kebijakan kustom dan berikan hanya izin yang diperlukan untuk melakukan tugas manajemen perubahan. Misalnya, Anda dapat menentukan izin yang membatasi pengguna ke kumpulan runbook tertentu berdasarkan fungsi pekerjaan yang Anda tentukan.

Misalnya, Anda dapat membuat fungsi tugas dengan nama `DBAdmin`. Untuk fungsi tugas ini, Anda mungkin hanya memberikan izin yang diperlukan untuk runbook yang terkait dengan database Amazon DynamoDB, seperti `AWS-CreateDynamoDbBackup` dan `AWSConfigRemediation-DeleteDynamoDbTable`.

Sebagai contoh lain, Anda mungkin ingin memberikan beberapa pengguna izin yang diperlukan saja untuk bekerja dengan runbook yang terkait dengan bucket Amazon Simple Storage Service (Amazon S3) seperti `AWS-ConfigureS3BucketLogging` dan `AWSConfigRemediation-ConfigureS3BucketPublicAccessBlock`.

Proses konfigurasi di Quick Setup for Change Manager juga membuat satu set izin administratif Systems Manager lengkap tersedia bagi Anda untuk diterapkan ke peran administratif yang Anda buat.

Setiap Change Manager Quick Setup konfigurasi yang Anda terapkan akan membuat fungsi pekerjaan di akun administrator yang didelegasikan dengan izin untuk menjalankan Change Manager templat dan runbook Otomasi di unit organisasi yang telah Anda pilih. Anda dapat membuat hingga 15 Quick Setup konfigurasi untukChange Manager.

- Tugas 3: Pilih akun anggota mana di organisasi Anda yang akan digunakan Change Manager

  Anda dapat menggunakan Change Manager dengan semua akun anggota di semua unit organisasi Anda yang diatur di Organizations, dan di semua akun Wilayah AWS tersebut beroperasi. Jika Anda mau, Anda dapat menggunakan hanya Change Manager dengan beberapa unit organisasi Anda.

> **Important**
>
> Kami sangat menyarankan, sebelum memulai prosedur ini, bahwa Anda membaca langkah-langkahnya untuk memahami pilihan konfigurasi yang Anda buat dan izin yang Anda berikan. Secara khusus, rencanakan fungsi tugas khusus yang akan Anda buat dan izin yang Anda tetapkan untuk setiap fungsi tugas. Ini memastikan bahwa ketika nanti Anda melampirkan kebijakan fungsi pekerjaan yang Anda buat ke pengguna individu, grup pengguna, atau peran IAM, mereka hanya diberikan izin yang Anda inginkan untuk mereka miliki.
> Sebagai praktik terbaik, mulailah dengan menyiapkan akun administrator yang didelegasikan menggunakan login untuk Akun AWS administrator. Kemudian, konfigurasikan fungsi tugas dan izinnya setelah Anda membuat templat perubahan dan mengidentifikasi runbook yang digunakan masing-masing.

Change ManagerUntuk mengatur penggunaan dengan organisasi, lakukan tugas berikut di Quick Setup area konsol Systems Manager.

Ulangi tugas ini untuk setiap fungsi tugas yang ingin Anda buat untuk organisasi Anda. Setiap fungsi tugas yang Anda buat dapat memiliki izin untuk serangkaian unit organisasi yang berbeda.

Untuk mengatur organisasi Change Manager di akun manajemen Organizations

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Quick Setup.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Quick Setupdi panel navigasi.
3. Pada Change Managerkartu, pilih Buat.
4. Untuk akun administrator yang didelegasikan, masukkan ID yang ingin Akun AWS Anda gunakan untuk mengelola templat perubahan, permintaan perubahan, dan alur kerja buku runbook. Change Manager

   Jika Anda sebelumnya telah menetapkan akun administrator yang didelegasikan untuk Systems Manager, ID akun tersebut sudah dilaporkan dalam bidang ini.

> **Important**
>
> Akun administrator yang didelegasikan harus menjadi satu-satunya anggota unit organisasi (OU) yang ditetapkan di Organizations.
> Jika akun administrator yang didelegasikan yang Anda daftarkan kemudian dibatalkan pendaftarannya dari peran itu, sistem akan menghapus izinnya untuk mengelola operasi Systems Manager secara bersamaan. Ingatlah bahwa Anda perlu kembali keQuick Setup, menunjuk akun administrator yang didelegasikan yang berbeda, dan tentukan semua fungsi dan izin pekerjaan lagi.
> Jika Anda menggunakan Change Manager di seluruh organisasi, sebaiknya selalu membuat perubahan dari akun administrator yang didelegasikan. Meskipun Anda dapat membuat perubahan dari akun lain di organisasi, perubahan tersebut tidak akan dilaporkan atau dapat dilihat dari akun administrator yang didelegasikan.

5. Di bagian Izin untuk meminta dan membuat perubahan, lakukan hal berikut.

> **Note**
>
> Setiap konfigurasi deployment yang Anda buat menyediakan kebijakan izin hanya untuk satu fungsi tugas. Anda dapat kembali ke Quick Setup nanti untuk membuat lebih banyak fungsi pekerjaan ketika Anda telah membuat template perubahan untuk digunakan dalam operasi Anda.

Untuk membuat peran administratif — Untuk fungsi tugas administrator yang memiliki izin IAM untuk semua tindakan AWS , lakukan hal berikut.

> **Important**
>
> Pemberian izin administratif penuh kepada pengguna harus dilakukan secara terbatas, dan hanya jika peran mereka memerlukan akses penuh Systems Manager. Untuk informasi penting tentang pertimbangan keamanan untuk akses Systems Manager, lihat Identity and access management untuk AWS Systems Manager dan Praktik terbaik keamanan untuk Systems Manager.

1. Untuk Fungsi tugas, masukkan nama untuk mengidentifikasi peran ini dan izinnya, seperti **`MyAWSAdmin`**.
2. Untuk Opsi peran dan izin, pilih Izin administrator.

Untuk membuat fungsi tugas lainnya – Untuk membuat peran non-administratif, lakukan hal berikut:

1. Untuk Fungsi tugas, masukkan nama untuk mengidentifikasi peran ini dan menyarankan izinnya. Nama yang Anda pilih harus mewakili cakupan runbook yang akan Anda berikan izinnya, seperti `DBAdmin` atau `S3Admin`.
2. Untuk Opsi peran dan izin, pilih Izin kustom.
3. Di Editor kebijakan izin, masukkan izin IAM, dalam format JSON, untuk memberikan fungsi tugas ini.

> **Tip**
>
> Kami menyarankan Anda untuk menggunakan editor kebijakan IAM untuk membangun kebijakan Anda lalu menempelkan JSON kebijakan tersebut ke bidang Kebijakan izin.

Contoh kebijakan: manajemen database DynamoDB

Misalnya, Anda mungkin mulai dengan konten kebijakan yang memberikan izin untuk bekerja dengan dokumen Systems Manager (dokumen SSM) yang perlu diakses oleh fungsi tugas. Berikut adalah contoh konten kebijakan yang memberikan akses ke semua runbook Otomasi AWS terkelola yang terkait dengan database DynamoDB dan dua templat perubahan yang telah dibuat dalam sampel Akun AWS `123456789012`, di Wilayah AS Timur (Ohio) (). `us-east-2`

Kebijakan ini juga mencakup izin untuk StartChangeRequestExecutionoperasi, yang diperlukan untuk membuat permintaan perubahan diChange Calendar.

> **Note**
>
> Contoh ini tidak komprehensif. Izin tambahan mungkin diperlukan untuk bekerja dengan AWS sumber daya lain, seperti database dan node.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:CreateDocument",
                "ssm:DescribeDocument",
                "ssm:DescribeDocumentParameters",
                "ssm:DescribeDocumentPermission",
                "ssm:GetDocument",
                "ssm:ListDocumentVersions",
                "ssm:ModifyDocumentPermission",
                "ssm:UpdateDocument",
                "ssm:UpdateDocumentDefaultVersion"
            ],
            "Resource": [
                "arn:aws:ssm:region:*:document/AWS-CreateDynamoDbBackup",
                "arn:aws:ssm:region:*:document/AWS-AWS-DeleteDynamoDbBackup",
                "arn:aws:ssm:region:*:document/AWS-DeleteDynamoDbTableBackups",
                "arn:aws:ssm:region:*:document/AWSConfigRemediation-
DeleteDynamoDbTable",
                "arn:aws:ssm:region:*:document/AWSConfigRemediation-
EnableEncryptionOnDynamoDbTable",
                "arn:aws:ssm:region:*:document/AWSConfigRemediation-
EnablePITRForDynamoDbTable",
                "arn:aws:ssm:region:123456789012:document/MyFirstDBChangeTemplate",
                "arn:aws:ssm:region:123456789012:document/MySecondDBChangeTemplate"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "ssm:ListDocuments",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ssm:StartChangeRequestExecution",
            "Resource": "arn:aws:ssm:region:123456789012:automation-definition/*:*"
        }
    ]
```

```
}
```

Untuk informasi selengkapnya tentang kebijakan IAM, lihat Manajemen akses untuk sumber daya AWS dan Membuat kebijakan IAM di Panduan Pengguna IAM.

6. Di bagian Target, pilih apakah akan memberikan izin untuk fungsi pekerjaan yang Anda buat ke seluruh organisasi atau hanya beberapa unit organisasi Anda.

   Jika Anda memilih Seluruh organisasi, lanjutkan ke langkah 9.

   Jika Anda memilih Kustom, lanjutkan ke langkah 8.

7. Di bagian Target OU, pilih kotak centang unit organisasi yang akan digunakanChange Manager.
8. Pilih Buat.

Setelah sistem selesai menyiapkan Change Manager untuk organisasi Anda, ini akan menampilkan ringkasan penerapan Anda. Informasi ringkasan ini mencakup nama peran yang dibuat untuk fungsi pekerjaan yang Anda konfigurasikan. Misalnya, `AWS-QuickSetup-SSMChangeMgr-DBAdminInvocationRole`.

> **Note**
>
> Quick Setupdigunakan AWS CloudFormation StackSets untuk menyebarkan konfigurasi Anda. Anda juga dapat melihat informasi tentang konfigurasi deployment yang telah selesai di konsol AWS CloudFormation . Untuk selengkapnya StackSets, lihat Bekerja dengan AWS CloudFormation StackSets di Panduan AWS CloudFormation Pengguna.

Langkah Anda selanjutnya adalah mengonfigurasi Change Manager opsi tambahan. Anda dapat menyelesaikan tugas ini baik di akun administrator yang didelegasikan atau akun apa pun di unit organisasi yang diizinkan untuk digunakan. Change Manager Anda mengonfigurasi opsi seperti memilih opsi manajemen identitas pengguna, menentukan pengguna mana yang dapat meninjau dan menyetujui atau menolak templat perubahan dan permintaan perubahan, dan memilih opsi praktik terbaik mana yang diizinkan untuk organisasi Anda. Untuk informasi, lihat MengonfigurasiChange Manager opsi dan praktik terbaik dan praktik terbaik.

MengonfigurasiChange Manager opsi dan praktik terbaik dan praktik terbaik

Tugas di bagian ini harus dilakukan baik Anda menggunakanChange Manager, kemampuan dariAWS Systems Manager, lintas organisasi atau dalam satuAkun AWS.

Jika Anda menggunakanChange Manager untuk organisasi, Anda dapat melakukan tugas berikut di akun administrator yang didelegasikan atau akun apa pun di unit organisasi yang telah Anda izinkan untuk digunakan denganChange Manager.

Topik

- Tugas 1: Mengonfigurasi manajemen identitasChange Manager pengguna dan peninjau templat
- Tugas 2: Mengonfigurasi pemberi persetujuan peristiwa pembekuanChange Manager perubahan dan praktik terbaik Change
- Mengonfigurasi topik Amazon SNS untukChange Managerpemberitahuan

Tugas 1: Mengonfigurasi manajemen identitasChange Manager pengguna dan peninjau templat

Lakukan tugas dalam prosedur ini saat pertama kali Anda mengaksesChange Manager. Anda dapat memperbarui pengaturan konfigurasi ini nanti dengan kembali keChange Manager dan memilih Edit di tab Pengaturan.

Untuk mengonfigurasi manajemen identitasChange Manager pengguna dan peninjau templat pengguna dan peninjau templat pengguna dan peninjau templat pengguna

1. Masuk ke AWS Management Console.

   Jika Anda menggunakanChange Manager untuk organisasi, masuk menggunakan kredensi Anda untuk akun administrator yang didelegasikan. Pengguna harus memiliki izinAWS Identity and Access Management (IAM) yang diperlukan untuk membuat pembaruan padaChange Manager pengaturan Anda.
2. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
3. Di panel navigasi, pilih Change Manager.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Change Manager.
4. Di halaman beranda layanan, tergantung pada opsi yang tersedia, lakukan salah satu hal berikut:

   - Jika Anda menggunakanChange Manager denganAWS Organizations, pilih Siapkan akun yang didelegasikan.

- Jika Anda menggunakanChange Manager dengan satuAkun AWS, pilih AturChange Manager.

  -atau-

  Pilih Buat permintaan perubahan sampel, Lewati, lalu pilih tab Pengaturan.

5. Untuk Manajemen identitas pengguna, pilih salah satu dari berikut ini.

   - AWS Identity and Access Management(IAM) — Mengidentifikasi pengguna yang membuat dan menyetujui permintaan dan melakukan tindakan lain dengan menggunakan pengguna, grup, grup, dan peran lain dengan menggunakan pengguna, grup, dan peran lainChange Manager dengan menggunakan pengguna, grup, grup, dan peran lain dengan menggunakan pengguna, grup, grup, dan peran yang ada.
   - AWS IAM Identity Center(IAM Identity Center) — Memungkinkan IAM Identity Center untuk membuat dan mengelola identitas, atau menghubungkan ke sumber identitas yang ada untuk mengidentifikasi pengguna yang melakukan tindakan diChange Manager.

6. Di bagian Pemberitahuan peninjau Template, tentukan topik Amazon Simple Notification Service (Amazon SNS) yang akan digunakan untuk memberi tahu pengulas template bahwa template perubahan baru atau versi template perubahan siap untuk ditinjau. Pastikan topik Amazon SNS yang Anda pilih dikonfigurasi untuk mengirim notifikasi ke peninjau templat Anda.

   Untuk informasi tentang membuat dan mengonfigurasi topik Amazon SNS untuk pemberitahuan peninjau templat perubahan, lihat Mengonfigurasi topik Amazon SNS untukChange Managerpemberitahuan.

   1. Untuk menentukan topik Amazon SNS untuk notifikasi peninjau templat, pilih salah satu opsi berikut:

      - Masukkan SNS Amazon Resource Name (ARN) – Untuk ARN Topik, masukkan ARN topik Amazon SNS yang ada. Topik ini dapat berada di salah satu akun organisasi Anda.
      - Pilih topik SNS yang ada – Untuk Topik notifikasi target, pilih ARN topik Amazon SNS yang ada di Akun AWS Anda saat ini. (Opsi ini tidak tersedia jika Anda belum membuat topik Amazon SNS apa pun di Akun AWS dan Wilayah AWS Anda saat ini.)

      > **Note**
      >
      > Topik Amazon SNS yang Anda pilih harus dikonfigurasi untuk menentukan notifikasi yang dikirim dan pelanggan yang menerimanya. Kebijakan aksesnya juga harus memberikan izin kepada Systems Manager sehinggaChange Manager dapat

mengirim notifikasi. Untuk informasi, lihat Mengonfigurasi topik Amazon SNS untukChange Managerpemberitahuan.

2. Pilih Tambahkan notifikasi.

7. Di bagian Ubah peninjau templat, pilih pengguna di organisasi atau akun Anda untuk meninjau templat perubahan baru atau mengubah versi templat sebelum dapat digunakan dalam operasi Anda.

   Peninjau templat Change bertanggung jawab untuk memverifikasi kesesuaian dan keamanan templat Change bertanggung jawab untuk digunakan dalam alur kerjaChange Manager runbook.

   Pilih ubah peninjau templat dengan melakukan hal berikut:

   1. Pilih Tambahkan.
   2. Pilih kotak centang di samping nama setiap peran pengguna, grup, atau peran IAM yang ingin Anda tetapkan sebagai peninjau templat perubahan.
   3. Pilih Tambahkan pemberi persetujuan.

8. Pilih Kirim.

Setelah Anda menyelesaikan proses pengaturan awal ini, konfigurasikanChange Manager pengaturan tambahan dan praktik terbaik dan praktik terbaik dan praktik terbaik dan praktik terbaik dan praktik terbaik dan praktik terbaik dan praktik terbaik dan praktik terbaik dan praktik terbaikTugas 2: Mengonfigurasi pemberi persetujuan peristiwa pembekuanChange Manager perubahan dan praktik terbaik Change

Tugas 2: Mengonfigurasi pemberi persetujuan peristiwa pembekuanChange Manager perubahan dan praktik terbaik Change

Setelah Anda menyelesaikan langkah-langkah dalamTugas 1: Mengonfigurasi manajemen identitasChange Manager pengguna dan peninjau templat, Anda dapat menunjuk peninjau tambahan untuk permintaan perubahan selama peristiwa pembekuan perubahan dan menentukan praktik terbaik yang tersedia yang ingin Anda izinkan untukChange Manager operasi Anda.

Sebuah peristiwa pembekuan perubahan berarti bahwa pembatasan berlaku di kalender perubahan saat ini (status kalender diAWS Systems ManagerChange Calendar adalah`CLOSED`). Dalam kasus ini, selain pemberi persetujuan reguler untuk permintaan perubahan, jika permintaan perubahan dibuat menggunakan templat yang memungkinkan persetujuan otomatis, pemberi persetujuan

pembekuan perubahan harus memberikan izin agar permintaan perubahan ini berjalan. Jika tidak, perubahan tidak akan diproses hingga status kalender menjadi status kalender kembali0PEN.

Untuk mengonfigurasi pemberi persetujuan peristiwa pembekuanChange Manager perubahan dan praktik terbaik Change

1. Di panel navigasi, pilih Change Manager.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Change Manager.
2. Pilih tab Pengaturan, dan kemudian pilih Edit.
3. Di bagian Pemberi persetujuan peristiwa pembekuan perubahan, pilih pengguna di organisasi atau akun Anda yang dapat menyetujui perubahan untuk dijalankan meskipun status kalender yang digunakan di saat iniChange Calendar adalah TUTUP.

   > **Note**
   >
   > Untuk mengizinkan peninjauan pembekuan perubahan, Anda harus mengaktifkan opsi Periksa Change Calendar untuk peristiwa perubahan terbatas di Praktik terbaik.

   Pilih pemberi persetujuan untuk peristiwa pembekuan perubahan dengan melakukan hal berikut:

   1. Pilih Tambahkan.
   2. Pilih kotak centang di samping nama setiap peran pengguna, grup, atau peran IAM yang ingin Anda tetapkan sebagai pemberi persetujuan untuk peristiwa pembekuan perubahan.
   3. Pilih Tambahkan pemberi persetujuan.
4. Di bagian Praktik terbaik di dekat bagian bawah halaman, aktifkan praktik terbaik yang ingin Anda terapkan untuk setiap opsi berikut.

   - Pilihan: Periksa Change Calendar untuk peristiwa perubahan terbatas

     Untuk menentukan bahwaChange Manager memeriksa kalender guna memastikan perubahan tidak diblokir oleh peristiwa terjadwal, pilih kotak centang Diaktifkan terlebih dahulu, lalu pilih kalender untuk memeriksa peristiwa terbatas dari daftar Change Calendar.Change Calendar

Untuk informasi selengkapnya tentang Change Calendar, lihat AWS Systems Manager Change Calendar.

- Opsi: Topik SNS untuk pemberi persetujuan untuk peristiwa tertutup

  1. Pilih salah satu dari berikut ini untuk menentukan topik Amazon Simple Notification Service (Amazon SNS) di akun Anda yang akan digunakan untuk mengirim notifikasi ke pemberi persetujuan selama peristiwa pembekuan perubahan. (Ingat bahwa Anda juga harus menentukan pemberi persetujuan di bagian Pemberi persetujuan untuk peristiwa pembekuan perubahan di atas Praktik terbaik.)

     - Masukkan SNS Amazon Resource Name (ARN) – Untuk ARN Topik, masukkan ARN topik Amazon SNS yang ada. Topik ini dapat berada di salah satu akun organisasi Anda.
     - Pilih topik SNS yang ada – Untuk Topik notifikasi target, pilih ARN topik Amazon SNS yang ada di Akun AWS Anda saat ini. (Opsi ini tidak tersedia jika Anda belum membuat topik Amazon SNS apa pun di Akun AWS dan Wilayah AWS Anda saat ini.)

     > **Note**
     >
     > Topik Amazon SNS yang Anda pilih harus dikonfigurasi untuk menentukan notifikasi yang dikirim dan pelanggan yang menerimanya. Kebijakan aksesnya juga harus memberikan izin kepada Systems Manager sehinggaChange Manager dapat mengirim notifikasi. Untuk informasi, lihat Mengonfigurasi topik Amazon SNS untukChange Managerpemberitahuan.

  2. Pilih Tambahkan notifikasi.

- Opsi: Memerlukan pemantauan untuk semua templat

  Jika Anda ingin memastikan bahwa semua templat untuk organisasi atau akun Anda menentukan CloudWatch alarm Amazon untuk memantau operasi perubahan Anda, pilih kotak centang Diaktifkan.

- Opsi: Memerlukan peninjauan dan persetujuan templat sebelum digunakan

  Untuk memastikan bahwa tidak ada permintaan perubahan yang dibuat, dan tidak ada alur kerja runbook yang berjalan, tanpa didasarkan pada templat yang telah ditinjau dan disetujui, pilih kotak centang Diaktifkan.

5. Pilih Simpan.

Mengonfigurasi topik Amazon SNS untukChange Managerpemberitahuan

Anda dapat mengonfigurasiChange Manager, suatu kemampuanAWS Systems Manager, untuk mengirim notifikasi ke topik Amazon Simple Notification Service (Amazon SNS) untuk peristiwa yang terkait dengan permintaan perubahan dan templat perubahan. Lengkapi tugas-tugas berikut untuk menerima notifikasi untukChange Manageracara yang Anda tambahkan topik.

Topik

- Tugas 1: Membuat dan berlangganan topik Amazon SNS
- Tugas 2: Memperbarui kebijakan akses Amazon SNS
- Tugas 3: (Opsional) MemperbaruiAWS Key Management Servicekebijakan akses

Tugas 1: Membuat dan berlangganan topik Amazon SNS

Pertama, Anda harus membuat dan berlangganan ke topik Amazon SNS. Untuk informasi selengkapnya, lihatMembuat topik Amazon SNSdanBerlangganan topik Amazon SNSdi dalamPanduan Developer Amazon Simple Notification Service.

> **Note**
>
> Untuk menerima notifikasi, Anda harus menentukan Amazon Resource Name (ARN) dari topik Amazon SNS yang ada di Wilayah AWS dan Akun AWS yang sama dengan admin yang didelegasikan.

Tugas 2: Memperbarui kebijakan akses Amazon SNS

Gunakan prosedur berikut untuk memperbarui kebijakan akses Amazon SNS sehingga Systems Manager dapat mempublikasikanChange Managernotifikasi untuk topik Amazon SNS yang Anda buat dalam Tugas 1. Tanpa menyelesaikan tugas ini,Change Managertidak memiliki izin untuk mengirim notifikasi untuk peristiwa yang Anda tambahkan topiknya.

1. Masuk ke AWS Management Console dan buka konsol Amazon SNS di https://console.aws.amazon.com/sns/v3/home.
2. Di panel navigasi, pilih Topik.
3. Pilih topik yang Anda buat di Tugas 1, lalu pilih Edit.
4. Perluas Kebijakan akses.

5. Tambahkan dan perbarui yang berikut`Sid`blok ke kebijakan yang ada dan ganti masing-masing*placeholder masukan pengguna*dengan informasi Anda sendiri.

```
{
    "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"
            ]
        }
    }
}
```

   Masukkan blok ini setelah yang ada`Sid`blok, dan ganti*daerah*,*account-id*, dan*topik-nama*dengan nilai yang sesuai untuk topik yang Anda buat.

6. Pilih Simpan perubahan.

Sistem sekarang mengirimkan notifikasi ke topik Amazon SNS saat jenis peristiwa yang Anda tambahkan ke topik terjadi.

> **Important**
>
> Jika Anda mengonfigurasi topik Amazon SNS dengan kunci enkripsi sisi server AWS Key Management Service (AWS KMS), Anda harus menyelesaikan Tugas 3.

Tugas 3: (Opsional) MemperbaruiAWS Key Management Servicekebijakan akses

Jika Anda mengaktifkan enkripsi sisi server AWS Key Management Service (AWS KMS) untuk topik Amazon SNS Anda, Anda juga harus memperbarui kebijakan akses AWS KMS key yang Anda pilih saat mengonfigurasi topik. Gunakan prosedur berikut untuk memperbarui kebijakan akses

sehingga Systems Manager dapat mempublikasikanChange Managernotifikasi persetujuan untuk topik Amazon SNS yang Anda buat dalam Tugas 1.

1. Buka konsol AWS KMS tersebut di https://console.aws.amazon.com/kms.
2. Di panel navigasi, pilih Kunci yang dikelola pelanggan.
3. Pilih ID kunci yang dikelola pelanggan yang Anda pilih saat membuat topik.
4. Di bagian Kebijakan Kunci, pilih Beralih ke tampilan kebijakan.
5. Pilih Edit.
6. Masukkan perintah berikut`Sid`blok setelah salah satu`Sid`blok dalam kebijakan yang ada. Ganti masing-masing*`placeholder masukan pengguna`*dengan informasi Anda sendiri.

```
{
    "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. Sekarang masukkan perintah berikut`Sid`blok setelah salah satu`Sid`blok dalam kebijakan sumber daya untuk membantu mencegahmasalah lintas layanan bingung.

   Blok ini menggunakan`aws:SourceArn`dan`aws:SourceAccount`kunci konteks kondisi global untuk membatasi izin yang diberikan Systems Manager layanan lain ke sumber daya.

   Ganti masing-masing*`placeholder masukan pengguna`*dengan informasi Anda sendiri.

```
{
```

```
  "Version": "2008-10-17",
  "Statement": [
    {
      "Sid": "Configure confused deputy protection for AWS KMS keys used in Amazon
 SNS topic when called from Systems Manager",
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": [
        "sns:Publish"
      ],
      "Resource": "arn:aws:sns:region:account-id:topic-name",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:ssm:region:account-id:*"
        },
        "StringEquals": {
          "aws:SourceAccount": "account-id"
        }
      }
    }
  ]
}
```

8. Pilih Simpan perubahan.

Mengonfigurasi peran dan izin untukChange Manager

Secara default,Change Manager tidak memiliki izin untuk melakukan tindakan pada sumber daya Anda. Anda harus memberikan akses dengan menggunakan peran layananAWS Identity and Access Management (IAM), atau peran asumsi. Peran ini memungkinkanChange Manager untuk menjalankan alur kerja runbook dengan aman yang ditentukan dalam permintaan perubahan yang disetujui atas nama Anda. Peran memberikanAWS Security Token Service (AWS STS) AssumeRolekepercayaan kepadaChange Manager.

Dengan memberikan izin ini kepada peran untuk bertindak atas nama pengguna dalam organisasi, pengguna tidak perlu diberikan array izin itu sendiri. Tindakan yang diizinkan oleh izin terbatas pada operasi yang disetujui saja.

Saat pengguna di akun atau organisasi Anda membuat permintaan perubahan, mereka dapat memilih peran asumsikan ini untuk melakukan operasi perubahan.

Anda dapat membuat peran asumsi baru untukChange Manager atau memperbarui peran yang sudah ada dengan izin yang diminta.

Untuk membuat peran layananChange Manager, selesaikan tugas berikut.

Tugas

- Tugas 1: Membuat kebijakan peran asumsikan untukChange Manager
- Tugas 2: Membuat peran asumsikan untukChange Manager
- Tugas 3: Melampirkaniam:PassRole kebijakan ke peran lain
- Tugas 4: Menambahkan kebijakan inline ke peran asumsikan untuk memanggil yang lainLayanan AWS
- Tugas 5: Mengkonfigurasi akses pengguna keChange Manager

Tugas 1: Membuat kebijakan peran asumsikan untukChange Manager

Gunakan prosedur berikut untuk membuat kebijakan yang akan Anda lampirkan pada peranChange Manager asumsi Anda.

Untuk membuat kebijakan peran asumsi untukChange Manager

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Pada panel navigasi, pilih Kebijakan, lalu pilih Buat kebijakan.
3. Pada halaman Buat kebijakan, pilih tab JSON dan ganti konten default dengan yang berikut ini, yang akan Anda modifikasi untukChange Manager operasi Anda sendiri dalam langkah-langkah berikut.

> **Note**
>
> Jika Anda membuat kebijakan untuk digunakan dengan satuAkun AWS, dan bukan organisasi dengan beberapa akun danWilayah AWS, Anda dapat menghilangkan blok pernyataan pertama. `iam:PassRole`Izin tidak diperlukan dalam kasus penggunaan satu akunChange Manager.

```
{
    "Version": "2012-10-17",
```

```
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::delegated-admin-account-id:role/AWS-
SystemsManager-job-functionAdministrationRole",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "ssm.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeDocument",
                "ssm:GetDocument",
                "ssm:StartChangeRequestExecution"
            ],
            "Resource": [
                "arn:aws:ssm:region:account-id:automation-definition/template-name:
$DEFAULT",
                "arn:aws:ssm:region::document/template-name"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:ListOpsItemEvents",
                "ssm:GetOpsItem",
                "ssm:ListDocuments",
                "ssm:DescribeOpsItems"
            ],
            "Resource": "*"
        }
    ]
}
```

4. Untuk`iam:PassRole` tindakan tersebut, perbarui`Resource` nilai untuk menyertakan ARN dari semua fungsi pekerjaan yang ditentukan untuk organisasi Anda yang ingin Anda berikan izin untuk memulai alur kerja runbook.
5. Ganti placeholder *`region`*, *`account-id delegated-admin-account-id`*, *`template-name`*, dan *`job-function`* dengan nilai untukChange Manager operasi Anda.

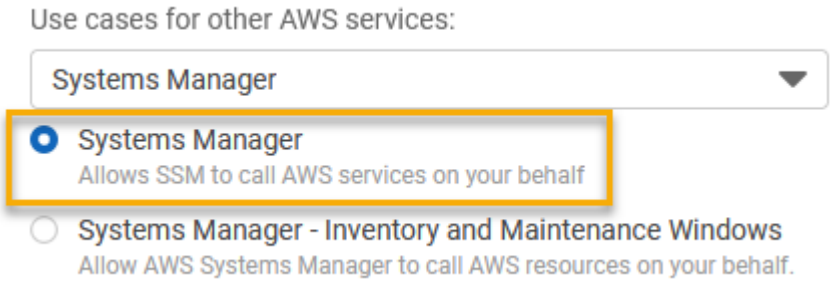
6. Untuk`Resource` pernyataan kedua, ubah daftar untuk menyertakan semua template perubahan yang ingin Anda berikan izin. Atau, tentukan`"Resource": "*"` untuk memberikan izin untuk semua templat perubahan di organisasi Anda.
7. Pilih Next: Tags (Selanjutnya: Tanda).
8. (Opsional) Tambahkan satu atau beberapa pasangan nilai kunci tag untuk mengatur, atau mengontrol akses untuk kebijakan ini.
9. Pilih Next: Review (Selanjutnya: Tinjauan).
10. Pada halaman Kebijakan ulasan, masukkan nama di kotak Nama, seperti**`MyChangeManagerAssumeRole`**, lalu masukkan deskripsi opsional.
11. Pilih Buat kebijakan, dan lanjutkan keTugas 2: Membuat peran asumsikan untukChange Manager.

Tugas 2: Membuat peran asumsikan untukChange Manager

Gunakan prosedur berikut untuk membuat peranChange Manager asumsi, jenis peran layanan, untukChange Manager.

Untuk membuat peran asumsikan untukChange Manager

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran, lalu pilih Buat peran.
3. Untuk Pilih entitas tepercaya, buat pilihan berikut:
    1. Untuk tipe entitas tepercaya, pilih AWSlayanan
    2. Untuk Kasus penggunaan untuk yang lainLayanan AWS, pilih Systems Manager
    3. Pilih Systems Manager, seperti yang ditunjukkan pada gambar berikut.



4. Pilih Selanjutnya.

5. Di halaman kebijakan Izin terlampir, cari kebijakan peran asumsikan yang Anda buatTugas 1: Membuat kebijakan peran asumsikan untukChange Manager, misalnya**MyChangeManagerAssumeRole**.
6. Pilih kotak centang di samping nama kebijakan peran asumsi, lalu pilih Berikutnya: Tag.
7. Untuk Nama peran, masukkan nama untuk profil instans baru Anda, seperti**MyChangeManagerAssumeRole**.
8. (Opsional) Untuk Deskripsi, perbarui deskripsi untuk peran instans ini.
9. (Opsional) Tambahkan satu atau beberapa pasangan nilai kunci tag untuk mengatur, atau mengontrol akses untuk peran ini.
10. Pilih Next: Review (Selanjutnya: Tinjauan).
11. (Opsional) Untuk Tag, tambahkan satu atau beberapa pasangan nilai kunci tag untuk mengatur, atau mengontrol akses untuk peran ini, lalu pilih Buat peran. Sistem mengembalikan Anda ke halaman Peran.
12. Pilih Buat peran. Sistem mengembalikan Anda ke halaman Peran.
13. Pada halaman Peran, pilih peran yang baru Anda buat untuk membuka halaman Ringkasan.

Tugas 3: Melampirkan**iam:PassRole** kebijakan ke peran lain

Gunakan prosedur berikut untuk melampirkan`iam:PassRole` kebijakan ke profil instans IAM atau peran layanan IAM. (Layanan Systems Manager menggunakan profil instans IAM untuk berkomunikasi dengan instans EC2. Untuk node yang dikelola non-EC2 di lingkungan hybrid dan multicloud, peran layanan IAM digunakan sebagai gantinya.)

Dengan melampirkan`iam:PassRole` kebijakan,Change Manager layanan dapat lulus izin peran asumsi izin peran untuk layanan lain atau kemampuan Systems Manager ketika menjalankan alur kerja runbook.

Untuk melampirkan**iam:PassRole** kebijakan ke profil instans IAM atau peran layanan

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran.
3. Cari peranChange Manager asumsikan yang Anda buat, misalnya**MyChangeManagerAssumeRole**, dan pilih namanya.
4. Di halaman Ringkasan untuk peran asumsikan, pilih tab Izin.
5. Pilih Tambahkan izin, Buat kebijakan sebaris.

6. Di halaman Buat kebijakan, pilih tab Visual editor.
7. Pilih Layanan, lalu pilih IAM.
8. Di kotak teks Tindakan filter, masukkan**`PassRole`**, lalu pilih PassRoleopsi.
9. Perluas Sumber Daya. Verifikasi bahwa Spesifik dipilih, dan kemudian pilih Tambahkan ARN.
10. Di bidang Tentukan ARN untuk peran, masukkan ARN peran profil instans IAM atau peran layanan IAM yang ingin Anda lewati dengan izin peran asumsi. Sistem mengisi Akun dan Nama peran dengan bidang jalur.
11. Pilih Tambahkan.
12. Pilih Tinjau kebijakan.
13. Untuk Nama, masukkan nama untuk mengidentifikasi kebijakan ini, lalu pilih Buat kebijakan.

Info lebih lanjut

- Mengonfigurasi izin instans untuk Systems Manager Systems Manager
- Membuat peran layanan IAM untuk lingkungan hibrid

Tugas 4: Menambahkan kebijakan inline ke peran asumsikan untuk memanggil yang lainLayanan AWS

Ketika permintaan perubahan menjalankan yang lainLayanan AWS dengan menggunakan peranChange Manager asumsi, peran asumsi harus dikonfigurasi dengan izin untuk menjalankan layanan tersebut. Persyaratan ini berlaku untuk semua runbookAWS Otomasi (runbook AWS-*) yang mungkin digunakan dalam permintaan perubahan, sepertiAWS-ConfigureS3BucketLogging,AWS-CreateDynamoDBBackup, danAWS-RestartEC2Instance runbook. Persyaratan ini juga berlaku untuk setiap runbook kustom yang Anda buat yang menjalankan yang lainLayanan AWS dengan menggunakan tindakan yang memanggil layanan lainnya. Misalnya, jika Anda menggunakan`aws:executeAwsApi`,`aws:CreateStack`, atau`aws:copyImage` tindakan, maka Anda harus mengonfigurasi peran layanan dengan izin untuk menjalankan layanan tersebut. Anda dapat mengaktifkan izin ke pihakLayanan AWS dengan menambahkan kebijakan inline IAM ke peran tersebut.

Untuk menambahkan kebijakan inline ke peran asumsikan untuk memanggil lainnyaLayanan AWS (konsol IAM)

1. Masuk ke AWS Management Console dan buka konsol IAM di https://console.aws.amazon.com/iam/.

2. Di panel navigasi, pilih Peran.
3. Dalam daftar peran asumsi, pilih nama peran asumsi yang ingin Anda perbarui, sepertiMyChangeManagerAssumeRole.
4. Pilih tab Izin.
5. Pilih Tambahkan izin, Buat kebijakan sebaris.
6. Pilih tab JSON.
7. Masukkan dokumen kebijakan JSON untuk yang inginLayanan AWS Anda panggil. Berikut adalah dua contoh dokumen kebijakan JSON.

   Amazon S3**PutObject** dan**GetObject** contoh

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject"
            ],
            "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
        }
    ]
}
```

   Amazon EC2**CreateSnapshot** dan**DescribeSnapShots** contoh

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"ec2:CreateSnapshot",
         "Resource":"*"
      },
      {
         "Effect":"Allow",
         "Action":"ec2:DescribeSnapshots",
         "Resource":"*"
```

```
        }
    ]
}
```

Untuk rincian bahasa kebijakan IAM, lihat referensi kebijakan IAM JSON di Panduan Pengguna IAM.

8. Setelah selesai, pilih Tinjau kebijakan. Validator Kebijakan melaporkan kesalahan sintaksis.
9. Untuk Nama, masukkan nama untuk mengidentifikasi kebijakan yang Anda buat. Ulas Ringkasan kebijakan untuk melihat izin yang diberikan oleh kebijakan Anda. Kemudian pilih Buat kebijakan untuk menyimpan pekerjaan Anda.
10. Setelah Anda membuat kebijakan yang selaras, ia akan secara otomatis tertanam di pengguna atau peran Anda.

Tugas 5: Mengkonfigurasi akses pengguna keChange Manager

Jika pengguna, grup, atau peran Anda diberikan izin administrator, maka Anda memiliki akses keChange Manager. Jika Anda tidak memiliki izin administrator, maka administrator harus menetapkan kebijakan`AmazonSSMFullAccess` terkelola, atau kebijakan yang menyediakan izin yang sebanding, ke pengguna, grup, atau peran Anda.

Gunakan prosedur berikut untuk mengonfigurasi pengguna untuk menggunakanChange Manager. Pengguna yang Anda pilih akan memiliki izin untuk mengonfigurasi dan menjalankanChange Manager.

Bergantung pada aplikasi identitas yang Anda gunakan di organisasi, Anda dapat memilih salah satu dari tiga opsi yang tersedia untuk mengonfigurasi akses pengguna. Saat mengkonfigurasi akses pengguna, tetapkan atau tambahkan yang berikut ini:

1. Tetapkan`AmazonSSMFullAccess` kebijakan atau kebijakan sebanding yang memberikan izin untuk mengakses Systems Manager.
2. Tetapkan`iam:PassRole` kebijakan.
3. Tambahkan ARN untuk peranChange Manager asumsikan yang Anda salin di akhirTugas 2: Membuat peran asumsikan untukChange Manager.

Untuk menyediakan akses, tambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup diAWS IAM Identity Center:

Buat set izin. Ikuti petunjuk di Buat set izin di PanduanAWS IAM Identity Center Pengguna.

- Pengguna yang dikelola dalam IAM melalui penyedia identitas:

  Buat peran untuk federasi identitas. Ikuti petunjuk dalam Membuat peran untuk penyedia identitas pihak ketiga (federasi) di Panduan Pengguna IAM.

- Pengguna IAM:
  - Buat peran yang dapat diasumsikan pengguna Anda. Ikuti petunjuk dalam Membuat peran untuk pengguna IAM di Panduan Pengguna IAM.
  - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk dalam Menambahkan izin ke pengguna (konsol) di Panduan Pengguna IAM.

Anda telah selesai mengonfigurasi peran yang diminta untukChange Manager. Anda sekarang dapat menggunakan peranChange Manager asumsi ARN dalamChange Manager operasi Anda.

Mengontrol akses ke alur kerja runbook persetujuan otomatis

Di setiap template perubahan yang dibuat untuk organisasi atau akun Anda, Anda dapat menentukan apakah permintaan perubahan yang dibuat dari template tersebut dapat berjalan sebagai permintaan perubahan yang disetujui secara otomatis, yang berarti bahwa mereka berjalan secara otomatis tanpa langkah peninjauan (dengan pengecualian peristiwa pembekuan perubahan).

Namun, Anda mungkin ingin mencegah pengguna, grup, atauAWS Identity and Access Management(IAM) peran dari menjalankan permintaan perubahan yang disetujui secara otomatis meskipun template perubahan mengizinkannya. Anda dapat melakukan ini melalui penggunaan`ssm:AutoApprove`kunci kondisi untuk`StartChangeRequestExecution`operasi dalam kebijakan IAM yang ditetapkan untuk pengguna, grup, atau peran IAM.

Anda dapat menambahkan kebijakan berikut sebagai kebijakan inline, di mana kondisi ditetapkan sebagai`false`, untuk mencegah pengguna menjalankan permintaan perubahan yang dapat disetujui secara otomatis.

```
{
    "Version": "2012-10-17",
    "Statement": [
            {
            "Effect": "Allow",
```

```
            "Action": "ssm:StartChangeRequestExecution",
            "Resource": "*",
            "Condition": {
                "BoolIfExists": {
                    "ssm:AutoApprove": "false"
                }
            }
        }
    ]
}
```

Untuk informasi tentang menentukan kebijakan sebaris, lihatKebijakan inlinedanMenambahkan dan menghapus izin identitas IAMdi dalamPanduan Pengguna IAM.

Untuk informasi lebih lanjut tentang kunci kondisi untuk Systems Manager kebijakan Systems, lihatKunci kondisi Systems Manager.

Bekerja dengan Change Manager

DenganChange Manager, suatu kemampuanAWS Systems Manager, pengguna di seluruh organisasi Anda atau dalam satuAkun AWSdapat melakukan tugas terkait perubahan yang telah diberikan izin yang diperlukan kepada mereka.Change Managertugas yang dimaksud meliputi:

- Membuat, meninjau, dan menyetujui atau menolak templat perubahan.

  Template perubahan adalah kumpulan pengaturan konfigurasi diChange Manageryang menentukan hal-hal seperti persetujuan yang diperlukan, runbook yang tersedia, dan opsi notifikasi untuk permintaan perubahan.
- Buat, tinjau, dan setujui atau tolak permintaan perubahan.

  Permintaan perubahan adalah permintaan diChange Manageruntuk menjalankan runbook Otomatisasi yang memperbarui satu atau beberapa sumber daya diAWSatau lingkungan on-premise. Permintaan perubahan dibuat menggunakan templat perubahan.
- Tentukan pengguna mana di organisasi atau akun Anda yang dapat dijadikan peninjau untuk templat perubahan dan permintaan perubahan.
- Mengedit pengaturan konfigurasi, seperti bagaimana identitas pengguna dikelolaChange Managerdan mana yang tersediapraktik terbaikopsi diberlakukan diChange Manageroperasi. Untuk informasi tentang mengonfigurasi pengaturan ini, lihat MengonfigurasiChange Manager opsi dan praktik terbaik dan praktik terbaik.

Topik

Bekerja dengan templat perubahan

Template perubahan adalah kumpulan pengaturan konfigurasi diChange Manageryang menentukan hal-hal seperti persetujuan yang diperlukan, runbook yang tersedia, dan opsi notifikasi untuk permintaan perubahan.

> **Note**
>
> AWSmenyediakan sampelHalo Duniamengubah template yang dapat Anda gunakan untuk mencobaChange Manager, kemampuanAWS Systems Manager. Namun, Anda membuat templat perubahan sendiri untuk menentukan perubahan yang ingin Anda izinkan ke sumber daya di organisasi atau akun Anda.

Perubahan yang dibuat saat alur kerja runbook berjalan didasarkan pada konten runbook Otomatisasi. Di setiap templat perubahan yang Anda buat, Anda dapat menyertakan satu atau beberapa runbook otomatisasi yang dapat dipilih oleh pengguna yang membuat permintaan perubahan untuk dijalankan selama pembaruan. Anda juga dapat membuat templat perubahan yang memungkinkan peminta memilih runbook Otomatisasi yang tersedia untuk permintaan perubahan.

Untuk membuat templat perubahan, Anda dapat menggunakan opsi Builder dalam halaman konsol Buat templat untuk membangun templat perubahan. Atau, menggunakan opsi Editor, Anda dapat menulis konten JSON atau YAML secara manual dengan konfigurasi yang Anda inginkan untuk alur kerja runbook Anda. Anda juga dapat menggunakan alat baris perintah untuk membuat templat perubahan, dengan konten JSON untuk template perubahan yang disimpan dalam file eksternal.

Topik

- Menghapus templat perubahan

Coba template**`Hello World`** perubahanAWS terkelola

Gunakan sampel templat perubahan`AWS-HelloWorldChangeTemplate`, yang menggunakan sampel runbook Otomatisasi`AWS-HelloWorld`, untuk menguji proses peninjauan dan persetujuan setelah Anda selesai menyiapkanChange Manager, kemampuan dariAWS Systems Manager. Templat ini dirancang untuk menguji atau memverifikasi izin yang dikonfigurasi, penetapan pemberi persetujuan, dan proses persetujuan Anda. Persetujuan untuk menggunakan templat perubahan ini di organisasi atau akun Anda telah disediakan oleh AWS. Namun, setiap permintaan perubahan berdasarkan templat perubahan ini harus tetap disetujui oleh peninjau di organisasi atau akun Anda.

Daripada membuat perubahan pada sumber daya, hasil dari alur kerja runbook yang terkait dengan templat ini adalah untuk mencetak pesan dalam output dari langkah Otomatisasi.

Sebelum Anda memulai

Sebelum Anda memulai, pastikan Anda telah menyelesaikan tugas berikut:

- Jika Anda menggunakan AWS Organizations untuk mengelola perubahan di seluruh organisasi, selesaikan tugas penyiapan organisasi yang dijelaskan di Menyiapkan Change Manager untuk organisasi (akun manajemen).
- KonfigurasikanChange Manager untuk akun administrator atau akun tunggal Anda yang didelegasikan, seperti yang dijelaskan dalamMengonfigurasiChange Manager opsi dan praktik terbaik dan praktik terbaik.

> **Note**
>
> Jika Anda mengaktifkan opsi praktik terbaik Memerlukan monitor untuk semua templat diChange Manager pengaturan Anda, nonaktifkan untuk sementara saat Anda menguji templat perubahan Hello World.

Untuk mencoba templat perubahan Hello World yang dikelola AWS

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Change Manager.

   -atau-

JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Change Manager.

3. Pilih Buat permintaan.
4. Pilih templat perubahan yang bernama `AWS-HelloWorldChangeTemplate`, lalu pilih Selanjutnya.
5. Untuk Nama, masukkan nama untuk permintaan perubahan yang membuat tujuannya mudah diidentifikasi, seperti **`MyChangeRequestTest`**.
6. Untuk langkah-langkah selanjutnya dalam membuat permintaan perubahan, lihat Membuat permintaan perubahan.

Langkah selanjutnya

Untuk informasi tentang menyetujui permintaan perubahan, lihat Meninjau dan menyetujui atau menolak permintaan perubahan.

Untuk melihat status dan hasil permintaan perubahan Anda, pilih nama permintaan perubahan di Change di Change di ChangeChange Manager.

Membuat templat perubahan

Templat perubahan adalah kumpulan pengaturan konfigurasi yang menentukan hal-hal seperti persetujuanChange Manager yang diperlukan, runbook yang tersedia, dan opsi notifikasi untuk permintaan perubahan.

Anda dapat membuat templat perubahan untuk operasi Anda diChange Manager, kemampuan dariAWS Systems Manager, menggunakan, yang mencakup opsi Builder dan Editor, atau alat baris perintah.

Topik

- Tentang persetujuan dalam template perubahan Anda
- Membuat templat perubahan menggunakan Builder
- Membuat templat perubahan menggunakan Editor
- Membuat templat perubahan menggunakan alat baris perintah

Tentang persetujuan dalam template perubahan Anda

Untuk setiap template perubahan yang Anda buat, Anda dapat menentukan hingga lima tingkat persetujuan untuk permintaan perubahan yang dibuat darinya. Untuk masing-masing level tersebut, Anda dapat menunjuk hingga lima pemberi persetujuan potensif. Penyetuju tidak terbatas pada satu pengguna. Anda juga dapat menentukan grup IAM atau peran IAM sebagai pemberi persetujuan individual. Untuk grup IAM dan peran IAM, satu atau beberapa pengguna yang termasuk dalam grup atau peran dapat memberikan persetujuan untuk menerima jumlah total persetujuan yang diperlukan untuk permintaan perubahan. Anda juga dapat menentukan lebih banyak pemberi persetujuan daripada yang diperlukan template perubahan Anda.

Change Managermendukung dua pendekatan utama untuk persetujuan: persetujuan per tingkat dan persetujuan per baris. Kombinasi kedua jenis ini juga dimungkinkan dalam beberapa situasi. Sebaiknya gunakan hanya persetujuan per level dalamChange Manager operasi Anda.

Per-level approvals

Direkomendasikan. Pada 23 Januari 2023,Change Manager mendukung persetujuan per level. Dalam model ini, untuk setiap tingkat persetujuan dalam template perubahan Anda, Anda terlebih dahulu menentukan berapa banyak persetujuan yang diperlukan untuk tingkat tersebut. Kemudian Anda menentukan setidaknya bahwa banyak pemberi persetujuan untuk tingkat dan dapat menentukan lebih pemberi persetujuan. Namun, hanya jumlah pemberi persetujuan per level yang Anda tentukan yang perlu menyetujui permintaan perubahan. Misalnya, Anda dapat menentukan lima pemberi persetujuan tetapi memerlukan tiga persetujuan.

Untuk tampilan konsol dan sampel JSON dari jenis persetujuan ini, lihatthe section called “Contoh konfigurasi persetujuan per tingkat”.

Per-line approvals

Didukung untuk kompatibilitas mundur. Rilis asli hanyaChange Manager didukung persetujuan per baris. Dalam model ini, setiap pemberi persetujuan yang ditentukan untuk tingkat persetujuan direpresentasikan sebagai garis persetujuan. Setiap pemberi persetujuan harus menyetujui permintaan perubahan agar disetujui pada tingkat tersebut. Sebelum 23 Januari 2023, ini adalah satu-satunya model yang didukung untuk persetujuan. Ubah template yang dibuat sebelum tanggal ini terus mendukung persetujuan per baris, tetapi sebaiknya gunakan persetujuan per level sebagai gantinya.

Untuk tampilan konsol dan sampel JSON dari jenis persetujuan ini, lihatthe section called “Contoh konfigurasi persetujuan per baris”.

Combined per-line and per-level approvals

Tidak direkomendasikan. Di konsol, tab Builder tidak lagi mendukung penambahan persetujuan per baris. Namun, dalam beberapa kasus Anda mungkin berakhir dengan persetujuan per-line dan per-level dalam template perubahan. Hal ini dapat terjadi jika Anda memperbarui template perubahan yang dibuat sebelum 23 Januari 2023, atau jika Anda membuat atau memperbarui template perubahan dengan mengedit konten YAML-nya secara manual,

Untuk tampilan konsol dan sampel JSON dari jenis persetujuan ini, lihatthe section called “Sampel gabungan konfigurasi persetujuan per level dan per-line”.

Important

Meskipun dimungkinkan untuk membuat template perubahan yang menggabungkan persetujuan per baris dan per level, konfigurasi ini tidak disarankan atau diperlukan. Jenis persetujuan apa pun yang memerlukan lebih banyak persetujuan (persetujuan per baris atau per tingkat) diutamakan. Misalnya:

- Jika template perubahan menetapkan tiga persetujuan per level tetapi lima persetujuan per baris, maka diperlukan lima persetujuan.
- Jika template perubahan menetapkan empat persetujuan per tingkat tetapi dua persetujuan per baris, maka empat persetujuan diperlukan.

Anda dapat membuat level yang mencakup persetujuan per baris dan per level dengan mengedit konten YAKL atau JSON secara manual. Kemudian, tab Builder menampilkan kontrol untuk menentukan jumlah persetujuan yang diperlukan untuk tingkat dan untuk baris individual. Namun, level baru yang Anda tambahkan menggunakan konsol masih mendukung hanya konfigurasi persetujuan per level.

Mengubah pemberitahuan permintaan dan penolakan

Notifikasi Amazon SNS

Saat permintaan perubahan dibuat menggunakan templat perubahan, notifikasi akan dikirim kepada pelanggan dari topik Amazon Simple Notification Service (Amazon SNS) yang ditunjuk untuk ditinjau untuk ditinjau pada tingkat tersebut. Anda dapat menentukan topik notifikasi di templat perubahan atau mengizinkan pengguna membuat permintaan perubahan untuk menentukannya.

Setelah jumlah minimum persetujuan yang diperlukan diterima pada satu tingkat, pemberitahuan dikirim ke pemberi persetujuan yang berlangganan topik Amazon SNS untuk tingkat berikutnya, dan seterusnya.

> **Important**
>
> Pastikan peran, grup, dan pengguna IAM yang Anda tetapkan bersama-sama memberikan persetujuan yang cukup untuk memenuhi jumlah persetujuan yang diperlukan yang Anda tentukan. Misalnya, jika Anda menetapkan hanya satu grup IAM sebagai pemberi persetujuan yang berisi tiga pengguna, Anda tidak dapat menentukan bahwa lima persetujuan wajib pada tingkat tersebut, hanya tiga atau kurang.

Ubah penolakan permintaan

Tidak peduli berapa banyak tingkat persetujuan dan pemberi persetujuan yang Anda tentukan, hanya satu penolakan terhadap permintaan perubahan yang diperlukan untuk mencegah alur kerja runbook agar permintaan tersebut tidak terjadi.

Change Managercontoh jenis persetujuan

Contoh berikut menunjukkan tampilan konsol dan konten JSON untuk tiga jenis jenis persetujuanChange Manager.

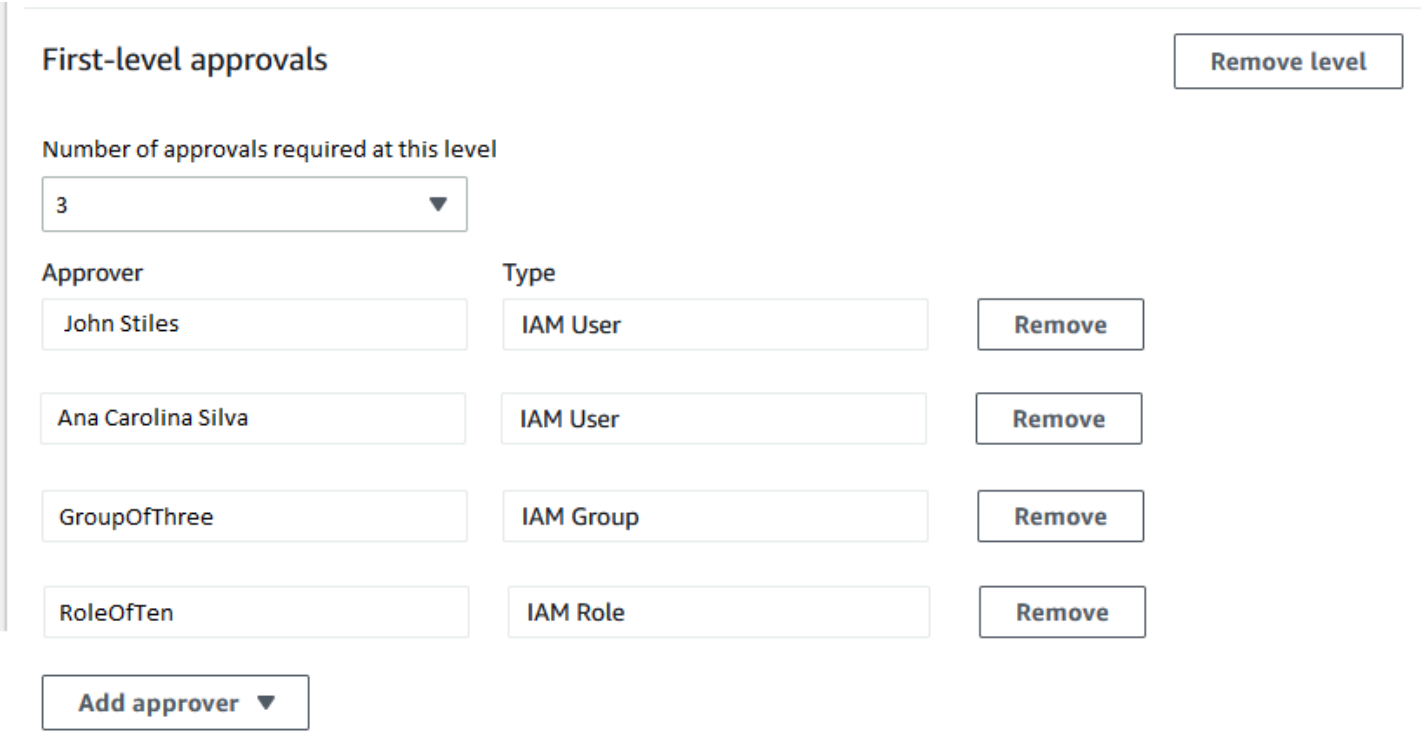
Topik

- Contoh konfigurasi persetujuan per tingkat
- Contoh konfigurasi persetujuan per baris
- Sampel gabungan konfigurasi persetujuan per level dan per-line

Contoh konfigurasi persetujuan per tingkat

Dalam pengaturan tingkat persetujuan per tingkat yang ditunjukkan pada gambar berikut, diperlukan tiga persetujuan. Persetujuan tersebut dapat berasal dari kombinasi pengguna IAM, grup, dan peran yang ditetapkan sebagai pemberi persetujuan. Penyetuju yang ditentukan mencakup dua pengguna IAM (John Stiles dan Ana Carolina Silva), grup pengguna yang berisi tiga anggota (`GroupOfThree`), dan peran pengguna yang mewakili sepuluh pengguna (`RoleOfTen`).

Jika ketiga pengguna dalam`GroupOfThree` grup menyetujui permintaan perubahan, itu disetujui untuk tingkat tersebut. Anda tidak perlu menerima persetujuan dari setiap pada pada pada pada pada pada pada pada pada pada pada pada setiap pada pada pada pada pada pada pada pada pada pada pada pada setiap pada pada pada pada pada Jumlah minimum persetujuan dapat berasal dari kombinasi pemberi persetujuan yang ditentukan. Kami merekomendasikan persetujuan per level untukChange Manager operasi Anda.



Contoh berikut menggambarkan bagian dari kode YAKL untuk konfigurasi ini.

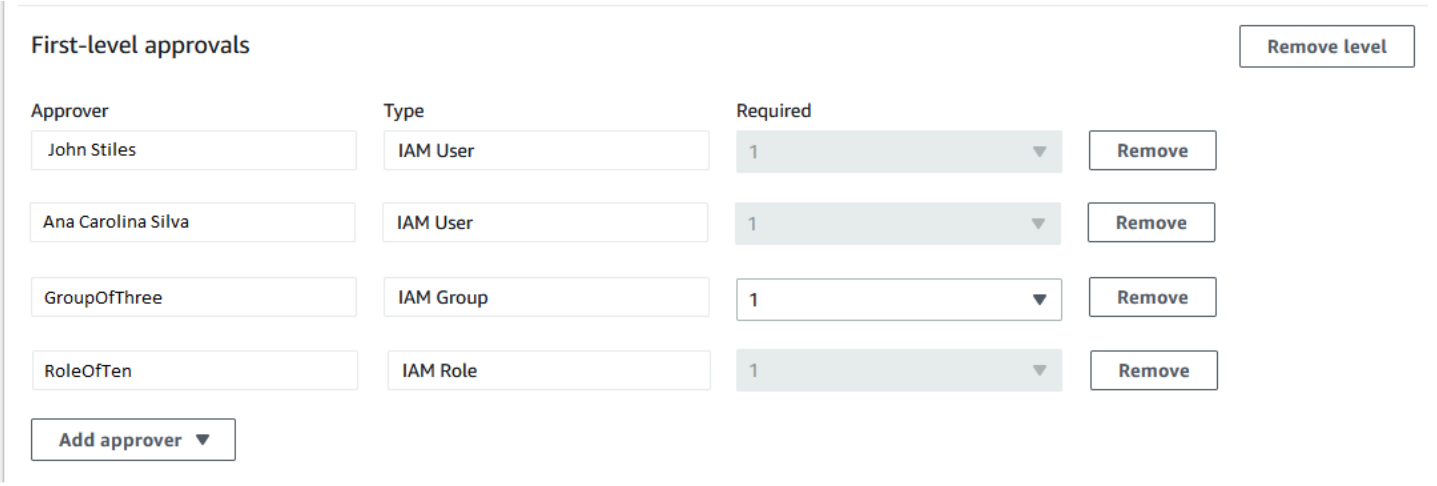
> **Note**
>
> Versi kode YAKL ini mencakup input tambahan,`MinRequiredApprovals` (dengan modal awal`M`). Nilai untuk input ini menunjukkan berapa banyak persetujuan yang diperlukan dari semua pengulas yang tersedia. Perhatikan juga bahwa nilai`minRequiredApprovals` (huruf kecil awal`m`) untuk setiap pemberi persetujuan dalam`Approvers` daftar adalah`0` (nol). Ini menunjukkan bahwa pemberi persetujuan dapat berkontribusi pada persetujuan keseluruhan tetapi tidak diharuskan untuk melakukannya.

```
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
```

Contoh konfigurasi persetujuan per baris

Dalam pengaturan tingkat persetujuan yang ditunjukkan pada gambar berikut, empat pemberi persetujuan ditentukan. Ini termasuk dua pengguna IAM (John Stiles dan Ana Carolina Silva), grup pengguna yang berisi tiga anggota (`GroupOfThree`), dan peran pengguna yang mewakili sepuluh pengguna (`RoleOfTen`). Persetujuan per baris didukung untuk kompatibilitas mundur tetapi tidak disarankan.



Agar permintaan perubahan disetujui dalam konfigurasi persetujuan per baris ini, itu harus disetujui oleh semua jalur penyetuju: John Stiles, Ana Carolina Silva, salah satu anggota`GroupOfThree` grup, dan satu anggota`RoleOfTen` peran.

Contoh berikut menggambarkan bagian dari kode YAKL untuk konfigurasi ini.

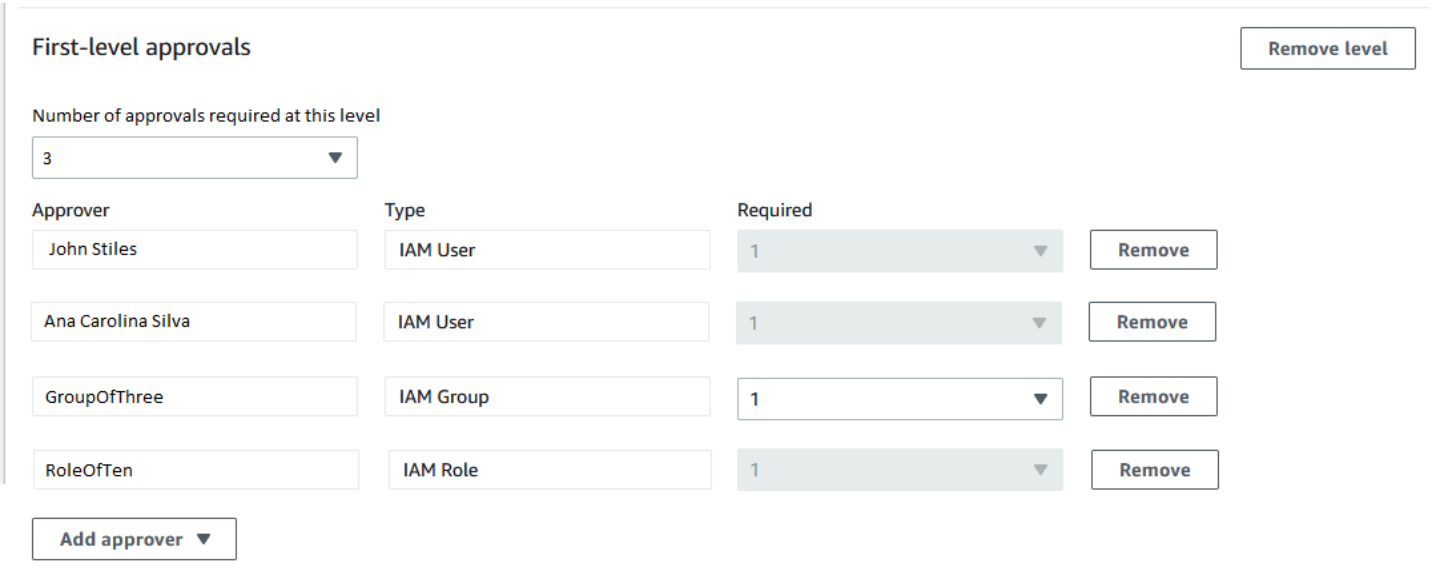
> **Note**
>
> Perhatikan bahwa nilai untuk setiap`minRequiredApprovals` pemberi persetujuan adalah`1`. Ini menunjukkan bahwa satu persetujuan diperlukan dari setiap pemberi persetujuan.

```
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
```

Sampel gabungan konfigurasi persetujuan per level dan per-line

Dalam pengaturan persetujuan per level dan per baris gabungan yang ditunjukkan pada gambar berikut, tiga persetujuan ditentukan untuk level tersebut, tetapi empat persetujuan ditentukan untuk persetujuan baris-item. Jenis persetujuan apa pun yang memerlukan lebih banyak persetujuan lebih diutamakan daripada yang lain, jadi empat persetujuan diperlukan oleh konfigurasi ini. Gabungan persetujuan per-level dan per-line tidak disarankan.




```
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
```

Topik

- Membuat templat perubahan menggunakan Builder
- Membuat templat perubahan menggunakan Editor
- Membuat templat perubahan menggunakan alat baris perintah

Membuat templat perubahan menggunakan Builder

Dengan Builder untuk templat perubahan diChange Manager, kemampuan dariAWS Systems Manager, Anda dapat mengonfigurasi alur kerja runbook yang ditentukan dalam templat perubahan Anda tanpa harus menggunakan sintaks JSON atau YAKL. Setelah Anda menentukan opsi, sistem akan mengonversi input Anda ke dalam format YAML yang dapat digunakan Systems Manager untuk menjalankan alur kerja runbook.

Untuk membuat templat perubahan menggunakan Builder

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Change Manager.

-atau-

JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Change Manager.

3. Pilih Buat templat.
4. Untuk Nama, masukkan nama untuk templat yang membuat tujuannya mudah diidentifikasi, seperti **`UpdateEC2LinuxAMI`**.
5. Di bagian Detail templat perubahan, lakukan hal berikut:

   - Untuk Deskripsi, berikan penjelasan singkat tentang bagaimana dan kapan templat perubahan yang Anda buat akan digunakan.

     Deskripsi ini membantu pengguna yang membuat permintaan perubahan menentukan apakah mereka menggunakan templat perubahan yang benar. Ini membantu mereka yang meninjau permintaan perubahan memahami apakah permintaan tersebut harus disetujui.
   - Untuk Ubah jenis templat, tentukan apakah Anda membuat templat perubahan standar atau templat perubahan darurat.

     Templat perubahan darurat digunakan untuk situasi ketika perubahan harus dilakukan meskipun perubahan diblokir oleh peristiwa di kalender yang digunakan olehAWS Systems ManagerChange Calendar. Permintaan perubahan yang dibuat dari templat perubahan darurat masih harus disetujui oleh pemberi persetujuan yang ditunjuk, tetapi perubahan yang diminta tetap dapat berjalan meskipun kalender diblokir.
   - Untuk Opsi runbook, tentukan runbook yang dapat dipilih pengguna saat membuat permintaan perubahan. Anda dapat menambahkan satu runbook atau beberapa runbook. Atau, Anda dapat mengizinkan peminta untuk menentukan runbook mana yang akan digunakan. Dalam kasus ini, hanya satu runbook yang dapat dimasukkan dalam permintaan perubahan.
   - Untuk Runbook, pilih nama runbook dan versi runbook tersebut yang dapat dipilih pengguna untuk permintaan perubahan mereka. Berapa pun banyaknya runbook yang Anda tambahkan ke templat perubahan, hanya satu yang dapat dipilih per permintaan perubahan.

     Anda tidak menentukan runbook jika memilih Runbook apa pun dapat digunakan sebelumnya.

> **Tip**
>
> Pilih runbook dan versi runbook, kemudian pilih Lihat untuk memeriksa isi runbook di antarmuka Dokumen Systems Manager.

6. Di bagian Informasi templat, gunakan Markdown untuk memasukkan informasi bagi pengguna yang membuat permintaan perubahan dari templat perubahan ini. Kami telah menyediakan serangkaian pertanyaan yang dapat Anda sertakan untuk pengguna yang membuat permintaan perubahan, atau Anda dapat menambahkan informasi dan pertanyaan lain sebagai gantinya.

   > **Note**
   >
   > Markdown adalah bahasa markup yang memungkinkan Anda untuk menambahkan deskripsi gaya wiki ke dokumen dan langkah-langkah individual dalam dokumen. Untuk informasi selengkapnya tentang penggunaan Markdown, lihat Menggunakan Markdown di AWS.

   Sebaiknya berikan pertanyaan kepada pengguna untuk dijawab tentang permintaan perubahan mereka untuk membantu pemberi persetujuan memutuskan apakah akan mengabulkan setiap permintaan perubahan atau tidak, seperti mencantumkan langkah manual apa pun yang diperlukan untuk dijalankan sebagai bagian dari perubahan dan rencana rollback.

   > **Tip**
   >
   > Alihkan antara Sembunyikan pratinjau dan Tampilkan pratinjau untuk melihat seperti apa konten Anda yang Anda tulis.

7. Di bagian Pemberi persetujuan permintaan perubahan, lakukan hal berikut:

   - (Opsional) Jika Anda ingin mengizinkan permintaan perubahan yang dibuat dari template perubahan ini berjalan secara otomatis, tanpa ditinjau oleh pemberi persetujuan apa pun (kecuali peristiwa pembekuan perubahan), pilih Aktifkan persetujuan otomatis.

> **Note**
>
> Mengaktifkan persetujuan otomatis dalam template perubahan memberi pengguna opsi untuk melewati pengulas. Mereka masih dapat memilih untuk menentukan peninjau saat membuat permintaan perubahan. Oleh karena itu, Anda masih harus menentukan opsi peninjau di template perubahan.

> **Important**
>
> Jika Anda mengaktifkan persetujuan otomatis untuk template perubahan, pengguna dapat mengirimkan permintaan perubahan menggunakan template yang tidak memerlukan peninjauan oleh pengulas sebelum dijalankan (dengan pengecualian pemberi persetujuan peristiwa pembekuan perubahan). Jika ingin membatasi peran pengguna, grup, atau IAM tertentu agar tidak mengirimkan permintaan persetujuan otomatis, Anda dapat menggunakan ketentuan dalam kebijakan IAM untuk tujuan ini. Untuk informasi selengkapnya, lihat Mengontrol akses ke alur kerja runbook persetujuan otomatis.

- Untuk Jumlah persetujuan yang diperlukan pada tingkat ini, pilih jumlah persetujuan yang mengubah permintaan yang dibuat dari template perubahan ini harus diterima untuk tingkat ini.
- Untuk menambahkan pemberi persetujuan tingkat pertama wajib, pilih Tambahkan pemberi persetujuan, lalu pilih dari berikut:
  - Pemberi persetujuan yang ditentukan templat — Pilih satu atau beberapa peran pengguna, grup, atauAWS Identity and Access Management (IAM) dari akun Anda untuk menyetujui permintaan perubahan yang dibuat dari templat perubahan ini. Setiap permintaan perubahan yang dibuat menggunakan templat ini harus ditinjau dan disetujui oleh setiap pemberi persetujuan yang Anda tentukan.
  - Minta pemberi persetujuan yang ditentukan - Pengguna yang membuat permintaan perubahan menentukan pengulas pada saat mereka mengajukan permintaan dan dapat memilih dari daftar pengguna di akun Anda.

    Nomor yang Anda masukkan dalam kolom Wajib menentukan berapa banyak peninjau yang harus ditentukan oleh permintaan perubahan yang menggunakan templat perubahan ini.

> **Important**
>
> Sebelum 23 Januari 2023, tab Builder hanya mendukung penentuan persetujuan per baris. Template perubahan baru dan level baru yang Anda tambahkan ke templat perubahan yang ada menggunakan tab Builder hanya mendukung persetujuan per level. Sebaiknya gunakan hanya persetujuan per level dalamChange Manager operasi Anda.
>
> Untuk informasi selengkapnya, lihat Tentang persetujuan dalam template perubahan Anda.

- Untuk Topik SNS untuk memberi tahu pemberi persetujuan, lakukan hal berikut:

  1. Pilih salah satu dari berikut ini untuk menentukan topik Amazon Simple Notification Service (Amazon SNS) di akun Anda yang akan digunakan untuk mengirim notifikasi kepada pemberi persetujuan bahwa permintaan perubahan siap untuk mereka tinjau:

     - Masukkan SNS Amazon Resource Name (ARN) – Untuk ARN Topik, masukkan ARN topik Amazon SNS yang ada. Topik ini dapat berada di salah satu akun organisasi Anda.
     - Pilih topik SNS yang ada – Untuk Topik notifikasi target, pilih ARN topik Amazon SNS yang ada di Akun AWS Anda saat ini. (Opsi ini tidak tersedia jika Anda belum membuat topik Amazon SNS apa pun di Akun AWS dan Wilayah AWS Anda saat ini.)
     - Tentukan topik SNS saat permintaan perubahan dibuat— Pengguna yang membuat permintaan perubahan dapat menentukan topik Amazon SNS yang akan digunakan untuk pemberitahuan.

     > **Note**
     >
     > Topik Amazon SNS yang Anda pilih harus dikonfigurasi untuk menentukan notifikasi yang dikirim dan pelanggan yang menerimanya. Kebijakan aksesnya juga harus memberikan izin kepada Systems Manager sehinggaChange Manager dapat mengirim notifikasi. Untuk informasi, lihat Mengonfigurasi topik Amazon SNS untukChange Managerpemberitahuan.

  2. Pilih Tambahkan notifikasi.

8. (Opsional) Untuk menambahkan tingkat pemberi persetujuan tambahan, pilih Tambahkan tingkat persetujuan dan pilih antara pemberi persetujuan yang ditentukan templat dan pemberi

persetujuan yang ditentukan permintaan untuk tingkat ini. Lalu, pilih topik SNS untuk memberi tahu tingkat pemberi persetujuan ini.

Setelah semua persetujuan diterima oleh pemberi persetujuan tingkat pertama, pemberi persetujuan tingkat kedua diberi tahu, dan seterusnya.

Anda dapat menambahkan maksimum lima tingkat pemberi persetujuan di tiap templat. Anda mungkin, misalnya, memerlukan persetujuan dari pengguna dalam peran teknis untuk tingkat pertama, lalu persetujuan manajerial untuk tingkat kedua.

9. Di bagian Pemantauan, agarCloudWatch alarm dapat dipantau, masukkan nama CloudWatch alarm Amazon di akun saat ini untuk memantau kemajuan alur kerja runbook yang didasarkan pada templat ini.

> **Tip**
>
> Untuk membuat alarm baru, atau untuk meninjau pengaturan alarm yang ingin Anda tentukan, pilih Buka CloudWatch konsol Amazon. Untuk informasi tentang bekerja dengan CloudWatch alarm, lihat Menggunakan CloudWatch Alarm di Panduan CloudWatch Pengguna Amazon.

10. Di bagian Pemberitahuan, lakukan hal berikut:

    1. Pilih salah satu dari berikut ini untuk menentukan topik Amazon SNS di akun Anda yang akan digunakan untuk mengirim notifikasi tentang permintaan perubahan yang dibuat menggunakan templat perubahan ini:

       - Masukkan SNS Amazon Resource Name (ARN) – Untuk ARN Topik, masukkan ARN topik Amazon SNS yang ada. Topik ini dapat berada di salah satu akun organisasi Anda.
       - Pilih topik SNS yang ada – Untuk Topik notifikasi target, pilih ARN topik Amazon SNS yang ada di Akun AWS Anda saat ini. (Opsi ini tidak tersedia jika Anda belum membuat topik Amazon SNS apa pun di Akun AWS dan Wilayah AWS Anda saat ini.)

       > **Note**
       >
       > Topik Amazon SNS yang Anda pilih harus dikonfigurasi untuk menentukan notifikasi yang dikirim dan pelanggan yang menerimanya. Kebijakan aksesnya juga harus memberikan izin kepada Systems Manager sehinggaChange Manager dapat

mengirim notifikasi. Untuk informasi, lihat Mengonfigurasi topik Amazon SNS untukChange Managerpemberitahuan.

2. Pilih Tambahkan notifikasi.

11. (Opsional) Dalam bagian Tag, terapkan satu atau beberapa pasangan nama/nilai kunci tag ke templat perubahan.

    Tag adalah metadata opsional yang Anda tetapkan ke sumber daya. Dengan menggunakan tag, Anda dapat mengategorikan sumber daya dengan cara yang berbeda, seperti berdasarkan tujuan, pemilik, atau lingkungan. Misalnya, Anda mungkin ingin menandai templat perubahan untuk mengidentifikasi jenis perubahan yang dibuat dan lingkungan tempat templat tersebut dijalankan. Dalam hal ini, Anda dapat menentukan pasangan nama/nilai kunci berikut:

    - `Key=TaskType,Value=InstanceRepair`
    - `Key=Environment,Value=Production`

    Untuk informasi selengkapnya tentang penandaan sumber daya Systems Manager, lihat Penandaan sumber daya Systems Manager.

12. Pilih Simpan dan pratinjau.

13. Tinjau detail templat perubahan yang Anda buat.

    Jika Anda ingin mengubah templat perubahan sebelum mengirimkannya untuk ditinjau, pilih Tindakan, Edit.

    Jika Anda puas dengan isi templat perubahan tersebut, pilih Kirim untuk ditinjau. Pengguna di organisasi atau akun Anda yang telah ditetapkan sebagai peninjau templat pada tab Pengaturan diChange Manager diberi tahu bahwa templat perubahan baru sedang menunggu tinjauan mereka.

    Jika topik Amazon SNS telah ditentukan untuk templat perubahan, pemberitahuan akan dikirim saat templat perubahan ditolak atau disetujui. Jika Anda tidak menerima notifikasi terkait templat perubahan ini, Anda dapat kembali keChange Manager nanti untuk memeriksa statusnya.

Membuat templat perubahan menggunakan Editor

Gunakan langkah-langkah dalam topik ini untuk mengonfigurasi templat perubahan diChange Manager, kemampuan dariAWS Systems Manager, dengan memasukkan JSON atau YAKL alih-alih menggunakan kontrol konsol.

Untuk membuat templat perubahan menggunakan Editor

1. Di panel navigasi, pilih Change Manager.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Change Manager.
2. Pilih Buat templat.
3. Untuk Nama, masukkan nama untuk templat yang membuat tujuannya mudah diidentifikasi, seperti **`RestartEC2LinuxInstance`**.
4. Di atas Ubah detail templat, pilih Editor.
5. Di bagian Editor dokumen, pilih Edit, dan kemudian masukkan konten JSON atau YAML untuk templat perubahan Anda.

   Berikut adalah contoh.

   > **Note**
   >
   > Parameter`minRequiredApprovals` ini digunakan untuk menentukan berapa banyak pengulas pada tingkat tertentu harus menyetujui permintaan perubahan yang dibuat menggunakan template ini.
   > Contoh ini menunjukkan dua tingkat persetujuan. Anda dapat menentukan hingga lima tingkat persetujuan, tetapi hanya satu tingkat yang diperlukan.
   > Di tingkat pertama, pengguna tertentu “John-Doe” harus menyetujui setiap permintaan perubahan. Setelah itu, tiga anggota peran IAM`Admin` harus menyetujui permintaan perubahan.
   > Untuk informasi lebih lanjut tentang persetujuan templat perubahan, lihatTentang persetujuan dalam template perubahan Anda.

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. Pilih Simpan dan pratinjau.
7. Tinjau detail templat perubahan yang Anda buat.

   Jika Anda ingin mengubah templat perubahan sebelum mengirimkannya untuk ditinjau, pilih Tindakan, Edit.

   Jika Anda puas dengan isi templat perubahan tersebut, pilih Kirim untuk ditinjau. Pengguna di organisasi atau akun Anda yang telah ditetapkan sebagai peninjau templat pada tab Pengaturan di akanChange Manager diberi tahu bahwa templat perubahan baru sedang menunggu tinjauan mereka.

Jika topik Amazon Simple Notification Service (Amazon SNS) telah ditentukan untuk templat perubahan, notifikasi akan dikirim saat templat perubahan ditolak atau disetujui. Jika Anda tidak menerima notifikasi terkait templat perubahan ini, Anda dapat kembali keChange Manager nanti untuk memeriksa statusnya.

Membuat templat perubahan menggunakan alat baris perintah

Prosedur berikut menjelaskan cara menggunakan AWS Command Line Interface (AWS CLI) (di LinuxmacOS,, atau Windows) atau AWS Tools for Windows PowerShell untuk membuat permintaan perubahanChange Manager, kemampuanAWS Systems Manager.

Untuk membuat templat perubahan

1. Instal dan konfigurasikan AWS CLI atau AWS Tools for PowerShell, jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaru AWS CLI dan Memasang AWS Tools for PowerShell.
2. Buat file JSON di komputer lokal Anda dengan nama seperti `MyChangeTemplate.json`, kemudian tempel konten untuk templat perubahan Anda ke dalamnya.

   > **Note**
   >
   > Templat perubahan menggunakan versi skema 0.3 yang tidak mencakup semua dukungan yang sama seperti untuk runbook Otomatisasi.

   Berikut adalah contohnya.

   > **Note**
   >
   > Parameter `minRequiredApprovals` ini digunakan untuk menentukan berapa banyak pengulas pada tingkat tertentu harus menyetujui permintaan perubahan yang dibuat menggunakan template ini.
   > Contoh ini menunjukkan dua tingkat persetujuan. Anda dapat menentukan hingga lima tingkat persetujuan, tetapi hanya satu tingkat yang diperlukan.

Di tingkat pertama, pengguna tertentu “John-Doe” harus menyetujui setiap permintaan perubahan. Setelah itu, tiga anggota peran IAM `Admin` harus menyetujui permintaan perubahan.

Untuk informasi selengkapnya tentang persetujuan templat perubahan, lihatTentang persetujuan dalam template perubahan Anda.

```
{
   "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. Jalankan perintah berikut untuk membuat templat perubahan.

   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"
   ```

   Untuk informasi tentang opsi lain yang dapat Anda tentukan, lihat create-document.

   Sistem mengembalikan informasi seperti berikut ini.

   ```
   {
      "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":[

      ]
   }
}
```

Pengguna di organisasi atau akun Anda yang telah ditetapkan sebagai peninjau templat pada tab Pengaturan di Change baru Change Manager sedang menunggu tinjauan mereka.

Jika topik Amazon Simple Notification Service (Amazon SNS) telah ditentukan untuk templat perubahan, notifikasi akan dikirim saat templat perubahan ditolak atau disetujui. Jika Anda tidak menerima notifikasi terkait templat perubahan ini, Anda dapat kembali ke Change Manager nanti untuk memeriksa statusnya.

Meninjau dan menyetujui atau menolak templat perubahan

Jika Anda ditetapkan sebagai peninjau untuk templat perubahan diChange Manager, kemampuan dariAWS Systems Manager, Anda akan diberi tahu saat templat perubahan baru, atau versi baru dari templat perubahan, sedang menunggu tinjauan Anda. Topik Amazon Simple Notification Service (Amazon SNS) mengirimkan notifikasi.

> **Note**
>
> Fungsi ini tergantung pada apakah akun Anda telah dikonfigurasi untuk menggunakan topik Amazon SNS untuk mengirim notifikasi tinjauan templat perubahan. Untuk informasi tentang menentukan topik notifikasi peninjau templat, lihat Tugas 1: Mengonfigurasi manajemen identitasChange Manager pengguna dan peninjau templat.

Untuk meninjau templat perubahan, ikuti tautan di notifikasi Anda, masuk ke AWS Management Console, dan ikuti langkah-langkah dalam prosedur ini.

Untuk meninjau, dan menyetujui atau menolak templat perubahan

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Change Manager.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Change Manager.
3. Di bagian Template perubahan di bagian bawah tab Gambaran Umum, pilih nomor di Menunggu tinjauan.

4. Dalam daftar Templat perubahan, temukan dan pilih nama templat perubahan yang akan ditinjau.
5. Di halaman ringkasan, tinjau konten yang diusulkan dari templat perubahan dan lakukan salah satu hal berikut:

   - Untuk menyetujui templat perubahan, yang memungkinkannya untuk digunakan dalam permintaan perubahan, pilih Setujui.
   - Untuk menolak templat perubahan, yang mencegahnya digunakan dalam permintaan perubahan, pilih Tolak.

Menghapus templat perubahan

Topik ini menjelaskan cara menghapus template yang telah Anda buatChange Manager, kemampuan Systems Manager. Jika Anda menggunakanChange Manager untuk organisasi, prosedur ini dilakukan di akun administrator yang didelegasikan.

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Change Manager.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih pilih Change Manager.
3. Pilih tab Template.
4. Pilih template yang akan dihapus.
5. Pilih Tindakan, Hapus template.
6. Di dialog konfirmasi, masukkan kata**`DELETE`**, lalu pilih Menghapus.

Bekerja dengan permintaan perubahan

Permintaan perubahan adalah permintaan diChange Manageruntuk menjalankan runbook Otomatisasi yang memperbarui satu atau beberapa sumber daya diAWSatau on-premise premise. Permintaan perubahan dibuat menggunakan templat perubahan.

Saat Anda membuat permintaan perubahanChange Manager, kemampuanAWS Systems Manager, satu atau beberapa pemberi persetujuan di organisasi atau akun Anda harus meninjau dan

menyetujui permintaan tersebut. Tanpa persetujuan yang diperlukan, alur kerja runbook, yang membuat perubahan yang Anda minta, tidak diizinkan untuk berjalan.

Topik

- Membuat permintaan perubahan
- Meninjau dan menyetujui atau menolak permintaan perubahan

Membuat permintaan perubahan

Saat Anda membuat permintaan perubahanChange Manager, kemampuan dariAWS Systems Manager, templat perubahan yang Anda pilih biasanya melakukan hal berikut:

- Menunjuk pemberi persetujuan untuk permintaan perubahan atau menentukan berapa banyak persetujuan yang diperlukan
- Menentukan topik Amazon Simple Notification Service (Amazon SNS) yang akan digunakan untuk memberi tahu pemberi persetujuan tentang permintaan perubahan Anda
- Menentukan CloudWatch alarm Amazon untuk memantau alur kerja runbook untuk permintaan perubahan
- Mengidentifikasi runbook Otomasi mana yang dapat Anda pilih untuk membuat perubahan yang diminta

Dalam beberapa kasus, templat perubahan mungkin dikonfigurasi sehingga Anda menentukan runbook Otomatisasi Anda sendiri untuk digunakan, dan untuk menentukan siapa yang harus meninjau dan menyetujui permintaan.

> **Important**
>
> Jika Anda menggunakan Change Manager di seluruh organisasi, kami sarankan untuk selalu membuat perubahan dari akun administrator yang didelegasikan. Meskipun Anda dapat membuat perubahan dari akun lain di organisasi, perubahan tersebut tidak akan dilaporkan atau dapat dilihat dari akun administrator yang didelegasikan.

Topik

- Tentang persetujuan permintaan perubahan
- Membuat permintaan perubahan (konsol)

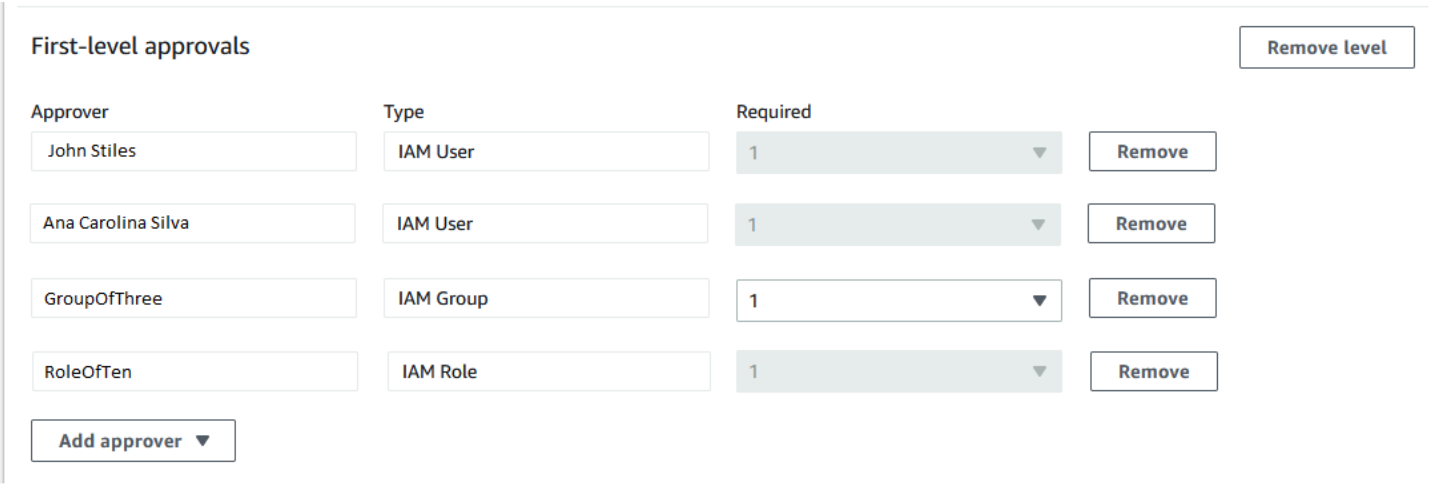
- Membuat permintaan perubahan (AWS CLI)

Tentang persetujuan permintaan perubahan

Bergantung pada persyaratan yang ditentukan dalam template perubahan, mengubah permintaan yang Anda buat darinya dapat memerlukan persetujuan dari hingga lima tingkat sebelum alur kerja runbook untuk permintaan dapat terjadi. Untuk masing-masing level tersebut, pembuat template dapat menentukan hingga lima pemberi persetujuan potensinya. Penyetuju tidak terbatas pada satu pengguna. Penyetuju dalam pengertian ini juga dapat berupa grup IAM atau peran IAM. Untuk grup IAM dan peran IAM, satu atau beberapa pengguna yang termasuk dalam grup atau peran dapat memberikan persetujuan untuk menerima jumlah total persetujuan yang diperlukan untuk permintaan perubahan. Pembuat template juga dapat menentukan lebih banyak pemberi persetujuan daripada yang dibutuhkan template perubahan.

Alur kerja persetujuan asli dan pembaruan dan/atau persetujuan

Menggunakan templat perubahan yang dibuat sebelum 23 Januari 2023, persetujuan harus diterima dari setiap pemberi persetujuan yang ditentukan agar permintaan perubahan disetujui pada tingkat tersebut. Misalnya, dalam penyiapan tingkat persetujuan yang ditunjukkan pada gambar berikut, empat pemberi persetujuan ditentukan. Penyetuju yang ditentukan mencakup dua pengguna (John Stiles dan Ana Carolina Silva), grup pengguna yang berisi tiga anggota (GroupOfThree), dan peran pengguna yang mewakili sepuluh pengguna (). RoleOfTen

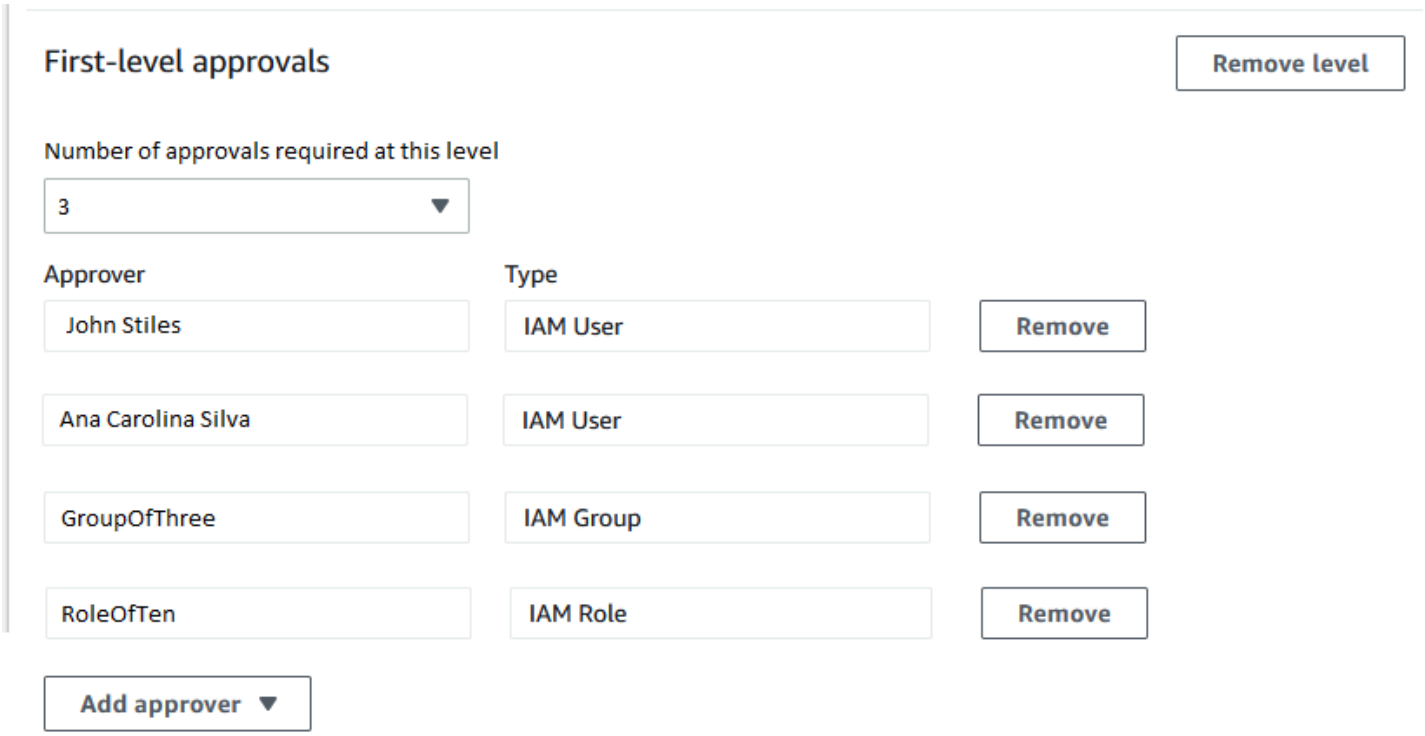


Agar permintaan perubahan disetujui pada tingkat ini, itu harus disetujui oleh John Stiles, Ana Carolina Silva, salah satu anggota `GroupOfThree` grup, dan satu anggota peran. `RoleOfTen`

Menggunakan template perubahan yang dibuat pada atau setelah 23 Januari 2023, untuk setiap tingkat persetujuan, pembuat template dapat menentukan jumlah total persetujuan yang diperlukan

secara keseluruhan. Persetujuan tersebut dapat berasal dari kombinasi pengguna, grup, dan peran apa pun yang telah ditetapkan sebagai pemberi persetujuan. Template perubahan hanya memerlukan satu persetujuan untuk satu level tetapi menentukan, misalnya, dua pengguna individual, dua grup, dan satu peran sebagai pemberi persetujuan potensional.

Misalnya, di area tingkat persetujuan yang ditunjukkan pada gambar berikut, diperlukan tiga persetujuan. Penyetuju yang ditentukan template mencakup dua pengguna (John Stiles dan Ana Carolina Silva), grup pengguna yang berisi tiga anggota (`GroupOfThree`), dan peran pengguna yang mewakili sepuluh pengguna (). `RoleOfTen`



Jika ketiga pengguna dalam `GroupOfThree` grup menyetujui permintaan perubahan Anda, itu disetujui untuk level tersebut. Tidak perlu menerima persetujuan dari setiap pengguna, grup, atau peran. Jumlah minimum persetujuan dapat berasal dari kombinasi pemberi persetujuan potensinya.

Saat permintaan perubahan Anda dibuat, notifikasi dikirim ke pelanggan topik Amazon SNS yang telah ditentukan untuk pemberitahuan persetujuan pada tingkat tersebut. Pembuat cetakan perubahan mungkin telah menentukan topik notifikasi yang harus digunakan atau memungkinkan Anda untuk menentukannya.

Setelah jumlah minimum persetujuan yang diperlukan diterima pada satu tingkat, pemberitahuan dikirim ke pemberi persetujuan yang berlangganan topik Amazon SNS untuk tingkat berikutnya, dan seterusnya.

Tidak peduli berapa banyak tingkat persetujuan dan pemberi persetujuan yang ditentukan, hanya satu penolakan terhadap permintaan perubahan yang diperlukan untuk mencegah alur kerja runbook agar permintaan tersebut tidak terjadi.

Membuat permintaan perubahan (konsol)

Prosedur berikut menjelaskan cara membuat permintaan perubahan dengan menggunakan konsol Systems Manager.

Untuk membuat permintaan perubahan (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Change Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Change Manager.
3. Pilih Buat permintaan.
4. Cari dan pilih templat perubahan yang ingin Anda gunakan untuk permintaan perubahan ini.
5. Pilih Selanjutnya.
6. Untuk Nama, masukkan nama untuk permintaan perubahan yang membuat tujuannya mudah diidentifikasi, seperti **`UpdateEC2LinuxAMI-us-east-2`**.
7. Untuk Runbook, pilih runbook yang ingin Anda gunakan untuk membuat perubahan yang diminta.

   > **Note**
   >
   > Jika opsi untuk memilih buku runbook tidak tersedia, penulis templat perubahan telah menentukan runbook mana yang harus digunakan.
8. Untuk Ubah informasi permintaan, gunakan Markdown untuk memberikan informasi tambahan tentang permintaan perubahan untuk membantu peninjau memutuskan apakah akan menyetujui atau menolak permintaan perubahan. Penulis templat yang Anda gunakan mungkin telah memberikan instruksi atau pertanyaan untuk Anda jawab.

> **Note**
>
> Markdown adalah bahasa markup yang memungkinkan Anda untuk menambahkan deskripsi gaya wiki ke dokumen dan langkah-langkah individual dalam dokumen. Untuk informasi selengkapnya tentang penggunaan Markdown, lihat Menggunakan Markdown di AWS.

9. Di bagian Waktu mulai alur kerja, pilih salah satu dari berikut ini:

   - Jalankan operasi pada waktu yang dijadwalkan – Untuk Waktu mulai yang diminta, masukkan tanggal dan waktu yang Anda usulkan untuk menjalankan alur kerja runbook untuk permintaan ini. Untuk Estimasi waktu selesai, masukkan tanggal dan waktu yang Anda harapkan alur kerja runbook selesai. (Kali ini hanya perkiraan yang Anda berikan untuk peninjau.)

     > **Tip**
     >
     > Pilih Lihat Change Calendar untuk memeriksa setiap peristiwa pemblokiran untuk waktu yang Anda tentukan.

   - Jalankan operasi sesegera mungkin setelah persetujuan – Jika permintaan perubahan disetujui, alur kerja runbook berjalan segera setelah ada periode yang tidak dibatasi saat perubahan dapat dilakukan.

10. Di bagian Persetujuan perubahan, lakukan hal berikut:

    1. Jika Opsi Jenis persetujuan disajikan, pilih salah satu dari berikut ini:

       - Persetujuan otomatis - Template perubahan yang Anda pilih dikonfigurasi untuk memungkinkan permintaan perubahan berjalan secara otomatis tanpa peninjauan oleh pemberi persetujuan apa pun. Lanjutkan ke Langkah 11.

         > **Note**
         >
         > Izin yang ditentukan dalam kebijakan IAM yang mengatur penggunaan Anda atas Systems Manager tidak boleh membatasi Anda untuk mengirimkan permintaan perubahan persetujuan otomatis agar dapat berjalan secara otomatis.

       - Tentukan pemberi persetujuan — Anda harus menambahkan satu atau beberapa pengguna, grup, atau peran IAM untuk meninjau dan menyetujui permintaan perubahan ini.

> **Note**
>
> Anda dapat memilih untuk menentukan pengulas meskipun izin yang ditentukan dalam kebijakan IAM yang mengatur penggunaan Systems Manager memungkinkan Anda menjalankan permintaan perubahan persetujuan otomatis.

2. Pilih Tambahkan pemberi persetujuan, lalu pilih satu atau lebih pengguna, grup, atau AWS Identity and Access Management (IAM role) role dari daftar peninjau yang tersedia.

> **Note**
>
> Satu atau beberapa pemberi persetujuan mungkin sudah ditentukan. Ini berarti bahwa pemberi persetujuan wajib sudah ditentukan dalam templat perubahan yang telah Anda pilih. Pemberi persetujuan ini tidak dapat dihapus dari permintaan. Jika Tambahkan pemberi persetujuan tidak tersedia, templat yang telah Anda pilih tidak mengizinkan peninjau tambahan ditambahkan ke permintaan.

Untuk informasi selengkapnya tentang persetujuan permintaan perubahan, lihatTentang persetujuan permintaan perubahan.

3. Di bawah Topik SNS untuk memberi tahu pemberi persetujuan, pilih salah satu dari berikut ini untuk menentukan topik Amazon SNS di akun Anda untuk digunakan untuk mengirim notifikasi kepada pemberi persetujuan yang Anda tambahkan ke permintaan perubahan ini.

> **Note**
>
> Jika opsi untuk menentukan topik Amazon SNS tidak tersedia, templat perubahan yang Anda pilih telah menentukan topik SNS Amazon yang akan digunakan.

- Masukkan SNS Amazon Resource Name (ARN) – Untuk ARN Topik, masukkan ARN topik Amazon SNS yang ada. Topik ini dapat berada di salah satu akun organisasi Anda.
- Pilih topik SNS yang ada – Untuk Topik notifikasi target, pilih ARN dari topik Amazon SNS yang ada di akun Anda saat ini. (Opsi ini tidak tersedia jika Anda belum membuat topik Amazon SNS apa pun di Akun AWS dan Wilayah AWS Anda saat ini.)

> **Note**
>
> Topik Amazon SNS yang Anda pilih harus dikonfigurasi untuk menentukan notifikasi yang dikirim dan pelanggan yang menerimanya. Kebijakan aksesnya juga harus memberikan izin kepada Systems Manager sehingga Change Manager dapat mengirim notifikasi. Untuk informasi, lihat Mengonfigurasi topik Amazon SNS untukChange Managerpemberitahuan.

4. Pilih Tambahkan notifikasi.

11. Pilih Selanjutnya.
12. Untuk IAM role, pilih role di akun Anda saat ini yang memiliki izin yang diperlukan untuk menjalankan runbook yang ditentukan untuk permintaan perubahan ini.

    Peran ini juga disebut sebagai peran layanan, atau peran asumsi, untuk Otomatisasi. Untuk informasi selengkapnya tentang peran ini, lihat Menyiapkan Otomatisasi.
13. Di bagian Lokasi deployment, pilih salah satu dari berikut ini:

> **Note**
>
> Jika Anda menggunakan Change Manager dengan satu Akun AWS saja dan tidak dengan organisasi yang disiapkan diAWS Organizations, Anda tidak perlu menentukan lokasi deployment.

- Terapkan perubahan ke akun ini – Alur kerja runbook hanya berjalan di akun saat ini. Untuk organisasi, ini berarti akun administrator yang didelegasikan.
- Terapkan perubahan ke beberapa unit organisasi (OU) – Lakukan hal berikut:
  1. Untuk Akun dan unit organisasi (OU), masukkan ID akun anggota di organisasi Anda, dalam format **`123456789012`**, atau ID unit organisasi, dalam format **`o-o96EXAMPLE`**.
  2. (Opsional) Untuk Nama peran eksekusi, masukkan nama IAM role di akun target atau OU yang memiliki izin yang diperlukan untuk menjalankan runbook yang ditentukan untuk permintaan perubahan ini. Semua account di OU yang Anda tentukan harus menggunakan nama yang sama untuk peran ini.
  3. (Opsional) Pilih Tambah lokasi target lain untuk setiap akun atau OU tambahan yang ingin Anda tentukan dan ulangi langkah a dan b.

4. UntukTargetWilayah AWS, pilih Wilayah untuk melakukan perubahan, seperti `Ohio (us-east-2)` untuk Wilayah US East (Ohio).
5. Perluas Pengendalian rate.

   Untuk Konkurensi, masukkan angka, lalu dari daftar pilih apakah ini mewakili jumlah atau persentase akun yang dapat dijalankan oleh alur kerja runbook secara bersamaan.

   Untuk Batas kesalahan, masukkan angka, lalu dari daftar pilih apakah ini mewakili jumlah atau persentase akun di mana alur kerja runbook dapat gagal sebelum operasi dihentikan.

14. Di bagian Target deployment, lakukan hal berikut:

    1. Pilih salah satu dari berikut:
       - Sumber daya tunggal – Perubahan harus dibuat hanya untuk satu sumber daya. Misalnya, satu node atau satu Amazon Machine Image (AMI), tergantung pada operasi yang ditentukan dalam runbook untuk permintaan perubahan ini.
       - Beberapa sumber daya – Untuk Parameter, pilih dari parameter yang tersedia dari runbook untuk permintaan perubahan ini. Pilihan ini mencerminkan jenis sumber daya yang diperbarui.

         Sebagai contoh, jika runbook untuk permintaan perubahan ini adalah `AWS-RetartEC2Instance`, Anda dapat memilih `InstanceId`, dan kemudian menentukan instans mana yang diperbarui dengan memilih dari berikut ini:
         - Tentukan tag – Masukkan pasangan nilai kunci yang ditandai dengan semua sumber daya yang akan diperbarui.
         - Pilih grup sumber daya – Pilih nama grup sumber daya tempat semua sumber daya yang akan diperbarui.
         - Tentukan nilai parameter – Identifikasi sumber daya yang akan diperbarui di bagian Parameter runbook.
         - Target semua instans — Buat perubahan pada semua node terkelola di lokasi target.
    2. Jika Anda memilih Beberapa sumber daya, perluas Pengendalian rate.

       Untuk Konkurensi, masukkan angka, lalu dari daftar pilih apakah ini mewakili jumlah atau persentase target yang dapat diperbarui alur kerja runbook pada saat yang sama.

       Untuk Batas kesalahan, masukkan angka, lalu dari daftar pilih apakah ini mewakili jumlah atau persentase target di mana pembaruan dapat gagal sebelum operasi dihentikan.

15. Jika Anda memilih Tentukan nilai parameter untuk memperbarui beberapa sumber daya di langkah sebelumnya: Di bagian Parameter runbook, tentukan nilai untuk parameter input yang diperlukan. Nilai parameter yang harus Anda berikan didasarkan pada konten runbook Otomatisasi yang terkait dengan templat perubahan yang Anda pilih.

    Misalnya, jika templat perubahan menggunakan `AWS-RetartEC2Instance` runbook, Anda harus memasukkan satu atau lebih ID instans untuk InstanceIdparameter. Atau, pilih Tampilkan pemilih instans interaktif dan pilih instans yang tersedia satu per satu.

16. Pilih Selanjutnya.
17. Di halaman Tinjau dan kirim, periksa kembali sumber daya dan opsi yang telah Anda tentukan untuk permintaan perubahan ini.

    Pilih tombol Edit untuk bagian mana pun yang ingin Anda ubah.

    Jika Anda puas dengan detail permintaan perubahan, pilih Kirim untuk persetujuan.

Jika topik Amazon SNS telah ditentukan dalam templat perubahan yang Anda pilih untuk permintaan, notifikasi akan dikirim ketika permintaan tersebut ditolak atau disetujui. Jika Anda tidak menerima notifikasi untuk permintaan, Anda dapat kembali Change Manager ke untuk memeriksa status permintaan Anda.

Membuat permintaan perubahan (AWS CLI)

Anda dapat membuat permintaan perubahan menggunakan AWS Command Line Interface (AWS CLI) dengan menentukan opsi dan parameter untuk permintaan perubahan dalam file JSON dan menggunakan `--cli-input-json` opsi untuk memasukkannya ke dalam perintah Anda.

Untuk membuat permintaan perubahan (AWS CLI)

1. Instal dan konfigurasikan AWS CLI atau AWS Tools for PowerShell, jika Anda belum melakukannya.

    Untuk informasi, lihat Menginstal atau memperbarui versi terbaru AWS CLI dan Memasang AWS Tools for PowerShell.

2. Buat file JSON di komputer lokal Anda dengan nama seperti `MyChangeRequest.json` dan tempel konten berikut ke dalamnya.

    Ganti *`placeholder`* dengan nilai untuk permintaan perubahan Anda.

> **Note**
>
> Contoh JSON ini membuat permintaan perubahan menggunakan template `AWS-HelloWorldChangeTemplate` perubahan dan `AWS-HelloWorld` runbook. Untuk membantu Anda menyesuaikan sampel ini untuk permintaan perubahan Anda sendiri, lihat StartChangeRequestExecutiondi Referensi AWS Systems Manager API untuk informasi tentang semua parameter yang tersedia
> Untuk informasi selengkapnya tentang persetujuan permintaan perubahan, lihatTentang persetujuan permintaan perubahan.

```
{
    "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. Dalam direktori tempat Anda membuat file JSON, jalankan perintah berikut.

```
aws ssm start-change-request-execution --cli-input-json file://MyChangeRequest.json
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "AutomationExecutionId": "b3c1357a-5756-4839-8617-2d2a4EXAMPLE"
}
```

Meninjau dan menyetujui atau menolak permintaan perubahan

Jika Anda ditetapkan sebagai peninjau untuk permintaan perubahan diChange Manager, kemampuan dariAWS Systems Manager, Anda akan diberi tahu melalui topik Amazon Simple Notification Service (Amazon SNS) saat permintaan perubahan baru menunggu tinjauan Anda.

> **Note**
>
> Fungsionalitas ini bergantung pada apakah Amazon SNS telah ditentukan dalam templat perubahan untuk mengirim notifikasi tinjauan. Untuk informasi, lihat Mengonfigurasi topik Amazon SNS untukChange Managerpemberitahuan.

Untuk meninjau permintaan perubahan, Anda dapat mengikuti tautan di notifikasi, atau masuk ke AWS Management Console secara langsung dan ikuti langkah-langkah dalam prosedur ini.

> **Note**
>
> Jika topik Amazon SNS ditetapkan untuk peninjau dalam templat perubahan, notifikasi akan dikirim ke pelanggan topik ketika permintaan perubahan mengubah status.
> Untuk informasi selengkapnya tentang persetujuan untuk permintaan perubahan, lihatTentang persetujuan permintaan perubahan.

Meninjau dan menyetujui atau menolak permintaan perubahan (konsol)

Prosedur berikut menjelaskan cara menggunakan konsol Systems Manager untuk meninjau dan menyetujui atau menolak permintaan perubahan.

Untuk meninjau dan menyetujui atau menolak permintaan perubahan tunggal

1. Buka tautan di notifikasi email yang Anda terima dan masuk ke AWS Management Console, yang mengarahkan Anda ke permintaan perubahan untuk tinjauan Anda.
2. Di halaman ringkasan, tinjau konten yang diusulkan dari permintaan perubahan.

   Untuk menyetujui permintaan perubahan, pilih Setujui. Di kotak dialog, berikan komentar yang ingin Anda tambahkan untuk persetujuan ini, lalu pilih Setujui. Alur kerja runbook yang diwakili oleh permintaan ini mulai berjalan baik ketika dijadwalkan, atau segera setelah perubahan tidak diblokir oleh pembatasan apa pun.

   -atau-

   Untuk menolak permintaan perubahan, pilih Tolak. Di kotak dialog, berikan komentar yang ingin Anda tambahkan untuk penolakan ini, lalu pilih Tolak.

Untuk meninjau dan menyetujui atau menolak permintaan perubahan secara massal

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Change Manager.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Change Manager.

3. Pilih tab Persetujuan.
4. (Opsional) Tinjau detail permintaan yang menunggu persetujuan Anda dengan memilih nama setiap permintaan, lalu kembali ke tab Persetujuan.
5. Beri tanda centang pada kotak centang pada tiap permintaan perubahan yang ingin Anda setujui.

   -atau-

   Beri tanda centang pada kotak centang pada tiap permintaan perubahan yang ingin Anda tolak.
6. Di kotak dialog, berikan komentar yang ingin Anda tambahkan untuk persetujuan atau penolakan ini.
7. Bergantung pada apakah Anda menyetujui atau menolak permintaan perubahan yang dipilih, pilih Setujui atau Tolak.

Meninjau dan menyetujui atau menolak permintaan perubahan (baris perintah)

Prosedur berikut menjelaskan cara menggunakan AWS Command Line Interface (AWS CLI) (di Linux, macOS, atau Windows) untuk meninjau dan menyetujui atau menolak permintaan perubahan.

Untuk meninjau dan menyetujui atau menolak permintaan perubahan

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaru dari AWS CLI.
2. Buat file JSON di komputer lokal Anda yang menentukan parameter untuk panggilan AWS CLI Anda.

```
{
  "OpsItemFilters":
  [
    {
      "Key": "OpsItemType",
      "Values": ["/aws/changerequest"],
```

```
      "Operator": "Equal"
    }
  ],
  "MaxResults": number
}
```

Anda dapat memfilter hasil untuk pemberi persetujuan tertentu dengan menentukan Amazon Resource Name (ARN) pemberi persetujuan dalam file JSON. Inilah contohnya.

```
{
  "OpsItemFilters":
  [
    {
      "Key": "OpsItemType",
      "Values": ["/aws/changerequest"],
      "Operator": "Equal"
    },
    {
      "Key": "ChangeRequestByApproverArn",
      "Values": ["arn:aws:iam::account-id:user/user-name"],
      "Operator": "Equal"
    }
  ],
  "MaxResults": number
}
```

3. Jalankan perintah berikut untuk melihat jumlah maksimum permintaan perubahan yang Anda tentukan di file 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. Jalankan perintah berikut untuk menyetujui atau menolak permintaan perubahan.

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"
```

Jika topik Amazon SNS telah ditentukan dalam templat perubahan yang Anda pilih untuk permintaan, notifikasi akan dikirim ketika permintaan tersebut ditolak atau disetujui. Jika Anda tidak menerima notifikasi untuk permintaan, Anda dapat kembali Change Manager ke untuk memeriksa status permintaan Anda. Untuk informasi tentang opsi lain saat menggunakan perintah ini, lihat send-automation-signaldi AWS Systems Manager bagian Referensi AWS CLI Perintah.

Meninjau detail permintaan, tugas, dan timeline perubahan (konsol)

Anda dapat melihat informasi tentang permintaan perubahan, termasuk permintaan yang perubahannya telah diproses, di dasborChange Manager, kemampuan dariAWS Systems Manager. Detail ini mencakup tautan ke operasi Otomatisasi yang menjalankan runbook yang membuat perubahan. ID eksekusi otomatisasi dibuat saat permintaan dibuat, tetapi proses tidak berjalan sampai semua persetujuan diberikan dan tidak ada batasan untuk memblokir perubahan.

Untuk meninjau detail permintaan, tugas, dan timeline perubahan

1. Di panel navigasi, pilih Change Manager.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Change Manager.

2. Pilih tab Permintaan.
3. Di bagian Permintaan perubahan, cari permintaan perubahan yang ingin Anda tinjau.

   Anda dapat menggunakan opsi Buat rentang tanggal untuk membatasi hasil hingga jangka waktu tertentu.

   Anda dapat memfilter permintaan berdasarkan properti berikut:

   - `Status`
   - `Request ID`
   - `Approver`
   - `Requester`

   Misalnya, untuk melihat detail tentang semua permintaan perubahan yang berhasil diselesaikan dalam 24 jam terakhir, lakukan hal berikut:

   1. Untuk Buat rentang tanggal, pilih 1d.
   2. Di kotak pencarian, pilih Status, pilih Status, pilih Status, pilih Status, pilih Status, CompletedWithSuccess pilih Status
   3. Di hasil, pilih nama permintaan perubahan yang berhasil diselesaikan untuk meninjau hasil.
4. Lihat informasi tentang permintaan perubahan pada tab berikut:

   - Detail permintaan – Lihat detail dasar tentang permintaan perubahan, termasuk peminta, templat perubahan, dan runbook otomatisasi yang dipilih untuk perubahan. Anda juga dapat mengikuti tautan ke detail operasi Otomatisasi dan melihat informasi tentang parameter runbook apa pun yang ditentukan dalam permintaan, CloudWatch alarm Amazon yang ditetapkan untuk permintaan perubahan, dan persetujuan serta komentar yang diberikan untuk permintaan tersebut.
   - Tugas – Lihat informasi tentang tugas dalam perubahan, termasuk status tugas untuk permintaan perubahan yang diselesaikan, sumber daya yang ditargetkan, langkah-langkah dalam runbook otomatisasi terkait, dan detail ambang batas kesalahan dan konkurensi.
   - Timeline – Lihat ringkasan semua peristiwa yang terkait dengan permintaan perubahan, yang dicantumkan menurut tanggal dan waktu. Ringkasan menunjukkan kapan permintaan perubahan dibuat, tindakan oleh pemberi persetujuan yang ditetapkan, catatan kapan permintaan perubahan yang disetujui dijadwalkan untuk dijalankan, detail alur kerja runbook,

dan perubahan status untuk keseluruhan proses perubahan dan setiap langkah dalam runbook.

- Peristiwa terkait - Lihat detail yang dapat diaudit tentang permintaan perubahan yang dicatat di AWS CloudTrailDanau. Detail mencakup tindakan API mana yang dijalankan, parameter permintaan yang disertakan untuk tindakan tersebut, akun pengguna yang menjalankan tindakan, sumber daya yang diperbarui selama proses, dan banyak lagi.

  Saat Anda mengaktifkan pelacakan peristiwa CloudTrail Lake, CloudTrail Lake membuat penyimpanan data peristiwa untuk acara yang terkait dengan permintaan perubahan Anda. Rincian acara tersedia untuk akun atau organisasi tempat permintaan perubahan dijalankan. Anda dapat mengaktifkan pelacakan acara CloudTrail Lake dari permintaan perubahan apa pun di akun atau organisasi Anda. Untuk informasi tentang mengaktifkan integrasi CloudTrail Danau dan membuat penyimpanan data acara, lihatMemantau peristiwa permintaan perubahan.

  > **Note**
  >
  > Ada biaya untuk menggunakan CloudTrail Danau. Untuk detailnya, lihat AWS CloudTrail harga.

Melihat jumlah agregat permintaan perubahan (baris perintah)

Anda dapat melihat jumlah agregat permintaan perubahan diChange Manager, kemampuan dariAWS Systems Manager, dengan menggunakan operasi GetOpsSummaryAPI. Operasi API ini dapat mengembalikan jumlah untuk satu Akun AWS dalam satu Wilayah AWS atau untuk beberapa akun dan beberapa Wilayah.

> **Note**
>
> Jika Anda ingin melihat jumlah agregat permintaan perubahan untuk beberapa Akun AWS dan beberapa Wilayah AWS, Anda harus menyiapkan dan mengonfigurasi sinkronisasi data sumber daya. Untuk informasi selengkapnya, lihat Pengonfigurasian sinkronisasi data sumber daya untuk Inventaris.

Prosedur berikut menjelaskan cara menggunakan AWS Command Line Interface (AWS CLI) (di Linux, macOS, atau Windows) untuk melihat jumlah agregat permintaan perubahan.

Untuk melihat jumlah agregat permintaan perubahan

1. Instal dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaruAWS CLI.

2. Masukkan salah satu perintah berikut.

   Akun dan Wilayah tunggal

   Perintah ini mengembalikan jumlah semua permintaan perubahan untuk Akun AWS dan Wilayah AWS yang sesi AWS CLI Anda dikonfigurasi.

   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
   ```

   Panggilan mengembalikan informasi seperti berikut.

   ```
   {
       "Entities": [
           {
               "Data": {
                   "AWS:OpsItem": {
                       "Content": [
                           {
                               "Count": "38",
                               "Status": "Open"
                           }
                       ]
                   }
               }
   ```

```
        }
    ]
}
```

Beberapa akun dan/atau Wilayah

Perintah ini mengembalikan jumlah semua permintaan perubahan untuk Akun AWS dan Wilayah AWS yang ditentukan dalam sinkronisasi data sumber daya.

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
```

Panggilan mengembalikan informasi seperti berikut.

```
{
    "Entities": [
        {
            "Data": {
                "AWS:OpsItem": {
                    "Content": [
                        {
                            "Count": "43",
                            "Status": "Open"
                        },
                        {
                            "Count": "2",
                            "Status": "Resolved"
                        }
```

```
                    ]
                }
            }
        }
    ]
}
```

Beberapa akun dan Wilayah tertentu

Perintah ini mengembalikan jumlah semua permintaan perubahan untuk Akun AWS yang ditentukan dalam sinkronisasi data sumber daya. Namun, ini hanya mengembalikan data dari Wilayah yang ditentukan dalam perintah.

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
```

Beberapa akun dan Wilayah

Perintah ini mengembalikan jumlah semua permintaan perubahan untuk Akun AWS dan Wilayah AWS yang ditentukan dalam sinkronisasi data sumber daya. Output menampilkan informasi jumlah per Wilayah.

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"}]}]'
```

Panggilan mengembalikan informasi seperti berikut.

```
{
        "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"
                        }
                    ]
                }
            }
        ]
    }
```

Beberapa akun dan Wilayah

Perintah ini mengembalikan jumlah semua permintaan perubahan untuk Akun AWS dan Wilayah AWS yang ditentukan dalam sinkronisasi data sumber daya. Output mengelompokkan informasi jumlah menurut akun dan Wilayah.

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"}]}]}]
```

Panggilan mengembalikan informasi seperti berikut.

```
{
```

```
    "Entities": [
        {
            "Data": {
                "AWS:OpsItem": {
                    "Content": [
                        {
                            "Count": "38",
                            "SourceAccountId": "123456789012",
                            "SourceRegion": "us-east-1",
                            "Status": "Open"
                        },
                        {
                            "Count": "4",
                            "SourceAccountId": "111122223333",
                            "SourceRegion": "us-east-2",
                            "Status": "Open"
                        },
                        {
                            "Count": "1",
                            "SourceAccountId": "111122223333",
                            "SourceRegion": "us-west-1",
                            "Status": "Open"
                        },
                        {
                            "Count": "2",
                            "SourceAccountId": "444455556666",
                            "SourceRegion": "us-east-2",
                            "Status": "Resolved"
                        },
                        {
                            "Count": "1",
                            "SourceAccountId": "222222222222",
                            "SourceRegion": "us-east-1",
                            "Status": "Open"
                        }
                    ]
                }
            }
        }
    ]
}
```

Audit dan loggingChange Manageraktivitas

Anda dapat mengaudit aktivitas diChange Manager, suatu kemampuanAWS Systems Manager, dengan menggunakan Amazon CloudWatch danAWS CloudTrailalarm.

Untuk informasi selengkapnya tentang opsi pengauditan dan pencatatan untuk Systems Manager, lihat Pemantauan AWS Systems Manager.

AuditChange Manageraktivitas menggunakan CloudWatch alarm

Anda dapat mengonfigurasi dan menetapkan CloudWatch alarm ke templat perubahan. Jika kondisi yang ditentukan dalam alarm terpenuhi, tindakan yang ditentukan untuk alarm akan diambil. Dalam konfigurasi alarm, Anda dapat menentukan topik Amazon Simple Notification Service (Amazon SNS) untuk memberi tahu saat kondisi alarm terpenuhi.

Untuk informasi tentang membuatChange ManagerTemplate, lihatBekerja dengan templat perubahan.

Untuk informasi tentang membuat CloudWatch alarm, lihatMenggunakan CloudWatchAlarmdi dalamAmazon CloudWatch Panduan Pengguna.

AuditChange Manageraktivitas menggunakan CloudTrail

CloudTrail menangkap panggilan API yang dibuat dalam konsol Systems Manager,AWS Command Line Interface(AWS CLI), dan SDK Systems Manager. Anda dapat melihat informasi tersebut di CloudTrail konsol atau di Simple Storage Service (Amazon S3), tempat informasi disimpan. Satu bucket digunakan untuk semua CloudTrail log untuk akun Anda.

Log dariChange Managertindakan menunjukkan pembuatan dokumen templat perubahan, perubahan template, dan persetujuan dan penolakan permintaan perubahan, aktivitas yang dibuat oleh runbook Otomatisasi, dan banyak lagi. Untuk informasi selengkapnya tentang melihat dan menggunakan CloudTrail log aktivitas Systems Manager, lihatPencatatan panggilan AWS Systems Manager API dengan AWS CloudTrail.

Pemecahan Masalah Change Manager

Gunakan informasi berikut untuk membantu Anda memecahkan masalah denganChange Manager, kemampuanAWS Systems Manager.

Topik

- Kesalahan “Grup {GUID} tidak ditemukan” selama perubahan permintaan persetujuan saat menggunakan grup Direktori Aktif

Kesalahan “Grup *{GUID}* tidak ditemukan” selama perubahan permintaan persetujuan saat menggunakan grup Direktori Aktif

Masalah: SaatAWS IAM Identity Center(IAM Identity Center) digunakan untuk manajemen identitas pengguna, anggota grup Direktori Aktif yang diberikan izin persetujuan diChange Managermenerima kesalahan “tidak berwenang” atau “grup tidak ditemukan”.

- Solusi: Bila Anda memilih grup Active Directory di IAM Identity Center untuk akses keAWS Management Console, sistem menjadwalkan sinkronisasi berkala yang menyalin informasi dari grup Direktori Aktif tersebut ke Pusat Identitas IAM. Proses ini harus selesai sebelum pengguna yang diotorisasi melalui keanggotaan grup Direktori Aktif berhasil menyetujui permintaan. Untuk informasi selengkapnya, lihatConnect ke direktori AD Microsoft Andadi dalamAWS IAM Identity CenterPanduan Pengguna.

AWS Systems ManagerOtomasi

Otomatisasi, kemampuanAWS Systems Manager, menyederhanakan tugas pemeliharaan, penyebaran, dan remediasi umum seperti Layanan AWS Amazon Elastic Compute Cloud (Amazon EC2), Amazon Relational Database Service (Amazon RDS), Amazon Redshift, Amazon Simple Storage Service (Amazon S3), dan masih banyak lagi. Untuk memulai dengan Automation, buka konsol Systems Manager. Pada panel navigasi, pilih Otomatisasi.

Otomasi membantu Anda membangun solusi otomatis untuk menerapkan, mengonfigurasi, dan mengelola AWS sumber daya dalam skala besar. Dengan Otomasi, Anda memiliki kontrol terperinci atas konkurensi otomatisasi Anda. Ini berarti Anda dapat menentukan berapa banyak sumber daya yang akan ditargetkan secara bersamaan, dan berapa banyak kesalahan yang dapat terjadi sebelum otomatisasi dihentikan.

Untuk membantu Anda memulai Otomasi, AWS kembangkan dan pertahankan beberapa runbook yang telah ditentukan sebelumnya. Bergantung pada kasus penggunaan Anda, Anda dapat menggunakan runbook yang telah ditentukan sebelumnya ini yang melakukan berbagai tugas, atau membuat runbook kustom Anda sendiri yang mungkin lebih sesuai dengan kebutuhan Anda. Untuk memantau kemajuan dan status otomatisasi, Anda dapat menggunakan konsol Otomasi Systems

Manager, atau alat baris perintah pilihan Anda. Otomatisasi juga terintegrasi dengan Amazon EventBridge untuk membantu Anda membangun arsitektur berbasis peristiwa dalam skala besar.

Bagaimana Otomasi dapat menguntungkan organisasi saya?

Otomasi menawarkan manfaat ini:

- Dukungan skrip dalam konten runbook

  Menggunakan `aws:executeScript` tindakan, Anda dapat menjalankan Python kustom dan PowerShell fungsi langsung dari runbook Anda. Ini memberi Anda fleksibilitas yang lebih besar dalam membuat runbook khusus karena Anda dapat menyelesaikan berbagai tugas yang tidak didukung oleh tindakan Otomasi lainnya. Anda juga memiliki kontrol yang lebih besar atas logika runbook. Untuk contoh bagaimana tindakan ini dapat digunakan dan bagaimana tindakan ini dapat membantu meningkatkan solusi otomatis yang ada, lihatMenyiapkan runbook Otomatisasi.

- Jalankan otomatisasi di beberapa Akun AWS dan Wilayah AWS dari lokasi terpusat

  Administrator dapat menjalankan otomatisasi pada sumber daya di beberapa akun dan Wilayah dari konsol Systems Manager.

- Keamanan operasi yang ditingkatkan

  Administrator memiliki tempat terpusat untuk memberikan dan mencabut akses ke runbook. Dengan hanya menggunakan kebijakan AWS Identity and Access Management (IAM), Anda dapat mengontrol pengguna atau grup individu mana di organisasi Anda yang dapat menggunakan Automation dan runbook mana yang dapat mereka akses.

- Otomatiskan tugas TI umum

  Mengotomatisasi tugas umum dapat membantu meningkatkan efisiensi operasional, menegakkan standar organisasi, dan mengurangi kesalahan operator. Misalnya, Anda dapat menggunakan `AWS-UpdateCloudFormationStackWithApproval` runbook untuk memperbarui sumber daya yang digunakan menggunakan templat. AWS CloudFormation Pembaruan memberlakukan templat baru. Anda dapat mengonfigurasi Otomasi untuk meminta persetujuan oleh satu atau beberapa pengguna sebelum pembaruan dimulai.

- Lakukan tugas yang mengganggu dengan aman dalam jumlah besar

  Otomatisasi mencakup fitur, seperti kontrol tarif, yang memungkinkan Anda mengontrol penyebaran otomatisasi di seluruh armada Anda dengan menentukan nilai konkurensi dan ambang

kesalahan. Untuk informasi selengkapnya tentang bekerja dengan kontrol tarif, lihatJalankan otomatisasi dalam skala besar.

- Merampingkan tugas-tugas kompleks

  Automation menyediakan runbook yang telah ditentukan sebelumnya yang merampingkan tugas-tugas kompleks dan memakan waktu seperti membuat golden (). Amazon Machine Images AMIs Misalnya, Anda dapat menggunakan `AWS-UpdateLinuxAmi` dan `AWS-UpdateWindowsAmi` runbook untuk membuat emas AMIs dari sumberAMI. Dengan menggunakan runbook ini, Anda dapat menjalankan skrip khusus sebelum dan sesudah pembaruan diterapkan. Anda juga dapat menyertakan atau mengecualikan paket perangkat lunak tertentu agar tidak diinstal. Untuk contoh cara menggunakan runbook ini, lihatTutorial.

- Tentukan kendala untuk input

  Anda dapat menentukan batasan dalam runbook khusus untuk membatasi nilai yang akan diterima Automation untuk parameter input tertentu. Misalnya, hanya `allowedPattern` akan menerima nilai untuk parameter input yang cocok dengan ekspresi reguler yang Anda tentukan. Jika Anda menentukan `allowedValues` parameter input, hanya nilai yang Anda tentukan di runbook yang diterima.

- Output tindakan otomatisasi log ke Amazon CloudWatch Logs

  Untuk memenuhi persyaratan operasional atau keamanan di organisasi Anda, Anda mungkin perlu memberikan catatan skrip yang dijalankan selama runbook. Dengan CloudWatch Log, Anda dapat memantau, menyimpan, dan mengakses file log dari berbagai fileLayanan AWS. Anda dapat mengirim output dari `aws:executeScript` tindakan ke grup CloudWatch log Log untuk tujuan debugging dan pemecahan masalah. Data log dapat dikirim ke grup log Anda dengan atau tanpa enkripsi AWS KMS menggunakan kunci KMS Anda. Untuk informasi selengkapnya, lihat Pencatatan output tindakan Otomatisasi dengan CloudWatch Logs.

- EventBridge Integrasi Amazon

  Otomatisasi didukung sebagai jenis target dalam EventBridge aturan Amazon. Ini berarti Anda dapat memicu runbook dengan menggunakan acara. Lihat informasi yang lebih lengkap di Pemantauan peristiwa Systems Manager dengan Amazon EventBridge dan Referensi: Pola dan jenis EventBridge acara Amazon untuk Systems Manager.

- Bagikan praktik terbaik organisasi

  Anda dapat menentukan praktik terbaik untuk pengelolaan sumber daya, tugas operasi, dan lainnya di runbook yang Anda bagikan di seluruh akun dan Wilayah.

Siapa yang harus menggunakan otomatisasi?

- Setiap AWS pelanggan yang ingin meningkatkan efisiensi operasional mereka dalam skala besar, mengurangi kesalahan yang terkait dengan intervensi manual, dan mengurangi waktu untuk menyelesaikan masalah umum.
- Pakar infrastruktur yang ingin mengotomatiskan tugas penerapan dan konfigurasi.
- Administrator yang ingin menyelesaikan masalah umum dengan andal, meningkatkan efisiensi pemecahan masalah, dan mengurangi operasi berulang.
- Pengguna yang ingin mengotomatiskan tugas yang biasanya mereka lakukan secara manual.

Apa itu otomatisasi?

Otomatisasi terdiri dari semua tugas yang didefinisikan dalam runbook, dan dilakukan oleh layanan Otomasi. Otomasi menggunakan komponen berikut untuk menjalankan otomatisasi.

Konsep	Detail
Otomatisasi runbook	Runbook Automation Systems Manager mendefinisikan otomatisasi (tindakan yang dilakukan Systems Manager pada node dan AWS sumber daya terkelola Anda). Otomatisasi mencakup beberapa runbook yang telah ditetapkan dan dapat Anda gunakan untuk melakukan beberapa tugas umum seperti memulai ulang satu instans Amazon EC2 atau lebih atau membuat Amazon Machine Image (AMI). Anda juga dapat membuat runbook Anda sendiri. Runbook menggunakan YAMAL atau JSON, dan mereka menyertakan langkah-langkah dan parameter yang Anda tentukan. Langkah-langkah berjalan secara berurutan. Untuk informasi selengkapnya, lihat Membuat runbook Anda sendiri. Runbooks adalah dokumen Systems Manager atau jenis `Automation` , sebagai lawan

Konsep	Detail
	`Command`, `Policy`, `Session` dokumen. Runbooks mendukung skema versi 0.3. Dokumen perintah menggunakan skema versi 1.2, 2.0, atau 2.2. Dokumen kebijakan menggunakan skema versi 2.0 atau yang lebih baru.
Tindakan otomasi	Otomatisasi yang didefinisikan dalam runbook mencakup satu langkah atau lebih. Setiap langkah dikaitkan dengan tindakan tertentu. Tindakan menentukan input, perilaku, dan output dari langkah. Langkah-langkah didefinis ikan dalam `mainSteps` bagian dari buku runbook Anda. Otomatisasi mendukung 20 jenis tindakan yang berbeda. Untuk informasi lebih lanjut, lihat Referensi tindakan Otomatisa si Systems Manager.
Kuota otomatisasi	Setiap Akun AWS dapat menjalankan 100 otomatisasi secara bersamaan. Ini termasuk otomatisasi anak (otomatisasi yang dimulai oleh otomatisasi lain), dan otomatisasi kontrol tarif. Jika Anda mencoba menjalankan lebih banyak otomatisasi daripada ini, Systems Manager menambahkan otomatisasi tambahan untuk antrian dan menampilkan status tertunda. Kuota ini dapat disesuaikan dengan menggunakan konkurensi adaptif. Untuk informasi selengkapnya, lihatMemungkinkan Otomasi untuk beradaptasi dengan kebutuhan konkurensi Anda. Untuk informasi selengkap nya tentang menjalankan otomatisasi, lihat. Menjalankan Otomatisasi

Konsep	Detail
Kuota antrian otomatisasi	Jika Anda mencoba untuk menjalankan otomatisasi lebih dari batas otomatisasi bersamaan, otomatisasi berikutnya ditambahkan ke antrean. Masing-masing Akun AWS dapat mengantri 5.000 otomatisasi. Ketika otomatisasi selesai (atau mencapai status berakhir), otomatisasi pertama dalam antrean dimulai.
Kuota otomatisasi kontrol tarif	Setiap Akun AWS dapat menjalankan 25 otomatisasi kontrol tarif secara bersamaan. Jika Anda mencoba untuk menjalankan kontrol tarif otomatisasi melebihi batas otomatisasi kontrol tarif bersamaan, Systems Manager menambahkan otomatisasi kontrol tarif berikutnya untuk antrean dan menampilkan status tertunda. Untuk informasi lebih lanjut tentang otomatisasi kontrol tarif berjalan, lihat Jalankan otomatisasi dalam skala besar.
Kuota antrean otomatisasi kontrol tarif	Jika Anda mencoba untuk menjalankan otomatisasi lebih dari batas otomatisasi kontrol tarif bersamaan, otomatisasi berikutnya ditambahkan ke antrean. Setiap Akun AWS dapat mengantre 1.000 otomatisasi kontrol tarif. Ketika otomatisasi selesai (atau mencapai status berakhir), otomatisasi pertama dalam antrean dimulai.

Topik

- Menyiapkan Otomatisasi
- Menjalankan Otomatisasi
- Penjadwalan otomatisasi
- Referensi tindakan Otomatisasi Systems Manager

- Membuat runbook Anda sendiri
- Referensi runbook Otomatisasi Systems Manager
- Tutorial
- Memahami status otomatisasi
- Pemecahan masalah Otomatisasi Systems Manager

Menyiapkan Otomatisasi

Untuk mengatur otomatisasi, kemampuan AWS Systems Manager, Anda harus memverifikasi akses pengguna ke Layanan otomatisasi dan mengonfigurasi peran secara situasional sehingga layanan dapat melakukan tindakan pada sumber daya Anda. Kami juga menyarankan Anda untuk ikut serta dalam mode konkurensi adaptif di preferensi Otomasi Anda. Konkurensi adaptif secara otomatis menskalakan kuota otomatisasi Anda untuk memenuhi kebutuhan Anda. Untuk informasi selengkapnya, lihat Memungkinkan Otomasi untuk beradaptasi dengan kebutuhan konkurensi Anda.

Untuk memastikan akses yang tepat ke AWS Systems Manager Otomatisasi, tinjau persyaratan peran layanan dan pengguna berikut.

Memverifikasi akses pengguna untuk runbook

Verifikasi bahwa Anda memiliki izin untuk menggunakan runbook. Jika pengguna, grup, atau peran Anda diberi izin administrator, maka Anda memiliki akses ke Otomasi Systems Manager. Jika Anda tidak memiliki izin administrator, administrator harus memberi Anda izin dengan menetapkan kebijakan `AmazonSSMFullAccess` terkelola, atau kebijakan yang memberikan izin yang sebanding, kepada pengguna, grup, atau peran Anda.

> **Important**
>
> Kebijakan IAM `AmazonSSMFullAccess` mengizinkan tindakan Systems Manager. Namun, beberapa runbook memerlukan izin untuk layanan lain, seperti runbook `AWS-ReleaseElasticIP`, yang memerlukan izin IAM untuk `ec2:ReleaseAddress`. Oleh karena itu, Anda harus meninjau tindakan yang diambil dalam buku runbook untuk memastikan pengguna, grup, atau peran Anda diberi izin yang diperlukan untuk melakukan tindakan yang disertakan dalam buku runbook.

Mengonfigurasi akses peran layanan (peran asumsi) untuk otomatisasi

Otomatisasi dapat dimulai di bawah konteks peran layanan (atau peran asumsi). Hal ini memungkinkan layanan untuk kemudian melakukan tindakan atas nama Anda. Jika Anda tidak menentukan peran asumsi, otomatisasi menggunakan konteks pengguna yang menjalankan otomatisasi.

Namun, situasi berikut mengharuskan Anda menentukan peran layanan untuk otomatisasi:

- Saat Anda ingin membatasi izin pengguna pada sumber daya, tetapi Anda ingin pengguna menjalankan otomatisasi yang memerlukan izin yang ditinggikan. Dalam skenario ini, Anda dapat membuat peran layanan dengan izin yang ditinggikan dan memungkinkan pengguna untuk menjalankan otomatisasi.
- Saat Anda membuat State Manager asosiasi Systems Manager yang menjalankan runbook.
- Saat Anda memiliki operasi yang Anda harapkan bisa berjalan lebih dari 12 jam.
- Ketika Anda menjalankan runbook yang tidak dimiliki oleh Amazon yang menggunakan `aws:executeScript` tindakan untuk memanggil AWS Operasi API atau untuk bertindak pada AWS sumber daya. Untuk informasi, lihat Izin untuk menggunakan runbook.

Jika Anda ingin membuat peran layanan untuk otomatisasi, Anda dapat menggunakan salah satu metode berikut.

Topik

Metode 1: Gunakan AWS CloudFormation untuk mengonfigurasi peran layanan untuk otomatisasi

Anda dapat membuat peran layanan untuk Otomatisasi, kapabilitas AWS Systems Manager, dari AWS CloudFormation templat. Setelah membuat peran layanan, Anda dapat menentukan peran layanan di runbook menggunakan parameter `AutomationAssumeRole`.

Buat peran layanan menggunakan AWS CloudFormation

Gunakan prosedur berikut untuk membuat yang AWS Identity and Access Management (IAM) role diperlukan untuk otomatisasi Systems Manager dengan menggunakan AWS CloudFormation.

Untuk membuat IAM role yang diperlukan

1. Unduh dan unzip `AWS-SystemsManager-AutomationServiceRole.zip` file. File ini mencakup `AWS-SystemsManager-AutomationServiceRole.yaml` AWS CloudFormation file templat.
2. Buka AWS CloudFormation konsol di https://console.aws.amazon.com/cloudformation.
3. Pilih Buat tumpukan.
4. Di bagian Tentukan templat, pilih Unggah file templat.
5. Pilih Telusuri, dan kemudian pilih `AWS-SystemsManager-AutomationServiceRole.yaml` AWS CloudFormation file templat.
6. Pilih Berikutnya.
7. Di halaman Tentukan detail tumpukan, di bidang Nama tumpukan, masukkan nama.
8. Pada halaman Konfigurasikan pilihan tumpukan, Anda tidak perlu membuat pilihan apa pun. Pilih Berikutnya.
9. Pada halaman Ulasan, gulir ke bawah dan pilih Saya mengakui AWS CloudFormation adanya kemungkinan membuat opsi sumber daya IAM.
10. Pilih Buat.

CloudFormation menunjukkan status CREATE_IN_PROGRESS selama sekitar tiga menit. Perubahan status ke CREATE_COMPLETE setelah tumpukan dibuat dan peran Anda siap untuk digunakan.

> **Important**
>
> Jika Anda menjalankan alur kerja otomatisasi yang menjalankan layanan lain dengan menggunakan AWS Identity and Access Management peran layanan (IAM), pastikan bahwa peran layanan harus dikonfigurasi dengan izin untuk menjalankan layanan tersebut. Persyaratan ini berlaku untuk semua AWS Runbook otomatisasi (`AWS-*` runbook) seperti `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup`, dan `AWS-RestartEC2Instance` runbook, untuk beberapa nama. Persyaratan ini juga berlaku untuk setiap runbook otomatisasi kustom yang Anda buat bahwa menjalankan lainLayanan AWS

dengan menggunakan tindakan yang memanggil layanan lainnya. Misalnya, jika Anda menggunakan `aws:executeAwsApi`, `aws:createStack`, atau `aws:copyImage` tindakan, konfigurasikan peran layanan dengan izin untuk menjalankan layanan tersebut. Anda dapat memberikan izin keLayanan AWS kebijakan inline IAM ke peran tersebut. Untuk informasi selengkapnya, lihat (Opsional) Tambahkan kebijakan sebaris Otomasi atau kebijakan terkelola pelanggan untuk memanggil lainnya Layanan AWS.

Salin informasi peran untuk Otomatisasi

Gunakan prosedur berikut untuk menyalin informasi tentang Peran layanan otomatisasi dari AWS CloudFormation konsol. Anda harus menentukan peran ini ketika menggunakan runbook.

> **Note**
>
> Anda tidak perlu menyalin informasi peran menggunakan prosedur ini jika Anda menjalankan `AWS-UpdateLinuxAmi` atau `AWS-UpdateWindowsAmi` runbook. Runbook ini sudah memiliki peran yang diperlukan yang ditetapkan sebagai nilai default. Peran yang ditentukan dalam runbook ini menggunakan kebijakan terkelola IAM.

Menyalin nama peran

1. Buka AWS CloudFormation konsol di https://console.aws.amazon.com/cloudformation.
2. Pilih Otomatisasi Nama tumpukan yang Anda buat dalam prosedur sebelumnya.
3. Pilih tab Sumber Daya.
4. Pilih tautan ID Fisik untuk AutomationServiceRole. Konsol IAM membuka ringkasan Peran layanan otomatisasi.
5. Salin Amazon Resource Name (ARN) di samping Peran ARN. ARN serupa dengan yang berikut ini: `arn:aws:iam::12345678:role/AutomationServiceRole`
6. Tempel ARN ke file teks untuk digunakan nanti.

Anda telah selesai mengonfigurasi peran layanan untuk otomatisasi. Anda sekarang dapat menggunakan peran layanan otomatisasi ARN di runbook Anda.

Metode 2: Gunakan IAM untuk mengonfigurasi peran untuk Otomatisasi

Jika Anda harus membuat peran layanan untuk otomatisasi, kemampuan AWS Systems Manager, selesaikan tugas berikut. Untuk informasi lebih lanjut tentang kapan peran layanan diperlukan untuk otomatisasi, lihat Menyiapkan Otomatisasi.

Tugas

- Tugas 1: Buat peran layanan untuk otomatisasi
- Tugas 2: Lampirkan PassRole kebijakan iam: ke peran Otomasi Anda

Tugas 1: Buat peran layanan untuk otomatisasi

Gunakan prosedur berikut untuk membuat peran layanan (atau peran asumsi) untuk Otomatisasi Systems Manager.

> **Note**
>
> Anda juga dapat menggunakan peran ini di runbook, seperti `AWS-CreateManagedLinuxInstance` runbook. Dengan menggunakan peran ini, atau Amazon Resource Name (ARN) AWS Identity and Access Management (IAM) role, dalam runbook memungkinkan Otomatisasi untuk melakukan tindakan di lingkungan Anda, seperti meluncurkan contoh baru dan melakukan tindakan atas nama Anda.

Untuk membuat IAM role dan mengizinkan Otomatisasi mengasumsikannya

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran, lalu pilih Buat peran.
3. Di bawah Pilih jenis entitas tepercaya, pilih AWS layanan.
4. Di bagian Pilih kasus penggunaan, pilih Systems Manager, lalu pilih Berikutnya: Izin.
5. Pada halaman Kebijakan izin terlampir, cari AutomationRole kebijakan AmazonSSM, pilih, lalu pilih Berikutnya: Tinjau.
6. Pada halaman Ulasan, masukkan nama di kotak Nama peran, dan kemudian masukkan deskripsi.
7. Pilih Buat peran. Sistem mengembalikan Anda ke halaman Peran.

8. Pada halaman Peran, pilih peran yang baru Anda buat untuk membuka halaman Ringkasan. Catat Nama peran dan Peran ARN. Anda akan menentukan peran ARN saat Anda melampirkan PassRole kebijakan iam: ke akun IAM Anda di prosedur berikutnya. Anda juga dapat menentukan nama peran dan ARN di runbook.

> **Note**
>
> Kebijakan `AmazonSSMAutomationRole` menetapkan Izin peran otomatisasi untuk subset AWS Lambda fungsi dalam akun Anda. Fungsi ini dimulai dengan "Otomatisasi". Jika Anda berencana untuk menggunakan otomatisasi dengan fungsi Lambda, Lambda ARN harus menggunakan format berikut:
> `"arn:aws:lambda:*:*:function:Automation*"`
> Jika Anda memiliki fungsi Lambda yang ARN-nya tidak menggunakan format ini, Anda juga harus melampirkan kebijakan Lambda tambahan ke peran otomatisasi Anda, seperti kebijakan. AWSLambdaRole Kebijakan atau peran tambahan harus menyediakan akses yang lebih luas ke fungsi Lambda dalam Akun AWS.

Setelah membuat peran layanan Anda, kami sarankan untuk mengedit kebijakan kepercayaan untuk membantu mencegah masalah wakil lintas layanan yang membingungkan. Masalah deputi yang membingungkan adalah masalah keamanan di mana entitas yang tidak memiliki izin untuk melakukan tindakan dapat memaksa entitas yang lebih istimewa untuk melakukan tindakan. Pada tahunAWS, peniruan lintas layanan dapat mengakibatkan masalah wakil yang membingungkan. Peniruan identitas lintas layanan dapat terjadi ketika satu layanan (layanan panggilan) memanggil layanan lain (layanan yang disebut). Layanan panggilan dapat dimanipulasi untuk menggunakan izinnya untuk bertindak atas sumber daya pelanggan lain dengan cara yang seharusnya tidak memiliki izin untuk mengakses. Untuk mencegah hal ini, AWS sediakan alat yang membantu Anda melindungi data Anda untuk semua layanan dengan prinsip layanan yang telah diberikan akses ke sumber daya di akun Anda.

Sebaiknya gunakan kunci konteks kondisi `aws:SourceAccount`global `aws:SourceArn`dan global dalam kebijakan sumber daya untuk membatasi izin yang diberikan Automation layanan lain ke sumber daya. Jika `aws:SourceArn` nilainya tidak berisi ID akun, seperti ARN bucket Amazon S3, Anda harus menggunakan kedua kunci konteks kondisi global untuk membatasi izin. Jika Anda menggunakan kunci konteks kondisi global dan `aws:SourceArn` nilainya berisi ID akun, `aws:SourceAccount` nilai dan akun dalam `aws:SourceArn` nilai harus menggunakan ID akun yang sama saat digunakan dalam pernyataan kebijakan yang sama.

Gunakan `aws:SourceArn` jika Anda hanya ingin satu sumber daya dikaitkan dengan akses lintas layanan. Gunakan `aws:SourceAccount` jika Anda ingin mengizinkan sumber daya apa pun di akun itu dikaitkan dengan penggunaan lintas layanan. Nilai `aws:SourceArn` harus ARN untuk eksekusi otomatisasi. Jika Anda tidak mengetahui ARN lengkap sumber daya atau jika Anda menentukan beberapa sumber daya, gunakan kunci kondisi konteks `aws:SourceArn` global dengan wildcard (*) untuk bagian ARN yang tidak diketahui. Sebagai contoh, `arn:aws:ssm:*:123456789012:automation-execution/*`.

Contoh berikut menunjukkan bagaimana Anda dapat menggunakan `aws:SourceArn` dan kunci konteks kondisi `aws:SourceAccount` global untuk Otomasi untuk mencegah masalah wakil yang membingungkan.

```
{
  "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/*"
        }
      }
    }
  ]
}
```

Untuk memodifikasi kebijakan kepercayaan peran

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran.
3. Dalam daftar peran di akun Anda, pilih nama peran layanan Otomasi Anda.

4. Pilih tab Hubungan kepercayaan, dan kemudian pilihUbah hubungan kepercayaan.
5. Edit kebijakan kepercayaan menggunakan `aws:SourceArn` dan kunci konteks kondisi `aws:SourceAccount` global untuk Otomasi untuk mencegah masalah wakil yang membingungkan.
6. Pilih Perbarui Kebijakan Kepercayaan untuk menyimpan perubahan Anda.

(Opsional) Tambahkan kebijakan sebaris Otomasi atau kebijakan terkelola pelanggan untuk memanggil lainnya Layanan AWS

Jika Anda menjalankan otomatisasi yang memanggil orang lain Layanan AWS dengan menggunakan peran layanan IAM, peran layanan harus dikonfigurasi dengan izin untuk memanggil layanan tersebut. Persyaratan ini berlaku untuk semua AWS Runbook otomatisasi (`AWS-*` runbook) seperti `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup`, dan `AWS-RestartEC2Instance` runbook, untuk beberapa nama. Persyaratan ini juga berlaku untuk setiap runbook kustom yang Anda buat yang memanggil orang lain Layanan AWS dengan menggunakan tindakan yang memanggil layanan lain. Misalnya, jika Anda menggunakan `aws:executeAwsApi`, `aws:CreateStack`, atau `aws:copyImage` tindakan,untuk beberapa nama, konfigurasikan peran layanan dengan izin untuk menjalankan layanan tersebut. Anda dapat memberikan izin kepada orang lain Layanan AWS dengan menambahkan kebijakan inline IAM atau kebijakan yang dikelola pelanggan ke peran tersebut.

Untuk menyematkan kebijakan yang selaras bagi peran layanan (konsol IAM)

1. Masuk ke AWS Management Console dan buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran.
3. Dalam daftar peran, pilih nama peran yang ingin Anda edit.
4. Pilih tab Izin.
5. Di menu tarik-turun Tambah izin, pilih Lampirkan kebijakan atau Buat kebijakan sebaris.
6. Jika Anda memilih Lampirkan kebijakan, pilih kotak centang di samping kebijakan yang ingin Anda tambahkan, lalu pilih Tambahkan izin.
7. Jika Anda memilih Buat kebijakan sebaris, pilih tab JSON.
8. Masukkan dokumen Kebijakan JSON untuk yang ingin Layanan AWS Anda panggil. Berikut adalah dua contoh dokumen Kebijakan JSON.

   Amazon S3 PutObject dan Contoh GetObject

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

Amazon EC2 CreateSnapshot dan Contoh DescribeSnapShots

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"ec2:CreateSnapshot",
         "Resource":"*"
      },
      {
         "Effect":"Allow",
         "Action":"ec2:DescribeSnapshots",
         "Resource":"*"
      }
   ]
}
```

Untuk rincian bahasa kebijakan IAM, lihat Referensi kebijakan IAM JSON di Panduan Pengguna IAM.

9. Setelah selesai, pilih Tinjau kebijakan. Validator Kebijakan melaporkan kesalahan sintaksis.
10. Pada halaman Kebijakan ulasan, masukkan Nama untuk kebijakan yang Anda buat. Ulas Ringkasan kebijakan untuk melihat izin yang diberikan oleh kebijakan Anda. Kemudian pilih Buat kebijakan untuk menyimpan pekerjaan Anda.

11. Setelah Anda membuat kebijakan yang selaras, ia akan secara otomatis tertanam di pengguna atau peran Anda.

Tugas 2: Lampirkan PassRole kebijakan iam: ke peran Otomasi Anda

Gunakan prosedur berikut untuk melampirkan `iam:PassRole` kebijakan untuk peran layanan otomatisasi Anda. Hal ini memungkinkan layanan Otomatisasi untuk lulus peran layanan lain atau kemampuan Systems Manager ketika menjalankan otomatisasi.

Untuk melampirkan PassRole kebijakan iam: ke peran Otomasi Anda

1. Di halaman Ringkasan untuk peran yang baru saja Anda buat, pilih tab Izin.
2. Pilih Tambah kebijakan inline.
3. Di halaman Buat kebijakan, pilih tab Visual editor.
4. Pilih Layanan, lalu pilih IAM.
5. Pilih Pilih tindakan.
6. Di kotak teks tindakan Filter**`PassRole`**, ketik, lalu pilih PassRoleopsi.
7. Pilih Sumber Daya. Verifikasi bahwa Spesifik dipilih, dan kemudian pilih Tambahkan ARN.
8. Di Spesifikasikan ARN untuk bidang peran, tempel otomatisasi peran ARN yang Anda salin pada akhir tugas 1. Sistem mengisi Akun dan Nama peran dengan bidang jalur.

   > **Note**
   >
   > Jika Anda ingin peran layanan otomatisasi untuk melampirkan peran profil instans IAM ke instans EC2, maka Anda harus menambahkan ARN peran profil instans IAM. Hal ini memungkinkan peran layanan otomatisasi untuk bisa melewati peran profil instans IAM ke instans EC2 target.

9. Pilih Tambahkan.
10. Pilih Tinjau kebijakan.
11. Pada halaman Tinjau kebijakan, masukkan nama dan deskripsi kebijakan, dan pilih Buat Kebijakan.

Memungkinkan Otomasi untuk beradaptasi dengan kebutuhan konkurensi Anda

Secara default, Automation memungkinkan Anda menjalankan hingga 100 otomatisasi bersamaan sekaligus. Automation juga menyediakan pengaturan opsional yang dapat Anda gunakan untuk menyesuaikan kuota otomatisasi konkurensi Anda secara otomatis. Dengan pengaturan ini, kuota otomatisasi konkurensi Anda dapat mengakomodasi hingga 500 otomatisasi bersamaan, tergantung pada sumber daya yang tersedia.

> **Note**
>
> Jika otomatisasi Anda memanggil operasi API, penskalaan secara adaptif ke target Anda dapat mengakibatkan pengecualian pembatasan. Jika pengecualian pembatasan berulang terjadi saat menjalankan otomatisasi dengan konkurensi adaptif diaktifkan, Anda mungkin harus meminta peningkatan kuota untuk operasi API jika tersedia.

Mengaktifkan konkurensi adaptif (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Pada panel navigasi, pilih Otomatisasi.
3. Pilih tab Preferensi, dan kemudian pilih Edit.
4. Pilih kotak centang di sampingAktifkan konkurensi adaptif.
5. Pilih Simpan.

Menerapkan kontrol perubahan untuk Otomasi

Secara default, Automation memungkinkan Anda untuk menggunakan runbook tanpa batasan tanggal dan waktu. Dengan mengintegrasikan Otomasi denganChange Calendar, Anda dapat menerapkan kontrol perubahan untuk semua otomatisasi di AndaAkun AWS. Dengan pengaturan ini, prinsipalAWS Identity and Access Management (IAM) di akun Anda hanya dapat menjalankan otomatisasi selama periode waktu yang diizinkan oleh kalender perubahan Anda. Untuk mempelajari selengkapnya tentang caraChange Calendar menggunakan aplikasiBekerja dengan Change Calendar.

Mengaktifkan kontrol perubahan (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.

2. Pada panel navigasi, pilih Otomatisasi.
3. Pilih tab Preferensi, dan kemudian pilih Edit.
4. Pilih kotak centang di samping AktifkanChange Calendar integrasi.
5. Dalam daftar dropdown Pilih kalender perubahan, pilih kalender perubahan yang Anda inginkan untuk diikuti Otomatisasi.
6. Pilih Simpan.

Menjalankan Otomatisasi

Bagian ini mencakup informasi tentang cara menjalankan runbook otomatisasi. Otomatisasi adalah kemampuan AWS Systems Manager. Untuk tutorial yang lebih rinci tentang cara menjalankan otomatisasi untuk kasus penggunaan Anda, lihatTutorial.

Konten

Jalankan otomatisasi

Ketika Anda menjalankan otomatisasi, secara default, otomatisasi berjalan dalam konteks pengguna yang memulai otomatisasi. Ini berarti, misalnya, jika pengguna Anda memiliki izin administrator, maka otomatisasi berjalan dengan izin administrator dan akses penuh ke sumber daya yang dikonfigurasi oleh otomatisasi. Sebagai praktik terbaik keamanan, kami menyarankan Anda menjalankan otomatisasi dengan menggunakan peran layanan IAM yang dikenal dalam kasus ini sebagai peran asumsi yang dikonfigurasi dengan kebijakan terkelola AutomationRole AmazonSSM. Anda mungkin perlu menambahkan kebijakan IAM tambahan ke peran asumsi Anda untuk menggunakan berbagai runbook. Menggunakan peran layanan IAM untuk menjalankan otomatisasi disebut administrasi terdelegasi.

Ketika Anda menggunakan peran layanan, otomatisasi diizinkan untuk berjalan terhadap AWS sumber daya, tetapi pengguna yang menjalankan otomatisasi telah membatasi akses (atau tidak

ada akses) ke sumber daya tersebut. Misalnya, Anda dapat mengonfigurasi peran layanan dan menggunakannya dengan otomatisasi untuk memulai ulang instans Amazon Elastic Compute Cloud (Amazon EC2). Otomatisasi adalah kemampuan AWS Systems Manager. Otomatisasi memulai ulang instas, tetapi peran layanan tidak memberikan izin pengguna untuk mengakses contoh tersebut.

Anda dapat menentukan peran layanan pada saat runtime ketika sedang menjalankan otomatisasi, atau Anda dapat membuat runbook kustom dan menentukan peran layanan secara langsung di runbook. Jika Anda menentukan peran layanan, baik saat runtime atau runbook, maka layanan berjalan dalam konteks peran layanan yang ditentukan. Jika Anda tidak menentukan peran layanan, maka sistem membuat sesi sementara dalam konteks pengguna dan menjalankan otomatisasi.

> **Note**
>
> Anda harus menentukan peran layanan untuk otomatisasi yang Anda harapkan agar bisa berjalan lebih dari 12 jam. Jika Anda memulai otomatisasi yang berjalan lama dalam konteks pengguna, sesi pengguna sementara berakhir setelah 12 jam.

Administrasi yang didelegasikan memastikan keamanan dan kontrol yang lebih tinggi terhadap AWS sumber daya. Hal ini juga memungkinkan pengalaman audit yang ditingkatkan karena tindakan yang dilakukan terhadap sumber daya Anda oleh peran layanan pusat bukan beberapa akun IAM.

Sebelum Anda memulai

Sebelum Anda menyelesaikan prosedur berikut, Anda harus membuat peran layanan IAM dan mengonfigurasi hubungan kepercayaan untuk Otomasi, kemampuan. AWS Systems Manager Untuk informasi selengkapnya, lihat Tugas 1: Buat peran layanan untuk otomatisasi.

Prosedur berikut menjelaskan cara menggunakan konsol Systems Manager atau alat baris perintah pilihan Anda untuk menjalankan otomatisasi sederhana.

Menjalankan otomatisasi sederhana (konsol)

Prosedur berikut menjelaskan cara menggunakan konsol Systems Manager untuk menjalankan otomatisasi sederhana.

Untuk menjalankan otomatisasi sederhana

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Otomatisasi, lalu pilih Eksekusi otomatisasi.

3. Di daftar Dokumen otomatisasi, pilih runbook. Pilih satu opsi atau lebih di panel Kategori dokumen untuk memfilter dokumen SSM sesuai dengan tujuannya. Untuk melihat runbook yang Anda miliki, pilih tab Dimiliki oleh saya. Untuk melihat runbook yang dibagikan dengan akun Anda, pilih tab Dibagikan dengan saya. Untuk melihat semua runbook, pilih tab Semua dokumen.

   > **Note**
   >
   > Anda dapat melihat informasi tentang runbook dengan memilih nama runbook.

4. Di bagian Detail dokumen, verifikasi bahwa Versi dokumen diatur ke versi yang ingin Anda jalankan. Sistem ini termasuk pilihan versi berikut:

   - Versi default saat runtime — Pilih opsi ini jika runbook Otomasi diperbarui secara berkala dan versi default baru ditetapkan.
   - Versi terbaru saat runtime — Pilih opsi ini jika runbook Otomasi diperbarui secara berkala, dan Anda ingin menjalankan versi yang terbaru diperbarui.
   - 1 (Default) - Pilih opsi ini untuk menjalankan versi pertama dokumen, yang merupakan default.

5. Pilih Berikutnya.
6. Dalam bagian Mode Eksekusi, pilih Eksekusi sederhana.
7. Di bagian Parameter input, tentukan input yang diperlukan. Secara opsional, Anda dapat memilih peran layanan IAM dari daftar. AutomationAssumeRole
8. (Opsional) Pilih CloudWatch alarm untuk diterapkan ke otomatisasi Anda untuk pemantauan. Untuk memasang CloudWatch alarm ke otomatisasi Anda, prinsip IAM yang memulai otomatisasi harus memiliki izin untuk `iam:createServiceLinkedRole` tindakan tersebut. Untuk informasi selengkapnya tentang CloudWatch alarm, lihat Menggunakan CloudWatch alarm Amazon. Perhatikan bahwa jika alarm Anda aktif, otomatisasi dihentikan. Jika Anda menggunakanAWS CloudTrail, Anda akan melihat panggilan API di jejak Anda.
9. Pilih Eksekusi.

Konsol menampilkan status otomatisasi. Jika otomatisasi gagal berjalan, lihat Pemecahan masalah Otomatisasi Systems Manager.

Menjalankan otomasi sederhana (baris perintah)

Prosedur berikut menjelaskan cara menggunakan AWS CLI (di Linux atau Windows) atau AWS Tools for PowerShell menjalankan otomatisasi sederhana.

Untuk menjalankan otomatisasi sederhana

1. Instal dan konfigurasikan AWS CLI atau AWS Tools for PowerShell, jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI dan Menginstal AWS Tools for PowerShell.

2. Jalankan perintah berikut untuk memulai otomatisasi sederhana. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   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
   ```

   Berikut adalah contoh menggunakan runbook `AWS-RestartEC2Instance` untuk memulai ulang instans EC2 tertentu.

   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"}
```

Sistem mengembalikan informasi seperti berikut.

Linux & macOS

```
{
    "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0123456789ab"
}
```

Windows

```
{
    "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0123456789ab"
}
```

PowerShell

```
4105a4fc-f944-11e6-9d32-0123456789ab
```

3. Jalankan perintah berikut untuk mengambil status otomatisasi.

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"}
```

Sistem mengembalikan informasi seperti berikut.

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                     : {}
```

Jalankan otomatisasi dengan pemberi persetujuan

Prosedur berikut menjelaskan cara menggunakan AWS Systems Manager konsol dan AWS Command Line Interface (AWS CLI) untuk menjalankan otomatisasi yang sudah disetujui menggunakan eksekusi sederhana. Otomatisasi menggunakan tindakan otomatisasi `aws:approve`, yang menghentikan otomatisasi sementara sampai otomatisasi utama menyetujui atau menolak tindakan tersebut. Otomatisasi berjalan dalam konteks pengguna saat ini. Ini berarti bahwa Anda tidak perlu mengkonfigurasi izin IAM tambahan selama Anda memiliki izin untuk menggunakan runbook, dan tindakan apa pun yang disebut oleh runbook. Jika Anda memiliki izin administrator di IAM, maka Anda sudah memiliki izin untuk menggunakan runbook ini.

Sebelum Anda memulai

Selain input standar yang diperlukan oleh runbook, tindakan `aws:approve` membutuhkan dua parameter berikut:

- Daftar pemberi persetujuan. Daftar pemberi persetujuan harus berisi setidaknya satu pemberi persetujuan dalam bentuk nama pengguna atau ARN pengguna. Jika tersedia beberapa pemberi persetujuan, jumlah persetujuan minimum yang terkait juga harus ditentukan dalam runbook.
- Topik Amazon Simple Notification Service (Amazon SNS) ARN. Nama topik Amazon SNS harus dimulai dengan `Automation`.

Prosedur ini mengasumsikan bahwa Anda telah membuat topik Amazon SNS, yang diperlukan untuk memberikan permintaan persetujuan. Untuk informasi lebih lanjut, lihat Buat topik dalam Panduan Developer Amazon Simple Notification Service.

Menjalankan otomatisasi dengan pemberi persetujuan (konsol)

Untuk menjalankan otomatisasi dengan pemberi persetujuan

Prosedur berikut menjelaskan cara menggunakan konsol Systems Manager untuk menjalankan otomatisasi dengan pemberi persetujuan.

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Otomatisasi, lalu pilih Eksekusi otomatisasi.
3. Di daftar Dokumen otomatisasi, pilih runbook. Pilih satu opsi atau lebih di panel Kategori dokumen untuk memfilter dokumen SSM sesuai dengan tujuannya. Untuk melihat runbook yang Anda miliki, pilih tab Dimiliki oleh saya. Untuk melihat runbook yang dibagikan dengan akun Anda, pilih tab Dibagikan dengan saya. Untuk melihat semua runbook, pilih tab Semua dokumen.

   > **Note**
   >
   > Anda dapat melihat informasi tentang runbook dengan memilih nama runbook.

4. Di bagian Detail dokumen, verifikasi bahwa Versi dokumen diatur ke versi yang ingin Anda jalankan. Sistem ini termasuk pilihan versi berikut:

   - Versi default saat runtime — Pilih opsi ini jika runbook Otomasi diperbarui secara berkala dan versi default baru ditetapkan.
   - Versi terbaru saat runtime — Pilih opsi ini jika runbook Otomasi diperbarui secara berkala, dan Anda ingin menjalankan versi yang terbaru diperbarui.
   - 1 (Default) - Pilih opsi ini untuk menjalankan versi pertama dokumen, yang merupakan default.
5. Pilih Berikutnya.
6. Pada halaman Eksekusi dokumen otomatisasi, pilih Eksekusi sederhana.
7. Di bagian Parameter input, tentukan parameter input yang diperlukan.

   Misalnya, jika Anda memilih **`AWS-StartEC2InstanceWithApproval`** runbook, maka Anda harus menentukan atau memilih ID instance untuk InstanceIdparameter tersebut.
8. Di bagian Penyetuju, tentukan nama pengguna atau ARN pengguna pemberi persetujuan untuk tindakan otomatisasi.
9. Di bagian SnstoCharn, tentukan topik SNS ARN untuk digunakan saat mengirim pemberitahuan persetujuan. Nama topik SNS harus dimulai dengan Otomatisasi.

10. Secara opsional, Anda dapat memilih peran layanan IAM dari daftar. AutomationAssumeRole Jika Anda menargetkan lebih dari 100 akun dan Wilayah, Anda harus menentukan. `AWS-SystemsManager-AutomationAdministrationRole`
11. Pilih Eksekusi otomatisasi.

Pemberi persetujuan yang ditentukan menerima notifikasi Amazon SNS dengan detail untuk menyetujui atau menolak otomatisasi. Tindakan persetujuan ini berlaku selama 7 hari sejak tanggal penerbitan dan dapat dikeluarkan menggunakan konsol Systems Manager atau AWS Command Line Interface (AWS CLI).

Jika Anda memilih untuk menyetujui otomatisasi, otomatisasi terus menjalankan langkah-langkah yang disertakan dalam runbook tertentu. Konsol menampilkan status otomatisasi. Jika otomatisasi gagal berjalan, lihat Pemecahan masalah Otomatisasi Systems Manager.

Untuk menyetujui atau menolak otomatisasi

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Otomatisasi, dan kemudian pilih otomatisasi yang dijalankan dalam prosedur sebelumnya.
3. Pilih Tindakan dan kemudian pilih Setujui/Tolak.
4. Pilih Setujui atau Tolak dan berikan komentar (opsional).
5. Pilih Kirim.

Menjalankan otomatisasi dengan pemberi persetujuan (bari perintah)

Prosedur berikut menjelaskan cara menggunakan AWS CLI (di Linux atau Windows) atau AWS Tools for PowerShell menjalankan otomatisasi sederhana.

Untuk menjalankan otomatisasi dengan pemberi persetujuan

1. Instal dan konfigurasikan AWS CLI atau AWS Tools for PowerShell, jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI dan Menginstal AWS Tools for PowerShell.

2. Jalankan perintah berikut untuk menjalankan otomatisasi dengan pemberi persetujuan. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri. Di bagian nama dokumen, tentukan runbook yang mencakup tindakan otomasi, `aws:approve`.

   Untuk`Approvers`, tentukan nama pengguna atau ARN pengguna dari pemberi persetujuan untuk tindakan tersebut. Untuk `SNSTopic`, tentukan topik SNS ARN untuk digunakan saat mengirim pemberitahuan persetujuan. Nama topik Amazon SNS harus dimulai dengan `Automation`.

   > **Note**
   >
   > Nama spesifik dari nilai parameter untuk pemberi persetujuan dan topik SNS bergantung pada nilai yang ditentukan dalam runbook yang Anda pilih.

   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"
   ```

```
    }
```

Sistem mengembalikan informasi seperti berikut.

Linux & macOS

```
{
    "AutomationExecutionId": "df325c6d-b1b1-4aa0-8003-6cb7338213c6"
}
```

Windows

```
{
    "AutomationExecutionId": "df325c6d-b1b1-4aa0-8003-6cb7338213c6"
}
```

PowerShell

```
df325c6d-b1b1-4aa0-8003-6cb7338213c6
```

Menyetujui otomatisasi

- Gunakan perintah berikut untuk menyetujui otomatisasi. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

  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"}
```

Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Untuk menyangkal otomatisasi

- Jalankan perintah berikut untuk menolak otomatisasi. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

  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"}
  ```

  Tidak ada output jika perintah berhasil.

Jalankan otomatisasi dalam skala besar

Dengan AWS Systems Manager Otomasi, Anda dapat menjalankan otomatisasi pada armada sumber AWS daya dengan menggunakan target. Selain itu, Anda dapat mengontrol deployment otomatisasi tersebut dengan menentukan nilai konkurensi dan ambang kesalahan. Fitur konkurensi dan fitur ambang batas kesalahan secara kolektif disebut pengendalian rate. Nilai konkurensi menentukan berapa banyak sumber daya yang diizinkan untuk menjalankan otomatisasi secara bersamaan. Otomasi juga menyediakan mode konkurensi adaptif yang dapat Anda pilih. Konkurensi adaptif secara otomatis menskalakan kuota otomatisasi Anda dari 100 otomatisasi yang berjalan secara bersamaan hingga 500. Ambang kesalahan menentukan berapa banyak otomatisasi yang diperbolehkan untuk gagal sebelum Systems Manager berhenti mengirim otomatisasi ke sumber daya lainnya.

Untuk informasi lebih lanjut tentang ambang batas konkurensi dan kesalahan, lihat Kontrol otomatisasi pada skala. Untuk informasi selengkapnya tentang target, lihat Memetakan target untuk otomatisasi.

Prosedur berikut menunjukkan cara mengaktifkan konkurensi adaptif, dan cara menjalankan otomatisasi dengan target dan kontrol tingkat dengan menggunakan konsol Systems Manager dan AWS Command Line Interface ()AWS CLI.

Menjalankan otomatisasi dengan kontrol tarif dan target (konsol)

Prosedur berikut menjelaskan cara menjalankan otomatisasi dengan target dan kontrol tarif dengan menggunakan konsol Systems Manager.

Untuk menjalankan otomatisasi dengan target dan kontrol tarif

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Otomatisasi, lalu pilih Eksekusi otomatisasi.
3. Di daftar Dokumen otomatisasi, pilih runbook. Pilih satu opsi atau lebih di panel Kategori dokumen untuk memfilter dokumen SSM sesuai dengan tujuannya. Untuk melihat runbook yang Anda miliki, pilih tab Dimiliki oleh saya. Untuk melihat runbook yang dibagikan dengan akun Anda, pilih tab Dibagikan dengan saya. Untuk melihat semua runbook, pilih tab Semua dokumen.

> **Note**
>
> Anda dapat melihat informasi tentang runbook dengan memilih nama runbook.

4. Di bagian Detail dokumen, verifikasi bahwa Versi dokumen diatur ke versi yang ingin Anda jalankan. Sistem ini termasuk pilihan versi berikut:

   - Versi default saat runtime — Pilih opsi ini jika runbook Otomasi diperbarui secara berkala dan versi default baru ditetapkan.
   - Versi terbaru saat runtime — Pilih opsi ini jika runbook Otomasi diperbarui secara berkala, dan Anda ingin menjalankan versi yang terbaru diperbarui.
   - 1 (Default) - Pilih opsi ini untuk menjalankan versi pertama dokumen, yang merupakan default.

5. Pilih Berikutnya.
6. Di bagian Eksekusi Mode, pilih Pengendalian Rate. Anda harus menggunakan mode ini atau Multi-Akun dan Wilayah jika Anda ingin menggunakan target dan kontrol tarif.
7. Di bagian Target, pilih bagaimana Anda ingin menargetkan AWS sumber daya tempat Anda ingin menjalankan otomatisasi. Pilihan ini diperlukan.

   a. Gunakan daftar Parameter untuk memilih parameter. Item dalam daftar Parameter ditentukan oleh parameter di Runbook otomatisasi yang Anda pilih pada awal prosedur ini. Dengan memilih parameter, Anda menentukan jenis sumber daya di mana alur kerja otomatisasi berjalan.

   b. Gunakan daftar Target untuk memilih cara Anda menargetkan sumber daya.

      i. Jika Anda ingin menargetkan sumber daya dengan menggunakan nilai parameter, masukkan nilai parameter untuk parameter yang Anda pilih di bagian Parameter input.

      ii. Jika Anda ingin menargetkan sumber daya dengan menggunakan AWS Resource Groups, lalu pilih nama grup dari daftar Grup Sumber Daya.

      iii. Jika Anda memilih untuk menargetkan sumber daya dengan menggunakan tag, masukkan kunci tag dan nilai tag dalam bidang yang disediakan (opsional). Pilih Tambahkan.

      iv. Jika Anda ingin menjalankan Runbook otomatisasi pada semua contoh di saat ini Akun AWS dan Wilayah AWS, pilih Semua instans.

8. Di bagian Parameter input, tentukan input yang diperlukan. Secara opsional, Anda dapat memilih peran layanan IAM dari daftar. AutomationAssumeRole

> **Note**
>
> Anda mungkin tidak perlu memilih beberapa opsi di bagian Parameter input. Hal ini karena Anda menargetkan sumber daya menggunakan tag atau kelompok sumber daya.

Misalnya, jika Anda memilih `AWS-RestartEC2Instance` runbook, Anda tidak perlu menentukan atau memilih ID instans di bagian Parameter input. Eksekusi otomatisasi menempatkan contoh untuk memulai ulang dengan menggunakan tag atau grup sumber daya yang Anda tentukan.

9. Gunakan opsi di bagian Pengendalian rate untuk membatasi jumlah AWS sumber daya yang dapat menjalankan otomatisasi dalam setiap pasangan Account-wilayah.

   Di bagian Konkurensi, pilih satu opsi:

   - Pilih target untuk memasukkan jumlah absolut target yang dapat menjalankan alur kerja otomatisasi secara bersamaan.
   - Pilih persentase untuk memasukkan persentase dari set target yang dapat menjalankan alur kerja otomatisasi secara bersamaan.

10. Di bagian Ambang kesalahan, pilih satu opsi:

    - Pilih kesalahan untuk memasukkan jumlah kesalahan absolut yang diizinkan sebelum Otomatisasi berhenti mengirim alur kerja ke sumber daya lainnya.
    - Pilih persentase untuk memasukkan persentase kesalahan absolut yang diizinkan sebelum Otomatisasi berhenti mengirim alur kerja ke sumber daya lainnya.

11. (Opsional) Pilih CloudWatch alarm untuk diterapkan ke otomatisasi Anda untuk pemantauan. Untuk memasang CloudWatch alarm ke otomatisasi Anda, prinsip IAM yang memulai otomatisasi harus memiliki izin untuk `iam:createServiceLinkedRole` tindakan tersebut. Untuk informasi selengkapnya tentang CloudWatch alarm, lihat Menggunakan CloudWatch alarm Amazon. Perhatikan bahwa jika alarm Anda aktif, otomatisasi dihentikan. Jika Anda menggunakanAWS CloudTrail, Anda akan melihat panggilan API di jejak Anda.

12. Pilih Eksekusi.

Untuk melihat otomatisasi yang dimulai oleh otomatisasi kontrol tarif Anda, di panel navigasi, pilih Otomatisasi, dan kemudian pilih Tampilkan otomatisasi anak.

Menjalankan otomatisasi dengan target dan kontrol tarif (baris perintah)

Prosedur berikut menjelaskan cara menggunakan AWS CLI (di Linux atau Windows) atau AWS Tools for PowerShell untuk menjalankan otomatisasi dengan target dan kontrol tarif.

Untuk menjalankan otomatisasi dengan target dan kontrol tarif

1. Instal dan konfigurasikan AWS CLI atau AWS Tools for PowerShell, jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI dan Menginstal AWS Tools for PowerShell.

2. Jalankan perintah berikut untuk menandai dokumen.

   Linux & macOS

   ```
   aws ssm list-documents
   ```

   Windows

   ```
   aws ssm list-documents
   ```

   PowerShell

   ```
   Get-SSMDocumentList
   ```

   Perhatikan nama runbook yang ingin Anda gunakan.

3. Jalankan perintah berikut untuk menampilkan detail tentang runbook. Ganti nama *runbook dengan nama* runbook yang detailnya ingin Anda lihat. Juga, perhatikan nama parameter (misalnya,`InstanceId`) yang ingin Anda gunakan untuk `--target-parameter-name` opsi tersebut. Parameter ini menentukan jenis sumber daya di mana otomatisasi berjalan.

   Linux & macOS

   ```
   aws ssm describe-document \
       --name runbook name
   ```

   Windows

   ```
   aws ssm describe-document ^
       --name runbook name
   ```

PowerShell

```
Get-SSMDocumentDescription `
    -Name runbook name
```

4. Buat perintah yang menggunakan target dan opsi kontrol tarif yang ingin Anda jalankan. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   Penargetan menggunakan tag

   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%"
```

Penargetan menggunakan nilai parameter

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%"
```

Penargetan menggunakan 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%"
```

Menargetkan semua instans Amazon EC2 saat ini dan Akun AWS Wilayah AWS

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%"
```

Perintah mengembalikan ID eksekusi. Salin ID ini ke clipboard. Anda dapat menggunakan ID ini untuk melihat status otomatisasi.

Linux & macOS

```
{
```

```
    "AutomationExecutionId": "a4a3c0e9-7efd-462a-8594-01234EXAMPLE"
}
```

Windows

```
{
    "AutomationExecutionId": "a4a3c0e9-7efd-462a-8594-01234EXAMPLE"
}
```

PowerShell

```
a4a3c0e9-7efd-462a-8594-01234EXAMPLE
```

5. Jalankan perintah berikut untuk melihat otomatisasi. Ganti setiap *ID eksekusi otomatisasi* dengan informasi Anda sendiri.

   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. Untuk melihat detail tentang kemajuan otomatisasi, jalankan perintah berikut. Ganti setiap *ID eksekusi otomatisasi* dengan informasi Anda sendiri.

   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
```

Sistem mengembalikan informasi seperti berikut.

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**
>
> Anda juga dapat memantau status otomatisasi di konsol. Di daftar Eksekusi otomatisasi, pilih otomatisasi yang baru Anda jalankan dan kemudian pilih tab Langkah eksekusi. Tab ini menampilkan status tindakan otomatisasi.

Memetakan target untuk otomatisasi

Gunakan `Targets` parameter untuk dengan cepat menentukan sumber daya mana yang ditargetkan oleh otomatisasi. Misalnya, jika Anda ingin menjalankan otomatisasi yang memulai ulang instans terkelola, alih-alih memilih puluhan ID instans di konsol atau mengetiknya secara otomatis dalam perintah, Anda dapat menargetkan instans dengan menentukan Amazon Elastic Compute Cloud (Amazon EC2) tag dengan `Targets` parameter.

Ketika Anda menjalankan otomatisasi yang menggunakan target, AWS Systems Manager menciptakan otomatisasi anak untuk setiap target. Sebagai contoh, jika Anda menargetkan volume Amazon Elastic Block Store (Amazon EBS) dengan menentukan tag, dan tag tersebut dapat digunakan untuk 100 Amazon EBS volume, maka Systems Manager menciptakan 100 anak otomatisasi. Otomatisasi induk selesai ketika semua anak otomatisasi mencapai keadaan akhir.

> **Note**
>
> Semua `input parameters` yang Anda tentukan pada saat bagian runtime (baik di Parameter input konsol atau dengan menggunakan `parameters` pilihan dari baris perintah) yang secara otomatis diproses oleh semua anak otomatisasi.

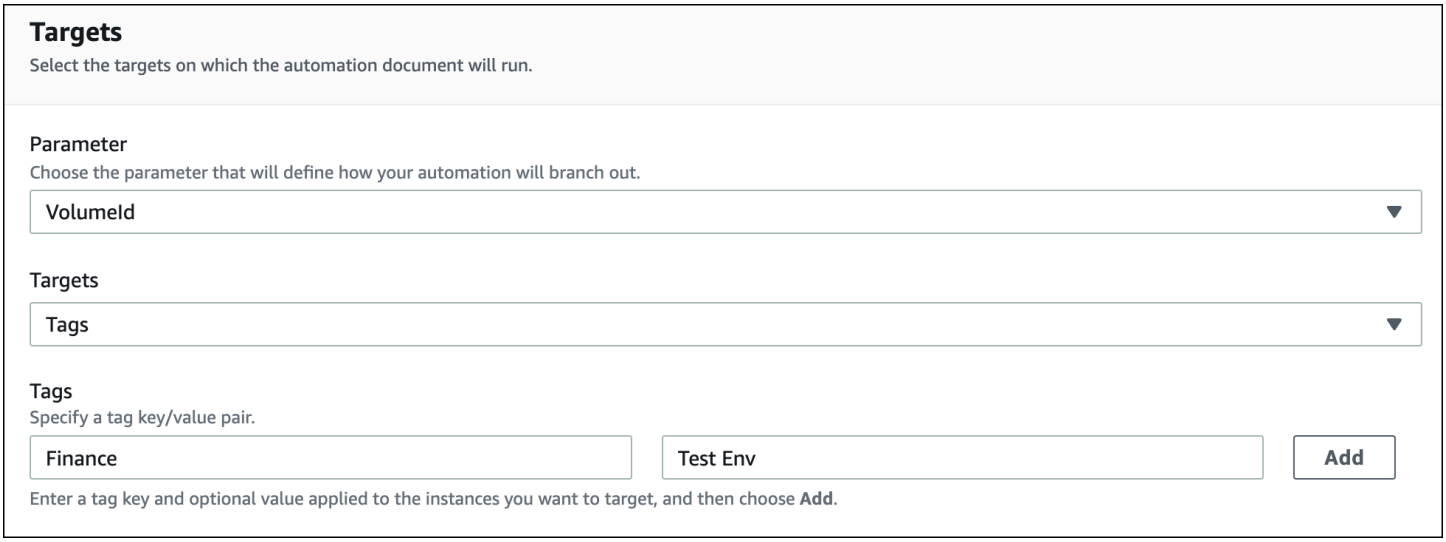
Anda dapat menargetkan sumber daya untuk otomatisasi dengan menggunakan tag, Resource Groups, dan nilai-nilai parameter. Selain itu, Anda dapat menggunakan `TargetMaps` untuk menargetkan beberapa nilai parameter dari baris perintah atau file. Bagian berikut menjelaskan setiap opsi penargetan ini secara lebih detail.

Menargetkan tag

Anda dapat menentukan satu tag sebagai target otomatisasi. Banyak AWS tag dukungan sumber daya, termasuk instans Amazon Elastic Compute Cloud (Amazon EC2) dan Amazon Relational Database Service (Amazon RDS), volume dan snapshot Amazon Elastic Block Store (Amazon EBS), Resource Groups, dan bucket Amazon Simple Storage Service (Amazon S3), untuk beberapa nama. Anda dapat dengan cepat menjalankan otomatisasi pada AWS sumber daya Anda dengan menargetkan tag. Tag adalah pasangan kunci-nilai, seperti Operating_System:Linux atau Department:Finance. Jika Anda menetapkan nama tertentu ke sumber daya, Anda juga dapat menggunakan kata "Nama" sebagai kunci, dan nama sumber daya sebagai nilainya.

Bila Anda menentukan tag sebagai target untuk otomatisasi, Anda juga menentukan parameter target. Parameter target menggunakan `TargetParameterName` pilihan. Dengan memilih parameter target, Anda menentukan jenis sumber daya tempat alur kerja otomatisasi berjalan. Parameter target yang Anda tentukan dengan tag harus merupakan parameter valid yang didefinisikan dalam runbook. Misalnya, jika Anda ingin menargetkan puluhan instans EC2 dengan menggunakan tag, kemudian pilih `InstanceId` parameter target. Dengan memilih parameter ini, anda menentukan instans sebagai jenis sumber daya untuk otomatisasi. Saat membuat runbook kustom, Anda harus menentukan jenis Target `/AWS::EC2::Instance` untuk memastikan hanya instance yang digunakan. Jika tidak, semua sumber daya dengan tag yang sama akan ditargetkan. Saat menargetkan instance dengan tag, instance yang dihentikan mungkin disertakan.

Screenshot berikut menggunaka `AWS-DetachEBSVolume` runbook. Parameter target logis adalah `VolumeId`.



Runbook `AWS-DetachEBSVolume` juga mencakup properti khusus yang disebut Jenis target, yang diatur ke `/AWS::EC2::Volume`. Ini berarti bahwa jika pasangan kunci tag `Finance:TestEnv` mengembalikan berbagai jenis sumber daya (misalnya, instans EC2, volume Amazon EBS, Amazon EBS snapshot) maka hanya volume Amazon EBS yang akan digunakan.

Important

Nama parameter target peka huruf besar dan kecil. Jika Anda menjalankan otomatisasi dengan menggunakan AWS Command Line Interface (AWS CLI) atau AWS Tools for Windows PowerShell, maka Anda harus memasukkan nama parameter target persis seperti yang didefinisikan dalam runbook. Jika tidak, sistem akan mengembalikan sebuah `InvalidAutomationExecutionParametersException` kesalahan. Anda dapat menggunakan operasi DescribeDocumentAPI untuk melihat informasi tentang parameter target yang tersedia di runbook tertentu. Berikut ini adalah contoh AWS CLI perintah yang menyediakan informasi tentang `AWS-DeleteSnapshot` dokumen.

```
aws ssm describe-document \
    --name AWS-DeleteSnapshot
```

Berikut adalah beberapa contoh AWS CLI perintah yang menargetkan sumber daya dengan menggunakan tag.

Contoh 1: Menargetkan tag menggunakan pasangan nilai kunci untuk memulai ulang instans Amazon EC2

Contoh ini memulai ulang semua instans Amazon EC2 yang ditandai dengan kunci Departemen dan nilai. HumanResources Parameter target menggunakan InstanceIdparameter dari runbook. Contoh menggunakan parameter tambahan untuk menjalankan otomatisasi dengan menggunakan peran layanan otomatisasi (juga disebut peran asumsi).

```
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"
```

Contoh 2: Menargetkan tag menggunakan pasangan nilai kunci untuk menghapus snapshot Amazon EBS

Contoh berikut menggunakan `AWS-DeleteSnapshot` runbook untuk menghapus semua snapshot dengan kunci Nama dan nilai Januari2018Backup. Parameter target menggunakan VolumeIdparameter.

```
aws ssm start-automation-execution \
    --document-name AWS-DeleteSnapshot \
    --targets Key=tag:Name,Values=January2018Backups \
    --target-parameter-name VolumeId
```

Penargetan AWS Resource Groups

Anda dapat menentukan satu AWS grup sumber daya sebagai target otomatisasi. Systems Manager menciptakan otomatisasi anak untuk setiap objek dalam Grup Sumber Daya target.

Misalnya, katakan bahwa salah satu Resource Groups Anda diberi nama PatchedAMIs. Kelompok sumber daya ini mencakup daftar 25 Windows Amazon Machine Images (AMIs) yang di-patch secara rutin. Jika Anda menjalankan otomatisasi yang menggunakan `AWS-CreateManagedWindowsInstance` runbook dan menargetkan Resource Groups ini, kemudian Systems Manager menciptakan otomatisasi anak untuk masing-masing 25 AMIs. Ini berarti, bahwa dengan menargetkan PatchedAMIs Resource Group, otomatisasi menciptakan 25 instans dari daftar patch AMIs. Otomatisasi induk selesai ketika semua otomatisasi anak menyelesaikan pemrosesan atau mencapai keadaan akhir.

Perintah berikut ini AWS CLI berlaku untuk contoh Resource Groups PatchAMIs. Perintah mengambil AmiIdparameter untuk `--target-parameter-name` opsi. Perintah tidak termasuk parameter

tambahan yang menentukan jenis instans untuk dibuat dari masing-masing AMI. Default `AWS-CreateManagedWindowsInstance` runbook ke jenis instans t2.medium, jadi perintah ini akan membuat 25 instans Amazon EC2 t2.medium untuk Windows Server.

```
aws ssm start-automation-execution \
    --document-name AWS-CreateManagedWindowsInstance \
    --targets Key=ResourceGroup,Values=PatchedAMIs  \
    --target-parameter-name AmiId
```

Contoh konsol berikut menggunakan Resource Group yang disebut t2-micro-instances.

Menanrgetkan nilai parameter

Anda juga dapat menargetkan nilai parameter. Anda memasukkan `ParameterValues` sebagai kunci dan kemudian memasukkan nilai sumber daya tertentu agar otomatisasi dapat berjalan. Jika Anda menentukan beberapa nilai, Systems Manager menjalankan otomatisasi anak pada setiap nilai yang ditentukan.

Sebagai contoh, katakan bahwa runbook Anda mencakup parameter InstanceID. Jika Anda menargetkan nilai-nilai parameter InstanceID ketika menjalankan otomatisasi, Systems Manager menjalankan otomatisasi anak untuk setiap nilai ID instans yang ditentukan. Otomatisasi induk selesai ketika otomatisasi selesai menjalankan setiap instans tertentu, atau jika otomatisasi gagal. Anda dapat menargetkan maksimum 50 nilai parameter.

Contoh berikut menggunakan `AWS-CreateImage` runbook. Nama parameter target yang ditentukan adalah InstanceId. Kegunaan kuncinya ParameterValues. Nilanya adalah dua ID Instans Amazon EC2. Perintah ini menciptakan otomatisasi untuk setiap instans, yang menghasilkan AMI dari setiap instans.

```
aws ssm start-automation-execution
    --document-name AWS-CreateImage \
    --target-parameter-name InstanceId \
    --targets Key=ParameterValues,Values=i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE
```

> **Note**
>
> `AutomationAssumeRole` bukan parameter yang valid. Jangan memilih item ini ketika menjalankan otomatisasi yang menargetkan nilai parameter.

Menargetkan peta nilai parameter

Pilihan `TargetMaps` memperluas kemampuan Anda untuk menargetkan `ParameterValues`. Anda dapat memasukkan array nilai parameter dengan menggunakan `TargetMaps` di baris perintah. Anda dapat menentukan maksimum 50 nilai parameter pada baris perintah. Jika Anda ingin menjalankan perintah yang menentukan lebih dari 50 nilai parameter, maka Anda dapat memasukkan nilai-nilai dalam file JSON. Anda kemudian dapat memanggil file dari baris perintah.

> **Note**
>
> Pilihan `TargetMaps` tidak didukung di konsol.

Gunakan format berikut untuk menentukan beberapa nilai parameter dengan menggunakan `TargetMaps` pilihan dalam perintah. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri.

```
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”
```

Jika Anda ingin memasukkan lebih dari 50 nilai parameter untuk `TargetMaps` pilihan, tentukan nilai dalam file menggunakan format JSON berikut. Menggunakan file JSON juga meningkatkan pembacaan ketika memberikan beberapa nilai parameter.

```
[
```

```
    {“parameter”: "value", “parameter 2”: "value", “parameter 3”: "value"},

    {“parameter 4”: "value", “parameter 5”: "value", "parameter 6": "value"}

]
```

Simpan file dengan ekstensi file .json. Anda dapat membuat daftar file dengan menggunakan perintah berikut. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

```
aws ssm start-automation-execution \
    --document-name runbook name \
    --parameters input parameters \
    --target-maps path to file/file name.json
```

Anda juga dapat mengunduh file dari bucket Amazon Simple Storage Service (Amazon S3), selama Anda memiliki izin untuk membaca data dari bucket. Gunakan format perintah berikut. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

```
aws ssm start-automation-execution \
    --document-name runbook name \
    --target-maps http://DOC-EXAMPLE-BUCKET.s3.amazonaws.com/file_name.json
```

Berikut adalah contoh skenario untuk membantu Anda memahami `TargetMaps` pilihan. Dalam skenario ini, pengguna ingin membuat instans Amazon EC2 dari berbagai jenis dari yang berbeda AMIs. Untuk melakukan tugas ini, pengguna menciptakan runbook bernama AMI_testing. Runbook ini mendefinisikan dua parameter input: `instanceType` dan `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"
  ]
}
```

Pengguna kemudian menentukan nilai parameter target berikut dalam sebuah file bernama `AMI_instance_types.json`.

```
[
  {
    "instanceType" : ["t2.micro"],
    "imageId" : ["ami-b70554c8"]
  },
  {
    "instanceType" : ["t2.small"],
    "imageId" : ["ami-b70554c8"]
  },
  {
    "instanceType" : ["t2.medium"],
    "imageId" : ["ami-cfe4b2b0"]
  },
  {
```

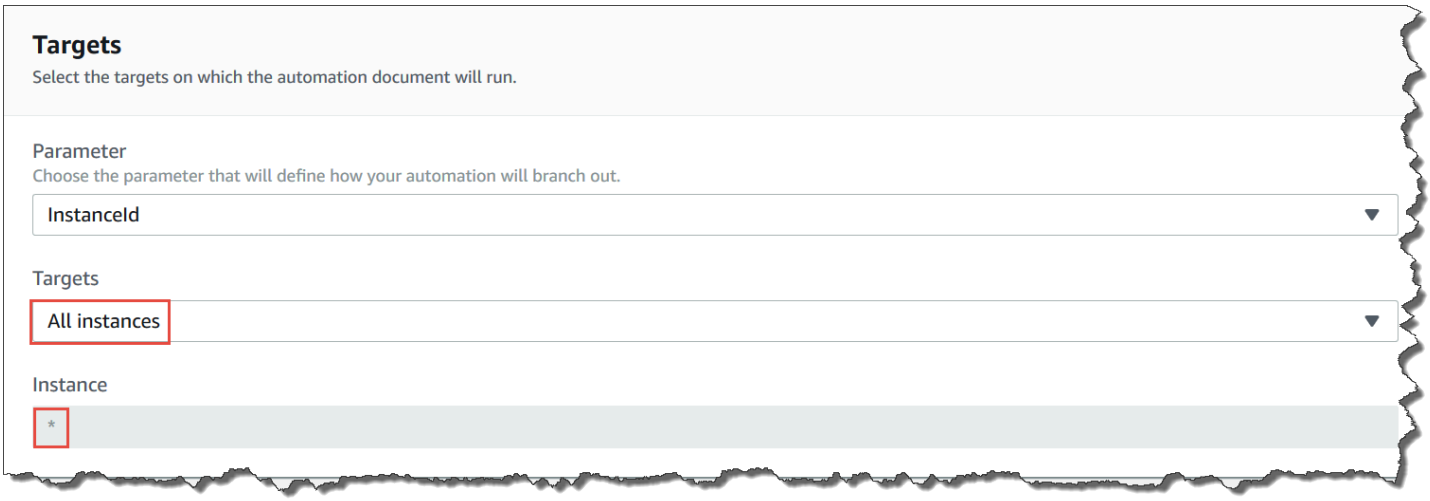
```
    "instanceType" : ["t2.medium"],
    "imageId" : ["ami-cfe4b2b0"]
  },
  {
    "instanceType" : ["t2.medium"],
    "imageId" : ["ami-cfe4b2b0"]
  }
]
```

Pengguna dapat menjalankan otomatisasi dan membuat lima instans EC2 yang didefinisikan dalam `AMI_instance_types.json` dengan menjalankan perintah berikut.

```
aws ssm start-automation-execution \
    --document-name AMI_Testing \
    --target-parameter-name imageId \
    --target-maps file:///home/TestUser/workspace/runinstances/AMI_instance_types.json
```

Menargetkan semua instans Amazon EC2

Anda dapat menjalankan otomatisasi pada semua instans Amazon EC2 saat ini Akun AWS dan Wilayah AWS dengan memilih Semua instans dalam daftar Target. Misalnya, jika Anda ingin memulai ulang semua instans Amazon EC2 Anda Akun AWS dan saat iniWilayah AWS, Anda dapat memilih **`AWS-RestartEC2Instance`** runbook lalu memilih Semua instans dari daftar Target.



Setelah Anda memilih Semua instans, Systems Manager mengisi Instans dengan tanda bintang (*) dan membuat bidang tidak dapat diubah (bidang berwarna abu-abu). Systems Manager juga membuat InstanceIdbidang di bidang parameter Input tidak tersedia untuk perubahan. Membuat

bidang ini tidak dapat diubah adalah perilaku yang diharapkan jika Anda memilih untuk menargetkan semua instans.

Kontrol otomatisasi pada skala

Selain itu, Anda dapat mengontrol deployment otomatisasi di seluruh armada AWS sumber daya dengan menentukan nilai konkurensi dan ambang batas kesalahan. Fitur konkurensi dan fitur ambang batas kesalahan secara kolektif disebut pengendalian rate.

Bersamaan

Gunakan Konkurensi untuk menentukan berapa banyak sumber daya yang diizinkan untuk menjalankan otomatisasi secara bersamaan. Konkurensi membantu untuk membatasi dampak atau downtime pada sumber daya Anda saat memproses otomatisasi. Anda dapat menentukan jumlah sumber daya absolut, misalnya 20, atau persentase target yang ditetapkan, misalnya 10%.

Sistem antrean memberikan otomatisasi ke sumber daya tunggal dan menunggu sampai penanganan awal selesai sebelum mengirim otomatisasi ke dua sumber daya lagi. Sistem secara eksponensial mengirimkan otomatisasi ke lebih banyak sumber daya sampai nilai konkurensi terpenuhi.

Ambang batas kesalahan

Ambang kesalahan menentukan berapa banyak otomatisasi yang diperbolehkan untuk gagal sebelum AWS Systems Manager berhenti mengirim otomatisasi ke sumber daya lainnya. Anda dapat menentukan jumlah sumber daya absolut, misalnya 10, atau persentase target yang ditetapkan, misalnya 10%.

Jika Anda menentukan jumlah absolut dari 3 kesalahan, misalnya, sistem berhenti menjalankan otomatisasi saat kesalahan keempat diterima. Jika Anda menentukan 0, maka sistem berhenti menjalankan otomatisasi pada target tambahan setelah hasil kesalahan pertama dikembalikan.

Jika Anda mengirim otomatisasi ke, misalnya, 50 instans dan mengatur ambang batas kesalahan 10%, sistem berhenti mengirim perintah ke instans tambahan ketika kesalahan kelima diterima. Pemanggilan yang sudah menjalankan otomatisasi ketika ambang batas kesalahan tercapai bisa diselesaikan, tetapi beberapa otomatisasi ini mungkin juga akan gagal. Jika Anda ingin memastikan bahwa tidak akan ada lebih banyak kesalahan daripada nomor yang ditentukan untuk ambang batas kesalahan, atur nilai Konkurensi ke 1 sehingga otomatisasi melanjutkan satu per satu.

Menjalankan otomatisasi dalam beberapa Wilayah AWS dan akun

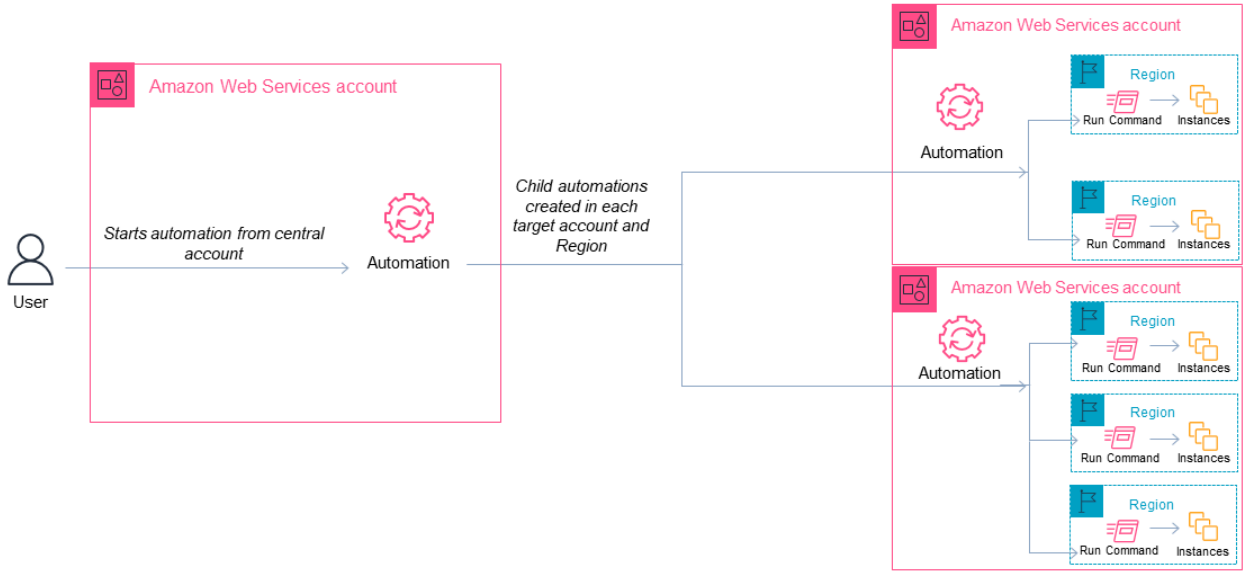
Anda dapat menjalankan AWS Systems Manager otomatisasi di beberapa unit Wilayah AWS dan Akun AWS atau AWS Organizations organisasi (OU) dari akun pusat. Otomatisasi adalah kemampuan AWS Systems Manager. Menjalankan otomatisasi di beberapa Wilayah dan akun atau OU untuk mengurangi waktu yang diperlukan guna mengelola AWS sumber daya dan meningkatkan keamanan.

Misalnya, Anda dapat melakukan hal berikut dengan menggunakan runbook otomatisasi:

- Menerapkan patching dan pembaruan keamanan secara terpusat.
- Memulihkan penyimpangan kepatuhan pada konfigurasi VPC atau kebijakan bucket Amazon S3.
- Kelola sumber daya, seperti instans Amazon Elastic Compute Cloud (Amazon EC2) EC2, dalam skala besar.

Diagram berikut menunjukkan contoh pengguna yang menjalankan `AWS-RestartEC2Instances` runbook di beberapa Wilayah dan akun dari akun pusat. Otomatisasi menempatkan instance dengan menggunakan tag yang ditentukan di Wilayah dan akun yang ditargetkan.

> **Note**
>
> Ketika Anda menjalankan otomatisasi di beberapa wilayah dan akun, Anda menargetkan sumber daya dengan menggunakan tag atau nama AWS grup sumber daya. Grup sumber daya harus ada di setiap akun target dan Wilayah. Nama grup sumber daya harus sama di setiap akun target dan Wilayah. Otomatisasi gagal dijalankan pada sumber daya yang tidak memiliki tag tertentu atau yang tidak termasuk dalam grup sumber daya tertentu.



Pilih akun pusat untuk Otomasi

Jika Anda ingin menjalankan otomatisasi di seluruh OU, akun pusat harus memiliki izin untuk mencantumkan semua akun di OU. Ini hanya dimungkinkan dari akun administrator yang didelegasikan, atau akun manajemen organisasi. Kami menyarankan Anda mengikuti praktik AWS Organizations terbaik dan menggunakan akun administrator yang didelegasikan. Untuk informasi selengkapnya tentang praktik AWS Organizations terbaik, lihat Praktik terbaik untuk akun manajemen di Panduan AWS Organizations Pengguna. Untuk membuat akun administrator yang didelegasikan untuk Systems Manager, Anda dapat menggunakan `register-delegated-administrator` perintah dengan AWS CLI seperti yang ditunjukkan pada contoh berikut.

```
aws organizations register-delegated-administrator \
    --account-id delegated admin account ID \
    --service-principal ssm.amazonaws.com
```

Jika Anda ingin menjalankan otomatisasi di beberapa akun yang tidak dikelola olehAWS Organizations, sebaiknya buat akun khusus untuk manajemen otomatisasi. Menjalankan semua otomatisasi lintas akun dari akun khusus menyederhanakan manajemen izin IAM, upaya pemecahan masalah, dan menciptakan lapisan pemisahan antara operasi dan administrasi. Pendekatan ini

juga disarankan jika Anda menggunakanAWS Organizations, tetapi hanya ingin menargetkan akun individu dan bukan OU.

Cara kerja menjalankan otomatisasi

Menjalankan otomatisasi di beberapa Wilayah dan akun atau OU bekerja sebagai berikut:

1. Verifikasi bahwa semua sumber daya di mana Anda ingin menjalankan otomatisasi, di semua wilayah dan akun atau OU, menggunakan tag identik. Jika tidak, Anda dapat menambahkannya ke AWS grup sumber daya dan target kelompok tersebut. Untuk informasi selengkapnya, lihat Apa itu grup sumber daya? di Panduan Pengguna AWS Resource Groups dan Tag.
2. Masuk ke akun yang ingin Anda konfigurasikan sebagai akun pusat Otomasi.
3. Gunakan Menyiapkan izin akun manajemen untuk otomatisasi multi-wilayah dan multi-akun prosedur dalam topik ini untuk membuat peran IAM berikut:
   - **`AWS-SystemsManager-AutomationAdministrationRole`**- Peran ini memberikan izin pengguna untuk menjalankan otomatisasi di beberapa akun dan OU.
   - **`AWS-SystemsManager-AutomationExecutionRole`**- Peran ini memberikan izin pengguna untuk menjalankan otomatisasi di akun yang ditargetkan.
4. Pilih runbook, Wilayah, dan akun atau OU tempat Anda ingin menjalankan otomatisasi.

> **Note**
>
> Otomatisasi tidak berjalan secara rekursif melalui OU. Pastikan target OU berisi akun yang diinginkan. Jika Anda memilih runbook kustom, runbook harus dibagikan dengan semua akun target. Untuk informasi tentang berbagi runbook, lihatMembagikan dokumen SSM. Untuk informasi tentang menggunakan runbook bersama, lihat Menggunakan dokumen SSM bersama.

5. Jalankan otomatisasi.

> **Note**
>
> Saat menjalankan otomatisasi di beberapa wilayah, akun, atau OU, otomatisasi yang Anda jalankan dari akun utama memulai otomatisasi anak di setiap akun target. Otomatisasi di akun utama berisi `aws:executeAutomation` langkah-langkah untuk setiap akun target. Jika Anda memulai otomatisasi dari Wilayah baru yang diluncurkan setelah 20 Maret 2019, dan menargetkan Wilayah yang diaktifkan secara default, otomatisasi gagal. Jika Anda

memulai otomatisasi dari Wilayah yang diaktifkan secara default dan menargetkan Wilayah yang telah Anda aktifkan, otomatisasi berjalan dengan sukses.

6. Gunakan GetAutomationExecution, DescribeAutomationStepExecutions, dan operasi DescribeAutomationExecutionsAPI dari AWS Systems Manager konsol atau AWS CLI untuk memantau kemajuan otomatisasi. Output langkah-langkah untuk otomatisasi di akun utama Anda akan menjadi `AutomationExecutionId` dari otomatisasi anak. Untuk melihat output otomatisasi anak yang dibuat di akun target Anda, pastikan untuk menentukan akun yang sesuai, Wilayah, dan `AutomationExecutionId` dalam permintaan Anda.

Menyiapkan izin akun manajemen untuk otomatisasi multi-wilayah dan multi-akun

Gunakan prosedur berikut untuk membuat IAM role yang diperlukan untuk otomatisasi mulati Wilayah dan multi akun Otomatisasi Systems Manager dengan menggunakan AWS CloudFormation. Prosedur ini menjelaskan cara membuat **`AWS-SystemsManager-AutomationAdministrationRole`** peran. Anda hanya perlu membuat peran ini di akun pusat Otomasi. Prosedur ini juga menjelaskan cara membuat **`AWS-SystemsManager-AutomationExecutionRole`** peran. Anda harus membuat peran ini dalam setiap akun yang ingin Anda targetkan untuk menjalankan otomatisasi multi-wilayah dan multi-akun. Sebaiknya gunakan AWS CloudFormation StackSets untuk membuat **`AWS-SystemsManager-AutomationExecutionRole`** peran di akun yang ingin Anda targetkan untuk menjalankan otomatisasi Multi-wilayah dan multi-akun.

Untuk membuat peran administrasi IAM yang diperlukan untuk otomatisasi Multi-wilayah dan multi-akun dengan menggunakan AWS CloudFormation

1. Unduh dan unzip file. `AWS-SystemsManager-AutomationAdministrationRole.zip` Atau, jika akun Anda dikelola oleh AWS Organizations `AWS-SystemsManager-AutomationAdministrationRole (org).zip`. File ini mencakup `AWS-SystemsManager-AutomationAdministrationRole.yaml` AWS CloudFormation file templat.
2. Buka AWS CloudFormation konsol di https://console.aws.amazon.com/cloudformation.
3. Pilih Buat tumpukan.
4. Di bagian Tentukan templat, pilih Unggah templat.
5. Pilih Pilih file, lalu pilih file `AWS-SystemsManager-AutomationAdministrationRole.yaml` AWS CloudFormation template.
6. Pilih Berikutnya.

7. Di halaman Tentukan detail tumpukan, di bidang Nama tumpukan, masukkan nama.
8. Pilih Selanjutnya.
9. Pada halaman Configure stack options, masukkan nilai untuk opsi apa pun yang ingin Anda gunakan. Pilih Selanjutnya.
10. Pada halaman Ulasan, gulir ke bawah dan pilih opsi Saya akui yang AWS CloudFormation mungkin membuat sumber daya IAM dengan nama khusus.
11. Pilih Buat tumpukan.

AWS CloudFormation menunjukkan status CREATE_IN_PROGRESS selama sekitar tiga menit. Perubahan status menjadi CREATE_COMPLETE.

Anda harus mengulangi prosedur berikut di setiap akun yang ingin Anda targetkan untuk menjalankan otomatisasi Multi-wilayah dan multi-akun.

Untuk membuat peran otomatisasi IAM yang diperlukan untuk otomatisasi Multi-wilayah dan multi-akun dengan menggunakan AWS CloudFormation

1. Unduh `AWS-SystemsManager-AutomationExecutionRole.zip`. Atau, jika akun Anda dikelola oleh AWS Organizations `AWS-SystemsManager-AutomationExecutionRole(org).zip`. File ini mencakup `AWS-SystemsManager-AutomationExecutionRole.yaml` AWS CloudFormation file templat.
2. Buka AWS CloudFormation konsol di https://console.aws.amazon.com/cloudformation.
3. Pilih Buat tumpukan.
4. Di bagian Tentukan templat, pilih Unggah templat.
5. Pilih Pilih file, lalu pilih file `AWS-SystemsManager-AutomationExecutionRole.yaml` AWS CloudFormation template.
6. Pilih Berikutnya.
7. Di halaman Tentukan detail tumpukan, di bidang Nama tumpukan, masukkan nama.
8. Di bagian Parameter, di AdminAccountIdlapangan, masukkan ID untuk akun pusat otomatisasi.
9. Jika Anda menyiapkan peran ini untuk AWS Organizations lingkungan, ada bidang lain di bagian yang disebut organizationId. Masukkan ID AWS organisasi Anda.
10. Pilih Selanjutnya.
11. Pada halaman Configure stack options, masukkan nilai untuk opsi apa pun yang ingin Anda gunakan. Pilih Selanjutnya.

12. Pada halaman Ulasan, gulir ke bawah dan pilih opsi Saya akui yang AWS CloudFormation mungkin membuat sumber daya IAM dengan nama khusus.
13. Pilih Buat tumpukan.

AWS CloudFormation menunjukkan status CREATE_IN_PROGRESS selama sekitar tiga menit. Perubahan status menjadi CREATE_COMPLETE.

Jalankan otomatisasi di beberapa Wilayah dan akun (konsol)

Prosedur berikut menjelaskan cara menggunakan konsol Systems Manager untuk menjalankan otomatisasi di beberapa Wilayah dan akun dari akun manajemen otomatisasi.

Sebelum Anda memulai

Sebelum Anda menyelesaikan prosedur berikut, perhatikan informasi berikut:

- Pengguna atau peran yang Anda gunakan untuk menjalankan otomatisasi Multi-wilayah atau multi-akun harus memiliki `iam:PassRole` izin untuk peran tersebut`AWS-SystemsManager-AutomationAdministrationRole`.
- Akun AWS ID atau OU tempat Anda ingin menjalankan otomatisasi.
- Wilayah yang didukung oleh Systems Manager tempat Anda ingin menjalankan otomatisasi.
- Kunci tag dan nilai tag, atau nama grup sumber daya, di mana Anda ingin menjalankan otomatisasi.

Menjalankan otomatisasi di beberapa Wilayah dan akun

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Otomatisasi, lalu pilih Eksekusi otomatisasi.
3. Di daftar Dokumen otomatisasi, pilih runbook. Pilih satu opsi atau lebih di panel Kategori dokumen untuk memfilter dokumen SSM sesuai dengan tujuannya. Untuk melihat runbook yang Anda miliki, pilih tab Dimiliki oleh saya. Untuk melihat runbook yang dibagikan dengan akun Anda, pilih tab Dibagikan dengan saya. Untuk melihat semua runbook, pilih tab Semua dokumen.

> **Note**
>
> Anda dapat melihat informasi tentang runbook dengan memilih nama runbook.

4. Di bagian Detail dokumen, verifikasi bahwa Versi dokumen diatur ke versi yang ingin Anda jalankan. Sistem ini termasuk pilihan versi berikut:

   - Versi default saat runtime — Pilih opsi ini jika runbook Otomasi diperbarui secara berkala dan versi default baru ditetapkan.
   - Versi terbaru saat runtime — Pilih opsi ini jika runbook Otomasi diperbarui secara berkala, dan Anda ingin menjalankan versi yang terbaru diperbarui.
   - 1 (Default) - Pilih opsi ini untuk menjalankan versi pertama dokumen, yang merupakan default.

5. Pilih Berikutnya.
6. Pada halaman Eksekusi dokumen otomatisasi, pilih Multi-Akun dan Wilayah.
7. Di bagian Target akun dan Wilayah, gunakan bidang Akun dan organisasi (OU) untuk menentukan unit organisasi (OU) yang berbeda Akun AWS atau AWS tempat Anda ingin menjalankan otomatisasi. Pisahkan beberapa akun atau OU dengan koma.
8. Gunakan daftar Wilayah AWS untuk memilih satu Wilayah atau lebih untuk menjalankan otomatisasi.
9. Gunakan pilihan Kontrol tarif multi Wilayah dan akun untuk membatasi otomatisasi ke sejumlah akun yang berjalan di sejumlah Wilayah. Pilihan ini tidak membatasi jumlah AWS sumber daya yang dapat menjalankan otomatisasi.

   a. Di bagian Lokasi (pasangan wilayah akun) konkurensi, pilih opsi untuk membatasi jumlah otomatisasi yang dapat berjalan di beberapa akun dan Wilayah secara bersamaan. Misalnya, jika Anda memilih untuk menjalankan otomatisasi sebanyak lima (5) Akun AWS, yang terletak di empat (4) Wilayah AWS, maka Systems Manager menjalankan otomatisasi di 20 pasangan wilayah akun. Anda dapat menggunakan opsi ini untuk menentukan jumlah absolut, seperti **2**, sehingga otomatisasi hanya berjalan dalam dua pasangan akun-wilayah pada waktu yang sama. Atau Anda dapat menentukan persentase pasangan Wilayah-akun yang dapat berjalan pada waktu yang sama. Misalnya, dengan 20 pasangan wilayah akun, jika Anda menentukan 20%, maka otomatisasi secara bersamaan menjalankan maksimal lima (5) pasangan wilayah akun.

      - Pilih target untuk memasukkan jumlah target absolut pasangan akun-Wilayah yang dapat menjalankan otomatisasi secara bersamaan.
      - Pilih persen untuk memasukkan persentase dari jumlah total pasangan wilayah yang dapat menjalankan otomatisasi secara bersamaan.

   b. Di bagian Ambang kesalahan, pilih satu opsi:

- Pilih kesalahan untuk memasukkan jumlah kesalahan absolut yang diizinkan sebelum Otomatisasi berhenti mengirim otomatisasi ke sumber daya lainnya.
- Pilih persen untuk memasukkan persentase kesalahan yang diizinkan sebelum Otomatisasi berhenti mengirim otomatisasi ke sumber daya lainnya.

10. Di bagian Target, pilih bagaimana Anda ingin menargetkan AWS sumber daya tempat Anda ingin menjalankan otomatisasi. Pilihan ini diperlukan.

    a. Gunakan daftar Parameter untuk memilih parameter. Item dalam daftar Parameter ditentukan oleh parameter di Runbook otomatisasi yang Anda pilih pada awal prosedur ini. Dengan memilih parameter, Anda menentukan jenis sumber daya di mana alur kerja otomatisasi berjalan.

    b. Gunakan daftar Target untuk memilih cara Anda menargetkan sumber daya.

        i. Jika Anda ingin menargetkan sumber daya dengan menggunakan nilai parameter, masukkan nilai parameter untuk parameter yang Anda pilih di bagian Parameter input.

        ii. Jika Anda ingin menargetkan sumber daya dengan menggunakan AWS Resource Groups, lalu pilih nama grup dari daftar Grup Sumber Daya.

        iii. Jika Anda memilih untuk menargetkan sumber daya dengan menggunakan tag, masukkan kunci tag dan nilai tag dalam bidang yang disediakan (opsional). Pilih Tambahkan.

        iv. Jika Anda ingin menjalankan Runbook otomatisasi pada semua contoh di saat ini Akun AWS dan Wilayah AWS, pilih Semua instans.

11. Di bagian Parameter input, tentukan input yang diperlukan. Pilih peran layanan `AWS-SystemsManager-AutomationAdministrationRole` IAM dari AutomationAssumeRoledaftar.

> **Note**
>
> Anda mungkin tidak perlu memilih beberapa opsi di bagian Parameter input. Hal ini karena Anda menargetkan sumber daya di beberapa wilayah dan akun dengan menggunakan tag atau grup sumber daya. Misalnya, jika Anda memilih `AWS-RestartEC2Instance` runbook, Anda tidak perlu menentukan atau memilih ID instans di bagian Parameter input. Otomatisasi menempatkan instans untuk memulai ulang dengan menggunakan tag yang Anda tentukan.

12. (Opsional) Pilih CloudWatch alarm untuk diterapkan ke otomatisasi Anda untuk pemantauan. Untuk memasang CloudWatch alarm ke otomatisasi Anda, prinsip IAM yang memulai otomatisasi harus memiliki izin untuk `iam:createServiceLinkedRole` tindakan tersebut. Untuk informasi selengkapnya tentang CloudWatch alarm, lihat Menggunakan CloudWatch alarm Amazon. Perhatikan bahwa jika alarm Anda aktif, otomatisasi dibatalkan dan `0nCancel` langkah apa pun yang telah Anda tetapkan dijalankan. Jika Anda menggunakanAWS CloudTrail, Anda akan melihat panggilan API di jejak Anda.
13. Gunakan opsi di bagian Pengendalian rate untuk membatasi jumlah AWS sumber daya yang dapat menjalankan otomatisasi dalam setiap pasangan Account-wilayah.

    Di bagian Konkurensi, pilih satu opsi:

    - Pilih target untuk memasukkan jumlah absolut target yang dapat menjalankan alur kerja otomatisasi secara bersamaan.
    - Pilih persentase untuk memasukkan persentase dari set target yang dapat menjalankan alur kerja otomatisasi secara bersamaan.
14. Di bagian Ambang kesalahan, pilih satu opsi:

    - Pilih kesalahan untuk memasukkan jumlah kesalahan absolut yang diizinkan sebelum Otomatisasi berhenti mengirim alur kerja ke sumber daya lainnya.
    - Pilih persentase untuk memasukkan persentase kesalahan absolut yang diizinkan sebelum Otomatisasi berhenti mengirim alur kerja ke sumber daya lainnya.
15. Pilih Eksekusi.

Jalankan otomatisasi di beberapa wilayah dan akun (baris perintah)

Prosedur berikut menjelaskan cara menggunakan AWS CLI (di Linux atau Windows) atau AWS Tools for PowerShell untuk menjalankan otomatisasi di beberapa Wilayah dan akun dari akun manajemen Otomatisasi.

Sebelum Anda memulai

Sebelum Anda menyelesaikan prosedur berikut, perhatikan informasi berikut:

- Akun AWS ID atau OU tempat Anda ingin menjalankan otomatisasi.
- Wilayah yang didukung oleh Systems Manager tempat Anda ingin menjalankan otomatisasi.

- Kunci tag dan nilai tag, atau nama grup sumber daya, di mana Anda ingin menjalankan otomatisasi.

Menjalankan otomatisasi di beberapa Wilayah dan akun

1. Instal dan konfigurasikan AWS CLI atau AWS Tools for PowerShell, jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI dan Menginstal AWS Tools for PowerShell.

2. Gunakan format berikut untuk membuat perintah untuk menjalankan otomatisasi di beberapa wilayah dan akun. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   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" }
```

Berikut adalah beberapa contoh tanda.

Contoh 1: Contoh ini memulai ulang instans EC2 di akun `123456789012` dan `987654321098`, yang terletak di Wilayah `us-east-2` dan `us-west-1`. Instans harus ditandai dengan nilai pasangan kunci `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" }
```

Contoh 2: Contoh ini memulai ulang instans EC2 di akun `123456789012` dan `987654321098`, yang terletak di `eu-central-1` Wilayah. Instans harus menjadi anggota `prod-instances` AWS grup sumber daya.

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" }
```

Contoh 3: Contoh ini memulai ulang instans EC2 di `ou-1a2b3c-4d5e6c` AWS unit organisasi (OU). Instans terletak di `us-west-1` dan `us-west-2` Wilayah. Instans harus menjadi anggota `WebServices` AWS grup sumber daya.

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" }
```

Sistem mengembalikan informasi seperti berikut ini.

Linux & macOS

```
{
        "AutomationExecutionId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
    }
```

Windows

```
{
        "AutomationExecutionId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
    }
```

PowerShell

```
4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE
```

3. Jalankan perintah berikut untuk menampilkan detail otomatisasi. Ganti *ID eksekusi otomatisasi* dengan informasi Anda sendiri.

   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. Jalankan perintah berikut untuk melihat detail tentang kemajuan otomatisasi.

   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**
>
> Anda juga dapat memantau status otomatisasi di konsol. Di daftar Eksekusi otomatisasi, pilih otomatisasi yang baru Anda jalankan dan kemudian pilih tab Langkah eksekusi. Tab ini menampilkan status tindakan otomatisasi.

Info lebih lanjut

Penambalan multi-akun dan Multi-wilayah terpusat dengan Otomasi AWS Systems Manager

Jalankan otomatisasi berdasarkan peristiwa

Anda dapat memulai otomatisasi dengan menentukan runbook sebagai target acara Amazon EventBridge. Anda dapat memulai otomatisasi sesuai dengan jadwal, atau saat peristiwa AWS sistem tertentu terjadi. Misalnya, katakanlah Anda membuat runbook bernama BootStrapInstancesyang menginstal perangkat lunak pada sebuah instance ketika sebuah instance dimulai. Untuk menentukan BootStrapInstancesrunbook (dan otomatisasi terkait) sebagai target EventBridge acara, pertama-tama Anda membuat EventBridge aturan baru. (Berikut ini adalah contoh aturan: Nama layanan: EC2, Jenis Peristiwa: Notifikasi Perubahan status Instans EC2, Keadaan spesifik: berjalan, Instans apa pun.) Kemudian Anda menggunakan prosedur berikut untuk menentukan BootStrapInstancesrunbook sebagai target acara menggunakan EventBridge konsol dan AWS Command Line Interface (AWS CLI). Ketika instans baru dimulai, sistem menjalankan otomatisasi dan menginstal perangkat lunak.

Untuk informasi tentang membuat peran, lihat Membuat runbook Anda sendiri.

Membuat EventBridge acara yang menggunakan runbook (konsol)

Gunakan prosedur berikut untuk mengonfigurasi runbook sebagai target EventBridge acara.

Untuk mengonfigurasi runbook sebagai target aturan EventBridge acara

1. Buka EventBridge konsol Amazon di https://console.aws.amazon.com/events/.
2. Di panel navigasi, pilih Aturan.
3. Pilih Buat aturan.
4. Masukkan nama dan deskripsi untuk aturan.

   Aturan tidak boleh memiliki nama yang sama dengan aturan lain di Wilayah yang sama dan di bus kejadian yang sama.
5. Untuk bus acara, pilih bus acara yang ingin Anda kaitkan dengan aturan ini. Jika Anda ingin aturan ini merespons peristiwa pencocokan yang berasal dari Anda sendiri Akun AWS, pilih default. Ketika Layanan AWS di akun Anda memancarkan acara, itu selalu masuk ke bus acara default akun Anda.
6. Pilih bagaimana aturan dipicu.

Untuk membuat aturan berdasarkan...	Lakukan ini...
Peristiwa	a. Untuk Tipe aturan, pilih Aturan dengan pola peristiwa. b. Pilih Selanjutnya. c. Untuk sumber Acara, pilih AWS acara atau acara EventBridge mitra. d. Di bagian Pola acara, lakukan salah satu hal berikut: • Untuk menggunakan templat untuk membuat pola acara Anda, pilih Formulir pola acara dan pilih Sumber acara, AWS layanan, dan jenis Acara. Jika Anda

Untuk membuat aturan berdasarkan...	Lakukan ini...	
	memilih Semua Acara sebagai jenis acara, semua peristiwa yang dipancarkan oleh Layanan AWS akan cocok dengan aturan. Untuk menyesuaikan template, pilih Pola kustom (editor JSON) dan buat perubahan Anda. • Untuk menggunakan pola acara khusus, pilih Pola kustom (editor JSON) dan buat pola acara Anda.	

Untuk membuat aturan berdasarkan...	Lakukan ini...	
Jadwal	a. Untuk jenis Aturan, pilih Jadwal. b. Pilih Berikutnya. c. Untuk pola Jadwal, lakukan salah satu hal berikut: • Untuk menggunakan ekspresi cron untuk menentukan jadwal, pilih Jadwal berbutir halus yang berjalan pada waktu tertentu, seperti pukul 8:00 pagi. PST pada hari Senin pertama setiap bulan dan masukkan ekspresi cron. • Untuk menggunakan ekspresi laju untuk menentukan jadwal, pilih Jadwal yang berjalan pada tingkat reguler, seperti setiap 10 menit dan masukkan ekspresi laju.	

7. Pilih Berikutnya.
8. Untuk Jenis Target, pilih Layanan AWS .
9. Untuk Pilih target, pilih Systems Manager Automation.
10. Untuk Dokumen, pilih runbook untuk digunakan ketika target anda dijalankan.
11. Di bagian Konfigurasi parameter otomatisasi, pertahankan nilai parameter default (jika tersedia) atau masukkan nilai Anda sendiri.

> **Note**
>
> Untuk membuat target, Anda harus menentukan nilai untuk setiap parameter yang diperlukan. Jika tidak, sistem membuat aturan, tetapi aturan tidak akan berjalan.

12. Untuk banyak jenis target, EventBridge perlu izin untuk mengirim acara ke target. Dalam kasus ini, EventBridge dapat membuat peran IAM yang diperlukan agar aturan Anda berjalan. Lakukan salah satu dari langkah berikut ini:

    - Untuk membuat IAM role secara otomatis, pilih Buat peran baru untuk sumber daya khusus ini.
    - Untuk menggunakan peran IAM yang Anda buat sebelumnya, pilih Gunakan peran yang ada dan pilih peran yang ada dari menu tarik-turun. Perhatikan bahwa Anda mungkin perlu memperbarui kebijakan kepercayaan untuk peran IAM Anda untuk disertakan EventBridge. Berikut ini adalah contohnya:

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

13. Pilih Berikutnya.
14. (Opsional) Masukkan satu atau lebih tanda untuk aturan. Untuk informasi selengkapnya, lihat Menandai EventBridge Sumber Daya Amazon Anda di Panduan EventBridge Pengguna Amazon.
15. Pilih Berikutnya.
16. Tinjau detail aturan dan pilih Buat aturan.

Buat EventBridge acara yang menggunakan runbook (baris perintah)

Prosedur berikut menjelaskan cara menggunakan AWS CLI (di Linux atau Windows) atau AWS Tools for PowerShell untuk membuat aturan EventBridge acara dan mengkonfigurasi runbook sebagai target.

Untuk mengonfigurasi runbook sebagai target aturan EventBridge acara

1. Instal dan konfigurasikan AWS CLI atau AWS Tools for PowerShell, jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI dan Menginstal AWS Tools for PowerShell.

2. Buat perintah untuk menentukan aturan EventBridge acara baru. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   Pemicu berdasarkan jadwal

   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"
   ```

   Contoh berikut membuat aturan EventBridge acara yang dimulai setiap hari pada pukul 9:00 pagi (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 * * ? *)"
```

Pemicu berdasarkan suatu peristiwa

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"]}'
```

Contoh berikut membuat aturan EventBridge acara yang dimulai ketika setiap instans EC2 di Region berubah status.

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"]}'
```

Perintah mengembalikan rincian untuk EventBridge aturan baru yang mirip dengan berikut ini.

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. Buat perintah untuk menentukan runbook sebagai target aturan EventBridge acara yang Anda buat di langkah 2. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   Linux & macOS

```
aws events put-targets \
--rule rule name \
--targets '{"Arn": "arn:aws:ssm:region:account ID:automation-definition/runbook
 name","Input":"{\"input parameter\":[\"value\"],\"AutomationAssumeRole\":
[\"arn:aws:iam::123456789012:role/AutomationServiceRole\"]}","Id": "target
 ID","RoleArn": "arn:aws:iam::123456789012:role/service-role/EventBridge service
 role"}'
```

   Windows

```
aws events put-targets ^
--rule rule name ^
--targets '{"Arn": "arn:aws:ssm:region:account ID:automation-definition/runbook
 name","Input":"{\"input parameter\":[\"value\"],\"AutomationAssumeRole\":
[\"arn:aws:iam::123456789012:role/AutomationServiceRole\"]}","Id": "target
 ID","RoleArn": "arn:aws:iam::123456789012:role/service-role/EventBridge service
 role"}'
```

   PowerShell

```
$Target = New-Object Amazon.CloudWatchEvents.Model.Target
$Target.Id = "target ID"
$Target.Arn = "arn:aws:ssm:region:account ID:automation-definition/runbook name"
$Target.RoleArn = "arn:aws:iam::123456789012:role/service-role/EventBridge
 service role"
$Target.Input = '{"input parameter":["value"],"AutomationAssumeRole":
["arn:aws:iam::123456789012:role/AutomationServiceRole"]}'

Write-CWETarget `
```

```
-Rule "rule name" `
-Target $Target
```

Contoh berikut membuat target EventBridge acara yang memulai ID instance tertentu menggunakan runbook`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
```

Sistem mengembalikan informasi seperti berikut.

Linux & macOS

```
{
"FailedEntries": [],
"FailedEntryCount": 0
}
```

Windows

```
{
"FailedEntries": [],
"FailedEntryCount": 0
}
```

PowerShell

Tidak ada output jika perintah berhasil. PowerShell

Jalankan otomatisasi secara manual

Prosedur berikut menjelaskan cara menggunakan AWS Systems Manager konsol dan AWS Command Line Interface (AWS CLI) untuk menjalankan otomatisasi menggunakan mode eksekusi manual. Dengan menggunakan mode eksekusi manual, otomatisasi dimulai dalam status Menunggu dan berhenti sebentar dalam status Menunggu di antara setiap langkah. Hal ini memungkinkan Anda untuk mengontrol kapan otomatisasi berlangsung, yang berguna jika Anda perlu meninjau hasil langkah sebelum melanjutkan.

Otomatisasi berjalan dalam konteks pengguna saat ini. Ini berarti bahwa Anda tidak perlu mengkonfigurasi izin IAM tambahan selama Anda memiliki izin untuk menggunakan runbook, dan tindakan apa pun yang disebut oleh runbook. Jika Anda memiliki izin administrator di IAM, maka Anda sudah memiliki izin untuk menjalankan otomatisasi ini.

Menjalankan otomatisasi langkah demi langkah (konsol)

Prosedur berikut menjelaskan cara menggunakan konsol Systems Manager untuk menjalankan otomatisasi langkah demi langkah.

Untuk menjalankan otomatisasi langkah demi langkah

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Otomatisasi, lalu pilih Eksekusi otomatisasi.
3. Di daftar Dokumen otomatisasi, pilih runbook. Pilih satu opsi atau lebih di panel Kategori dokumen untuk memfilter dokumen SSM sesuai dengan tujuannya. Untuk melihat runbook yang Anda miliki, pilih tab Dimiliki oleh saya. Untuk melihat runbook yang dibagikan dengan akun Anda, pilih tab Dibagikan dengan saya. Untuk melihat semua runbook, pilih tab Semua dokumen.

   > **Note**
   >
   > Anda dapat melihat informasi tentang runbook dengan memilih nama runbook.

4. Di bagian Detail dokumen, verifikasi bahwa Versi dokumen diatur ke versi yang ingin Anda jalankan. Sistem ini termasuk pilihan versi berikut:

   - Versi default saat runtime — Pilih opsi ini jika runbook Otomasi diperbarui secara berkala dan versi default baru ditetapkan.
   - Versi terbaru saat runtime — Pilih opsi ini jika runbook Otomasi diperbarui secara berkala, dan Anda ingin menjalankan versi yang terbaru diperbarui.
   - 1 (Default) - Pilih opsi ini untuk menjalankan versi pertama dokumen, yang merupakan default.

5. Pilih Berikutnya.
6. Di bagian Mode eksekusi, pilih Eksekusi manual.
7. Di bagian Parameter input, tentukan input yang diperlukan. Secara opsional, Anda dapat memilih peran layanan IAM dari daftar. AutomationAssumeRole
8. Pilih Eksekusi.
9. Pilih Eksekusi langkah ini saat Anda siap memulai langkah otomatisasi pertama. Otomatisasi dimulai dengan langkah satu dan berhenti sebelum menjalankan langkah-langkah berikutnya yang ditentukan dalam runbook yang Anda pilih pada langkah 3 prosedur ini. Jika runbook memiliki beberapa langkah, Anda harus memilih Eksekusi langkah ini untuk setiap langkah otomatisasi guna melanjutkan. Setiap kali Anda memilih Eksekusi langkah ini tindakan berjalan.

> **Note**
>
> Konsol menampilkan status otomatisasi. Jika otomatisasi gagal untuk menjalankan langkah, lihat Pemecahan masalah Otomatisasi Systems Manager.

10. Setelah Anda menyelesaikan semua langkah yang ditentukan dalam runbook, pilih Selesaikan dan lihat hasil untuk menyelesaikan otomatisasi dan melihat hasilnya.

Menjalankan otomatisasi langkah demi langkah (baris perintah)

Prosedur berikut menjelaskan cara menggunakan AWS CLI (di Linux,macOS, atau Windows) atau AWS Tools for PowerShell secara manual menjalankan otomatisasi langkah demi langkah.

Untuk menjalankan otomatisasi langkah demi langkah

1. Instal dan konfigurasikan AWS CLI atau AWS Tools for PowerShell, jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI dan Menginstal AWS Tools for PowerShell.

2. Jalankan perintah berikut untuk memulai otomatisasi manual. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   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
```

Berikut adalah contoh menggunakan runbook `AWS-RestartEC2Instance` untuk memulai ulang instans EC2 tertentu.

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"}
```

Sistem mengembalikan informasi seperti berikut.

Linux & macOS

```
{
    "AutomationExecutionId": "ba9cd881-1b36-4d31-a698-0123456789ab"
}
```

Windows

```
{
    "AutomationExecutionId": "ba9cd881-1b36-4d31-a698-0123456789ab"
}
```

PowerShell

```
ba9cd881-1b36-4d31-a698-0123456789ab
```

3. Jalankan perintah berikut bila Anda siap untuk memulai langkah pertama otomatisasi. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri. Otomatisasi dimulai dengan langkah satu dan berhenti sebelum menjalankan langkah-langkah berikutnya yang ditentukan dalam runbook yang Anda pilih pada langkah 1 prosedur ini. Jika runbook memiliki beberapa langkah, Anda harus menjalankan perintah berikut untuk setiap otomatisasi guna melanjutkan.

   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"}
   ```

Jika perintah berhasil, tidak ada output yang akan ditampilkan.

4. Jalankan perintah berikut untuk mengambil status setiap eksekusi langkah dalam otomatisasi.

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
```

Sistem mengembalikan informasi seperti berikut.

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. Jalankan perintah berikut untuk menyelesaikan otomatisasi setelah semua langkah yang ditentukan dalam runbook yang dipilih telah selesai. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   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
   ```

   Tidak ada output jika perintah berhasil.

Penjadwalan otomatisasi

Topik berikut mencakup informasi tentang cara menjadwalkan otomatisasi agar berjalan pada interval tertentu atau pada waktu tertentu yang Anda tentukan.

Konten

Menjadwalkan otomatisasi denganState Manager asosiasi

Anda dapat memulai otomatisasi dengan membuatState Manager asosiasi dengan membuat asosiasi dengan runbook. State Manageradalah kemampuanAWS Systems Manager. Dengan membuatState Manager asosiasi dengan runbook, Anda dapat menargetkan jenisAWS sumber daya yang berbeda. Sebagai contoh, Anda dapat membuat asosiasi yang menerapkan keadaan yang diinginkan di AWS sumber daya, termasuk yang berikut ini:

- Lampirkan peran Systems Manager untuk instans Amazon Elastic Compute Cloud (Amazon EC2) untuk membuatnya menjadi instans terkelola.
- Tegakkan masuk dan keluarnya aturan yang diinginkan untuk grup keamanan.
- Buat atau hapus backup Amazon DynamoDB.
- Buat atau hapus snapshot Amazon Elastic Block Store (Amazon EBS).
- Matikan izin baca dan tulis di bucket Amazon Simple Storage Service (Amazon S3).
- Mulai, mulai ulang, atau hentikan instans terkelola dan instans Amazon Relational Database Service (Amazon RDS).
- Terapkan patch ke Linux, macOS, dan Window AMIs.

Gunakan prosedur berikut untuk membuatState Manager asosiasi yang menjalankan otomatisasi menggunakanAWS Systems Manager konsol danAWS Command Line Interface (AWS CLI).

Sebelum Anda memulai

Berhati-hatilah dengan detail penting berikut sebelum Anda menjalankan otomatisasi dengan menggunakanState Manager:

- Sebelum Anda dapat membuat asosiasi yang menggunakan runbook, verifikasi bahwa Anda mengonfigurasi izin untuk otomatisasi, kemampuan AWS Systems Manager. Untuk informasi selengkapnya, lihat Menyiapkan Otomatisasi.
- State Managerasosiasi yang menggunakan runbook berkontribusi pada jumlah otomatisasi maksimum yang berjalan bersamaan diAkun AWS. Anda dapat memiliki maksimum 100 otomatisasi yang berjalan bersamaan. Untuk selengkapnya, lihat kuota layanan Systems Manager di bagian Referensi Umum Amazon Web Services.
- Saat menjalankan otomatisasi,State Manager tidak mencatat operasi API yang diprakarsai oleh otomatisasi diAWS CloudTrail.
- Systems Manager secara otomatis membuat layanan terkait peran sehinggaState Manager memiliki izin untuk memanggil operasi API Otomatisasi Systems Manager. Jika Anda ingin, Anda dapat membuat peran terkait layanan dengan menjalankan perintah berikut dari AWS CLI atau AWS Tools for PowerShell.

  Linux & macOS

  ```
  aws iam create-service-linked-role \
  --aws-service-name ssm.amazonaws.com
  ```

  Windows

  ```
  aws iam create-service-linked-role ^
  --aws-service-name ssm.amazonaws.com
  ```

  PowerShell

  ```
  New-IAMServiceLinkedRole `
  -AWSServiceName ssm.amazonaws.com
  ```

  Untuk informasi selengkapnya tentang peran terkait layanan, lihat Menggunakan peran terkait layanan untuk Systems Manager.

Mewujudkan asosiasi yang menjalankan otomatisasi (konsol)

Prosedur berikut menjelaskan cara menggunakan konsol Systems Manager untuk membuatState Manager asosiasi yang menjalankan otomatisasi.

Untuk membuatState Manager asosiasi yang menjalankan otomatisasi

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih State Manager, lalu pilih Buat asosiasi.
3. Di bidang Nama, tentukan nama. Ini memang opsional, tetapi direkomendasikan.
4. Di daftar Dokumen, pilih runbook. Gunakan bilah Pencarian untuk memfilter Jenis dokumen: Sama : Otomasi runbook. Untuk melihat lebih banyak runbook, gunakan angka di sebelah kanan bilah Pencarian.

   > **Note**
   >
   > Anda dapat melihat informasi tentang runbook dengan memilih nama runbook.

5. Pilih Eksekusi sederhana untuk menjalankan otomatisasi pada satu target atau lebih dengan menentukan ID sumber daya untuk target tersebut. Pilih Pengendalian rate untuk menjalankan otomatisasi di seluruh armada AWS dengan menentukan opsi penargetan seperti tag atau AWS Resource Groups. Anda juga dapat mengontrol pengoperasian otomatisasi di seluruh sumber daya Anda dengan menentukan konkurensi dan ambang batas kesalahan.

   Jika Anda memilih Pengendalian rate, bagian Target akan ditampilkan.
6. Di bagian Target, pilih metode untuk menargetkan sumber daya.

   a. (Diperlukan) Gunakan daftar Parameter untuk memilih parameter. Item dalam daftar Parameter ditentukan oleh parameter di runbook yang Anda pilih pada awal prosedur ini. Dengan memilih parameter, Anda menentukan jenis sumber daya tempat alur kerja otomatisasi berjalan.

   b. (Diperlukan) Dalam daftar Target, pilih metode untuk menargetkan sumber daya.

      - Grup Sumber Daya: Pilih nama grup dari daftar Grup Sumber Daya. Untuk informasi lebih lanjut tentang penargetan AWS Resource Groups di runbook, lihat Penargetan AWS Resource Groups.
      - Tag: Masukkan kunci tag dan nilai tag di bidang yang disediakan (opsional). Pilih Tambahkan. Untuk informasi lebih lanjut tentang penargetan tag di runbook, lihat Menargetkan tag.
      - Nilai Parameter: Masukkan nilai dalam bagian Parameter input. Jika Anda menentukan beberapa nilai, Systems Manager menjalankan otomatisasi anak pada setiap nilai yang ditentukan.

Sebagai contoh, katakan bahwa runbook Anda mencakup parameter InstanceID. Jika Anda menargetkan nilai-nilai parameter InstanceID ketika menjalankan otomatisasi, Systems Manager menjalankan otomatisasi anak untuk setiap nilai ID instans yang ditentukan. Otomatisasi induk selesai ketika otomatisasi selesai menjalankan setiap instans tertentu, atau jika otomatisasi gagal. Anda dapat menargetkan maksimum 50 nilai parameter. Untuk informasi selengkapnya tentang nilai parameter penargetan dalam runbook, lihat Menanrgetkan nilai parameter.

7. Di bagian Parameter input, tentukan parameter input yang diperlukan.

   Jika Anda memilih untuk menargetkan sumber daya dengan menggunakan tag atau grup sumber daya, maka Anda mungkin tidak perlu memilih beberapa opsi di bagian Parameter input. Misalnya, jika Anda memilih `AWS-RestartEC2Instance` runbook, dan Anda memilih untuk menargetkan instans dengan menggunakan tag, maka Anda tidak perlu menentukan atau memilih ID instans di bagian Parameter input. Otomatisasi menempatkan instans untuk memulai ulang dengan menggunakan tag yang Anda tentukan.

   > **Important**
   >
   > Anda harus menentukan peran ARN di AutomationAssumeRolelapangan. State Managermenggunakan peran asumsi untuk memanggilLayanan AWS yang ditentukan dalam runbook dan menjalankan asosiasi Otomatisasi atas nama Anda.

8. Di bagian Tentukan jadwal, pilih Sesuai jadwal jika Anda ingin menjalankan asosiasi secara berkala. Jika Anda memilih opsi ini, gunakan opsi yang disediakan untuk membuat jadwal menggunakan ekspresi Cron atau Rate. Untuk informasi lebih lanjut tentang ekspresi Cron dan Rate untukState Manager, lihatEkspresi cron dan rate untuk associate.

   > **Note**
   >
   > Ekspresi laju adalah mekanisme penjadwalan pilihan untukState Manager asosiasi yang menggunakan runbook. Ekspresi laju memungkinkan lebih banyak fleksibilitas untuk menjalankan asosiasi dalam hal bahwa Anda mencapai jumlah konkurensi maksimum yang menjalankan otomatisasi. Dengan jadwal laju, Systems Manager dapat kembali mencoba otomatisasi setelah menerima pemberitahuan bahwa otomatisasi bersamaan telah mencapai batas maksimumnya dan telah dicekal.

Pilih Tidak ada jadwal jika Anda ingin menjalankan asosiasi satu kali.

9. (Opsional) dalam bagian Pengendalian Rate, pilih opsi Konkurensi dan Ambang batas kesalahan untuk mengontrol deployment otomasi di seluruh AWS sumber daya Anda.

   a. Di bagian Konkurensi, pilih satu opsi:

      - Pilih target untuk memasukkan jumlah target absolut yang dapat menjalankan otomatisasi secara bersamaan.
      - Pilih persentase untuk memasukkan persentase set target yang dapat menjalankan alur kerja otomatisasi secara bersamaan.

   b. Di bagian Ambang kesalahan, pilih satu opsi:

      - Pilih kesalahan untuk memasukkan jumlah kesalahan absolut yang diizinkan sebelum Otomatisasi berhenti mengirim otomatisasi ke sumber daya lainnya.
      - Pilih persentase untuk memasukkan persentase kesalahan absolut yang diizinkan sebelum Otomatisasi berhenti mengirim otomatisasi ke sumber daya lainnya.

   Untuk informasi selengkapnya tentang target dan kontrol tarif dengan Otomatisasi, lihat Jalankan otomatisasi dalam skala besar.

10. Pilih Buat Asosiasi.

> **Important**
>
> Bila Anda membuat asosiasi, asosiasi tersebut segera berjalan melawan target yang ditentukan. Asosiasi kemudian berjalan berdasarkan ekspresi cron atau laju yang Anda pilih. Jika Anda memilih Tidak ada jadwal, asosiasi tidak berjalan lagi.

Mewujudkan asosiasi yang menjalankan otomatisasi (baris perintah)

Prosedur berikut menjelaskan cara menggunakanAWS CLI (di Linux atau Windows) atauAWS Tools for PowerShell untuk membuatState Manager asosiasi yang menjalankan otomatisasi.

Sebelum Anda memulai

Sebelum Anda menyelesaikan prosedur berikut, pastikan Anda telah membuat peran layanan IAM yang berisi izin yang diperlukan untuk menjalankan runbook, dan mengkonfigurasi hubungan

kepercayaan untuk Otomasi, kemampuanAWS Systems Manager. Untuk informasi selengkapnya, lihat Tugas 1: Buat peran layanan untuk otomatisasi.

Untuk membuat asosiasi yang menjalankan otomatisasi

1. Instal dan konfigurasikan AWS CLI atau AWS Tools for PowerShell, jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaruAWS CLI dan MemasangAWS Tools for PowerShell.

2. Jalankan perintah berikut untuk menandai dokumen.

   Linux & macOS

   ```
   aws ssm list-documents
   ```

   Windows

   ```
   aws ssm list-documents
   ```

   PowerShell

   ```
   Get-SSMDocumentList
   ```

   Perhatikan nama runbook yang ingin Anda gunakan untuk asosiasi.

3. Jalankan perintah berikut untuk menampilkan detail tentang runbook. Dalam perintah berikut, ganti *nama runbook* dengan informasi Anda sendiri.

   Linux & macOS

   ```
   aws ssm describe-document \
   --name runbook name
   ```

   Catat nama parameter (misalnya, `InstanceId`) yang ingin Anda gunakan untuk `--automation-target-parameter-name` pilihan. Parameter ini menentukan jenis sumber daya di mana otomatisasi berjalan.

Windows

```
aws ssm describe-document ^
--name runbook name
```

Catat nama parameter (misalnya, `InstanceId`) yang ingin Anda gunakan untuk `--automation-target-parameter-name` pilihan. Parameter ini menentukan jenis sumber daya di mana otomatisasi berjalan.

PowerShell

```
Get-SSMDocumentDescription `
-Name runbook name
```

Catat nama parameter (misalnya, `InstanceId`) yang ingin Anda gunakan untuk `AutomationTargetParameterName` pilihan. Parameter ini menentukan jenis sumber daya di mana otomatisasi berjalan.

4. Buat perintah yang menjalankan otomatisasi menggunakanState Manager asosiasi. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

   Penargetan menggunakan tag

   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**
   >
   > Jika Anda membuat asosiasi dengan menggunakan AWS CLI, gunakan `--targets` parameter untuk menargetkan instans untuk asosiasi. Jangan gunakan `--instance-id` parameter. Parameter `--instance-id` adalah parameter warisan.

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**
>
> Jika Anda membuat asosiasi dengan menggunakan AWS CLI, gunakan `--targets` parameter untuk menargetkan instans untuk asosiasi. Jangan gunakan `--instance-id` parameter. Parameter `--instance-id` adalah parameter warisan.

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**
>
> Jika Anda membuat asosiasi dengan menggunakan AWS Tools for PowerShell, gunakan `Target` parameter untuk menargetkan instans untuk asosiasi. Jangan

gunakan `InstanceId` parameter. Parameter `InstanceId` adalah parameter warisan.

Menargetkan menggunakan nilai parameter

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"
```

Penargetan menggunakanAWS 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"
```

Menargetkan beberapa akun dan Wilayah

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"]
```

Perintah akan menampilkan detail asosiasi baru yang serupa dengan yang berikut ini.

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**
>
> Jika Anda menggunakan tag untuk membuat asosiasi pada satu kasus target atau lebih, dan kemudian Anda menghapus tag dari instans yang tidak lagi menjalankan asosiasi. Instans dipisahkan dariState Manager dokumen.

Pemecahan masalah otomatisasi dijalankan olehState Manager asosiasi

otomatisasi Systems Manager memberlakukan batas 100 otomatisasi bersamaan, dan 1.000 antri otomatisasi per akun, per Wilayah. JikaState Manager asosiasi yang menggunakan runbook menunjukkan status Gagal dan status AutomationExecutionLimitExceededotomatisasi mungkin telah mencapai batas. Akibatnya, Systems Manager mencekal otomatisasi. Untuk mengatasi masalah ini, lakukan solusi berikut:

- Gunakan tingkat yang berbeda atau ekspresi cron untuk asosiasi Anda. Misalnya, jika asosiasi dijadwalkan untuk berjalan setiap 30 menit, ubah ekspresi agar bisa berjalan setiap satu atau dua jam sekali.
- Hapus otomatisasi yang ada yang memiliki status Menunggu. Dengan menghapus otomatisasi ini, Anda menghapus antrean saat ini.

Jadwalkan otomatisasi dengan jendela pemeliharaan

Anda dapat memulai otomatisasi dengan mengonfigurasi runbook sebagai tugas terdaftar untuk jendela pemeliharaan. Dengan mendaftarkan runbook sebagai tugas terdaftar, jendela pemeliharaan menjalankan otomatisasi selama periode pemeliharaan terjadwal.

Sebagai contoh, katakanlah Anda membuat runbook bernama `CreateAMI` yang menciptakan Amazon Machine Image (AMI) dari instans terdaftar sebagai target ke jendela pemeliharaan. Untuk menentukan `CreateAMI` runbook (dan otomatisasi yang sesuai) sebagai tugas terdaftar dari jendela pemeliharaan, Anda harus terlebih dahulu membuat jendela pemeliharaan dan mendaftarkan target. Kemudian Anda menggunakan prosedur berikut untuk menentukan `CreateAMI` dokumen sebagai tugas terdaftar dalam jendela pemeliharaan. Ketika jendela pemeliharaan dimulai selama periode yang dijadwalkan, sistem menjalankan otomatisasi dan menciptakan AMI dari target terdaftar.

Untuk informasi lebih lanjut tentang pembuatan runbook Otomatisasi, lihat Membuat runbook Anda sendiri. Otomatisasi adalah kemampuan AWS Systems Manager.

Gunakan prosedur berikut untuk mengonfigurasi otomatisasi sebagai tugas terdaftar untuk jendela pemeliharaan menggunakan AWS Systems Manager konsol, AWS Command Line Interface (AWS CLI), atau AWS Tools for Windows PowerShell.

Mendaftarkan tugas otomatisasi ke jendela pemeliharaan (konsol)

Prosedur berikut menjelaskan cara menggunakan konsol Systems Manager untuk mengonfigurasi otomatisasi sebagai tugas terdaftar untuk jendela pemeliharaan.

Sebelum Anda memulai

Sebelum Anda menyelesaikan prosedur berikut, Anda harus membuat jendela pemeliharaan dan mendaftarkan setidaknya satu target. Untuk informasi selengkapnya, lihat prosedur berikut:

- Membuat jendela pemeliharaan (konsol).
- Menetapkan target ke jendela pemeliharaan (konsol)

Untuk mengonfigurasi otomatisasi sebagai tugas terdaftar untuk jendela pemeliharaan

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Pada panel navigasi yang ada di sebelah kiri Maintenance Windows, pilih, dan kemudian pilih jendela pemeliharaan yang ingin Anda daftarkan untuk tugas otomatisasi.
3. Pilih Tindakan. Kemudian pilih Mendaftarkan tugas otomatisasi untuk menjalankan pilihan otomatisasi Anda pada target dengan menggunakan runbook.
4. Untuk Nama, masukkan nama untuk tugas.
5. Untuk Deskripsi, masukkan deskriptif.
6. Untuk Dokumen, pilih runbook yang mendefinisikan tugas yang akan dijalankan.
7. Untuk Versi dokumen, pilih versi runbook untuk digunakan.
8. Untuk Prioritas tugas, tentukan prioritas untuk tugas ini. 1 adalah prioritas tertinggi. Tugas di jendela pemeliharaan dijadwalkan dalam urutan prioritas; tugas yang memiliki prioritas yang sama dijadwalkan secara paralel.
9. Di bagian Target, jika runbook yang Anda pilih adalah salah satu yang menjalankan tugas di sumber daya, identifikasi target tempat Anda ingin menjalankan otomatisasi ini dengan menentukan tag atau memilih instans secara manual.

> **Note**
>
> Jika Anda ingin melewati sumber daya melalui parameter input bukan target, Anda tidak perlu menentukan target jendela pemeliharaan.
> Dalam banyak kasus, Anda tidak perlu secara eksplisit menentukan target untuk tugas otomatisasi. Misalnya, katakanlah bahwa Anda membuat tugas jenis otomatisasi untuk memperbarui Amazon Machine Image (AMI) untuk Linux menggunakan `AWS-UpdateLinuxAmi` runbook. Ketika tugas berjalan, AMI diperbarui dengan paket distribusi Linux terbaru yang tersedia dan perangkat lunak Amazon. Contoh baru dibuat dari AMI telah menginstal pembaruan ini. Karena ID AMI yang akan diperbarui ditentukan dalam parameter input untuk runbook, tidak perlu untuk menentukan target lagi dalam tugas jendela pemeliharaan.

Untuk informasi tentang tugas jendela pemeliharaan yang tidak memerlukan target, lihat the section called “Pendaftaran tugas jendela pemeliharaan tanpa target”.

10. (Opsional) Untuk Pengendalian rate:

> **Note**
>
> Jika tugas yang Anda jalankan tidak menentukan target, Anda tidak perlu menentukan kontrol tarif.

- Untuk Konkurensi, tentukan jumlah atau persentase target untuk menjalankan perintah pada saat yang sama.

  Jika Anda memilih target dengan memilih pasangan kunci nilai tag, dan Anda tidak yakin berapa banyak target yang menggunakan tag terpilih, maka batasi jumlah instans yang dapat menjalankan dokumen pada waktu yang sama dengan menentukan persentase.

  Ketika jendela pemeliharaan berjalan, otomatisasi baru dimulai per target. Ada batas 100 otomatisasi bersamaan per Akun AWS. Jika Anda menentukan tingkat konkurensi lebih besar dari 100, otomatisasi bersama yang lebih dari 100 akan ditambahkan secara otomatis ke antrean otomatisasi. Untuk selengkapnya, lihat kuota layanan Systems Manager di bagian. Referensi Umum Amazon Web Services

- Untuk Ambang batas kesalahan, tetapkan kapan harus berhenti menjalankan otomatisasi pada target lain setelah gagal pada sejumlah atau persentase instans. Misalnya, jika Anda menentukan tiga kesalahan, maka Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Target yang masih memproses otomatisasi juga dapat mengirim kesalahan.

11. Di bagian Parameter input, tentukan parameter untuk runbook. Untuk runbook, sistem akan secara otomatis mengisi beberapa nilai. Anda dapat menyimpan atau mengganti nilai-nilai ini.

> **Important**
>
> Untuk runbook, Anda dapat menentukan Peran Asumsi Otomatisasi. Jika Anda tidak menentukan peran untuk parameter ini, maka otomatisasi mengasumsikan peran layanan jendela pemeliharaan yang Anda pilih di langkah 11. Dengan demikian, Anda harus memastikan bahwa peran layanan jendela pemeliharaan yang Anda pilih memiliki AWS Identity and Access Management izin (IAM) untuk melakukan tindakan yang ditetapkan dalam buku runbook.
> Misalnya, peran terkait layanan untuk Systems Manager tidak memiliki izin IAM `ec2:CreateSnapshot`, yang diperlukan untuk menggunakan runbook `AWS-CopySnapshot`. Dalam skenario ini, Anda harus menggunakan peran layanan jendela pemeliharaan kustom atau menentukan Peran Asumsi Otomatisasi yang memiliki `ec2:CreateSnapshot` izin. Untuk informasi, lihat Menyiapkan Otomatisasi.

12. Di area Peran layanan IAM, pilih peran untuk mengizinkan Systems Manager memulai otomatisasi.

    Untuk membuat peran layanan untuk tugas jendela pemeliharaan, lihatGunakan konsol untuk mengonfigurasi izin untuk jendela pemeliharaan.

13. Pilih Daftarkan tugas otomatisasi.

Mendaftarkan tugas otomatisasi ke jendela pemeliharaan (baris perintah)

Prosedur berikut menjelaskan cara menggunakan AWS CLI (pada Linux atau Windows) atau AWS Tools for PowerShell untuk mengonfigurasi otomatisasi sebagai tugas terdaftar untuk jendela pemeliharaan.

Sebelum Anda memulai

Sebelum Anda menyelesaikan prosedur berikut, Anda harus membuat jendela pemeliharaan dan mendaftarkan setidaknya satu target. Untuk informasi selengkapnya, lihat prosedur berikut:

- Langkah 1: Membuat jendela pemeliharaan (AWS CLI).
- Langkah 2: Mendaftarkan node target dengan jendela pemeliharaan (AWS CLI)

Untuk mengonfigurasi otomatisasi sebagai tugas terdaftar untuk jendela pemeliharaan

1. Instal dan konfigurasikan AWS CLI atau AWS Tools for PowerShell, jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaru AWS CLI dan Memasang AWS Tools for PowerShell.

2. Buat perintah untuk mengonfigurasi otomatisasi sebagai tugas terdaftar untuk jendela pemeliharaan. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

   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**
>
> Jika Anda mengonfigurasi otomatisasi sebagai tugas terdaftar dengan menggunakan AWS CLI, gunakan `--Task-Invocation-Parameters` parameter untuk menentukan parameter yang akan diteruskan ke tugas ketika berjalan. Jangan gunakan `--Task-Parameters` parameter. Parameter `--Task-Parameters` adalah parameter warisan.

Untuk tugas jendela pemeliharaan tanpa target yang ditentukan, Anda tidak dapat memberikan nilai untuk `--max-errors` dan `--max-concurrency`. Sebaliknya, sistem menyisipkan nilai placeholder dari1, yang mungkin dilaporkan sesuai dengan perintah seperti describe-maintenance-window-tasksdan. get-maintenance-window-task Nilai-nilai ini tidak mempengaruhi tugas Anda yang sedang berjalan dan dapat diabaikan.
Untuk informasi tentang tugas jendela pemeliharaan yang tidak memerlukan target, lihat Pendaftaran tugas jendela pemeliharaan tanpa target.

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

Jika Anda mengonfigurasi otomatisasi sebagai tugas terdaftar dengan menggunakan AWS CLI, gunakan `--task-invocation-parameters` parameter untuk menentukan parameter yang akan diteruskan ke tugas ketika berjalan. Jangan gunakan `--task-parameters` parameter. Parameter `--task-parameters` adalah parameter warisan.
Untuk tugas jendela pemeliharaan tanpa target yang ditentukan, Anda tidak dapat memberikan nilai untuk `--max-errors` dan `--max-concurrency`. Sebaliknya, sistem menyisipkan nilai placeholder dari1, yang mungkin dilaporkan sesuai dengan perintah seperti describe-maintenance-window-tasksdan. get-maintenance-window-task Nilai-nilai ini tidak mempengaruhi tugas Anda yang sedang berjalan dan dapat diabaikan.

Untuk informasi tentang tugas jendela pemeliharaan yang tidak memerlukan target, lihat Pendaftaran tugas jendela pemeliharaan tanpa target.

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

Jika Anda mengonfigurasi otomatisasi sebagai tugas terdaftar dengan menggunakan AWS Tools for PowerShell, gunakan `-Automation_Parameter` parameter untuk menentukan parameter yang akan diteruskan ke tugas yang sedang berjalan. Jangan gunakan `-TaskParameters` parameter. Parameter `-TaskParameters` adalah parameter warisan.

Untuk tugas jendela pemeliharaan tanpa target yang ditentukan, Anda tidak dapat memberikan nilai untuk `-MaxError` dan `-MaxConcurrency`. Sebaliknya, sistem menyisipkan nilai placeholder dari 1, yang mungkin dilaporkan sesuai dengan perintah seperti `Get-SSMMaintenanceWindowTaskList` dan `Get-SSMMaintenanceWindowTask`. Nilai-nilai ini tidak mempengaruhi tugas Anda yang sedang berjalan dan dapat diabaikan.

Untuk informasi tentang tugas jendela pemeliharaan yang tidak memerlukan target, lihat Pendaftaran tugas jendela pemeliharaan tanpa target.

Contoh berikut mengonfigurasi otomatisasi sebagai tugas terdaftar ke jendela pemeliharaan dengan prioritas 1. Hal ini juga menunjukkan menghilangkan `--targets`, `--max-errors`, dan `--max-concurrency` pilihan tugas jendela pemeliharaan tanpa target. Otomatisasi

menggunakan `AWS-StartEC2Instance` runbook dan peran asumsi otomatisasi untuk memulai instans EC2 terdaftar sebagai target untuk jendela pemeliharaan. Jendela pemeliharaan menjalankan otomatisasi secara bersamaan pada 5 instans maksimum pada waktu tertentu. Juga, asosiasi ini berhenti berjalan pada lebih banyak instans untuk interval eksekusi tertentu jika jumlah kesalahan melebihi 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
```

Perintah akan menampilkan detail untuk tugas baru yang terdaftar serupa dengan yang berikut ini.

Linux & macOS

```
{
"WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
}
```

Windows

```
{
"WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
}
```

PowerShell

```
4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE
```

3. Untuk melihat tugas terdaftar, jalankan perintah berikut. Ganti *ID jendela pemeliharaan* dengan informasi Anda sendiri.

   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
```

Sistem mengembalikan informasi seperti berikut.

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
```

Referensi tindakan Otomatisasi Systems Manager

Referensi ini menjelaskan tindakan otomatisasi yang dapat Anda tentukan di runbook Otomatisasi. Otomatisasi adalah kemampuan AWS Systems Manager. Tindakan ini tidak dapat digunakan dalam dokumen Systems Manager (SSM) jenis yang lain. Untuk informasi tentang plugin untuk jenis dokumen SSM lainnya, lihat Referensi plugin dokumen perintah.

Otomatisasi Systems Manager menjalankan langkah-langkah yang ditetapkan dalam runbook otomatisasi. Setiap langkah dikaitkan dengan tindakan tertentu. Tindakan menentukan input, perilaku, dan output dari langkah. Langkah-langkah didefinisikan dalam `mainSteps` bagian dari buku runbook Anda.

Anda tidak perlu menentukan output dari suatu tindakan atau langkah. Output yang telah ditentukan oleh tindakan yang terkait dengan langkah. Ketika Anda menentukan input langkah di runbook, Anda

dapat referensi satu output atau lebih dari langkah sebelumnya. Misalnya, Anda dapat membuat output dari `aws:runInstances` yang tersedia untuk tindakan berikutnya `aws:runCommand`. Anda juga dapat referensi output dari langkah-langkah sebelumnya di `Output` bagian runbook.

> **Important**
>
> Jika Anda menjalankan alur kerja otomatisasi yang menjalankan layanan lain dengan menggunakan AWS Identity and Access Management peran layanan (IAM), pastikan bahwa peran layanan harus dikonfigurasi dengan izin untuk menjalankan layanan tersebut. Persyaratan ini berlaku untuk semua AWS Runbook otomatisasi (`AWS-*` runbook) seperti `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup`, dan `AWS-RestartEC2Instance` runbook, untuk beberapa nama. Persyaratan ini juga berlaku untuk setiap runbook Otomasi kustom yang Anda buat yang memanggil orang lain Layanan AWS dengan menggunakan tindakan yang memanggil layanan lain. Misalnya, jika Anda menggunakan `aws:executeAwsApi`, `aws:createStack`, atau `aws:copyImage` tindakan, konfigurasikan peran layanan dengan izin untuk menjalankan layanan tersebut. Anda dapat memberikan izin kepada orang lain Layanan AWS dengan menambahkan kebijakan inline IAM ke peran tersebut. Untuk informasi selengkapnya, lihat (Opsional) Tambahkan kebijakan sebaris Otomasi atau kebijakan terkelola pelanggan untuk memanggil lainnya Layanan AWS.

Topik

- aws:executeAwsApi— Panggil dan jalankan operasi AWS API
- aws:executeScript – Jalankan skrip
- aws:executeStateMachine – Jalankan AWS Step Functions mesin status
- aws:invokeWebhook- Memanggil integrasi webhook Otomasi
- aws:invokeLambdaFunction – Jalankan AWS Lambda fungsi
- aws:loop— Ulangi langkah-langkah dalam otomatisasi
- aws:pause – Jeda otomatisasi
- aws:runCommand – Jalankan perintah pada instans terkelola
- aws:runInstances – Luncurkan Instans Amazon EC2
- aws:sleep – Menunda otomatisasi
- aws:updateVariable— Memperbarui nilai untuk variabel runbook
- aws:waitForAwsResourceProperty – Tunggu di AWS properti sumber daya
- Variabel sistem Otomatisasi

Properti dibagi oleh semua tindakan

Sifat umum adalah parameter atau opsi yang ditemukan di semua tindakan. Beberapa pilihan menentukan perilaku untuk langkah, seperti berapa lama menunggu langkah selesai dan apa yang harus dilakukan jika langkah gagal. Properti berikut umum untuk semua tindakan.

description

Informasi yang Anda berikan untuk menggambarkan tujuan runbook atau langkah.

Tipe: String

Wajib: Tidak

name

Pengenal yang harus unik di semua nama langkah di runbook.

Jenis: String

Pola yang diizinkan: [A-za-Z0-9_] +$

Wajib: Ya

action

Nama tindakan langkah adalah untuk menjalankan. `aws:runCommand` – Jalankan perintah pada instans terkelola adalah contoh dari tindakan yang dapat Anda tentukan di sini. Dokumen ini memberikan informasi mendetail tentang semua tindakan yang tersedia.

Jenis: String

Wajib: Ya

maxAttempts

Berapa kali langkah harus dicoba lagi jika terjadi kegagalan. Jika nilai lebih besar dari 1, langkah tidak dianggap gagal sampai semua upaya coba lagi telah gagal. Nilai default adalah 1.

Jenis: Bilangan bulat

Wajib: Tidak

timeoutSeconds

Nilai batas waktu untuk langkah. Jika batas waktu tercapai dan nilai `maxAttempts` lebih besar dari 1, maka langkah ini tidak dianggap kedaluwarsa sampai semua percobaan telah dicoba.

Jenis: Bilangan bulat

Wajib: Tidak

onFailure

Menunjukkan apakah otomatisasi harus berhenti, melanjutkan, atau meneruskan ke langkah yang berbeda pada kegagalan. Nilai default untuk opsi ini adalah batalkan.

Jenis: String

Nilai yang valid: Batalkan | Lanjutkan | langkah:*`step_name`*

Wajib: Tidak

onCancel

Menunjukkan langkah mana yang harus dilakukan otomatisasi jika pengguna membatalkan otomatisasi. Otomatisasi menjalankan alur kerja pembatalan untuk maksimal dua menit.

Jenis: String

Nilai yang valid: Batalkan | langkah: *`step_name`*

Wajib: Tidak

Properti `onCancel` tidak mendukung pindah ke tindakan berikut:

- `aws:approve`
- `aws:copyImage`
- `aws:createImage`
- `aws:createStack`
- `aws:createTags`
- `aws:loop`
- `aws:pause`
- `aws:runInstances`
- `aws:sleep`

isEnd

Opsi ini menghentikan otomatisasi pada akhir langkah tertentu. Otomatisasi berhenti jika langkah gagal atau berhasil. Nilai default salah.

Jenis: Boolean

Nilai yang valid: benar/salah

Wajib: Tidak

nextStep

Menentukan langkah mana dalam otomatisasi yang harus diproses setelah berhasil menyelesaikan langkah.

Jenis: String

Wajib: Tidak

isCritical

Menunjuk langkah sebagai kepentingan untuk berhasil menyelesaikan otomatisasi. Jika langkah dengan penunjukan ini gagal, maka otomatisasi melaporkan status akhir otomatisasi sebagai gagal. Properti ini hanya dievaluasi jika Anda secara eksplisit mendefinisikannya dalam langkah Anda. Jika `onFailure` properti diatur ke `Continue` dalam langkah, nilai default diatur ke salah. Jika tidak, nilai default untuk opsi ini adalah benar.

Jenis: Boolean

Nilai yang valid: benar/salah

Wajib: Tidak

inputs

Sifat khusus tindakan.

Jenis: Peta

Wajib: Ya

Contoh

```
---
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` – Jeda otomatisasi untuk persetujuan manual

Menjeda Otomatisasi untuk sementara waktu sampai otomatisasi utama yang ditunjuk menyetujui atau menolak tindakan. Setelah jumlah persetujuan yang diperlukan tercapai, otomatisasi dilanjutkan. Anda dapat memasukkan langkah persetujuan di mana saja `mainSteps` di bagian runbook Anda.

> **Note**
>
> Tindakan ini tidak mendukung otomatisasi multi-akun dan Wilayah. Batas waktu default untuk tindakan ini adalah 7 hari (604800 detik) dan nilai maksimum adalah 30 hari (2592000 detik). Anda dapat membatasi atau memperpanjang batas waktu dengan menentukan `timeoutSeconds` parameter untuk `aws:approve` langkah. Jika langkah otomatisasi mencapai nilai timeout sebelum menerima semua keputusan persetujuan yang diperlukan, maka langkah dan otomatisasi berhenti berjalan dan kembali status Timed Out.

Pada contoh berikut, `aws:approve` tindakan menghentikan otomatisasi sementara hingga satu pemberi persetujuan menerima atau menolak otomatisasi. Setelah disetujui, otomatisasi menjalankan PowerShell perintah sederhana.

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"
                  ]
               }
            }
         }
      ]
}
```

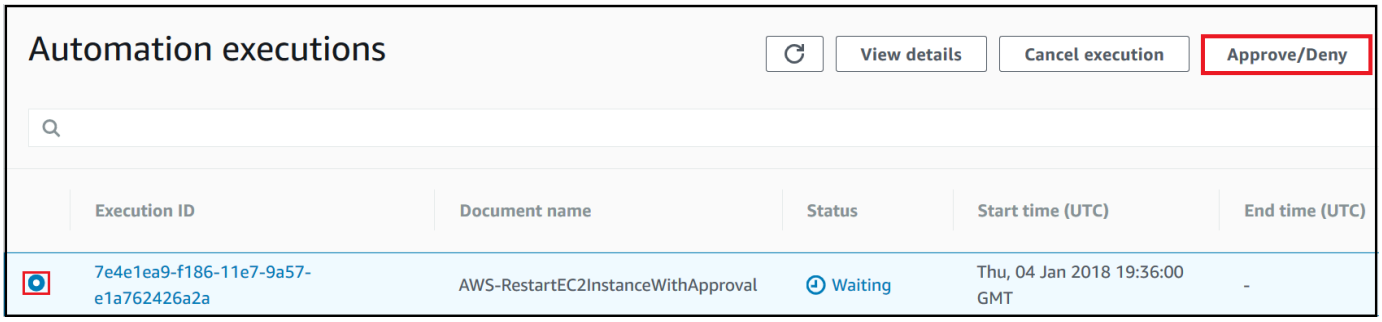
Anda dapat menyetujui atau menolak otomatisasi yang menunggu persetujuan di konsol.

Untuk menyetujui atau menolak otomatisasi menunggu

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Pada panel navigasi, pilih Otomatisasi.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Otomasi.
3. Pilih opsi di sebelah Otomatisasi dengan status Menunggu.



4. Pilih Menyetujui/Menolak.
5. Tinjau detail Otomatisasi.

6. Pilih Setuju atau Tolak, ketik komentar opsional, dan kemudian pilih Kirim.

Contoh masukan

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

Amazon Resource Name (ARN topik Amazon Simple Notification Service (Amazon SNS) untuk persetujuan otomatisasi. Bila Anda menentukan `aws:approve` langkah dalam runbook, Otomatisasi mengirimkan pesan ke topik ini agar otomatisasi utama tahu bahwa mereka harus menyetujui atau menolak langkah otomatisasi. Judul topik Amazon SNS harus diawali dengan "Otomatisi".

Jenis: String

Wajib: Tidak

Pesan

Informasi yang ingin Anda sertakan dalam topik Amazon SNS ketika permintaan persetujuan dikirim. Panjang pesan maksimum adalah 4096 karakter.

Jenis: String

Wajib: Tidak

MinRequiredApprovals

Jumlah minimum persetujuan yang diperlukan untuk melanjutkan otomatisasi. Jika Anda tidak menentukan nilai, default sistem adalah satu. Nilai untuk parameter ini harus berupa angka positif. Nilai untuk parameter ini tidak dapat melebihi jumlah pemberi persetujuan yang ditentukan oleh `Approvers` parameter.

Jenis: Bilangan bulat

Wajib: Tidak

Pemberi persetujuan

Daftar kepala sekolah yang AWS diautentikasi yang dapat menyetujui atau menolak tindakan tersebut. Jumlah maksimum pemberi persetujuan adalah 10. Anda dapat menyebutkan salah satu prinsip dasar berikut dalam kebijakan:

- Nama pengguna
- Pengguna ARN
- IAM role ARN
- IAM mengambil peran ARN

Jenis: StringList

Diperlukan: Ya

Keluaran

ApprovalStatus

Status persetujuan langkah. Status dapat berupa salah satu hal berikut: Disetujui, Ditolak, atau Tunggu. Menunggu berarti Otomasi sedang menunggu masukan dari pemberi persetujuan.

Jenis: String

ApproverDecisions

Sebuah peta JSON yang mencakup keputusan persetujuan setiap pemberi persetujuan.

Jenis: MapList

`aws:assertAwsResourceProperty` – Tegaskan AWS status sumber daya atau status peristiwa

Tindakan `aws:assertAwsResourceProperty` tersebut mengizinkan Anda menegaskan status sumber daya atau status peristiwa tertentu untuk langkah Otomatisasi tertentu. Misalnya, Anda dapat menentukan bahwa langkah Otomatisasi harus menunggu instans Amazon Elastic Compute Cloud (Amazon EC2) untuk memulai. Kemudian akan memanggil operasi DescribeInstanceStatusAPI Amazon EC2 dengan DesiredValue propertirunning. Hal ini memastikan bahwa otomatisasi menunggu instans yang sedang berjalan dan kemudian berlanjut ketika instans tersebut sebenarnya sedang berjalan.

Untuk contoh lebih banyak tentang cara menggunakan tindakan ini, lihatContoh runbook tambahan.

Input

Input didefinisikan oleh operasi API yang Anda pilih.

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"
    ]
  }
}
```

Layanan

Layanan AWSNamespaces yang berisi operasi API yang ingin Anda jalankan. Misalnya, namespace untuk Systems Manager adalah `ssm`. Namespace untuk Amazon EC2 adalah `ec2`. Anda dapat melihat daftarLayanan AWS namespaces yang didukung dalam bagian Layanan yang Tersedia dari ReferensiAWS CLI Perintah.

Tipe: String

Wajib: Ya

Api

Nama operasi API yang ingin Anda jalankan. Anda dapat melihat operasi API (juga disebut metode) dengan memilih layanan di navigasi kiri pada halaman Referensi Layanan. Pilih metode di bagian Klien untuk layanan yang ingin Anda jalankan. Misalnya, semua operasi API (metode) untuk Amazon Relational Database Service (Amazon RDS) tercantum di halaman berikut: Metode Amazon RDS.

Jenis: String

Wajib: Ya

Input operasi API

Satu input operasi API atau lebih. Anda dapat melihat input yang tersediaI (dikenal dengan parameter) dengan memilih layanan di navigasi kiri pada halaman Referensi Layanan berikut. Pilih metode di bagian Klien untuk layanan yang ingin Anda jalankan. Misalnya, semua metode untuk Amazon RDS tercantum di halaman berikut: Metode Amazon RDS. Pilih metode describe_db_instances dan gulir ke bawah untuk melihat parameter yang tersedia, seperti DB, NameInstanceIdentifier, dan Values. Gunakan format berikut untuk menentukan lebih dari satu masukan.

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"
}
```

Jenis: Ditentukan oleh operasi API yang dipilih

Wajib: Ya

PropertySelector

JsonPath untuk atribut tertentu dalam objek respon. Anda dapat melihat obyek respon dengan memilih layanan di navigasi kiri pada halaman Referensi Layanan berikut. Pilih metode di bagian Klien untuk layanan yang ingin Anda jalankan. Misalnya, semua metode untuk Amazon RDS tercantum di halaman berikut: Metode Amazon RDS. Pilih metode describe_db_instances dan gulir ke bawah ke bagian Struktur Respon. DbInstances terdaftar sebagai objek respon.

Jenis: String

Wajib: Ya

DesiredValues

Status yang diharapkan untuk melanjutkan otomatisasi. Jika Anda menentukan nilai Boolean, Anda harus menggunakan huruf kapital seperti Benar atau Salah.

Jenis: StringList

Wajib: Ya

`aws:branch` – Jalankan langkah-langkah otomatisasi bersyarat

Tindakan `aws:branch` tersebut mengizinkan Anda membuat otomatisasi dinamis yang mengevaluasi pilihan yang berbeda dalam satu langkah dan kemudian melompat ke langkah di runbook yang berbeda berdasarkan hasil evaluasi tersebut.

Bila Anda menentukan `aws:branch` tindakan untuk sebuah langkah, Anda menentukan `Choices` bahwa otomatisasi harus mengevaluasi. Dapat `Choices` didasarkan pada nilai yang Anda tentukan dalam `Parameters` bagian runbook, atau nilai dinamis yang dihasilkan sebagai output dari langkah sebelumnya. Otomatisasi mengevaluasi setiap pilihan dengan menggunakan ekspresi Boolean. Jika pilihan pertama adalah benar, maka otomatisasi melompat ke langkah yang ditetapkan untuk pilihan tersebut. Jika pilihan pertama salah, otomatisasi mengevaluasi pilihan berikutnya. Otomatisasi terus mengevaluasi setiap pilihan sampai memproses pilihan yang benar. Selanjutnya, otomatisasi melompat ke langkah yang ditetapkan untuk pilihan yang benar tersebut.

Jika tidak ada pilihan yang benar, otomatisasi memeriksa untuk melihat apakah langkah berisi `default` nilai. Nilai default menentukan langkah yang harus dilakukan otomatisasi jika tidak ada pilihan yang benar. Jika tidak ada `default` nilai yang ditentukan untuk langkah, otomatisasi akan memproses langkah berikutnya dalam runbook.

Tindakan `aws:branch` tersebut mendukung evaluasi pilihan kompleks dengan menggunakan kombinasi `And`, `Not`, dan `Or` operator. Untuk informasi lebih lanjut tentang cara menggunakan `aws:branch`, termasuk contoh runbook dan contoh yang menggunakan operator yang berbeda, lihat Menggunakan pernyataan bersyarat di runbook.

Input

Tentukan satu atau lebih `Choices` dalam satu langkah. Dapat `Choices` didasarkan pada nilai yang Anda tentukan dalam `Parameters` bagian runbook, atau nilai dinamis yang dihasilkan sebagai output dari langkah sebelumnya. Berikut adalah sampel YAML yang mengevaluasi parameter.

```
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
```

Berikut adalah sampel YAML yang mengevaluasi output dari langkah sebelumnya.

```
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
```

Pilihan

Satu ekspresi atau lebih bahwa Otomatisasi harus mengevaluasi ketika menentukan langkah berikutnya untuk proses. Pilihan dievaluasi dengan menggunakan ekspresi Boolean. Setiap templat menyertakan menentukan opsi berikut:

- NextStep: Langkah runbook berikutnya adalah memproses apa pilihan yang ditunjuk true.
- Variabel: Tenentukan nama parameter yang didefinisikan dalam `Parameters` bagian dari buku runbook. Atau tentukan objek output dari langkah sebelumnya dalam runbook. Untuk informasi lebih lanjut tentang permbuatan variabel untuk `aws:branch`, lihat Tentang membuat variabel output.
- Operasi: Kriteria yang digunakan untuk mengevaluasi pilihan. Tindakan `aws:branch` tersebut mendukung operasi berikut:

  Operasi String

  - StringEquals
  - EqualsIgnoreCase

- StartsWith
- EndsWith
- Berisi

Operasi numerik

- NumericEquals
- NumericGreater
- NumericLesser
- NumericGreaterOrEquals
- NumericLesser
- NumericLesserOrEquals

Operasi Boolean

- BooleanEquals

> **Important**
>
> Ketika Anda membuat runbook, sistem memvalidasi setiap operasi di runbook. Jika operasi tidak didukung, sistem akan mengembalikan kesalahan saat Anda mencoba membuat runbook.

Default

Nama langkah yang harus dilakukan otomatisasi jika tidak ada `Choices` yang benar.

Jenis: String

Wajib: Tidak

> **Note**
>
> Tindakan `aws:branch` tersebut mendukung `And`, `Or`, dan `Not` operator. Misalnya `aws:branch` yang menggunakan operator, lihat Menggunakan pernyataan bersyarat di runbook.

`aws:changeInstanceState` – Ubah atau tegaskan status instans

Ubah atau tegaskan status instans.

Tindakan ini dapat digunakan dalam modus menegaskan (tidak menjalankan API untuk mengubah status tetapi memverifikasi instans dalam keadaan yang diinginkan.) Untuk menggunakan modus menegaskan, atur `CheckStateOnly` parameter ke benar. Mode ini berguna saat menjalankan perintah Sysprep pada Windows, yang merupakan perintah asynchronous yang bisa berjalan di latar belakang untuk waktu yang lama. Anda dapat memastikan bahwa instans dihentikan sebelum Anda membuat Amazon Machine Image (AMI).

> **Note**
>
> Nilai batas waktu default untuk tindakan ini adalah 3600 detik (satu jam). Anda dapat membatasi atau memperpanjang batas waktu dengan menentukan `timeoutSeconds` parameter untuk `aws:changeInstanceState` langkah.

Input

YAML

```
name: stopMyInstance
action: aws:changeInstanceState
maxAttempts: 3
timeoutSeconds: 3600
onFailure: Abort
inputs:
  InstanceIds:
  - i-1234567890abcdef0
  CheckStateOnly: true
  DesiredState: stopped
```

JSON

```
{
    "name":"stopMyInstance",
    "action": "aws:changeInstanceState",
    "maxAttempts": 3,
    "timeoutSeconds": 3600,
    "onFailure": "Abort",
```

```
    "inputs": {
        "InstanceIds": ["i-1234567890abcdef0"],
        "CheckStateOnly": true,
        "DesiredState": "stopped"
    }
}
```

InstanceIds

ID instans.

Jenis: StringList

Diperlukan: Ya

CheckStateOnly

Jika salah, tetapkan status instans ke status yang diinginkan. Jika benar, tegaskan status yang diinginkan menggunakan polling.

Default: `false`

Jenis: Boolean

Diperlukan: Tidak

DesiredState

Status yang diinginkan. Ketika diatur ke `running`, aksi ini menunggu status Amazon EC2 menjadi `Running`, Status Instans menjadi `OK`, dan Status Sistem yang menjadi `OK` sebelum selesai.

Jenis: String

Nilai yang valid: `running` | `stopped` | `terminated`

Diperlukan: Ya

Kekuatan

Jika diatur, paksa instans untuk berhenti. Instans yang ada tidak memiliki peluang untuk membersihkan cache sistem file atau metadata sistem file. Jika Anda menggunakan opsi ini, Anda harus melakukan prosedur pemeriksaan dan perbaikan sistem file. Opsi ini tidak disarankan untuk instans EC2 untuk Windows Server.

Jenis: Boolean

Diperlukan: Tidak

AdditionalInfo

Dicadangkan.

Jenis: Tali

Diperlukan: Tidak

Output

Tidak ada

`aws:copyImage` – Salin atau enkripsi Amazon Machine Image

Menyalin Amazon Machine Image (AMI) dari semua Wilayah AWS ke Wilayah saat ini. Tindakan ini juga dapat mengenkripsi AMI.

Input

Tindakan ini mendukung sebagian besar `CopyImage` parameter. Untuk informasi selengkapnya, lihatCopyImage.

Contoh berikut membuat salinan AMI di wilayah Seoul (`SourceImageID`: ami-0fe10819. `SourceRegion`: ap-northeast-2). Baru AMI disalin ke wilayah di mana Anda memulai tindakan otomatisasi. Yang disalin AMI akan dienkripsi karena bendera opsional `Encrypted` diatur ke `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

Wilayah tempat sumbernya AMI berada.

Jenis: Tali

Diperlukan: Ya

SourceImageId

ID AMI untuk menyalin dari Sumber Wilayah.

Jenis: Tali

Diperlukan: Ya

ImageName

Nama gambar baru.

Jenis: Tali

Diperlukan: Ya

ImageDescription

Deskripsi gambar target.

Jenis: Tali

Diperlukan: Tidak

Dienkripsi

Enkripsi target AMI.

Jenis: Boolean

Diperlukan: Tidak

KmsKeyId

Amazon Resource Name (ARN) yang penuh AWS KMS key akan digunakan ketika mengenkripsi snapshot gambar selama operasi penyalinan. Untuk informasi selengkapnya, lihatCopyImage.

Jenis: Tali

Diperlukan: Tidak

ClientToken

Pengenal unik dan peka huruf besar yang Anda berikan untuk memastikan permintaan idempotensi. Untuk informasi selengkapnya, lihatCopyImage.

Jenis: Tali

Diperlukan: Tidak

Output

ImageId

ID dari gambar yang disalin.

ImageState

Keadaan gambar yang disalin.

Nilai yang Valid: `available` | `pending` | `failed`

`aws:createImage` – Buat Amazon machine image (AMI)

MembuatAmazon Machine Image (AMI) dari sebuah instans yang berjalan, berhenti, atau berhenti.

Input

Tindakan ini mendukung`CreateImage` parameter berikut. Untuk informasi lebih lanjut, lihat CreateImage.

YAML

```
name: createMyImage
action: aws:createImage
maxAttempts: 3
onFailure: Abort
inputs:
  InstanceId: i-1234567890abcdef0
  ImageName: AMI Created on{{global:DATE_TIME}}
  NoReboot: true
  ImageDescription: My newly created AMI
```

JSON

```
{
    "name": "createMyImage",
    "action": "aws:createImage",
    "maxAttempts": 3,
    "onFailure": "Abort",
    "inputs": {
        "InstanceId": "i-1234567890abcdef0",
        "ImageName": "AMI Created on{{global:DATE_TIME}}",
        "NoReboot": true,
        "ImageDescription": "My newly created AMI"
    }
}
```

InstanceId

ID instance.

Jenis: String

Wajib: Ya

ImageName

Nama gambar.

Jenis: String

Wajib: Ya

ImageDescription

Deskripsi alias.

Jenis: String

Wajib: Tidak

NoReboot

Boolean literal.

Secara default, Amazon Elastic Compute Cloud (Amazon EC2) mencoba untuk mematikan dan reboot instans sebelum membuat gambar. Jika pilihan Tidak Reboot diatur ke `true`, Amazon EC2 tidak menutup instans sebelum membuat gambar. Bila opsi ini digunakan, integritas sistem file pada gambar yang dibuat tidak dapat dijamin.

Jika Anda tidak ingin instance berjalan setelah Anda membuatAMI darinya, pertama-tama gunakan`aws:changeInstanceState` – Ubah atau tegaskan status instans tindakan untuk menghentikan instance, dan kemudian gunakan`aws:createImage` tindakan ini dengan NoRebootopsi yang disetel ke`true`.

Tipe: Boolean

Wajib: Tidak

BlockDeviceMappings

Perangkat blok untuk instans.

Jenis: Peta

Wajib: Tidak

Output

ImageId

ID gambar yang baru dibuat.

Jenis: String

ImageState

Keadaan gambar saat ini (). Jika status tersedia, gambar berhasil terdaftar dan dapat digunakan untuk meluncurkan sebuah instans.

Jenis: String

`aws:createStack` – Buat AWS CloudFormation tumpukan

Membuat AWS CloudFormation tumpukan dari templat.

Untuk informasi tambahan tentang membuat CloudFormation tumpukan, lihat CreateStackdi Referensi API. AWS CloudFormation

Masukan

YAML

```
name: makeStack
action: aws:createStack
maxAttempts: 1
onFailure: Abort
inputs:
  Capabilities:
  - CAPABILITY_IAM
  StackName: myStack
  TemplateURL: http://s3.amazonaws.com/doc-example-bucket/myStackTemplate
  TimeoutInMinutes: 5
  Parameters:
    - ParameterKey: LambdaRoleArn
      ParameterValue: "{{LambdaAssumeRole}}"
    - ParameterKey: createdResource
      ParameterValue: createdResource-{{automation:EXECUTION_ID}}
```

JSON

```
{
    "name": "makeStack",
    "action": "aws:createStack",
    "maxAttempts": 1,
```

```
    "onFailure": "Abort",
    "inputs": {
        "Capabilities": [
            "CAPABILITY_IAM"
        ],
        "StackName": "myStack",
        "TemplateURL": "http://s3.amazonaws.com/doc-example-bucket/myStackTemplate",
        "TimeoutInMinutes": 5,
        "Parameters": [
          {
            "ParameterKey": "LambdaRoleArn",
            "ParameterValue": "{{LambdaAssumeRole}}"
          },
          {
            "ParameterKey": "createdResource",
            "ParameterValue": "createdResource-{{automation:EXECUTION_ID}}"
          }
    }
}
```

Kemampuan

Daftar nilai yang Anda tentukan sebelumnya CloudFormation dapat membuat tumpukan tertentu layanan. Beberapa templat tumpukan mencakup sumber daya yang dapat mempengaruhi izin di Akun AWS. Untuk tumpukan tersebut, Anda harus secara eksplisit mengakui kemampuan mereka dengan menentukan parameter ini.

Nilai yang valid termasuk `CAPABILITY_IAM`, `CAPABILITY_NAMED_IAM`, dan `CAPABILITY_AUTO_EXPAND`.

CAPABILITY_IAM dan CAPABILITY_NAMED_IAM

Jika Anda memiliki sumber daya IAM, Anda dapat menentukan salah satu kemampuan. Jika Anda memiliki sumber daya IAM dengan nama kustom, Anda harus menentukan `CAPABILITY_NAMED_IAM`. Jika Anda tidak menentukan parameter ini, tindakan ini mengembalikan sebuah `InsufficientCapabilities` kesalahan. Sumber daya berikut mengharuskan Anda untuk menentukan salah satu `CAPABILITY_IAM`atau `CAPABILITY_NAMED_IAM`.

- `AWS::IAM::AccessKey`
- `AWS::IAM::Group`

- AWS::IAM::InstanceProfile
- AWS::IAM::Policy
- AWS::IAM::Role
- AWS::IAM::User
- AWS::IAM::UserToGroupAddition

Jika templat tumpukan berisi sumber daya ini, sebaiknya Anda meninjau semua izin yang terkait dengannya dan mengedit izinnya, jika diperlukan.

Untuk informasi lebih lanjut, lihat Mengakui Sumber Daya IAM di AWS CloudFormation Templat.

CAPABILITY_AUTO_EXPAND

Beberapa templat berisi makro. Macro melakukan pengolahan kustom pada templat; ini dapat mencakup tindakan sederhana seperti find-and-replace operasi, sampai ke transformasi ekstensif seluruh templat. Karena ini, pengguna biasanya membuat perubahan yang ditetapkan dari templat yang diproses, sehingga mereka dapat meninjau perubahan yang dihasilkan dari makro sebelum benar-benar membuat tumpukan. Jika templat tumpukan berisi satu makro atau lebih, dan Anda memilih untuk membuat tumpukan langsung dari templat yang diproses, tanpa terlebih dahulu meninjau perubahan yang dihasilkan dalam satu set perubahan, Anda harus mengakui kemampuan ini.

Untuk informasi lebih lanjut, lihat MenggunakanAWS CloudFormation Macro untuk Melakukan Pengolahan Kustom pada Template di AWS CloudFormation Panduan Pengguna.

Jenis: Array string

Nilai yang Valid: `CAPABILITY_IAM | CAPABILITY_NAMED_IAM | CAPABILITY_AUTO_EXPAND`

Wajib: Tidak

ClientRequestToken

Pengidentifikasi unik untuk CreateStack permintaan ini. Tentukan token ini jika Anda menetapkan maxAttempts dalam langkah ini untuk nilai yang lebih besar dari 1. Dengan menentukan token ini, CloudFormation tahu bahwa Anda tidak mencoba untuk membuat tumpukan baru dengan nama yang sama.

Tipe: String

Wajib: Tidak

Panjang Batasan: Panjang minimum 1. Panjang maksimum 128.

Pola: [a-zA-Z0-9][-a-zA-Z0-9]*

DisableRollback

Atur ke `true` untuk menonaktifkan rollback tumpukan jika pembuatan tumpukan gagal.

Bersyarat: Anda dapat menentukan salah satu `DisableRollback` parameter atau `OnFailure` parameter, tapi tidak keduanya.

Default: `false`

Jenis: Boolean

Wajib: Tidak

NotificationARNS

ARN topik Amazon Simple Notification Service (Amazon SNS) untuk menerbitkan kejadian terkait tumpukan. Anda dapat menemukan ARN topik SNS menggunakan konsol Amazon SNS, https://console.aws.amazon.com/sns/v3/home.

Jenis: Array string

Anggota Array: Jumlah maksimum 5 item.

Wajib: Tidak

OnFailure

Menentukan tindakan yang harus diambil jika pembuatan tumpukan gagal. Anda harus menentukan `DO_NOTHING`, `ROLLBACK`, atau `DELETE`.

Bersyarat: Anda dapat menentukan salah satu `OnFailure` parameter atau `DisableRollback` parameter, tapi tidak keduanya.

Default: `ROLLBACK`

Jenis: String

Nilai yang Valid: `DO_NOTHING | ROLLBACK | DELETE`

Wajib: Tidak

Parameter

Daftar `Parameter` struktur yang menentukan parameter input untuk tumpukan. Untuk informasi selengkapnya, lihat jenis data Parameter.

Jenis: array dari objek Parameter

Wajib: Tidak

ResourceTypes

Jenis sumber daya templat yang izinnya Anda gunakan untuk membuat tindakan tumpukan ini. Misalnya: `AWS::EC2::Instance`, `AWS::EC2::*`, atau `Custom::MyCustomInstance`. Gunakan sintaks berikut untuk menggambarkan jenis sumber daya templat.

- Untuk semua AWS sumber daya:

```
AWS::*
```

- Untuk semua sumber daya kustom:

```
Custom::*
```

- Untuk sumber daya kustom tertentu:

```
Custom::logical_ID
```

- Untuk semua sumber daya tertentu layananLayanan AWS:

```
AWS::service_name::*
```

- Untuk AWS sumber daya tertentu:

```
AWS::service_name::resource_logical_ID
```

Jika daftar jenis sumber daya tidak termasuk sumber daya yang Anda buat, pembuatan tumpukan gagal. Secara default, CloudFormation memberikan izin ke semua jenis sumber daya. IAM menggunakan parameter ini untuk kunci kondisi CloudFormation -spesifik dalam kebijakan

IAM. Untuk informasi selengkapnya, lihat Mengontrol Akses dengan AWS Identity and Access Management.

Jenis: Array string

Panjang Batasan: Panjang minimum 1. Panjang maksimum 256.

Wajib: Tidak

RoleArn

Amazon Resource Name (ARN) IAM role yang diterima CloudFormation untuk membuat tumpukan. CloudFormationmenggunakan kredensial peran untuk melakukan panggilan atas nama Anda. CloudFormationselalu menggunakan peran ini untuk semua operasi di future di tumpukan. Selama pengguna memiliki izin untuk beroperasi di tumpukan, CloudFormation gunakan peran ini meskipun pengguna tidak memiliki izin untuk melewatinya. Pastikan bahwa peran memberikan sedikitnya jumlah hak istimewa.

Jika Anda tidak menentukan nilai, CloudFormation gunakan peran yang sebelumnya terkait dengan tumpukan. Jika tidak ada peran yang tersedia, CloudFormation gunakan sesi sementara yang dihasilkan dari kredensial pengguna Anda.

Jenis: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Wajib: Tidak

StackName

Nama yang terkait dengan tumpukan. Nama harus unik di Daerah di mana Anda membuat tumpukan.

> **Note**
>
> Sebuah nama tumpukan dapat berisi karakter alfanumerik (peka huruf besar/kecil) dan tanda hubung. Ini harus dimulai dengan karakter abjad dan tidak boleh lebih dari 128 karakter.

Jenis: String

Wajib: Ya

StackPolicyBody

Struktur yang berisi badan kebijakan tumpukan. Untuk informasi lebih lanjut, lihat Cegah Pembaruan untuk Sumber Daya Tumpukan.

Bersyarat: Anda dapat menentukan salah satu `StackPolicyBody` parameter atau `StackPolicyURL` parameter, tapi tidak keduanya.

Jenis: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 16384.

Wajib: Tidak

StackPolicyURL

Lokasi file yang berisi kebijakan tumpukan. URL harus menunjuk ke kebijakan yang terletak di bucket S3 di wilayah yang sama dengan tumpukan. Ukuran file maksimum yang diizinkan untuk kebijakan tumpukan adalah 16 KB.

Bersyarat: Anda dapat menentukan salah satu `StackPolicyBody` parameter atau `StackPolicyURL` parameter, tapi tidak keduanya.

Jenis: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 1350.

Wajib: Tidak

Tag

Pasangan nilai kunci untuk berasosiasi dengan tumpukan ini. CloudFormationjuga menyebarkan tag ini ke sumber daya yang dibuat dalam tumpukan. Anda dapat menentukan jumlah maksimum 10 tag.

Jenis: Array objek Tag

Wajib: Tidak

TemplateBody

Struktur yang mengandungi body templat dengan panjang minimum 1 byte dan panjang maksimum 51.200 byte. Untuk informasi lebih lanjut, lihat Anatomi Templat.

Bersyarat: Anda dapat menentukan salah satu `TemplateBody` parameter atau `TemplateURL` parameter, tapi tidak keduanya.

Jenis: String

Panjang Batasan: Panjang minimum 1.

Wajib: Tidak

TemplateURL

Lokasi file yang mengandungi badan templat. URL harus menunjuk ke templat yang terletak di bucket S3. Ukuran maksimum yang diizinkan untuk template adalah 460.800 byte. Untuk informasi lebih lanjut, lihat Anatomi Templat.

Bersyarat: Anda dapat menentukan salah satu `TemplateBody` parameter atau `TemplateURL` parameter, tapi tidak keduanya.

Jenis: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 1024.

Wajib: Tidak

TimeoutInMinutes

Lamanya waktu yang dapat berlalu sebelum status tumpukan menjadi `CREATE_FAILED`. Jika `DisableRollback` tidak diatur atau diatur ke `false`, tumpukan akan dibatalkan.

Jenis: Bilangan bulat

Rentang yang Valid: Nilai minimum 1.

Wajib: Tidak

Output

StackId

Pengenal tumpukan yang unik.

Jenis: String

StackStatus

Status tumpukan saat ini.

Jenis: String

Nilai yang Valid: `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`

Wajib: Ya

StackStatusReason

Pesan sukses atau gagal yang terkait dengan status tumpukan.

Jenis: String

Wajib: Tidak

Untuk informasi lebih lanjut, lihat CreateStack.

Pertimbangan keamanan

Sebelum Anda dapat menggunakan `aws:createStack` tindakan, Anda harus menetapkan kebijakan berikut untuk peran asumsi Otomatisasi IAM. Untuk informasi lebih lanjut tentang peran asumsi, lihat Tugas 1: Buat peran layanan untuk otomatisasi.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "sqs:*",
            "cloudformation:CreateStack",
            "cloudformation:DescribeStacks"
         ],
         "Resource":"*"
      }
   ]
}
```

`aws:createTags` – Buat tag untuk AWS sumber daya

Menciptakan tag baru untuk instans Amazon Elastic Compute Cloud (Amazon EC2) atau AWS Systems Manager instans terkelola.

Input

Tindakan ini mendukung sebagian besar parameter Amazon EC2 `CreateTags` dan Systems Manager `AddTagsToResource`. Untuk informasi selengkapnya, lihatCreateTagsdanAddTagsToResource.

Contoh berikut menunjukkan cara menandai tag Amazon Machine Image (AMI) dan sebuah instans sebagai sumber daya produksi untuk departemen tertentu.

YAML

```
name: createTags
action: aws:createTags
maxAttempts: 3
onFailure: Abort
inputs:
  ResourceType: EC2
  ResourceIds:
  - ami-9a3768fa
  - i-02951acd5111a8169
  Tags:
  - Key: production
    Value: ''
  - Key: department
    Value: devops
```

JSON

```
{
    "name": "createTags",
    "action": "aws:createTags",
    "maxAttempts": 3,
    "onFailure": "Abort",
    "inputs": {
        "ResourceType": "EC2",
        "ResourceIds": [
            "ami-9a3768fa",
```

```
            "i-02951acd5111a8169"
        ],
        "Tags": [
            {
                "Key": "production",
                "Value": ""
            },
            {
                "Key": "department",
                "Value": "devops"
            }
        ]
    }
}
```

ResourceIds

ID sumber daya (s) yang akan ditandai. Jika jenis sumber daya bukan “EC2”, bidang ini hanya dapat berisi satu item.

Jenis: Daftar string

Diperlukan: Ya

Tanda

Tag untuk mengasosiasikan dengan sumber daya.

Jenis: Daftar Peta

Diperlukan: Ya

ResourceType

ID sumber daya yang akan ditandai. Jika tidak disediakan, nilai default “EC2” digunakan.

Jenis: Tali

Diperlukan: Tidak

Nilai yang benar: `EC2` | `ManagedInstance` | `MaintenanceWindow` | `Parameter`

Output

Tidak ada

`aws:deleteImage` – Hapus Amazon Machine Image

Menghapus Amazon Machine Image (AMI) khusus dan semua snapshot terkait.

Input

Tindakan ini hanya mendukung satu parameter. Untuk informasi lebih lanjut, lihat dokumentasi untukDeregisterImagedanDeleteSnapshot.

YAML

```
name: deleteMyImage
action: aws:deleteImage
maxAttempts: 3
timeoutSeconds: 180
onFailure: Abort
inputs:
  ImageId: ami-12345678
```

JSON

```
{
    "name": "deleteMyImage",
    "action": "aws:deleteImage",
    "maxAttempts": 3,
    "timeoutSeconds": 180,
    "onFailure": "Abort",
    "inputs": {
        "ImageId": "ami-12345678"
    }
}
```

ImageId

ID gambar yang akan dihapus.

Jenis: Tali

Diperlukan: Ya

Output

Tidak ada

aws:deleteStack – Hapus AWS CloudFormation tumpukan

Menghapus AWS CloudFormation tumpukan.

Masukan

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

Pengidentifikasi unik untuk `DeleteStack` permintaan ini. Tentukan token ini jika Anda berencana untuk mencoba permintaan kembali sehingga CloudFormation tahu bahwa Anda tidak mencoba menghapus tumpukan dengan nama yang sama. Anda dapat mencoba`DeleteStack` permintaan kembali untuk memverifikasi bahwa CloudFormation menerimanya.

Jenis: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 128.

Pola: [a-zA-Z][-a-zA-Z0-9]*

Wajib: Tidak

RetainResources.Member.n

Masukan ini hanya berlaku untuk tumpukan yang berada di `DELETE_FAILED` status. Daftar ID sumber daya logis untuk sumber daya yang ingin Anda pertahankan. Selama penghapusan, CloudFormation menghapus tumpukan, namun tidak menghapus sumber daya yang disimpan.

Mempertahankan sumber daya berguna bila Anda tidak dapat menghapus sumber daya, seperti bucket S3 yang tidak kosong, namun Anda ingin menghapus tumpukan.

Jenis: array string

Wajib: Tidak

RoleArn

Amazon Resource Name (ARN) dari peranAWS Identity and Access Management (IAM) yang CloudFormation diasumsikan untuk membuat tumpukan. CloudFormation menggunakan kredensial peran untuk melakukan panggilan atas nama Anda. CloudFormation selalu menggunakan peran ini untuk semua operasi di future di tumpukan. Selama pengguna memiliki izin untuk beroperasi di tumpukan, CloudFormation gunakan peran ini meskipun pengguna tidak memiliki izin untuk melewatinya. Pastikan bahwa peran memberikan sedikitnya jumlah hak istimewa.

Jika Anda tidak menentukan nilai, CloudFormation gunakan peran yang sebelumnya terkait dengan tumpukan. Jika tidak ada peran yang tersedia, CloudFormation gunakan sesi sementara yang dihasilkan dari kredensial pengguna Anda.

Jenis: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Wajib: Tidak

StackName

Nama atau ID tumpukan unik yang berhubungan dengan tumpukan.

Jenis: String

Wajib: Ya

Pertimbangan keamanan

Sebelum Anda dapat menggunakan `aws:deleteStack` tindakan, Anda harus menetapkan kebijakan berikut untuk peran asumsi Otomatisasi IAM. Untuk informasi lebih lanjut tentang peran asumsi, lihat Tugas 1: Buat peran layanan untuk otomatisasi.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "sqs:*",
            "cloudformation:DeleteStack",
            "cloudformation:DescribeStacks"
         ],
         "Resource":"*"
      }
   ]
}
```

`aws:executeAutomation` – Jalankan otomatisasi lain

Menjalankan otomatisasi sekunder dengan memanggil runbook sekunder. Dengan tindakan ini, Anda dapat membuat runbook untuk operasi Anda yang paling umum, dan referensi runbook tersebut selama otomatisasi. Tindakan ini dapat menyederhanakan runbook Anda dengan menghapus kebutuhan untuk menduplikasi langkah-langkah di runbook serupa.

Otomatisasi sekunder berjalan dalam konteks pengguna yang memulai otomatisasi utama. Ini berarti bahwa otomatisasi sekunder menggunakanAWS Identity and Access Management (IAM) role yang sama atau pengguna sebagai pengguna yang memulai otomatisasi pertama.

> **Important**
>
> Jika Anda menetapkan parameter dalam otomatisasi sekunder yang menggunakan peran asumsi (peran yang menggunakan kebijakan iam:passRole), maka pengguna atau peran yang memulai otomatisasi utama harus memiliki izin untuk melewati peran asumsi yang ditentukan dalam otomatisasi sekunder. Untuk informasi lebih lanjut tentang pengaturan peran asumsi untuk otomatisasi, lihat Metode 2: Gunakan IAM untuk mengonfigurasi peran untuk Otomatisasi.

Masukan

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

Nama runbook sekunder yang dijalankan selama langkah. Untuk runbook yang sama Akun AWS, tentukan nama runbook. Untuk runbook bersama dari Akun AWS yang berbeda, tentukan Amazon Resource Name (ARN) runbook. Untuk informasi tentang menggunakan runbook bersama, lihat Menggunakan dokumen SSM bersama.

Jenis: String

Wajib: Ya

DocumentVersion

Runbook versi sekunder yang akan dijalankan. Jika tidak ditentukan, Otomatisasi menjalankan runbook versi default.

Jenis: String

Wajib: Tidak

MaxConcurrency

Jumlah maksimum target yang diizinkan untuk menjalankan tugas ini secara parallel. Anda dapat menentukan nomor, seperti 10, atau persentase, seperti 10%.

Tipe: String

Wajib: Tidak

MaxErrors

Jumlah kesalahan yang diizinkan sebelum maka sistem berhenti menjalankan otomatisasi pada target tambahan. Anda dapat menentukan jumlah kesalahan absolut, misalnya 10, atau persentase target yang ditetapkan, misalnya 10%. Jika Anda menentukan 3, misalnya, maka sistem berhenti menjalankan otomatisasi saat kesalahan keempat diterima. Jika Anda menentukan 0, maka sistem berhenti menjalankan otomatisasi pada target tambahan setelah hasil kesalahan pertama dikembalikan. Jika Anda menjalankan otomatisasi pada 50 sumber daya dan mengatur`MaxErrors` menjadi 10%, maka sistem berhenti menjalankan otomatisasi pada target tambahan saat kesalahan keenam diterima.

Otomatisasi yang sudah berjalan saat`MaxErrors` ambang batas tercapai diperbolehkan untuk menyelesaikan, tetapi beberapa otomatisasi ini mungkin gagal juga. Jika Anda perlu memastikan bahwa tidak akan ada otomatisasi yang lebih gagal daripada yang ditentukan`MaxErrors`, atur`MaxConcurrency` ke 1 sehingga otomatisasi dilanjutkan satu per satu.

Tipe: String

Wajib: Tidak

RuntimeParameters

Diperlukan parameter untuk runbook sekunder. Pemetaan menggunakan format berikut: {"parameter1" : "value1", "parameter2" : "value2" }

Jenis: Peta

Wajib: Tidak

Tanda

Metadata opsional yang Anda tetapkan ke sumber daya. Anda dapat menentukan maksimal lima tag untuk otomatisasi.

Jenis: MapList

Wajib: Tidak

TargetLocations

Lokasi adalah kombinasi dariWilayah AWS dan/atauAkun AWS tempat Anda ingin menjalankan otomatisasi. Jumlah minimum 1 item harus ditentukan dan jumlah maksimum 100 item dapat ditentukan.

Jenis: MapList

Wajib: Tidak

TargetMaps

Daftar pemetaan nilai-kunci dari parameter dokumen untuk menargetkan sumber daya. Keduanya`Targets` dan tidak`TargetMaps` dapat ditentukan bersama.

Jenis: MapList

Wajib: Tidak

TargetParameterName

Nama parameter yang digunakan sebagai sumber daya target untuk otomatisasi tingkat dikendalikan. Diperlukan jika Anda menentukan`Targets`.

Tipe: String

Wajib: Tidak

Target

Daftar pemetaan nilai-kunci untuk menargetkan sumber daya. Diperlukan jika Anda menentukan`TargetParameterName`.

Jenis: MapList

Wajib: Tidak

Output

Output

Output yang dihasilkan oleh otomatisasi sekunder. Anda dapat mereferensikan output dengan menggunakan format berikut: *Secondary_Automation_Step_Name*.Output

Jenis: StringList

Berikut ini contohnya:

```
- 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

ID otomatisasi sekunder.

Jenis: String

Status

Status otomatisasi sekunder.

Jenis: String

aws:executeAwsApi— Panggil dan jalankan operasi AWS API

Memanggil dan menjalankan operasi AWS API. Sebagian besar operasi API didukung, meskipun tidak semua operasi API telah diuji. Operasi API streaming, seperti GetObjectoperasi, tidak didukung. Jika Anda tidak yakin apakah operasi API yang ingin Anda gunakan adalah operasi streaming, tinjau dokumentasi Boto3 untuk layanan tersebut guna menentukan apakah API memerlukan input atau output streaming. Kami secara teratur memperbarui versi Boto3 yang digunakan oleh tindakan ini. Namun, setelah rilis versi Boto3 baru, diperlukan waktu hingga beberapa minggu agar perubahan tercermin dalam tindakan ini. Setiap `aws:executeAwsApi` tindakan dapat berjalan hingga durasi maksimum 25 detik. Untuk contoh lebih lanjut tentang cara menggunakan tindakan ini, lihatContoh runbook tambahan.

Masukan

Input didefinisikan oleh operasi API yang Anda pilih.

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"
      }
   ]
}
```

Layanan

Layanan AWS Namespace yang berisi operasi API yang ingin Anda jalankan. Anda dapat melihat daftar Layanan AWS ruang nama yang didukung di Layanan yang tersedia dari. AWS SDK for Python (Boto3) Namespace dapat ditemukan di bagian Klien. Misalnya, namespace untuk Systems Manager adalah `ssm`. Namespace untuk Amazon Elastic Compute Cloud (Amazon EC2) adalah `ec2`.

Jenis: String

Wajib: Ya

Api

Nama operasi API yang ingin Anda jalankan. Anda dapat melihat operasi API (juga disebut metode) dengan memilih layanan di navigasi kiri pada halaman Referensi Layanan. Pilih metode di bagian Klien untuk layanan yang ingin Anda jalankan. Misalnya, semua operasi API (metode)

untuk Amazon Relational Database Service (Amazon RDS) tercantum di halaman berikut: Metode Amazon RDS.

Jenis: String

Wajib: Ya

Input operasi API

Satu input operasi API atau lebih. Anda dapat melihat input yang tersediaI (dikenal dengan parameter) dengan memilih layanan di navigasi kiri pada halaman Referensi Layanan berikut. Pilih metode di bagian Klien untuk layanan yang ingin Anda jalankan. Misalnya, semua metode untuk Amazon RDS tercantum di halaman berikut: Metode Amazon RDS. Pilih metode describe_db_instances dan gulir ke bawah untuk melihat parameter yang tersedia, seperti DB, Nama, dan Nilai. InstanceIdentifier

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"
}
```

Jenis: Ditentukan oleh operasi API yang dipilih

Diperlukan: Ya

Output

Output ditentukan oleh pengguna berdasarkan respon dari operasi API yang dipilih.

Nama

Nama untuk output.

Jenis: String

Wajib: Ya

Pemilih

JsonPath untuk atribut tertentu dalam objek respon. Anda dapat melihat obyek respon dengan memilih layanan di navigasi kiri pada halaman Referensi Layanan berikut. Pilih metode di bagian Klien untuk layanan yang ingin Anda jalankan. Misalnya, semua metode untuk Amazon RDS tercantum di halaman berikut: Metode Amazon RDS. Pilih metode describe_db_instances dan gulir ke bawah ke bagian Struktur Respon. DbInstances terdaftar sebagai objek respon.

Jenis: Integer, Boolean, String,, StringList, StringMap atau MapList

Diperlukan: Ya

Jenis

Jenis data untuk elemen respon.

Jenis: Bervariasi

Diperlukan: Ya

`aws:executeScript` – Jalankan skrip

Menjalankan Python atau PowerShell skrip yang disediakan menggunakan runtime dan handler yang ditentukan. Setiap `aws:executeScript` tindakan dapat menjalankan hingga durasi maksimum 600 detik (10 menit). Anda dapat membatasi batas waktu dengan menentukan `timeoutSeconds` parameter untuk `aws:executeScript` langkah.

Gunakan pernyataan pengembalian dalam fungsi Anda untuk menambahkan output ke payload keluaran Anda. Untuk contoh mendefinisikan output untuk `aws:executeScript` tindakan Anda, lihat. Contoh 2: Skrip runbook Anda juga dapat mengirim output dari `aws:executeScript` tindakan di runbook ke grup CloudWatch log Amazon Logs yang Anda tentukan. Untuk informasi selengkapnya, lihat Pencatatan output tindakan Otomatisasi dengan CloudWatch Logs.

Jika Anda ingin mengirim output dari `aws:executeScript` tindakan ke CloudWatch Log, atau jika skrip yang Anda tentukan untuk `aws:executeScript` tindakan memanggil operasi AWS API, peran layanan AWS Identity and Access Management (IAM) (atau mengambil peran) selalu diperlukan untuk menjalankan runbook.

`aws:executeScript`Tindakan ini berisi modul PowerShell Core prainstal berikut:

- Microsoft. PowerShell.Tuan rumah
- Microsoft. PowerShell.Manajemen
- Microsoft. PowerShell.Keamanan
- Microsoft. PowerShell.Utilitas
- PackageManagement
- PowerShellGet

Untuk menggunakan modul PowerShell Core yang tidak diinstal sebelumnya, skrip Anda harus menginstal modul dengan `-Force` bendera, seperti yang ditunjukkan pada perintah berikut. `AWSPowerShell.NetCore`Modul tidak didukung. Ganti *`ModuleName`*dengan modul yang ingin Anda instal.

```
Install-Module ModuleName -Force
```

Untuk menggunakan cmdlet PowerShell Core dalam skrip Anda, sebaiknya gunakan `AWS.Tools` modul, seperti yang ditunjukkan pada perintah berikut. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri.

- cmdlet Amazon S3.

  ```
  Install-Module AWS.Tools.S3 -Force
  Get-S3Bucket -BucketName bucketname
  ```

- cmdlet Amazon EC2.

  ```
  Install-Module AWS.Tools.EC2 -Force
  Get-EC2InstanceStatus -InstanceId instanceId
  ```

- Umum, atau layanan independen AWS Tools for Windows PowerShell cmdlet.

  ```
  Install-Module AWS.Tools.Common -Force
  ```

```
Get-AWSRegion
```

Jika skrip Anda menginisialisasi objek baru selain menggunakan cmdlet PowerShell Core, Anda juga harus mengimpor modul seperti yang ditunjukkan pada perintah berikut.

```
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
```

Untuk contoh menginstal dan mengimpor `AWS.Tools` modul, dan menggunakan cmdlet PowerShell Core di runbook, lihat. Menggunakan Document Builder untuk membuat runbook

Input

Berikan informasi yang diperlukan untuk menjalankan skrip Anda. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri.

> **Note**
>
> Lampiran untuk skrip Python dapat berupa file.py atau file.zip yang berisi skrip. PowerShell skrip harus disimpan dalam file.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"
    }
}
```

Waktu Aktif

Bahasa runtime yang akan digunakan untuk menjalankan skrip yang disediakan. `aws:executeScript`mendukung Python 3.7 (python3.7), Python 3.8 (python3.8), Core 6.0 (dotnetcore2.1), dan 7.0 (dotnetcore3.1) skrip. PowerShell PowerShell

Nilai yang didukung: **`python3.7`** | **`python3.8`** | **`PowerShell Core 6.0`** | **`PowerShell 7.0`**

Tipe: String

Wajib: Ya

Handler

Nama fungsi Anda. Anda harus memastikan fungsi yang didefinisikan dalam handler memiliki dua parameter, `events` dan `context`. PowerShell Runtime tidak mendukung parameter ini.

Jenis: String

Diperlukan: Ya (Python) | Tidak didukung () PowerShell

InputPayload

Objek JSON atau YAML yang akan diteruskan ke parameter handler pertama. Ini dapat digunakan untuk melewatkan data input ke script.

Jenis: String

Wajib: Tidak

Python

```
description: Tag an instance
schemaVersion: '0.3'
assumeRole: '{{AutomationAssumeRole}}'
parameters:
    AutomationAssumeRole:
        type: String
        description: '(Required) The Amazon Resource Name (ARN) of the IAM role
 that allows Automation to perform the actions on your behalf. If no role is
 specified, Systems Manager Automation uses your IAM permissions to operate this
 runbook.'
    InstanceId:
        type: String
        description: (Required) The ID of the EC2 instance you want to tag.
mainSteps:
  - name: tagInstance
    action: 'aws:executeScript'
    inputs:
        Runtime: "python3.8"
        Handler: tagInstance
        InputPayload:
            instanceId: '{{InstanceId}}'
        Script: |-
          def tagInstance(events,context):
            import boto3

            #Initialize client
            ec2 = boto3.client('ec2')
            instanceId = events['instanceId']
            tag = {
                "Key": "Env",
                "Value": "Example"
            }
            ec2.create_tags(
                Resources=[instanceId],
                Tags=[tag]
            )
```

PowerShell

```
description: Tag an instance
schemaVersion: '0.3'
```

```
assumeRole: '{{AutomationAssumeRole}}'
parameters:
    AutomationAssumeRole:
        type: String
        description: '(Required) The Amazon Resource Name (ARN) of the IAM role
 that allows Automation to perform the actions on your behalf. If no role is
 specified, Systems Manager Automation uses your IAM permissions to operate this
 runbook.'
    InstanceId:
        type: String
        description: (Required) The ID of the EC2 instance you want to tag.
mainSteps:
  - name: tagInstance
    action: 'aws:executeScript'
    inputs:
        Runtime: PowerShell 7.0
        InputPayload:
          instanceId: '{{InstanceId}}'
        Script: |-
          Install-Module AWS.Tools.EC2 -Force
          Import-Module AWS.Tools.EC2

          $input = $env:InputPayload | ConvertFrom-Json

          $tag = New-Object Amazon.EC2.Model.Tag
          $tag.Key = "Env"
          $tag.Value = "Example"

          New-EC2Tag -Resource $input.instanceId -Tag $tag
```

Skrip

Skrip tertanam yang ingin Anda jalankan selama otomatisasi.

Jenis: String

Wajib: Tidak (Python) | Ya () PowerShell

Lampiran

Nama file skrip mandiri atau file .zip yang dapat dijalankan oleh tindakan. Tentukan nilai yang sama dengan `Name` file lampiran dokumen yang Anda tentukan dalam parameter `Attachments` permintaan. Untuk informasi selengkapnya, lihat Lampiran di Referensi AWS Systems Manager API. Jika Anda menyediakan skrip menggunakan lampiran, Anda juga harus menentukan `files`

bagian di elemen tingkat atas runbook Anda. Untuk informasi selengkapnya, lihat Skema versi 0.3.

Guna menjalankan file untuk Python, gunakan `filename.method_name` format dalam `Handler`.

> **Note**
>
> Lampiran untuk skrip Python dapat berupa file.py atau file.zip yang berisi skrip. PowerShell skrip harus disimpan dalam file.zip.

Ketika menyertakan pustaka Python di lampiran anda, kami sarankan menambahkan sebuah `__init__.py` file kosong dalam setiap direktori modul. Hal ini mengizinkan Anda untuk mengimpor modul dari pustaka di lampiran dalam konten skrip Anda. Sebagai contoh: `from library import module`

Jenis: String

Wajib: Tidak

Output

Muatan

Representasi objek JSON dikembalikan oleh fungsi Anda. Hingga 100KB yang dikembalikan. Jika Anda menampilkan daftar, maksimal 100 item dikembalikan.

`aws:executeStateMachine` – Jalankan AWS Step Functions mesin status

Menjalankan AWS Step Functions mesin status.

Masukan

Tindakan ini mendukung sebagian besar parameter untuk operasi StartExecutionAPI Step Functions Functions.

Izin AWS Identity and Access Management (IAM) yang diperlukan

- `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

Amazon Resource Name (ARN) mesin status Step Functions.

Jenis: String

Wajib: Ya

nama

Nama eksekusi.

Jenis: String

Wajib: Tidak

input

Sebuah string yang berisi data input JSON untuk eksekusi.

Jenis: String

Wajib: Tidak

Output

Output berikut telah ditetapkan untuk tindakan ini.

ExecutionARN

ARN eksekusi.

Jenis: String

input

The string yang berisi data input JSON eksekusi. Kendala panjang berlaku untuk ukuran muatan, dan dinyatakan sebagai byte dalam pengkodean UTF-8..

Jenis: String

nama

Nama eksekusi.

Jenis: String

output

Output data JSON eksekusi. Kendala panjang berlaku untuk ukuran payload, dan dinyatakan sebagai byte dalam pengkodean UTF-8.

Jenis: String

StartDate

Tanggal eksekusi dimulai.

Jenis: String

stateMachineArn

ARN dari mesin menyatakan dieksekusi.

Jenis: String

status

Status eksekusi saat ini.

Jenis: String

StopDate

Jika eksekusi sudah berakhir, tanggal eksekusi berhenti.

Jenis: String

`aws:invokeWebhook`- Memanggil integrasi webhook Otomasi

Memanggil integrasi webhook Otomasi yang ditentukan. Untuk informasi lebih lanjut tentang pembuatan integrasi Otomatisasi, lihatMembuat integrasi webhook untuk Otomasi.

> **Note**
>
> Untuk menggunakan `aws:invokeWebhook` tindakan tersebut, peran pengguna atau layanan Anda harus mengizinkan tindakan berikut:
>
> - ssm: GetParameter
> - kms:Decrypt
>
> Izin untuk `Decrypt` operasi AWS Key Management Service (AWS KMS) hanya diperlukan jika Anda menggunakan kunci yang dikelola pelanggan untuk mengenkripsi parameter untuk integrasi Anda.

Input

Berikan informasi untuk integrasi Otomatisasi yang ingin Anda panggil.

YAML

```
action: "aws:invokeWebhook"
inputs:
 IntegrationName: "exampleIntegration"
 Body: "Request body"
```

JSON

```
{
    "action": "aws:invokeWebhook",
    "inputs": {
        "IntegrationName": "exampleIntegration",
        "Body": "Request body"
    }
}
```

IntegrationName

Nama integrasi Otomatisasi. Sebagai contoh, `exampleIntegration`. Integrasi yang Anda tentukan harus sudah ada.

Tipe: String

Wajib: Ya

Tubuh

Payload yang ingin Anda kirim saat integrasi webhook Anda dipanggil.

Tipe: String

Wajib: Tidak

Output

Response

Teks yang diterima dari respons penyedia webhook.

ResponseCode

Kode status HTTP diterima dari respons penyedia webhook.

`aws:invokeLambdaFunction` – Jalankan AWS Lambda fungsi

Menjalankan fungsi AWS Lambda yang ditentukan.

> **Note**
>
> Setiap `aws:invokeLambdaFunction` tindakan dapat menjalankan hingga durasi maksimum 300 detik (5 menit). Anda dapat membatasi batas waktu dengan menentukan `timeoutSeconds` parameter untuk `aws:invokeLambdaFunction` langkah.

Input

Tindakan ini mendukung parameter yang paling dijalankan untuk layanan Lambda. Untuk informasi selengkapnya, lihat Jalankan.

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

Nama fungsi Lambda. Fungsi ini harus ada.

Jenis: String

Wajib: Ya

Pengualifikasi

Versi fungsi atau alias.

Jenis: String

Wajib: Tidak

InvocationType

Jenis penanganan. Nilai default-nya adalah `RequestResponse`.

Jenis: String

Nilai yang valid: `Event` | `RequestResponse` | `DryRun`

Wajib: Tidak

LogType

Jika nilai default-nya adalah `Tail`, jenis penanganan harus berupa `RequestResponse`. Lambda mengembalikan 4 KB data log terakhir yang dihasilkan oleh fungsi Lambda Anda, base64-dikodekan.

Jenis: String

Nilai yang valid: `None` | `Tail`

Wajib: Tidak

ClientContext

Informasi khusus klien.

Wajib: Tidak

InputPayload

Objek YAKL atau JSON yang diteruskan ke parameter handler pertama. Anda dapat menggunakan input ini untuk meneruskan data ke fungsi. Input ini memberikan lebih banyak fleksibilitas dan dukungan daripada`Payload` input warisan. Jika Anda mendefinisikan keduanya`InputPayload` dan`Payload` untuk tindakan,`InputPayload` diutamakan dan`Payload` nilainya tidak digunakan.

Jenis: StringMap

Wajib: Tidak

Muatan

String JSON yang diteruskan ke parameter handler pertama. Ini dapat digunakan untuk melewatkan data input data ke fungsi. Sebaiknya gunakan`InputPayload` input untuk fungsionalitas tambahan.

Tipe: String

Wajib: Tidak

Output

StatusCode

Kode status HTTP.

FunctionError

Jika ada, ini menunjukkan kesalahan terjadi kesalahan saat menjalankan fungsi. Detil kesalahan disertakan dalam muatan respons respons respons.

LogResult

The base64-encoded mencatat penanganan fungsi Lambda. Catatan hadir hanya jika jenis penanganan adalah `RequestResponse`, dan log diminta.

Muatan

Representasi objek JSON dikembalikan oleh fungsi Lambda. Muatan hadir hanya jika jenis doa adalah `RequestResponse`. Hingga 200KB yang dikembalikan

Berikut ini adalah bagian dari`AWS-PatchInstanceWithRollback` runbook yang menunjukkan bagaimana referensi output dari`aws:invokeLambdaFunction` tindakan.

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— Ulangi langkah-langkah dalam otomatisasi

Tindakan ini mengulangi subset langkah dalam runbook otomatisasi. Anda dapat memilih loop `do while` atau `for each` gaya. Untuk membangun `do while` loop, gunakan parameter `LoopCondition` input. Untuk membangun `for each` loop, gunakan parameter `Iterators` dan `IteratorDataType` input. Saat menggunakan `aws:loop` tindakan, hanya tentukan parameter `Iterators` atau `LoopCondition` input. Jumlah maksimum iterasi adalah 100.

`onCancel`Properti hanya dapat didefinisikan untuk langkah-langkah yang didefinisikan dalam loop. `onCancel`Properti tidak didukung untuk `aws:loop` tindakan tersebut.

Contoh-contoh

Berikut ini adalah contoh bagaimana membangun berbagai jenis tindakan loop.

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
```

Input

Intputnya adalah sebagai berikut.

Iterator

Daftar item untuk langkah-langkah untuk mengulangi. Jumlah maksimum iterator adalah 100.

Jenis: StringList

Diperlukan: Tidak

IteratorDataType

Parameter opsional untuk menentukan tipe data dari file`Iterators`. Nilai untuk parameter ini dapat diberikan bersama dengan parameter `Iterators` input. Jika Anda tidak menentukan nilai untuk parameter ini dan`Iterators`, maka Anda harus menentukan nilai untuk `LoopCondition` parameter tersebut.

Jenis: String

Nilai yang valid: Boolean | Integer | String | StringMap

Default: String

Diperlukan: Tidak

LoopCondition

Terdiri dari kondisi a `Variable` dan operator untuk mengevaluasi. Jika Anda tidak menentukan nilai untuk parameter ini, maka Anda harus menentukan nilai untuk `Iterators` dan `IteratorDataType` parameter. Anda dapat menggunakan evaluasi operator yang kompleks dengan menggunakan kombinasi`And`,`Not`, dan `Or` operator. Kondisi ini dievaluasi setelah langkah-langkah dalam loop selesai. Jika kondisinya `true` dan `MaxIterations` nilainya belum tercapai, langkah-langkah dalam loop berjalan lagi. Kondisi operator adalah sebagai berikut:

Operasi String

- StringEquals
- EqualsIgnoreCase
- StartsWith
- EndsWith
- Berisi

Operasi numerik

- NumericEquals
- NumericGreater
- NumericLesser
- NumericGreaterOrEquals
- NumericLesser
- NumericLesserOrEquals

Operasi Boolean

- BooleanEquals

Jenis: StringMap

Diperlukan: Tidak

MaxIterations

Jumlah maksimum kali langkah-langkah dalam loop dijalankan. Setelah nilai yang ditentukan untuk input ini tercapai, loop berhenti berjalan bahkan jika `LoopCondition` masih `true` atau jika ada objek yang tersisa dalam `Iterators` parameter.

Jenis: Integer

Nilai yang valid: 1 - 100

Diperlukan: Tidak

Langkah-langkah

Daftar langkah-langkah untuk dijalankan dalam loop. Ini berfungsi seperti runbook bersarang. Dalam langkah-langkah ini Anda dapat mengakses nilai iterator saat ini untuk `for each` loop menggunakan sintaks`{{loopStepName.CurrentIteratorValue}}`. Anda juga dapat mengakses nilai integer dari iterasi saat ini untuk kedua jenis loop menggunakan sintaks. `{{loopStepName.CurrentIteration}}`

Jenis: Daftar langkah

Diperlukan: Ya

Output

CurrentIteration

Iterasi loop saat ini sebagai bilangan bulat. Nilai iterasi mulai dari 1.

Jenis: Integer

CurrentIteratorValue

Nilai iterator saat ini sebagai string. Output ini hanya ada dalam `for each` loop.

Jenis: String

`aws:pause` – Jeda otomatisasi

Tindakan ini menghentikan otomatisasi. Setelah dijeda, status otomasi adalah Menunggu. Untuk melanjutkan otomatisasi, gunakan SendAutomationSignalAPI dengan tipe`Resume` sinyal. Kami merekomendasikan penggunaan`aws:sleep` atau`aws:approve` tindakan untuk kontrol yang lebih terperinci dari alur kerja Anda.

Input

Intputnya adalah sebagai berikut.

YAML

```
name: pauseThis
action: aws:pause
inputs: {}
```

JSON

```
{
    "name": "pauseThis",
    "action": "aws:pause",
    "inputs": {}
}
```

Output

Tidak ada

`aws:runCommand` – Jalankan perintah pada instans terkelola

Menjalankan perintah yang ditentukan.

> **Note**
>
> Otomatisasi hanya mendukung output dari satu AWS Systems Manager Run Command tindakan. Runbook dapat menyertakan beberapa Run Command tindakan, tetapi output didukung hanya untuk satu tindakan pada satu waktu.

Input

Tindakan ini mendukung sebagian besar parameter perintah kirim. Untuk informasi lebih lanjut, lihat 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

Jika dokumen tipe Command dimiliki oleh Anda atau AWS, tentukan nama dokumen. Jika Anda menggunakan dokumen yang dibagikan dengan Anda oleh Akun AWS yang berbeda, tentukan Amazon Resource Name (ARN) dokumen. Untuk informasi selengkapnya tentang penggunaan dokumen bersama, lihat Menggunakan dokumen SSM bersama.

Jenis: String

Diperlukan: Ya

InstanceIds

ID instans tempat Anda ingin menjalankan perintah. Anda dapat menentukan maksimum 50 ID.

Anda juga dapat menggunakan parameter semu `{{RESOURCE_ID}}` di tempat ID instans untuk menjalankan perintah pada semua instans dalam grup target. Untuk informasi selengkapnya tentang parameter semu, lihatMenggunakan parameter semu saat mendaftarkan tugas jendela pemeliharaan.

Alternatif lain adalah untuk mengirim perintah ke armada instans dengan menggunakan `Targets` parameter. Parameter `Targets` menerima tag Amazon Elastic Compute Cloud (Amazon EC2). Untuk informasi selengkapnya tentang cara menggunakan `Targets` parameter, lihat Menjalankan perintah saat skala.

Jenis: StringList

Wajib: Tidak (Jika Anda tidak menentukan InstanceIds atau menggunakan parameter `{{RESOURCE_ID}}` semu, maka Anda harus menentukan `Targets` parameternya.)

Target

Array kriteria pencarian yang menargetkan instans dengan menggunakan kombinasi Nilai Kunci yang Anda tentukan. `Targets` diperlukan jika Anda tidak memberikan satu ID instans atau lebih

dalam panggilan. Untuk informasi selengkapnya tentang cara menggunakan `Targets` parameter, lihat Menjalankan perintah saat skala.

Jenis: MapList (Skema peta dalam daftar harus cocok dengan objek.) Untuk informasi lebih lanjut, lihat Target dalam AWS Systems Manager Referensi API.

Wajib: Tidak (Jika Anda tidak menentukan`Targets`, maka Anda harus menentukan InstanceIds atau menggunakan parameter `{{RESOURCE_ID}}` semu.)

Berikut adalah contohnya.

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"
            ]
        }
    }
}
```

Parameter

Parameter yang diperlukan dan opsional yang ditentukan dalam dokumen.

Jenis: Peta

Wajib: Tidak

CloudWatchOutputConfig

Opsi konfigurasi untuk mengirim output perintah ke Amazon CloudWatch Logs.
Untuk informasi selengkapnya tentang mengirim output perintah ke CloudWatch Log, lihatMengonfigurasiCloudWatch Log Amazon untukRun Command.

Jenis: StringMap (Skema peta harus cocok dengan objek. Untuk informasi selengkapnya, lihat CloudWatchOutputConfigdi Referensi AWS Systems Manager API).

Diperlukan: Tidak

Berikut adalah contohnya.

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
        }
    }
}
```

Komentar

Informasi yang ditetapkan pengguna tentang perintah.

Jenis: String

Wajib: Tidak

DocumentHash

Hash untuk dokumen.

Jenis: String

Wajib: Tidak

DocumentHashType

Jenis hash.

Jenis: String

Nilai yang valid: `Sha256` | `Sha1`

Diperlukan: Tidak

NotificationConfig

Konfigurasi untuk mengirim notifikasi.

Diperlukan: Tidak

Keluaran3 BucketName

Nama bucket S3 untuk tanggapan output perintah.

Jenis: String

Wajib: Tidak

Keluaran3 KeyPrefix

Prefiks.

Jenis: String

Wajib: Tidak

ServiceRoleArn

ARN dari peran AWS Identity and Access Management (IAM).

Tipe: String

Wajib: Tidak

TimeoutSeconds

Jumlah waktu dalam hitungan detik untuk menunggu perintah dikirimkan ke AWS Systems Manager SSM Agent pada sebuah instance. Jika perintah tidak diterima oleh SSM Agent pada instance sebelum nilai yang ditentukan tercapai, maka status perintah berubah menjadi`Delivery Timed Out`.

Tipe: Integer

Wajib: Tidak

Nilai yang valid: 30-2592000

Output

CommandId

ID perintah.

Status

Status perintah.

ResponseCode

Kode respon perintah. Jika dokumen yang Anda jalankan memiliki lebih dari 1 langkah, nilai tidak dikembalikan untuk output ini.

Output

Output perintah.

aws:runInstances – Luncurkan Instans Amazon EC2

Luncurkan Amazon Elastic Compute Cloud (Amazon EC2) yang baru.

Input

Tindakan ini mendukung sebagian besar parameter API. Untuk informasi selengkapnya, lihat dokumentasi RunInstancesAPI.

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

Dicadangkan.

Jenis: String

Wajib: Tidak

BlockDeviceMappings

Perangkat blok untuk instans.

Jenis: MapList

Wajib: Tidak

ClientToken

Pengenal untuk memastikan idempotensi permintaan.

Jenis: String

Wajib: Tidak

DisableApiTermination

Menghidupkan atau mematikan penghentian API instans.

Jenis: Boolean

Wajib: Tidak

EbsOptimized

Menghidupkan atau mematikan optimisasi Amazon Elastic Block Store (Amazon EBS).

Jenis: Boolean

Wajib: Tidak

IamInstanceProfileArn

Amazon Resource Name (ARN) AWS Identity and Access Management profil instans (IAM) untuk instans.

Jenis: String

Wajib: Tidak

IamInstanceProfileName

Nama profil instans (IAM) untuk instans.

Jenis: String

Wajib: Tidak

ImageId

ID Amazon Machine Image (AMI).

Jenis: String

Wajib: Ya

InstanceInitiatedShutdownBehavior

Menunjukkan apakah instans berhenti atau berakhir pada sistem shutdown.

Jenis: String

Wajib: Tidak

InstanceType

Jenis instance.

> **Note**
>
> Jika nilai jenis instans tidak disediakan, jenis instans m1.small digunakan.

Jenis: String

Wajib: Tidak

KernelId

ID kernel.

Jenis: String

Wajib: Tidak

KeyName

Nama pasangan kunci.

Jenis: String

Wajib: Tidak

MaxInstanceCount

Jumlah instans maksimum untuk diluncurkan.

Jenis: String

Wajib: Tidak

MetadataOptions

Opsi metadata untuk instans. Untuk informasi lebih lanjut, lihat InstanceMetadataOptionsRequest.

Jenis: StringMap

Wajib: Tidak

MinInstanceCount

Jumlah instans minimum untuk diluncurkan.

Jenis: String

Wajib: Tidak

Pemantauan

Menghidupkan atau mematikan pemantauan terperinci.

Jenis: Boolean

Wajib: Tidak

NetworkInterfaces

Antarmuka jaringan.

Jenis: MapList

Wajib: Tidak

Penempatan

Penempatan untuk instans.

Jenis: StringMap

Wajib: Tidak

PrivateIpAddress

Primer alamat penyuratan IPv4.

Jenis: String

Wajib: Tidak

RamdiskId

ID disk RAM.

Jenis: String

Wajib: Tidak

SecurityGroupIds

ID grup keamanan untuk instans.

Jenis: StringList

Wajib: Tidak

SecurityGroups

Nama grup keamanan untuk instans.

Jenis: StringList

Wajib: Tidak

SubnetId

ID subnet.

Jenis: String

Wajib: Tidak

TagSpecifications

Tag untuk diterapkan ke sumber daya selama peluncuran. Anda hanya dapat menandai instans dan volume saat peluncuran. Tag tertentu diterapkan untuk semua instans atau volume yang dibuat selama peluncuran. Untuk menandai instans setelah diluncurkan, gunakan `aws:createTags – Buat tag untuk AWS sumber daya` tindakan.

Tipe: MapList (Untuk informasi lebih lanjut, lihat TagSpecification.)

Wajib: Tidak

UserData

Sebuah skrip yang disediakan sebagai nilai literal string. Jika nilai literal dimasukkan, maka harus Base64-encoded.

Jenis: String

Wajib: Tidak

Output

InstanceIds

ID instans.

InstanceStates

Status tabel saat ini ().

aws:sleep – Menunda otomatisasi

Menunda otomatisasi selama beberapa waktu tertentu. Tindakan ini menggunakan format tanggal dan waktu Organisasi Internasional untuk Standardisasi (ISO) 8601. Untuk informasi selengkapnya tentang format gal dan waktu, lihat ISO 8601.

Input

Anda dapat menunda otomatisasi untuk durasi tertentu.

YAML

```
name: sleep
action: aws:sleep
inputs:
  Duration: PT10M
```

JSON

```
{
```

```
   "name":"sleep",
   "action":"aws:sleep",
   "inputs":{
      "Duration":"PT10M"
   }
}
```

Anda juga dapat menunda otomatisasi sampai tanggal dan waktu yang ditentukan. Jika tanggal dan waktu yang ditentukan telah berlalu, tindakan akan segera berlanjut.

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**
>
> Otomatisasi mendukung penundaan maksimum 604799 detik (7 hari).

Durasi

Durasi ISO 8601. Anda tidak dapat menentukan durasi negatif.

Jenis: Tali

Diperlukan: Tidak

Timestamp

Sebuah cap waktu ISO 8601. Jika Anda tidak menentukan nilai untuk parameter ini, Anda harus menentukan nilai untuk `Duration` parameter tersebut.

Jenis: Tali

Diperlukan: Tidak

Output

Tidak ada

`aws:updateVariable`— Memperbarui nilai untuk variabel runbook

Tindakan ini memperbarui nilai untuk variabel runbook. Tipe data dari nilai harus sesuai dengan tipe data variabel yang ingin Anda perbarui. Konversi tipe data tidak didukung. `onCancel`Properti tidak didukung untuk `aws:updateVariable` tindakan tersebut.

Input

Intputnya adalah sebagai berikut.

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"]
    }
}
```

Nama

Nama variabel yang nilainya ingin Anda perbarui. Anda harus menggunakan format `variable:`*`variable name`*

Tipe: String

Diperlukan: Ya

Nilai

Nilai baru untuk menetapkan ke variabel. Nilai harus sesuai dengan tipe data variabel. Konversi tipe data tidak didukung.

Jenis: Boolean | Integer | | String MapList | | StringList StringMap

Diperlukan: Ya

Batasan:

- MapList dapat berisi jumlah maksimum 200 item.
- Panjang kunci bisa menjadi panjang minimum 1 dan panjang maksimum 50.
- StringList dapat berupa jumlah minimum 0 item dan jumlah maksimum 50 item.
- Panjang string bisa menjadi panjang minimum 1 dan panjang maksimum 512.

Output

Tidak ada

`aws:waitForAwsResourceProperty` – Tunggu di AWS properti sumber daya

Tindakan `aws:waitForAwsResourceProperty` tersebut mengizinkan otomatisasi Anda untuk menunggu status sumber daya tertentu atau status peristiwa sebelum melanjutkan otomatisasi. Untuk contoh lebih lanjut tentang cara menggunakan tindakan ini, lihatContoh runbook tambahan.

> **Note**
>
> Nilai batas waktu default untuk tindakan ini adalah 3600 detik (satu jam). Anda dapat membatasi atau memperpanjang batas waktu dengan menentukan `timeoutSeconds` parameter untuk `aws:waitForAwsResourceProperty` langkah. Untuk informasi selengkapnya dan contoh tentang cara menggunakan tindakan ini, lihat Menangani waktu habis di runbook.

Input

Input didefinisikan oleh operasi API yang Anda pilih.

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"
    ]
  }
}
```

Layanan

Layanan AWSNamespace yang berisi operasi API yang ingin Anda jalankan. Sebagai contoh, namespace untuk AWS Systems Manager adalah `ssm`. Namespace untuk Amazon Elastic Compute Cloud (Amazon EC2) adalah `ec2`. Anda dapat melihat daftar Layanan AWS ruang nama yang didukung di bagian Layanan yang Tersedia pada Referensi AWS CLI Perintah.

Tipe: String

Wajib: Ya

Api

Nama operasi API yang ingin Anda jalankan. Anda dapat melihat operasi API (juga disebut metode) dengan memilih layanan di navigasi kiri pada halaman Referensi Layanan. Pilih metode di bagian Klien untuk layanan yang ingin Anda jalankan. Misalnya, semua operasi API (metode) untuk Amazon Relational Database Service (Amazon RDS) tercantum di halaman berikut: Metode Amazon RDS.

Jenis: String

Wajib: Ya

Input operasi API

Satu input operasi API atau lebih. Anda dapat melihat input yang tersediaI (dikenal dengan parameter) dengan memilih layanan di navigasi kiri pada halaman Referensi Layanan berikut. Pilih metode di bagian Klien untuk layanan yang ingin Anda jalankan. Misalnya, semua metode untuk Amazon RDS tercantum di halaman berikut: Metode Amazon RDS. Pilih metode describe_db_instances dan gulir ke bawah untuk melihat parameter yang tersedia, seperti DB InstanceIdentifier, Name, dan 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"
}
```

Jenis: Ditentukan oleh operasi API yang dipilih

Wajib: Ya

PropertySelector

JsonPath untuk atribut tertentu dalam objek respon. Anda dapat melihat obyek respon dengan memilih layanan di navigasi kiri pada halaman Referensi Layanan berikut. Pilih metode di bagian Klien untuk layanan yang ingin Anda jalankan. Misalnya, semua metode untuk Amazon RDS tercantum di halaman berikut: Metode Amazon RDS. Pilih metode describe_db_instances dan gulir ke bawah ke bagian Struktur Respon. DbInstances terdaftar sebagai objek respon.

Jenis: String

Wajib: Ya

DesiredValues

Status yang diharapkan untuk melanjutkan otomatisasi.

Jenis:MapList, StringList

Wajib: Ya

Variabel sistem Otomatisasi

AWS Systems Manager Runbook otomatisasi menggunakan variabel berikut. Untuk contoh cara penggunaan variabel-variabel ini digunakan, lihat sumber JSON `AWS-UpdateWindowsAmi` runbook.

Untuk melihat sumber JSON **`AWS-UpdateWindowsAmi`** runbook

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.
3. Dalam daftar dokumen, gunakan bilah Pencarian atau angka di sebelah kanan bilah Pencarian untuk memilih buku runbook **`AWS-UpdateWindowsAmi`**.

4. Pilih tab Daftar Isi.

Variabel sistem

Runbook otomatisasi mendukung variabel berikut.

Variabel	Detail
`global:ACCOUNT_ID`	Akun AWSID pengguna atau peran lokasi otomatisasi berjalan.
`global:DATE`	Tanggal (pada waktu aktif) dalam format yyyy-MM-dd.
`global:DATE_TIME`	Tanggal dan waktu (pada waktu aktif) dalam format yyyy-MM-dd_HH.mm.ss.
`global:AWS_PARTITION`	Partisi tempat sumber daya berada. Untuk standard Wilayah AWS, partisi-nya adalah `aws`. Jika Anda memiliki sumber daya di partisi lain, maka partisi-nya adalah `aws-`*`partition name`* . Sebagai contoh, partisi untuk sumber daya di Wilayah AWS GovCloud (US-West). `aws-us-gov`
`global:REGION`	Wilayah tempat runbook dijalankan. Misalnya, us–east-2.

Variabel otomatisasi

Runbook otomatisasi mendukung variabel otomatisasi berikut.

Variabel	Detail
`automation:EXECUTION_ID`	Pengenal unik yang ditugaskan untuk otomatisasi saat ini. Sebagai contoh, `1a2b3c-1a2b3c-1a2b3c-1a2b3c 1a2b3c1a2b3c` .

Topik

Terminologi

Istilah berikut menjelaskan cara menyelesaikan variabel dan parameter.

Jangka waktu	Definisi	Contoh
ARN Konstan	Amazon Resource Name (ARN) yang valid tanpa variabel.	`arn:aws:iam::123456789012:role/roleName`
Parameter runbook	Parameter yang didefinisikan pada tingkat runbook (misalnya, `instanceId`). Parameter tersebut digunakan dalam pengganti string dasar. Nilainya diberikan pada saat Memulai Eksekusi.	`{ "description": "Create Image Demo", "version": "0.3", "assumeRole": "Your_Automation_Assume_Role_ARN ", "parameters":{ "instanceId": { "type": "String", "description": "Instance to create image from" } }`
Variabel sistem	Variabel umum diganti ke runbook ketika setiap bagian runbook dievaluasi.	`"activities": [ { "id": "copyImage", "activityType": "AWS-CopyImage", "maxAttempts": 1,`

Jangka waktu	Definisi	Contoh
		"onFailure": "Continue", "inputs": { "ImageName": "{{imageName}}", "SourceImageId": "{{sourceImageId}}", "SourceRegion": "{{sourceRegion}}", "Encrypted": true, "ImageDescription": "Test CopyImage Description created on **{{global:DATE}}** " } }]

Jangka waktu	Definisi	Contoh
Variasi otomatis	Variabel yang berkaitan dengan otomatisasi diganti ke runbook ketika setiap bagian runbook dievaluasi.	<pre>{ "name": "runFixed Cmds", "action": "aws:runC ommand", "maxAttempts": 1, "onFailure": "Continue", "inputs": { "DocumentName": "AWS-RunPowerShell Script", "InstanceIds": ["{{Launch Instance.InstanceI ds}}"], "Parameters": { "commands": ["dir", "date", "“{{outpu tFormat}}” -f “left”,”r ight”,”{{global:DA TE}}”,” **{{automat ion:EXECUTION_ID}}** ”] } } }</pre>

Jangka waktu	Definisi	Contoh
Parameter Systems Manager	Sebuah variabel yang didefinisikan dalam AWS Systems ManagerParameter Store. Hal ini tidak dapat langsung direferensikan dalam langkah masukan. Izin mungkin diperlukan untuk mengakses parameter.	`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`

Jangka waktu	Definisi	Contoh
		timeoutSeconds: 1200 onFailure: Abort inputs: ImageId: '{{LatestAmi}}' ...

Skenario yang didukung

Skenario	Komentar	Contoh
ARN Konstan `assumeRole` saat penciptaan.	Pemeriksaan otorisasi dilakukan untuk memverifikasi bahwa pengguna panggilan diizinkan untuk melewati diberikan `assumeRole` .	{ "description": "Test all Automation resolvable parameters", "schemaVersion": "0.3", "assumeRole": ***"arn:aws:iam::123456789012:role/roleName"*** , "parameters": { ...
Parameter Runbook disediakan untuk `AssumeRole` saat otomasi dimulai.	Harus didefinisikan dalam daftar parameter runbook.	{ "description": "Test all Automation resolvable parameters", "schemaVersion": "0.3", "assumeRole": **"{{dynamicARN}}"** , "parameters": { ...

Skenario	Komentar	Contoh
Nilai yang disediakan untuk parameter runbook di awal.	Pelanggan memasok nilai yang akan digunakan untuk parameter. Setiap input yang diberikan pada waktu mulai harus didefinisikan dalam daftar parameter runbook.	<pre>... "parameters": { "amiId": { "type": "String", "default": "ami-12345678 ", "description": "list of commands to run as part of first step" }, ...</pre> Masukan untuk Memulai Eksekusi Otomatisasi meliputi : {"amiId" : ["ami-12345678 "] }

Skenario	Komentar	Contoh
Parameter Systems Manager direferensikan dalam konten runbook.	Variabel ada dalam akun pelanggan, atau merupakan parameter yang dapat diakses publik, dan `AssumeRole` untuk runbook memiliki akses ke variabel. Pemeriksaan dilakukan pada waktu membuat untuk mengonfirmasi `AssumeRole` memiliki akses. Parameter tidak dapat langsung direferensikan dalam langkah input.	<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>

Skenario	Komentar	Contoh
Variabel sistem direferensikan dalam definisi langkah	Sebuah variabel sistem diganti ke runbook ketika otomatisasi dimulai. Nilai yang disuntikkan ke dalam runbook relatif saat substitusi terjadi. Artinya, nilai variabel waktu disuntikkan pada langkah 1 berbeda dari nilai variabel yang disuntikk an pada langkah 3 karena waktu yang dibutuhkan untuk menjalankan langkah-langkah di antaranya. Variabel sistem tidak perlu diatur dalam daftar parameter runbook.	<pre>... "mainSteps": [{ "name": "RunSomeC ommands", "action": "aws:runCommand", "maxAttempts": 1, "onFailure": "Continue", "inputs": { "DocumentName": "AWS:RunPowerShell", "InstanceIds": ["{{LaunchInstance .InstanceIds}}"], "Parameters": { "commands " : ["echo {The time is now {{global:DATE_TIME }}}"] } } }, ...</pre>

Skenario	Komentar	Contoh
Variabel otomatisasi direferensikan dalam definisi langkah.	Variabel otomatisasi tidak perlu diatur dalam daftar parameter runbook. Variabel Otomatisasi yang hanya didukung adalah otomatisasi:EXECUTION_ID.	<pre>... "mainSteps": [{ "name": "invokeLambdaFunction", "action": "aws:invokeLambdaFunction", "maxAttempts": 1, "onFailure": "Continue", "inputs": { "FunctionName": "Hello-World-LambdaFunction", "Payload" : "{ "executionId" : "{{automation:EXECUTION_ID}}" }" } } ...</pre>

Skenario	Komentar	Contoh
Lihat output langkah sebelumnya dalam definisi langkah berikutnya.	Ini adalah pengalihan parameter. Output langkah sebelumnya direferensikan menggunakan sintaks `{{stepName.OutputName}}` . Sintaks ini tidak dapat digunakan oleh pelanggan untuk parameter runbook. Hal ini teratasi saat langkah pengarah berjalan. Parameter tidak tercantum dalam parameter runbook.	<pre>... "mainSteps": [{ "name": "LaunchInstance", "action": "aws:runInstances", "maxAttempts": 1, "onFailure": "Continue", "inputs": { "ImageId": "{{amiId}}", "MinInstanceCount": 1, "MaxInstanceCount": 2 } }, { "name":"changeState", "action": "aws:changeInstanceState", "maxAttempts": 1, "onFailure": "Continue", "inputs": { "InstanceIds": ["{{LaunchInstance.InstanceIds}}"], "DesiredState": "terminated" } } ...</pre>

Skenario tidak didukung

Skenario	Komentar	Contoh
Parameter Systems Manager disediakan untuk `assumeRole` saat membuat	Tidak didukung.	`...` `{` `  "description":` `"Test all Automation` `resolvable parameters",` `  "schemaVersion":` `"0.3",` `  "assumeRole":` `"{{ssm:administratorRoleARN}}  ",` `  "parameters": {` `...`
Parameter Systems Manager langsung direferensikan dalam langkah input.	Pengembalian `InvalidDocumentContent` pengecualian pada waktu pembuatan.	`...` `mainSteps:` `  - name: launchInstance` `    action: 'aws:runInstances'` `    maxAttempts: 3` `    timeoutSeconds: 1200` `    onFailure: Abort` `    inputs:` `      ImageId: '{{ssm:/aws/service/ami-windows-latest/Windows_Server-2016-English-Full-Base}}'` `...`

Skenario	Komentar	Contoh
Definisi langkah variabel	Definisi langkah dalam runbook dibangun oleh variabel.	... "mainSteps": [{ "name": "LaunchInstance", "action": "aws:runInstances", "**{{attemptModel}}** ": 1, "onFailure": "Continue", "inputs": { "ImageId": "*ami-12345678* ", "MinInstanceCount": 1, "MaxInstanceCount": 2 } ... User supplies input : { "attemptModel" : "**minAttempts** " }

Skenario	Komentar	Contoh
Parameter runbook referensi silang	Pengguna memasok parameter input pada waktu mulai, yang merupakan referensi ke parameter lain dalam runbook.	... "parameters": { "amiId": { "type": "String", "default": "ami-7f2e6015 ", "description": "list of commands to run as part of first step" }, "alternateAmiId": { "type": "String", "description": "The alternate AMI to try if this first fails". "default" : "{{amiId} }" }, ...

Skenario	Komentar	Contoh
Ekspansi multi level	Runbook mendefinisikan variabel yang mengevaluasi ke nama variabel. Ini berada dalam pembatas variabel (yaitu {{}}) dan diperluas ke nilai variabel/parameter tersebut.	... "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}}*** }}"] }

Skenario	Komentar	Contoh
		`` ... Note: The customer intention here would be to run a command of "echo {Hello world}" ```

Skenario	Komentar	Contoh
Referensi output langkah runbook yang merupakan jenis variabel yang berbeda	Pengguna mereferensikan output dari langkah runbook sebelumnya dalam langkah berikutnya. Output adalah jenis variabel yang tidak memenuhi persyaratan tindakan pada langkah berikutnya.	``` ... mainSteps: - name: getImageId action: aws:execu teAwsApi inputs: Service: ec2 Api: DescribeImages Filters: - Name: "name" Values: - "{{ImageName}}" outputs: - Name: ImageIdList Selector: "$.Images " Type: "StringList" - name: copyMyImages action: aws:copyImage maxAttempts: 3 onFailure: Abort inputs: SourceImageId: {{getImageId.Image IdList}} SourceRegion: ap- northeast-2 ImageName: Encrypted Copies of LAMP base AMI in ap- northeast-2 Encrypted: true ... Note: You must provide the type required by the Automation action. In this case, aws:copyI mage requires a "String" type variable but the preceding step ```

Skenario	Komentar	Contoh
		`outputs a "StringList" type variable.`

Membuat runbook Anda sendiri

Runbook Otomasi menentukan tindakan yang dilakukan Systems Manager pada instans terkelola dan AWS sumber daya lainnya saat otomatisasi berjalan. Otomasi adalah kemampuan AWS Systems Manager. Runbook berisi satu langkah atau lebih yang berjalan dalam urutan yang tepat. Setiap langkah dibangun di sekitar satu tindakan. Output satu langkah dapat digunakan sebagai masukan dalam langkah selanjutnya.

Proses menjalankan tindakan ini dan langkah-langkah mereka disebut Otomatisasi.

Jenis tindakan yang didukung untuk runbook memungkinkan Anda mengotomatiskan berbagai operasi di lingkungan Anda AWS . Misalnya, menggunakan tipe `executeScript` tindakan, Anda dapat menyematkan python atau PowerShell skrip langsung di runbook Anda. (Ketika Anda membuat runbook kustom, Anda dapat menambahkan skrip sebaris, atau melampirkannya dari bucket S3 atau dari mesin lokal Anda.) Anda dapat mengotomatiskan pengelolaan AWS CloudFormation sumber daya Anda dengan menggunakan jenis `createStack` dan `deleteStack` tindakan. Selain itu, dengan menggunakan tipe `executeAwsApi` tindakan, sebuah langkah dapat menjalankan operasi API apa pun Layanan AWS, termasuk membuat atau menghapus AWS sumber daya, memulai proses lain, memulai pemberitahuan, dan banyak lagi.

Untuk daftar 20 jenis tindakan yang mendukung otomatisasi, lihat Referensi tindakan Otomatisasi Systems Manager.

AWS Systems Manager Automation menyediakan beberapa runbook dengan langkah-langkah yang telah ditentukan sebelumnya yang dapat Anda gunakan untuk melakukan tugas umum seperti memulai ulang satu atau beberapa instans Amazon Elastic Compute Cloud (Amazon EC2) atau membuat (). Amazon Machine Image AMI Anda juga dapat membuat runbook Anda sendiri dan membagikannya dengan yang lain Akun AWS, atau menjadikannya publik untuk semua pengguna Otomasi.

Runbook ditulis menggunakan YAMAL atau JSON. Menggunakan Pembuat Dokumen di konsol Otomatisasi Systems Manager, namun, Anda dapat membuat runbook tanpa harus menulis dalama JSON atau YAML.

> **Important**
>
> Jika Anda menjalankan alur kerja otomatisasi yang menjalankan layanan lain dengan menggunakan AWS Identity and Access Management peran layanan (IAM), pastikan bahwa peran layanan harus dikonfigurasi dengan izin untuk menjalankan layanan tersebut. Persyaratan ini berlaku untuk semua AWS Runbook otomatisasi (`AWS-*` runbook) seperti `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup`, dan `AWS-RestartEC2Instance` runbook, untuk beberapa nama. Persyaratan ini juga berlaku untuk setiap runbook Otomasi kustom yang Anda buat yang memanggil orang lain Layanan AWS dengan menggunakan tindakan yang memanggil layanan lain. Misalnya, jika Anda menggunakan `aws:executeAwsApi`, `aws:createStack`, atau `aws:copyImage` tindakan, konfigurasikan peran layanan dengan izin untuk menjalankan layanan tersebut. Anda dapat memberikan izin kepada orang lain Layanan AWS dengan menambahkan kebijakan inline IAM ke peran tersebut. Untuk informasi selengkapnya, lihat (Opsional) Tambahkan kebijakan sebaris Otomasi atau kebijakan terkelola pelanggan untuk memanggil lainnya Layanan AWS.

Untuk informasi tentang tindakan yang dapat Anda tentukan di buku runbook, lihat Referensi tindakan Otomatisasi Systems Manager.

Untuk informasi tentang penggunaan AWS Toolkit for Visual Studio Code untuk membuat runbook, lihat Bekerja dengan dokumen Otomasi Systems Manager di Panduan AWS Toolkit for Visual Studio Code Pengguna.

Untuk informasi tentang menggunakan Pembuat Dokumen guna membuat runbook kustom, lihat Menggunakan Document Builder untuk membuat runbook.

Daftar Isi

Pengalaman desain visual untuk runbook Otomasi

AWS Systems ManagerOtomasi memberikan pengalaman desain visual kode rendah yang membantu Anda membuat runbook otomatisasi. Pengalaman desain visual menyediakan drag-and-drop antarmuka dengan opsi untuk menambahkan kode Anda sendiri sehingga Anda dapat membuat dan mengedit runbook dengan lebih mudah. Dengan pengalaman desain visual, Anda dapat melakukan hal berikut:

- Kontrol pernyataan bersyarat.
- Kontrol bagaimana input dan output disaring atau diubah untuk setiap tindakan.
- Konfigurasikan penanganan kesalahan.
- Prototipe runbook baru.
- Gunakan runbook prototipe Anda sebagai titik awal untuk pengembangan lokal dengan. AWS Toolkit for Visual Studio Code

Saat membuat atau mengedit runbook, Anda dapat mengakses pengalaman desain visual dari konsol Otomasi. Saat Anda membuat runbook, pengalaman desain visual memvalidasi pekerjaan Anda dan menghasilkan kode secara otomatis. Anda dapat meninjau kode yang dihasilkan, atau mengekspornya untuk pengembangan lokal. Setelah selesai, Anda dapat menyimpan runbook, menjalankannya, dan memeriksa hasilnya di konsol Otomasi Systems Manager.

Sebelum Anda memulai

Untuk menggunakan pengalaman desain visual, Anda memerlukanAkun AWS, dan kredensional yang memberikan izin yang benar untuk sumber daya apa pun yang ingin Anda gunakan.

Dalam pengalaman desain visual, Automation terintegrasi dengan Amazon CodeGuru Security untuk membantu Anda mendeteksi pelanggaran kebijakan keamanan dan kerentanan dalam skrip AndaPython. Untuk menggunakan fitur ini untuk `aws:executeScript` tindakan, kebijakan AWS Identity and Access Management (IAM) Anda harus menyertakan izin berikut:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "codeguru-security:CreateUploadUrl",
```

```
                "codeguru-security:CreateScan",
                "codeguru-security:GetScan",
                "codeguru-security:GetFindings"
            ]
        }
    ]
}
```

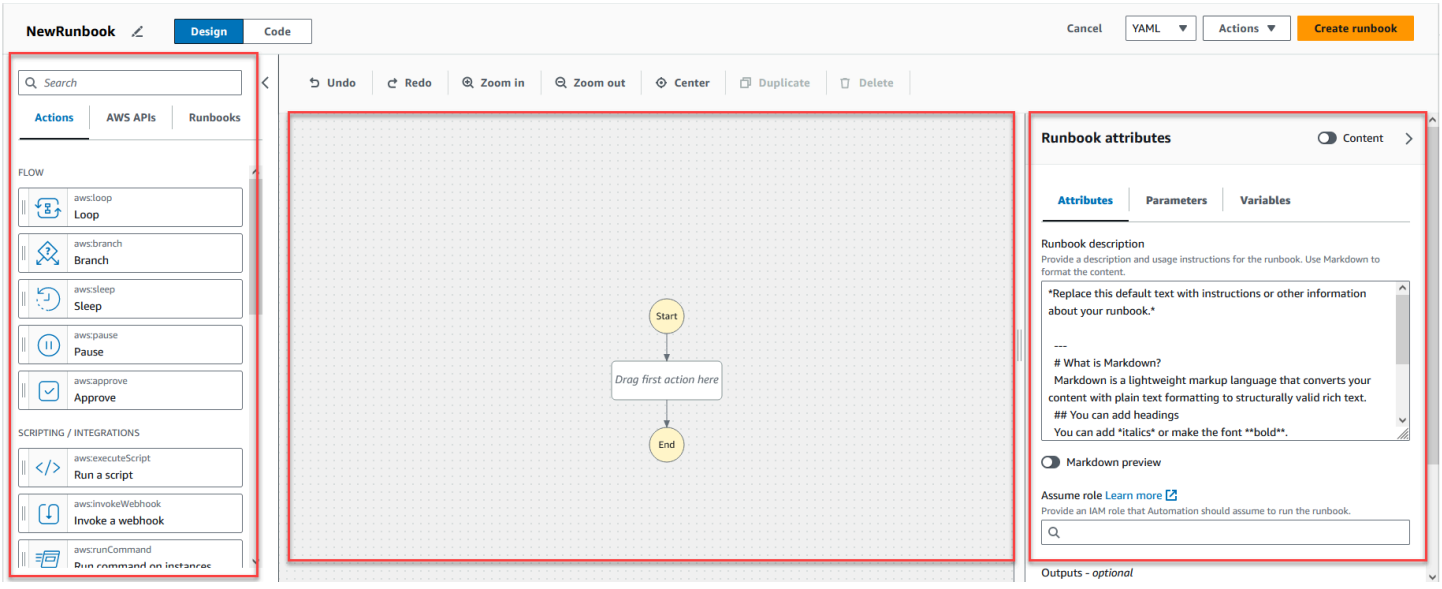
Topik

- Ikhtisar antarmuka pengalaman desain visual
- Menggunakan pengalaman desain visual
- Mengkonfigurasi input dan output untuk tindakan Anda
- Penanganan kesalahan dengan pengalaman desain visual
- Tutorial: Buat runbook menggunakan pengalaman desain visual

Ikhtisar antarmuka pengalaman desain visual

Pengalaman desain visual untuk Systems Manager Automation adalah desainer alur kerja visual kode rendah yang membantu Anda membuat runbook otomatisasi.

Kenali pengalaman desain visual dengan ikhtisar komponen antarmuka:



- Browser Actions berisi tab Actions, AWS API, dan Runbooks.
- Kanvas adalah tempat Anda menyeret dan melepaskan tindakan ke dalam grafik alur kerja Anda, mengubah urutan tindakan, dan memilih tindakan untuk dikonfigurasi atau dilihat.

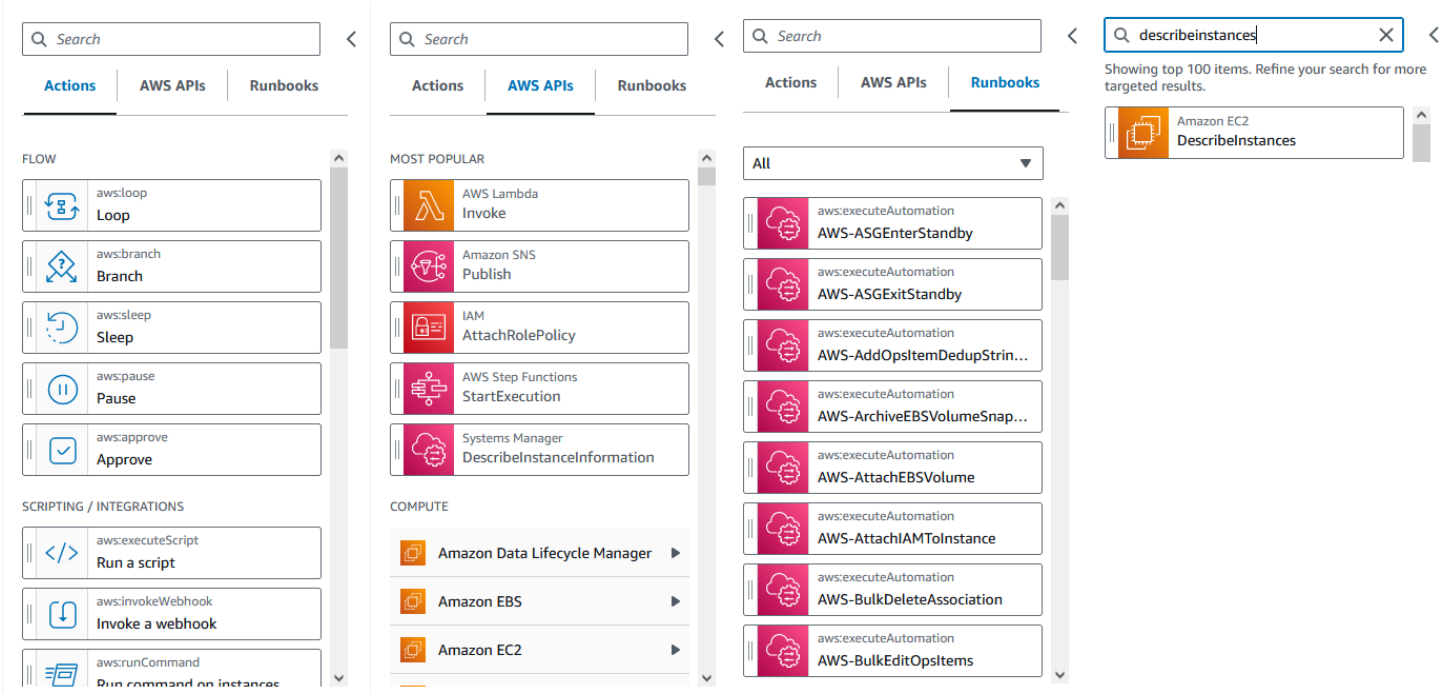
- Panel Formulir adalah tempat Anda dapat melihat dan mengedit properti tindakan apa pun yang Anda pilih di kanvas. Pilih sakelar Konten untuk melihat YAMAL atau JSON untuk runbook Anda, dengan tindakan yang dipilih saat ini disorot.

Tautan Info membuka panel dengan informasi kontekstual saat Anda memerlukan bantuan. Panel ini juga menyertakan tautan ke topik terkait dalam dokumentasi Otomasi Systems Manager.

Browser tindakan

Dari browser Tindakan, Anda dapat memilih tindakan untuk menyeret dan melepas ke grafik alur kerja Anda. Anda dapat mencari semua tindakan menggunakan bidang pencarian di bagian atas browser Tindakan. Browser Actions berisi tab berikut:

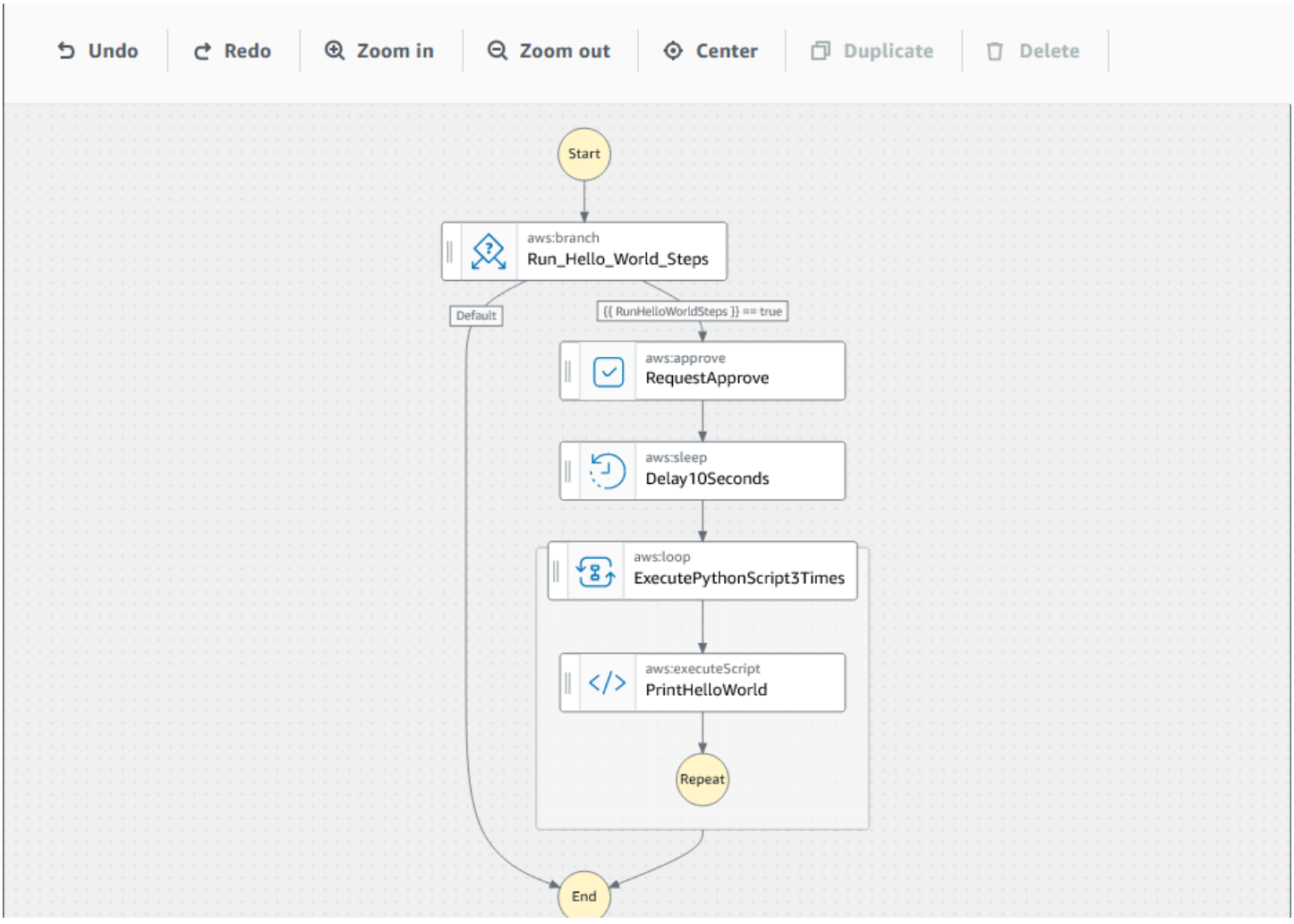
- Tab Tindakan menyediakan daftar tindakan otomatisasi yang dapat Anda seret dan lepas ke grafik alur kerja buku runbook Anda di kanvas.
- Tab AWS API menyediakan daftar AWS API yang dapat Anda seret dan lepas ke grafik alur kerja buku runbook di kanvas.
- Tab Runbooks menyediakan beberapa ready-to-use runbook yang dapat digunakan kembali sebagai blok bangunan yang dapat Anda gunakan untuk berbagai kasus penggunaan. Misalnya, Anda dapat menggunakan runbook untuk melakukan tugas remediasi umum di instans Amazon EC2 dalam alur kerja Anda tanpa harus membuat ulang tindakan yang sama.



Kanvas

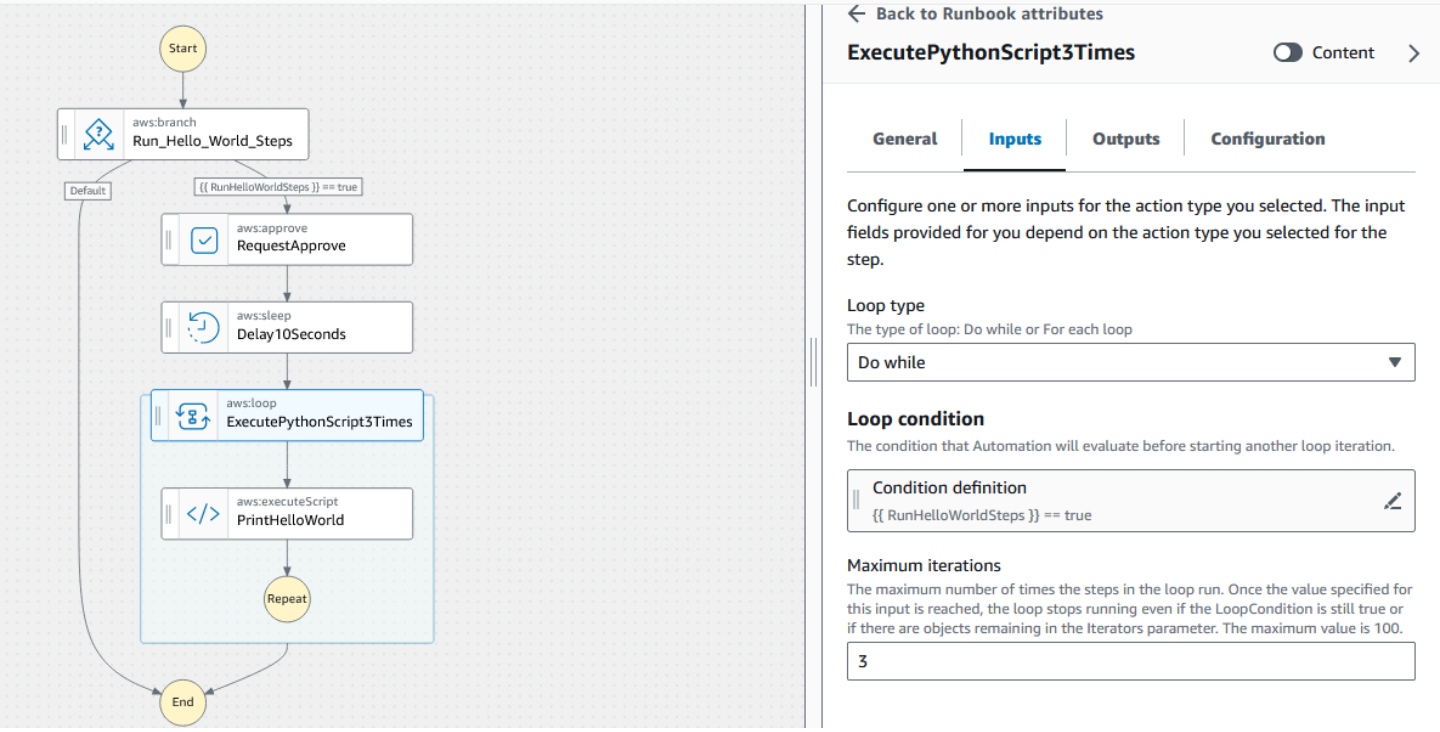
Setelah Anda memilih tindakan untuk ditambahkan ke otomatisasi Anda, seret ke kanvas dan jatuhkan ke grafik alur kerja Anda. Anda juga dapat menarik dan melepas tindakan untuk memindahkannya ke tempat yang berbeda dalam alur kerja runbook Anda. Jika alur kerja Anda rumit, Anda mungkin tidak dapat melihat semuanya di panel kanvas. Gunakan kendali di bagian atas kanvas untuk memperbesar atau memperkecil. Untuk melihat bagian alur kerja yang berbeda, Anda dapat menyeret grafik alur kerja di kanvas.

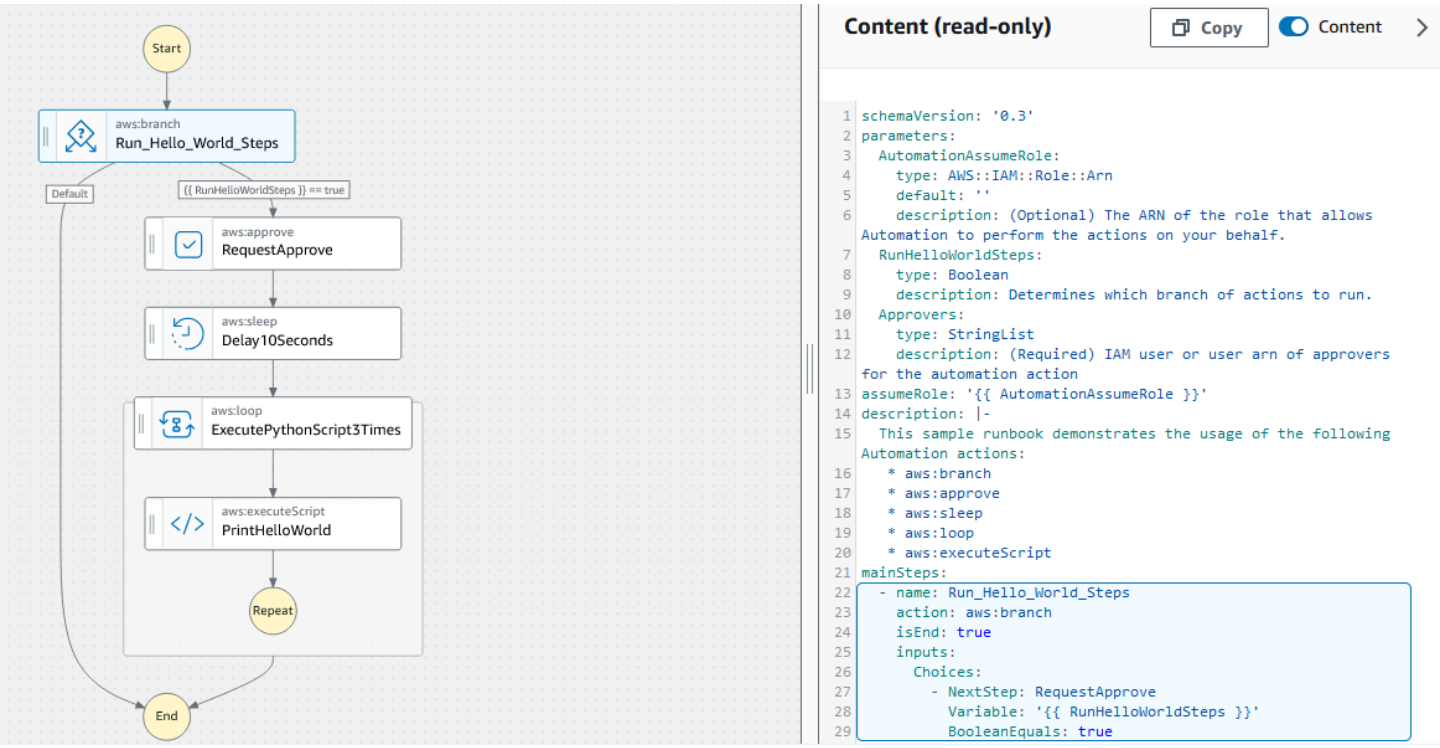
Seret tindakan dari browser Actions, dan masukkan ke dalam grafik alur kerja runbook Anda. Baris menunjukkan tempat status akan ditempatkan di alur kerja Anda. Untuk mengubah urutan tindakan, Anda dapat menyeretnya ke tempat lain di alur kerja Anda. Tindakan baru telah ditambahkan ke alur kerja Anda, dan kodenya dibuat secara otomatis.



Formulir

Setelah menambahkan tindakan ke alur kerja runbook, Anda dapat mengonfigurasinya untuk memenuhi kasus penggunaan. Pilih tindakan yang ingin Anda konfigurasikan, dan Anda akan melihat parameter dan opsinya di panel Formulir. Anda juga dapat melihat kode YAMAL atau JSON dengan memilih toggle Konten. Kode yang terkait dengan tindakan yang Anda pilih disorot.





Pintasan keyboard

Pengalaman desain visual mendukung pintasan keyboard yang ditunjukkan pada tabel berikut.

Pintasan keyboard	Tindakan
Ctrl+Z	Batalkan operasi terakhir.
Ctrl+Shift+Z	Ulangi operasi terakhir.
Alt+Q	Pusatkan fokus kerja di kanvas.

Pintasan keyboard	
Spasi	Memilih status yang dipilih.
Hapus	Hapus semua status yang dipilih.
Ctrl + D	Gandakan status yang dipilih.

Menggunakan pengalaman desain visual

Belajar membuat, mengedit, dan menjalankan alur kerja runbook menggunakan pengalaman desain visual. Setelah alur kerja Anda siap, Anda dapat menyimpannya atau mengekspornya. Anda juga dapat menggunakan pengalaman desain visual untuk pembuatan prototipe cepat.

Buat alur kerja runbook

1. Masuk ke konsol Otomasi Systems Manager.
2. Pilih Buat runbook.
3. Di kotak Nama, masukkan nama untuk runbook Anda, misalnya,*MyNewRunbook*.
4. Di sebelah sakelar Desain dan Kode, pilih ikon pensil dan masukkan nama untuk runbook Anda.

Anda sekarang dapat merancang alur kerja untuk runbook baru Anda.

Desain buku runbook

Untuk mendesain alur kerja runbook menggunakan pengalaman desain visual, Anda menyeret tindakan otomatisasi dari browser Actions ke kanvas, menempatkannya di tempat yang Anda inginkan dalam alur kerja runbook Anda. Anda juga dapat mengatur ulang tindakan dalam alur kerja Anda dengan menyeretnya ke lokasi yang berbeda. Saat Anda menyeret tindakan ke kanvas, sebuah garis muncul di mana pun Anda dapat melepaskan tindakan dalam alur kerja Anda. Setelah tindakan dijatuhkan ke kanvas, kodenya dibuat secara otomatis dan ditambahkan di dalam konten runbook Anda.

Jika Anda tahu nama tindakan yang ingin Anda tambahkan, gunakan kotak pencarian di bagian atas browser Tindakan untuk menemukan tindakan.

Setelah Anda menjatuhkan tindakan ke kanvas, konfigurasikan menggunakan panel Formulir di sebelah kanan. Panel ini berisi tab Umum, Input, Output, dan Konfigurasi untuk setiap tindakan otomatisasi atau tindakan API yang Anda tempatkan di kanvas. Misalnya, tab Umum terdiri dari bagian-bagian berikut:

- Nama Langkah mengidentifikasi langkah. Tentukan nilai unik untuk nama langkah.
- Deskripsi membantu Anda menjelaskan tindakan yang dilakukan dalam alur kerja runbook Anda.

Tab Input berisi bidang yang bervariasi berdasarkan tindakan. Misalnya, tindakan `aws:executeScript` otomatisasi terdiri dari bagian-bagian berikut:

- Runtime adalah bahasa yang digunakan untuk menjalankan skrip yang disediakan.
- Handler adalah nama fungsi Anda. Anda harus memastikan bahwa fungsi yang didefinisikan dalam handler memiliki dua parameter: `events` dan`context`. PowerShellRuntime tidak mendukung parameter ini.
- Script adalah skrip tertanam yang ingin Anda jalankan selama alur kerja.
- (Opsional) Lampiran adalah untuk skrip mandiri atau file.zip yang dapat dipanggil oleh tindakan. Parameter ini diperlukan untuk runbook JSON.

Tab Output membantu Anda menentukan nilai yang ingin Anda keluarkan dari suatu tindakan. Anda dapat mereferensikan nilai keluaran dalam tindakan selanjutnya dari alur kerja Anda, atau menghasilkan output dari tindakan untuk tujuan pencatatan. Tidak semua tindakan akan memiliki tab Output karena tidak semua tindakan mendukung output. Misalnya, `aws:pause` tindakan tidak mendukung output. Untuk tindakan yang mendukung output, tab Output terdiri dari bagian berikut:

- Nama adalah nama yang akan digunakan untuk nilai output. Anda dapat mereferensikan output dalam tindakan selanjutnya dari alur kerja Anda.
- Selector adalah string JSONPath ekspresi yang dimulai dengan `"$."` yang digunakan untuk memilih satu atau lebih komponen dalam elemen JSON.
- Tipe adalah tipe data untuk nilai output. Misalnya, tipe `Integer` data `String` atau.

Tab Konfigurasi berisi properti dan opsi yang dapat digunakan oleh semua tindakan otomatisasi. Tindakan ini terdiri dari bagian-bagian berikut:

- Properti upaya Max adalah berapa kali tindakan mencoba lagi jika gagal.
- Properti detik Timeout menentukan nilai batas waktu untuk tindakan.
- Properti Is critical menentukan apakah kegagalan tindakan menghentikan seluruh otomatisasi.
- Properti Next step menentukan tindakan otomatisasi berikutnya di runbook.
- Properti On failure menentukan tindakan otomatisasi berikutnya di runbook jika tindakan gagal.
- Properti On cancel menentukan tindakan otomatisasi berikutnya di runbook jika tindakan dibatalkan oleh pengguna.

Untuk menghapus tindakan, Anda dapat menggunakan backspace, toolbar di atas kanvas, atau klik kanan dan pilih Hapus tindakan.

Saat alur kerja Anda tumbuh, mungkin tidak muat di kanvas. Untuk membantu membuat alur kerja sesuai di kanvas, coba salah satu opsi berikut:

- Gunakan kontrol pada panel samping untuk mengubah ukuran atau menutup panel.
- Gunakan bilah alat di bagian atas kanvas untuk memperbesar grafik alur kerja masuk atau keluar.

Perbarui runbook Anda

Anda dapat memperbarui alur kerja runbook yang ada dengan membuat versi baru dari runbook Anda. Pembaruan untuk runbook Anda dapat dilakukan dengan menggunakan pengalaman desain visual, atau dengan mengedit kode secara langsung. Untuk memperbarui runbook yang ada, gunakan prosedur berikut:

1. Masuk ke konsol Otomasi Systems Manager.
2. Pilih runbook yang ingin Anda perbarui.

3. Pilih Buat versi baru.
4. Pengalaman desain visual memiliki dua panel: Panel kode dan panel alur kerja visual. Pilih Desain di panel alur kerja visual untuk mengedit alur kerja Anda dengan pengalaman desain visual. Setelah selesai, pilih Buat versi baru untuk menyimpan perubahan dan keluar.
5. (Opsional) Gunakan panel kode untuk mengedit konten runbook di YAMAL atau JSON.

Ekspor runbook Anda

Untuk mengekspor alur kerja kode YAMAL atau JSON runbook Anda, dan juga grafik alur kerja Anda, gunakan prosedur berikut:

1. Pilih runbook Anda di konsol Dokumen.
2. Pilih Buat versi baru.
3. Di dropdown Tindakan, pilih apakah Anda ingin mengekspor grafik atau runbook, dan format mana yang Anda inginkan.

Mengkonfigurasi input dan output untuk tindakan Anda

Setiap tindakan otomatisasi merespons berdasarkan masukan yang diterimanya. Dalam kebanyakan kasus, Anda kemudian meneruskan output ke tindakan selanjutnya. Dalam pengalaman desain visual, Anda dapat mengonfigurasi data input dan output tindakan di tab Input dan Output pada panel Formulir.

Untuk informasi rinci tentang cara mendefinisikan dan menggunakan output untuk tindakan otomatisasi, lihatMenggunakan output tindakan sebagai input.

Menyediakan data masukan untuk suatu tindakan

Setiap tindakan otomatisasi memiliki satu atau lebih input yang harus Anda berikan nilainya. Nilai yang Anda berikan untuk input tindakan ditentukan oleh tipe data dan format yang diterima oleh tindakan. Misalnya, `aws:sleep` tindakan memerlukan nilai string berformat ISO 8601 untuk input. `Duration`

Umumnya, Anda menggunakan tindakan dalam alur kerja runbook yang mengembalikan output yang ingin Anda gunakan dalam tindakan selanjutnya. Penting untuk memastikan nilai input Anda benar untuk menghindari kesalahan dalam alur kerja runbook Anda. Nilai input juga penting karena menentukan apakah tindakan mengembalikan output yang diharapkan. Misalnya, saat menggunakan

`aws:executeAwsApi` tindakan, Anda ingin memastikan bahwa Anda memberikan nilai yang tepat untuk operasi API.

Tentukan data keluaran untuk suatu tindakan

Beberapa tindakan otomatisasi mengembalikan output setelah melakukan operasi yang ditentukan. Tindakan yang mengembalikan output memiliki output yang telah ditentukan sebelumnya, atau memungkinkan Anda untuk menentukan output sendiri. Misalnya, `aws:createImage` tindakan memiliki output yang telah ditentukan sebelumnya yang `ImageId` mengembalikan dan. `ImageState` Secara komparatif, dengan `aws:executeAwsApi` tindakan, Anda dapat menentukan output yang Anda inginkan dari operasi API yang ditentukan. Akibatnya, Anda dapat mengembalikan satu atau beberapa nilai dari satu operasi API untuk digunakan dalam tindakan selanjutnya.

Mendefinisikan output Anda sendiri untuk tindakan otomatisasi mengharuskan Anda menentukan nama output, tipe data, dan nilai output. Untuk terus menggunakan `aws:executeAwsApi` tindakan sebagai contoh, katakanlah Anda memanggil operasi `DescribeInstances` API dari Amazon EC2. Dalam contoh ini, Anda ingin mengembalikan, atau mengeluarkan, instans Amazon EC2 dan cabang alur kerja runbook Anda berdasarkan output. `State` Anda memilih untuk memberi nama output**`InstanceState`**, dan menggunakan tipe **`String`** data.

Proses untuk menentukan nilai aktual dari output berbeda, tergantung pada tindakan. Misalnya, jika Anda menggunakan `aws:executeScript` tindakan, Anda harus menggunakan `return` pernyataan dalam fungsi Anda untuk memberikan data ke output Anda. Dengan tindakan lain seperti`aws:executeAwsApi`,`aws:waitForAwsResourceProperty`, dan`aws:assertAwsResourceProperty`, a `Selector` diperlukan. The`Selector`, atau `PropertySelector` sebagai beberapa tindakan merujuk padanya, adalah JSONPath string yang digunakan untuk memproses respons JSON dari operasi API. Penting untuk memahami bagaimana objek respons JSON dari operasi API terstruktur sehingga Anda dapat memilih nilai yang benar untuk output Anda. Menggunakan operasi `DescribeInstances` API yang disebutkan sebelumnya, lihat contoh respons JSON berikut:

```
{
  "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
          }
        }
      }
    }
  }
}
```

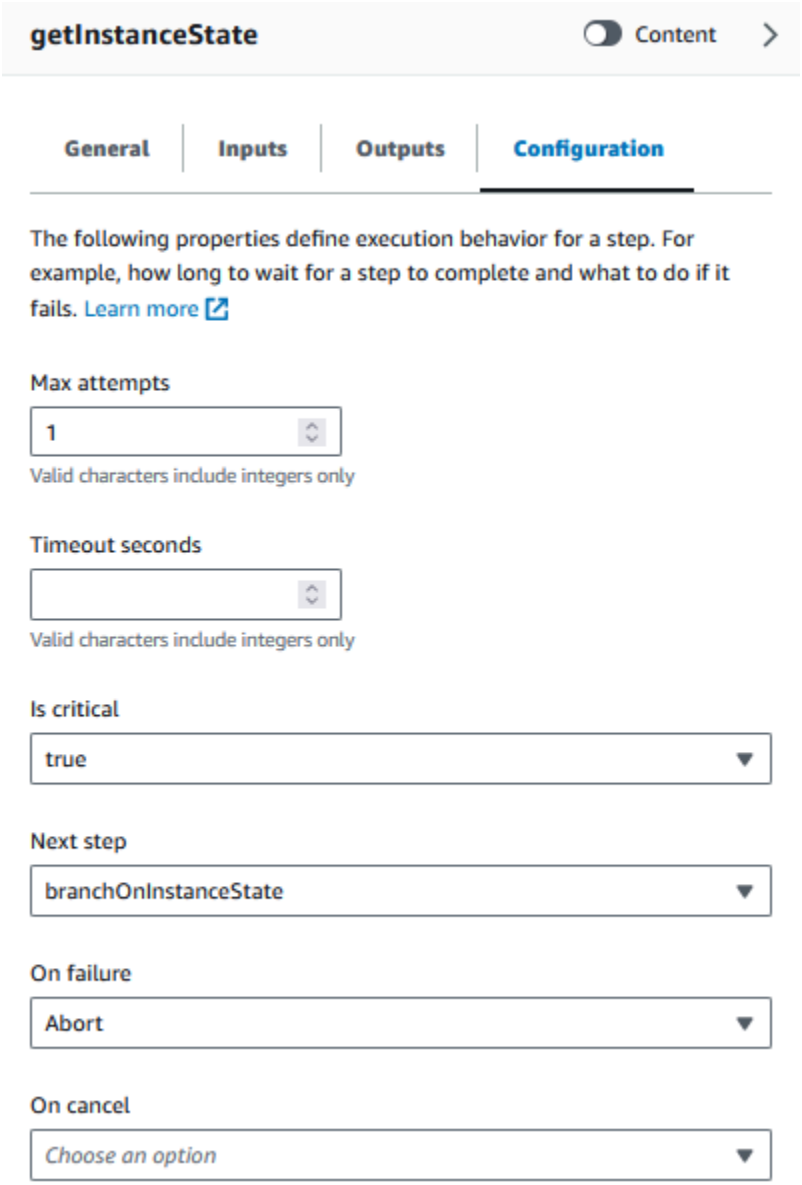
Dalam objek respons JSON, instance `State` bersarang di `Instances` objek, yang bersarang di objek. `Reservations` Untuk mengembalikan nilai instance`State`, gunakan string berikut untuk `Selector` sehingga nilai dapat digunakan dalam output kami:**`$.Reservations[0].Instances[0].State.Name`**.

Untuk mereferensikan nilai output dalam tindakan selanjutnya dari alur kerja runbook Anda, format berikut digunakan:. {{ *`StepName`*`.`*`NameOfOutput`* }} Sebagai contoh, **`{{ GetInstanceState.InstanceState }}`**. Dalam pengalaman desain visual, Anda dapat memilih nilai output untuk digunakan dalam tindakan selanjutnya menggunakan dropdown untuk input. Saat menggunakan output dalam tindakan selanjutnya, tipe data output harus cocok dengan

tipe data untuk input. Dalam contoh ini, `InstanceState` outputnya adalah a`String`. Oleh karena itu, untuk menggunakan nilai dalam input tindakan selanjutnya, input harus menerima a`String`.

Penanganan kesalahan dengan pengalaman desain visual

Secara default, saat tindakan melaporkan kesalahan, Automation menghentikan alur kerja runbook sepenuhnya. Ini karena nilai default untuk on`Failure` properti pada semua tindakan adalah`Abort`. Anda dapat mengonfigurasi cara Automation menangani kesalahan dalam alur kerja runbook Anda. Bahkan jika Anda telah mengonfigurasi penanganan kesalahan, beberapa kesalahan mungkin masih menyebabkan otomatisasi gagal. Untuk informasi selengkapnya, lihat Pemecahan masalah Otomatisasi Systems Manager. Dalam pengalaman desain visual, Anda mengonfigurasi penanganan kesalahan di panel Konfigurasi.



Coba lagi tindakan pada kesalahan

Untuk mencoba lagi tindakan jika terjadi kesalahan, tentukan nilai untuk properti Upaya Maks. Nilai default adalah 1. Jika Anda menentukan nilai yang lebih besar dari 1, tindakan tersebut tidak dianggap gagal sampai semua upaya percobaan ulang gagal.

Timeout

Anda dapat mengonfigurasi batas waktu untuk tindakan untuk mengatur jumlah maksimum detik tindakan yang dapat dijalankan sebelum gagal. Untuk mengonfigurasi batas waktu, masukkan jumlah detik yang harus ditunggu tindakan Anda sebelum tindakan gagal di properti detik Timeout. Jika batas waktu tercapai dan tindakan memiliki nilai `Max attempts` yang lebih besar dari 1, langkah tersebut tidak dianggap telah habis waktu sampai percobaan ulang selesai.

Tindakan yang gagal

Secara default, ketika suatu tindakan gagal, Automation menghentikan alur kerja runbook sepenuhnya. Anda dapat mengubah perilaku ini dengan menentukan nilai alternatif untuk properti On failure dari tindakan di buku runbook Anda. Jika Anda ingin alur kerja melanjutkan ke langkah berikutnya di runbook, pilih Lanjutkan. Jika Anda ingin alur kerja melompat ke langkah berikutnya yang berbeda di runbook, pilih Langkah lalu masukkan nama langkahnya.

Tindakan yang dibatalkan

Secara default, saat tindakan dibatalkan oleh pengguna, Automation menghentikan alur kerja runbook sepenuhnya. Anda dapat mengubah perilaku ini dengan menentukan nilai alternatif untuk properti On cancel dari tindakan di buku runbook Anda. Jika Anda ingin alur kerja melompat ke langkah berikutnya yang berbeda di runbook, pilih Langkah lalu masukkan nama langkahnya.

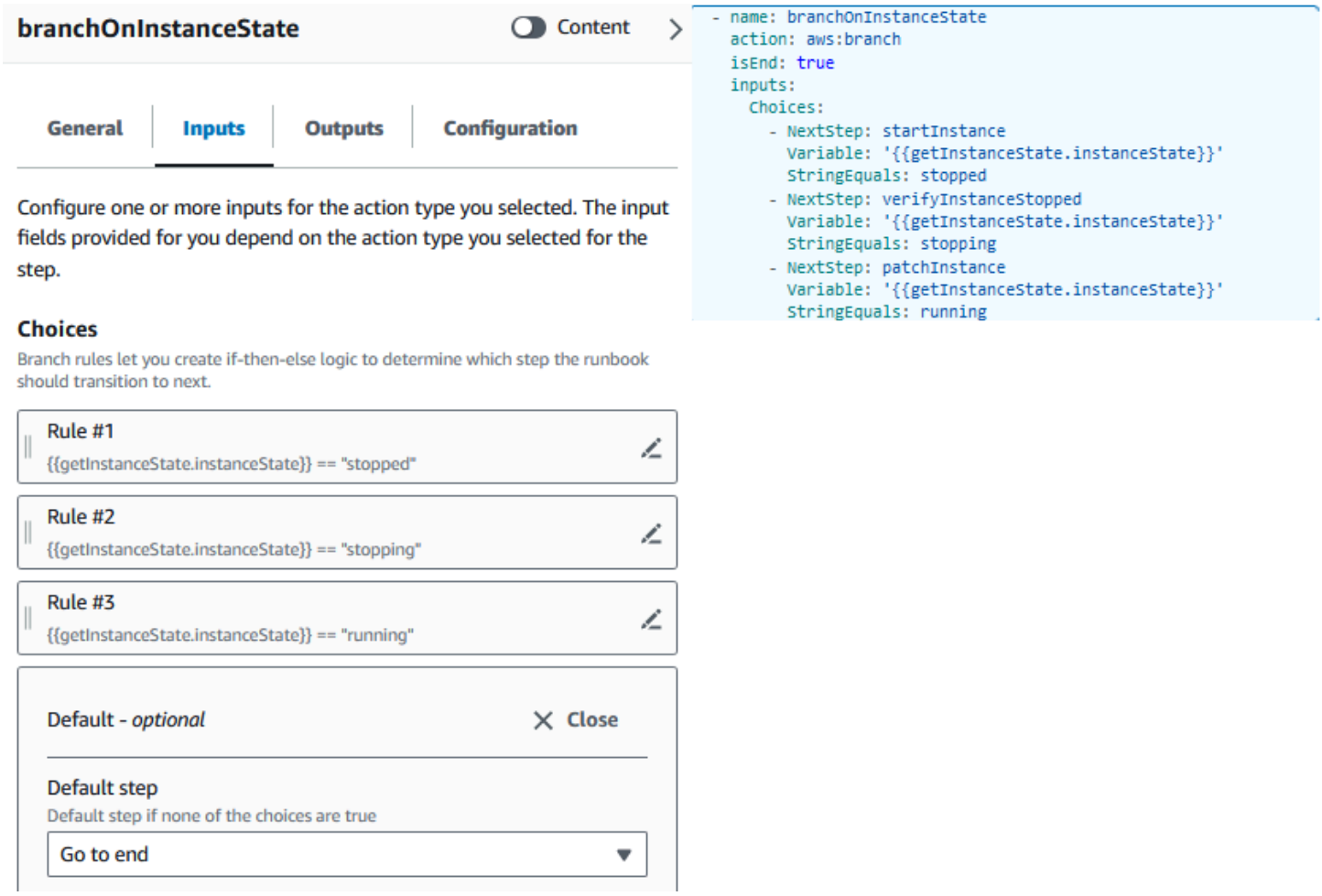
Tindakan kritis

Anda dapat menetapkan suatu tindakan sebagai tindakan penting, artinya tindakan tersebut menentukan status pelaporan keseluruhan otomatisasi Anda. Jika langkah dengan penunjukan ini gagal, Otomasi melaporkan status akhir sebagai `Failed` terlepas dari keberhasilan tindakan lain. Untuk mengonfigurasi tindakan sebagai kritis, biarkan nilai default sebagai True untuk properti Is critical.

Mengakhiri tindakan

Properti Is end menghentikan otomatisasi pada akhir tindakan yang ditentukan. Nilai default untuk properti ini adalah`false`. Jika Anda mengonfigurasi properti ini untuk suatu tindakan,

otomatisasi akan berhenti apakah tindakan berhasil atau gagal. Properti ini paling sering digunakan dengan `aws:branch` tindakan untuk menangani nilai input yang tidak terduga atau tidak ditentukan. Contoh berikut menunjukkan runbook yang mengharapkan status instance dari salah satu`running`,`stopping`, atau. `stopped` Jika sebuah instance berada dalam keadaan yang berbeda, otomatisasi berakhir.



Tutorial: Buat runbook menggunakan pengalaman desain visual

Dalam tutorial ini, Anda akan mempelajari dasar-dasar bekerja dengan pengalaman desain visual yang disediakan oleh Systems Manager Automation. Dalam pengalaman desain visual, Anda dapat membuat runbook yang menggunakan beberapa tindakan. Anda menggunakan fitur drag and drop untuk mengatur tindakan di kanvas. Anda juga mencari, memilih, dan mengonfigurasi tindakan ini. Kemudian, Anda dapat melihat kode YAMAL yang dibuat secara otomatis untuk alur kerja runbook Anda, keluar dari pengalaman desain visual, menjalankan runbook, dan meninjau detail eksekusi.

Tutorial ini juga menunjukkan cara memperbarui runbook dan melihat versi baru. Di akhir tutorial, Anda melakukan langkah pembersihan dan menghapus runbook Anda.

Setelah Anda menyelesaikan tutorial ini, Anda akan tahu cara menggunakan pengalaman desain visual untuk membuat runbook. Anda juga akan tahu cara memperbarui, menjalankan, dan menghapus runbook Anda.

> **Note**
>
> Sebelum Anda memulai tutorial ini, pastikan untuk menyelesaikannyaMenyiapkan Otomatisasi.

Topik

Langkah 1: Arahkan ke pengalaman desain visual

1. Masuk ke konsol Otomasi Systems Manager.
2. Pilih Buat runbook otomatisasi.

Langkah 2: Buat alur kerja

Dalam pengalaman desain visual, alur kerja adalah representasi grafis dari runbook Anda di kanvas. Anda dapat menggunakan pengalaman desain visual untuk menentukan, mengonfigurasi, dan memeriksa tindakan individual dari runbook Anda.

Untuk membuat alur kerja

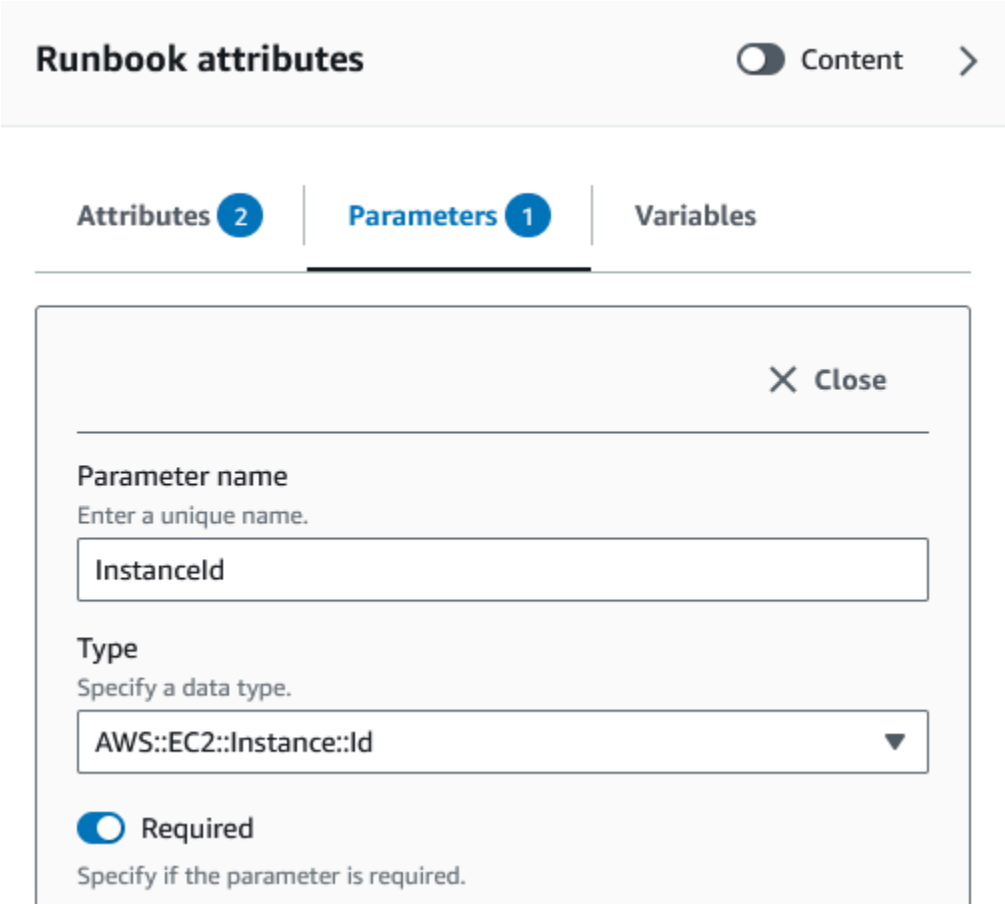
1. Di sebelah sakelar Desain dan Kode, pilih ikon pensil dan masukkan nama untuk runbook Anda. Untuk tutorial ini, masukkan **`VisualDesignExperienceTutorial`**.

2. Di bagian Atribut dokumen pada panel Formulir, perluas dropdown parameter Input, dan pilih Tambahkan parameter.

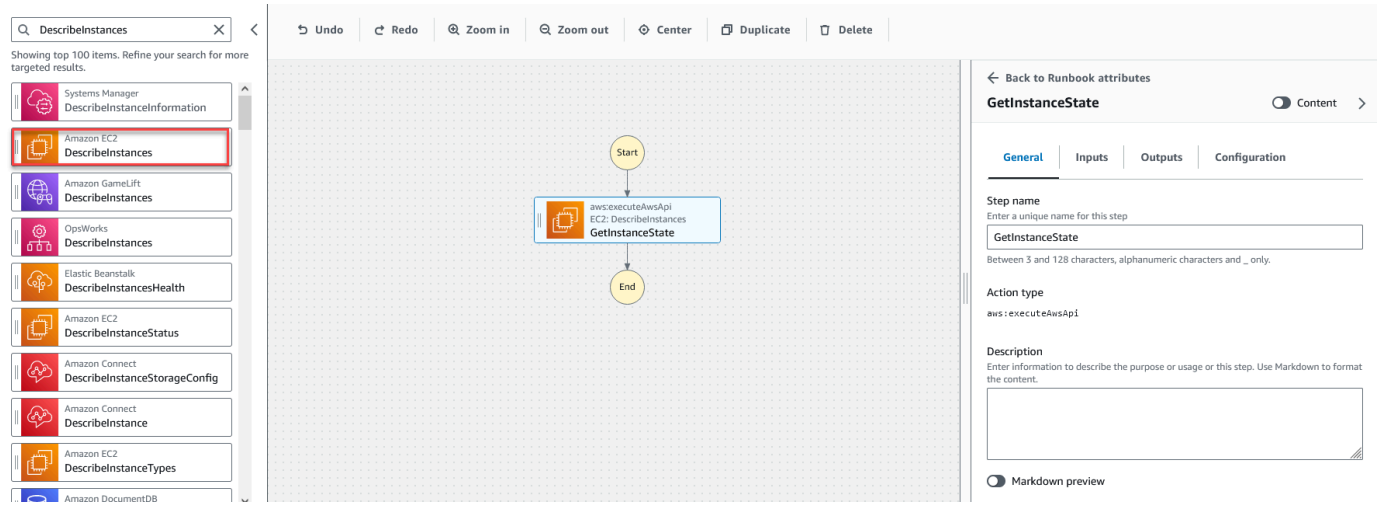
a. Di bidang Nama parameter, masukkan**`InstanceId`**.

b. Di dropdown Type, pilih. AWS::EC2::Instance

c. Pilih sakelar yang Diperlukan.



3. Di browser AWS API, masukkan **`DescribeInstances`** di bilah pencarian.
4. Seret Amazon EC2 — DescribeInstances tindakan ke kanvas kosong.
5. Untuk nama Langkah, masukkan nilai. Untuk tutorial ini, Anda dapat menggunakan nama**`GetInstanceState`**.



a. Perluas tarik-turun input tambahan, dan di bidang Nama input, masukkan. **`InstanceIds`**

b. Pilih tab Input.

c. Di bidang Nilai input, pilih input **`InstanceId`** dokumen. Ini mereferensikan nilai parameter input yang Anda buat di awal prosedur. Karena InstanceIdsinput untuk `DescribeInstances` tindakan menerima `StringList` nilai, Anda harus membungkus InstanceIdinput dalam tanda kurung siku. YAMG untuk nilai Input harus cocok dengan yang berikut:**`['{{ InstanceId }}']`**.

d. Di tab Output, pilih Tambahkan output dan masukkan **`InstanceState`** di bidang Nama.

e. Di bidang Selector, masukkan**`$.Reservations[0].Instances[0].State.Name`**.

f. Di dropdown Type, pilih String.

6. Seret tindakan Branch dari browser Actions, dan jatuhkan di bawah **`GetInstanceState`**langkah.

7. Untuk nama Langkah, masukkan nilai. Untuk tutorial ini, gunakan namanya**`BranchOnInstanceState`**.

   Untuk menentukan logika percabangan, lakukan hal berikut:

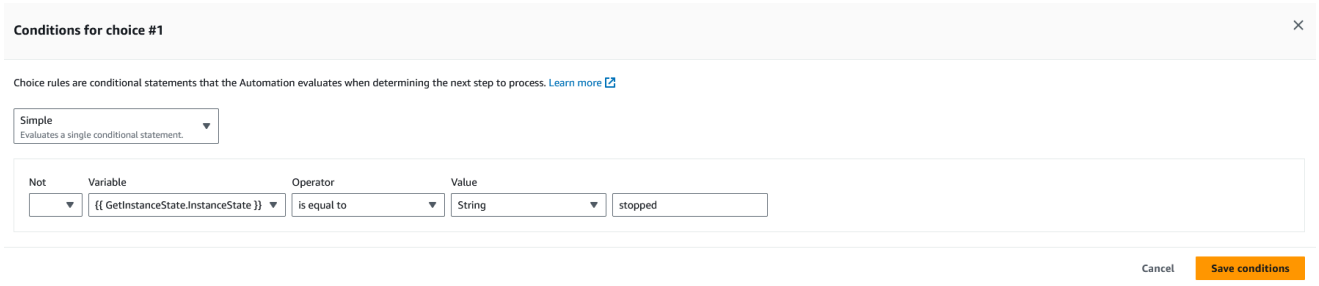
   a. Pilih **`Branch`**status di kanvas. Kemudian, di bawah Input dan Pilihan, pilih ikon pensil untuk mengedit Aturan #1.

   b. Pilih Tambahkan kondisi.

   c. Dalam kotak dialog Conditions for rule #1, pilih output **`GetInstanceState.InstanceState`** langkah dari dropdown Variable.

   d. Untuk Operator, pilih sama dengan.

e. Untuk Nilai, pilih String dari daftar dropdown. Masukkan **`stopped`**.



f. Pilih Simpan kondisi.

g. Pilih Tambahkan aturan pilihan baru.

h. Pilih Tambahkan kondisi untuk Aturan #2.

i. Dalam kotak dialog Conditions for rule #2, pilih output **`GetInstanceState.InstanceState`** langkah dari dropdown Variable.

j. Untuk Operator, pilih sama dengan.

k. Untuk Nilai, pilih String dari daftar dropdown. Masukkan **`stopping`**.

l. Pilih Simpan kondisi.

m. Pilih Tambahkan aturan pilihan baru.

n. Untuk Aturan #3, pilih Tambahkan kondisi.

o. Dalam kotak dialog Conditions for rule #3, pilih output **`GetInstanceState.InstanceState`** langkah dari dropdown Variable.

p. Untuk Operator, pilih sama dengan.

q. Untuk Nilai, pilih String dari daftar dropdown. Masukkan **`running`**.

r. Pilih Simpan kondisi.

s. Dalam aturan Default, pilih Go to end untuk langkah Default.

8. Seret tindakan Change an instance state ke kotak Drag action here kosong di bawah {{ GetInstanceState. InstanceState }} == kondisi “berhenti”.

   a. Untuk nama Langkah, masukkan**`StartInstance`**.

   b. Di tab Inputs, di bawah ID Instance, pilih nilai input InstanceIddokumen dari dropdown.

   c. Untuk status yang diinginkan, tentukan **`running`**.

9. Seret aksi Tunggu AWS sumber daya ke kotak Drag action here kosong di bawah {{ GetInstanceState. InstanceState }} == kondisi “berhenti”.

10. Untuk nama Langkah, masukkan nilai. Untuk tutorial ini, gunakan namanya**`WaitForInstanceStop`**.

    a. Untuk bidang Layanan, pilih Amazon EC2.

    b. Untuk bidang API, pilih DescribeInstances.

    c. Untuk bidang pemilih Properti, masukkan**`$.Reservations[0].Instances[0].State.Name`**.

    d. Untuk parameter Nilai yang diinginkan, masukkan**`["stopped"]`**.

    e. Di tab Konfigurasi WaitForInstanceStoptindakan, pilih StartInstancedari dropdown langkah Berikutnya.

11. Seret perintah Jalankan pada tindakan instance ke kotak Drag action here kosong di bawah {{ GetInstanceState. InstanceState }} == kondisi “berjalan”.

12. Untuk nama Langkah, masukkan**`SayHello`**.

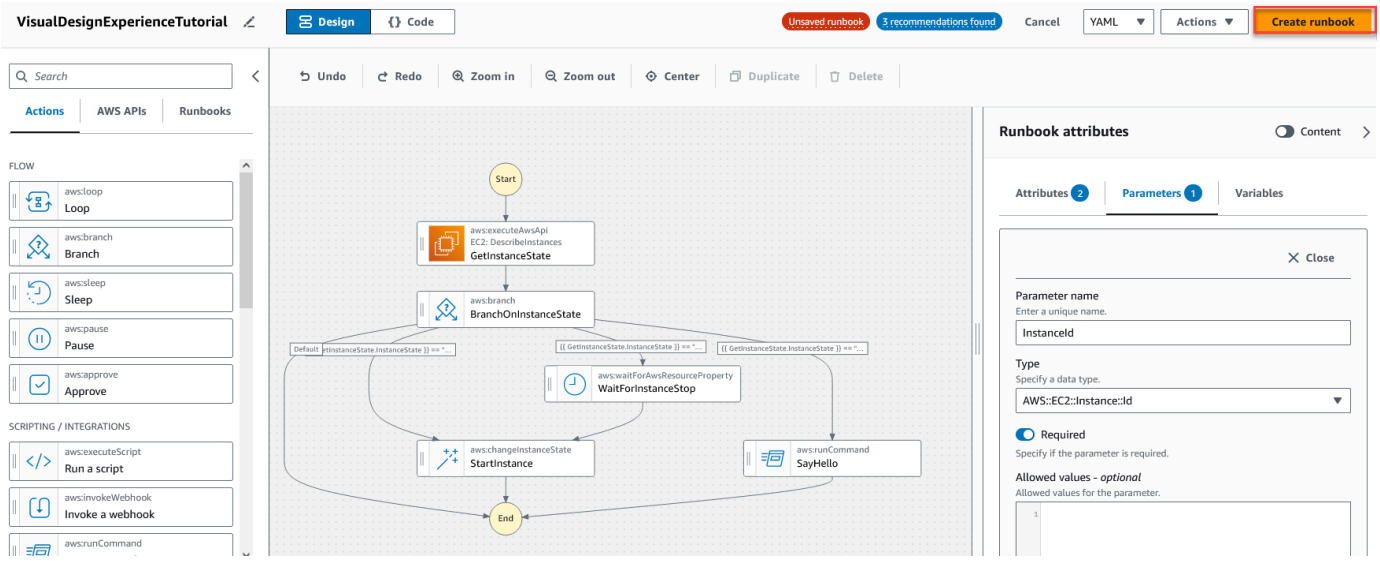
    a. Di tab Input, masukkan **`AWS-RunShellScript`** parameter nama Dokumen.

    b. Untuk InstanceIds, pilih nilai input InstanceIddokumen dari dropdown.

    c. Perluas tarik-turun input tambahan, dan di dropdown nama Input, pilih Parameter.

    d. Di bidang Nilai input, masukkan**`{"commands": "echo 'Hello World'"}`**.

13. Tinjau runbook yang sudah selesai di kanvas dan pilih Create runbook untuk menyimpan runbook tutorial.



Langkah 3: Tinjau kode yang dibuat secara otomatis

Saat Anda menyeret dan melepaskan tindakan dari browser Actions ke kanvas, pengalaman desain visual secara otomatis menyusun konten YAMG atau JSON dari runbook Anda secara real-time. Anda dapat melihat dan mengedit kode ini. Untuk melihat kode yang dibuat secara otomatis, pilih Kode untuk Desain dan Kode sakelar.

Langkah 4: Jalankan runbook baru Anda

Setelah membuat runbook Anda, Anda dapat menjalankan otomatisasi.

Untuk menjalankan runbook otomatisasi baru Anda

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Otomatisasi, lalu pilih Eksekusi otomatisasi.
3. Di daftar Dokumen otomatisasi, pilih runbook. Pilih satu opsi atau lebih di panel Kategori dokumen untuk memfilter dokumen SSM sesuai dengan tujuannya. Untuk melihat runbook yang Anda miliki, pilih tab Dimiliki oleh saya. Untuk melihat runbook yang dibagikan dengan akun Anda, pilih tab Dibagikan dengan saya. Untuk melihat semua runbook, pilih tab Semua dokumen.

   > **Note**
   >
   > Anda dapat melihat informasi tentang runbook dengan memilih nama runbook.

4. Di bagian Detail dokumen, verifikasi bahwa Versi dokumen diatur ke versi yang ingin Anda jalankan. Sistem ini termasuk pilihan versi berikut:

   - Versi default saat runtime — Pilih opsi ini jika runbook Otomasi diperbarui secara berkala dan versi default baru ditetapkan.
   - Versi terbaru saat runtime — Pilih opsi ini jika runbook Otomasi diperbarui secara berkala, dan Anda ingin menjalankan versi yang terbaru diperbarui.
   - 1 (Default) - Pilih opsi ini untuk menjalankan versi pertama dokumen, yang merupakan default.
5. Pilih Berikutnya.
6. Di bagian Execute Automation Runbook, pilih Eksekusi sederhana.
7. Di bagian Parameter input, tentukan input yang diperlukan. Secara opsional, Anda dapat memilih peran layanan IAM dari daftar. AutomationAssumeRole
8. (Opsional) Pilih CloudWatch alarm Amazon untuk diterapkan ke otomatisasi Anda untuk pemantauan. Untuk memasang CloudWatch alarm ke otomatisasi Anda, prinsip IAM yang

memulai otomatisasi harus memiliki izin untuk `iam:createServiceLinkedRole` tindakan tersebut. Untuk informasi selengkapnya tentang CloudWatch alarm, lihat Menggunakan CloudWatch alarm Amazon. Jika alarm Anda aktif, otomatisasi dihentikan. Jika Anda menggunakan AWS CloudTrail, Anda akan melihat panggilan API di jejak Anda.

9. Pilih Eksekusi.

Langkah 5: Bersihkan

Untuk menghapus runbook Anda

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.
3. Pilih tab Dimiliki oleh saya.
4. Temukan VisualDesignExperienceTutorialrunbook.
5. Pilih tombol pada halaman kartu dokumen, lalu pilih Hapus dokumen dari dropdown Tindakan.

Menyiapkan runbook Otomatisasi

Setiap runbook di OtomatisasiAWS Systems Manager mendefinisikan otomatisasi. Runbook otomatisasi menentukan tindakan yang dilakukan selama otomatisasi. Dalam konten runbook, Anda mendefinisikan parameter input, output, dan tindakan yang dilakukan Systems Manager pada instans terkelola Anda danAWS sumber daya.

Otomatisasi mencakup beberapa runbook yang telah ditetapkan dan dapat Anda gunakan untuk melakukan beberapa tugas umum seperti memulai ulang satu instans Amazon Elastic Compute Cloud (Amazon EC2) atau lebih atau membuatAmazon Machine Image (AMI). Namun, kasus penggunaan Anda mungkin melampaui kemampuan runbook yang telah ditentukan sebelumnya. Jika ini masalahnya, Anda dapat membuat runbook Anda sendiri dan memodifikasinya sesuai kebutuhan Anda.

Runbook terdiri dari tindakan otomatisasi, parameter untuk tindakan tersebut, dan parameter input yang Anda tentukan. Konten runbook ditulis dalam YAKL atau JSON. Jika Anda tidak terbiasa dengan YAKL atau JSON, sebaiknya gunakan Document Builder, atau pelajari lebih lanjut tentang bahasa markup sebelum mencoba membuat runbook Anda sendiri. Untuk informasi selengkapnya tentang Document Builder, lihatMenggunakan Document Builder untuk membuat runbook.

Bagian berikut akan membantu Anda menulis runbook pertama Anda.

Identifikasi kasus penggunaan Anda

Langkah pertama dalam menulis runbook adalah mengidentifikasi kasus penggunaan Anda. Misalnya, Anda menjadwalkan`AWS-CreateImage` runbook untuk berjalan setiap hari di semua instans Amazon EC2 produksi Anda. Di akhir bulan, Anda memutuskan Anda memiliki lebih banyak gambar daripada yang diperlukan untuk titik pemulihan. Ke depan, Anda ingin secara otomatis menghapus instans Amazon EC2 tertuaAMI saat baruAMI dibuat. Untuk mencapai hal ini, Anda membuat runbook baru yang melakukan hal berikut:

1. Menjalankan`aws:createImage` tindakan dan menentukan ID contoh dalam deskripsi gambar.
2. Menjalankan`aws:waitForAwsResourceProperty` tindakan untuk polling keadaan gambar sampai itu`available`.
3. Setelah status gambar`available`,`aws:executeScript` aksi menjalankan skrip Python khusus yang mengumpulkan ID semua gambar yang terkait dengan instans Amazon EC2 Anda. Skrip melakukan ini dengan memfilter, menggunakan ID instance dalam deskripsi gambar yang Anda tentukan saat pembuatan. Kemudian, skrip mengurutkan daftar ID gambar berdasarkan gambar dan mengeluarkan ID yang tertuaAMI.`creationDate`
4. Terakhir,`aws:deleteImage` tindakan berjalan untuk menghapus yang tertuaAMI menggunakan ID dari output di langkah sebelumnya.

Dalam skenario ini, Anda sudah menggunakan`AWS-CreateImage` runbook tetapi menemukan bahwa kasus penggunaan Anda membutuhkan fleksibilitas yang lebih besar. Ini adalah situasi umum karena mungkin ada tumpang tindih antara runbook dan tindakan otomatisasi. Akibatnya, Anda mungkin harus menyesuaikan runbook atau tindakan yang Anda gunakan untuk mengatasi kasus penggunaan Anda.

Misalnya,`aws:executeScript` dan`aws:invokeLambdaFunction` tindakan keduanya memungkinkan Anda menjalankan skrip khusus sebagai bagian dari otomatisasi Anda. Untuk memilih di antara mereka, Anda mungkin lebih suka`aws:invokeLambdaFunction` karena

bahasa runtime tambahan yang didukung. Namun, Anda mungkin lebih sukaaws:executeScript karena memungkinkan Anda untuk menulis konten skrip Anda langsung di runbook YAKL dan menyediakan konten skrip sebagai lampiran untuk runbook JSON. Anda mungkin juga mempertimbangkanaws:executeScript untuk menjadi lebih sederhana dalam hal pengaturanAWS Identity and Access Management (IAM). Karena menggunakan izin yang disediakan diAutomationAssumeRole,aws:executeScript tidak memerlukan peran eksekusiAWS Lambda fungsi tambahan.

Dalam skenario tertentu, satu tindakan mungkin memberikan lebih banyak fleksibilitas, atau fungsionalitas tambahan, di atas yang lain. Oleh karena itu, kami sarankan Anda meninjau parameter input yang tersedia untuk runbook atau tindakan yang ingin Anda gunakan untuk menentukan mana yang paling sesuai dengan kasus penggunaan dan preferensi Anda.

Siapkan lingkungan pengembangan Anda

Setelah Anda mengidentifikasi kasus penggunaan Anda dan runbook yang telah ditentukan atau tindakan otomatisasi yang ingin Anda gunakan di runbook Anda, saatnya untuk mengatur lingkungan pengembangan Anda untuk konten runbook Anda. Untuk mengembangkan konten runbook Anda, sebaiknya gunakan konsol Systems Manager Documents.AWS Toolkit for Visual Studio Code

Toolkit for VS Code adalah ekstensi open-source untuk Visual Studio Code (VS Code) yang menawarkan lebih banyak fitur daripada konsol Systems Manager Documents. Fitur yang bermanfaat termasuk validasi skema untuk YAKL dan JSON, cuplikan untuk jenis tindakan otomatisasi, dan dukungan pelengkapan otomatis untuk berbagai opsi di YAKL dan JSON.

Untuk informasi selengkapnya tentang menginstal Toolkit for VS Code, lihat MenginstalAWS Toolkit for Visual Studio Code. Untuk informasi selengkapnya tentang penggunaan Toolkit for VS Code untuk mengembangkan runbook, lihat Bekerja dengan dokumen Otomatisasi Systems Manager di PanduanAWS Toolkit for Visual Studio Code Pengguna.

Kembangkan konten runbook

Dengan kasus penggunaan Anda diidentifikasi dan lingkungan diatur, Anda siap untuk mengembangkan konten untuk runbook Anda. Kasus penggunaan dan preferensi Anda sebagian besar akan menentukan tindakan otomatisasi atau runbook yang Anda gunakan dalam konten runbook Anda. Beberapa tindakan hanya mendukung subset parameter input jika dibandingkan dengan tindakan lain yang memungkinkan Anda menyelesaikan tugas serupa. Tindakan lain memiliki output tertentu, sepertiaws:createImage, di mana beberapa tindakan memungkinkan Anda untuk menentukan output Anda sendiri, sepertiaws:executeAwsApi.

Jika Anda tidak yakin bagaimana menggunakan tindakan tertentu dalam runbook Anda, kami sarankan meninjau entri yang sesuai untuk tindakan diReferensi tindakan Otomatisasi Systems Manager. Kami juga merekomendasikan untuk meninjau konten runbook yang telah ditentukan sebelumnya untuk melihat contoh dunia nyata tentang bagaimana tindakan ini digunakan. Untuk lebih banyak contoh aplikasi runbook dunia nyata, lihatContoh runbook tambahan.

Untuk mendemonstrasikan perbedaan dalam kesederhanaan dan fleksibilitas yang disediakan konten runbook, tutorial berikut memberikan contoh cara menambal grup instans Amazon EC2 secara bertahap:

- the section called “Contoh 1: Membuat runbook orangtua-anak”- Dalam contoh ini, dua runbook digunakan dalam hubungan orang tua-anak. Runbook induk memulai otomatisasi kontrol tingkat runbook anak.
- the section called “Contoh 2: Skrip runbook”- Contoh ini menunjukkan bagaimana Anda dapat menyelesaikan tugas Contoh 1 yang sama dengan mengondensasi konten menjadi satu runbook dan menggunakan skrip di buku runbook Anda.

Contoh 1: Membuat runbook orangtua-anak

Contoh berikut ini menunjukkan cara membuat dua runbook yang menambal grup instans Amazon Elastic Compute Cloud (Amazon EC2) secara bertahap. Runbook ini digunakan dalam hubungan orangtua-anak dengan runbook induk yang digunakan untuk memulai otomatisasi kontrol tingkat runbook anak. Untuk informasi selengkapnya tentang otomatisasi kontrol tarif tarif, lihatJalankan otomatisasi dalam skala besar. Untuk informasi selengkapnya tentang tindakan otomatisasi yang digunakan dalam contoh ini, lihatReferensi tindakan Otomatisasi Systems Manager.

Buat runbook anak

Contoh ini runbook membahas skenario berikut: Emily adalah Systems Engineer di AnyCompany Consultants, LLC. Dia perlu mengonfigurasi patching untuk grup instans Amazon Elastic Compute Cloud (Amazon EC2) yang meng-host database primer dan sekunder. Aplikasi mengakses database ini 24 jam sehari, jadi salah satu instance database harus selalu tersedia.

Dia memutuskan bahwa menambal instance secara bertahap adalah pendekatan terbaik. Kelompok utama contoh database akan ditambal pertama, diikuti oleh kelompok sekunder contoh database. Selain itu, untuk menghindari biaya tambahan dengan membiarkan instance berjalan yang sebelumnya dihentikan, Emily ingin instance yang ditambal dikembalikan ke keadaan semula sebelum penambalan terjadi.

Emily mengidentifikasi grup instance database primer dan sekunder dengan tag yang terkait dengan instance. Dia memutuskan untuk membuat runbook induk yang memulai otomatisasi kontrol tingkat dari runbook anak. Dengan melakukan itu, dia dapat menargetkan tag yang terkait dengan grup instance database primer dan sekunder dan mengelola konkurensi otomatisasi anak. Setelah meninjau dokumen Systems Manager (SSM) yang tersedia untuk ditambal, dia memilih`AWS-RunPatchBaseline` dokumen tersebut. Dengan menggunakan dokumen SSM ini, rekan-rekannya dapat meninjau informasi kepatuhan patch terkait setelah operasi patching selesai.

Untuk mulai membuat konten runbook-nya, Emily meninjau tindakan otomatisasi yang tersedia dan mulai menulis konten untuk runbook anak sebagai berikut:

1. Pertama, dia memberikan nilai untuk skema dan deskripsi runbook, dan mendefinisikan parameter input untuk runbook anak.

   Dengan menggunakan`AutomationAssumeRole` parameter, Emily dan rekan-rekannya dapat menggunakan peran IAM yang ada yang memungkinkan Automation untuk melakukan tindakan di runbook atas nama mereka. Emily menggunakan`InstanceId` parameter untuk menentukan instance yang harus ditambal. Opsional,`Operation`,`RebootOption`, dan`SnapshotId` parameter dapat digunakan untuk memberikan nilai-nilai untuk parameter dokumen untuk`AWS-RunPatchBaseline`. Untuk mencegah nilai yang tidak valid diberikan kepada parameter dokumen tersebut, dia mendefinisikan`allowedValues` sesuai kebutuhan.

   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. Dengan unsur-unsur tingkat atas didefinisikan, Emily melanjutkan dengan authoring tindakan`mainSteps` yang membentuk runbook. Langkah pertama menampilkan status saat ini dari instance target yang ditentukan dalam parameter`InstanceId` input menggunakan`aws:executeAwsApi` tindakan. Output dari tindakan ini digunakan dalam tindakan selanjutnya.

   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. Alih-alih memulai dan melacak status asli setiap instance secara manual yang perlu ditambal, Emily menggunakan output dari tindakan sebelumnya untuk mencabang otomatisasi berdasarkan status instance target. Hal ini memungkinkan otomatisasi untuk menjalankan langkah-langkah

yang berbeda tergantung pada kondisi yang ditentukan dalam`aws:branch` tindakan dan meningkatkan efisiensi keseluruhan otomatisasi tanpa intervensi manual.

Jika status instance sudah`running`, otomatisasi akan berlanjut dengan menambal instance dengan`AWS-RunPatchBaseline` dokumen menggunakan`aws:runCommand` action.

Jika status instance adalah`stopping`, otomatisasi jajak pendapat untuk instance untuk mencapai`stopped` status menggunakan`aws:waitForAwsResourceProperty` tindakan, memulai instance menggunakan`executeAwsApi` tindakan, dan polling untuk instance untuk mencapai`running` status sebelum menambal instance.

Jika status instance`stopped`, otomatisasi memulai instance dan melakukan polling untuk instance untuk mencapai`running` status sebelum menambal instance menggunakan tindakan yang sama.

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. Setelah operasi patching selesai, Emily ingin otomatisasi mengembalikan instance target ke keadaan yang sama sebelum otomatisasi dimulai. Dia melakukan ini dengan lagi menggunakan output dari tindakan pertama. Cabang otomatisasi berdasarkan keadaan asli dari instance target menggunakan`aws:branch` aksi. Jika instance sebelumnya dalam keadaan apa pun selain`running`, instance dihentikan. Jika tidak, jika status instance`running`, otomatisasi berakhir.

   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 meninjau konten runbook anak yang telah selesai dan membuat runbook dalam contoh yang samaAkun AWS danWilayah AWS sebagai target. Sekarang dia siap untuk melanjutkan pembuatan konten runbook induk. Berikut ini adalah konten runbook anak yang telah selesai.

   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}}"
            ]
         }
      }
   ]
}
```

Untuk informasi selengkapnya tentang tindakan otomatisasi yang digunakan dalam contoh ini, lihatReferensi tindakan Otomatisasi Systems Manager.

Buat runbook induk

Contoh ini runbook melanjutkan skenario yang dijelaskan pada bagian sebelumnya. Sekarang Emily telah membuat runbook anak, dia mulai menulis konten untuk runbook induk sebagai berikut:

1. Pertama, dia memberikan nilai untuk skema dan deskripsi runbook, dan mendefinisikan parameter input untuk runbook induk.

   Dengan menggunakan`AutomationAssumeRole` parameter, Emily dan rekan-rekannya dapat menggunakan peran IAM yang ada yang memungkinkan Automation untuk melakukan tindakan di runbook atas nama mereka. Emily menggunakan`PatchGroupPrimaryValue` parameter`PatchGroupPrimaryKey` and untuk menentukan tag yang terkait dengan kelompok utama instance database yang akan ditambal. Dia menggunakan`PatchGroupSecondaryKey` dan`PatchGroupSecondaryValue` parameter untuk menentukan tag yang terkait dengan kelompok sekunder contoh database yang akan ditambal.

   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. Dengan unsur-unsur tingkat atas didefinisikan, Emily melanjutkan dengan authoring tindakan`mainSteps` yang membentuk runbook.

   Tindakan pertama memulai otomatisasi kontrol tingkat menggunakan runbook anak yang baru saja dia buat yang menargetkan instance yang terkait dengan tag yang ditentukan dalam parameter`PatchGroupPrimaryKey` dan`PatchGroupPrimaryValue` input. Dia menggunakan nilai yang diberikan kepada parameter input untuk menentukan kunci dan nilai tag yang terkait dengan kelompok utama instance database yang ingin ditambal.

   Setelah otomatisasi pertama selesai, tindakan kedua memulai otomatisasi kontrol tingkat lain menggunakan runbook anak yang menargetkan instance yang terkait dengan tag yang ditentukan dalam parameter`PatchGroupSecondaryKey` dan`PatchGroupSecondaryValue` input. Dia menggunakan nilai-nilai yang diberikan kepada parameter input untuk menentukan kunci dan nilai tag yang terkait dengan kelompok sekunder contoh database yang dia ingin patch.

   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 meninjau konten runbook induk yang telah selesai dan membuat runbook dalam contoh yang samaAkun AWS danWilayah AWS sebagai target. Sekarang, dia siap untuk menguji runbook-nya untuk memastikan otomatisasi beroperasi seperti yang diinginkan sebelum menerapkannya ke lingkungan produksinya. Berikut ini adalah konten runbook induk yang telah selesai.

   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"
         }
      }
   ]
}
```

Untuk informasi selengkapnya tentang tindakan otomatisasi yang digunakan dalam contoh ini, lihatReferensi tindakan Otomatisasi Systems Manager.

Contoh 2: Skrip runbook

Contoh runbook ini membahas skenario berikut: Emily adalah Systems Engineer di AnyCompany Konsultan, LLC. Dia sebelumnya membuat dua runbook yang digunakan dalam hubungan orangtua-anak untuk melakukan patch pada grup instans Amazon Elastic Compute Cloud (Amazon EC2) yang meng-host database primer dan sekunder. Aplikasi mengakses database ini 24 jam sehari, jadi salah satu instance database harus selalu tersedia.

Berdasarkan persyaratan ini, ia membangun solusi yang menambal instance secara bertahap menggunakan`AWS-RunPatchBaseline`Dokumen Systems Manager (SSM). Dengan menggunakan dokumen SSM ini, rekan-rekannya dapat meninjau informasi kepatuhan patch terkait setelah operasi patching selesai.

Kelompok utama contoh database ditambal pertama, diikuti oleh kelompok sekunder contoh database. Selain itu, untuk menghindari biaya tambahan dengan membiarkan instance berjalan yang sebelumnya dihentikan, Emily memastikan bahwa otomatisasi mengembalikan instance yang ditambal ke keadaan semula sebelum penambalan terjadi. Emily menggunakan tag yang terkait dengan grup instance database primer dan sekunder untuk mengidentifikasi instance mana yang harus ditambal sesuai urutan yang diinginkannya.

Solusi otomatisnya yang ada berhasil, tetapi dia ingin meningkatkan solusinya jika memungkinkan. Untuk membantu pemeliharaan konten runbook dan untuk memudahkan upaya pemecahan masalah,

dia ingin memadatkan otomatisasi menjadi satu runbook dan menyederhanakan jumlah parameter input. Juga, dia ingin menghindari membuat beberapa otomatisasi anak.

Setelah Emily meninjau tindakan otomatisasi yang tersedia, dia menentukan bahwa dia dapat meningkatkan solusinya dengan menggunakan`aws:executeScript`tindakan untuk menjalankan skrip Python kustomnya. Dia sekarang mulai menulis konten untuk runbook sebagai berikut:

1. Pertama, dia memberikan nilai untuk skema dan deskripsi runbook, dan mendefinisikan parameter input untuk runbook induk.

   Dengan menggunakan`AutomationAssumeRole`parameter, Emily dan rekan-rekannya dapat menggunakan peran IAM yang ada yang memungkinkan Automation untuk melakukan tindakan dalam runbook atas nama mereka. Tidak seperti Contoh 1, yang`AutomationAssumeRole`parameter sekarang diperlukan daripada opsional. Karena runbook ini termasuk`aws:executeScript`tindakan, sebuahAWS Identity and Access Management(IAM) peran layanan (atau menganggap peran) selalu diperlukan. Persyaratan ini diperlukan karena beberapa skrip Python ditentukan untuk tindakan panggilanAWSOperasi API.

   Emily menggunakan`PrimaryPatchGroupTag`dan`SecondaryPatchGroupTag`parameter untuk menentukan tag yang terkait dengan kelompok primer dan sekunder contoh database yang akan ditambal. Untuk menyederhanakan parameter input yang diperlukan, dia memutuskan untuk menggunakan`StringMap`parameter daripada menggunakan beberapa`String`parameter seperti yang dia gunakan dalam contoh 1 runbook. Secara opsional,`Operation`,`RebootOption`, dan`SnapshotId`parameter dapat digunakan untuk memberikan nilai untuk mendokumentasikan parameter untuk`AWS-RunPatchBaseline`. Untuk mencegah nilai-nilai yang tidak valid dari yang disediakan untuk parameter dokumen, dia mendefinisikan`allowedValues`sesuai kebutuhan.

   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. Dengan elemen tingkat atas didefinisikan, Emily melanjutkan dengan authoring tindakan yang membentuk`mainSteps`dari runbook. Langkah pertama mengumpulkan ID dari semua instance yang terkait dengan tag yang ditentukan dalam`PrimaryPatchGroupTag`parameter dan output`StringMap`parameter yang berisi ID instans dan status tabel tabel saat ini (). Output dari tindakan ini digunakan dalam tindakan selanjutnya.

   Perhatikan bahwa`script`parameter input tidak mendukung runbook JSON. JSON runbook harus menyediakan konten skrip menggunakan`attachment`parameter masukan.

   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 menggunakan output dari tindakan sebelumnya di tindakan lain`aws:executeScript`tindakan untuk memverifikasi semua instance yang terkait dengan tag yang ditentukan dalam`PrimaryPatchGroupTag`parameter berada dalam`running`status.

Jika status instance sudah`running`atau`shutting-down`, script terus loop melalui contoh yang tersisa.

Jika keadaan instance`stopping`, script polling untuk instance untuk mencapai`stopped`Status dan mulai instans.

Jika keadaan instance`stopped`, script memulai instance.

YAML

```
- name: verifyPrimaryInstancesRunning
    action: 'aws:executeScript'
    timeoutSeconds: 600
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: verifyInstancesRunning
      InputPayload:
        targetInstances: '{{getPrimaryInstanceState.originalInstanceStates}}'
      Script: |-
        def verifyInstancesRunning(events,context):
          import boto3

          #Initialize client
          ec2 = boto3.client('ec2')
          instanceDict = events['targetInstances']
          for instance in instanceDict:
            if instanceDict[instance] == 'stopped':
                print("The target instance " + instance + " is stopped. The
 instance will now be started.")
                ec2.start_instances(
                    InstanceIds=[instance]
                    )
            elif instanceDict[instance] == 'stopping':
                print("The target instance " + instance + " is stopping. Polling
 for instance to reach stopped state.")
                while instanceDict[instance] != 'stopped':
                    poll = ec2.get_waiter('instance_stopped')
                    poll.wait(
                        InstanceIds=[instance]
                    )
                ec2.start_instances(
                    InstanceIds=[instance]
```

```
                )
            else:
              pass
    nextStep: waitForPrimaryRunningInstances
```

JSON

```
{
         "name":"verifyPrimaryInstancesRunning",
         "action":"aws:executeScript",
         "timeoutSeconds":600,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"verifyInstancesRunning",
            "InputPayload":{

 "targetInstances":"{{getPrimaryInstanceState.originalInstanceStates}}"
            },
            "Script":"..."
         },
         "nextStep":"waitForPrimaryRunningInstances"
      },
```

4. Emily memverifikasi bahwa semua instance yang terkait dengan tag yang ditentukan dalam`PrimaryPatchGroupTag`parameter yang dimulai atau sudah dalam`running`status. Kemudian dia menggunakan skrip lain untuk memverifikasi bahwa semua contoh, termasuk yang dimulai pada tindakan sebelumnya, telah mencapai`running`status.

   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 menggunakan dua skrip lagi untuk mengembalikan individu`String`nilai kunci dan nilai tag yang ditentukan dalam`PrimaryPatchGroupTag`parameter. Nilai-nilai yang dikembalikan oleh tindakan ini memungkinkannya untuk memberikan nilai langsung ke`Targets`parameter untuk`AWS-RunPatchBaseline`dokumen. Otomatisasi kemudian dilanjutkan dengan menambal instance dengan`AWS-RunPatchBaseline`dokumen menggunakan`aws:runCommand`tindakan.

   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. Setelah operasi patching selesai, Emily ingin otomatisasi mengembalikan instance target yang terkait dengan tag yang ditentukan dalam`PrimaryPatchGroupTag`parameter ke keadaan yang sama sebelum otomatisasi dimulai. Dia melakukan ini dengan lagi menggunakan output dari tindakan pertama dalam skrip. Berdasarkan keadaan asli dari instance target, jika instance sebelumnya dalam keadaan apa pun selain`running`, instans dihentikan. Jika tidak, jika keadaan instance`running`, script terus loop melalui contoh yang tersisa.

   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. Operasi patching selesai untuk instance yang terkait dengan tag yang ditentukan dalam`PrimaryPatchGroupTag`parameter. Sekarang Emily menduplikasi semua tindakan sebelumnya dalam konten runbook-nya untuk menargetkan instance yang terkait dengan tag yang ditentukan dalam`SecondaryPatchGroupTag`parameter.

   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 meninjau konten runbook scripted yang telah selesai dan membuat runbook dalam hal yang samaAkun AWSdanWilayah AWSsebagai contoh target. Sekarang dia siap untuk menguji runbooknya untuk memastikan otomatisasi beroperasi seperti yang diinginkan sebelum menerapkannya ke lingkungan produksinya. Berikut ini adalah konten runbook scripted yang telah selesai.

   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":"..."
         }
      }
   ]
}
```

Untuk informasi selengkapnya tentang tindakan otomatisasi yang digunakan dalam contoh ini, lihatReferensi tindakan Otomatisasi Systems Manager.

Contoh runbook tambahan

Runbook contoh berikut menunjukkan cara menggunakan tindakanAWS Systems Manager otomatisasi untuk mengotomatiskan deployment umum, pemecahan masalah, dan tugas pemeliharaan.

> **Note**
>
> Runbook contoh di bagian ini disediakan untuk menunjukkan kepada Anda cara membuat runbook kustom untuk mendukung kebutuhan operasional spesifik Anda. Runbook ini tidak dirancang untuk digunakan di lingkungan produksi sebagaimana adanya. Namun, Anda dapat menyesuaikannya untuk penggunaan Anda sendiri.

Contoh

- Deploy arsitektur VPC dan pengendali domain Microsoft Active Directory
- Kembalikan volume root dari snapshot terbaru
- Buat AMI dan salinan lintas wilayah

Deploy arsitektur VPC dan pengendali domain Microsoft Active Directory

Untuk meningkatkan efisiensi dan menstandarisasi tugas umum, Anda dapat memilih untuk mengotomatiskan deployment. Hal ini berguna jika secara teratur Anda menggunakan arsitektur yang sama di beberapa akun dan Wilayah AWS. Mengotomatiskan deployment arsitektur juga dapat mengurangi potensi kesalahan manusia yang dapat terjadi ketika menyebarkan arsitektur secara manual. AWS Systems Manager Tindakan otomatisasi dapat membantu Anda mencapai hal ini. Otomatisasi adalah kemampuan AWS Systems Manager.

ContohAWS Systems Manager runbook berikut melakukan tindakan ini:

- Mengambil Windows Server 2016Amazon Machine Image (AMI) terbaru Systems ManagerParameter Store untuk digunakan ketika meluncurkan instans EC2 yang akan dikonfigurasi sebagai pengendali domain. Parameter Storeadalah kemampuanAWS Systems Manager.
- Menggunakan `aws:executeAwsApi` tindakan otomatisasi untuk memanggil beberapa AWS operasi API guna membuat arsitektur VPC. Contoh kontroler domain diluncurkan di subnet pribadi, dan terhubung ke internet menggunakan gateway NAT. Hal iniSSM Agent mengizinkan instans untuk mengakses titik akhir Systems Manager yang diperlukan.
- Menggunakan `aws:waitForAwsResourceProperty` tindakan otomatisasi untuk mengonfirmasi instans yang diluncurkan oleh tindakan sebelumnya `Online` untuk AWS Systems Manager.
- Menggunakan `aws:runCommand` tindakan otomatisasi untuk mengonfigurasi contoh yang diluncurkan sebagai pengendali domain Microsoft Active Directory.

YAML

```
---
description: Custom Automation Deployment Example
schemaVersion: '0.3'
parameters:
  AutomationAssumeRole:
    type: String
    default: ''
    description: >-
      (Optional) The ARN of the role that allows Automation to perform the
      actions on your behalf. If no role is specified, Systems Manager
      Automation uses your IAM permissions to run this runbook.
mainSteps:
```

```
      - name: getLatestWindowsAmi
        action: aws:executeAwsApi
        onFailure: Abort
        inputs:
          Service: ssm
          Api: GetParameter
          Name: >-
            /aws/service/ami-windows-latest/Windows_Server-2016-English-Full-Base
        outputs:
          - Name: amiId
            Selector: $.Parameter.Value
            Type: String
        nextStep: createSSMInstanceRole
      - name: createSSMInstanceRole
        action: aws:executeAwsApi
        onFailure: Abort
        inputs:
          Service: iam
          Api: CreateRole
          AssumeRolePolicyDocument: >-
            {"Version":"2012-10-17","Statement":[{"Effect":"Allow","Principal":
{"Service":["ec2.amazonaws.com"]},"Action":["sts:AssumeRole"]}]}
          RoleName: sampleSSMInstanceRole
        nextStep: attachManagedSSMPolicy
      - name: attachManagedSSMPolicy
        action: aws:executeAwsApi
        onFailure: Abort
        inputs:
          Service: iam
          Api: AttachRolePolicy
          PolicyArn: 'arn:aws:iam::aws:policy/service-role/
AmazonSSMManagedInstanceCore'
          RoleName: sampleSSMInstanceRole
        nextStep: createSSMInstanceProfile
      - name: createSSMInstanceProfile
        action: aws:executeAwsApi
        onFailure: Abort
        inputs:
          Service: iam
          Api: CreateInstanceProfile
          InstanceProfileName: sampleSSMInstanceRole
        outputs:
          - Name: instanceProfileArn
            Selector: $.InstanceProfile.Arn
```

```
        Type: String
    nextStep: addSSMInstanceRoleToProfile
  - name: addSSMInstanceRoleToProfile
    action: aws:executeAwsApi
    onFailure: Abort
    inputs:
      Service: iam
      Api: AddRoleToInstanceProfile
      InstanceProfileName: sampleSSMInstanceRole
      RoleName: sampleSSMInstanceRole
    nextStep: createVpc
  - name: createVpc
    action: aws:executeAwsApi
    onFailure: Abort
    inputs:
      Service: ec2
      Api: CreateVpc
      CidrBlock: 10.0.100.0/22
    outputs:
      - Name: vpcId
        Selector: $.Vpc.VpcId
        Type: String
    nextStep: getMainRtb
  - name: getMainRtb
    action: aws:executeAwsApi
    onFailure: Abort
    inputs:
      Service: ec2
      Api: DescribeRouteTables
      Filters:
        - Name: vpc-id
          Values:
            - '{{ createVpc.vpcId }}'
    outputs:
      - Name: mainRtbId
        Selector: '$.RouteTables[0].RouteTableId'
        Type: String
    nextStep: verifyMainRtb
  - name: verifyMainRtb
    action: aws:assertAwsResourceProperty
    onFailure: Abort
    inputs:
      Service: ec2
      Api: DescribeRouteTables
```

```
          RouteTableIds:
            - '{{ getMainRtb.mainRtbId }}'
          PropertySelector: '$.RouteTables[0].Associations[0].Main'
          DesiredValues:
            - 'True'
        nextStep: createPubSubnet
      - name: createPubSubnet
        action: aws:executeAwsApi
        onFailure: Abort
        inputs:
          Service: ec2
          Api: CreateSubnet
          CidrBlock: 10.0.103.0/24
          AvailabilityZone: us-west-2c
          VpcId: '{{ createVpc.vpcId }}'
        outputs:
          - Name: pubSubnetId
            Selector: $.Subnet.SubnetId
            Type: String
        nextStep: createPubRtb
      - name: createPubRtb
        action: aws:executeAwsApi
        onFailure: Abort
        inputs:
          Service: ec2
          Api: CreateRouteTable
          VpcId: '{{ createVpc.vpcId }}'
        outputs:
          - Name: pubRtbId
            Selector: $.RouteTable.RouteTableId
            Type: String
        nextStep: createIgw
      - name: createIgw
        action: aws:executeAwsApi
        onFailure: Abort
        inputs:
          Service: ec2
          Api: CreateInternetGateway
        outputs:
          - Name: igwId
            Selector: $.InternetGateway.InternetGatewayId
            Type: String
        nextStep: attachIgw
      - name: attachIgw
```

```
      action: aws:executeAwsApi
      onFailure: Abort
      inputs:
        Service: ec2
        Api: AttachInternetGateway
        InternetGatewayId: '{{ createIgw.igwId }}'
        VpcId: '{{ createVpc.vpcId }}'
      nextStep: allocateEip
    - name: allocateEip
      action: aws:executeAwsApi
      onFailure: Abort
      inputs:
        Service: ec2
        Api: AllocateAddress
        Domain: vpc
      outputs:
        - Name: eipAllocationId
          Selector: $.AllocationId
          Type: String
      nextStep: createNatGw
    - name: createNatGw
      action: aws:executeAwsApi
      onFailure: Abort
      inputs:
        Service: ec2
        Api: CreateNatGateway
        AllocationId: '{{ allocateEip.eipAllocationId }}'
        SubnetId: '{{ createPubSubnet.pubSubnetId }}'
      outputs:
        - Name: natGwId
          Selector: $.NatGateway.NatGatewayId
          Type: String
      nextStep: verifyNatGwAvailable
    - name: verifyNatGwAvailable
      action: aws:waitForAwsResourceProperty
      timeoutSeconds: 150
      inputs:
        Service: ec2
        Api: DescribeNatGateways
        NatGatewayIds:
          - '{{ createNatGw.natGwId }}'
        PropertySelector: '$.NatGateways[0].State'
        DesiredValues:
          - available
```

```
      nextStep: createNatRoute
    - name: createNatRoute
      action: aws:executeAwsApi
      onFailure: Abort
      inputs:
        Service: ec2
        Api: CreateRoute
        DestinationCidrBlock: 0.0.0.0/0
        NatGatewayId: '{{ createNatGw.natGwId }}'
        RouteTableId: '{{ getMainRtb.mainRtbId }}'
      nextStep: createPubRoute
    - name: createPubRoute
      action: aws:executeAwsApi
      onFailure: Abort
      inputs:
        Service: ec2
        Api: CreateRoute
        DestinationCidrBlock: 0.0.0.0/0
        GatewayId: '{{ createIgw.igwId }}'
        RouteTableId: '{{ createPubRtb.pubRtbId }}'
      nextStep: setPubSubAssoc
    - name: setPubSubAssoc
      action: aws:executeAwsApi
      onFailure: Abort
      inputs:
        Service: ec2
        Api: AssociateRouteTable
        RouteTableId: '{{ createPubRtb.pubRtbId }}'
        SubnetId: '{{ createPubSubnet.pubSubnetId }}'
    - name: createDhcpOptions
      action: aws:executeAwsApi
      onFailure: Abort
      inputs:
        Service: ec2
        Api: CreateDhcpOptions
        DhcpConfigurations:
          - Key: domain-name-servers
            Values:
              - '10.0.100.50,10.0.101.50'
          - Key: domain-name
            Values:
              - sample.com
      outputs:
        - Name: dhcpOptionsId
```

```
          Selector: $.DhcpOptions.DhcpOptionsId
          Type: String
      nextStep: createDCSubnet1
    - name: createDCSubnet1
      action: aws:executeAwsApi
      onFailure: Abort
      inputs:
        Service: ec2
        Api: CreateSubnet
        CidrBlock: 10.0.100.0/24
        AvailabilityZone: us-west-2a
        VpcId: '{{ createVpc.vpcId }}'
      outputs:
        - Name: firstSubnetId
          Selector: $.Subnet.SubnetId
          Type: String
      nextStep: createDCSubnet2
    - name: createDCSubnet2
      action: aws:executeAwsApi
      onFailure: Abort
      inputs:
        Service: ec2
        Api: CreateSubnet
        CidrBlock: 10.0.101.0/24
        AvailabilityZone: us-west-2b
        VpcId: '{{ createVpc.vpcId }}'
      outputs:
        - Name: secondSubnetId
          Selector: $.Subnet.SubnetId
          Type: String
      nextStep: createDCSecGroup
    - name: createDCSecGroup
      action: aws:executeAwsApi
      onFailure: Abort
      inputs:
        Service: ec2
        Api: CreateSecurityGroup
        GroupName: SampleDCSecGroup
        Description: Security Group for Sample Domain Controllers
        VpcId: '{{ createVpc.vpcId }}'
      outputs:
        - Name: dcSecGroupId
          Selector: $.GroupId
          Type: String
```

```
      nextStep: authIngressDCTraffic
    - name: authIngressDCTraffic
      action: aws:executeAwsApi
      onFailure: Abort
      inputs:
        Service: ec2
        Api: AuthorizeSecurityGroupIngress
        GroupId: '{{ createDCSecGroup.dcSecGroupId }}'
        IpPermissions:
          - FromPort: -1
            IpProtocol: '-1'
            IpRanges:
              - CidrIp: 0.0.0.0/0
                Description: Allow all traffic between Domain Controllers
      nextStep: verifyInstanceProfile
    - name: verifyInstanceProfile
      action: aws:waitForAwsResourceProperty
      maxAttempts: 5
      onFailure: Abort
      inputs:
        Service: iam
        Api: ListInstanceProfilesForRole
        RoleName: sampleSSMInstanceRole
        PropertySelector: '$.InstanceProfiles[0].Arn'
        DesiredValues:
          - '{{ createSSMInstanceProfile.instanceProfileArn }}'
      nextStep: iamEventualConsistency
    - name: iamEventualConsistency
      action: aws:sleep
      inputs:
        Duration: PT2M
      nextStep: launchDC1
    - name: launchDC1
      action: aws:executeAwsApi
      onFailure: Abort
      inputs:
        Service: ec2
        Api: RunInstances
        BlockDeviceMappings:
          - DeviceName: /dev/sda1
            Ebs:
              DeleteOnTermination: true
              VolumeSize: 50
              VolumeType: gp2
```

```
            - DeviceName: xvdf
              Ebs:
                DeleteOnTermination: true
                VolumeSize: 100
                VolumeType: gp2
          IamInstanceProfile:
            Arn: '{{ createSSMInstanceProfile.instanceProfileArn }}'
          ImageId: '{{ getLatestWindowsAmi.amiId }}'
          InstanceType: t2.micro
          MaxCount: 1
          MinCount: 1
          PrivateIpAddress: 10.0.100.50
          SecurityGroupIds:
            - '{{ createDCSecGroup.dcSecGroupId }}'
          SubnetId: '{{ createDCSubnet1.firstSubnetId }}'
          TagSpecifications:
            - ResourceType: instance
              Tags:
                - Key: Name
                  Value: SampleDC1
        outputs:
          - Name: pdcInstanceId
            Selector: '$.Instances[0].InstanceId'
            Type: String
        nextStep: launchDC2
      - name: launchDC2
        action: aws:executeAwsApi
        onFailure: Abort
        inputs:
          Service: ec2
          Api: RunInstances
          BlockDeviceMappings:
            - DeviceName: /dev/sda1
              Ebs:
                DeleteOnTermination: true
                VolumeSize: 50
                VolumeType: gp2
            - DeviceName: xvdf
              Ebs:
                DeleteOnTermination: true
                VolumeSize: 100
                VolumeType: gp2
          IamInstanceProfile:
            Arn: '{{ createSSMInstanceProfile.instanceProfileArn }}'
```

```
        ImageId: '{{ getLatestWindowsAmi.amiId }}'
        InstanceType: t2.micro
        MaxCount: 1
        MinCount: 1
        PrivateIpAddress: 10.0.101.50
        SecurityGroupIds:
          - '{{ createDCSecGroup.dcSecGroupId }}'
        SubnetId: '{{ createDCSubnet2.secondSubnetId }}'
        TagSpecifications:
          - ResourceType: instance
            Tags:
              - Key: Name
                Value: SampleDC2
      outputs:
        - Name: adcInstanceId
          Selector: '$.Instances[0].InstanceId'
          Type: String
      nextStep: verifyDCInstanceState
    - name: verifyDCInstanceState
      action: aws:waitForAwsResourceProperty
      inputs:
        Service: ec2
        Api: DescribeInstanceStatus
        IncludeAllInstances: true
        InstanceIds:
          - '{{ launchDC1.pdcInstanceId }}'
          - '{{ launchDC2.adcInstanceId }}'
        PropertySelector: '$.InstanceStatuses[0].InstanceState.Name'
        DesiredValues:
          - running
      nextStep: verifyInstancesOnlineSSM
    - name: verifyInstancesOnlineSSM
      action: aws:waitForAwsResourceProperty
      timeoutSeconds: 600
      inputs:
        Service: ssm
        Api: DescribeInstanceInformation
        InstanceInformationFilterList:
          - key: InstanceIds
            valueSet:
              - '{{ launchDC1.pdcInstanceId }}'
              - '{{ launchDC2.adcInstanceId }}'
        PropertySelector: '$.InstanceInformationList[0].PingStatus'
        DesiredValues:
```

```
            - Online
        nextStep: installADRoles
      - name: installADRoles
        action: aws:runCommand
        inputs:
          DocumentName: AWS-RunPowerShellScript
          InstanceIds:
            - '{{ launchDC1.pdcInstanceId }}'
            - '{{ launchDC2.adcInstanceId }}'
          Parameters:
            commands: |-
              try {
                  Install-WindowsFeature -Name AD-Domain-Services -
IncludeManagementTools
              }
              catch {
                  Write-Error "Failed to install ADDS Role."
              }
        nextStep: setAdminPassword
      - name: setAdminPassword
        action: aws:runCommand
        inputs:
          DocumentName: AWS-RunPowerShellScript
          InstanceIds:
            - '{{ launchDC1.pdcInstanceId }}'
          Parameters:
            commands:
              - net user Administrator "sampleAdminPass123!"
        nextStep: createForest
      - name: createForest
        action: aws:runCommand
        inputs:
          DocumentName: AWS-RunPowerShellScript
          InstanceIds:
            - '{{ launchDC1.pdcInstanceId }}'
          Parameters:
            commands: |-
              $dsrmPass = 'sample123!' | ConvertTo-SecureString -asPlainText -Force
              try {
                  Install-ADDSForest -DomainName "sample.com" -DomainMode 6
 -ForestMode 6 -InstallDNS -DatabasePath "D:\NTDS" -SysvolPath "D:\SYSVOL" -
SafeModeAdministratorPassword $dsrmPass -Force
              }
              catch {
```

```
                Write-Error $_
            }
            try {
                Add-DnsServerForwarder -IPAddress "10.0.100.2"
            }
            catch {
                Write-Error $_
            }
        nextStep: associateDhcpOptions
      - name: associateDhcpOptions
        action: aws:executeAwsApi
        onFailure: Abort
        inputs:
          Service: ec2
          Api: AssociateDhcpOptions
          DhcpOptionsId: '{{ createDhcpOptions.dhcpOptionsId }}'
          VpcId: '{{ createVpc.vpcId }}'
        nextStep: waitForADServices
      - name: waitForADServices
        action: aws:sleep
        inputs:
          Duration: PT1M
        nextStep: promoteADC
      - name: promoteADC
        action: aws:runCommand
        inputs:
          DocumentName: AWS-RunPowerShellScript
          InstanceIds:
            - '{{ launchDC2.adcInstanceId }}'
          Parameters:
            commands: |-
              ipconfig /renew
              $dsrmPass = 'sample123!' | ConvertTo-SecureString -asPlainText -Force
              $domAdminUser = "sample\Administrator"
              $domAdminPass = "sampleAdminPass123!" | ConvertTo-SecureString -
asPlainText -Force
              $domAdminCred = New-Object
 System.Management.Automation.PSCredential($domAdminUser,$domAdminPass)

              try {
                  Install-ADDSDomainController -DomainName "sample.com" -InstallDNS
 -DatabasePath "D:\NTDS" -SysvolPath "D:\SYSVOL" -SafeModeAdministratorPassword
 $dsrmPass -Credential $domAdminCred -Force
              }
```

```
            catch {
                Write-Error $_
            }
```

JSON

```
   {
     "description": "Custom Automation Deployment Example",
     "schemaVersion": "0.3",
     "assumeRole": "{{ AutomationAssumeRole }}",
     "parameters": {
       "AutomationAssumeRole": {
         "type": "String",
         "description": "(Optional) The ARN of the role that allows Automation
 to perform the actions on your behalf. If no role is specified, Systems Manager
 Automation uses your IAM permissions to run this runbook.",
         "default": ""
       }
     },
     "mainSteps": [
       {
         "name": "getLatestWindowsAmi",
         "action": "aws:executeAwsApi",
         "onFailure": "Abort",
         "inputs": {
           "Service": "ssm",
           "Api": "GetParameter",
           "Name": "/aws/service/ami-windows-latest/Windows_Server-2016-English-
Full-Base"
         },
         "outputs": [
           {
             "Name": "amiId",
             "Selector": "$.Parameter.Value",
             "Type": "String"
           }
         ],
         "nextStep": "createSSMInstanceRole"
       },
       {
         "name": "createSSMInstanceRole",
         "action": "aws:executeAwsApi",
```

```
          "onFailure": "Abort",
          "inputs": {
            "Service": "iam",
            "Api": "CreateRole",
            "AssumeRolePolicyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":
[{\"Effect\":\"Allow\",\"Principal\":{\"Service\":[\"ec2.amazonaws.com\"]},\"Action
\":[\"sts:AssumeRole\"]}]}",
            "RoleName": "sampleSSMInstanceRole"
          },
          "nextStep": "attachManagedSSMPolicy"
        },
        {
          "name": "attachManagedSSMPolicy",
          "action": "aws:executeAwsApi",
          "onFailure": "Abort",
          "inputs": {
            "Service": "iam",
            "Api": "AttachRolePolicy",
            "PolicyArn": "arn:aws:iam::aws:policy/service-role/
AmazonSSMManagedInstanceCore",
            "RoleName": "sampleSSMInstanceRole"
          },
          "nextStep": "createSSMInstanceProfile"
        },
        {
          "name": "createSSMInstanceProfile",
          "action":"aws:executeAwsApi",
          "onFailure": "Abort",
          "inputs": {
            "Service": "iam",
            "Api": "CreateInstanceProfile",
            "InstanceProfileName": "sampleSSMInstanceRole"
          },
          "outputs": [
            {
              "Name": "instanceProfileArn",
              "Selector": "$.InstanceProfile.Arn",
              "Type": "String"
            }
          ],
          "nextStep": "addSSMInstanceRoleToProfile"
        },
        {
          "name": "addSSMInstanceRoleToProfile",
```

```
            "action": "aws:executeAwsApi",
            "onFailure": "Abort",
            "inputs": {
              "Service": "iam",
              "Api": "AddRoleToInstanceProfile",
              "InstanceProfileName": "sampleSSMInstanceRole",
              "RoleName": "sampleSSMInstanceRole"
            },
            "nextStep": "createVpc"
          },
          {
            "name": "createVpc",
            "action": "aws:executeAwsApi",
            "onFailure": "Abort",
            "inputs": {
              "Service": "ec2",
              "Api": "CreateVpc",
              "CidrBlock": "10.0.100.0/22"
            },
            "outputs": [
              {
                "Name": "vpcId",
                "Selector": "$.Vpc.VpcId",
                "Type": "String"
              }

            "nextStep": "getMainRtb"
          },
          {
            "name": "getMainRtb",
            "action": "aws:executeAwsApi",
            "onFailure": "Abort",
            "inputs": {
              "Service": "ec2",
              "Api": "DescribeRouteTables",
              "Filters": [
                {
                  "Name": "vpc-id",
                  "Values": ["{{ createVpc.vpcId }}"]
                }
              ]
            },
            "outputs": [
              {
```

```
            "Name": "mainRtbId",
            "Selector": "$.RouteTables[0].RouteTableId",
            "Type": "String"
          }
        ],
        "nextStep": "verifyMainRtb"
      },
      {
        "name": "verifyMainRtb",
        "action": "aws:assertAwsResourceProperty",
        "onFailure": "Abort",
        "inputs": {
          "Service": "ec2",
          "Api": "DescribeRouteTables",
          "RouteTableIds": ["{{ getMainRtb.mainRtbId }}"],
          "PropertySelector": "$.RouteTables[0].Associations[0].Main",
          "DesiredValues": ["True"]
        },
        "nextStep": "createPubSubnet"
      },
      {
        "name": "createPubSubnet",
        "action": "aws:executeAwsApi",
        "onFailure": "Abort",
        "inputs": {
          "Service": "ec2",
          "Api": "CreateSubnet",
          "CidrBlock": "10.0.103.0/24",
          "AvailabilityZone": "us-west-2c",
          "VpcId": "{{ createVpc.vpcId }}"
        },
        "outputs":[
          {
            "Name": "pubSubnetId",
            "Selector": "$.Subnet.SubnetId",
            "Type": "String"
          }
        ],
        "nextStep": "createPubRtb"
      },
      {
        "name": "createPubRtb",
        "action": "aws:executeAwsApi",
        "onFailure": "Abort",
```

```
      "inputs": {
        "Service": "ec2",
        "Api": "CreateRouteTable",
        "VpcId": "{{ createVpc.vpcId }}"
      },
      "outputs": [
        {
          "Name": "pubRtbId",
          "Selector": "$.RouteTable.RouteTableId",
          "Type": "String"
        }
      ],
      "nextStep": "createIgw"
    },
    {
      "name": "createIgw",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "CreateInternetGateway"
      },
      "outputs": [
        {
          "Name": "igwId",
          "Selector": "$.InternetGateway.InternetGatewayId",
          "Type": "String"
        }
      ],
      "nextStep": "attachIgw"
    },
    {
      "name": "attachIgw",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "AttachInternetGateway",
        "InternetGatewayId": "{{ createIgw.igwId }}",
        "VpcId": "{{ createVpc.vpcId }}"
      },
      "nextStep": "allocateEip"
    },
    {
```

```
      "name": "allocateEip",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "AllocateAddress",
        "Domain": "vpc"
      },
      "outputs": [
        {
          "Name": "eipAllocationId",
          "Selector": "$.AllocationId",
          "Type": "String"
        }
      ],
      "nextStep": "createNatGw"
    },
    {
      "name": "createNatGw",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "CreateNatGateway",
        "AllocationId": "{{ allocateEip.eipAllocationId }}",
        "SubnetId": "{{ createPubSubnet.pubSubnetId }}"
      },
      "outputs":[
        {
          "Name": "natGwId",
          "Selector": "$.NatGateway.NatGatewayId",
          "Type": "String"
        }
      ],
      "nextStep": "verifyNatGwAvailable"
    },
    {
      "name": "verifyNatGwAvailable",
      "action": "aws:waitForAwsResourceProperty",
      "timeoutSeconds": 150,
      "inputs": {
        "Service": "ec2",
        "Api": "DescribeNatGateways",
        "NatGatewayIds": [
```

```
            "{{ createNatGw.natGwId }}"
          ],
          "PropertySelector": "$.NatGateways[0].State",
          "DesiredValues": [
            "available"
          ]
        },
        "nextStep": "createNatRoute"
      },
      {
        "name": "createNatRoute",
        "action": "aws:executeAwsApi",
        "onFailure": "Abort",
        "inputs": {
          "Service": "ec2",
          "Api": "CreateRoute",
          "DestinationCidrBlock": "0.0.0.0/0",
          "NatGatewayId": "{{ createNatGw.natGwId }}",
          "RouteTableId": "{{ getMainRtb.mainRtbId }}"
        },
        "nextStep": "createPubRoute"
      },
      {
        "name": "createPubRoute",
        "action": "aws:executeAwsApi",
        "onFailure": "Abort",
        "inputs": {
          "Service": "ec2",
          "Api": "CreateRoute",
          "DestinationCidrBlock": "0.0.0.0/0",
          "GatewayId": "{{ createIgw.igwId }}",
          "RouteTableId": "{{ createPubRtb.pubRtbId }}"
        },
        "nextStep": "setPubSubAssoc"
      },
      {
        "name": "setPubSubAssoc",
        "action": "aws:executeAwsApi",
        "onFailure": "Abort",
        "inputs": {
          "Service": "ec2",
          "Api": "AssociateRouteTable",
          "RouteTableId": "{{ createPubRtb.pubRtbId }}",
          "SubnetId": "{{ createPubSubnet.pubSubnetId }}"
```

```
      }
    },
    {
      "name": "createDhcpOptions",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "CreateDhcpOptions",
        "DhcpConfigurations": [
          {
            "Key": "domain-name-servers",
            "Values": ["10.0.100.50,10.0.101.50"]
          },
          {
            "Key": "domain-name",
            "Values": ["sample.com"]
          }
        ]
      },
      "outputs": [
        {
          "Name": "dhcpOptionsId",
          "Selector": "$.DhcpOptions.DhcpOptionsId",
          "Type": "String"
        }
      ],
      "nextStep": "createDCSubnet1"
    },
    {
      "name": "createDCSubnet1",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "CreateSubnet",
        "CidrBlock": "10.0.100.0/24",
        "AvailabilityZone": "us-west-2a",
        "VpcId": "{{ createVpc.vpcId }}"
      },
      "outputs": [
        {
          "Name": "firstSubnetId",
          "Selector": "$.Subnet.SubnetId",
```

```
            "Type": "String"
          }
        ],
        "nextStep": "createDCSubnet2"
      },
      {
        "name": "createDCSubnet2",
        "action": "aws:executeAwsApi",
        "onFailure": "Abort",
        "inputs": {
          "Service": "ec2",
          "Api": "CreateSubnet",
          "CidrBlock": "10.0.101.0/24",
          "AvailabilityZone": "us-west-2b",
          "VpcId": "{{ createVpc.vpcId }}"
        },
        "outputs": [
          {
            "Name": "secondSubnetId",
            "Selector": "$.Subnet.SubnetId",
            "Type": "String"
          }
        ],
        "nextStep": "createDCSecGroup"
      },
      {
        "name": "createDCSecGroup",
        "action": "aws:executeAwsApi",
        "onFailure": "Abort",
        "inputs": {
          "Service": "ec2",
          "Api": "CreateSecurityGroup",
          "GroupName": "SampleDCSecGroup",
          "Description": "Security Group for Example Domain Controllers",
          "VpcId": "{{ createVpc.vpcId }}"
        },
        "outputs": [
          {
            "Name": "dcSecGroupId",
            "Selector": "$.GroupId",
            "Type": "String"
          }
        ],
        "nextStep": "authIngressDCTraffic"
```

```
        },
        {
          "name": "authIngressDCTraffic",
          "action": "aws:executeAwsApi",
          "onFailure": "Abort",
          "inputs": {
            "Service": "ec2",
            "Api": "AuthorizeSecurityGroupIngress",
            "GroupId": "{{ createDCSecGroup.dcSecGroupId }}",
            "IpPermissions": [
              {
                "FromPort": -1,
                "IpProtocol": "-1",
                "IpRanges": [
                  {
                    "CidrIp": "0.0.0.0/0",
                    "Description": "Allow all traffic between Domain Controllers"
                  }
                ]
              }
            ]
          },
          "nextStep": "verifyInstanceProfile"
        },
        {
          "name": "verifyInstanceProfile",
          "action": "aws:waitForAwsResourceProperty",
          "maxAttempts": 5,
          "onFailure": "Abort",
          "inputs": {
            "Service": "iam",
            "Api": "ListInstanceProfilesForRole",
            "RoleName": "sampleSSMInstanceRole",
            "PropertySelector": "$.InstanceProfiles[0].Arn",
            "DesiredValues": [
              "{{ createSSMInstanceProfile.instanceProfileArn }}"
            ]
          },
          "nextStep": "iamEventualConsistency"
        },
        {
          "name": "iamEventualConsistency",
          "action": "aws:sleep",
          "inputs": {
```

```
            "Duration": "PT2M"
          },
          "nextStep": "launchDC1"
        },
        {
          "name": "launchDC1",
          "action": "aws:executeAwsApi",
          "onFailure": "Abort",
          "inputs": {
            "Service": "ec2",
            "Api": "RunInstances",
            "BlockDeviceMappings": [
              {
                "DeviceName": "/dev/sda1",
                "Ebs": {
                  "DeleteOnTermination": true,
                  "VolumeSize": 50,
                  "VolumeType": "gp2"
                }
              },
              {
                "DeviceName": "xvdf",
                "Ebs": {
                  "DeleteOnTermination": true,
                  "VolumeSize": 100,
                  "VolumeType": "gp2"
                }
              }
            ],
            "IamInstanceProfile": {
              "Arn": "{{ createSSMInstanceProfile.instanceProfileArn }}"
            },
            "ImageId": "{{ getLatestWindowsAmi.amiId }}",
            "InstanceType": "t2.micro",
            "MaxCount": 1,
            "MinCount": 1,
            "PrivateIpAddress": "10.0.100.50",
            "SecurityGroupIds": [
              "{{ createDCSecGroup.dcSecGroupId }}"
            ],
            "SubnetId": "{{ createDCSubnet1.firstSubnetId }}",
            "TagSpecifications": [
              {
                "ResourceType": "instance",
```

```
            "Tags": [
              {
                "Key": "Name",
                "Value": "SampleDC1"
              }
            ]
          }
        ]
      },
      "outputs": [
        {
          "Name": "pdcInstanceId",
          "Selector": "$.Instances[0].InstanceId",
          "Type": "String"
        }
      ],
      "nextStep": "launchDC2"
    },
    {
      "name": "launchDC2",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "RunInstances",
        "BlockDeviceMappings": [
          {
            "DeviceName": "/dev/sda1",
            "Ebs": {
              "DeleteOnTermination": true,
              "VolumeSize": 50,
              "VolumeType": "gp2"
            }
          },
          {
            "DeviceName": "xvdf",
            "Ebs": {
              "DeleteOnTermination": true,
              "VolumeSize": 100,
              "VolumeType": "gp2"
            }
          }
        ],
        "IamInstanceProfile": {
```

```
            "Arn": "{{ createSSMInstanceProfile.instanceProfileArn }}"
          },
          "ImageId": "{{ getLatestWindowsAmi.amiId }}",
          "InstanceType": "t2.micro",
          "MaxCount": 1,
          "MinCount": 1,
          "PrivateIpAddress": "10.0.101.50",
          "SecurityGroupIds": [
            "{{ createDCSecGroup.dcSecGroupId }}"
          ],
          "SubnetId": "{{ createDCSubnet2.secondSubnetId }}",
          "TagSpecifications": [
            {
              "ResourceType": "instance",
              "Tags": [
                {
                  "Key": "Name",
                  "Value": "SampleDC2"
                }
              ]
            }
          ]
        },
        "outputs": [
          {
            "Name": "adcInstanceId",
            "Selector": "$.Instances[0].InstanceId",
            "Type": "String"
          }
        ],
        "nextStep": "verifyDCInstanceState"
      },
      {
        "name": "verifyDCInstanceState",
        "action": "aws:waitForAwsResourceProperty",
        "inputs": {
          "Service": "ec2",
          "Api": "DescribeInstanceStatus",
          "IncludeAllInstances": true,
          "InstanceIds": [
            "{{ launchDC1.pdcInstanceId }}",
            "{{ launchDC2.adcInstanceId }}"
          ],
          "PropertySelector": "$.InstanceStatuses[0].InstanceState.Name",
```

```
            "DesiredValues": [
              "running"
            ]
          },
          "nextStep": "verifyInstancesOnlineSSM"
        },
        {
          "name": "verifyInstancesOnlineSSM",
          "action": "aws:waitForAwsResourceProperty",
          "timeoutSeconds": 600,
          "inputs": {
            "Service": "ssm",
            "Api": "DescribeInstanceInformation",
            "InstanceInformationFilterList": [
              {
                "key": "InstanceIds",
                "valueSet": [
                  "{{ launchDC1.pdcInstanceId }}",
                  "{{ launchDC2.adcInstanceId }}"
                ]
              }
            ],
            "PropertySelector": "$.InstanceInformationList[0].PingStatus",
            "DesiredValues": [
              "Online"
            ]
          },
          "nextStep": "installADRoles"
        },
        {
          "name": "installADRoles",
          "action": "aws:runCommand",
          "inputs": {
            "DocumentName": "AWS-RunPowerShellScript",
            "InstanceIds": [
              "{{ launchDC1.pdcInstanceId }}",
              "{{ launchDC2.adcInstanceId }}"
            ],
            "Parameters": {
              "commands": [
                "try {",
                "  Install-WindowsFeature -Name AD-Domain-Services -
IncludeManagementTools",
                "}",
```

```
                "catch {",
                "  Write-Error \"Failed to install ADDS Role.\"",
                "}"
              ]
            }
          },
          "nextStep": "setAdminPassword"
        },
        {
          "name": "setAdminPassword",
          "action": "aws:runCommand",
          "inputs": {
            "DocumentName": "AWS-RunPowerShellScript",
            "InstanceIds": [
              "{{ launchDC1.pdcInstanceId }}"
            ],
            "Parameters": {
              "commands": [
                "net user Administrator \"sampleAdminPass123!\""
              ]
            }
          },
          "nextStep": "createForest"
        },
        {
          "name": "createForest",
          "action": "aws:runCommand",
          "inputs": {
            "DocumentName": "AWS-RunPowerShellScript",
            "InstanceIds": [
              "{{ launchDC1.pdcInstanceId }}"
            ],
            "Parameters": {
              "commands": [
                "$dsrmPass = 'sample123!' | ConvertTo-SecureString -asPlainText -
Force",
                "try {",
                "   Install-ADDSForest -DomainName \"sample.com\" -DomainMode 6 -
ForestMode 6 -InstallDNS -DatabasePath \"D:\\NTDS\" -SysvolPath \"D:\\SYSVOL\" -
SafeModeAdministratorPassword $dsrmPass -Force",
                "}",
                "catch {",
                "   Write-Error $_",
                "}",
```

```
                "try {",
                "   Add-DnsServerForwarder -IPAddress \"10.0.100.2\"",
                "}",
                "catch {",
                "   Write-Error $_",
                "}"
              ]
            }
          },
          "nextStep": "associateDhcpOptions"
        },
        {
          "name": "associateDhcpOptions",
          "action": "aws:executeAwsApi",
          "onFailure": "Abort",
          "inputs": {
            "Service": "ec2",
            "Api": "AssociateDhcpOptions",
            "DhcpOptionsId": "{{ createDhcpOptions.dhcpOptionsId }}",
            "VpcId": "{{ createVpc.vpcId }}"
          },
          "nextStep": "waitForADServices"
        },
        {
          "name": "waitForADServices",
          "action": "aws:sleep",
          "inputs": {
            "Duration": "PT1M"
          },
          "nextStep": "promoteADC"
        },
        {
          "name": "promoteADC",
          "action": "aws:runCommand",
          "inputs": {
            "DocumentName": "AWS-RunPowerShellScript",
            "InstanceIds": [
              "{{ launchDC2.adcInstanceId }}"
            ],
            "Parameters": {
              "commands": [
                "ipconfig /renew",
                "$dsrmPass = 'sample123!' | ConvertTo-SecureString -asPlainText -
Force",
```

```
                "$domAdminUser = \"sample\\Administrator\"",
                "$domAdminPass = \"sampleAdminPass123!\" | ConvertTo-SecureString -
asPlainText -Force",
                "$domAdminCred = New-Object
 System.Management.Automation.PSCredential($domAdminUser,$domAdminPass)",
                "try {",
                "   Install-ADDSDomainController -DomainName \"sample.com
\" -InstallDNS -DatabasePath \"D:\\NTDS\" -SysvolPath \"D:\\SYSVOL\" -
SafeModeAdministratorPassword $dsrmPass -Credential $domAdminCred -Force",
                "}",
                "catch {",
                "   Write-Error $_",
                "}"
              ]
            }
          }
        }
      ]
    }
```

Kembalikan volume root dari snapshot terbaru

Sistem operasi pada volume akar dapat rusak karena berbagai alasan. Misalnya, setelah operasi patching, instance mungkin gagal boot dengan sukses karena kernel atau registri rusak. Mengotomatiskan tugas pemecahan masalah umum, seperti memulihkan volume root dari snapshot terbaru yang diambil sebelum operasi patch, dapat mengurangi waktu henti dan mempercepat upaya pemecahan masalah Anda. AWS Systems Manager Tindakan otomatisasi dapat membantu Anda mencapai hal ini. Otomatisasi adalah kemampuan AWS Systems Manager.

Contoh AWS Systems Manager runbook berikut melakukan tindakan ini:

- Menggunakan `aws:executeAwsApi` tindakan otomatisasi untuk mengambil detail dari volume akar instans.
- Menggunakan `aws:executeScript` tindakan otomatisasi untuk mengambil snapshot terbaru untuk volume akar.
- Menggunakan `aws:branch` otomatisasi tindakan untuk melanjutkan otomatisasi jika snapshot ditemukan untuk volume akar.

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"
               }
           }
       }
   }
```

Buat AMI dan salinan lintas wilayah

Membuat Amazon Machine Image (AMI) dari sebuah instans adalah proses umum yang digunakan dalam backup dan recovery. Anda juga dapat memilih untuk menyalin AMI ke yang lain Wilayah AWS sebagai bagian dari arsitektur pemulihan bencana. Mengotomatiskan tugas pemeliharaan umum dapat mengurangi waktu henti jika masalah memerlukan failover. AWS Systems Manager Tindakan otomatisasi dapat membantu Anda mencapai hal ini. Otomatisasi adalah kemampuan AWS Systems Manager.

Contoh AWS Systems Manager runbook berikut melakukan tindakan ini:

- Menggunakan `aws:executeAwsApi` tindakan otomatisasi untuk membuat AMI.
- Menggunakan `aws:waitForAwsResourceProperty` tindakan otomatisasi untuk mengonfirmasi ketersediaan AMI.
- Menggunakan `aws:executeScript` tindakan otomatisasi untuk menyalin AMI ke wilayah tujuan.

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"
              }
          }
      }
  }
```

Membuat parameter masukan yang mengisiAWSsumber daya

Otomatisasi, sebuah kemampuan Systems Manager, mengisiAWSsumber daya diAWS Management Consoleyang cocok dengan jenis sumber daya yang Anda tentukan untuk parameter input. Sumber daya di sumber daya AndaAkun AWSyang cocok dengan jenis sumber daya ditampilkan dalam daftar dropdown untuk Anda pilih. Anda dapat menentukan jenis parameter input untuk Instans Amazon Elastic Compute Cloud (Amazon EC2), dan bucket Amazon Simple Storage Service (Amazon S3), danAWS Identity and Access Management(IAM) peran. Definisi tipe yang didukung dan ekspresi reguler yang digunakan untuk menemukan sumber daya yang cocok adalah sebagai berikut:

- `AWS::EC2::Instance::Id` - `^m?i-[a-z0-9]{8,17}$`
- `List<AWS::EC2::Instance::Id>` - `^m?i-[a-z0-9]{8,17}$`
- `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/.*$`
- `List<AWS::IAM::Role::Arn>` - `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):iam::[0-9]{12}:role/.*$`

Berikut ini adalah contoh dari jenis parameter masukan didefinisikan dalam konten 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
      }
   ]
}
```

Menggunakan Document Builder untuk membuat runbook

Jika runbook AWS Systems Manager publik tidak mendukung semua tindakan yang ingin Anda lakukan pada AWS sumber daya Anda, Anda dapat membuat runbook sendiri. Untuk membuat runbook kustom, Anda dapat membuat file format YAMAL atau JSON lokal secara manual dengan tindakan otomatisasi yang sesuai. Atau, Anda dapat menggunakan Document Builder di konsol Automation Systems Manager untuk membuat runbook kustom.

Menggunakan Document Builder, Anda dapat menambahkan tindakan otomatisasi ke runbook kustom Anda dan memberikan parameter yang diperlukan tanpa harus menggunakan sintaks JSON atau YAMAL. Setelah Anda menambahkan langkah-langkah dan membuat runbook, sistem mengubah tindakan yang telah ditambahkan ke dalam format YAML yang dapat digunakan Systems Manager untuk menjalankan otomatisasi.

Runbooks mendukung penggunaan penurunan harga, bahasa markup, yang memungkinkan Anda menambahkan deskripsi gaya wiki ke runbook dan langkah-langkah individual dalam runbook. Untuk informasi lebih lanjut tentang penggunaan penurunan harga, lihat Menggunakan Markdown di AWS.

Buat runbook menggunakan Document Builder

Sebelum Anda memulai

Kami menyarankan Anda membaca tentang berbagai tindakan yang dapat Anda gunakan dalam runbook. Untuk informasi selengkapnya, lihat Referensi tindakan Otomatisasi Systems Manager.

Untuk membuat runbook menggunakan Pembuat Dokumen

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.
3. Pilih Buat otomatisasi.
4. Untuk Nama, masukkan nama deskriptif untuk runbook.
5. Untuk Deskripsi dokumen, berikan deskripsi gaya penurunan harga untuk runbook. Anda dapat memberikan petunjuk untuk menggunakan runbook, langkah-langkah bernomor, atau jenis informasi lainnya untuk menggambarkan runbook. Lihat teks default untuk informasi tentang memformat konten Anda.

   > **Tip**
   >
   > Alihkan antara Sembunyikan pratinjau dan Tampilkan pratinjau untuk melihat seperti apa konten deskripsi Anda yang Anda tulis.

6. (Opsional) Untuk Peran asumsi, masukkan nama atau ARN peran layanan untuk melakukan tindakan atas nama Anda. Jika Anda tidak menentukan peran, Otomatisasi menggunakan izin akses pengguna yang menjalankan otomatisasi.

   > **Important**
   >
   > Untuk runbook yang tidak dimiliki oleh Amazon yang menggunakan `aws:executeScript` tindakan, peran harus ditentukan. Untuk informasi, lihat Izin untuk menggunakan runbook.

7. (Opsional) Untuk Keluaran, masukkan output apapun untuk otomatisasi runbook ini supaya bisa melakukan proses lainnya.

Misalnya, jika runbook Anda membuat AMI baru, Anda dapat menentukan [". CreateImage ImageId"], dan kemudian gunakan output ini untuk membuat instance baru dalam otomatisasi berikutnya.

8. (Opsional) Perluas bagian Parameter input dan lakukan hal-hal berikut ini.

   1. Untuk Nama parameter, masukkan nama deskriptif untuk parameter runbook yang Anda buat.
   2. Untuk Jenis, pilih jenis untuk parameter, seperti `String` atau `MapList`.
   3. Untuk Yang dibutuhkan, lakukan salah satu hal berikut ini:
      - Pilih Ya jika nilai untuk parameter runbook ini harus diberikan pada saat waktu aktif.
      - Pilih Tidak jika parameter tidak diperlukan, dan masukkan nilai parameter default di Nilai default (opsional).
   4. Di Deskripsi, masukkan deskripsi untuk parameter runbook.

   > **Note**
   >
   > Untuk menambahkan parameter runbook lainnya, pilih Tambahkan parameter. Untuk menghapus parameter runbook, pilih tombol X (Hapus).

9. (Opsional) Perluas Jenis target dan pilih jenis target untuk menentukan lokasi jenis sumber daya otomatisasi dapat berjalan. Misalnya, untuk menggunakan runbook pada instans EC2, pilih `/AWS::EC2::Instance`.

   > **Note**
   >
   > Jika Anda menentukan nilai '/', runbook dapat berjalan pada semua jenis sumber daya. Untuk daftar jenis sumber daya yang valid, lihat AWS Referensi Jenis Sumber Daya di AWS CloudFormation Panduan Pengguna.

10. Perluas bagian Tag dokumen dan masukkan satu tag kunci-nilai pasangan atau lebih untuk diterapkan ke runbook (opsional). Tag membuatnya lebih mudah untuk mengidentifikasi, mengatur, dan mencari sumber daya. Untuk informasi selengkapnya, lihat Menandai dokumen Systems Manager.
11. Di bagian Langkah 1, berikan informasi berikut.

    - Untuk Nama langkah, masukkan nama deskriptif untuk langkah pertama otomatisasi.

- Untuk Jenis tindakan, pilih jenis tindakan yang akan digunakan untuk langkah ini.

  Untuk daftar dan informasi tentang jenis tindakan yang tersedia, lihat Referensi tindakan Otomatisasi Systems Manager.
- Untuk Deskripsi, masukkan deskripsi alangkah otomatisasi. Anda dapat menggunakan penurunan harga untuk memformat teks Anda.
- Tergantung pada Jenis tindakan yang dipilih, masukkan input yang diperlukan untuk jenis tindakan di bagian Input langkah. Misalnya, jika Anda memilih tindakan `aws:approve`, Anda harus menentukan nilai untuk `Approvers` properti.

  Untuk informasi tentang bidang input langkah, lihat entri di Referensi tindakan Otomatisasi Systems Manager untuk jenis tindakan yang Anda pilih. Sebagai contoh: `aws:executeStateMachine` – Jalankan AWS Step Functions mesin status.
- (Opsional) Untuk Input tambahan, berikan nilai masukan tambahan yang diperlukan untuk runbook Anda. Jenis input yang tersedia bergantung pada jenis tindakan yang Anda pilih untuk langkah tersebut. (Perhatikan bahwa beberapa jenis tindakan memerlukan nilai input.)

  > **Note**
  >
  > Untuk menambahkan input lainnya, pilih Tambahkan input opsional. Untuk menghapus input, pilih tombol X (Hapus).
- (Opsional) Untuk Output, masukkan output apapun untuk langkah ini supaya bisa melakukan proses lainnya.

  > **Note**
  >
  > Output tidak tersedia untuk semua jenis tindakan.
- Perluas bagian Properti umum (opsional) dan tentukan properti untuk tindakan yang umum di semua tindakan otomatisasi. Misalnya, untuk detik waktu habis, Anda dapat memberikan nilai dalam hitungan detik untuk menentukan berapa lama langkah dapat berjalan sebelum berhenti.

  Untuk informasi selengkapnya, lihat Properti dibagi oleh semua tindakan.

> **Note**
>
> Untuk menambahkan langkah lainnya, pilih Tambahkan langkah dan ulangi prosedur untuk membuat langkah. Untuk menghapus langkah, pilih Hapus langkah.

12. Pilih Buat Otomatisasi untuk menyimpan runbook.

Buat runbook yang menjalankan skrip

Prosedur berikut menunjukkan cara menggunakan Document Builder di konsol AWS Systems Manager Automation untuk membuat runbook kustom yang menjalankan skrip.

Langkah pertama dari runbook yang Anda buat menjalankan skrip untuk meluncurkan instans Amazon Elastic Compute Cloud (Amazon EC2). Langkah kedua menjalankan skrip lain untuk memantau pemeriksaan status instans untuk mengubah ke `ok`. Kemudian, status keseluruhan `Success` dilaporkan untuk otomatisasi.

Sebelum Anda memulai

Pastikan Anda telah menyelesaikan langkah-langkah berikut:

- Verifikasi bahwa Anda memiliki hak administrator, atau bahwa Anda telah diberikan izin yang sesuai untuk mengakses Systems Manager in AWS Identity and Access Management (IAM).

  Untuk informasi, lihat Memverifikasi akses pengguna untuk runbook.
- Verifikasi bahwa Anda memiliki peran layanan IAM untuk Otomatisasi (juga dikenal sebagai peran asumsi) di Akun AWS. Peran ini diperlukan karena panduan ini menggunakan `aws:executeScript` tindakan.

  Untuk informasi lebih lanjut tentang pembuatan peran, lihat Mengonfigurasi akses peran layanan (peran asumsi) untuk otomatisasi.

  Untuk informasi tentang persyaratan peran layanan IAM untuk menjalanka `naws:executeScript`, lihat Izin untuk menggunakan runbook.
- Verifikasi bahwa Anda memiliki izin untuk meluncurkan Instans EC2.

  Untuk informasi lebih lanjut, lihat IAM dan Amazon EC2 dalam Panduan Pengguna Amazon EC2 untuk Instans Linux.

Untuk membuat runbook kustom yang menjalankan skrip menggunakan Document Builder

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.
3. Pilih Buat otomatisasi.
4. Untuk Nama, ketik nama deskriptif ini untuk runbook:. **`LaunchInstanceAndCheckStatus`**
5. (Opsional) Untuk Deskripsi dokumen, ganti teks default dengan deskripsi untuk runbook ini, menggunakan Markdown. Berikut adalah contohnya.

```
##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. Untuk Peran asumsi, masukkan ARN peran layanan IAM untuk Otomatisasi (Peran asumsi) untuk otomatisasi, dalam format **`arn:aws:iam::111122223333:role/AutomationServiceRole`**. Ganti Akun AWS ID Anda dengan 111122223333.

   Peran yang Anda tentukan digunakan untuk memberikan izin yang diperlukan untuk memulai otomatisasi.

> **Important**
>
> Untuk runbook yang tidak dimiliki oleh Amazon yang menggunakan `aws:executeScript` tindakan, peran harus ditentukan. Untuk informasi, lihat Izin untuk menggunakan runbook.

7. Perluas Parameter input dan lakukan hal berikut.

   1. Untuk Nama parameter, masukkan **`imageId`**.
   2. Untuk Jenis, pilih **`String`**.
   3. Untuk yang diperlukan, pilih No.
   4. Untuk Nilai default, masukkan berikut ini.

      ```
      {{ ssm:/aws/service/ami-amazon-linux-latest/amzn-ami-hvm-x86_64-gp2 }}
      ```

      > **Note**
      >
      > Nilai ini meluncurkan instans Amazon EC2 menggunakan ID Amazon Linux Amazon Machine Image 1 AMI () terbaru. Jika Anda ingin menggunakan yang berbeda AMI, ganti nilai dengan AMI ID.

   5. Untuk Deskripsi, masukkan yang berikut ini.

      ```
      (Optional) The AMI ID to use for launching the instance. The default value uses
       the latest released Amazon Linux AMI ID.
      ```

8. Pilih Tambahkan parameter untuk membuat parameter kedua, **`tagValue`**, dan masukkan yang berikut ini.

   1. Untuk Nama parameter, masukkan **`tagValue`**.
   2. Untuk Jenis, pilih **`String`**.
   3. Untuk yang diperlukan, pilih No.
   4. Untuk Nilai default, masukkan **`LaunchedBySsmAutomation`**. Ini menambahkan nilai kunci-pasangan tag `Name:LaunchedBySsmAutomation` ke instans.
   5. Untuk Deskripsi, masukkan yang berikut ini.

```
(Optional) The tag value to add to the instance. The default value is
 LaunchedBySsmAutomation.
```

9. Pilih Tambahkan parameter untuk membuat parameter ketiga, **`instanceType`**, dan masukkan informasi berikut ini.

    1. Untuk Nama parameter, masukkan **`instanceType`**.
    2. Untuk Jenis, pilih **`String`**.
    3. Untuk yang diperlukan, pilih No.
    4. Untuk Nilai default, masukkan **`t2.micro`**.
    5. Untuk Deskripsi parameter, masukkan yang berikut ini.

    ```
    (Optional) The instance type to use for the instance. The default value is
     t2.micro.
    ```

10. Perluas Jenis target dan pilih **`"/"`**.
11. (Opsional) Perluas Tag dokumen untuk menerapkan tag sumber daya ke runbook Anda. Untuk Kunci tag, masukkan **`Purpose`**, dan untuk Nilai tag, masukkan **`LaunchInstanceAndCheckState`**.
12. Di bagian Langkah 1, selesaikan langkah berikut.

    1. Untuk Nama langkah, masukkan nama langkah deskriptif untuk langkah pertama otomatisasi: **`LaunchEc2Instance`**.
    2. Untuk Jenis tindakan, pilih Jalankan skrip (**`aws:executeScript`**).
    3. Untuk Deskripsi, masukkan deskripsi langkah otomatisasi, seperti berikut.

    ```
    **About This Step**

       This step first launches an EC2 instance using the ```aws:executeScript```
     action and the provided script.
    ```

    4. Perluas input.
    5. Untuk Waktu aktif, pilih bahasa waktu aktif untuk digunakan dalam menjalankan skrip yang disediakan.
    6. Untuk Handler, masukkan **`launch_instance`**. Ini adalah nama fungsi yang dinyatakan dalam skrip berikut.

> **Note**
>
> Ini tidak diperlukan untuk PowerShell.

7. Untuk Skrip, ganti isi default dengan berikut ini. Pastikan untuk mencocokkan skrip dengan nilai waktu aktif yang sesuai.

   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. Perluas Input tambahan.
9. Untuk nama Input, pilih InputPayload. Untuk Nilai input, masukkan data YAML berikut.

```
image_id: "{{ imageId }}"
    tag_value: "{{ tagValue }}"
    instance_type: "{{ instanceType }}"
```

13. Perluas Output dan lakukan hal berikut:
    - Untuk Nama, masukkan **`payload`**.
    - Untuk Pemilih, masukkan **`$.Payload`**.
    - Untuk Jenis, pilih `StringMap`.
14. Pilih Tambahkan langkah untuk menambahkan langkah kedua untuk runbook. Langkah kedua mengkueri status instans yang diluncurkan pada Langkah 1 dan menunggu sampai status kembali ok.
15. Di bagian Langkah 2, lakukan hal berikut.
    1. Untuk Nama langkah, masukkan nama deskriptif ini untuk langkah kedua otomatisasi: **`WaitForInstanceStatusOk`**.
    2. Untuk Jenis tindakan, pilih Jalankan skrip (**`aws:executeScript`**).
    3. Untuk Deskripsi, masukkan deskripsi langkah otomatisasi, seperti berikut.

```
**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. Untuk Waktu aktif, pilih bahasa waktu aktif yang akan digunakan untuk mengeksekusi skrip yang disediakan.
5. Untuk Handler, masukkan **`poll_instance`**. Ini adalah nama fungsi yang dinyatakan dalam skrip berikut.

> **Note**
>
> Ini tidak diperlukan untuk PowerShell.

6. Untuk Skrip, ganti isi default dengan berikut ini. Pastikan untuk mencocokkan skrip dengan nilai waktu aktif yang sesuai.

   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. Perluas Input tambahan.
8. Untuk nama Input, pilih InputPayload. Untuk Nilai input, masukkan berikut ini:

```
{{ LaunchEc2Instance.payload }}
```

16. Pilih Buat Otomatisasi untuk menyimpan runbook.

Menggunakan skrip di runbook

Otomatisasi runbook mendukung menjalankan skrip sebagai bagian dari otomatisasi. Otomatisasi adalah kemampuan AWS Systems Manager. Dengan menggunakan runbook, Anda dapat menjalankan skrip secara langsung di AWS tanpa membuat lingkungan komputasi terpisah untuk

menjalankan skrip Anda. Karena runbook dapat menjalankan langkah-langkah skrip bersama dengan jenis langkah otomatisasi lainnya, seperti persetujuan, Anda dapat berpartisipasi langsung dalam situasi kritis atau ambigu. Anda dapat mengirim output dari `aws:executeScript` tindakan di runbook Anda ke Amazon CloudWatch Logs. Untuk informasi selengkapnya, lihat Pencatatan output tindakan Otomatisasi dengan CloudWatch Logs.

Izin untuk menggunakan runbook

Untuk menggunakan runbook, Systems Manager harus menggunakan izin AWS Identity and Access Management (IAM) role. Metode yang menggunakan otomatisasi untuk menentukan peran izin yang digunakan tergantung pada beberapa faktor, dan apakah langkah menggunakan `aws:executeScript` tindakan.

Untuk runbook yang tidak menggunakan `aws:executeScript`, Otomatisasi menggunakan salah satu dari dua sumber izin:

- Izin peran layanan IAM, atau Peran asumsi, yang ditentukan dalam runbook atau diteruskan sebagai parameter.
- Jika tidak ada peran layanan IAM yang ditentukan, izin pengguna yang memulai otomatisasi.

Namun, ketika langkah dalam runbook menyertakan `aws:executeScript` tindakan, peran layanan IAM (Asumsikan peran) selalu diperlukan jika Python atau PowerShell skrip yang ditentukan untuk tindakan tersebut memanggil operasi API apa pun. AWS Otomatisasi memeriksa peran ini dalam urutan berikut:

- Izin peran layanan IAM, atau Peran asumsi, yang ditentukan dalam runbook atau diteruskan sebagai parameter.
- Jika tidak ada peran yang ditemukan, Automation mencoba menjalankan Python atau PowerShell skrip yang ditentukan untuk `aws:executeScript` tanpa izin apa pun. Jika skrip memanggil AWS Operasi API (misalnya Amazon EC2 `CreateImage`), operasi atau mencoba untuk bertindak pada AWS sumber daya (seperti instans EC2), langkah yang berisi skrip gagal, dan Systems Manager mengembalikan pesan kesalahan yang melaporkan kegagalan.

Menambahkan skrip ke runbook

Anda dapat menambahkan skrip untuk runbook dengan memasukkan skrip sebaris sebagai bagian dari langkah dalam runbook. Anda juga dapat melampirkan skrip ke runbook dengan mengunggah skrip dari mesin lokal atau dengan menentukan bucket Amazon Simple Storage Service (Amazon

S3) tempat skrip berada. Setelah langkah yang menjalankan skrip selesai, output dari skrip tersedia sebagai objek JSON, yang kemudian dapat Anda gunakan sebagai masukan untuk langkah-langkah berikutnya dalam runbook Anda.

Kendala skrip untuk runbook

Runbook menetapkan batas lima lampiran file. Skrip dapat berupa skrip Python (.py), skrip Core (.ps1), PowerShell atau dilampirkan sebagai konten dalam file.zip.

Menggunakan pernyataan bersyarat di runbook

Secara default, langkah-langkah yang Anda tentukan di bagian `mainSteps` runbook yang dijalankan secara berurutan. Setelah satu tindakan selesai, tindakan berikutnya ditentukan dalam `mainSteps` bagian dimulai. Selain itu, jika tindakan gagal dijalankan, seluruh otomatisasi gagal (secara default). Anda dapat menggunakan `aws:branch` tindakan otomasi dan opsi runbook yang dijelaskan di bagian ini untuk membuat otomatisasi yang berkinerja Percabangan bersyarat. Buat otomatisasi yang melompat ke langkah yang berbeda setelah mengevaluasi pilihan yang berbeda atau yang secara dinamis menanggapi perubahan ketika langkah selesai. Berikut adalah daftar opsi yang dapat Anda gunakan untuk membuat otomatisasi dinamis:

- **`aws:branch`**: Tindakan ini mengizinkan Anda membuat otomatisasi dinamis yang mengevaluasi pilihan yang berbeda dalam satu langkah dan kemudian melompat ke langkah di runbook yang berbeda berdasarkan hasil evaluasi tersebut.
- **`nextStep`**: Opsi ini menentukan langkah otomatisasi mana yang harus diproses setelah berhasil menyelesaikan langkah.
- **`isEnd`**: Opsi ini menghentikan otomatisasi pada akhir langkah tertentu. Nilai default opsi ini adalah palsu.
- **`isCritical`**: Opsi ini menunjukkan langkah sebagai kepentingan untuk berhasil menyelesaikan otomatisasi. Jika langkah dengan penunjukan ini gagal, maka Otomatisasi melaporkan status akhir otomatisasi sebagai `Failed`. Nilai default opsi ini adalah `true`.
- **`onFailure`**: Opsi ini menunjukkan apakah otomatisasi harus berhenti, melanjutkan, atau meneruskan ke langkah kegagalan yang berbeda. Nilai default untuk opsi ini adalah batalkan.

Bagian berikut menjelaskan `aws:branch` tindakan otomatisi. Untuk informasi lebih lanjut tentang pilihan `nextStep`, `isEnd`, `isCritical`, dan `onFailure`, lihat Contoh `aws:branch` runbook.

Bekerja dengan **aws:branch** tindakan

Tindakan `aws:branch` tersebut menawarkan opsi percabangan bersyarat yang paling dinamis untuk otomatisasi. Seperti disebutkan sebelumnya, tindakan ini mengizinkan otomatisasi Anda untuk mengevaluasi beberapa kondisi dalam satu langkah dan kemudian melompat ke langkah baru berdasarkan hasil evaluasi tersebut. Tindakan `aws:branch` berfungsi seperti `IF-ELIF-ELSE` pernyataan dalam pemrograman.

Berikut adalah contoh YAML dari `aws:branch` langkah.

```
- name: ChooseOSforCommands
  action: aws:branch
  inputs:
    Choices:
    - NextStep: runPowerShellCommand
      Variable: "{{GetInstance.platform}}"
      StringEquals: Windows
    - NextStep: runShellCommand
      Variable: "{{GetInstance.platform}}"
      StringEquals: Linux
    Default:
      PostProcessing
```

Bila Anda menentukan `aws:branch` tindakan untuk sebuah langkah, Anda menentukan `Choices` bahwa otomatisasi harus mengevaluasi. Otomatisasi dapat mengevaluasi `Choices` berdasarkan nilai parameter yang Anda tentukan dalam `Parameters` bagian runbook tersebut. Otomatisasi juga dapat mengevaluasi `Choices` berdasarkan output dari langkah sebelumnya.

Otomatisasi mengevaluasi setiap pilihan dengan menggunakan ekspresi Boolean. Jika evaluasi menentukan bahwa pilihan pertama adalah `true`, maka otomatisasi melompat ke langkah yang ditetapkan di pilihan tersebut. Jika evaluasi menentukan bahwa pilihan pertama adalah `false`, maka otomatisasi mengevaluasi pilihan berikutnya. Jika langkah Anda mencakup tiga atau lebih `Choices`, maka otomatisasi menilai setiap pilihan dalam urutan yang tepat sehingga otomatisasi menilai pilihan yang `true`. Selanjutnya, otomatisasi melompat ke langkah yang ditetapkan untuk pilihan `true` tersebut.

Jika tidak ada pilihan `Choices` yang `true`, otomatisasi memeriksa untuk melihat apakah langkah berisi `Default` nilai. Nilai `Default` menentukan langkah yang harus dilakukan otomatisasi jika tidak ada pilihan yang `true`. Jika tidak ada `Default` nilai yang ditentukan untuk langkah, otomatisasi akan memproses langkah berikutnya dalam runbook.

Berikut adalah `aws:branch` langkah dalam YAMAL bernama SfromParameterChooSEO. Langkahnya mencakup dua `Choices`: (`NextStep: runWindowsCommand`) dan (`NextStep: runLinuxCommand`). Otomatisasi mengevaluasi ini `Choices` untuk menentukan perintah mana yang dijalankan untuk sistem operasi yang sesuai. `Variable` untuk setiap pilihan menggunakan `{{OSName}}`, yang merupakan parameter yang ditentukan oleh penulis runbook `Parameters` di bagian runbook.

```
mainSteps:
- name: chooseOSfromParameter
  action: aws:branch
  inputs:
    Choices:
    - NextStep: runWindowsCommand
      Variable: "{{OSName}}"
      StringEquals: Windows
    - NextStep: runLinuxCommand
      Variable: "{{OSName}}"
      StringEquals: Linux
```

Berikut adalah `aws:branch` langkah dalam YAMAL bernama SfromOutputChooSEO. Langkahnya mencakup dua `Choices`: (`NextStep: runPowerShellCommand`) dan (`NextStep: runShellCommand`). Otomatisasi mengevaluasi ini `Choices` untuk menentukan perintah mana yang dijalankan untuk sistem operasi yang sesuai. `Variable` untuk setiap pilihan menggunakan `{{GetInstance.platform}}`, yang merupakan output dari langkah sebelumnya di runbook. Contoh ini juga mencakup opsi yang disebut `Default`. Jika otomatisasi mengevaluasi keduanya `Choices`, dan tidak ada pilihan `true`, maka otomatisasi melompat ke langkah yang bernama `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
```

Membuat **aws:branch** langkah dalam runbook

Saat Anda membuat `aws:branch` langkah dalam runbook, Anda menentukan `Choices` otomatisasi harus mengevaluasi untuk menentukan langkah mana lagi yang harus dilalui otomatisasi. Seperti disebutkan sebelumnya, `Choices` dievaluasi dengan menggunakan ekspresi Boolean. Setiap templat menyertakan menentukan opsi berikut:

- NextStep: Langkah selanjutnya dalam runbook untuk memproses jika pilihan yang ditunjuk adalah`true`.
- Variabel: Tentukan nama parameter yang didefinisikan di `Parameters` bagian runbook, variabel yang didefinisikan di `Variables` bagian, atau tentukan objek keluaran dari langkah sebelumnya.

  Tentukan nilai variabel dengan menggunakan formulir berikut.

  `Variable: "{{variable name}}"`

  Tentukan nilai parameter dengan menggunakan formulir berikut.

  `Variable: "{{parameter name}}"`

  Tentukan variabel objek output dengan menggunakan formulir berikut.

  `Variable: "{{previousStepName.outputName}}"`

  > **Note**
  >
  > Membuat variabel output dijelaskan secara lebih detail pada bagian berikutnya, Tentang membuat variabel output.

- Operasi: Kriteria yang digunakan untuk mengevaluasi pilihan, seperti `StringEquals: Linux`. Tindakan `aws:branch` tersebut mendukung operasi berikut:

  Operasi String

  - StringEquals
  - EqualsIgnoreCase
  - StartsWith
  - EndsWith

- Berisi

Operasi numerik

- NumericEquals
- NumericGreater
- NumericLesser
- NumericGreaterOrEquals
- NumericLesser
- NumericLesserOrEquals

Operasi Boolean

- BooleanEquals

> **Important**
>
> Ketika Anda membuat runbook, sistem memvalidasi setiap operasi di runbook. Jika operasi tidak didukung, sistem akan mengembalikan kesalahan saat Anda mencoba membuat runbook.

- Default: Tentukan langkah mundur yang harus dilewati otomatisasi jika tidak ada `Choices` adalah `true`.

> **Note**
>
> Jika Anda tidak ingin menentukan `Default` nilai, maka Anda dapat menentukan `isEnd` pilihan. Jika tidak ada `Choices` adalah `true` dan tidak ada `Default` nilai yang ditentukan, maka otomatisasi berhenti di akhir langkah.

Gunakan template berikut untuk membantu Anda membangun `aws:branch` langkah dalam runbook Anda. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri.

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"
         }
      }
   ]
}
```

Tentang membuat variabel output

Untuk membuat `aws:branch` pilihan yang mereferensikan output dari langkah sebelumnya, Anda perlu mengidentifikasi nama langkah sebelumnya dan namabidang output. Anda kemudian menggabungkan nama-nama langkah dan bidang dengan menggunakan format berikut.

`Variable: "{{previousStepName.outputName}}"`

Sebagai contoh, langkah pertama dalam contoh berikut bernama `GetInstance`. Dan kemudian, di bawah `outputs`, terdapat sebuah bidang yang disebut `platform`. Pada langkah kedua (`ChooseOSforCommands`), penulis ingin mereferensikan output dari bidang platform sebagai variabel. Untuk membuat variabel, cukup gabungkan nama langkah (GetInstance) dan nama bidang output (platform) untuk membuat`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
```

Berikut adalah contoh yang menunjukkan bagaimana *“Variable”: “{{describeInstance.platform}}”* dibuat dari langkah sebelumnya dan output.

```
- 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
```

Contoh **`aws:branch`** runbook

Berikut adalah beberapa contoh runbook yang menggunakan `aws:branch`.

Contoh 1: Menggunakan **`aws:branch`** dengan variabel output untuk menjalankan perintah berdasarkan jenis sistem operasi

Pada langkah pertama contoh ini (`GetInstance`), penulis runbook menggunakan `aws:executeAwsApi` tindakan untuk memanggil `ssm` `DescribeInstanceInformation` Operasi API. Penulis menggunakan tindakan ini untuk menentukan jenis sistem operasi yang digunakan oleh sebuah contoh. Tindakan `aws:executeAwsApi` tersebut mengeluarkan instans ID dan jenis platform.

Pada langkah kedua (`ChooseOSforCommands`), penulis menggunakan `aws:branch` tindakan dengan dua `Choices` (`NextStep: runPowerShellCommand`) dan (`NextStep: runShellCommand`). Otomatisasi mengevaluasi sistem operasi instans dengan menggunakan

output dari langkah sebelumnya (`Variable: "{{GetInstance.platform}}"`). Otomatisasi melompat ke langkah untuk sistem operasi yang ditunjuk.

```
---
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
```

Contoh 2: Menggunakan **`aws:branch`** dengan variabel parameter untuk menjalankan perintah berdasarkan jenis sistem operasi

Penulis runbook mendefinisikan beberapa pilihan parameter pada awal runbook di `parameters` bagian. Satu parameter bernama `OperatingSystemName`. Pada langkah pertama (`ChooseOS`), penulis menggunakan `aws:branch` tindakan dengan dua `Choices` (`NextStep: runWindowsCommand`) dan (`NextStep: runLinuxCommand`). Variabel untuk ini `Choices` mereferensikan opsi parameter yang ditentukan dalam bagian parameter (`Variable: "{{OperatingSystemName}}"`). Ketika pengguna menjalankan runbook ini, mereka menentukan nilai pada saat waktu aktif untuk `OperatingSystemName`. Otomatisasi menggunakan parameter waktu aktif selama `Choices` evaluasi. Otomatisasi melanjutkan ke langkah untuk sistem operasi yang ditunjuk berdasarkan parameter waktu aktif yang ditentukan untuk `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
```

Membuat otomatisasi percabangan yang kompleks dengan operator

Anda dapat membuat otomatisasi percabangan kompleks dengan menggunakan `And`, `Or`, dan `Not` operator di `aws:branch` langkah.

Operator 'Dan'

Gunakan And operator ketika Anda ingin beberapa variabel menjadi `true` untuk sebuah pilihan. Pada contoh berikut, pilihan pertama mengevaluasi apakah instans adalah `running` dan menggunakan`Windows` sistem operasi. Jika evaluasi kedua variabel ini benar, maka otomasi melompat ke `runPowerShellCommand` langkah. Jika satu atau beberapa variabel tersebut `false`, maka otomatisasi mengevaluasi variabel untuk pilihan kedua.

```
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
```

Operator 'Atau'

Gunakan `Or` operator ketika Anda ingin salah satu dari beberapa variabel menjadi benar untuk suatu pilihan. Pada contoh berikut, pilihan pertama mengevaluasi apakah string parameter adalah `Windows` dan apakah output dari AWS Lambda langkah adalah benar. Jika evaluasi menentukan bahwa salah satu dari dua variabel ini benar, maka otomasi melompat ke `RunPowerShellCommand` langkah. Jika kedua variabel tersebut salah, maka otomatisasi mengevaluasi variabel untuk pilihan kedua.

```
- Or:
  - Variable: "{{parameter1}}"
    StringEquals: Windows
  - Variable: "{{BooleanParam1}}"
```

```
    BooleanEquals: true
  NextStep: RunPowershellCommand
- Or:
  - Variable: "{{parameter2}}"
    StringEquals: Linux
  - Variable: "{{BooleanParam2}}"
    BooleanEquals: true
  NextStep: RunShellScript
```

Operator 'Tidak'

Gunakan `Not` operator ketika Anda ingin melompat ke langkah yang didefinisikan ketika variabel tidak benar. Pada contoh berikut, pilihan pertama mengevaluasi apakah string parameter adalah `Not Linux`. Jika evaluasi menentukan bahwa itu adalah Linux, maka otomasi melompat ke `sleep2` langkah. Jika evaluasi pilihan pertama menentukan bahwa itu adalah Linux, maka otomatisasi mengevaluasi pilihan berikutnya.

```
mainSteps:
- name: switch
  action: aws:branch
  inputs:
    Choices:
    - NextStep: sleep2
      Not:
        Variable: "{{testParam}}"
        StringEquals: Linux
    - NextStep: sleep1
      Variable: "{{testParam}}"
      StringEquals: Windows
    Default:
      sleep3
```

Contoh cara menggunakan opsi bersyarat

Bagian ini mencakup contoh yang berbeda tentang cara menggunakan opsi dinamis dalam runbook. Setiap contoh di bagian ini memperluas runbook berikut. Runbook ini memiliki dua tindakan. Tindakan pertama bernama `InstallMsiPackage`. Menggunakan perintah `aws:runCommand` tindakan untuk menginstal aplikasi pada Windows Server instans. Tindakan kedua bernama `TestInstall`. Menggunakan `aws:invokeLambdaFunction` tindakan untuk melakukan tes aplikasi yang diinstal jika aplikasi berhasil diinstal. Langkah satu menentukan `onFailure: Abort`. Ini berarti bahwa jika aplikasi tidak berhasil diinstal, otomatisasi berhenti sebelum langkah kedua.

Contoh 1: Runbook dengan dua tindakan linier

```
---
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
...
```

Membuat otomatisasi dinamis yang melompat ke langkah berbeda dengan menggunakan opsi **`onFailure`**

Contoh berikut menggunakan `onFailure: step:`*`step name`*, `nextStep`, dan `isEnd` pilihan untuk membuat otomatisasi dinamis. Dengan contoh ini, jika `InstallMsiPackage` tindakan gagal, maka otomatisasi melompat ke tindakan yang disebut PostFailure(`onFailure:`

`step:PostFailure`) untuk menjalankan AWS Lambda fungsi untuk melakukan beberapa tindakan jika penginstalan gagal. Jika instalasi berhasil, maka otomatisasi melompat ke TestInstall action ()`nextStep: TestInstall`. Kedua langkah `TestInstall` dan `PostFailure` menggunakan `isEnd` pilihan (`isEnd: true`) sehingga otomatisasi selesai ketika salah satu dari langkah tersebut selesai.

> **Note**
>
> Menggunakan `isEnd` pilihan di langkah terakhir `mainSteps` bagian adalah opsional. Jika langkah terakhir tidak melompat ke langkah lain, maka otomatisasi berhenti setelah menjalankan tindakan di langkah terakhir.

Contoh 2: Otomatisasi dinamis yang melompat ke berbagai langkah

```
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**
>
> Sebelum memproses runbook, sistem memverifikasi bahwa runbook tidak membuat perulangan tak terbatas. Jika perulangan tak terbatas terdeteksi, Otomatisasi mengembalikan kesalahan dan jejak lingkaran yang menunjukkan langkah mana yang membuat perulangan.

Membuat otomatisasi dinamis yang mendefinisikan langkah-langkah penting

Anda dapat menentukan bahwa langkah sangat penting untuk keberhasilan otomatisasi secara keseluruhan. Jika langkah kritis gagal, maka otomatisasi melaporkan status otomatisasi sebagai `Failed`, bahkan jika satu langkah atau lebih berjalan dengan sukses. Dalam contoh berikut, pengguna mengidentifikasi VerifyDependencieslangkah jika InstallMsiPackagelangkah gagal (`onFailure: step:VerifyDependencies`). Pengguna menentukan bahwa `InstallMsiPackage` langkah ini tidak penting (`isCritical: false`). Dalam contoh ini, jika aplikasi gagal untuk menginstal, Otomatisasi memproses `VerifyDependencies` langkah untuk menentukan apakah satu dependensi atau lebih hilang, yang oleh karena itu menyebabkan penginstalan aplikasi gagal.

Contoh 3: Mendefinisikan langkah-langkah penting untuk otomatisasi

```
---
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
...
```

Menggunakan output tindakan sebagai input

Beberapa tindakan otomatisasi mengembalikan output yang telah ditentukan sebelumnya. Anda dapat meneruskan output ini sebagai input ke langkah selanjutnya di runbook Anda menggunakan format. {{*stepName*.*outputName*}} Anda juga dapat menentukan output khusus untuk tindakan otomatisasi di runbook Anda. Ini memungkinkan Anda menjalankan skrip, atau menjalankan operasi API untuk yang lain Layanan AWS sekali sehingga Anda dapat menggunakan kembali nilai sebagai input dalam tindakan selanjutnya. Jenis parameter dalam runbook bersifat statis. Ini berarti jenis parameter tidak dapat diubah setelah ditentukan. Untuk menentukan output langkah menyediakan bidang-bidang berikut:

- Nama: (Wajib) Nama keluaran yang digunakan untuk mereferensikan nilai output di langkah selanjutnya.
- Selector: (Required) Ekspresi JsonPath yang digunakan untuk menentukan nilai output.
- Jenis: (Opsional) Tipe data dari nilai yang dikembalikan oleh bidang pemilih. Nilai tipe yang valid adalah`String`,`Integer`,`Boolean`,`StringList`,`StringMap`,`MapList`. Nilai default-nya adalah `String`.

Jika nilai output tidak cocok dengan tipe data yang Anda tentukan, Automation mencoba mengonversi tipe data. Misalnya, jika nilai yang dikembalikan adalah`Integer`, tetapi yang `Type` ditentukan adalah`String`, nilai output akhir adalah `String` nilai. Jenis konversi berikut didukung:

- `String`nilai dapat dikonversi ke`StringList`, `Integer` dan`Boolean`.
- `Integer`nilai dapat dikonversi ke `String` dan`StringList`.
- `Boolean`nilai dapat dikonversi ke `String` dan`StringList`.
- `StringList`,`IntegerList`, atau `BooleanList` nilai-nilai yang mengandung satu elemen dapat dikonversi ke`String`,`Integer`, atau`Boolean`.

Saat menggunakan parameter atau output dengan tindakan otomatisasi, tipe data tidak dapat diubah secara dinamis dalam input tindakan.

Berikut adalah contoh runbook yang menunjukkan cara mendefinisikan output tindakan, dan referensi nilai sebagai input untuk tindakan selanjutnya. Runbook melakukan hal berikut:

- Menggunakan `aws:executeAwsApi` tindakan untuk memanggil operasi Amazon EC2 DescribeImages API untuk mendapatkan nama Windows Server 2016 tertentu. AMI Ini menampilkan ID gambar sebagai `ImageId`.
- Menggunakan `aws:executeAwsApi` tindakan untuk memanggil operasi Amazon EC2 RunInstances API untuk meluncurkan satu instance yang menggunakan `ImageId` dari langkah sebelumnya. Ini menampilkan instans ID sebagai `InstanceId`.
- Menggunakan `aws:waitForAwsResourceProperty` tindakan untuk melakukan polling operasi Amazon EC2 DescribeInstanceStatus API untuk menunggu instans mencapai status. `running` Tindakan akan berakhir dalam 60 detik. Langkah berakhir jika status instans gagal mencapai `running` setelah 60 detik polling.
- Menggunakan `aws:assertAwsResourceProperty` tindakan untuk memanggil operasi API Amazon EC2 `DescribeInstanceStatus` untuk menegaskan bahwa instans ada dalam `running` status. Langkah gagal jika status instans tidak `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"
...
```

Setiap tindakan otomatisasi yang dijelaskan sebelumnya mengizinkan Anda memanggil operasi API tertentu dengan menentukan namespace layanan, nama operasi API, parameter input, dan parameter output. Input didefinisikan oleh operasi API yang Anda pilih. Anda dapat melihat operasi

API (juga disebut metode) dengan memilih layanan di navigasi kiri pada halaman Referensi Layanan. Pilih metode di bagian Klien untuk layanan yang ingin Anda jalankan. Misalnya, semua operasi API (metode) untuk Amazon Relational Database Service (Amazon RDS) tercantum di halaman berikut: Metode Amazon RDS.

Anda dapat melihat skema untuk setiap tindakan otomatisasi di beberapa lokasi berikut:

- `aws:assertAwsResourceProperty` – Tegaskan AWS status sumber daya atau status peristiwa
- `aws:executeAwsApi`— Panggil dan jalankan operasi AWS API
- `aws:waitForAwsResourceProperty` – Tunggu di AWS properti sumber daya

Skema termasuk deskripsi dari bidang yang diperlukan untuk menggunakan setiap tindakan.

Menggunakan Pemilih/ bidang PropertySelector

Setiap tindakan otomatisasi mengharuskan Anda menentukan output `Selector` (untuk `aws:executeAwsApi`) atau `PropertySelector` (untuk `aws:assertAwsResourceProperty` dan `aws:waitForAwsResourceProperty`). Bidang ini digunakan untuk memproses respons JSON dari operasi AWS API. Bidang ini menggunakan sintaks JSONPath.

Berikut adalah contoh untuk membantu menggambarkan konsep ini untuk `aws:executeAwsAPi` tindakan.

```
---
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"
...
```

Di `aws:executeAwsApi` langkah `getImageId`, otomatisasi menjalankan `DescribeImages` operasi API dan menerima respon dari `ec2`. Otomatisasi kemudian berlaku `Selector - "$.Images[0].ImageId"` ke respon API dan memberikan nilai yang dipilih ke variabel `ImageId` output. Beberapa langkah lain dalam otomatisasi yang sama dapat menggunakan nilai `ImageId` dengan menentukan `"{{ getImageId.ImageId }}"`.

Berikut adalah contoh untuk membantu menggambarkan konsep ini untuk `aws:waitForAwsResourceProperty` tindakan.

```
---
- 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
...
```

Di `aws:waitForAwsResourceProperty` langkah `waitUntilInstanceStateRunning`, otomatisasi menjalankan `DescribeInstanceStatus` operasi API dan menerima respon dari `ec2`. Otomatisasi kemudian berlaku `PropertySelector - "$.InstanceStatuses[0].InstanceState.Name"` untuk respon dan memeriksa apakah nilai yang dikembalikan dan sudah ditentukan cocok dengan nilai dalam `DesiredValues` daftar (dalam hal ini `running`). Langkah tersebut mengulangi proses sampai respon mengembalikan status instans `running`.

Menggunakan JsonPath di runbook

Ekspresi JsonPath adalah string yang dimulai dengan "$." yang digunakan untuk memilih salah satu komponen yang lebih dalam elemen JSON. Daftar berikut mencakup informasi tentang operator JSONPath yang didukung oleh otomatisasi Systems Manager:

- Anak dot-notated (.): Gunakan dengan objek JSON. Operator ini memilih nilai kunci tertentu.

- Deep-scan (..): Gunakan dengan elemen JSON. Operator ini memindai tingkat demi tingkat elemen JSON dan memilih daftar nilai dengan kunci tertentu. Jenis kembalinya operator ini selalu array JSON. Dalam konteks jenis keluaran tindakan otomatisasi, operator dapat berupa StringList atau MapList.
- Indeks Array ([ ]): Gunakan dengan array JSON. Operator ini mendapat nilai indeks tertentu.
- Filter ([? (***expression***)]): Gunakan dengan array JSON. Operator ini memfilter nilai array JSON yang cocok dengan kriteria yang ditentukan dalam ekspresi filter. Ekspresi filter hanya dapat menggunakan operator berikut: ==,! =, >, <, >=, atau <=. Menggabungkan beberapa ekspresi filter dengan AND (&&) atau OR (||) tidak didukung. Jenis kembalinya operator ini selalu array JSON.

Untuk lebih memahami operator JSONPath, tinjau respon JSON berikut dari `DescribeInstances` Operasi API ec2. Respon berikut ini adalah beberapa contoh yang menunjukkan hasil yang berbeda dengan menerapkan ekspresi JSONPath yang berbeda untuk respon dari `DescribeInstances` operasi API.

```
{
    "NextToken": "abcdefg",
    "Reservations": [
        {
            "OwnerId": "123456789012",
            "ReservationId": "r-abcd12345678910",
            "Instances": [
                {
                    "ImageId": "ami-12345678",
                    "BlockDeviceMappings": [
                        {
                            "Ebs": {
                                "DeleteOnTermination": true,
                                "Status": "attached",
                                "VolumeId": "vol-000000000000"
                            },
                            "DeviceName": "/dev/xvda"
                        }
                    ],
                    "State": {
                        "Code": 16,
                        "Name": "running"
                    }
                }
            ],
```

```
            "Groups": []
        },
        {
            "OwnerId": "123456789012",
            "ReservationId": "r-12345678910abcd",
            "Instances": [
                {
                    "ImageId": "ami-12345678",
                    "BlockDeviceMappings": [
                        {
                            "Ebs": {
                                "DeleteOnTermination": true,
                                "Status": "attached",
                                "VolumeId": "vol-111111111111"
                            },
                            "DeviceName": "/dev/xvda"
                        }
                    ],
                    "State": {
                        "Code": 80,
                        "Name": "stopped"
                    }
                }
            ],
            "Groups": []
        }
    ]
}
```

JsonPath Contoh 1: Dapatkan String tertentu dari respons JSON

```
JSONPath:
$.Reservations[0].Instances[0].ImageId

Returns:
"ami-12345678"

Type: String
```

Contoh JSONPath 2: Dapatkan Boolean tertentu dari respons JSON

```
JSONPath:
$.Reservations[0].Instances[0].BlockDeviceMappings[0].Ebs.DeleteOnTermination
```

```
Returns:
true

Type: Boolean
```

JsonPath Contoh 3: Dapatkan Integer tertentu dari respons JSON

```
JSONPath:
$.Reservations[0].Instances[0].State.Code

Returns:
16

Type: Integer
```

Contoh JSONPath 4: Memindai respons JSON secara mendalam, lalu dapatkan semua nilai sebagai VolumeId StringList

```
JSONPath:
$.Reservations..BlockDeviceMappings..VolumeId

Returns:
[
   "vol-000000000000",
   "vol-111111111111"
]

Type: StringList
```

JsonPath Contoh 5: Dapatkan BlockDeviceMappings objek tertentu sebagai StringMap

```
JSONPath:
$.Reservations[0].Instances[0].BlockDeviceMappings[0]

Returns:
{
   "Ebs" : {
      "DeleteOnTermination" : true,
      "Status" : "attached",
      "VolumeId" : "vol-000000000000"
   },
```

```
   "DeviceName" : "/dev/xvda"
}

Type: StringMap
```

JsonPath Contoh 6: Memindai respons JSON secara mendalam, lalu dapatkan semua objek State sebagai MapList

```
JSONPath:
$.Reservations..Instances..State

Returns:
[
   {
      "Code" : 16,
      "Name" : "running"
   },
   {
      "Code" : 80,
      "Name" : "stopped"
   }
]

Type: MapList
```

JsonPath Contoh 7: Filter untuk instance di negara bagian **`running`**

```
JSONPath:
$.Reservations..Instances[?(@.State.Name == 'running')]

Returns:
[
  {
    "ImageId": "ami-12345678",
    "BlockDeviceMappings": [
      {
        "Ebs": {
          "DeleteOnTermination": true,
          "Status": "attached",
          "VolumeId": "vol-000000000000"
        },
        "DeviceName": "/dev/xvda"
      }
```

```
    ],
    "State": {
      "Code": 16,
      "Name": "running"
    }
  }
]

Type: MapList
```

JsonPath Contoh 8: Kembalikan instance yang tidak dalam status **`ImageId` `running`**

```
JSONPath:
$.Reservations..Instances[?(@.State.Name != 'running')].ImageId

Returns:
[
  "ami-12345678"
]

Type: StringList | String
```

Membuat integrasi webhook untuk Otomasi

Untuk mengirim pesan menggunakan webhook selama otomatisasi, buat integrasi. Integrasi dapat dipanggil selama otomatisasi dengan menggunakan `aws:invokeWebhook` tindakan di runbook Anda. Jika Anda belum membuat webhook, lihatMembuat webhook untuk integrasi. Untuk mempelajari lebih lanjut tentang `aws:invokeWebhook` tindakan, lihat`aws:invokeWebhook`-Memanggil integrasi webhook Otomasi.

Seperti yang ditunjukkan dalam prosedur berikut, Anda dapat membuat integrasi dengan menggunakan konsol Otomasi Systems Manager atau alat baris perintah pilihan Anda.

Membuat integrasi (konsol)

Untuk membuat integrasi untuk Otomasi (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Pada panel navigasi, pilih Otomatisasi.
3. Pilih tab Integrasi.

4. Pilih Tambahkan integrasi, dan pilih Webhook.
5. Masukkan nilai yang diperlukan dan nilai opsional apa pun yang ingin Anda sertakan untuk integrasi.
6. Pilih Tambah untuk membuat integrasi.

Membuat integrasi (baris perintah)

Untuk membuat integrasi menggunakan alat baris perintah, Anda harus membuat `SecureString` parameter yang diperlukan untuk integrasi. Automation menggunakan namespace cadangan diParameter Store, kemampuan Systems Manager, untuk menyimpan informasi tentang integrasi Anda. Jika Anda membuat integrasi menggunakanAWS Management Console, Automation menangani proses ini untuk Anda. Mengikuti namespace, Anda harus menentukan jenis integrasi yang ingin Anda buat dan kemudian nama integrasi Anda. Saat ini, Otomasi mendukung integrasi `webhook` tipe.

Bidang yang didukung untuk integrasi `webhook` tipe adalah sebagai berikut:

- Deskripsi
- headers
- payload
- URL

Sebelum Anda memulai

Jika Anda belum menjalankannya, Instal dan konfigurasikan AWS Command Line Interface (AWS CLI) atau AWS Tools for PowerShell. Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI dan Menginstal AWS Tools for PowerShell.

Untuk membuat integrasi untuk Otomasi (baris perintah)

- Jalankan perintah berikut untuk membuat `SecureString` parameter yang diperlukan untuk integrasi. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri. `/d9d01087-4a3f-49e0-b0b4-d568d7826553/ssm/integrations/webhook/`Namespace dicadangkan Parameter Store untuk integrasi. Nama parameter Anda harus menggunakan namespace ini diikuti dengan nama integrasi Anda. Misalnya, `/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"}'
```

Membuat webhook untuk integrasi

Saat membuat webhook dengan penyedia Anda, perhatikan hal berikut:

- Protokol harus HTTPS.
- Header permintaan khusus didukung.
- Sebuah badan permintaan default dapat ditentukan.
- Badan permintaan default dapat diganti ketika integrasi dipanggil dengan menggunakan tindakan. `aws:invokeWebhook`

Menangani waktu habis di runbook

Properti `timeoutSeconds` dibagi oleh semua tindakan otomatisasi. Anda dapat menggunakan properti ini untuk menentukan nilai batas waktu eksekusi untuk suatu tindakan. Selanjutnya, Anda dapat mengubah bagaimana waktu tunggu tindakan memengaruhi otomatisasi dan status eksekusi secara keseluruhan. Anda dapat melakukan ini dengan juga mendefinisikan `onFailure` dan `isCritical` properti bersama untuk suatu tindakan.

Misalnya, bergantung pada kasus penggunaan Anda, Anda mungkin ingin otomatisasi Anda melanjutkan ke tindakan yang berbeda dan tidak memengaruhi status otomatisasi secara keseluruhan jika waktu tindakan habis. Dalam contoh ini, Anda menentukan lama waktu untuk menunggu sebelum waktu tindakan habis menggunakan `timeoutSeconds` properti. Kemudian Anda menentukan tindakan, atau langkah yang harus dilakukan otomatisasi jika waktu habis. Tentukan nilai menggunakan format `step:`*`step name`* untuk `onFailure` properti daripada nilai default dari `Abort`. Secara default, jika waktu tindakan habis, status eksekusi otomatisasi akan menjadi `Timed Out`. Agar waktu habis tidak mempengaruhi status eksekusi otomatisasi, tentukan `false` untuk `isCritical` properti.

Contoh berikut menunjukkan cara menentukan properti bersama untuk tindakan yang dijelaskan dalam skenario ini.

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"
}
```

Untuk informasi lebih lanjut tentang properti bersama oleh semua tindakan otomatisasi, lihat Properti dibagi oleh semua tindakan.

Referensi runbook Otomatisasi Systems Manager

Untuk membantu Anda memulai dengan cepat, AWS Systems Manager menyediakan runbook yang telah ditetapkan. Runbook ini dikelola oleh Amazon Web Services, AWS Support, dan AWS Config. Referensi runbook menjelaskan masing-masing runbook standar yang disediakan oleh Systems Manager, AWS Support, dan AWS Config. Untuk informasi lebih lanjut, lihat Referensi runbook Otomatisasi Systems Manager.

Tutorial

Tutorial berikut membantu Anda menggunakanAWS Systems Manager Otomasi untuk mengatasi kasus penggunaan umum. Tutorial ini menunjukkan bagaimana menggunakan runbook Anda sendiri, runbook yang telah ditentukan yang disediakan oleh Automation, dan kemampuan Systems Manager lainnya dengan lainnyaLayanan AWS.

Daftar Isi

Memperbarui AMIs

Tutorial berikut menjelaskan cara memperbarui Amazon Machine Image (AMIs) untuk menyertakan tambalan terbaru.

Topik

Perbarui Linux AMI

Panduan Otomasi Systems Manager ini menunjukkan cara menggunakan konsol atau AWS CLI `AWS-UpdateLinuxAmi` runbook untuk memperbarui AMI Linux dengan tambalan paket terbaru yang Anda tentukan. Otomatisasi adalah kemampuan AWS Systems Manager. `AWS-UpdateLinuxAmi` runbook juga mengotomatiskan instalasi paket dan konfigurasi khusus situs tambahan. Anda dapat memperbarui berbagai distribusi Linux menggunakan panduan ini, termasuk, Ubuntu Server CentOS, RHEL, SLES, atau Amazon Linux. AMIs Untuk daftar lengkap versi Linux yang didukung, lihat Prasyarat Patch Manager.

`AWS-UpdateLinuxAmi`Runbook memungkinkan Anda untuk mengotomatiskan tugas pemeliharaan gambar tanpa harus membuat runbook di JSON atau YAMAL. Anda dapat menggunakan `AWS-UpdateLinuxAmi` runbook untuk melakukan jenis tugas berikut.

- Tingkatkan semua paket distribusi dan perangkat lunak Amazon di Amazon Linux,,,Red Hat Enterprise Linux, Ubuntu ServerSUSE Linux Enterprise Server, atau CentOS Amazon Machine Image ()AMI. Ini adalah perilaku runbook default.
- Instal AWS Systems Manager SSM Agent pada gambar yang ada untuk mengaktifkan kemampuan Systems Manager, seperti menjalankan perintah jarak jauh menggunakan AWS Systems Manager Run Command atau pengumpulan inventaris perangkat lunak menggunakan Inventaris.
- Instal paket perangkat lunak tambahan.

Sebelum Anda memulai

Sebelum Anda mulai bekerja dengan runbook, konfigurasikan peran dan, secara opsional, EventBridge untuk Otomasi. Untuk informasi selengkapnya, lihat Menyiapkan Otomatisasi. Panduan ini juga mengharuskan Anda menentukan nama profil instance AWS Identity and Access Management (IAM). Untuk informasi selengkapnya tentang membuat profil instans IAM, lihat Mengonfigurasi izin instans untuk Systems Manager.

`AWS-UpdateLinuxAmi` runbook menerima parameter masukan berikut.

Parameter	Jenis	Deskripsi
SourceAmiId	String	(Diperlukan) Sumber AMI ID.
IamInstanceProfileName	String	(Wajib) Nama peran profil instans IAM yang Anda buat di Konfigurasikan izin instans untuk Systems Manager. Peran profil instans memberikan izin otomatisasi untuk melakukan tindakan pada instans Anda, seperti menjalankan perintah atau memulai dan menghentikan layanan. Runbook hanya menggunakan nama peran profil instans. Jika Anda menentukan Amazon Resource Name (ARN), otomatisasi akan gagal.
AutomationAssumeRole	String	(Wajib) Nama peran layanan IAM yang Anda buat di Menyiapkan Otomatisasi. Peran layanan (juga disebut peran asumsi) memberikan izin Otomatisasi untuk menganggap IAM role Anda dan melakukan tindakan atas nama Anda. Sebagai contoh,

Parameter	Jenis	Deskripsi
		peran layanan mengizinkan Otomatisasi untuk membuat baru AMI ketika menjalank an `aws:createImage` tindakan dalam runbook. Untuk parameter ini, ARN lengkap mesti ditentukan.
TargetAmiName	String	(Opsional) Nama baru AMI setelah dibuat. Nama default adalah string yang dihasilka n sistem yang mencakup sumber AMI ID, serta waktu pembuatan dan tanggal.
InstanceType	String	(Opsional) Jenis instans yang akan diluncurkan sebagai host ruang kerja. Jenis instans bervariasi menurut wilayah. Jenis default adalah t2.micro.
PreUpdateScript	String	(Opsional) URL skrip untuk berjalan sebelum pembaruan diterapkan. Default (\"none\") adalah untuk tidak menjalank an skrip.
PostUpdateScript	String	(Opsional) URL skrip yang dijalankan setelah menerapka n pembaruan paket. Default (\"none\") adalah untuk tidak menjalankan skrip.

Parameter	Jenis	Deskripsi
IncludePackages	String	(Opsional) Hanya perbarui paket dengan nama ini. Secara default (\"all\"), semua pembaruan yang tersedia diterapkan.
ExcludePackages	String	(Opsional) Nama paket untuk menahan pembaruan, dalam semua kondisi. Secara default (\"none\"), tidak ada paket yang dikecualikan.

Langkah Otomatisasi

`AWS-UpdateLinuxAmi` runbook mencakup tindakan otomatisasi berikut, secara default.

Langkah 1: launchInstance (**`aws:runInstances`** tindakan)

Langkah ini meluncurkan instans menggunakan data pengguna Amazon Elastic Compute Cloud (Amazon EC2) dan peran profil instans IAM. Userdata menginstal yang sesuaiSSM Agent, berdasarkan sistem operasi. Instalasi SSM Agent memungkinkan Anda untuk memanfaatkan kemampuan Systems Manager sepertiRun Command,State Manager, dan Inventaris.

Langkah 2: UpdateosSoftware (**`aws:runCommand`** tindakan)

Langkah ini menjalankan perintah berikut pada instans yang diluncurkan:

- Unduh skrip pembaruan dari Amazon S3.
- Menjalankan skrip pra-pembaruan opsional.
- Memperbarui paket distribusi dan perangkat lunak Amazon.
- Menjalankan skrip pasca-pembaruan opsional.

Log eksekusi disimpan dalam folder/tmp agar pengguna dapat melihat nanti.

Jika Anda ingin meningkatkan paket tertentu, Anda dapat menyediakan daftar menggunakan `IncludePackages` parameter. Ketika disediakan, sistem mencoba untuk memperbarui paket ini saja dan dependensinya. Tidak ada pembaruan lain yang dilakukan. Secara default, bila tidak ada paket penyertaan yang ditentukan, program memperbarui semua paket yang tersedia.

Jika Anda ingin mengecualikan peningkatan paket tertentu, Anda dapat menyediakan daftar menggunakan `ExcludePackages` parameter. Jika tersedia, paket ini tetap pada versi mereka saat ini, terlepas dari pilihan lain yang ditentukan. Secara default, bila tidak ada paket penyertaan yang ditentukan, tidak ada paket yang dikecualikan.

Langkah 3: stopInstance (**`aws:changeInstanceState`** tindakan)

Langkah ini menghentikan instans yang diperbarui.

Langkah 4: CreateImage (**`aws:createImage`** tindakan)

Langkah ini menciptakan AMI baru dengan nama deskriptif yang menghubungkannya ke ID sumber dan waktu pembuatan. Misalnya: “AMIDihasilkan oleh EC2 Automation on {{global:Date_time}} from {{}SourceAmiId}” di mana DATE_TIME dan SourceID mewakili variabel Otomasi.

Langkah 5: terminateInstance (**`aws:changeInstanceState`** tindakan)

Langkah ini membersihkan otomatisasi dengan mengakhiri instans berjalan.

Output

Otomatisasi mengembalikan AMI ID baru sebagai output.

> **Note**
>
> Secara default, ketika Otomatisasi menjalankan `AWS-UpdateLinuxAmi` runbook, sistem menciptakan instans sementara dalam VPC default (172.30.0.0/16). Jika Anda menghapus VPC default, Anda akan menerima kesalahan berikut:
> `VPC not defined 400`
> Untuk mengatasi masalah ini, Anda harus membuat salinan `AWS-UpdateLinuxAmi` runbook dan menentukan ID subnet. Untuk informasi selengkapnya, lihat VPC tidak didefinisikan 400.

Untuk membuat patch AMI menggunakan Otomatisasi (AWS Systems Manager)

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Pada panel navigasi, pilih Otomatisasi.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Otomasi.

3. Pilih Eksekusi otomatisasi.
4. Di daftar Dokumen Otomatisasi, pilih **`AWS-UpdateLinuxAmi`**.
5. Di bagian Detail dokumen, verifikasi bahwa Versi dokumen diatur ke Versi default saat waktu aktif.
6. Pilih Berikutnya.
7. Dalam bagian Mode Eksekusi, pilih Eksekusi sederhana.
8. Di bagian Parameter input, masukkan informasi yang Anda kumpulkan di bagian Sebelum Anda memulai.
9. Pilih Eksekusi. Konsol menampilkan status eksekusi Otomatisasi.

Setelah otomatisasi selesai, luncurkan instans uji dari yang diperbarui AMI untuk memverifikasi perubahan.

> **Note**
>
> Jika setiap langkah dalam otomatisasi gagal, informasi tentang kegagalan tercantum pada halaman Eksekusi Otomatisasi. Otomatisasi dirancang untuk mengakhiri instans sementara setelah berhasil menyelesaikan semua tugas. Jika langkah gagal, sistem mungkin tidak mengakhiri instans. Jadi jika langkah gagal, akhiri instans sementara secara manual.

Perbarui Linux AMI (AWS CLI)

Panduan AWS Systems Manager Otomasi ini menunjukkan cara menggunakan `AWS-UpdateLinuxAmi` runbook AWS Command Line Interface (AWS CLI) dan Systems Manager untuk secara otomatis menambal Linux Amazon Machine Image (AMI) dengan versi paket terbaru yang Anda tentukan. Otomasi adalah kemampuan AWS Systems Manager. `AWS-UpdateLinuxAmi` runbook juga mengotomatiskan instalasi paket dan konfigurasi khusus situs tambahan. Anda dapat memperbarui berbagai distribusi Linux menggunakan panduan ini, termasuk, Ubuntu Server CentOS, RHEL, SLES, atau Amazon Linux. AMIs Untuk daftar lengkap versi Linux yang didukung, lihat Prasyarat Patch Manager.

`AWS-UpdateLinuxAmi`Runbook memungkinkan Anda untuk mengotomatiskan tugas pemeliharaan gambar tanpa harus membuat runbook di JSON atau YAMM. Anda dapat menggunakan `AWS-UpdateLinuxAmi` runbook untuk melakukan jenis tugas berikut.

- Tingkatkan semua paket distribusi dan perangkat lunak Amazon di Amazon Linux,, Red Hat Enterprise LinuxUbuntu Server, SLES, atau Cent OS Amazon Machine Image (AMI). Ini adalah perilaku runbook default.
- Instal AWS Systems Manager SSM Agent pada gambar yang ada untuk mengaktifkan kemampuan Systems Manager, seperti menjalankan perintah jarak jauh menggunakan AWS Systems Manager Run Command atau pengumpulan inventaris perangkat lunak menggunakan Inventaris.
- Instal paket perangkat lunak tambahan.

Sebelum Anda memulai

Sebelum Anda mulai bekerja dengan runbook, konfigurasikan peran dan, secara opsional, EventBridge untuk Otomasi. Untuk informasi selengkapnya, lihat Menyiapkan Otomatisasi. Panduan ini juga mengharuskan Anda menentukan nama profil instance AWS Identity and Access Management (IAM). Untuk informasi selengkapnya tentang membuat profil instans IAM, lihat Mengonfigurasi izin instans untuk Systems Manager.

`AWS-UpdateLinuxAmi` runbook menerima parameter masukan berikut.

Parameter	Jenis	Deskripsi
SourceAmiId	String	(Diperlukan) Sumber AMI ID. Anda dapat secara otomatis mereferensikan ID terbaru Amazon EC2 AMI untuk Linux dengan menggunak an parameter AWS Systems Manager Parameter Store publik. Untuk informasi selengkapnya, lihat Kueri untuk AMI ID Amazon Linux terbaru yang digunakan AWS Systems ManagerParameter Store.

Parameter	Jenis	Deskripsi
IamInstanceProfileName	String	(Wajib) Nama peran profil instans IAM yang Anda buat di Konfigurasikan izin instans untuk Systems Manager. Peran profil instans memberikan izin otomatisasi untuk melakukan tindakan pada instans Anda, seperti menjalankan perintah atau memulai dan menghentikan layanan. Runbook hanya menggunakan nama peran profil instans.
AutomationAssumeRole	String	(Wajib) Nama peran layanan IAM yang Anda buat di Menyiapkan Otomatisasi. Peran layanan (juga disebut peran asumsi) memberikan izin Otomatisasi untuk menganggap IAM role Anda dan melakukan tindakan atas nama Anda. Sebagai contoh, peran layanan mengizinkan Otomatisasi untuk membuat baru AMI ketika menjalankan `aws:createImage` tindakan dalam runbook. Untuk parameter ini, ARN lengkap mesti ditentukan.

Parameter	Jenis	Deskripsi
TargetAmiName	String	(Opsional) Nama baru AMI setelah dibuat. Nama default adalah string yang dihasilkan sistem yang mencakup sumber AMI ID, serta waktu pembuatan dan tanggal.
InstanceType	String	(Opsional) Jenis instans yang akan diluncurkan sebagai host ruang kerja. Jenis instans bervariaso menurut Wilayah. Jenis default adalah t2.micro.
PreUpdateScript	String	(Opsional) URL skrip untuk berjalan sebelum pembaruan diterapkan. Default (\"none\") adalah untuk tidak menjalankan skrip.
PostUpdateScript	String	(Opsional) URL skrip yang dijalankan setelah menerapkan pembaruan paket. Default (\"none\") adalah untuk tidak menjalankan skrip.
IncludePackages	String	(Opsional) Hanya perbarui paket dengan nama ini. Secara default (\"all\"), semua pembaruan yang tersedia diterapkan.
ExcludePackages	String	(Opsional) Nama paket untuk menahan pembaruan, dalam semua kondisi. Secara default (\"none\"), tidak ada paket yang dikecualikan.

Langkah Otomatisasi

`AWS-UpdateLinuxAmi` runbook mencakup beberapa langkah berikut, secara default.

Langkah 1: launchInstance (**`aws:runInstances`** tindakan)

Langkah ini meluncurkan instance menggunakan data pengguna Amazon Elastic Compute Cloud (Amazon EC2) dan peran profil instans IAM. Data pengguna menginstal Agen SSM yang sesuai, berdasarkan sistem operasi. Instalasi SSM Agent memungkinkan Anda untuk memanfaatkan kemampuan Systems Manager sepertiRun Command,State Manager, dan Inventaris.

Langkah 2: UpdateosSoftware (**`aws:runCommand`** tindakan)

Langkah ini menjalankan perintah berikut pada instans yang diluncurkan:

- Unduh skrip pembaruan dari Amazon Simple Storage Service (Amazon S3).
- Menjalankan skrip pra-pembaruan opsional.
- Memperbarui paket distribusi dan perangkat lunak Amazon.
- Menjalankan skrip pasca-pembaruan opsional.

Log eksekusi disimpan dalam folder/tmp agar pengguna dapat melihat nanti.

Jika Anda ingin meningkatkan paket tertentu, Anda dapat menyediakan daftar menggunakan `IncludePackages` parameter. Ketika disediakan, sistem mencoba untuk memperbarui paket ini saja dan dependensinya. Tidak ada pembaruan lain yang dilakukan. Secara default, bila tidak ada paket penyertaan yang ditentukan, program memperbarui semua paket yang tersedia.

Jika Anda ingin mengecualikan peningkatan paket tertentu, Anda dapat menyediakan daftar menggunakan `ExcludePackages` parameter. Jika tersedia, paket ini tetap pada versi mereka saat ini, terlepas dari pilihan lain yang ditentukan. Secara default, bila tidak ada paket penyertaan yang ditentukan, tidak ada paket yang dikecualikan.

Langkah 3: stopInstance (**`aws:changeInstanceState`** tindakan)

Langkah ini menghentikan instans yang diperbarui.

Langkah 4: CreateImage (**`aws:createImage`** tindakan)

Langkah ini menciptakan AMI baru dengan nama deskriptif yang menghubungkannya ke ID sumber dan waktu pembuatan. Misalnya: “AMI Dihasilkan oleh EC2 Automation on {{global:Date_time}} from {{}SourceAmiId}” dimana DATE_TIME dan SourceID mewakili variabel Automation.

Langkah 5: terminateInstance (**`aws:changeInstanceState`** tindakan)

Langkah ini membersihkan otomatisasi dengan mengakhiri instans berjalan.

Output

Otomatisasi mengembalikan AMI ID baru sebagai output.

> **Note**
>
> Secara default, ketika Otomatisasi menjalankan `AWS-UpdateLinuxAmi` runbook, sistem menciptakan instans sementara dalam VPC default (172.30.0.0/16). Jika Anda menghapus VPC default, Anda akan menerima kesalahan berikut:
> `VPC not defined 400`
> Untuk mengatasi masalah ini, Anda harus membuat salinan `AWS-UpdateLinuxAmi` runbook dan menentukan ID subnet. Untuk informasi selengkapnya, lihat VPC tidak didefinisikan 400.

Untuk membuat patch AMI menggunakan Otomatisasi

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI.
2. Jalankan perintah berikut untuk menjalankan `AWS-UpdateLinuxAmi` runbook. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri.

   ```
   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'
   ```

   Perintah mengembalikan ID eksekusi. Salin ID ini ke clipboard. Anda akan menggunakan ID ini untuk melihat status otomatisasi.

   ```
   {
       "AutomationExecutionId": "automation execution ID"
   ```

```
}
```

3. Untuk melihat otomatisasi menggunakan AWS CLI, jalankan perintah berikut:

```
aws ssm describe-automation-executions
```

4. Untuk melihat detail tentang kemajuan otomatisasi, jalankan perintah berikut. Ganti *ID eksekusi otomatisasi* dengan informasi Anda sendiri.

```
aws ssm get-automation-execution --automation-execution-id automation execution ID
```

Proses pembaruan dapat memakan waktu 30 menit atau lebih untuk diselesaikan.

> **Note**
>
> Anda juga dapat memantau status otomatisasi di konsol. Dalam daftar, pilih otomatisasi yang baru saja Anda jalankan dan pilih tab Langkah. Tab ini memperlihatkan kepada Anda status tindakan otomatisasi.

Setelah otomatisasi selesai, luncurkan instans uji dari yang diperbarui AMI untuk memverifikasi perubahan.

> **Note**
>
> Jika setiap langkah dalam otomatisasi gagal, informasi tentang kegagalan tercantum pada halaman Eksekusi Otomatisasi. Otomatisasi dirancang untuk mengakhiri instans sementara setelah berhasil menyelesaikan semua tugas. Jika langkah gagal, sistem mungkin tidak mengakhiri instans. Jadi jika langkah gagal, akhiri instans sementara secara manual.

MemperbaruiWindows ServerAMI

`AWS-UpdateWindowsAmi`runbook mengizinkan Anda untuk mengotomatiskan beberapa tugas pemeliharaan gambar di Amazon Windows Anda Amazon WindowsAmazon Machine Image (AMI) tanpa harus menulis runbook di JSON atau YAKL. Runbook ini didukung untuk Windows Server 2008 R2 atau yang lebih baru. Anda dapat menggunakan `AWS-UpdateWindowsAmi` runbook untuk melakukan jenis tugas berikut.

- Instal semua pembaruan Windows dan tingkatkan perangkat lunak Amazon (perilaku default).
- Instal pembaruan Windows tertentu dan tingkatkan perangkat lunak Amazon.
- Sesuaikan AMI menggunakan skrip Anda.

Sebelum Anda memulai

Sebelum Anda mulai bekerja dengan runbook, konfigurasikan peran untuk Otomatisasi guna menambahkan `iam:PassRole` kebijakan yang mereferensikan ARN profil instans yang ingin Anda berikan akses. Opsional, konfigurasikan Amazon EventBridge untuk otomatisasi, kemampuanAWS Systems Manager. Untuk informasi selengkapnya, lihat Menyiapkan Otomatisasi. Panduan ini juga mengharuskan Anda menentukan nama AWS Identity and Access Management profil instans (IAM). Untuk informasi lebih lanjut tentang cara membuat profil instans IAM, lihat Mengonfigurasi izin instans untuk Systems Manager.

> **Note**
>
> Perbaruan untukAWS Systems ManagerSSM Agent biasanya diluncurkan ke wilayah yang berbeda pada waktu yang berbeda. Ketika Anda menyesuaikan atau memperbarui AMI, hanya gunakan sumber AMIs yang dipublikasikan untuk wilayah tempat Anda bekerja. Hal ini akan memastikan bahwa Anda bekerja denganSSM Agent rilis terbaru untuk wilayah tersebut dan menghindari masalah kompatibilitas.

`AWS-UpdateWindowsAmi` runbook menerima parameter masukan berikut.

Parameter	Jenis	Deskripsi
SourceAmiId	String	(Diperlukan) Sumber AMI ID. Anda dapat secara otomatis mereferensikanAMI ID Windows Server terbaru dengan menggunakan parameterParameter Store publik Systems Manager. Untuk informasi selengkap nya, lihat Kueri untukAMI ID Windows terbaru yang

Parameter	Jenis	Deskripsi
		digunakanAWS Systems ManagerParameter Store.
SubnetId	String	(Opsional) Subnet yang ingin Anda luncurkan instance sementara. Anda harus menentukan nilai untuk parameter ini jika Anda menghapus VPC default Anda.
IamInstanceProfileName	String	(Diperlukan) Nama peran profil instans IAM yang Anda buat di Mengonfigurasi izin instans untuk Systems Manager. Peran profil instans memberikan izin otomatisasi untuk melakukan tindakan pada instans Anda, seperti menjalankan perintah atau memulai dan menghentikan layanan. Runbook hanya menggunakan nama peran profil instans.

Parameter	Jenis	Deskripsi
AutomationAssumeRole	String	(Wajib) Nama peran layanan IAM yang Anda buat di Menyiapkan Otomatisasi. Peran layanan (juga disebut peran asumsi) memberika n izin Otomatisasi untuk menganggap IAM role Anda dan melakukan tindakan atas nama Anda. Sebagai contoh, peran layanan mengizinkan Otomatisasi untuk membuat baru AMI ketika menjalank an `aws:createImage` tindakan dalam runbook. Untuk parameter ini, ARN lengkap mesti ditentukan.
TargetAmiName	String	(Opsional) Nama baruAMI setelah diciptakan. Nama default adalah string yang dihasilkan sistem yang mencakup sumber AMI ID, serta waktu pembuatan dan tanggal.
InstanceType	String	(Opsional) Jenis instans yang akan diluncurkan sebagai host ruang kerja. Jenis instans bervariasi menurut wilayah. Jenis default adalah t2.medium.

Parameter	Jenis	Deskripsi
PreUpdateScript	String	(Opsional) Skrip yang dijalankan sebelum memperbarui AMI. Masukkan skrip dalam runbook atau pada saat waktu aktif sebagai parameter.
PostUpdateScript	String	(Opsional) Skrip yang akan dijalankan setelah memperbarui AMI. Masukkan skrip dalam runbook atau pada saat waktu aktif sebagai parameter.
IncludeKbs	String	(Opsional) Tentukan satu ID artikel Pangkalan Pengetahuan Microsoft (KB) atau lebih untuk disertakan. Anda dapat menginstal beberapa ID menggunakan nilai dipisahkan koma. Format yang valid: KB9876543 atau 9876543.
ExcludeKbs	String	(Opsional) Tentukan satu ID artikel Pangkalan Pengetahuan Microsoft (KB) atau lebih untuk dikecualikan. Anda dapat mengecualikan beberapa ID menggunakan nilai yang dipisahkan koma. Format yang valid: KB9876543 atau 9876543.

Parameter	Jenis	Deskripsi
Kategori	String	(Opsional)Tentukan satu kategori pembaruan atau lebih. Anda dapat memfilter kategori menggunakan nilai yang dipisahkan koma. Opsi: Pembaruan Kritis, Pembaruan Keamanan, Pembaruan Definisi, Batal Pembaruan, Paket Layanan, Alat, Pembaruan, atau Driver. Format yang valid mencakup satu entri, misalnya: Pembaruan Kritis. Atau, Anda dapat menentukan daftar yang dipisahkan koma: Pembaruan Kritis,Pembaruan Keamanan,Pembaruan Definisi.
SeverityLevels	String	(Opsional) Tentukan satu tingkat keparahan MSRC atau lebih yang terkait dengan pembaruan. Anda dapat memfilter tingkat keparahan menggunak an nilai yang dipisahkan koma. Pilihan: Kritis, Penting, Rendah, Sedang atau Tidak Ditentukan. Format yang valid mencakup satu entri, misalnya: Kritis. Atau, Anda dapat menentukan daftar yang dipisahkan koma: Kritis,Pe nting,Rendah.

Langkah Otomatisasi

`AWS-UpdateWindowsAmi` runbook mencakup beberapa langkah berikut, secara default.

Langkah 1: launchInstance (**`aws:runInstances`** tindakan)

Langkah ini meluncurkan sebuah instans dengan peran profil instans IAM dari yang ditentukan `SourceAmiID`.

Langkah 2: runPreUpdate Script (**`aws:runCommand`**action)

Langkah ini mengizinkan Anda menentukan skrip sebagai string yang berjalan sebelum pembaruan diinstal.

Langkah 3: UpdateEC2Config (**`aws:runCommand`** tindakan)

Langkah ini menggunakan`AWS-InstallPowerShellModule` runbook untuk mengunduh PowerShell modulAWS publik. Systems Manager memverifikasi integritas modul dengan menggunakan hash SHA-256. Systems Manager kemudian memeriksa sistem operasi untuk menentukan apakah akan memperbarui EC2Config atau EC2Launch. EC2Config berjalan di Windows Server 2008 R2 melalui Windows Server 2012 R2. EC2Launch berjalan pada Windows Server 2016.

Langkah 4: updateSSMAgent (**`aws:runCommand`** tindakan)

Langkah ini diperbaruiSSM Agent dengan menggunakan`AWS-UpdateSSMAgent` runbook.

Langkah 5: updateAWSPVDriver (**`aws:runCommand`**action)

Langkah ini memperbarui AWS driver PV dengan menggunakan `AWS-ConfigureAWSPackage` runbook.

Langkah 6: updateAwsEnaNetworkDriver (**`aws:runCommand`**tindakan)

Langkah ini memperbarui AWS driver Jaringan ENA dengan menggunakan `AWS-ConfigureAWSPackage` runbook.

Langkah 7: installWindowsUpdates (**`aws:runCommand`**tindakan)

Langkah ini menginstal pembaruan Windows dengan menggunakan `AWS-InstallWindowsUpdates` runbook. Secara default, Systems Manager mencari dan menginstal semua pembaruan yang hilang. Anda dapat mengubah perilaku default dengan menentukan salah satu parameter berikut: `IncludeKbs`, `ExcludeKbs`, `Categories`, atau `SeverityLevels`.

Langkah 8: runPostUpdate Script (**`aws:runCommand`**action)

Langkah ini mengizinkan Anda menentukan skrip sebagai string yang berjalan sebelum menginstal pembaruan.

Langkah 9: runSysprepGeneralize (**`aws:runCommand`**tindakan)

Langkah ini menggunakan`AWS-InstallPowerShellModule` runbook untuk mengunduh PowerShell modulAWS publik. Systems Manager memverifikasi integritas modul dengan menggunakan hash SHA-256. Systems Manager kemudian menjalankan sysprep menggunakan AWS-didukung metode EC2Launch (Windows Server 2016) atau EC2Config (Windows Server 2008 R2 melalui 2012 R2).

Langkah 10: stopInstance (**`aws:changeInstanceState`** tindakan)

Langkah ini menghentikan instans yang diperbarui.

Langkah 11: CreateImage (**`aws:createImage`** tindakan)

Langkah ini menciptakan AMI baru dengan nama deskriptif yang menghubungkannya ke ID sumber dan waktu pembuatan. Misalnya: “AMI yang Dihasilkan oleh Otomatisasi EC2 pada {{global:Date_time}} dari {{SourceAmiId}}” di mana DATE_TIME dan SourceID mewakili variabel Otomasi.

Langkah 12: TerminateInstance (**`aws:changeInstanceState`**tindakan)

Langkah ini membersihkan otomatisasi dengan mengakhiri instans berjalan.

Output

Bagian ini memungkinkan Anda untuk menunjuk output dari berbagai langkah atau nilai-nilai dari setiap parameter sebagai output Otomatisasi. Secara default, output adalah ID dari Windows yang diperbarui AMI yang diciptakan oleh otomatisasi.

> **Note**
>
> Secara default, ketika Otomatisasi menjalankan `AWS-UpdateWindowsAmi` runbook dan membuat instans sementara, sistem menggunakan VPC default (172.30.0.0/16). Jika Anda menghapus VPC default, Anda akan menerima kesalahan berikut:
> VPC tidak didefinisikan 400

Untuk mengatasi masalah ini, Anda harus membuat salinan `AWS-UpdateWindowsAmi` runbook dan menentukan ID subnet. Untuk informasi selengkapnya, lihat VPC tidak didefinisikan 400.

Untuk membuat Windows yang di-patch AMI dengan menggunakan Otomatisasi

1. Instal dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaruAWS CLI.

2. Jalankan perintah berikut untuk menjalankan `AWS-UpdateWindowsAmi` runbook. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri. Contoh perintah di bawah menggunakan Amazon EC2 terbaruAMI untuk meminimalkan bilangan patch yang perlu digunakan. Jika Anda menjalankan perintah ini lebih dari sekali, Anda harus menentukan nilai unik untuk `targetAMIname`. AMI nama harus unik.

   ```
   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'
   ```

   Perintah mengembalikan ID eksekusi. Salin ID ini ke clipboard. Anda akan menggunakan ID ini untuk melihat status otomatisasi.

   ```
   {
       "AutomationExecutionId": "automation execution ID"
   }
   ```

3. Untuk melihat otomatisasi menggunakan AWS CLI, jalankan perintah berikut:

   ```
   aws ssm describe-automation-executions
   ```

4. Untuk melihat detail tentang kemajuan otomatisasi, jalankan perintah berikut.

   ```
   aws ssm get-automation-execution
       --automation-execution-id automation execution ID
   ```

> **Note**
>
> Tergantung pada jumlah patch yang diterapkan, untuk menyelesaikan proses patch Windows berjalan dalam otomatisasi sampel ini membutuhkan waktu 30 menit atau lebih.

Perbarui emas AMI menggunakan Otomasi,AWS Lambda, dan Parameter Store

Contoh berikut menggunakan model di mana organisasi memelihara dan secara berkala menambal milik mereka sendiri AMIs daripada membangun dari Amazon Elastic Compute Cloud (Amazon EC2). AMIs

Prosedur berikut menunjukkan cara menerapkan patch sistem operasi (OS) secara otomatis ke patch AMI yang sudah dianggap paling up-to-date atau terbaruAMI. Dalam contoh, nilai default parameter `SourceAmiId` ditentukan oleh AWS Systems Manager Parameter Store parameter yang disebut`latestAmi`. Nilai dari `latestAmi` diperbarui oleh AWS Lambda fungsi yang dijalankan pada akhir otomatisasi. Sebagai hasil dari proses Otomasi ini, waktu dan upaya yang dihabiskan untuk menambal AMIs diminimalkan karena penambalan selalu diterapkan secara maksimal. up-to-date AMI Parameter Storedan otomatisasi adalah kemampuanAWS Systems Manager.

Sebelum Anda memulai

Konfigurasikan peran Otomasi dan, secara opsional, Amazon EventBridge for Automation. Untuk informasi selengkapnya, lihat Menyiapkan Otomatisasi.

Daftar Isi

Tugas 1: Buat parameter di Systems Manager Parameter Store

Buat parameter string Parameter Store yang menggunakan informasi berikut:

- Nama: `latestAmi`.
- Nilai: Sebuah AMI ID. Sebagai contoh: `ami-188d6e0e`.

Untuk informasi tentang cara membuat parameter Parameter Store string, lihatMenandai parameter Systems Manager.

Tugas 2: Buat IAM role untuk AWS Lambda

Gunakan prosedur berikut untuk membuat peran layanan IAM untuk AWS Lambda. Kebijakan ini memberikan izin Lambda untuk memperbarui nilai `latestAmi` parameter menggunakan fungsi Lambda dan Systems Manager.

Untuk membuat peran layanan IAM untuk Lambda

1. Masuk ke AWS Management Console dan buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Pada panel navigasi, pilih Kebijakan, lalu pilih Buat kebijakan.
3. Pilih tab JSON.
4. Ganti konten default dengan kebijakan berikut. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "logs:CreateLogGroup",
            "Resource": "arn:aws:logs:region:123456789012:*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": [
                "arn:aws:logs:region:123456789012:log-group:/aws/lambda/function
 name:*"
            ]
        }
    ]
}
```

5. Pilih Next: Tags (Selanjutnya: Tanda).

6. (Opsional) Tambahkan satu atau beberapa pasangan nilai kunci tag untuk mengatur, melacak, atau mengontrol akses kebijakan ini.
7. Pilih Next: Review (Selanjutnya: Tinjauan).
8. Pada halaman Tinjau kebijakan, untuk Nama, masukkan nama untuk kebijakan inline, seperti **`amiLambda`**.
9. Pilih Buat kebijakan.
10. Ulangi langkah 2 dan 3.
11. Tempel kebijakan berikut. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ssm:PutParameter",
            "Resource": "arn:aws:ssm:region:123456789012:parameter/latestAmi"
        },
        {
            "Effect": "Allow",
            "Action": "ssm:DescribeParameters",
            "Resource": "*"
        }
    ]
}
```

12. Pilih Next: Tags (Selanjutnya: Tanda).
13. (Opsional) Tambahkan satu atau beberapa pasangan nilai kunci tag untuk mengatur, melacak, atau mengontrol akses kebijakan ini.
14. Pilih Next: Review (Selanjutnya: Tinjauan).
15. Pada halaman Tinjau kebijakan, untuk Nama, masukkan nama untuk kebijakan inline, seperti **`amiParameter`**.
16. Pilih Buat kebijakan.
17. Di panel navigasi, pilih Peran, lalu pilih Buat peran.
18. Segera di bawah Kasus penggunaan, pilih Lambda, lalu pilih Berikutnya.
19. Pada halaman Tambahkan izin, gunakan bidang Pencarian untuk menemukan dua kebijakan yang Anda buat sebelumnya.

20. Pilih kotak centang di samping kebijakan, lalu pilih Berikutnya.
21. Untuk Nama peran, masukkan nama untuk peran baru Anda, seperti **`lambda-ssm-role`** atau nama lain yang Anda inginkan.

    > **Note**
    >
    > Karena berbagai entitas mungkin mereferensikan peran, Anda tidak dapat mengubah nama peran setelah dibuat.

22. (Opsional) Tambahkan satu atau beberapa pasangan nilai kunci tag untuk mengatur, melacak, atau mengontrol akses untuk peran ini, lalu pilih Buat peran.

Tugas 3: Buat AWS Lambda fungsi

Gunakan prosedur berikut untuk membuat fungsi Lambda yang secara otomatis memperbarui nilai `latestAmi` parameter.

Untuk membuat fungsi Lambda

1. Masuk ke AWS Management Console dan buka konsol AWS Lambda di https://console.aws.amazon.com/lambda/.
2. Pilih Buat fungsi.
3. Pilih halaman Buat fungsi, pilih Penulis dari scratch.
4. Untuk Nama fungsi, masukkan **`Automation-UpdateSsmParam`**.
5. Untuk Waktu pengoperasian, pilih Python 3.8.
6. Untuk Arsitektur, pilih jenis prosesor komputer untuk Lambda untuk digunakan untuk menjalankan fungsi, x86_64 atau arm64,
7. Di bagian Izin, perluas Ubah peran eksekusi default.
8. Pilih Gunakan peran yang sudah ada, dan kemudian pilih peran layanan untuk Lambda yang Anda buat di Tugas 2.
9. Pilih Buat fungsi.
10. Di area sumber Kode, pada tab lambda_function, hapus kode yang telah diisi sebelumnya di bidang, lalu tempel contoh kode berikut.

```
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. Pilih File, Simpan.
12. Untuk menguji fungsi Lambda, dari menu Test, pilih Configure test event.
13. Untuk Nama peristiwa, masukkan nama untuk peristiwa pengujian, seperti **`MyTestEvent`**.
14. Ganti teks yang ada dengan JSON berikut. Ganti *`ID AMI`* dengan informasi Anda sendiri untuk menetapkan nilai `latestAmi` parameter Anda.

```
{
   "parameterName":"latestAmi",
   "parameterValue":"AMI ID"
}
```

15. Pilih Save (Simpan).
16. Pilih Uji untuk menguji fungsi. Pada tab Hasil eksekusi, status harus dilaporkan sebagai Berhasil, bersama dengan detail lain tentang pembaruan.

Tugas 4: Buat runbook dan tambal AMI

Gunakan prosedur berikut untuk membuat dan menjalankan runbook yang mem-patch AMI yang Anda tentukan untuk parameter latestAmi. Setelah otomatisasi selesai, nilai LatEstami diperbarui dengan ID yang baru ditambal. AMI Otomatisasi berikutnya menggunakan AMI yang dibuat oleh eksekusi sebelumnya.

Untuk membuat dan menjalankan runbook

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.

3. Untuk Buat dokumen, pilih Otomasi.
4. Untuk Nama, masukkan **`UpdateMyLatestWindowsAmi`**.
5. Pilih tab Editor, dan kemudian pilih Edit.
6. Pilih OK saat diminta.
7. Di bidang editor Dokumen, ganti konten default dengan konten runbook sampel YAMAL berikut.

```
---
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. Pilih Buat otomatisasi.
9. Di panel navigasi, pilih Otomatisasi, lalu pilih Eksekusi otomatisasi.
10. Di halaman Pilih dokumen, pilih tab Dimiliki oleh saya.
11. Cari UpdateMyLatestWindowsAmirunbook, dan pilih tombol di UpdateMyLatestWindowsAmikartu.
12. Pilih Berikutnya.
13. Pilih Eksekusi sederhana.
14. Tentukan nilai untuk parameter input.
15. Pilih Eksekusi.
16. Setelah otomatisasi selesai, pilih Parameter Storedi panel navigasi dan konfirmasikan bahwa nilai baru untuk `latestAmi` cocok dengan nilai yang dikembalikan oleh otomatisasi. Anda juga dapat memverifikasi AMI ID baru yang cocok dengan output otomatisasi di bagian AMI dari konsol Amazon EC2.

Memperbarui AMIs menggunakan Otomasi dan Jenkins

Jika organisasi Anda menggunakan Jenkins perangkat lunak dalam pipeline CI/CD, Anda dapat menambahkan Automation sebagai langkah pasca-build untuk pra-instal rilis aplikasi ke (). Amazon Machine Images AMIs Otomasi adalah kemampuan AWS Systems Manager. Anda juga dapat menggunakan fitur Jenkins penjadwalan untuk memanggil Otomasi dan membuat irama patching sistem operasi (OS) Anda sendiri.

Contoh di bawah ini menunjukkan cara memanggil Otomasi dari Jenkins server yang berjalan baik lokal atau di Amazon Elastic Compute Cloud (Amazon EC2). Untuk otentikasi, Jenkins server menggunakan AWS kredensional berdasarkan kebijakan IAM yang Anda buat dalam contoh dan lampirkan ke profil instans Anda.

> **Note**
>
> Pastikan untuk mengikuti praktik terbaik Jenkins keamanan saat mengonfigurasi instans Anda.

Sebelum Anda mulai

Selesaikan tugas-tugas berikut sebelum Anda mengonfigurasi Otomasi denganJenkins:

- Selesaikan Perbarui emas AMI menggunakan Otomasi,AWS Lambda, dan Parameter Store contoh. Contoh berikut menggunakan UpdateMyLatestWindowsAmirunbook yang dibuat dalam contoh itu.
- Konfigurasikan IAM role untuk Otomatisasi. Systems Manager memerlukan peran profil instans dan peran layanan ARN untuk memproses otomatisasi. Untuk informasi selengkapnya, lihat Menyiapkan Otomatisasi.

Untuk membuat kebijakan IAM untuk server Jenkins

1. Masuk ke AWS Management Console dan buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Pada panel navigasi, pilih Kebijakan, lalu pilih Buat kebijakan.
3. Pilih tab JSON.
4. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

```
{
        "Version": "2012-10-17",
        "Statement": [
            {
                "Effect": "Allow",
                "Action": "ssm:StartAutomationExecution",
                "Resource": [
                    "arn:aws:ssm:region:account ID:document/
UpdateMyLatestWindowsAmi",
                    "arn:aws:ssm:region:account ID:automation-definition/
UpdateMyLatestWindowsAmi:$DEFAULT"
                ]
            }
        ]
```

```
    }
```

5. Pilih Tinjau kebijakan.
6. Pada halaman Tinjau kebijakan, untuk Nama, masukkan nama untuk kebijakan inline, seperti **`JenkinsPolicy`**.
7. Pilih Buat kebijakan.
8. Di panel navigasi, pilih Peran.
9. Pilih profil instance yang dilampirkan ke Jenkins server Anda.
10. Di tab Izin, pilih Tambahkan izin dan pilih Lampirkan kebijakan.
11. Di bagian Kebijakan izin lainnya, masukkan nama kebijakan yang Anda buat di langkah sebelumnya. Misalnya, JenkinsPolicy.
12. Pilih kotak centang di samping kebijakan Anda, lalu pilih Lampirkan kebijakan.

Gunakan prosedur berikut untuk mengkonfigurasi AWS CLI pada Jenkins server Anda.

Untuk mengkonfigurasi Jenkins server untuk Otomasi

1. Connect ke Jenkins server Anda pada port 8080 menggunakan browser pilihan Anda untuk mengakses antarmuka manajemen.
2. Masukkan kata sandi yang ditemukan di `/var/lib/jenkins/secrets/initialAdminPassword`. Untuk menampilkan kata sandi Anda, jalankan perintah berikut.

```
sudo cat /var/lib/jenkins/secrets/initialAdminPassword
```

3. Skrip Jenkins instalasi mengarahkan Anda ke Jenkins halaman Customize. Pilih Pasang plugin yang disarankan.
4. Setelah instalasi selesai, pilih Administrator Credentials, pilih Save Credentials, dan kemudian pilih Start Using. Jenkins
5. Di panel navigasi kiri, pilih KelolaJenkins, lalu pilih Kelola Plugin.
6. Pilih tab Tersedia, dan kemudian masukkan **`Amazon EC2 plugin`**.
7. Pilih kotak centang untuk **`Amazon EC2 plugin`**, dan kemudian pilih Instal tanpa memulai ulang.
8. Ketika instalasi selesai, pilih Kembali ke halaman atas.
9. Pilih Kelola Jenkins, lalu pilih Kelola node dan awan.
10. Di bagian Konfigurasi Awan, pilih Tambahkan cloud baru, lalu pilih Amazon EC2.

11. Masukkan informasi Anda di bidang yang tersisa. Pastikan Anda memilih opsi Gunakan profil instans EC2 untuk mendapatkan kredensi.

Gunakan prosedur berikut untuk mengonfigurasi Jenkins proyek Anda untuk menjalankan Otomasi.

Untuk mengonfigurasi Jenkins server Anda untuk memanggil Otomasi

1. Buka Jenkins konsol di browser web.
2. Pilih proyek yang ingin Anda konfigurasikan dengan Otomatisasi, lalu pilih Konfigurasi.
3. Pada tab Membanguntab, pilihTambah Langkah Bangun.
4. Pilih Eksekusi shell atau Eksekusi perintah batch Windows(tergantung pada sistem operasi Anda).
5. Di bidang Command, jalankan AWS CLI perintah seperti berikut ini. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri.

```
aws ssm start-automation-execution \
        --document-name runbook name \
        --region Wilayah AWS of your source AMI \
        --parameters runbook parameters
```

Perintah contoh berikut menggunakan UpdateMyLatestWindowsAmirunbook dan Parameter Systems Manager yang `latestAmi` dibuat diPerbarui emas AMI menggunakan Otomasi,AWS Lambda, dan Parameter Store.

```
aws ssm start-automation-execution \
        --document-name UpdateMyLatestWindowsAmi \
        --parameters \
            "sourceAMIid='{{ssm:latestAmi}}'"
        --region region
```

DalamJenkins, perintah terlihat seperti contoh di screenshot berikut.

6. Dalam Jenkins proyek, pilih Build Now. Jenkinsmengembalikan output mirip dengan contoh berikut.

MemperbaruiAMIs untuk Auto Scaling

Contoh berikut memperbarui grup Auto Scaling dengan yang baru ditambalAMI. Pendekatan ini memastikan bahwa gambar baru yang dibuat secara otomatis tersedia untuk lingkungan komputasi yang berbeda yang menggunakan grup Auto Scaling.

Langkah terakhir otomatisasi dalam contoh ini menggunakan fungsi Python untuk membuat templat peluncuran baru yang menggunakan yang baru ditambalAMI. Lalu grup Auto Scaling diperbarui untuk menggunakan templat Auto Scaling baru. Dalam jenis skenario Auto Scaling, pengguna dapat mengakhiri instans yang ada di grup Auto Scaling untuk memaksa instans baru memulai yang menggunakan gambar baru. Atau, pengguna dapat menunggu dan mengizinkan peristiwa scale-in atau scale-out untuk meluncurkan instans baru secara alami.

Sebelum Anda memulai

Selesaikan tugas-tugas berikut sebelum Anda mulai contoh ini.

- Mengonfigurasi IAM role untuk otomatisasi, kemampuan AWS Systems Manager. Systems Manager memerlukan peran profil instans dan peran layanan ARN untuk memproses otomatisasi. Untuk informasi selengkapnya, lihat Menyiapkan Otomatisasi.

Buat runbook PatchamiAndUpdate ASG

Gunakan prosedur berikut untuk membuat runbook PatchamiAndUpdate ASG yang menambal yangAMI Anda tentukan untuk parameter sourceAMI. Runbook juga memperbarui grup Auto Scaling untuk menggunakan yang terbaru, ditambalAMI.

Untuk membuat dan menjalankan runbook

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.
3. Di menu tarik-turun Buat dokumen, pilih Otomasi.
4. Di bidang Nama, masukkan **`PatchAMIAndUpdateASG`**.
5. Pilih tab Editor, dan pilih Edit.
6. Pilih OKE saat diminta, dan hapus konten di bidang Editor dokumen.
7. Di bidang Editor dokumen, tempel runbook sampel YAML berikut ini.

```
---
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. Pilih Buat otomatisasi.
9. Di panel navigasi, pilih Otomatisasi, lalu pilih Eksekusi otomatisasi.
10. Di halaman Pilih dokumen, pilih tab Dimiliki oleh saya.
11. Cari runbook PatchamiAndUpdate ASG, dan pilih tombol di kartu PatchamiAndUpdate ASG.
12. Pilih Berikutnya.
13. Pilih Eksekusi sederhana.

14. Tentukan nilai untuk parameter input. Pastikan`SubnetId` dan`SecurityGroupIds` Anda menentukan memungkinkan akses ke titik akhir Systems Manager publik, atau titik akhir antarmuka Anda untuk Systems Manager.
15. Pilih Eksekusi.
16. Setelah otomatisasi selesai, di konsol Amazon EC2, pilih Auto Scaling, lalu pilih Templat Peluncuran. Verifikasi bahwa Anda melihat templat baru peluncuran baru, dan menggunakan yang baruAMI.
17. Pilih Auto Scaling, lalu pilih Grup Auto Scaling. Verifikasi grup Auto Scaling baru untuk menggunakan templat Auto Scaling baru.
18. Akhiri satu atau beberapa instans dalam grup Auto Scaling Anda. Penggantian akan diluncurkan menggunakan baru baruAMI.

Menggunakan runbookAWS Support layanan mandiri

Bagian ini menjelaskan cara menggunakan beberapa otomatisasi layanan mandiri yang dibuat oleh AWS Support tim. Otomatisasi ini membantu Anda mengelola AWS sumber daya.

Support Alur kerja Otomatisasi

Support otomatisasi Workflow (SAW) adalah runbook otomatisasi ditulis dan dikelola oleh AWS Support tim. Runbook ini membantu Anda memecahkan masalah umum dengan AWS sumber daya Anda, pantau dan identifikasi masalah jaringan secara proaktif, kumpulkan dan analisis log, dan banyak lagi.

SAW runbook menggunakan **`AWSSupport`**prefiks. Sebagai contoh, `AWSSupport-ActivateWindowsWithAmazonLicense`.

Selain itu,AWS Pelanggan Korporasi dan Support Business juga memiliki akses ke runbook yang menggunakan **`AWSPremiumSupport`**prefiks. Sebagai contoh, `AWSPremiumSupport-TroubleshootEC2DiskUsage`.

Untuk mempelajari selengkapnya tentang AWS Support, lihat Memulai denganAWS Support.

Topik

- Jalankan alat EC2Rescue pada instans yang tidak dapat dijangkau
- Reset password dan kunci SSH pada Instans EC2

Jalankan alat EC2Rescue pada instans yang tidak dapat dijangkau

EC2Rescue dapat membantu Anda mendiagnosis dan memecahkan masalah di instans Amazon Elastic Compute Cloud (Amazon EC2) untuk Linux dan Windows Server. Anda dapat menjalankan alat secara manual, seperti yang dijelaskan di Menggunakan EC2Rescue untuk Linux Server dan Menggunakan EC2Rescue untuk Windows Server. Atau, Anda dapat menjalankan alat secara otomatis dengan menggunakan Systems Manager Automation dan **`AWSSupport-ExecuteEC2Rescue`**runbook. Otomasi adalah kemampuan AWS Systems Manager. **`AWSSupport-ExecuteEC2Rescue`**Runbook dirancang untuk melakukan kombinasi tindakan, AWS CloudFormation tindakan, dan fungsi Lambda Systems Manager yang mengotomatiskan langkah-langkah yang biasanya diperlukan untuk menggunakan EC2Rescue.

Anda dapat menggunakan **`AWSSupport-ExecuteEC2Rescue`**runbook untuk memecahkan masalah dan berpotensi memulihkan berbagai jenis masalah sistem operasi (OS). Instans dengan volume root terenkripsi tidak didukung. Lihat topik berikut untuk daftar lengkap:

Windows: Lihat Tindakan penyelamatan di Menggunakan EC2Rescue untuk Windows Server dengan Baris Perintah.

Linux dan macOS: Beberapa EC2Rescue untuk modul Linux mendeteksi dan mencoba untuk memperbaiki masalah. Untuk informasi selengkapnya, lihat `aws-ec2rescue-linux`dokumentasi untuk setiap modul diGitHub.

Cara kerjanya

Pemecahan masalah instans dengan otomatisasi dan **`AWSSupport-ExecuteEC2Rescue`** runbook bekerja sebagai berikut:

- Anda menentukan ID instans yang tidak dapat dijangkau dan memulai runbook.
- Sistem menciptakan VPC sementara, dan kemudian menjalankan serangkaian fungsi Lambda untuk mengonfigurasi VPC.
- Sistem mengidentifikasi subnet untuk VPC sementara Anda di Availability Zone yang sama dengan instans asli Anda.
- Sistem meluncurkan instans pembantu berkemampuan SSM sementara.
- Sistem mengentikan instans asli Anda, dan membuat cadangan. Selanjutanya, sistem akan melampirkan volume akar asli untuk instans pembantu.
- Sistem menggunakan Run Command untuk menjalankan EC2Rescue pada instance helper. EC2Rescue mengidentifikasi dan mencoba untuk memperbaiki masalah pada volume akar asli terlampir. Setelah selesai, EC2Rescue melampirkan kembali volume root ke instans asli.

- Sistem memulai ulang instans asli Anda, dan mengakhiri instans sementara. Sistem ini juga menghentikan VPC sementara dan fungsi Lambda yang dibuat pada permulaan otomatisasi.

Sebelum Anda memulai

Sebelum Anda menjalankan otomatisasi berikut, lakukan solusi berikut:

- Salin ID instans dari instans yang tidak terjangkau. Anda akan menentukan ID ini dalam prosedur.
- Opsional, kumpulkan ID subnet di zona ketersediaan yang sama sebagai instans yang dapat dijangkau. Instans EC2Rescue akan dibuat di subnet ini. Jika Anda tidak menentukan subnet, maka Automation membuat VPC sementara baru di Anda. Akun AWS Verifikasi bahwa Anda Akun AWS memiliki setidaknya satu VPC yang tersedia. Secara default, Anda dapat membuat lima VPC di Wilayah. Jika Anda telah membuat lima VPC di Wilayah, otomatisasi gagal tanpa membuat perubahan pada instans Anda. Untuk informasi selengkapnya tentang kuota Amazon VPC, lihat VPC dan Subnet di Panduan Pengguna Amazon VPC.
- Secara opsional, Anda dapat membuat dan menentukan peran AWS Identity and Access Management (IAM) untuk Otomasi. Jika Anda tidak menentukan peran ini, Otomatisasi beroperasi dalam konteks pengguna yang dipanggil otomatisasi.

Memberikan **`AWSSupport-EC2Rescue`** izin untuk melakukan tindakan pada instans Anda

EC2Rescue membutuhkan izin untuk melakukan serangkaian tindakan pada instans Anda selama otomatisasi. Tindakan ini meminta layanan AWS Lambda, IAM, dan Amazon EC2 untuk mencoba memperbaiki masalah dengan instans Anda dengan aman dan aman. Jika Anda memiliki izin tingkat Administrator di dan/atau Akun AWS VPC, Anda mungkin dapat menjalankan otomatisasi tanpa mengonfigurasi izin, seperti yang dijelaskan di bagian ini. Jika Anda tidak memiliki izin tingkat administrator, maka Anda atau administrator harus mengonfigurasi izin dengan menggunakan salah satu opsi berikut.

- Memberikan izin menggunakan kebijakan IAM
- Memberikan izin dengan menggunakan template AWS CloudFormation

Memberikan izin menggunakan kebijakan IAM

Anda dapat melampirkan kebijakan IAM berikut ke pengguna, grup, atau peran Anda sebagai kebijakan inline; atau, Anda dapat membuat kebijakan terkelola IAM baru dan melampirkannya ke pengguna, grup, atau peran Anda. Untuk informasi selengkapnya tentang menambahkan kebijakan

sebaris ke pengguna, grup, atau peran Anda, lihat Bekerja Dengan Kebijakan Sebaris. Untuk informasi lebih lanjut tentang membuat kebijakan terkelola baru, lihat Bekerja Dengan Kebijakan Terkelola.

> **Note**
>
> Jika Anda membuat kebijakan terkelola IAM baru, Anda juga harus melampirkan kebijakan AutomationRole terkelola AmazonSSM agar instance Anda dapat berkomunikasi dengan Systems Manager API.

Kebijakan IAM untuk AWSSupport -EC2Rescue

Ganti *ID akun* dengan informasi Anda sendiri.

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Action": [
            "lambda:InvokeFunction",
            "lambda:DeleteFunction",
            "lambda:GetFunction"
         ],
         "Resource": "arn:aws:lambda:*:account ID:function:AWSSupport-EC2Rescue-*",
         "Effect": "Allow"
      },
      {
         "Action": [
            "s3:GetObject",
            "s3:GetObjectVersion"
         ],
         "Resource": [
            "arn:aws:s3:::awssupport-ssm.*/*.template",
            "arn:aws:s3:::awssupport-ssm.*/*.zip"
         ],
         "Effect": "Allow"
      },
      {
         "Action": [
            "iam:CreateRole",
            "iam:CreateInstanceProfile",
```

```
            "iam:GetRole",
            "iam:GetInstanceProfile",
            "iam:PutRolePolicy",
            "iam:DetachRolePolicy",
            "iam:AttachRolePolicy",
            "iam:PassRole",
            "iam:AddRoleToInstanceProfile",
            "iam:RemoveRoleFromInstanceProfile",
            "iam:DeleteRole",
            "iam:DeleteRolePolicy",
            "iam:DeleteInstanceProfile"
         ],
         "Resource": [
            "arn:aws:iam::account ID:role/AWSSupport-EC2Rescue-*",
            "arn:aws:iam::account ID:instance-profile/AWSSupport-EC2Rescue-*"
         ],
         "Effect": "Allow"
      },
      {
         "Action": [
            "lambda:CreateFunction",
            "ec2:CreateVpc",
            "ec2:ModifyVpcAttribute",
            "ec2:DeleteVpc",
            "ec2:CreateInternetGateway",
            "ec2:AttachInternetGateway",
            "ec2:DetachInternetGateway",
            "ec2:DeleteInternetGateway",
            "ec2:CreateSubnet",
            "ec2:DeleteSubnet",
            "ec2:CreateRoute",
            "ec2:DeleteRoute",
            "ec2:CreateRouteTable",
            "ec2:AssociateRouteTable",
            "ec2:DisassociateRouteTable",
            "ec2:DeleteRouteTable",
            "ec2:CreateVpcEndpoint",
            "ec2:DeleteVpcEndpoints",
            "ec2:ModifyVpcEndpoint",
            "ec2:Describe*"
         ],
         "Resource": "*",
         "Effect": "Allow"
      }
```

```
    ]
}
```

Memberikan izin dengan menggunakan template AWS CloudFormation

AWS CloudFormation mengotomatiskan proses pembuatan peran dan kebijakan IAM dengan menggunakan templat yang telah dikonfigurasi sebelumnya. Gunakan prosedur berikut untuk membuat IAM role yang diperlukan dan kebijakan untuk Otomatisasi EC2Rescue dengan menggunakan AWS CloudFormation.

Untuk membuat kebijakan dan IAM role yang diperlukan untuk EC2Rescue

1. Unduh `AWSSupport-EC2RescueRole.zip` dan ekstrak `AWSSupport-EC2RescueRole.json` file ke direktori pada mesin lokal Anda.
2. Jika Anda Akun AWS berada di partisi khusus, edit template untuk mengubah nilai ARN ke yang untuk partisi Anda.

   Misalnya, untuk Wilayah Cina, ubah semua kasus `arn:aws` ke `arn:aws-cn`.
3. Masuk ke AWS Management Console dan buka AWS CloudFormation konsol di https://console.aws.amazon.com/cloudformation.
4. Pilih Buat tumpukan, Dengan sumber daya baru (standar).
5. Pada halaman Buat tumpukan, untuk Prasyarat - Siapkan templat, pilih Templat sudah siap.
6. Untuk Tentukan templat, pilih Unggah file templat.
7. Pilih Pilih file, lalu telusuri ke dan pilih `AWSSupport-EC2RescueRole.json` file dari direktori tempat Anda mengekstraknya.
8. Pilih Berikutnya.
9. Pada halaman Tentukan detail tumpukan, untuk bidang Nama tumpukan, masukkan nama untuk mengidentifikasi tumpukan ini, dan kemudian pilih Berikutnya.
10. (Opsional) Dalam area Tag, terapkan satu pasangan nilai kunci tag atau lebih ke parameter.

    Tag adalah metadata opsional yang Anda tetapkan ke sumber daya. Tanda memungkinkan Anda untuk mengategorikan sumber daya dengan berbagai cara, seperti berdasarkan tujuan, pemilik, atau lingkungan. Misalnya, Anda mungkin ingin menandai tumpukan untuk mengidentifikasi jenis tugas yang dijalankannya, jenis target atau sumber daya lainnya, dan lingkungan tempat ia dijalankan.
11. Pilih Berikutnya

12. Pada halaman Ulasan, tinjau detail tumpukan, lalu gulir ke bawah dan pilih opsi Saya akui yang AWS CloudFormation mungkin membuat sumber daya IAM.
13. Pilih Buat tumpukan.

    AWS CloudFormation menunjukkan status CREATE_IN_PROGRESS selama beberapa menit. Status berubah menjadi CREATE_COMPLETE setelah tumpukan dibuat. Anda juga dapat memilih ikon refresh untuk memeriksa status proses pembuatan.
14. Di daftar Tumpukan, pilih tombol pilihan tumpukan yang baru saja Anda buat, dan kemudian pilih kotak tab Output.
15. Catat Nilai. Itu adalah ARN dari. AssumeRole Anda menentukan ARN ini ketika menjalankan otomatisasi dalam prosedur berikutnya, Menjalankan Otomatisasi.

Menjalankan Otomatisasi

> **Important**
>
> Otomatisasi berikut menghentikan instans yang tidak dapat dijangkau. Menghentikan contoh dapat mengakibatkan hilangnya data pada volume penyimpanan instans terlampir (jika ada). Menghentikan instans juga dapat menyebabkan IP publik berubah, jika tidak ada Elastic IP terkait.

Untuk menjalankan **`AWSSupport-ExecuteEC2Rescue`** Otomatisasi

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Pada panel navigasi, pilih Otomatisasi.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Otomasi.
3. Pilih Eksekusi otomatisasi.
4. Di bagian Dokumen otomatisasi, pilih Dimiliki oleh Amazon dari daftar.
5. Dalam daftar runbook, pilih tombol di kartu untuk **`AWSSupport-ExecuteEC2Rescue`**, lalu pilih Berikutnya.

6. Pada halaman Eksekusi dokumen otomatisasi, pilih Eksekusi sederhana.
7. Di bagian Detail dokumen verifikasi bahwa Versi dokumen diatur ke versi default tertinggi. Misalnya, $DEFAULT atau 3 (default).
8. Di bagian Parameter input, tentukan parameter berikut:
   a. Untuk UnreachableInstanceId, tentukan ID dari instance yang tidak dapat dijangkau.
   b. (Opsional) Untuk EC2 RescueInstanceType, tentukan jenis instance untuk instance EC2Rescue. Nilai instans default adalah `t2.medium`.
   c. Untuk AutomationAssumeRole, jika Anda membuat peran untuk Otomasi ini dengan menggunakan AWS CloudFormation prosedur yang dijelaskan sebelumnya dalam topik ini, lalu pilih ARN dari AssumeRole yang Anda buat di AWS CloudFormation konsol.
   d. (Opsional) Untuk LogDestination, tentukan bucket S3 jika Anda ingin mengumpulkan log tingkat sistem operasi saat memecahkan masalah instance Anda. Log secara otomatis diunggah ke bucket yang ditentukan.
   e. Untuk SubnetId, tentukan subnet di VPC yang ada di zona ketersediaan yang sama dengan instance yang tidak dapat dijangkau. Secara default, Systems Manager menciptakan VPC baru, tetapi Anda dapat menentukan subnet di VPC yang ada jika Anda ingin.

   > **Note**
   >
   > Jika Anda tidak melihat opsi untuk menentukan bucket atau ID subnet, verifikasi bahwa Anda menggunakan versi Default terbaru dari runbook.

9. (Opsional) Dalam area Tag, terapkan satu pasangan nama/nilai kunci tag atau lebih untuk membantu mengidentifikasi otomatisasi, misalnya `Key=Purpose,Value=EC2Rescue`.
10. Pilih Eksekusi.

Runbook membuat cadangan AMI sebagai bagian dari otomatisasi. Semua sumber daya lain yang dibuat oleh otomatisasi dihapus secara otomatis, tetapi AMI ini tetap ada di akun Anda. AMI Dinamai menggunakan konvensi berikut:

Cadangan AMI: AWSSupport -EC2Rescue: *`UnreachableInstanceId`*

Anda dapat menemukan ini AMI di konsol Amazon EC2 dengan mencari ID eksekusi Otomatisasi.

Reset password dan kunci SSH pada Instans EC2

Anda dapat menggunakan `AWSSupport-ResetAccess` runbook untuk mengaktifkan kembali pembuatan kata sandi Administrator lokal secara otomatis di instans Amazon Elastic Compute Cloud Amazon EC2 untuk Windows Server dan untuk menghasilkan kunci SSH baru pada instans EC2 untuk Linux. `AWSSupport-ResetAccess`Runbook dirancang untuk melakukan kombinasi AWS Systems Manager tindakan, AWS CloudFormation tindakan, dan AWS Lambda fungsi yang mengotomatiskan langkah-langkah yang biasanya diperlukan untuk mengatur ulang kata sandi administrator lokal.

Anda dapat menggunakan Otomasi, kemampuan AWS Systems Manager, dengan `AWSSupport-ResetAccess` runbook untuk memecahkan masalah berikut:

Windows

Anda kehilangan key pair EC2: Untuk mengatasi masalah ini, Anda dapat menggunakan AWSSupport- ResetAccess runbook untuk membuat sandi yang diaktifkan AMI dari instans Anda saat ini, meluncurkan instance baru dari AMI, dan memilih key pair yang Anda miliki.

Anda kehilangan kata sandi Administrator lokal: Untuk mengatasi masalah ini, Anda dapat menggunakan `AWSSupport-ResetAccess` runbook untuk menghasilkan kata sandi baru yang dapat Anda dekripsi dengan pasangan kunci EC2 saat ini.

Linux

Anda kehilangan key pair EC2 Anda, atau Anda mengonfigurasi akses SSH ke instance dengan kunci yang hilang: Untuk mengatasi masalah ini, Anda dapat menggunakan `AWSSupport-ResetAccess` runbook untuk membuat kunci SSH baru untuk instans Anda saat ini, yang memungkinkan Anda untuk terhubung ke instance lagi.

> **Note**
>
> Jika instans EC2 Anda Windows Server dikonfigurasi untuk Systems Manager, Anda juga dapat mengatur ulang kata sandi Administrator lokal Anda dengan menggunakan EC2Rescue dan. AWS Systems Manager Run Command Untuk informasi selengkapnya, lihat Menggunakan EC2Rescue untuk Windows Server dengan Systems Manager Run Command di Panduan Pengguna Amazon EC2 untuk Instans Windows.

Informasi terkait

Connect ke instans Linux Anda dari Windows menggunakan PuTTY di Panduan Pengguna Amazon EC2 untuk Instans Linux

Cara kerjanya

Pemecahan masalah instans dengan otomatisasi dan `AWSSupport-ResetAccess` runbook bekerja sebagai berikut:

- Anda menentukan ID instans yang tidak dapat dijangkau dan menjalankan runbook.
- Sistem menciptakan VPC sementara, dan kemudian menjalankan serangkaian fungsi Lambda untuk mengonfigurasi VPC.
- Sistem mengidentifikasi subnet untuk VPC sementara Anda di Availability Zone yang sama dengan instans asli Anda.
- Sistem meluncurkan instans pembantu berkemampuan SSM sementara.
- Sistem mengentikan instans asli Anda, dan membuat cadangan. Selanjutanya, sistem akan melampirkan volume akar asli untuk instans pembantu.
- Sistem menggunakan Run Command untuk menjalankan EC2Rescue pada instance helper. Di Windows, EC2Rescue memungkinkan pembuatan kata sandi untuk Administrator lokal dengan menggunakan EC2config atau EC2launch pada volume root asli yang terlampir. Di Linux, EC2Rescue menghasilkan dan menyuntikkan kunci SSH baru dan menyimpan kunci privat, dienkripsi, di. Parameter Store Setelah selesai, EC2Rescue melampirkan kembali volume root ke instans asli.
- Sistem membuat Amazon Machine Image (AMI) baru dari instance Anda, sekarang pembuatan kata sandi diaktifkan. Anda dapat menggunakan ini AMI untuk membuat instans EC2 baru, dan mengasosiasikan pasangan kunci baru jika diperlukan.
- Sistem memulai ulang instans asli Anda, dan mengakhiri instans sementara. Sistem ini juga menghentikan VPC sementara dan fungsi Lambda yang dibuat pada permulaan otomatisasi.
- Windows: Instans Anda menghasilkan kata sandi baru yang kodenya dapat Anda pecahkan dari konsol Amazon EC2 menggunakan pasangan kunci saat ditugaskan untuk instans.

  Linux*: Anda dapat SSH ke instance dengan menggunakan kunci SSH yang disimpan di Systems Manager Parameter Store sebagai* ***/ec2rl/openssh/instance ID /key.***

Sebelum Anda mulai

Sebelum Anda menjalankan otomatisasi berikut, lakukan solusi berikut:

- Salin ID instans tempat Anda ingin menyetel ulang kata sandi Administrator. Anda akan menentukan ID ini dalam prosedur.
- Opsional, kumpulkan ID subnet di zona ketersediaan yang sama sebagai instans yang dapat dijangkau. Instans EC2Rescue akan dibuat di subnet ini. Jika Anda tidak menentukan subnet, maka Automation membuat VPC sementara baru di Anda. Akun AWS Verifikasi bahwa Anda Akun AWS memiliki setidaknya satu VPC yang tersedia. Secara default, Anda dapat membuat lima VPC di Wilayah. Jika Anda telah membuat lima VPC di Wilayah, otomatisasi gagal tanpa membuat perubahan pada instans Anda. Untuk informasi selengkapnya tentang kuota Amazon VPC, lihat VPC dan Subnet di Panduan Pengguna Amazon VPC.
- Secara opsional, Anda dapat membuat dan menentukan peran AWS Identity and Access Management (IAM) untuk Otomasi. Jika Anda tidak menentukan peran ini, Otomatisasi beroperasi dalam konteks pengguna yang dipanggil otomatisasi.

Memberikan izin AWSSupport -EC2Rescue untuk melakukan tindakan pada instans Anda

EC2Rescue membutuhkan izin untuk melakukan serangkaian tindakan pada instans Anda selama otomatisasi. Tindakan ini meminta layanan AWS Lambda, IAM, dan Amazon EC2 untuk mencoba memperbaiki masalah dengan instans Anda dengan aman dan aman. Jika Anda memiliki izin tingkat Administrator di dan/atau Akun AWS VPC, Anda mungkin dapat menjalankan otomatisasi tanpa mengonfigurasi izin, seperti yang dijelaskan di bagian ini. Jika Anda tidak memiliki izin tingkat administrator, maka Anda atau administrator harus mengonfigurasi izin dengan menggunakan salah satu opsi berikut.

- Memberikan izin menggunakan kebijakan IAM
- Memberikan izin dengan menggunakan template AWS CloudFormation

Memberikan izin menggunakan kebijakan IAM

Anda dapat melampirkan kebijakan IAM berikut ke pengguna, grup, atau peran Anda sebagai kebijakan inline; atau, Anda dapat membuat kebijakan terkelola IAM baru dan melampirkannya ke pengguna, grup, atau peran Anda. Untuk informasi selengkapnya tentang menambahkan kebijakan sebaris ke pengguna, grup, atau peran Anda, lihat Bekerja Dengan Kebijakan Sebaris. Untuk

informasi lebih lanjut tentang membuat kebijakan terkelola baru, lihat Bekerja Dengan Kebijakan Terkelola.

> **Note**
>
> Jika Anda membuat kebijakan terkelola IAM baru, Anda juga harus melampirkan kebijakan AutomationRole terkelola AmazonSSM agar instance Anda dapat berkomunikasi dengan Systems Manager API.

Kebijakan IAM untuk **`AWSSupport-ResetAccess`**

Ganti *ID akun* dengan informasi Anda sendiri.

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Action": [
            "lambda:InvokeFunction",
            "lambda:DeleteFunction",
            "lambda:GetFunction"
         ],
         "Resource": "arn:aws:lambda:*:account ID:function:AWSSupport-EC2Rescue-*",
         "Effect": "Allow"
      },
      {
         "Action": [
            "s3:GetObject",
            "s3:GetObjectVersion"
         ],
         "Resource": [
            "arn:aws:s3:::awssupport-ssm.*/*.template",
            "arn:aws:s3:::awssupport-ssm.*/*.zip"
         ],
         "Effect": "Allow"
      },
      {
         "Action": [
            "iam:CreateRole",
            "iam:CreateInstanceProfile",
            "iam:GetRole",
```

```
            "iam:GetInstanceProfile",
            "iam:PutRolePolicy",
            "iam:DetachRolePolicy",
            "iam:AttachRolePolicy",
            "iam:PassRole",
            "iam:AddRoleToInstanceProfile",
            "iam:RemoveRoleFromInstanceProfile",
            "iam:DeleteRole",
            "iam:DeleteRolePolicy",
            "iam:DeleteInstanceProfile"
         ],
         "Resource": [
            "arn:aws:iam::account ID:role/AWSSupport-EC2Rescue-*",
            "arn:aws:iam::account ID:instance-profile/AWSSupport-EC2Rescue-*"
         ],
         "Effect": "Allow"
      },
      {
         "Action": [
            "lambda:CreateFunction",
            "ec2:CreateVpc",
            "ec2:ModifyVpcAttribute",
            "ec2:DeleteVpc",
            "ec2:CreateInternetGateway",
            "ec2:AttachInternetGateway",
            "ec2:DetachInternetGateway",
            "ec2:DeleteInternetGateway",
            "ec2:CreateSubnet",
            "ec2:DeleteSubnet",
            "ec2:CreateRoute",
            "ec2:DeleteRoute",
            "ec2:CreateRouteTable",
            "ec2:AssociateRouteTable",
            "ec2:DisassociateRouteTable",
            "ec2:DeleteRouteTable",
            "ec2:CreateVpcEndpoint",
            "ec2:DeleteVpcEndpoints",
            "ec2:ModifyVpcEndpoint",
            "ec2:Describe*"
         ],
         "Resource": "*",
         "Effect": "Allow"
      }
   ]
```

```
}
```

Memberikan izin dengan menggunakan template AWS CloudFormation

AWS CloudFormation mengotomatiskan proses pembuatan peran dan kebijakan IAM dengan menggunakan templat yang telah dikonfigurasi sebelumnya. Gunakan prosedur berikut untuk membuat IAM role yang diperlukan dan kebijakan untuk Otomatisasi EC2Rescue dengan menggunakan AWS CloudFormation.

Untuk membuat kebijakan dan IAM role yang diperlukan untuk EC2Rescue

1. Unduh `AWSSupport-EC2RescueRole.zip` dan ekstrak `AWSSupport-EC2RescueRole.json` file ke direktori pada mesin lokal Anda.
2. Jika Anda Akun AWS berada di partisi khusus, edit template untuk mengubah nilai ARN ke yang untuk partisi Anda.

   Misalnya, untuk Wilayah Cina, ubah semua kasus `arn:aws` ke `arn:aws-cn`.
3. Masuk ke AWS Management Console dan buka AWS CloudFormation konsol di https://console.aws.amazon.com/cloudformation.
4. Pilih Buat tumpukan, Dengan sumber daya baru (standar).
5. Pada halaman Buat tumpukan, untuk Prasyarat - Siapkan templat, pilih Templat sudah siap.
6. Untuk Tentukan templat, pilih Unggah file templat.
7. Pilih Pilih file, lalu telusuri ke dan pilih `AWSSupport-EC2RescueRole.json` file dari direktori tempat Anda mengekstraknya.
8. Pilih Berikutnya.
9. Pada halaman Tentukan detail tumpukan, untuk bidang Nama tumpukan, masukkan nama untuk mengidentifikasi tumpukan ini, dan kemudian pilih Berikutnya.
10. (Opsional) Dalam area Tag, terapkan satu pasangan nilai kunci tag atau lebih ke parameter.

    Tag adalah metadata opsional yang Anda tetapkan ke sumber daya. Tanda memungkinkan Anda untuk mengategorikan sumber daya dengan berbagai cara, seperti berdasarkan tujuan, pemilik, atau lingkungan. Misalnya, Anda mungkin ingin menandai tumpukan untuk mengidentifikasi jenis tugas yang dijalankannya, jenis target atau sumber daya lainnya, dan lingkungan tempat ia dijalankan.
11. Pilih Berikutnya

12. Pada halaman Ulasan, tinjau detail tumpukan, lalu gulir ke bawah dan pilih opsi Saya akui yang AWS CloudFormation mungkin membuat sumber daya IAM.
13. AWS CloudFormation menunjukkan status CREATE_IN_PROGRESS selama beberapa menit. Status berubah menjadi CREATE_COMPLETE setelah tumpukan dibuat. Anda juga dapat memilih ikon refresh untuk memeriksa status proses pembuatan.
14. Dalam daftar tumpukan, pilih opsi di samping tumpukan yang baru saja Anda buat, lalu pilih tab Output.
15. Salin Nilai. Itu adalah ARN dari. AssumeRole Anda akan menentukan ARN ini ketika menjalankan otomatisasi dalam prosedur berikutnya.

Menjalankan Otomatisasi

Prosedur berikut menjelaskan cara menjalankan `AWSSupport-ResetAccess` runbook dengan menggunakan AWS Systems Manager konsol.

> **Important**
>
> Otomatisasi berikut menghentikan instans. Menghentikan contoh dapat mengakibatkan hilangnya data pada volume penyimpanan instans terlampir (jika ada). Menghentikan instans juga dapat menyebabkan IP publik berubah, jika tidak ada Elastic IP terkait. Untuk menghindari perubahan konfigurasi ini, gunakan Run Command untuk mengatur ulang akses. Untuk informasi selengkapnya, lihat Menggunakan EC2Rescue untuk Windows Server dengan Systems Manager Run Command di Panduan Pengguna Amazon EC2 untuk Instans Windows.

Untuk menjalankan AWSSupport - ResetAccess Otomasi

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Pada panel navigasi, pilih Otomatisasi.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Otomasi.
3. Pilih Eksekusi otomatisasi.

4. Di bagian Dokumen otomatisasi, pilih Dimiliki oleh Amazon dari daftar.
5. Dalam daftar runbook, pilih tombol di kartu untuk AWSSupport- ResetAccess, lalu pilih Berikutnya.
6. Pada halaman Eksekusi dokumen otomatisasi, pilih Eksekusi sederhana.
7. Di bagian Detail dokumen verifikasi bahwa Versi dokumen diatur ke versi default tertinggi. Misalnya, $DEFAULT atau 3 (default).
8. Di bagian Parameter input, tentukan parameter berikut:
   a. Untuk InstanceID, tentukan ID instans yang tidak terjangkau.
   b. Untuk SubnetId, tentukan subnet di VPC yang ada di zona ketersediaan yang sama dengan instance yang Anda tentukan. Secara default, Systems Manager menciptakan VPC baru, tetapi Anda dapat menentukan subnet di VPC yang ada jika Anda ingin.

      > **Note**
      >
      > Jika Anda tidak melihat opsi untuk menentukan ID subnet, verifikasi bahwa Anda menggunakan versi Default terbaru dari runbook.

   c. Untuk EC2 RescueInstanceType, tentukan jenis instance untuk instance EC2Rescue. Nilai instans default adalah `t2.medium`.
   d. Untuk AssumeRole, jika Anda membuat peran untuk Otomasi ini dengan menggunakan AWS CloudFormation prosedur yang dijelaskan sebelumnya dalam topik ini, maka tentukan AssumeRole ARN yang Anda catat di AWS CloudFormation konsol.
9. (Opsional) Dalam area Tag, terapkan satu pasangan nama/nilai kunci tag atau lebih untuk membantu mengidentifikasi otomatisasi, misalnya `Key=Purpose,Value=ResetAccess`.
10. Pilih Eksekusi.
11. Untuk memantau kemajuan otomatisasi, pilih otomatisasi berjalan, dan kemudian pilih tab Langkah. Setelah otomatisasi selesai, pilih tab Deskripsi, dan kemudian pilih Tampilkan output untuk melihat hasilnya. Untuk menampilkan output langkah-langkah individual, pilih tab Langkah, dan kemudian pilih Tampilkan Output di samping satu langkah.

Runbook membuat cadangan AMI dan kata sandi yang diaktifkan AMI sebagai bagian dari otomatisasi. Semua sumber lain yang dibuat oleh otomatisasi dihapus secara otomatis, tetapi ini AMIs tetap ada di akun Anda. AMIs Dinamai menggunakan konvensi berikut:

- Backup AMI: `AWSSupport-EC2Rescue:`*`InstanceID`*
- *`AMI yang diaktifkan kata sandi AWSSupport: -EC2Rescue: AMI yang diaktifkan kata sandi dari ID Instance`*

Anda dapat menemukan ini AMIs dengan mencari ID eksekusi Otomatisasi.

Untuk Linux, kunci pribadi SSH baru untuk instans Anda disimpan, dienkripsi, di. Parameter Store *`Nama parameter adalah`* ***`/ec2rl/openssh/instance`*** *`ID /key.`*

Melewati data ke Otomasi menggunakan transformator input

TutorialAWS Systems Manager otomasi ini menunjukkan cara menggunakan fitur transformator input Amazon EventBridge untuk mengekstrak instans Amazon Elastic Compute Cloud (Amazon EC2) dari kejadian perubahan status instans.`instance-id` Otomatisasi adalah kemampuan AWS Systems Manager. Kami menggunakan trafo input untuk menyampaikan data tersebut ke `AWS-CreateImage` target runbook sebagai `InstanceId` parameter input. Aturan ini akan dipicu ketika instans mana pun mengalami perubahan`stopped` status.

Untuk informasi lebih lanjut tentang bekerja dengan transformator input, lihat Tutorial: Gunakan Input Transformer untuk Menyesuaikan Apa yang Dilewatkan ke Target Acara di Panduan EventBridge Pengguna Amazon.

Sebelum Anda memulai

Verifikasi bahwa Anda menambahkan izin yang diperlukan dan kepercayaan kebijakan EventBridge untuk peran layanan Otomatisasi Systems Manager Anda. Untuk informasi selengkapnya, lihat Ringkasan Mengelola Izin Akses ke EventBridge Sumber Daya Anda di Panduan EventBridge Pengguna Amazon.

Cara menggunakan transformator input dengan Otomatisasi

1. Buka EventBridge konsol Amazon di https://console.aws.amazon.com/events/.
2. Di panel navigasi, pilih Aturan.
3. Pilih Buat aturan.
4. Masukkan nama dan deskripsi untuk aturan.

   Aturan tidak boleh memiliki nama yang sama dengan aturan lain di Wilayah yang sama dan di bus kejadian yang sama.

5. Untuk bus peristiwa, pilih bus peristiwa yang ingin Anda kaitkan dengan aturan ini. Jika Anda menginginkan aturan ini menanggapi peristiwa yang berasal dari akun AndaAkun AWS, pilih default. Saat akunLayanan AWS Anda menghasilkan kejadian, peristiwa tersebut akan selalu masuk ke bus kejadian default akun Anda.
6. Untuk jenis Aturan, pilih Aturan dengan pola peristiwa.
7. Pilih Selanjutnya.
8. Untuk Sumber acara, pilih AWSacara atau acara EventBridge mitra.
9. Di bagian Pola acara, pilih Bentuk pola acara.
10. Untuk sumber Event, pilih AWSlayanan.
11. Untuk AWSlayanan, pilih EC2.
12. Untuk Jenis peristiwa, pilih EC2 Instance State-change Notification.
13. Untuk keadaan tertentu (s), pilih berhenti.
14. Pilih Selanjutnya.
15. Untuk jenis Target, pilih AWSlayanan.
16. Untuk Pilih target, pilih Otomatisasi Systems Manager.
17. Untuk Dokumen, pilih AWS-CreateImage.
18. Di bagian Konfigurasikan parameter otomatisasi, pilih Transformer input.
19. Untuk Jalur input, masukkan**`{"instance":"$.detail.instance-id"}`**.
20. Untuk Template, masukkan**`{"InstanceId":[<instance>]}`**.
21. Untuk peran eksekusi, pilih Gunakan peran yang ada dan pilih peran layanan Otomatisasi Anda.
22. Pilih Selanjutnya.
23. (Opsional) Masukkan satu atau lebih tanda untuk aturan. Untuk informasi selengkapnya, lihat Menandai EventBridge Sumber Daya Amazon Anda di Panduan EventBridge Pengguna Amazon.
24. Pilih Selanjutnya.
25. Tinjau detail aturan dan pilih Buat aturan.

Memahami status otomatisasi

Otomatisasi AWS Systems Manager melaporkan detail status informasi tentang berbagai status tindakan otomatisasi atau langkah berjalan melalui ketika Anda menjalankan otomatisasi dan

otomatisasi keseluruhan. Otomatisasi adalah kemampuan AWS Systems Manager. Anda dapat memantau status otomatisasi menggunakan metode berikut:

- Pantau Status eksekusi di konsol Otomatisasi Systems Manager.
- Gunakan alat baris perintah pilihan Anda. Untuk AWS Command Line Interface (AWS CLI), Anda dapat menggunakan describe-automation-step-executionsatau get-automation-execution. Untuk ituAWS Tools for Windows PowerShell, Anda dapat menggunakan Get-SSM atau Get-SSM AutomationStepExecution. AutomationExecution
- Konfigurasikan Amazon EventBridge untuk merespons tindakan atau perubahan status otomatisasi.

Tentang status otomatisasi

Otomatisasi melaporkan detail status untuk tindakan otomatisasi individu selain otomatisasi keseluruhan.

Status otomatisasi keseluruhan dapat berbeda dari status yang dilaporkan oleh tindakan individu atau langkah seperti yang tercantum dalam tabel berikut.

Status mendetail untuk tindakan

Status	Detail
Tertunda	Langkahnya belum mulai berjalan. Jika otomatisasi Anda menggunakan tindakan bersyarat, langkah-langkah tetap dalam keadaan ini setelah otomatisasi selesai jika kondisi tidak bisa menjalankan langkah. Langkah-langkah juga tetap dalam keadaan ini jika otomatisasi dibatalkan sebelum langkah berjalan.
InProgress	Langkah sedang berjalan.
Menunggu	Langkahnya sedang menunggu masukan.
Sukses	Langkah berhasil diselesaikan. Ini adalah status terakhir.

Status	Detail
TimedOut	Langkah atau persetujuan tidak diselesaikan sebelum periode batas waktu yang ditentukan. Ini adalah status terakhir.
Membatalkan	Langkahnya adalah dalam proses berhenti setelah dibatalkan oleh pemohon.
Dibatalkan	Langkah tersebut dihentikan oleh pemohon sebelum selesai. Ini adalah status terakhir.
Gagal	Langkahnya tidak berhasil diselesaikan. Ini adalah status terakhir.
Keluar	Hanya dikembalikan oleh `aws:loop` aksi. Loop tidak sepenuhnya selesai. Sebuah langkah di dalam loop dipindahkan ke langkah luar menggunakan`nextStep`,`onCancel`, atau `onFailure` properti.

Status mendetail untuk otomatisasi

Status	Detail
Tertunda	Otomatisasi belum mulai berjalan.
InProgress	Otomatisasi sedang berjalan.
Menunggu	Otomatisasi sedang menunggu masukan.
Sukses	Otomatisasi berhasil diselesaikan. Ini adalah status terakhir.
TimedOut	Langkah atau persetujuan tidak diselesaikan sebelum periode batas waktu yang ditentukan. Ini adalah status terakhir.

Status	Detail
Membatalkan	Otomatisasi sedang dalam proses berhenti setelah dibatalkan oleh pemohon.
Dibatalkan	Otomatisasi dihentikan oleh pemohon sebelum selesai. Ini adalah status terakhir.
Gagal	Otomatisasi tidak berhasil diselesaikan. Ini adalah status terakhir.

Pemecahan masalah Otomatisasi Systems Manager

Gunakan informasi berikut untuk membantu Anda memecahkan masalah dengan AWS Systems Manager Otomatisasi, kemampuan AWS Systems Manager. Topik ini mencakup tugas khusus untuk menyelesaikan masalah yang didasarkan pada pesan kesalahan Otomatisasi.

Topik

Kesalahan Otomatisasi umum

Bagian ini mencakup informasi tentang kesalahan otomatisasi umum.

VPC tidak didefinisikan 400

Secara default, ketika otomatisasi menjalankan `AWS-UpdateLinuxAmi` runbook atau `AWS-UpdateWindowsAmi` runbook, sistem menciptakan instans sementara dalam VPC default (172.30.0.0/16). Jika Anda menghapus VPC default, Anda akan menerima kesalahan berikut:

```
VPC not defined 400
```

Untuk mengatasi masalah ini, Anda harus menentukan nilai `SubnetId` parameter input.

Eksekusi otomatisasi gagal dimulai

Otomatisasi dapat gagal dengan kesalahan akses ditolak atau kesalahan peran asumsi yang tidak valid jika Anda belum mengonfigurasi peran AWS Identity and Access Management (IAM) dengan benar, dan kebijakan untuk Otomasi.

Akses ditolak

Contoh berikut menjelaskan situasi ketika otomatisasi gagal memulai dengan kesalahan akses ditolak.

Akses Ditolak ke API Systems Manager

Pesan kesalahan: `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)`

- Kemungkinan penyebab 1: Pengguna yang mencoba memulai otomatisasi tidak memiliki izin untuk menjalankan API. `StartAutomationExecution` Untuk mengatasi masalah ini, lampirkan kebijakan IAM yang diperlukan kepada pengguna yang digunakan untuk memulai otomatisasi.
- Kemungkinan penyebab 2: Pengguna yang mencoba memulai otomatisasi memiliki izin untuk menjalankan `StartAutomationExecution` API tetapi tidak memiliki izin untuk menjalankan API dengan menggunakan runbook tertentu. Untuk mengatasi masalah ini, lampirkan kebijakan IAM yang diperlukan kepada pengguna yang digunakan untuk memulai otomatisasi.

Akses Ditolak Karena PassRole Izin Hilang

Pesan kesalahan: `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)`

Pengguna yang mencoba memulai otomatisasi tidak memiliki PassRole izin untuk mengambil peran. Untuk mengatasi masalah ini, lampirkan PassRole kebijakan iam: ke peran pengguna yang mencoba memulai otomatisasi. Untuk informasi selengkapnya, lihat Tugas 2: Lampirkan PassRole kebijakan iam: ke peran Otomasi Anda.

Peran asumsi tidak valid

Ketika Anda menjalankan Otomatisasi, peran asumsi disediakan dalam runbook atau dilewatkan sebagai nilai parameter untuk runbook. Berbagai jenis kesalahan dapat terjadi jika peran aumsi tidak ditentukan atau dikonfigurasi dengan benar.

Peran Asumsi yang Salah Bentuk

Pesan kesalahan: `The format of the supplied assume role ARN isn't valid.` Peran asumsi tidak diformat dengan benar. Untuk mengatasi masalah ini, verifikasi bahwa peran asumsi yang valid ditentukan dalam runbook Anda atau sebagai parameter waktu aktif saat memulai otomatisasi.

Peran Asumsi Tidak Dapat Diasumsikan

Pesan kesalahan: `The defined assume role is unable to be assumed. (Service: AWSSimpleSystemsManagement; Status Code: 400; Error Code: InvalidAutomationExecutionParametersException; Request ID: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)`

- Kemungkinan penyebab 1: Peran asumsi tidak ada. Untuk mengatasi masalah ini, buat peran. Untuk informasi selengkapnya, lihat the section called “Menyiapkan Otomatisasi”. Detail khusus untuk membuat peran ini dijelaskan dalam topik berikut, Tugas 1: Buat peran layanan untuk otomatisasi.
- Kemungkinan penyebab 2: Peran asumsi tidak memiliki hubungan kepercayaan dengan layanan Systems Manager. Untuk mengatasi masalah ini, buat hubungan kepercayaan. Untuk informasi lebih lanjut, lihat Saya Tidak Bisa Mengasumsikan Peran di Panduan Pengguna IAM.

Eksekusi dimulai, tetapi status gagal

Kegagalan khusus tindakan

Runbook berisi langkah-langkah dan langkah-langkah yang dijalankan secara berurutan. Setiap langkah memanggil satu atau beberapa Layanan AWS API. API menentukan input, perilaku, dan output dari langkah. Ada beberapa tempat di mana kesalahan dapat menyebabkan langkah gagal. Pesan kegagalan menunjukkan kapan dan di mana kesalahan terjadi.

Untuk melihat pesan kegagalan di konsol Amazon Elastic Compute Cloud (Amazon EC2), pilih tautan Tampilkan Output dari langkah yang gagal. Untuk melihat pesan kegagalan dari AWS CLI, panggil

`get-automation-execution` dan cari `FailureMessage` atribut dalam sebuah yang gagal `StepExecution`.

Dalam contoh berikut, langkah yang terkait dengan `aws:runInstance` tindakan gagal. Setiap contoh mengeksplorasi jenis kesalahan yang berbeda.

Gambar yang Hilang

Pesan kesalahan: `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.`

Tindakan `aws:runInstances` menerima masukan untuk sebuah `ImageId` yang tidak ada. Untuk mengatasi masalah ini, perbarui runbook atau nilai parameter dengan ID AMI yang benar.

Kebijakan Peran Asumsi Tidak Memiliki Izin yang Memadai

Pesan kesalahan: `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.`

Peran asumsi tidak memiliki izin yang cukup untuk menjalankan `RunInstances` API pada instans EC2. Untuk mengatasi masalah ini, lampirkan kebijakan IAM untuk peran asumsi yang memiliki izin untuk menjalankan `RunInstances` API. Untuk informasi lebih lanjut, lihat Metode 2: Gunakan IAM untuk mengonfigurasi peran untuk Otomatisasi.

Keadaan tak terduga

Pesan kesalahan: `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.`

- Kemungkinan penyebab 1: Ada masalah dengan instans atau layanan Amazon EC2. Untuk mengatasi masalah ini, masuk ke instans atau tinjau log sistem instans untuk memahami mengapa instans mulai mematikan.
- Kemungkinan penyebab 2: Skrip data pengguna yang ditentukan untuk `aws:runInstances` tindakan memiliki masalah atau sintaks yang salah. Verifikasi sintaks skrip data pengguna. Juga, verifikasi bahwa skrip data pengguna tidak mematikan instans, atau menjalankan skrip lain yang mematikan instans.

Referensi Kegagalan Khusus Tindakan

Ketika langkah gagal, pesan kegagalan mungkin menunjukkan layanan yang sedang dijalankan ketika kegagalan terjadi. Tabel berikut mencantumkan layanan yang dijalankan oleh setiap tindakan. Tabel ini juga menyediakan tautan ke informasi tentang setiap layanan.

Action	Layanan AWSdipanggil oleh tindakan ini	Untuk informasi tentang layanan ini	Penyelesaian masalah konten
`aws:runInstances`	Amazon EC2	Panduan Pengguna Amazon EC2 untuk Instans Linux	Pemecahan Masalah Instans EC2
`aws:changeInstanceState`	Amazon EC2	Panduan Pengguna Amazon EC2 untuk Instans Linux	Memecahkan masalah instans EC2
`aws:runCommand`	Systems Manager	AWS Systems Manager Run Command	Memecahkan masalah Run Command Systems Manager
`aws:createImage`	Amazon EC2	Amazon Machine Images	
`aws:createStack`	AWS CloudFormation	AWS CloudFormationPanduan Pengguna	Pemecahan masalah AWS CloudFormation

Action	Layanan AWSdipanggil oleh tindakan ini	Untuk informasi tentang layanan ini	Penyelesaian masalah konten
`aws:deleteStack`	AWS CloudFormation	AWS CloudFormationPanduan Pengguna	Pemecahan masalah AWS CloudFormation
`aws:deleteImage`	Amazon EC2	Gambar Amazon Machines	
`aws:copyImage`	Amazon EC2	Amazon Machine Images	
`aws:createTag`	Amazon EC2, Systems Manager	Sumber Daya dan Tag EC2	
`aws:invokeLambdaFunction`	AWS Lambda	Panduan Developer AWS Lambda	Pemecahan Masalah Lambda

Kesalahan internal layanan otomatisasi

Pesan kesalahan: `Internal Server Error. Please refer to Automation Service Troubleshooting Guide for more diagnosis details.`

Masalah dengan layanan otomatisasi adalah mencegah runbook tertentu berjalan dengan benar. Untuk mengatasi masalah ini, hubungi AWS Support. Sediakan ID eksekusi dan ID pelanggan, jika tersedia.

Eksekusi dimulai, tapi habis waktu

Pesan kesalahan: `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.`

Sebuah langkah di `aws:runInstances` waktu aksi habis. Hal ini dapat terjadi jika langkah tindakan membutuhkan waktu lebih lama untuk menjalankan dari nilai yang ditentukan untuk `timeoutSeconds` dalam langkah. Untuk mengatasi masalah ini, tentukan nilai yang lebih lama untuk `timeoutSeconds` parameter dalam `aws:runInstances` tindakan. Jika itu tidak

menyelesaikan masalah, selidiki mengapa langkah tersebut membutuhkan waktu lebih lama untuk berjalan dari yang diharapkan

AWS Systems Manager Change Calendar

Change Calendar, kemampuan dariAWS Systems Manager, memungkinkan Anda untuk menyiapkan rentang tanggal dan waktu saat tindakan yang Anda tentukan (misalnya, di runbook Otomatisasi Systems Manager) mungkin atau mungkin tidak dilakukan diAkun AWS. DalamChange Calendar, rentang ini disebut peristiwa. Saat membuatChange Calendar entri, Anda membuat dokumen Systems Manager jenis`ChangeCalendar`. DiChange Calendar, dokumen menyimpan data iCalendar 2.0 dalam format plaintext. Peristiwa yang Anda tambahkan keChange Calendar entri menjadi bagian dari dokumen. Untuk memulaiChange Calendar, buka konsol Systems Manager. Di panel navigasi, pilih Change Calendar.

Anda dapat membuat kalender dan peristiwa di konsol Systems Manager. Anda juga dapat mengimpor file iCalendar (`.ics`) yang telah diekspor dari penyedia kalender pihak ketiga yang didukung untuk menambahkan acaranya ke kalender Anda. Penyedia yang didukung mencakup Kalender Google, Microsoft Outlook, dan Kalender iCloud.

Change CalendarEntri dapat menjadi salah satu dari dua jenis:

`DEFAULT_OPEN`, atau Buka secara default

Semua tindakan dapat dijalankan secara default, kecuali selama acara kalender. Selama peristiwa, status`DEFAULT_OPEN` kalender adalah`CLOSED` dan peristiwa yang diblokir dari berjalan.

`DEFAULT_CLOSED`, atau Ditutup secara default

Semua tindakan diblokir secara default, kecuali selama acara kalender. Selama peristiwa, status`DEFAULT_CLOSED` kalender adalah`OPEN` dan tindakan diizinkan untuk dijalankan.

Anda dapat memilih agar semua alur kerja Otomasi terjadwal, jendela pemeliharaan, danState Manager asosiasi ditambahkan secara otomatis ke kalender. Anda juga dapat menghapus salah satu jenis individu dari tampilan kalender.

Siapa yang harus menggunakanChange Calendar?

- AWSpelanggan yang melakukan jenis tindakan berikut:
  - Buat atau jalankan runbook Otomasi.

- Buat permintaan perubahan diChange Manager.
- Jalankan jendela pemeliharaan.
- Buat asosiasi diState Manager.

OtomatisasiChange ManagerMaintenance Windows,, danState Manager semua kemampuanAWS Systems Manager. Dengan mengintegrasikan kemampuan iniChange Calendar, Anda dapat mengizinkan atau memblokir jenis tindakan ini bergantung pada status saat ini dari perubahan kalender yang Anda kaitkan dengan masing-masing jenis tindakan.

- Administrator yang bertanggung jawab untuk menjaga konfigurasi node Systems Manager Systems Manager Systems Manager tetap konsisten, stabil, dan fungsional.

ManfaatChange Calendar

Berikut ini adalah beberapa manfaat dariChange Calendar.

- Meninjau perubahan sebelum diterapkan

  Change CalendarEntri dapat membantu memastikan bahwa perubahan yang berpotensi merusak lingkungan Anda telah ditinjau sebelum diterapkan.

- Menerapkan perubahan hanya pada waktu yang tepat

  Change Calendarentri membantu menjaga lingkungan Anda tetap stabil selama waktu peristiwa. Misalnya, Anda dapat membuatChange Calendar entri untuk memblokir perubahan saat Anda mengharapkan permintaan tinggi pada sumber daya Anda, seperti selama konferensi atau promosi pemasaran publik. Entri kalender juga dapat memblokir perubahan saat Anda mengharapkan dukungan administrator terbatas, seperti selama liburan atau hari libur. Anda dapat menggunakan entri kalender untuk mengizinkan perubahan kecuali untuk waktu-waktu tertentu dalam sehari atau minggu ketika ada dukungan administrator terbatas untuk memecahkan masalah tindakan atau deployment yang gagal.

- Mengetahui status kalender saat ini atau yang akan datang

  Anda dapat menjalankan operasi API `GetCalendarState` Systems Manager untuk menunjukkan kepada Anda status kalender saat ini, status pada waktu tertentu, atau saat berikutnya status kalender dijadwalkan untuk berubah.

- EventBridge dukungan

Kemampuan Systems Manager ini didukung sebagai jenis peristiwa di EventBridge aturan Amazon. Untuk informasi, lihat Pemantauan peristiwa Systems Manager dengan Amazon EventBridge dan Referensi: Pola dan jenis EventBridge acara Amazon untuk Systems Manager.

Topik

Menyiapkan Change Calendar

Selesaikan yang berikut sebelum menggunakanChange Calendar, kemampuan dariAWS Systems Manager.

Menginstal alat baris perintah terbaru

Instal alat baris perintah terbaru untuk mendapatkan informasi status tentang kalender.

Persyaratan	Deskripsi
AWS CLI	(Opsional) Untuk menggunakan AWS Command Line Interface (AWS CLI) untuk mendapatkan informasi status tentang kalender, menginstal rilis AWS CLI terbaru di komputer lokal Anda. Untuk informasi tentang cara menginstal atau meningkatkan CLI, lihat Menginstal, memperbarui, dan menghapus instalasi AWS CLI di AWS Command Line Interface Panduan Pengguna.
AWS Tools for PowerShell	(Opsional) Untuk menggunakan Tools for PowerShell mendapatkan informasi status

Persyaratan	Deskripsi
	tentang kalender, instal rilis terbaru Tools for PowerShell di komputer lokal Anda. Untuk informasi tentang cara menginstal atau meningkatkan Tools forPowerShell, lihat Menginstal AWS Tools for PowerShell dalam Panduan AWS Tools for PowerShell Pengguna.

Menyiapkan izin

Jika pengguna, grup, atau peran Anda diberi izin administrator, Anda memiliki akses penuh keChange Calendar. Jika Anda tidak memiliki izin administrator, maka administrator harus memberikan izin dengan menetapkan kebijakan `AmazonSSMFullAccess` terkelola, atau menetapkan kebijakan yang menyediakan izin yang diperlukan untuk pengguna, grup, atau peran Anda.

Izin berikut diperlukan untuk bekerja denganChange Calendar.

Change Calendarentri

Untuk membuat, memperbarui, atau menghapus Change Calendar entri, termasuk menambahkan dan menghapus peristiwa dari entri, kebijakan yang dilampirkan pada pengguna, grup, atau peran Anda harus mengizinkan tindakan berikut:

- `ssm:CreateDocument`
- `ssm:DeleteDocument`
- `ssm:DescribeDocument`
- `ssm:DescribeDocumentPermission`
- `ssm:GetCalendar`
- `ssm:ListDocuments`
- `ssm:ModifyDocumentPermission`
- `ssm:PutCalendar`
- `ssm:UpdateDocument`
- `ssm:UpdateDocumentDefaultVersion`

Status kalender

Untuk mendapatkan informasi tentang status kalender saat ini atau yang akan datang, kebijakan yang dilampirkan ke pengguna, grup, atau peran Anda harus mengizinkan tindakan berikut:

- `ssm:GetCalendarState`

Acara operasional

Untuk melihat peristiwa operasional, seperti jendela pemeliharaan, asosiasi, dan otomatisasi yang direncanakan, kebijakan yang dilampirkan ke pengguna, grup, atau peran Anda harus mengizinkan tindakan berikut:

- `ssm:DescribeMaintenanceWindows`
- `ssm:DescribeMaintenanceWindowExecution`
- `ssm:DescribeAutomationExecutions`
- `ssm:ListAssociations`

> **Note**
>
> Change Calendarentri yang dimiliki oleh (yaitu, dibuat oleh) akun selain milik Anda bersifat hanya-baca, meskipun dibagikan dengan akun Anda.

Bekerja dengan Change Calendar

Anda dapat menggunakanAWS Systems Managerkonsol untuk menambahkan, mengelola, atau menghapus entri diChange Calendar, suatu kemampuanAWS Systems Manager. Anda juga dapat mengimpor acara dari penyedia kalender pihak ketiga yang didukung dengan mengimpor iCalendar (`.ics`) file yang Anda ekspor dari kalender sumber. Dan, Anda dapat menggunakan`GetCalendarState`Operasi API atau`get-calendar-state` AWS Command Line Interface(AWS CLI) perintah untuk mendapatkan informasi tentang keadaanChange Calendarpada waktu tertentu.

Topik

- Membuat Change Change Change Change Calendar
- Membuat dan mengelola acara diChange Calendar
- Mengimpor dan mengelola acara dari kalender pihak ketiga

- Perbarui kalender perubahan
- Membagikan kalender perubahan
- Menghapus kalender perubahan
- Mendapatkan status perubahan kalender

Membuat Change Change Change Change Calendar

Saat Anda membuat entri diChange Calendar, kemampuan dariAWS Systems Manager, Anda membuat dokumen Systems Manager (SSM dokumen) yang menggunakan`text` format.

Untuk membuat Change Change Change Calendar

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Change Calendar.
3. Pilih Buat kalender.

   -atau-

   Jika halaman Change Calendarberanda terbuka terlebih dahulu, pilih Buat kalender perubahan.
4. Di halaman Buat kalender, di Detail kalender, masukkan nama untuk entri kalender Anda. Nama entri kalender dapat berisi huruf, angka, titik, tanda hubung, dan garis bawah. Nama harus cukup spesifik untuk mengidentifikasi tujuan entri kalender secara sekilas. Contohnya adalah **`support-off-hours`**. Anda tidak dapat memperbarui nama ini setelah membuat entri kalender.
5. (Opsional) Untuk Deskripsi, masukkan deskripsi untuk entri kalender Anda.
6. (Opsional) Di area Impor kalender, pilih Pilih file untuk memilih file iCalendar (`.ics`) yang telah Anda ekspor dari penyedia kalender pihak ketiga. Mengimpor file akan menambahkan acaranya ke kalender Anda.

   Penyedia yang didukung mencakup Kalender Google, Microsoft Outlook, dan Kalender iCloud.

   Untuk informasi selengkapnya, lihat Mengimpor acara dari penyedia kalender pihak ketiga.
7. Dalam Jenis kalender, pilih salah satu dari berikut ini.

   - Buka secara default - Kalender terbuka (Tindakan otomatisasi dapat berjalan hingga peristiwa dimulai), lalu ditutup selama peristiwa terkait.

- Buka secara default - Kalender terbuka (Tindakan otomatisasi dapat berjalan hingga peristiwa dimulai), lalu ditutup selama durasi peristiwa terkait.

8. (Opsional) Di Ubah acara manajemen, pilih Tambahkan acara manajemen perubahan ke kalender. Pilihan ini menampilkan semua jendela pemeliharaan terjadwal,State Manager asosiasi, alur kerja Otomasi, dan permintaanChange Manager perubahan di tampilan kalender bulanan Anda.

   > **Tip**
   >
   > Jika nanti Anda ingin menghapus jenis peristiwa ini secara permanen dari tampilan kalender, edit kalender, kosongkan kotak centang ini, lalu pilih Simpan.

9. Pilih Buat kalender.

   Setelah entri kalender dibuat, Systems Manager menampilkan entri kalender Anda dalam Change Calendardaftar. Kolom menunjukkan versi kalender danAkun AWS nomor pemilik pemilik kalender. Entri kalender Anda tidak dapat mencegah atau mengizinkan tindakan apa pun hingga Anda membuat atau mengimpor setidaknya satu peristiwa. Untuk informasi tentang membuat peristiwa, lihatMembuatChange Calendarperistiwa. Untuk informasi tentang mengimpor peristiwa, lihatMengimpor acara dari penyedia kalender pihak ketiga.

Membuat dan mengelola acara diChange Calendar

Setelah membuat kalenderAWS Systems ManagerChange Calendar, Anda dapat membuat, memperbarui, dan menghapus acara yang disertakan dalam kalender terbuka atau tertutup. Change Calendaradalah kemampuanAWS Systems Manager.

> **Tip**
>
> Sebagai alternatif untuk membuat acara langsung di konsol Systems Manager, Anda dapat mengimpor file iCalendar (`.ics`) dari aplikasi kalender pihak ketiga yang didukung. Untuk informasi, lihat Mengimpor dan mengelola acara dari kalender pihak ketiga.

Topik

- MembuatChange Calendarperistiwa
- Memperbarui sebuahChange Calendarperistiwa

- MenghapusChange Calendar acara

MembuatChange Calendarperistiwa

Saat Anda menambahkan acara ke entriChange Calendar, kemampuan dalamAWS Systems Manager, Anda menentukan periode waktu di mana tindakan default entri kalender ditangguhkan. Sebagai contoh, jika jenis entri kalender ditutup secara default, kalender akan terbuka untuk perubahan selama peristiwa. (Atau, Anda dapat membuat acara konsultasi, yang hanya menyajikan peran informasi di kalender.)

Saat ini, Anda hanya dapat membuatChange Calendarperistiwa dengan menggunakan konsol. Acara ditambahkan keChange Calendardokumen yang Anda buat ketika Anda membuatChange Calendarentri.

Untuk membuatChange Calendarperistiwa

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Change Calendar.
3. Dalam daftar kalender, pilih nama entri kalender yang ingin Anda tambahkan peristiwa.
4. Di halaman detail entri kalender, pilih Buat peristiwa.
5. Di halaman Buat peristiwa terjadwal, di Detail peristiwa, masukkan nama tampilan untuk peristiwa Anda. Nama peristiwa dapat berisi huruf, angka, titik, tanda hubung, dan garis bawah. Nama harus cukup spesifik untuk mengidentifikasi tujuan peristiwa. Contohnya adalah **`nighttime-hours`**.
6. UntukDeskripsi, masukkan deskripsi untuk acara Anda. Sebagai contoh, **`The support team isn't available during these hours`**.
7. (Opsional) Jika Anda ingin acara ini berfungsi sebagai pemberitahuan visual atau pengingat saja, pilihPenasehatkotak centang. Acara penasehat tidak memainkan peran fungsional pada kalender Anda. Mereka melayani tujuan informasi hanya bagi mereka yang melihat kalender Anda.
8. UntukTanggal mulai peristiwa, masukkan atau pilih hari dalam format`MM/DD/YYYY`untuk memulai acara, dan masukkan waktu pada hari yang ditentukan dalam format`hh:mm:ss`(jam, menit, dan detik) untuk memulai acara.
9. UntukTanggal akhir peristiwa, masukkan atau pilih hari dalam format`MM/DD/YYYY`untuk mengakhiri acara, dan masukkan waktu pada hari yang ditentukan dalam format`hh:mm:ss`(jam, menit, dan detik) untuk mengakhiri acara.

10. UntukJadwalkan zona waktu, pilih zona waktu yang berlaku untuk waktu awal dan selesai peristiwa. Anda dapat memasukkan sebagian nama kota atau perbedaan zona waktu dari Greenwich Mean Time (GMT) untuk menemukan zona waktu lebih cepat. Default-nya adalah Universal Coordinated Time (UTC).
11. (Opsional) Untuk membuat acara yang berulang tiap hari, minggu, atau bulan, aktifkanKekambuhan, dan kemudian tentukan frekuensi dan tanggal akhir opsional untuk kekambuhan.
12. Pilih Buat peristiwa terjadwal. Peristiwa baru ditambahkan ke entri kalender Anda, dan ditampilkan di tab Peristiwa di halaman detail entri kalender.

Memperbarui sebuahChange Calendarperistiwa

Gunakan prosedur berikut untuk memperbarui prosedur berikut untuk memperbarui prosedur berikutChange Calendarperistiwa diAWS Systems Managerkonsol.Change Calendaradalah kemampuanAWS Systems Manager.

Untuk memperbarui fileChange Calendarperistiwa

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Change Calendar.
3. Dalam daftar kalender, pilih nama entri kalender yang ingin Anda edit peristiwanya.
4. Di halaman detail entri kalender, pilih Peristiwa.
5. Di halaman kalender, pilih peristiwa yang ingin Anda edit.

> **Tip**
>
> Gunakan tombol di kiri atas untuk mundur atau maju satu tahun, atau mundur atau maju satu bulan. Ubah zona waktu, jika perlu, dengan memilih zona waktu yang benar dari daftar di kanan atas.

6. MasukDetail peristiwapilihDiedit.

   Untuk mengubah nama acara dan deskripsi, menambah atau mengganti nilai teks saat ini.
7. Untuk mengubah metodeTanggal mulai peristiwanilai, pilih tanggal mulai saat ini, dan kemudian pilih tanggal baru dari kalender. Untuk mengubah waktu mulai, pilih waktu mulai saat ini, lalu pilih waktu baru dari daftar.

8. Untuk mengubah metodeTanggal akhir peristiwanilai, pilih tanggal saat ini, dan kemudian pilih tanggal akhir baru dari kalender. Untuk mengubah waktu akhir, pilih waktu akhir saat ini, dan kemudian pilih waktu baru dari daftar.
9. Untuk mengubah metodeJadwalkan zona waktunilai, pilih zona waktu untuk waktu awal dan selesai peristiwa. Anda dapat memasukkan sebagian nama kota atau perbedaan zona waktu dari Greenwich Mean Time (GMT) untuk menemukan zona waktu lebih cepat. Default-nya adalah Universal Coordinated Time (UTC).
10. (Opsional) Jika Anda ingin acara ini berfungsi sebagai pemberitahuan visual atau pengingat saja, pilihPenasehatkotak centang. Acara penasehat tidak memainkan peran fungsional pada kalender Anda. Mereka melayani tujuan informasi hanya bagi mereka yang melihat kalender Anda.
11. Pilih Save (Simpan). Perubahan Anda ditampilkan di tab Peristiwa dari halaman detail entri kalender. Pilih peristiwa yang Anda perbarui untuk melihat perubahan Anda.

MenghapusChange Calendar acara

Anda dapat menghapus satu per satu peristiwa diChange Calendar, kemampuan diAWS Systems Manager, dengan menggunakanAWS Management Console.

> **Tip**
>
> Jika Anda memilih Tambahkan acara manajemen perubahan ke kalender saat membuat kalender, Anda dapat melakukan hal berikut:
>
> - Untuk menyembunyikan sementara jenis peristiwa manajemen perubahan dari tampilan kalender, pilih X untuk jenis di bagian atas pratinjau bulanan.
> - Untuk menghapus jenis ini secara permanen dari tampilan kalender, edit kalender, kosongkan kotak centang Tambahkan acara manajemen perubahan ke kalender, lalu pilih Simpan. Menghapus jenis dari tampilan kalender tidak akan menghapusnya dari akun Anda.

Untuk menghapusChange Calendar peristiwa

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Change Calendar.
3. Di daftar kalender, pilih nama entri kalender yang ingin Anda hapus peristiwanya.

4. Di halaman detail entri kalender, pilih Peristiwa.
5. Di halaman kalender, pilih peristiwa yang ingin Anda hapus.

> **Tip**
>
> Gunakan tombol di kiri atas untuk memindahkan kalender mundur atau maju satu tahun, atau mundur atau maju satu bulan. Ubah zona waktu, jika perlu, dengan memilih zona waktu yang benar dari daftar di kanan atas.

6. Di halaman Detail peristiwa, pilih Hapus. Saat Anda diminta untuk mengonfirmasi bahwa Anda ingin menghapus peristiwa, pilih Konfirmasi.

Mengimpor dan mengelola acara dari kalender pihak ketiga

Sebagai alternatif untuk membuat acara langsung diAWS Systems Managerkonsol, Anda dapat mengimpor iCalendar (`.ics`) file dari aplikasi kalender pihak ketiga yang didukung. Kalender Anda dapat menyertakan acara dan acara impor yang Anda buatChange Calendar, yang merupakan kemampuanAWS Systems Manager.

Sebelum Anda memulai

Sebelum Anda mencoba mengimpor file kalender, tinjau persyaratan dan batasan berikut:

Format file kalender

Hanya file iCalendar yang valid (`.ics`) didukung.

Penyedia kalender yang didukung

HANYA`.ics`didukung file yang diekspor dari penyedia kalender pihak ketiga berikut:

- Kalender Google (Petunjuk ekspor)
- Microsoft OutlookPetunjuk ekspor)
- Kalender iCloud (Petunjuk ekspor)

Ukuran file

Anda dapat mengimpor sejumlah valid`.ics`berkas. Namun, ukuran total semua file yang diimpor untuk setiap kalender tidak boleh melebihi 64KB.

> **Tip**
>
> Untuk meminimalkan ukuran `.ics` file, pastikan bahwa Anda hanya mengekspor rincian dasar tentang entri kalender Anda. Jika perlu, kurangi durasi periode waktu yang Anda ekspor.

Zona waktu

Selain nama kalender, penyedia kalender, dan setidaknya satu acara, Anda diekspor `.ics` juga harus menunjukkan zona waktu untuk kalender. Jika tidak, atau ada masalah mengidentifikasi zona waktu, Anda akan diminta untuk menentukan satu setelah Anda mengimpor file.

Batasan acara berulang berulang

Anda diekspor `.ics` file dapat mencakup peristiwa berulang. Namun, jika satu atau lebih kejadian peristiwa berulang telah dihapus di kalender sumber, impor gagal.

Topik

- Mengimpor acara dari penyedia kalender pihak ketiga
- Memperbarui semua acara dari penyedia kalender pihak ketiga
- Menghapus semua acara yang diimpor dari kalender pihak ketiga

Mengimpor acara dari penyedia kalender pihak ketiga

Gunakan prosedur berikut untuk mengimpor file iCalendar (`.ics`) dari aplikasi kalender pihak ketiga yang didukung. Peristiwa yang terkandung dalam file dimasukkan ke dalam aturan untuk kalender terbuka atau tertutup Anda. Anda dapat mengimpor file ke kalender baru yang Anda buat denganChange Calendar (kemampuanAWS Systems Manager) atau ke kalender yang ada.

Setelah Anda mengimpor `.ics` file, Anda dapat menghapus peristiwa individual darinya menggunakanChange Calendar antarmuka. Untuk informasi, lihat MenghapusChange Calendar acara. Anda juga dapat menghapus semua acara dari kalender sumber dengan menghapus `.ics` file. Untuk informasi, lihat Menghapus semua acara yang diimpor dari kalender pihak ketiga.

Mengimpor acara dari penyedia kalender pihak ketiga

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.

2. Di panel navigasi, pilih Change Calendar.
3. Untuk memulai dengan kalender baru, pilih Buat kalender. Di daerah Impor kalender, pilih Pilih file. Untuk informasi tentang langkah-langkah lain untuk membuat kalender baru, lihatMembuat Change Change Change Change Calendar.

   -atau-

   Untuk mengimpor acara pihak ketiga ke kalender yang ada, pilih nama kalender yang ada untuk membukanya.
4. Pilih Tindakan, Edit, dan kemudian di area Impor kalender, pilih Pilih file.
5. Arahkan ke dan pilih `.ics` file yang diekspor di komputer lokal Anda.
6. Jika diminta, untuk Pilih zona waktu, pilih zona waktu yang berlaku untuk kalender.
7. Pilih Simpan.

Memperbarui semua acara dari penyedia kalender pihak ketiga

Jika beberapa peristiwa ditambahkan atau dihapus dari kalender sumber setelah Anda mengimpor `.ics` file iCalendar, Anda dapat mencerminkan perubahan tersebutChange Calendar. Pertama, ekspor ulang kalender sumber, dan kemudian impor file baru keChange Calendar, yang merupakan kemampuanAWS Systems Manager. Acara di kalender perubahan Anda akan diperbarui untuk mencerminkan isi file yang lebih baru.

Untuk memperbarui semua acara dari penyedia kalender pihak ketiga

1. Di kalender pihak ketiga, tambahkan atau hapus acara sesuai keinginan AndaChange Calendar, lalu ekspor ulang kalender ke `.ics` file baru.
2. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
3. Di panel navigasi, pilih Change Calendar.
4. Dari daftar kalender, pilih nama kalender dari daftar.
5. Pilih Pilih file, lalu telusuri ke dan pilih `.ics` file pengganti.
6. Menanggapi pemberitahuan tentang menimpa file yang ada, pilih Konfirmasi.

Menghapus semua acara yang diimpor dari kalender pihak ketiga

Jika Anda tidak lagi menginginkan peristiwa apa pun yang Anda impor dari penyedia pihak ketiga yang disertakan dalam kalender, Anda dapat menghapus `.ics` file iCalendar yang diimpor.

Menghapus semua acara yang diimpor dari kalender pihak ketiga

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Change Calendar.
3. Dari daftar kalender, pilih nama kalender dari daftar.
4. Di daerah Impor kalender, di bawah Kalender impor saya, cari nama kalender yang diimpor, lalu pilih X di kartunya.
5. Pilih Simpan.

Perbarui kalender perubahan

Anda dapat memperbarui deskripsi kalender perubahan, tapi bukan namanya. Meskipun Anda dapat mengubah status default kalender, perlu diketahui bahwa ini membalikkan perilaku tindakan perubahan selama peristiwa yang terkait dengan kalender. Misalnya, jika Anda mengubah status kalender dari Buka secara default menjadi Ditutup secara default, perubahan yang tidak diinginkan mungkin dilakukan selama periode peristiwa saat pengguna yang membuat peristiwa terkait tidak mengharapkan perubahan.

Saat memperbarui kalender perubahan, Anda mengeditChange Calendardokumen yang Anda buat saat membuat entri.Change Calendaradalah kemampuanAWS Systems Manager.

Untuk memperbarui kalender perubahan

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Change Calendar.
3. Di daftar kalender, pilih nama kalender yang ingin Anda perbarui.
4. Di halaman detail kalender, pilihTindakan, Edit.
5. Di Deskripsi, Anda dapat mengubah teks deskripsi. Anda tidak dapat mengedit nama kalender perubahan.
6. Untuk mengubah status kalender, di Jenis kalender, pilih nilai yang berbeda. Perlu diketahui bahwa hal ini membalikkan perilaku tindakan perubahan selama peristiwa yang terkait dengan kalender. Sebelum mengubah jenis kalender, Anda harus memverifikasi dengan yang lainChange Calendarpengguna yang mengubah jenis kalender tidak mengizinkan perubahan yang tidak diinginkan selama peristiwa yang telah dibuat.

- Buka secara default- Kalender terbuka (Tindakan otomatisasi dapat berjalan hingga peristiwa dimulai) lalu ditutup selama peristiwa terkait.
- Ditutup secara default- Kalender terbuka (Tindakan otomatisasi dapat berjalan hingga peristiwa dimulai), lalu ditutup selama durasi peristiwa terkait.

7. Pilih Save (Simpan).

   Kalender Anda tidak dapat mencegah atau mengizinkan tindakan apa pun hingga Anda menambahkan setidaknya satu peristiwa. Untuk informasi tentang cara menambahkan peristiwa, lihat MembuatChange Calendarperistiwa.

Membagikan kalender perubahan

Anda dapat berbagi kalender diChange Calendar, suatu kemampuanAWS Systems Manager, dengan lainnyaAkun AWSdengan menggunakanAWS Systems Managerkonsol. Saat Anda berbagi kalender, kalender akan bersifat hanya-baca untuk pengguna di akun bersama.

Untuk berbagi kalender perubahan

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Change Calendar.
3. Di daftar kalender, pilih nama kalender yang ingin Anda bagikan.
4. Di halaman detail kalender, pilihPembagianTab.
5. MemiilihTindakan, Bagikan.
6. Dalam Bagikan kalender, untuk ID Akun, masukkan nomor ID Akun AWS yang valid, lalu pilih Bagikan.

   Pengguna akun bersama dapat membaca kalender perubahan, tetapi mereka tidak dapat melakukan perubahan.

Menghapus kalender perubahan

Anda dapat menghapus kalender diChange Calendar, suatu kemampuanAWS Systems Manager, dengan menggunakan konsol Systems Manager atauAWS Command Line Interface(AWS CLI). Menghapus kalender perubahan akan menghapus semua peristiwa terkait.

Untuk menghapus kalender perubahan

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Change Calendar.
3. Di daftar kalender, pilih nama kalender yang ingin Anda hapus.
4. Di halaman detail kalender, pilihTindakan, Hapus. Saat Anda diminta untuk mengonfirmasi bahwa Anda ingin menghapus kalender, pilihHapus.

Mendapatkan status perubahan kalender

Anda bisa mendapatkan status keseluruhan kalender atau status kalender pada waktu tertentu diChange Calendar, kemampuan diAWS Systems Manager. Anda juga dapat menunjukkan waktu berikutnya bahwa status kalender berubah dari`OPEN` ke`CLOSED`, atau sebaliknya.

Anda dapat melakukan tugas ini hanya dengan menggunakan operasi API `GetCalendarState`. Prosedur di bagian ini menggunakan AWS Command Line Interface (AWS CLI).

Untuk mendapatkan status perubahan kalender

- Jalankan perintah berikut untuk menampilkan status satu atau beberapa kalender pada waktu tertentu. Parameter `--calendar-names` diperlukan, tetapi `--at-time` adalah opsional. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri.

  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"
  ```

  Berikut adalah contoh.

  ```
  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"
```

Berikut adalah contoh.

```
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"
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
{
    "State": "OPEN",
    "AtTime": "2020-07-30T16:18:18Z",
    "NextTransitionTime": "2020-07-31T00:00:00Z"
}
```

Hasil menunjukkan status kalender (apakah kalender tersebut tipe `DEFAULT_OPEN` atau `DEFAULT_CLOSED`) untuk entri kalender tertentu yang dimiliki oleh atau dibagikan dengan akun Anda, pada waktu yang ditentukan sebagai nilai `--at-time`, dan waktu transisi berikutnya. Jika Anda tidak menambahkan parameter `--at-time`, waktu saat ini digunakan.

> **Note**
>
> Jika Anda menentukan lebih dari satu kalender dalam permintaan, perintah mengembalikan status `OPEN` hanya jika semua kalender dalam permintaan terbuka. Jika satu atau lebih kalender dalam permintaan ditutup, status yang dikembalikan adalah `CLOSED`.

Menambahkan Change Calendar dependensi ke runbook Otomasi

Untuk membuat tindakan Otomasi mematuhiChange Calendar, kemampuanAWS Systems Manager, tambahkan langkah dalam runbook Otomasi yang menggunakan `aws:assertAwsResourceProperty`tindakan. Konfigurasikan tindakan untuk menjalankan `GetCalendarState` guna memverifikasi bahwa entri kalender tertentu sedang dalam status yang Anda inginkan (`OPEN` atau `CLOSED`). Runbook Otomatisasi hanya diperbolehkan untuk melanjutkan ke langkah berikutnya jika status kalender adalah `OPEN`. Berikut ini adalah kutipan sampel berbasis YAML dari runbook Otomatisasi yang tidak dapat melanjutkan ke langkah berikutnya, `LaunchInstance`, kecuali jika status kalender cocok dengan `OPEN`, status yang ditentukan dalam `DesiredValues`.

Berikut adalah contohnya.

```
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
...
```

Pemecahan Masalah Change Calendar

Gunakan informasi berikut untuk membantu memecahkan masalahChange Calendar, suatu kemampuanAWS Systems Manager.

Topik

- Galat 'Impor kalender gagal'

Galat 'Impor kalender gagal'

Masalah: Saat mengimpor iCalendar (`.ics`) file, sistem melaporkan bahwa impor kalender gagal.

- Solusi 1- Pastikan Anda mengimpor file yang diekspor dari penyedia kalender pihak ketiga yang didukung, yang mencakup yang berikut ini:
  - Kalender GOOGLEPetunjuk ekspor)
  - Microsoft Outlook (Petunjuk ekspor)
  - Kalender iCloud (Petunjuk ekspor)
- Solusi 2- Jika kalender sumber Anda berisi peristiwa berulang, pastikan tidak ada kejadian individual dari acara yang dibatalkan atau dihapus. Saat ini,Change Calendartidak mendukung mengimpor peristiwa berulang dengan pembatalan individu. Untuk mengatasi masalah, hapus acara berulang dari kalender sumber, ekspor ulang kalender dan impor kembali keChange Calendar, dan kemudian tambahkan acara berulang menggunakanChange CalendarAntarmuka. Untuk informasi, lihat MembuatChange Calendarperistiwa.
- Solusi 3- Pastikan kalender sumber Anda berisi setidaknya satu acara. Mengunggah`.ics`file yang tidak berisi peristiwa tidak berhasil.
- Solusi 4— Jika sistem melaporkan bahwa impor gagal karena`.ics`terlalu besar, pastikan Anda hanya mengekspor detail dasar tentang entri kalender Anda. Jika perlu, kurangi durasi periode waktu yang Anda ekspor.
- Solusi 5— JikaChange Calendartidak dapat menentukan zona waktu kalender yang diekspor saat Anda mencoba mengimpornya dariPeristiwaTab, Anda mungkin menerima pesan ini: “Impor kalender gagal.Change Calendartidak dapat menemukan zona waktu yang valid. Anda dapat mengimpor kalender dari menu Edit.” Dalam hal ini, pilihTindakan, Edit, dan kemudian mencoba mengimpor file dariMengedit kalenderhalaman.
- Solusi 6- Jangan mengedit`.ics`file sebelum impor. Mencoba untuk memodifikasi isi file dapat merusak data kalender. Jika Anda telah memodifikasi file sebelum mencoba impor, ekspor kalender dari kalender sumber lagi, dan kemudian coba ulang unggahan.

AWS Systems Manager Maintenance Windows

Maintenance WindowsKemampuanAWS Systems Manager, membantu Anda menentukan jadwal kapan harus melakukan tindakan yang berpotensi mengganggu pada node Anda seperti menambal sistem operasi, memperbarui driver, atau menginstal perangkat lunak atau tambalan. DenganMaintenance Windows, Anda dapat menjadwalkan tindakan pada berbagai jenis AWS

sumber daya lainnya, seperti bucket Amazon Simple Storage Service (Amazon S3), Amazon Simple Queue Service (Amazon AWS Key Management Service SQS) antrian, () kunci, dan banyak lagi. AWS KMS Untuk daftar lengkap jenis sumber daya yang didukung yang dapat Anda sertakan dalam target jendela pemeliharaan, lihat Sumber daya yang dapat Anda gunakan AWS Resource Groups dan Editor Tag di Panduan AWS Resource Groups Pengguna. Untuk memulaiMaintenance Windows, buka konsol Systems Manager. Di panel navigasi, pilih Maintenance Windows.

> **Note**
>
> State Managerdan Maintenance Windows dapat melakukan beberapa jenis pembaruan serupa pada node terkelola Anda. Pilihan mana yang Anda pilih bergantung pada apakah Anda perlu mengotomatisasi kepatuhan sistem atau melakukan tugas prioritas tinggi dan sensitif terhadap waktu selama periode yang ditentukan.
> Untuk informasi selengkapnya, lihat Memilih antara State Manager dan Maintenance Windows.

Setiap jendela pemeliharaan memiliki jadwal, durasi maksimum, satu set target terdaftar (node atau AWS sumber daya lain yang ditindaklanjuti), dan serangkaian tugas terdaftar. Anda dapat menambahkan tanda ke windows pemeliharaan pada saat membuatnya. (Tanda adalah kunci yang membantu mengidentifikasi dan mengurutkan sumber daya dalam organisasi Anda.) Anda juga dapat menentukan tanggal agar jendela pemeliharaan tidak dijalankan sebelum atau sesudah, dan Anda dapat menentukan zona waktu internasional untuk mendasarkan jadwal jendela pemeliharaan.

Untuk penjelasan tentang cara berbagai pilihan terkait jadwal untuk windows pemeliharaan berkaitan satu sama lain, lihat Penjadwalan jendela pemeliharaan dan pilihan periode aktif.

Untuk informasi lebih lanjut tentang penggunaan pilihan `--schedule`, lihat Referensi: Ekspresi cron dan rate untuk Systems Manager.

Jenis data yang didukung

Windows pemeliharaan dapat menjalankan empat jenis tugas:

- Perintah diRun Command, kemampuan Systems Manager

  Untuk informasi selengkapnya tentang Run Command, lihat AWS Systems Manager Run Command.
- Alur kerja di Otomatisasi, sebuah kemampuan Systems Manager

Untuk informasi lebih lanjut tentang alur kerja Otomatisasi, lihat AWS Systems ManagerOtomasi.

- Fungsi di AWS Lambda

  Untuk informasi selengkapnya tentang fungsi Lambda, lihat Memulai Lambda di Panduan Pengembang. AWS Lambda

- Tugas di AWS Step Functions

  Untuk informasi lebih lanjut tentang Step Functions, lihat Panduan Developer AWS Step Functions.

> **Note**
>
> Satu atau lebih target harus ditentukan untuk tugas Run Command tipe jendela pemeliharaan. Tergantung dari tugas, target bersifat opsional untuk jenis tugas jendela pemeliharaan lainnya (Otomatisasi, AWS Lambda, dan AWS Step Functions). Untuk informasi lebih lanjut tentang menjalankan tugas yang tidak menentukan target, lihat Pendaftaran tugas jendela pemeliharaan tanpa target.

Artinya Anda dapat menggunakan windows pemeliharaan untuk melakukan tugas seperti berikut pada target yang Anda pilih.

- Menginstal atau memperbarui aplikasi.
- Menerapkan patch.
- Instal atau perbaruiSSM Agent.
- Jalankan PowerShell perintah dan skrip shell Linux dengan menggunakan Run Command tugas Systems Manager.
- Build Amazon Machine Images (AMIs), boot-strap software, dan konfigurasi node dengan menggunakan tugas Systems Manager Automation.
- Jalankan AWS Lambda fungsi yang memanggil tindakan tambahan, seperti memindai node Anda untuk pembaruan tambalan.
- Jalankan AWS Step Functions state machine untuk melakukan tugas-tugas seperti menghapus node dari lingkungan Elastic Load Balancing, menambal node, dan kemudian menambahkan node kembali ke lingkungan Elastic Load Balancing.
- Target node yang offline dengan menentukan grup AWS sumber daya sebagai target.

EventBridge dukungan

Kemampuan Systems Manager ini didukung sebagai jenis peristiwa dalam EventBridge aturan Amazon. Untuk informasi selengkapnya, lihat Pemantauan peristiwa Systems Manager dengan Amazon EventBridge dan Referensi: Pola dan jenis EventBridge acara Amazon untuk Systems Manager.

Konten

Menyiapkan Maintenance Windows

Sebelum pengguna di AndaAkun AWS dapat membuat dan menjadwalkan tugas jendela pemeliharaan menggunakanMaintenance WindowsAWS Systems Manager, sebuah kemampuan, pemberian izin yang diperlukan harus dilakukan.

Sebelum Anda memulai

Untuk menyelesaikan tugas di bagian ini, Anda memerlukan salah satu atau kedua sumber berikut sudah diatur:

- Izin yang ditetapkan ke entitas IAM (seperti pengguna atau grup). Pengguna atau grup ini seharusnya sudah memiliki izin umum untuk menggunakan windows pemeliharaan. Lakukan hal ini dengan menetapkan kebijakan IAM`AmazonSSMFullAccess` ke pengguna atau grup, atau kebijakan IAM lain yang menyediakan himpunan izin akses yang lebih kecil untuk Systems Manager yang mencakup tugas jendela pemeliharaan.
- (Opsional) Untuk windows pemeliharaan yang menjalankanRun Command tugas, Anda dapat memilih untuk mengirim notifikasi status Amazon Simple Notification Service (Amazon SNS).

Run Commandadalah suatu kemampuan Systems Manager. Jika Anda ingin menggunakan opsi ini, konfigurasikan topik Amazon SNS sebelum menyelesaikan tugas penyiapan ini. Untuk informasi tentang pengonfigurasian notifikasi Amazon SNS untuk Systems Manager, termasuk informasi tentang pembuatan IAM role untuk digunakan sebagai pengiriman notifikasi SNS, lihat Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS.

GamUmum Pengaturan

Untuk memberikan izin yang dibutuhkan pengguna untuk mendaftarkan jendela pemeliharaan, administrator melakukan tugas-tugas berikut. Petunjuk lengkap disediakan dalamGunakan konsol untuk mengonfigurasi izin untuk jendela pemeliharaan).

Tugas 1: Membuat kebijakan untuk digunakan dengan peran jendela pemeliharaan khusus

Tugas jendela pemeliharaan memerlukan peran IAM untuk memberikan izin yang diperlukan untuk berjalan pada sumber daya target. Jenis tugas yang Anda jalankan dan persyaratan operasional lainnya menentukan isi kebijakan ini.

Kami menyediakan kebijakan dasar yang dapat Anda adaptasi dalam topikTugas 1: Membuat kebijakan untuk peran layanan jendela pemeliharaan kustom Anda.

Tugas 2: Membuat peran layanan kustom untuk tugas jendela pemeliharaan

Kebijakan yang Anda buat di Tugas 1 dilampirkan ke peran jendela pemeliharaan yang Anda buat di Tugas 2. Saat pengguna mendaftarkan tugas jendela pemeliharaan, mereka menentukan peran layanan kustom ini sebagai bagian dari konfigurasi tugas. Izin dalam peran ini memungkinkan Systems Manager untuk menjalankan tugas di windows pemeliharaan atas nama Anda.

> **Important**
>
> Sebelumnya, konsol Systems Manager memberi Anda kemampuan untuk memilih peran terkait layanan IAM yangAWS dikelolaAWSServiceRoleForAmazonSSM untuk digunakan sebagai peran pemeliharaan untuk tugas Anda. Menggunakan peran ini dan kebijakan terkait,AmazonSSMServiceRolePolicy, untuk pemeliharaan jendela tugas tidak lagi dianjurkan. Jika Anda menggunakan peran ini untuk tugas jendela pemeliharaan sekarang, kami mendorong Anda untuk berhenti menggunakannya. Sebagai gantinya, buat peran IAM Anda sendiri yang memungkinkan komunikasi antara Systems Manager dan lainnyaLayanan AWS saat tugas jendela pemeliharaan Anda berjalan.

Tugas 3: Berikan izin untuk menggunakan peran layanan kepada pengguna yang mendaftarkan tugas jendela pemeliharaan

Menyediakan pengguna dengan izin untuk mengakses peran jendela pemeliharaan kustom memungkinkan mereka menggunakannya dengan tugas-tugas pemeliharaan jendela mereka. Ini adalah tambahan untuk izin yang telah Anda berikan kepada mereka untuk bekerja dengan perintah Systems Manager APIMaintenance Windows. Peran ini menyampaikan izin perlu menjalankan tugas pemeliharaan jendela. Akibatnya, pengguna tidak dapat menetapkan tugas ke jendela pemeliharaan menggunakan peran layanan kustom Anda tanpa kemampuan untuk meneruskan izin IAM ini.

Tugas 4: (Opsional) Tolak izin untuk pengguna yang tidak diizinkan untuk mendaftarkan tugas jendela pemeliharaan

Anda dapat menolak`ssm:RegisterTaskWithMaintenanceWindow` izin untuk pengguna di Anda jika AndaAkun AWS tidak ingin ia mendaftarkan tugas dengan windows pemeliharaan. Ini memberikan lapisan pencegahan tambahan bagi pengguna yang tidak boleh mendaftarkan tugas jendela pemeliharaan.

Topik

- Gunakan konsol untuk mengonfigurasi izin untuk jendela pemeliharaan

Gunakan konsol untuk mengonfigurasi izin untuk jendela pemeliharaan

Prosedur berikut menjelaskan cara menggunakan konsol AWS Systems Manager untuk membuat peran dan izin yang diperlukan untuk windows pemeliharaan.

Topik

- Tugas 1: Membuat kebijakan untuk peran layanan jendela pemeliharaan kustom Anda
- Tugas 2: Buat peran layanan kustom untuk windows pemeliharaan (konsol)
- Tugas 3: Mengonfigurasi izin untuk pengguna yang diizinkan untuk mendaftarkan tugas jendela pemeliharaan (konsol)
- Tugas 4: Mengonfigurasi izin untuk pengguna yang tidak diizinkan untuk mendaftarkan tugas jendela pemeliharaan

Tugas 1: Membuat kebijakan untuk peran layanan jendela pemeliharaan kustom Anda

Anda dapat menggunakan kebijakan berikut dalam format JSON untuk membuat kebijakan yang akan digunakan dengan peran jendela pemeliharaan Anda. Anda kemudian perlu melampirkan kebijakan ini pada peran yang Anda buat nantiTugas 2: Buat peran layanan kustom untuk windows pemeliharaan (konsol).

> **Important**
>
> Bergantung pada tugas dan jenis tugas yang dijalankan jendela pemeliharaan Anda, Anda mungkin tidak memerlukan semua izin dalam kebijakan ini, dan Anda mungkin perlu menyertakan izin tambahan.

Membuat kebijakan untuk peran layanan jendela pemeliharaan kustom Anda

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Pada panel navigasi, pilih Kebijakan, lalu pilih Buat kebijakan.
3. Pilih tab JSON.
4. Ganti isi default dengan yang berikut ini:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:SendCommand",
                "ssm:CancelCommand",
                "ssm:ListCommands",
                "ssm:ListCommandInvocations",
                "ssm:GetCommandInvocation",
                "ssm:GetAutomationExecution",
                "ssm:StartAutomationExecution",
                "ssm:ListTagsForResource",
                "ssm:GetParameters"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
```

```
            "Action": [
                "states:DescribeExecution",
                "states:StartExecution"
            ],
            "Resource": [
                "arn:aws:states:*:*:execution:*:*",
                "arn:aws:states:*:*:stateMachine:*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "lambda:InvokeFunction"
            ],
            "Resource": [
                "arn:aws:lambda:*:*:function:*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "resource-groups:ListGroups",
                "resource-groups:ListGroupResources"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "tag:GetResources"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
```

```
                        "ssm.amazonaws.com"
                    ]
                }
            }
        }
    ]
}
```

5. Ubah konten JSON sesuai kebutuhan untuk tugas pemeliharaan yang Anda jalankan di akun Anda. Perubahan yang Anda buat khusus untuk operasi yang Anda rencanakan.

   Misalnya:

   - Anda dapat menyediakan Amazon Resource Names (ARN) untuk fungsi dan mesin status tertentu alih-alih menggunakan qualifier wildcard (*).
   - Jika Anda tidak berencana untuk menjalankanAWS Step Functions tugas, Anda dapat menghapus`states` izin dan (ARN).
   - Jika Anda tidak berencana untuk menjalankanAWS Lambda tugas, Anda dapat menghapus`lambda` izin dan ARN.
   - Jika Anda tidak berencana untuk menjalankan tugas Otomasi, Anda dapat menghapus`ssm:GetAutomationExecution` dan`ssm:StartAutomationExecution` izin.
   - Tambahkan izin tambahan yang mungkin diperlukan agar tugas dapat dijalankan. Misalnya, beberapa tindakan Otomatisasi menggunakan tumpukan AWS CloudFormation. Oleh karena itu, izin `cloudformation:CreateStack`, `cloudformation:DescribeStacks`, dan `cloudformation:DeleteStack` diperlukan.

     Untuk contoh lain, runbook Otomatisasi`AWS-CopySnapshot` memerlukan izin untuk membuat snapshot Amazon Elastic Block Store (Amazon EBS). Oleh karena itu, peran layanan membutuhkan izin`ec2:CreateSnapshot`.

     Untuk informasi tentang izin peran yang diperlukan oleh runbook Otomatisasi, lihat deskripsi runbook di referensi runbookAWS Systems Manager Otomatisasi.

6. Setelah menyelesaikan revisi kebijakan, pilih Berikutnya: Tag.
7. (Opsional) Tambahkan satu atau beberapa pasangan nilai kunci tag untuk menata, melacak, atau mengontrol akses untuk kebijakan ini, lalu pilih Berikutnya: Tinjau.
8. Untuk Nama, masukkan nama yang mengidentifikasi ini sebagai kebijakan yang digunakan peranMaintenance Windows layanan yang Anda buat. Misalnya: **`my-maintenance-window-role-policy`**.

9. Pilih Buat kebijakan, dan catat nama yang Anda tentukan untuk kebijakan tersebut. Anda kemudian perlu merujuknya di prosedur berikutnyaTugas 2: Buat peran layanan kustom untuk windows pemeliharaan (konsol).

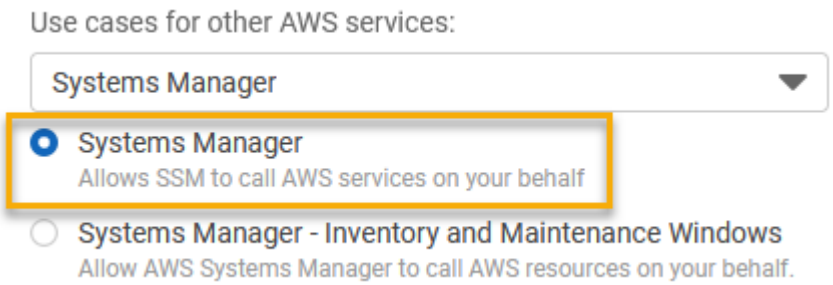
Tugas 2: Buat peran layanan kustom untuk windows pemeliharaan (konsol)

Gunakan prosedur berikut untuk membuat peran layanan kustom untukMaintenance Windows, agar Systems Manager dapat menjalankanMaintenance Windows tugas atas nama Anda. Anda akan melampirkan kebijakan yang Anda buat di tugas sebelumnya ke peran layanan kustom yang Anda buat.

> **Important**
>
> Sebelumnya, konsol Systems Manager memberi Anda kemampuan untuk memilih peran terkait layanan IAM yangAWS dikelolaAWSServiceRoleForAmazonSSM untuk digunakan sebagai peran pemeliharaan untuk tugas Anda. Menggunakan peran ini dan kebijakan terkait,AmazonSSMServiceRolePolicy, untuk pemeliharaan jendela tugas tidak lagi dianjurkan. Jika Anda menggunakan peran ini untuk tugas jendela pemeliharaan sekarang, kami mendorong Anda untuk berhenti menggunakannya. Sebagai gantinya, buat peran IAM Anda sendiri yang memungkinkan komunikasi antara Systems Manager dan lainnyaLayanan AWS saat tugas jendela pemeliharaan Anda berjalan.

Untuk membuat peran layanan kustom (konsol)

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran, lalu pilih Buat peran.
3. Untuk Pilih entitas tepercaya, buat pilihan berikut:
    1. Untuk tipe entitas tepercaya, pilih AWSlayanan
    2. Untuk Kasus penggunaan untukAWS layanan lain, pilih Systems Manager
    3. Pilih Systems Manager, seperti yang ditunjukkan pada gambar berikut.



4. Pilih Selanjutnya.
5. Di kotak pencarian, masukkan nama kebijakan yang Anda buatTugas 1: Membuat kebijakan untuk peran layanan jendela pemeliharaan kustom Anda, centang kotak di samping namanya, lalu pilih Berikutnya.
6. Untuk Nama peran, masukkan nama yang mengidentifikasi peran ini sebagaiMaintenance Windows peran. Misalnya: **`my-maintenance-window-role`**.
7. (Opsional) Ubah deskripsi peran default untuk mencerminkan tujuan dari peran ini. Sebagai contoh: **`Performs maintenance window tasks on your behalf`**.
8. (Opsional) Tambahkan satu atau beberapa pasangan nilai kunci tag untuk mengatur, melacak, atau mengontrol akses untuk peran ini, lalu pilih Berikutnya: Tinjau.
9. Pilih Buat peran. Sistem mengembalikan Anda ke Peran yang baru.
10. Pilih nama dari peran yang baru saja Anda buat.
11. Pilih tab Hubungan kepercayaan, lalu verifikasi bahwa kebijakan berikut ditampilkan di kotak Entitas tepercaya.

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

12. Salin atau catat nama peran dan nilai ARN di area Ringkasan. Pengguna di akun Anda menentukan informasi ini saat mereka membuat jendela pemeliharaan.

Tugas 3: Mengonfigurasi izin untuk pengguna yang diizinkan untuk mendaftarkan tugas jendela pemeliharaan (konsol)

Ketika Anda mendaftarkan tugas dengan jendela pemeliharaan, Anda menentukan peran layanan kustom atau peran terkait layanan Systems Manager untuk menjalankan operasi tugas yang sebenarnya. Ini merupakan peran yang diasumsikan layanan saat menjalankan tugas atas nama Anda. Sebelum itu, untuk mendaftarkan tugas itu sendiri, tetapkan PassRole kebijakan IAM ke entitas IAM (seperti pengguna atau grup). Hal ini memungkinkan entitas IAM (pengguna atau grup) untuk menentukan, sebagai bagian dari pendaftaran tugas tersebut dengan jendela pemeliharaan, peran yang harus digunakan saat menjalankan tugas. Untuk informasi, lihat Memberikan izin pengguna untuk meneruskan peranLayanan AWS ke Panduan Pengguna IAM.

Untuk mengonfigurasi izin untuk pengguna yang diizinkan untuk mendaftarkan tugas jendela pemeliharaan

Jika entitas IAM (pengguna, peran, atau grup) diatur dengan izin administrator, maka pengguna atau peran memiliki akses ke Windows Pemeliharaan. Untuk entitas IAM tanpa izin administrator, administrator harus memberikan izin berikut kepada entitas IAM. Ini adalah izin minimum yang diperlukan untuk mendaftarkan tugas dengan jendela pemeliharaan:

- Kebijakan yang`AmazonSSMFullAccess` dikelola, atau kebijakan yang memberikan izin yang sebanding.
- Berikut`iam:PassRole` dan`iam:ListRoles` izin.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::account-id:role/my-maintenance-window-role"
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::account-id:role/"
        },
```

```
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::account-id:role/aws-service-role/
ssm.amazonaws.com/"
        }
    ]
}
```

*my-maintenance-window-role*mewakili nama peran jendela pemeliharaan kustom yang Anda buat sebelumnya.

account-id mewakili ID AndaAkun AWS. Penambahan izin ini untuk `arn:aws:iam::`*account-id*`:role/` sumber daya memungkinkan pengguna untuk melihat dan memilih dari peran pelanggan di konsol saat membuat tugas jendela pemeliharaan. Penambahan izin ini untuk `arn:aws:iam::`*account-id*`:role/aws-service-role/ssm.amazonaws.com/` memungkinkan pengguna untuk memilih peran terkait layanan Systems Manager di konsol saat membuat tugas jendela pemeliharaan.

Untuk menyediakan akses, tambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup diAWS IAM Identity Center:

  Buat set izin. Ikuti petunjuk di Buat set izin di PanduanAWS IAM Identity Center Pengguna.

- Pengguna yang dikelola dalam IAM melalui penyedia identitas:

  Buat peran untuk federasi identitas Ikuti petunjuk dalam Membuat peran untuk penyedia identitas pihak ketiga (federasi) di Panduan Pengguna IAM.

- Pengguna IAM:
  - Buat peran yang dapat diasumsikan pengguna Anda. Ikuti petunjuk dalam Membuat peran untuk pengguna IAM di Panduan Pengguna IAM.
  - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk dalam Menambahkan izin ke pengguna (konsol) di Panduan Pengguna IAM.

Untuk mengonfigurasi izin untuk grup yang diizinkan untuk mendaftarkan tugas jendela pemeliharaan (konsol)

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.

2. Pada panel navigasi, pilih User groups (Grup pengguna).
3. Di daftar grup, pilih nama grup di mana Anda ingin menetapkan izin `iam:PassRole`.
4. Pada tab Izin, pilih Tambahkan izin, Buat kebijakan sebaris, lalu pilih tab JSON.
5. Ganti isi default kotak dengan berikut.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::account-id:role/my-maintenance-window-role"
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::account-id:role/"
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::account-id:role/aws-service-role/
ssm.amazonaws.com/"
        }
    ]
}
```

*my-maintenance-window-role*mewakili nama peran jendela pemeliharaan kustom yang Anda buat sebelumnya.

account-id mewakili ID AndaAkun AWS. Penambahan izin ini untuk `arn:aws:iam::`*account-id*`:role/` sumber daya memungkinkan pengguna untuk melihat dan memilih dari peran pelanggan di konsol saat membuat tugas jendela pemeliharaan. Penambahan izin ini untuk `arn:aws:iam::`*account-id*`:role/aws-service-role/ssm.amazonaws.com/` memungkinkan pengguna untuk memilih peran terkait layanan Systems Manager di konsol saat membuat tugas jendela pemeliharaan.

6. Pilih Tinjau kebijakan.
7. Pada halaman Tinjau kebijakan, masukkan nama dalam kotak Nama untuk mengidentifikasi kebijakan `PassRole` ini, seperti **`my-group-iam-passrole-policy`**, lalu pilih Buat kebijakan.

Tugas 4: Mengonfigurasi izin untuk pengguna yang tidak diizinkan untuk mendaftarkan tugas jendela pemeliharaan

Tergantung dari apakah Anda menolak izin `ssm:RegisterTaskWithMaintenanceWindow` untuk pengguna individu atau grup, gunakan salah satu dari prosedur berikut ini untuk mencegah pengguna dari mendaftarkan tugas dengan jendela pemeliharaan.

Untuk mengonfigurasi izin untuk pengguna yang tidak diizinkan untuk mendaftarkan tugas jendela pemeliharaan

- Administrator harus menambahkan pembatasan berikut ke entitas IAM.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "ssm:RegisterTaskWithMaintenanceWindow",
            "Resource": "*"
        }
    ]
}
```

Untuk mengonfigurasi izin untuk grup yang tidak diizinkan untuk mendaftarkan tugas jendela pemeliharaan (konsol)

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Pada panel navigasi, pilih User groups (Grup pengguna).
3. Di daftar grup, pilih nama grup di mana Anda ingin menolak izin `ssm:RegisterTaskWithMaintenanceWindow`.
4. Pada tab Izin, pilih Tambahkan izin, Buat kebijakan sebaris.
5. Pilih tab JSON, dan kemudian ganti isi default kotak dengan yang berikut ini.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "ssm:RegisterTaskWithMaintenanceWindow",
```

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

6. Pilih Tinjau kebijakan.
7. Pada halaman Kebijakan tinjauan, untuk Nama, masukkan nama untuk mengidentifikasi kebijakan ini, misalnya**`my-groups-deny-mw-tasks-policy`**, lalu pilih Buat kebijakan.

Menggunakan windows pemeliharaan (konsol)

Bagian ini menjelaskan cara membuat, mengonfigurasi, memperbarui, dan menghapus windows pemeliharaan menggunakan konsol AWS Systems Manager. Bagian ini juga menyediakan informasi tentang pengelolaan target dan tugas jendela pemeliharaan.

> **Important**
>
> Kami merekomendasikan agar Anda terlebih dahulu membuat dan mengonfigurasi windows pemeliharaan di lingkungan pengujian.

Sebelum Anda memulai

Sebelum Anda membuat jendela pemeliharaan, Anda harus mengonfigurasi akses keMaintenance Windows, sebuah kemampuanAWS Systems Manager. Untuk informasi selengkapnya, lihat Menyiapkan Maintenance Windows.

Topik

- Membuat jendela pemeliharaan (konsol)
- Menetapkan target ke jendela pemeliharaan (konsol)
- Menetapkan tugas ke jendela pemeliharaan (konsol)
- Menonaktifkan atau mengaktifkan jendela pemeliharaan
- Memperbarui atau menghapus sumber daya jendela pemeliharaan (konsol)

Membuat jendela pemeliharaan (konsol)

Di prosedur ini, Anda membuat jendela pemeliharaan di prosedur pemeliharaan di prosedur pemeliharaan di prosedur iniMaintenance Windows, suatu kemampuanAWS Systems Manager. Anda dapat menentukan pilihan dasarnya, seperti nama, jadwal, dan durasi. Di langkah berikutnya, Anda memilih target, atau sumber daya, yang diperbarui olehnya dan tugas yang dijalankan saat jendela pemeliharaan berjalan.

> **Note**
>
> Untuk penjelasan tentang cara berbagai pilihan terkait jadwal untuk windows pemeliharaan berkaitan satu sama lain, lihat Penjadwalan jendela pemeliharaan dan pilihan periode aktif. Untuk informasi lebih lanjut tentang penggunaan pilihan `--schedule`, lihat Referensi: Ekspresi cron dan rate untuk Systems Manager.

Untuk membuat jendela pemeliharaan (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Maintenance Windows.
3. Pilih Buat jendela pemeliharaan.
4. UntukNama, masukkan nama deskriptif untuk membantu Anda mengidentifikasi jendela pemeliharaan ini.
5. (Opsional) UntukDeskripsi, masukkan deskripsi untuk mengidentifikasi bagaimana jendela pemeliharaan ini akan digunakan.
6. (Opsional) Jika Anda ingin mengizinkan tugas jendela pemeliharaan berjalan pada node terkelola, meski Anda belum mendaftarkan node tersebut sebagai target, pilihIzinkan target yang tidak terdaftar.

   Jika Anda memilih pilihan ini, maka Anda dapat memilih node yang tidak terdaftar (menurut ID node) saat mendaftarkan tugas dengan jendela pemeliharaan.

   Jika Anda tidak memilih pilihan ini, maka Anda harus memilih target yang terdaftar sebelumnya saat mendaftarkan tugas dengan jendela pemeliharaan.
7. Tentukan jadwal untuk jendela pemeliharaan dengan menggunakan salah satu dari tiga pilihan penjadwalan.

Untuk informasi tentang pembangunan ekspresi cron/rate, lihat Referensi: Ekspresi cron dan rate untuk Systems Manager.

8. Untuk Durasi, masukkan jumlah jam yang akan dijalankan oleh jendela pemeliharaan. Nilai yang Anda tentukan menentukan waktu selesai tertentu untuk jendela pemeliharaan berdasarkan waktu dimulainya. Tidak ada tugas jendela pemeliharaan yang diizinkan untuk memulai setelah waktu selesai dikurangi jumlah jam yang Anda tentukan untuk Berhenti memulai tugas di langkah berikutnya.

   Sebagai contoh, jika jendela pemeliharaan dimulai pada pukul 3 sore, durasinya tiga jam, dan nilai Berhenti memulai tugas adalah satu jam, tidak ada tugas jendela pemeliharaan yang dapat memulai setelah pukul 5 sore.

9. Untuk Berhenti memulai tugas, masukkan jumlah jam sebelum akhir jendela pemeliharaan dimana sistem harus berhenti menjadwalkan tugas baru untuk dijalankan.

10. (Opsional) UntukTanggal mulai jendela, tentukan tanggal dan waktu, dalam format ISO-8601 Extended, jika Anda ingin menjadikan jendela pemeliharaan aktif. Ini memungkinkan Anda menunda aktivasi jendela pemeliharaan hingga tanggal yang ditentukan di masa depan.

11. (Opsional) UntukTanggal akhir jendela, tentukan tanggal dan waktu, dalam format ISO-8601 Extended, jika Anda ingin menjadikan jendela pemeliharaan tidak aktif. Hal ini memungkinkan Anda untuk mengatur tanggal dan waktu di masa depan setelah jendela pemeliharaan tidak lagi berjalan.

12. (Opsional) UntukZona waktu, tentukan zona waktu untuk digunakan sebagai dasar ketika jendela pemeliharaan terjadwal berjalan, dalam format Internet Assigned Numbers Authority (IANA). Misalnya: “Amerika/Los_Angeles”, “etc/UTC”, atau “Asia/Seoul”.

    Untuk informasi lebih lanjut tentang format yang valid, lihat Basis Data Zona Waktu di situs web IANA.

13. (Opsional) UntukOffset Jadwal, masukkan jumlah hari untuk menunggu setelah tanggal dan waktu yang ditentukan oleh ekspresi cron atau rate sebelum menjalankan jendela pemeliharaan. Anda dapat menentukan antara satu sampai enam hari.

> **Note**
>
> Opsi ini hanya tersedia jika Anda menentukan jadwal dengan memasukkan ekspresi cron atau tingkat secara manual.

14. (Opsional) Di area Kelola tanda, terapkan satu atau beberapa pasangan nama/nilai kunci tanda ke jendela pemeliharaan.

    Tanda adalah metadata opsional yang Anda tetapkan ke sumber daya. Tag memungkinkan Anda untuk mengkategorikan sumber daya dengan berbagai cara, seperti berdasarkan tujuan, pemilik, atau lingkungan. Misalnya, Anda mungkin ingin menandai jendela pemeliharaan untuk mengidentifikasi jenis tugas yang dijalankannya, jenis target, dan lingkungan tempat ia berjalan. Dalam hal ini, Anda dapat menentukan pasangan nama/nilai kunci berikut:

    - `Key=TaskType,Value=AgentUpdate`
    - `Key=OS,Value=Windows`
    - `Key=Environment,Value=Production`
15. Pilih Buat jendela pemeliharaan. Sistem mengembalikan Anda ke halaman jendela pemeliharaan. Tahapan jendela pemeliharaan yang baru saja Anda buat adalahDiaktifkan.

Menetapkan target ke jendela pemeliharaan (konsol)

Di prosedur ini, Anda mendaftarkan target dengan jendela pemeliharaan. Dengan kata lain, Anda menentukan sumber daya yang ditindak oleh jendela pemeliharaan.

> **Note**
>
> Jika satu tugas jendela pemeliharaan terdaftar dengan beberapa target, permintaan tugasnya terjadi secara berurutan dan tidak secara paralel. Jika tugas Anda harus berjalan pada beberapa target di waktu yang sama, daftarkan tugas untuk setiap target secara individual dan tetapkan pada setiap tugas tingkat prioritas yang sama.

Untuk menetapkan target ke jendela pemeliharaan (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Maintenance Windows.
3. Di daftar windows pemeliharaan, pilih jendela pemeliharaan untuk menambahkan target.
4. Pilih Tindakan, lalu pilih Daftarkan target.
5. (Opsional) Untuk Nama target, masukkan nama untuk target.
6. (Opsional) Untuk Deskripsi, masukkan deskripsi.

7. (Opsional) Untuk informasi Pemilik, tentukan informasi yang akan disertakan dalam EventBridge peristiwa Amazon apa pun yang muncul saat menjalankan tugas untuk target ini di jendela pemeliharaan ini.

   Untuk informasi tentang penggunaan EventBridge untuk memantau peristiwa Systems Manager, lihatPemantauan peristiwa Systems Manager dengan Amazon EventBridge.

8. Di area Target, pilih salah satu pilihan yang dijelaskan di tabel berikut.

Opsi	Deskripsi
Tentukan tag contoh	Untuk kotak Tentukan tag instance, tentukan satu atau beberapa kunci tag dan nilai (opsional) yang telah atau akan ditambahk an ke node terkelola di akun Anda. Ketika jendela pemeliharaan berjalan, ia mencoba untuk melakukan tugas pada semua node terkelola yang tag ini telah ditambahkan. Jika Anda menentukan lebih dari satu kunci tag, sebuah node harus ditandai dengan semua kunci tag dan nilai yang Anda tentukan untuk dimasukkan dalam grup target.
Pilih instans secara manual	Dari daftar, pilih kotak untuk setiap node yang ingin Anda sertakan dalam target jendela pemeliharaan. Daftar ini mencakup semua node di akun Anda yang dikonfigurasi untuk digunakan dengan Systems Manager. Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

Opsi	Deskripsi
	Untuk perangkat edge dan server lokal dan mesin virtual (VM), lihat Menyiapkan Manajer Sistem untuk lingkungan hybrid dan multicloud

Opsi	Deskripsi
Pilih grup sumber daya	Untuk Grup sumber daya, pilih nama grup sumber daya yang ada di akun Anda dari daftar. Untuk informasi tentang pembuatan dan penggunaan grup sumber daya, lihat topik berikut: • Apa itu kelompok sumber daya? di Panduan AWS Resource Groups Pengguna • Resource Groups dan Penandaan untukAWS di Blog Berita AWS (Opsional) Untuk jenis Sumber Daya, pilih hingga lima jenis sumber daya yang tersedia, atau pilih Semua jenis sumber daya. Jika tugas yang Anda tetapkan ke jendela pemeliharaan tidak berfungsi pada salah satu jenis sumber daya yang ditambahkan ke target, sistem mungkin akan melaporkan kesalahan. Tugas di mana jenis sumber daya yang didukung ditemukan akan terus berjalan meskipun ada kesalahan ini. Misalnya, anggap saja Anda menambahkan jenis sumber daya berikut ke target ini: • `AWS::S3::Bucket` • `AWS::DynamoDB::Table` • `AWS::EC2::Instance` Namun kemudian, ketika Anda menambahk an tugas ke jendela pemeliharaan, Anda

Opsi	Deskripsi
	hanya menyertakan tugas yang melakukan tindakan pada node, seperti menerapka n baseline patch atau me-reboot node. Di log jendela pemeliharaan, kesalahan mungkin dilaporkan karena tidak ada bucket Amazon Simple Storage Service (Amazon S3) atau tabel Amazon DynamoDB yang ditemukan. Namun, jendela pemeliharaan masih menjalankan tugas pada node di grup sumber daya Anda.

9. Pilih Daftarkan target.

Jika Anda ingin menetapkan lebih banyak target untuk jendela pemeliharaan ini, pilih tab Target, dan kemudian pilih Daftarkan target. Dengan pilihan ini, Anda dapat memilih berbagai cara penargetan. Misalnya, jika sebelumnya Anda menargetkan node berdasarkan ID node, Anda dapat mendaftarkan target baru dan node target dengan menentukan tag yang diterapkan pada node terkelola atau memilih jenis sumber daya dari grup sumber daya.

Menetapkan tugas ke jendela pemeliharaan (konsol)

Di prosedur ini, Anda menambahkan tugas ke jendela pemeliharaan. Tugas adalah tindakan yang dilakukan saat jendela pemeliharaan berjalan.

Empat jenis tugas berikut dapat ditambahkan ke jendela pemeliharaan:

- AWS Systems ManagerRun Commandperintah
- Alur kerja Otomatisasi Systems Manager
- Tugas AWS Step Functions
- Fungsi AWS Lambda

> **Important**
>
> Kebijakan IAM untuk Maintenance Windows mengharuskan Anda menambahkan awalan ke nama fungsi `SSM` Lambda (atau alias). Sebelum melanjutkan untuk mendaftarkan jenis

tugas ini, perbarui namanya di AWS Lambda untuk menyertakan `SSM`. Misalnya, jika nama fungsi Lambda Anda adalah `MyLambdaFunction`, ubah ke `SSMMyLambdaFunction`.

Untuk menetapkan tugas ke jendela pemeliharaan

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Maintenance Windows.
3. Di daftar windows pemeliharaan, pilih jendela pemeliharaan.
4. Pilih Tindakan, lalu pilih pilihan untuk jenis tugas yang ingin Anda daftarkan dengan jendela pemeliharaan.
   - Daftar Jalankan tugas perintah
   - Daftarkan tugas Otomasi
   - Daftarkan tugas Lambda
   - Daftarkan tugas Step Functions
5. (Opsional) Untuk Nama, masukkan nama untuk tugas tersebut.
6. (Opsional) Untuk Deskripsi, masukkan deskripsi.
7. Untuk cutoff pemanggilan tugas baru, jika Anda tidak ingin pemanggilan tugas baru dimulai setelah batas waktu jendela pemeliharaan tercapai, pilih Diaktifkan.

   Ketika opsi ini tidak diaktifkan, tugas terus berjalan ketika batas waktu tercapai dan memulai pemanggilan tugas baru hingga selesai.

   > **Note**
   >
   > Status untuk tugas yang tidak selesai saat Anda mengaktifkan opsi ini adalah`TIMED_OUT`.

8. Untuk langkah ini, ikuti sub-langkah untuk jenis tugas yang Anda pilih.

   Run Command

   1. Dalam daftar dokumen Command, pilih dokumen Systems Manager Command (dokumen SSM) yang mendefinisikan tugas yang akan dijalankan.
   2. Untuk versi Dokumen, pilih versi dokumen yang akan digunakan.

3. Untuk Prioritas tugas, tentukan prioritas untuk tugas ini. Nol (`0`) adalah prioritas tertinggi. Tugas di jendela pemeliharaan dijadwalkan dalam urutan prioritas dengan tugas yang memiliki prioritas yang sama dijadwalkan secara paralel.

Automation

1. Dalam daftar dokumen Otomasi, pilih runbook Otomasi yang mendefinisikan tugas yang akan dijalankan.
2. Untuk Versi dokumen, pilih versi runbook untuk digunakan.
3. Untuk Prioritas tugas, tentukan prioritas untuk tugas ini. Nol (`0`) adalah prioritas tertinggi. Tugas di jendela pemeliharaan dijadwalkan dalam urutan prioritas dengan tugas yang memiliki prioritas yang sama dijadwalkan secara paralel.

Lambda

1. Di area parameter Lambda, pilih fungsi Lambda dari daftar.
2. (Opsional) Berikan konten apa pun untuk Payload, Konteks Klien, atau Kualifikasi yang ingin Anda sertakan.

> **Note**
>
> Dalam beberapa kasus, Anda dapat menggunakan parameter semu sebagai bagian dari `Payload` nilai Anda. Kemudian ketika tugas jendela pemeliharaan berjalan, ia melewati nilai yang benar alih-alih placeholder parameter semu. Untuk informasi, lihat Menggunakan parameter semu saat mendaftarkan tugas jendela pemeliharaan.

3. Untuk Prioritas tugas, tentukan prioritas untuk tugas ini. Nol (`0`) adalah prioritas tertinggi. Tugas di jendela pemeliharaan dijadwalkan dalam urutan prioritas dengan tugas yang memiliki prioritas yang sama dijadwalkan secara paralel.

Step Functions

1. Di area parameter Step Functions, pilih mesin status dari daftar.
2. (Opsional) Berikan nama untuk eksekusi mesin status dan konten apa pun untuk Input yang ingin Anda sertakan.

> **Note**
>
> Dalam beberapa kasus, Anda dapat menggunakan parameter semu sebagai bagian dari `Input` nilai Anda. Kemudian ketika tugas jendela pemeliharaan berjalan, ia melewati nilai yang benar alih-alih placeholder parameter semu. Untuk informasi, lihat Menggunakan parameter semu saat mendaftarkan tugas jendela pemeliharaan.

3. Untuk Prioritas tugas, tentukan prioritas untuk tugas ini. Nol (`0`) adalah prioritas tertinggi. Tugas di jendela pemeliharaan dijadwalkan dalam urutan prioritas dengan tugas yang memiliki prioritas yang sama dijadwalkan secara paralel.

9. Di area Target, pilih salah satu dari berikut ini:

- Pemilihan grup target terdaftar: Pilih satu atau beberapa target jendela pemeliharaan yang telah Anda daftarkan dengan jendela pemeliharaan saat ini.
- Pemilihan target yang tidak terdaftar: Pilih sumber daya yang tersedia satu per satu sebagai target untuk tugas.

  Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.
- Target tugas tidak diperlukan: Target untuk tugas mungkin sudah ditentukan dalam fungsi lain untuk semua kecuali tugas Run Command -type.

  Tentukan satu atau beberapa target untuk tugas Run Command tipe jendela pemeliharaan. Tergantung dari tugas, target bersifat opsional untuk jenis tugas jendela pemeliharaan lainnya (Otomatisasi, AWS Lambda, dan AWS Step Functions). Untuk informasi lebih lanjut tentang menjalankan tugas yang tidak menentukan target, lihat Pendaftaran tugas jendela pemeliharaan tanpa target.

> **Note**
>
> Dalam banyak kasus, Anda tidak perlu secara eksplisit menentukan target untuk tugas otomatisasi. Misalnya, katakanlah bahwa Anda membuat tugas jenis otomatisasi untuk memperbarui Amazon Machine Image (AMI) untuk Linux menggunakan `AWS-UpdateLinuxAmi` runbook. Ketika tugas berjalan, AMI diperbarui dengan paket distribusi Linux terbaru yang tersedia dan perangkat lunak Amazon. Contoh baru dibuat dari AMI telah menginstal pembaruan ini. Karena ID AMI yang akan diperbarui

ditentukan dalam parameter input untuk runbook, tidak perlu untuk menentukan target lagi dalam tugas jendela pemeliharaan.

10. Hanya tugas otomatisasi:

    Di area parameter Input, berikan nilai untuk parameter yang diperlukan atau opsional yang diperlukan untuk menjalankan tugas Anda.

    > **Note**
    >
    > Dalam beberapa kasus, Anda dapat menggunakan parameter semu untuk nilai parameter input tertentu. Kemudian ketika tugas jendela pemeliharaan berjalan, ia melewati nilai yang benar alih-alih placeholder parameter semu. Untuk informasi, lihat Menggunakan parameter semu saat mendaftarkan tugas jendela pemeliharaan.

11. Untuk Pengendalian rate:

    - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

      > **Note**
      >
      > Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

    - Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

12. Di area peran layanan IAM, pilih peran untuk memberikan izin bagi Systems Manager untuk menjalankan tugas pada node target Anda.

    Jika Anda perlu membuat peran layanan kustom untuk tugas jendela pemeliharaan, lihatGunakan konsol untuk mengonfigurasi izin untuk jendela pemeliharaan.

13. Run Commandtugas saja:

(Opsional) Untuk opsi Output, lakukan hal berikut:

- Pilih kotak centang Aktifkan penulisan ke S3 untuk menyimpan output perintah ke file. Masukkan nama bucket dan prefiks (folder) di dalam kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin dari profil instance yang ditetapkan ke node, bukan izin pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager. Selain itu, jika bucket S3 yang ditentukan berbedaAkun AWS, verifikasi bahwa profil instance yang terkait dengan node memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

- Pilih kotak centang CloudWatch output untuk menulis output lengkap ke Amazon CloudWatch Logs. Masukkan nama grup CloudWatch log Log.

14. Run Commandtugas saja:

    Di bagian Notifikasi SNS, jika Anda ingin notifikasi dikirim tentang status eksekusi perintah, pilih kotak centang Aktifkan notifikasi SNS.

    Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Run Command Amazon SNS, lihat. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

15. Run Commandtugas saja:

    Di area Parameter, tentukan parameter untuk dokumen.

> **Note**
>
> Dalam beberapa kasus, Anda dapat menggunakan parameter semu untuk nilai parameter input tertentu. Kemudian ketika tugas jendela pemeliharaan berjalan, ia melewati nilai yang benar alih-alih placeholder parameter semu. Untuk informasi, lihat Menggunakan parameter semu saat mendaftarkan tugas jendela pemeliharaan.

16. Run Commanddan tugas Otomasi saja:

    (Opsional) Di area CloudWatch alarm, untuk nama Alarm, pilih CloudWatch alarm yang ada untuk diterapkan pada tugas Anda untuk pemantauan.

Jika alarm aktif, tugas dihentikan.

> **Note**
>
> Untuk melampirkan CloudWatch alarm ke tugas Anda, kepala sekolah IAM yang menjalankan tugas harus memiliki izin untuk `iam:createServiceLinkedRole` tindakan tersebut. Untuk informasi selengkapnya tentang CloudWatch alarm, lihat Menggunakan CloudWatch alarm Amazon.

17. Bergantung pada jenis tugas Anda, pilih salah satu dari yang berikut ini:
    - Daftar Jalankan tugas perintah
    - Daftarkan tugas Otomasi
    - Daftarkan tugas Lambda
    - Daftarkan tugas Step Functions

Menonaktifkan atau mengaktifkan jendela pemeliharaan

Anda dapat menonaktifkan atau mengaktifkan jendela pemeliharaan di dalamnyaMaintenance WindowsAWS Systems Manager. Anda dapat memilih satu jendela pemeliharaan sekaligus untuk menonaktifkan atau mengaktifkan jendela pemeliharaan agar tidak berjalan. Anda juga dapat memilih beberapa atau semua jendela pemeliharaan untuk mengaktifkan dan menonaktifkan.

Bagian ini menjelaskan cara menonaktifkan atau mengaktifkan jendela pemeliharaan dengan menggunakan konsol Systems Manager. Untuk contoh cara melakukannya dengan menggunakan AWS Command Line Interface (AWS CLI), lihat Tutorial: Memperbarui jendela pemeliharaan (AWS CLI).

Topik

- Menonaktifkan jendela pemeliharaan (konsol)
- Mengaktifkan jendela pemeliharaan (konsol)

Menonaktifkan jendela pemeliharaan (konsol)

Anda dapat menonaktifkan jendela pemeliharaan untuk menjeda tugas selama periode tertentu, dan itu akan tetap tersedia untuk diaktifkan lagi nanti.

Untuk menonaktifkan jendela pemeliharaan

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Maintenance Windows.
3. Centang kotak di samping jendela pemeliharaan yang ingin Anda nonaktifkan, centang kotak di samping jendela pemeliharaan yang ingin Anda nonaktifkan.
4. Pilih Nonaktifkan jendela pemeliharaan di menu Tindakan. Sistem meminta Anda untuk mengonfirmasi tindakan Anda.

Mengaktifkan jendela pemeliharaan (konsol)

Anda dapat mengaktifkan jendela pemeliharaan untuk melanjutkan tugas.

Untuk mengaktifkan jendela pemeliharaan

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Maintenance Windows.
3. Centang kotak di samping jendela pemeliharaan yang ingin Anda aktifkan, centang kotak pemeliharaan yang ingin Anda aktifkan.
4. Pilih Aktifkan jendela pemeliharaan di menu Tindakan. Sistem meminta Anda untuk mengonfirmasi tindakan Anda.

Memperbarui atau menghapus sumber daya jendela pemeliharaan (konsol)

Anda dapat memperbarui atau menghapus jendela pemeliharaanMaintenance Windows, suatu kemampuanAWS Systems Manager. Anda juga dapat memperbarui atau menghapus target atau tugas dari jendela pemeliharaan. Jika Anda mengedit detail jendela pemeliharaan, Anda dapat mengubah jadwal, target, dan tugas. Anda juga dapat menentukan nama dan deskripsi untuk windows, target, dan tugas, yang membantu Anda memahami kegunaan dengan lebih baik, dan mempermudah untuk mengelola antrean windows Anda.

Bagian ini menjelaskan cara memperbarui atau menghapus jendela pemeliharaan, target, dan tugas dengan menggunakan konsol Systems Manager. Untuk contoh cara melakukannya dengan

menggunakan AWS Command Line Interface (AWS CLI), lihat Tutorial: Memperbarui jendela pemeliharaan (AWS CLI).

Topik

- Memperbarui atau menghapus jendela pemeliharaan (konsol)
- Memperbarui atau membatalkan pendaftaran target jendela pemeliharaan (konsol)
- Memperbarui atau membatalkan pendaftaran tugas jendela pemeliharaan (konsol)

Memperbarui atau menghapus jendela pemeliharaan (konsol)

Anda dapat memperbarui jendela pemeliharaan untuk mengubah nama, deskripsi, dan jadwal, dan apakah jendela pemeliharaan harus mengizinkan target yang tidak terdaftar.

Untuk memperbarui atau menghapus jendela pemeliharaan

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Maintenance Windows.
3. Pilih tombol di samping jendela pemeliharaan yang ingin Anda perbarui atau hapus, lalu lakukan salah satu hal berikut:
   - Pilih Hapus. Sistem meminta Anda untuk mengonfirmasi tindakan Anda.
   - Pilih Edit. PadaMengedit jendela pemeliharaanhalaman, ubah nilai dan pilihan yang Anda inginkan, lalu pilihSimpan perubahan.

     Untuk informasi tentang pilihan konfigurasi yang dapat Anda buat, lihat Membuat jendela pemeliharaan (konsol).

Memperbarui atau membatalkan pendaftaran target jendela pemeliharaan (konsol)

Anda dapat memperbarui atau membatalkan pendaftaran target jendela pemeliharaan. Jika Anda memilih untuk memperbarui target jendela pemeliharaan, Anda dapat menentukan nama, deskripsi, dan pemilik target yang baru. Anda juga dapat memilih berbagai target.

Untuk memperbarui atau menghapus target jendela pemeliharaan

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Maintenance Windows.

3. Pilih nama jendela pemeliharaan yang ingin Anda perbarui, pilihTargettab, kemudian lakukan salah satu hal berikut:

   - Untuk memperbarui target, pilih tombol di samping target yang akan diperbarui, lalu pilihMengedit.
   - Untuk membatalkan pendaftaran target, pilih tombol di sebelah target untuk membatalkan pendaftaran, lalu pilihBatalkan pendaftaran target. DiDeregister pemeliharaan jendela targetkotak dialog, pilihBatalkan pendaftaran.

Memperbarui atau membatalkan pendaftaran tugas jendela pemeliharaan (konsol)

Anda dapat memperbarui atau membatalkan pendaftaran tugas jendela pemeliharaan. Jika Anda memilih untuk memperbarui, Anda dapat menentukan nama, deskripsi, dan pemilik tugas yang baru. UntukRun Commanddan Tugas Otomatisasi, Anda dapat memilih dokumen SSM yang berbeda untuk tugas. Akan tetapi, Anda tidak dapat mengedit tugas untuk mengubah jenisnya. Misalnya, jika Anda membuat tugas Otomatisasi, Anda tidak dapat mengedit tugas tersebut dan mengubahnya menjadiRun Commandtugas.

Untuk memperbarui atau menghapus tugas jendela pemeliharaan (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Maintenance Windows.
3. Pilih nama jendela pemeliharaan yang ingin Anda perbarui.
4. PilihTugastab, kemudian pilih tombol di samping tugas untuk memperbarui.
5. Lakukan salah satu dari berikut:

   - Untuk membatalkan pendaftaran tugas, pilihBatalkan pendaftaran tugas.
   - Untuk mengedit tugas, pilihMengedit. Ubah nilai dan pilihan yang Anda inginkan, lalu pilihMengedit tugas.

Systems ManagerMaintenance Windows tutorial (AWS CLI)

Bagian ini menyertakan tutorial yang membantu Anda mempelajari cara menggunakan AWS Command Line Interface (AWS CLI) untuk melakukan hal berikut:

- Membuat dan mengonfigurasi jendela pemeliharaan
- Melihat informasi tentang jendela pemeliharaan

- Melihat informasi tentang tugas windows pemeliharaan dan eksekusi tugas
- Memperbarui jendela pemeliharaan
- Menghapus jendela pemeliharaan

Prasyarat lengkap

Sebelum mencoba tutorial ini, lengkapi prasyarat berikut.

- MengonfigurasiAWS CLI pada mesin lokal Anda — Sebelum Anda dapat menjalankanAWS CLI perintah, Anda harus menginstal dan mengonfigurasi CLI pada mesin lokal Anda. Untuk informasi, lihat Menginstal atau memperbarui versi terbaruAWS CLI dan MemasangAWS Tools for PowerShell.
- Memverifikasi peran dan izin jendela pemeliharaan -AWS Administrator di akun Anda harus memberikan izinAWS Identity and Access Management (IAM) yang Anda butuhkan untuk mengelola windows pemeliharaan menggunakan CLI. Untuk informasi, lihat Menyiapkan Maintenance Windows.
- Membuat atau mengonfigurasi instans yang kompatibel dengan Systems Manager — Anda memerlukan setidaknya satu instans Amazon Elastic Compute Cloud (Amazon EC2) yang dikonfigurasi untuk digunakan dengan Systems Manager untuk menyelesaikan tutorial. SSM AgentIni artinya diinstal pada instans, dan profil instans IAM untuk Systems Manager dilampirkan pada instans.

  Sebaiknya luncurkan instance dari satuAWS managedAmazon Machine Image (AMI) dengan agen yang sudah diinstal sebelumnya. Untuk informasi selengkapnya, lihat Amazon Machine Images(AMIs) dengan SSM Agent prainstal.

  Untuk informasi tentang menginstalSSM Agent pada instans, lihat topik berikut:

  - Bekerja dengan SSM Agent instans EC2 untuk Windows Server
  - Bekerja denganSSM Agent instans EC2 untuk Linux

  Untuk informasi tentang mengonfigurasi izin IAM untuk Systems Manager ke instans Anda, lihat Mengonfigurasi izin instans untuk Systems Manager.
- Membuat sumber daya tambahan sesuai kebutuhan —Run Command, sebuah kemampuan Systems Manager, mencakup banyak tugas yang tidak mengharuskan Anda untuk membuat sumber daya selain yang tercantum dalam topik prasyarat ini. Untuk alasan itu, kami menyediakanRun Command tugas sederhana bagi Anda untuk digunakan saat pertama kali menjalani tutorial. Anda juga memerlukan instans EC2 yang dikonfigurasi untuk digunakan dengan

Systems Manager, seperti yang dijelaskan sebelumnya di topik ini. Setelah mengonfigurasi instans tersebut, Anda dapat mendaftarkanRun Command tugas sederhana.

Maintenance WindowsKemampuan Systems Manager mendukung untuk menjalankan empat jenis tugas:

- Run Commandperintah
- Alur kerja Otomatisasi Systems Manager
- Fungsi AWS Lambda
- Tugas AWS Step Functions

Secara umum, jika tugas jendela pemeliharaan yang ingin Anda jalankan memerlukan sumber daya tambahan, Anda harus membuatnya terlebih dahulu. Misalnya, jika Anda menginginkan jendela pemeliharaan yang menjalankanAWS Lambda fungsi membuat fungsi Lambda sebelum Anda memulai; untukRun Command tugas, buat bucket S3 agar Anda dapat menyimpan perintah output (jika Anda berencana untuk melakukannya); dan seterusnya.

Melacak ID sumber daya

Saat menyelesaikan tugas di tutorial AWS CLI ini, lacak ID sumber daya yang dihasilkan oleh perintah yang Anda jalankan. Anda menggunakan banyak dari hal ini sebagai input untuk perintah berikutnya. Misalnya, saat membuat jendela pemeliharaan, sistem memberi Anda ID jendela pemeliharaan dalam format berikut.

```
{
   "WindowId":"mw-0c50858d01EXAMPLE"
}
```

Catat ID yang dihasilkan sistem berikut karena tutorial di bagian ini menggunakannya:

- `WindowId`
- `WindowTargetId`
- `WindowTaskId`
- `WindowExecutionId`
- `TaskExecutionId`
- `InvocationId`
- `ExecutionId`

Anda juga memerlukan ID dari instans EC2 yang ingin Anda gunakan di tutorial. Sebagai contoh: `i-02573cafcfEXAMPLE`

Tutorial

- Tutorial: Membuat dan mengonfigurasi jendela pemeliharaan (AWS CLI)
- Tutorial: Melihat informasi tentang windows pemeliharaan (AWS CLI)
- Tutorial: Melihat informasi tentang tugas dan eksekusi tugas (AWS CLI)
- Tutorial: Memperbarui jendela pemeliharaan (AWS CLI)
- Tutorial: Menghapus jendela pemeliharaan (AWS CLI)

Tutorial: Membuat dan mengonfigurasi jendela pemeliharaan (AWS CLI)

Tutorial ini mendemonstrasikan cara menggunakan AWS Command Line Interface (AWS CLI) untuk membuat dan mengonfigurasi jendela pemeliharaan, target, dan tugasnya. Jalur utama pada tutorial terdiri dari langkah sederhana. Anda membuat satu jendela pemeliharaan, mengidentifikasi satu target, dan mengatur tugas sederhana untuk jendela pemeliharaan untuk dijalankan. Sepanjang perjalanan, kami memberi informasi yang dapat Anda gunakan untuk mencoba skenario yang lebih rumit.

Saat Anda mengikuti langkah di tutorial ini, ganti nilai di teks *merah* italik dengan pilihan dan ID Anda sendiri. Misalnya, ganti ID jendela pemeliharaan *mw-0c50858d01EXAMPLE* dan ID instans *i-02573cafcfEXAMPLE* dengan ID sumber daya yang Anda buat.

Konten

- Langkah 1: Membuat jendela pemeliharaan (AWS CLI)
- Langkah 2: Mendaftarkan node target dengan jendela pemeliharaan (AWS CLI)
- Langkah 3: Mendaftarkan tugas dengan jendela pemeliharaan (AWS CLI)

Langkah 1: Membuat jendela pemeliharaan (AWS CLI)

Pada langkah ini, Anda membuat jendela pemeliharaan dan menentukan pilihan dasar, seperti nama, jadwal, dan durasi. Di langkah berikutnya, Anda memilih instans yang diperbarui olehnya dan tugas yang dijalankannya.

Di contoh, Anda membuat jendela pemeliharaan yang berjalan setiap lima menit. Biasanya, Anda tidak akan menjalankan jendela pemeliharaan sesering ini. Akan tetapi, dengan nilai ini Anda dapat

melihat hasil tutorial dengan cepat. Kami akan menunjukkan cara untuk mengubah ke nilai yang lebih tidak sering kepada Anda setelah tugas berjalan dengan sukses.

> **Note**
>
> Untuk penjelasan tentang cara berbagai pilihan terkait jadwal untuk windows pemeliharaan berkaitan satu sama lain, lihat Penjadwalan jendela pemeliharaan dan pilihan periode aktif. Untuk informasi lebih lanjut tentang penggunaan pilihan `--schedule`, lihat Referensi: Ekspresi cron dan rate untuk Systems Manager.

Untuk membuat jendela pemeliharaan (AWS CLI)

1. Buka AWS Command Line Interface (AWS CLI) dan jalankan perintah berikut pada mesin lokal Anda untuk membuat jendela pemeliharaan yang melakukan hal berikut:

   - Berjalan setiap lima menit hingga dua jam (sesuai kebutuhan).
   - Mencegah tugas baru dimulai dalam waktu satu jam setelah akhir operasi jendela pemeliharaan.
   - Mengizinkan target yang tidak terkait (instans yang belum Anda daftarkan dengan jendela pemeliharaan).
   - Menunjukkan melalui penggunaan tanda kustom yang ingin digunakan oleh pembuatnya di 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"
```

Sistem mengembalikan informasi seperti berikut.

```
{
   "WindowId":"mw-0c50858d01EXAMPLE"
}
```

2. Sekarang jalankan perintah berikut untuk melihat detail tentang windows pemeliharaan ini dan yang lainnya yang sudah ada di akun Anda.

```
aws ssm describe-maintenance-windows
```

Sistem mengembalikan informasi seperti berikut.

```
{
   "WindowIdentities":[
      {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "My-First-Maintenance-Window",
            "Enabled": true,
            "Duration": 2,
            "Cutoff": 1,
            "NextExecutionTime": "2019-05-11T16:46:16.991Z"
      }
   ]
}
```

Lanjutkan ke Langkah 2: Mendaftarkan node target dengan jendela pemeliharaan (AWS CLI).

Langkah 2: Mendaftarkan node target dengan jendela pemeliharaan (AWS CLI)

Pada langkah ini, Anda mendaftarkan target dengan jendela pemeliharaan baru. Dalam hal ini, Anda menentukan node mana yang diperbarui saat jendela pemeliharaan berjalan.

Untuk contoh pendaftaran lebih dari satu node pada satu waktu menggunakan ID node, contoh penggunaan tanda untuk mengidentifikasi beberapa node, dan contoh penentuan grup sumber daya sebagai target, lihatContoh: Mendaftarkan target dengan jendela pemeliharaan.

> **Note**
>
> Anda harus sudah menciptakan instans Amazon Elastic Compute Cloud (Amazon EC2) untuk digunakan di langkah ini, seperti yang dijelaskan di PrasyaratMaintenance Windows tutorial.

Untuk mendaftarkan node target dengan jendela pemeliharaan (AWS CLI)

1. Jalankan perintah berikut di mesin lokal Anda. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

   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"
   ```

   Sistem mengembalikan informasi seperti berikut.

   ```
   {
      "WindowTargetId":"e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
   }
   ```

2. Sekarang jalankan perintah berikut pada mesin lokal Anda untuk melihat detail tentang target jendela pemeliharaan Anda.

Linux & macOS

```
aws ssm describe-maintenance-window-targets \
    --window-id "mw-0c50858d01EXAMPLE"
```

Windows

```
aws ssm describe-maintenance-window-targets ^
    --window-id "mw-0c50858d01EXAMPLE"
```

Sistem mengembalikan informasi seperti berikut.

```
{
    "Targets": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE",
            "ResourceType": "INSTANCE",
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-02573cafcfEXAMPLE"
                    ]
                }
            ]
        }
    ]
}
```

Lanjutkan ke Langkah 3: Mendaftarkan tugas dengan jendela pemeliharaan (AWS CLI).

Contoh: Mendaftarkan target dengan jendela pemeliharaan

Anda dapat mendaftarkan satu node sebagai target menggunakan ID simsnya, seperti yang didemonstrasikan diLangkah 2: Mendaftarkan node target dengan jendela pemeliharaan (AWS CLI). Anda juga dapat mendaftarkan satu atau beberapa node sebagai target menggunakan format perintah pada halaman ini.

Secara umum, ada dua metode untuk mengidentifikasi node yang ingin Anda gunakan sebagai target jendela pemeliharaan: penentuan node individu, dan penggunaan tanda sumber daya. Metode tanda sumber daya menyediakan lebih banyak pilihan, seperti yang ditunjukkan di contoh 2-3.

Anda juga dapat menentukan satu atau beberapa grup sumber daya sebagai target jendela pemeliharaan. Grup sumber daya dapat menyertakan node dan berbagai jenisAWS sumber daya lainnya yang didukung. Contoh 4 dan 5, berikutnya, mendemonstrasikan cara menambahkan grup sumber daya ke target jendela pemeliharaan Anda.

> **Note**
>
> Jika satu tugas jendela pemeliharaan terdaftar dengan beberapa target, permintaan tugasnya terjadi secara berurutan dan tidak secara paralel. Jika tugas Anda harus berjalan pada beberapa target di waktu yang sama, daftarkan tugas untuk setiap target secara individual dan tetapkan pada setiap tugas tingkat prioritas yang sama.

Untuk informasi selengkapnya tentang membuat dan mengelola grup sumber daya, lihat Apa itu grup sumber daya? dalam PanduanAWS Resource Groups Pengguna dan Resource Groups dan Penandaan untukAWS di BlogAWS Berita.

Untuk informasi tentang kuota untukMaintenance Windows, sebuah kemampuanAWS Systems Manager, selain yang ditentukan dalam contoh berikut, liha t Kuota layanan Systems Manager di bagian Referensi Umum Amazon Web Services.

Contoh 1: Mendaftarkan beberapa target menggunakan ID simpul

Jalankan perintah berikut pada format mesin lokal Anda untuk mendaftarkan beberapa node sebagai target menggunakan ID simsnya. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri.

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
```

Penggunaan yang direkomendasikan: Paling berguna saat mendaftarkan grup node unik dengan jendela pemeliharaan apa pun untuk pertama kalinya dan tidak berbagi tanda simpul umum.

Kuota: Anda dapat menentukan hingga total 50 node untuk setiap target jendela pemeliharaan.

Contoh 2: Mendaftarkan target menggunakan tanda sumber daya yang diterapkan ke node

Jalankan perintah berikut pada mesin lokal Anda untuk mendaftarkan node yang semuanya sudah ditandai dengan pasangan nilai kunci yang telah Anda tetapkan. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

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"
```

Penggunaan yang direkomendasikan: Paling berguna saat mendaftarkan grup node unik dengan jendela pemeliharaan apa pun untuk pertama kalinya dan memang berbagi tanda simpul umum.

Kuota: Anda dapat menentukan hingga total lima pasangan nilai kunci untuk setiap target. Jika Anda menentukan lebih dari satu pasangan nilai kunci, simpul harus ditandai dengan semua kunci tanda dan nilai yang Anda tentukan untuk disertakan dalam grup target.

> **Note**
>
> Anda dapat menandai grup node dengan kunci tandaPatch Group atauPatchGroup dan menetapkan nilai kunci umum pada node, sepertimy-patch-group. (Anda harus menggunakanPatchGroup, tanpa spasi, jika Anda telah mengizinkan tag dalam metadata instans EC2.) Patch Manager, sebuah kemampuan Systems Manager, mengevaluasiPatch Group atauPatchGroup kunci pada node untuk membantu menentukan dasar patch mana yang berlaku. Jika tugas akan menjalankan dokumenAWS-RunPatchBaseline SSM (atau dokumenAWS-ApplyPatchBaseline SSM warisan), Anda dapat menentukan pasanganPatchGroup nilai kunciPatch Group atau nilai kunci saat mendaftarkan target dengan jendela pemeliharaan. Misalnya: `--target "Key=tag:PatchGroup,Values=`*`my-patch-group`*`"`. Melakukan hal ini memungkinkan Anda untuk menggunakan jendela pemeliharaan untuk memperbarui patch grup node yang sudah dikaitkan dengan dasar patch yang sama. Untuk informasi selengkapnya, lihat Tentang grup patch.

Contoh 3: Mendaftarkan target menggunakan grup kunci tanda (tanpa nilai tanda)

Jalankan perintah berikut pada mesin lokal Anda untuk mendaftarkan node yang semuanya memiliki satu atau beberapa kunci tanda yang ditetapkan padanya, apa pun nilai kuncinya. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri.

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"
```

Penggunaan yang direkomendasikan: Paling berguna saat Anda ingin menargetkan node dengan menentukan beberapa kunci tanda (tanpa nilainya) daripada hanya satu kunci tanda atau pasangan nilai kunci tanda.

Kuota: Anda dapat menentukan hingga total lima kunci tanda untuk setiap target. Jika Anda menentukan lebih dari satu kunci tanda, node harus ditandai dengan semua kunci tanda yang Anda tentukan untuk disertakan dalam grup target.

Contoh 4: Mendaftarkan target menggunakan nama grup sumber daya

Jalankan perintah berikut pada mesin lokal Anda untuk mendaftarkan grup sumber daya tertentu, apa pun jenis sumber daya yang dikandungnya. Ganti *MW-0C50858D01Example* dengan informasi Anda sendiri. Jika tugas yang Anda tetapkan ke jendela pemeliharaan tidak berfungsi pada jenis sumber daya yang disertakan dalam grup sumber daya ini, sistem mungkin melaporkan kesalahan. Tugas di mana jenis sumber daya yang didukung ditemukan akan terus berjalan meskipun ada kesalahan ini.

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"
```

Penggunaan yang direkomendasikan: Paling berguna saat Anda ingin dengan cepat menentukan grup sumber daya sebagai target tanpa mengevaluasi apakah semua jenis sumber dayanya akan ditargetkan oleh jendela pemeliharaan, atau saat Anda tahu bahwa grup sumber daya hanya berisi jenis sumber daya yang ditindak oleh tugas Anda.

Kuota: Anda hanya dapat menentukan satu grup sumber daya sebagai target.

Contoh 5: Mendaftarkan target dengan memfilter jenis sumber daya di grup sumber daya

Jalankan perintah berikut pada mesin lokal Anda untuk mendaftarkan jenis sumber daya tertentu dimiliki oleh grup sumber daya yang Anda tentukan saja. Ganti *MW-0C50858D01Example* dengan informasi Anda sendiri. Dengan pilihan ini, meski Anda menambahkan tugas untuk jenis sumber daya yang dimiliki oleh grup sumber daya, tugas tidak akan berjalan jika Anda belum secara eksplisit menambahkan jenis sumber daya ke filter.

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"
```

Penggunaan yang direkomendasikan: Paling berguna saat Anda ingin mempertahankan kendali yang ketat atas jenis sumber daya AWS di mana jendela pemeliharaan Anda dapat menjalankan tindakan, atau saat grup sumber daya Anda berisi sejumlah besar jenis sumber daya dan Anda ingin menghindari laporan kesalahan yang tidak perlu di log jendela pemeliharaan Anda.

Kuota: Anda hanya dapat menentukan satu grup sumber daya sebagai target.

Langkah 3: Mendaftarkan tugas dengan jendela pemeliharaan (AWS CLI)

Pada langkah tutorial ini, Anda mendaftarkan AWS Systems Manager Run Command tugas yang menjalankan `df` perintah pada instans Amazon Elastic Compute Cloud (Amazon EC2) untuk Linux. Hasil perintah Linux standar ini menunjukkan berapa banyak ruang yang kosong dan berapa banyak yang digunakan pada sistem file disk dari instans Anda.

-atau-

Jika Anda menargetkan instans Amazon EC2 untuk Windows Server dan bukan Linux, ganti df di perintah berikut dengan ipconfig. Output dari perintah ini mencantumkan detail tentang alamat IP, subnet mask, dan gateway default untuk adapter pada instans target.

Saat Anda siap mendaftarkan jenis tugas lain, atau menggunakan lebih banyak Run Command opsi Systems Manager yang tersedia, lihatContoh: Mendaftarkan tugas dengan jendela pemeliharaan. Di sana, kami menyediakan informasi lebih lanjut tentang keempat jenis tugas, dan beberapa pilihan yang paling penting, untuk membantu Anda merencanakan skenario dunia nyata yang lebih ekstensif.

Untuk mendaftarkan tugas dengan jendela pemeliharaan

1. Jalankan perintah berikut pada mesin lokal Anda. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri. Versi yang dijalankan dari mesin Windows lokal mencakup karakter keluar (“/”) yang Anda perlukan untuk menjalankan perintah dari alat baris perintah Anda.

   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\"]}}}
```

Sistem mengembalikan informasi seperti berikut ini:

```
{
    "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
}
```

2. Sekarang jalankan perintah berikut untuk melihat detail tentang tugas jendela pemeliharaan yang Anda buat.

   Linux & macOS

   ```
   aws ssm describe-maintenance-window-tasks \
       --window-id mw-0c50858d01EXAMPLE
   ```

   Windows

   ```
   aws ssm describe-maintenance-window-tasks ^
       --window-id mw-0c50858d01EXAMPLE
   ```

3. Sistem mengembalikan informasi seperti berikut ini.

```
{
    "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. Tunggu sampai waktunya tugas dijalankan, berdasarkan jadwal yang Anda tentukan di Langkah 1: Membuat jendela pemeliharaan (AWS CLI). Misalnya, jika Anda menentukan **`--schedule "rate(5 minutes)"`**, tunggu selama lima menit. Lalu jalankan perintah berikut untuk melihat informasi tentang eksekusi yang terjadi untuk tugas ini.

   Linux & macOS

   ```
   aws ssm describe-maintenance-window-executions \
       --window-id mw-0c50858d01EXAMPLE
   ```

   Windows

   ```
   aws ssm describe-maintenance-window-executions ^
       --window-id mw-0c50858d01EXAMPLE
   ```

   Sistem mengembalikan informasi seperti berikut ini.

   ```
   {
       "WindowExecutions": [
           {
               "WindowId": "mw-0c50858d01EXAMPLE",
               "WindowExecutionId": "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE",
               "Status": "SUCCESS",
               "StartTime": 1557593493.096,
               "EndTime": 1557593498.611
           }
       ]
   }
   ```

Tip

Setelah tugas berjalan dengan sukses, Anda dapat mengurangi nilai yang dijalankan jendela pemeliharaan. Misalnya, jalankan perintah berikut untuk mengurangi frekuensi ke seminggu sekali. Ganti *MW-0C50858D01Example* dengan informasi Anda sendiri.

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)"
```

Untuk informasi lebih lanjut tentang pengelolaan jadwal jendela pemeliharaan, lihat Referensi: Ekspresi cron dan rate untuk Systems Manager dan Penjadwalan jendela pemeliharaan dan pilihan periode aktif.
Untuk informasi tentang penggunaan AWS Command Line Interface (AWS CLI) untuk mengubah jendela pemeliharaan, lihat Tutorial: Memperbarui jendela pemeliharaan (AWS CLI).

Untuk latihan menjalankan perintah AWS CLI untuk melihat detail lebih lanjut tentang tugas jendela pemeliharaan dan eksekusinya, lanjutkan ke Tutorial: Melihat informasi tentang tugas dan eksekusi tugas (AWS CLI).

Tentang output perintah tutorial

Ini di luar cakupan tutorial ini untuk menggunakan AWS CLI untuk melihat output dari Run Command perintah yang terkait dengan eksekusi tugas jendela pemeliharaan Anda.

Akan tetapi, Anda dapat melihat data ini menggunakan AWS CLI. (Anda juga dapat melihat output di konsol Systems Manager atau di berkas log yang disimpan di bucket Amazon Simple Storage Service (Amazon S3), jika Anda telah mengonfigurasi jendela pemeliharaan untuk menyimpan output

perintah di sana.) Anda akan menemukan bahwa output dari perintah df pada instans EC2 untuk Linux seperti berikut ini.

```
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% /
```

Output dari perintah ipconfig pada instans EC2 untuk Windows Server seperti berikut ini.

```
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  . :
```

Contoh: Mendaftarkan tugas dengan jendela pemeliharaan

Anda dapat mendaftarkan tugas diRun Command, kemampuanAWS Systems Manager, dengan jendela pemeliharaan menggunakan AWS Command Line Interface (AWS CLI), seperti yang ditunjukkan dalam Daftarkan tugas dengan jendela pemeliharaan. Anda juga dapat mendaftarkan tugas untuk alur kerja Otomatisasi Systems Manager, fungsi AWS Lambda, dan tugas AWS Step Functions, seperti yang didemonstrasikan nanti di topik ini.

> **Note**
>
> Tentukan satu atau beberapa target untuk tugas Run Command tipe jendela pemeliharaan. Tergantung dari tugas, target bersifat opsional untuk jenis tugas jendela pemeliharaan lainnya (Otomatisasi, AWS Lambda, dan AWS Step Functions). Untuk informasi lebih lanjut tentang menjalankan tugas yang tidak menentukan target, lihat Pendaftaran tugas jendela pemeliharaan tanpa target.

Di topik ini, kami memberikan contoh penggunaan perintah AWS Command Line Interface (AWS CLI) `register-task-with-maintenance-window` untuk mendaftarkan masing-masing dari keempat jenis tugas yang didukung dengan jendela pemeliharaan. Contoh hanya untuk demonstrasi, tetapi Anda dapat mengubahnya untuk membuat perintah pendaftaran tugas yang berfungsi.

Menggunakan cli-input-json opsi --

Untuk mengelola pilhan tugas Anda dengan lebih baik, Anda dapat menggunakan pilihan perintah `--cli-input-json`, dengan nilai pilihan direferensikan di file JSON.

Untuk menggunakan sampel konten file JSON yang kami sediakan dalam contoh berikut, lakukan hal berikut pada mesin lokal Anda:

1. Membuat file dengan nama seperti `MyRunCommandTask.json`, `MyAutomationTask.json`, atau nama lainnya yang Anda inginkan.

2. Salin konten sampel JSON kami ke dalam file.
3. Ubah konten file untuk pendaftaran tugas Anda, lalu simpan file.
4. Di direktori yang sama tempat Anda menyimpan file, jalankan perintah berikut. Ganti nama file Anda *MyFiledengan.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
   ```

Tentang parameter semu

Dalam beberapa contoh, kami menggunakan parameter pseudo sebagai metode untuk meneruskan informasi ID ke tugas Anda. Untuk instans, `{{TARGET_ID}}` dan `{{RESOURCE_ID}}` dapat digunakan untuk meneruskan ID dari sumber daya AWS ke tugas Otomatisasi, Lambda, dan Step Functions. Untuk informasi lebih lanjut tentang parameter semu di konten `--task-invocation-parameters`, lihat Menggunakan parameter semu saat mendaftarkan tugas jendela pemeliharaan.

Info lebih lanjut

- Tentang register-task-with-maintenance opsi -windows.
- register-task-with-maintenance-windowdalam Referensi AWS CLI Perintah
- RegisterTaskWithMaintenanceWindow di Referensi API AWS Systems Manager

Contoh pendaftaran tugas

Bagian berikut memberikan sampel perintah AWS CLI untuk mendaftarkan jenis tugas dan sampel JSON yang didukung yang dapat digunakan dengan pilihan `--cli-input-json`.

Mendaftarkan Run Command tugas Systems Manager

Contoh berikut menunjukkan cara mendaftarkan Run Command tugas Systems Manager dengan jendela pemeliharaan menggunakan fileAWS 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\"]}}}"
```

Konten JSON untuk digunakan dengan opsi **`--cli-input-json`** file:

```
{
    "TaskType": "RUN_COMMAND",
    "WindowId": "mw-0c50858d01EXAMPLE",
    "Description": "My Run Command task to update SSM Agent on an instance",
    "MaxConcurrency": "1",
    "MaxErrors": "1",
    "Name": "My-Run-Command-Task",
    "Priority": 10,
    "Targets": [
        {
            "Key": "WindowTargetIds",
            "Values": [
                "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
            ]
        }
    ],
    "TaskArn": "AWS-UpdateSSMAgent",
    "TaskInvocationParameters": {
        "RunCommand": {
```

```
            "Comment": "A TaskInvocationParameters test comment",
            "NotificationConfig": {
                "NotificationArn": "arn:aws:sns:region:123456789012:my-sns-topic-name",
                "NotificationEvents": [
                    "All"
                ],
                "NotificationType": "Invocation"
            },
            "OutputS3BucketName": "DOC-EXAMPLE-BUCKET",
            "OutputS3KeyPrefix": "DOC-EXAMPLE-FOLDER",
            "TimeoutSeconds": 3600
        }
    }
}
```

Mendaftarkan tugas Otomatisasi Systems Manager

Contoh berikut mendemonstrasikan cara untuk mendaftarkan tugas Otomatisasi Systems Manager dengan jendela pemeliharaan menggunakan AWS CLI.

AWS CLIperintah:

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"
```

Konten JSON untuk digunakan dengan opsi **`--cli-input-json`** file:

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
        "TaskArn": "AWS-PatchInstanceWithRollback",
    "TaskType": "AUTOMATION","TaskInvocationParameters": {
        "Automation": {
            "DocumentVersion": "1",
            "Parameters": {
                "instanceId": [
                    "{{RESOURCE_ID}}"
                ]
            }
        }
    }
}
```

Mendaftarkan tugas AWS Lambda

Contoh berikut mendemonstrasikan cara untuk mendaftarkan tugas fungsi Lambda dengan jendela pemeliharaan menggunakan AWS CLI.

Untuk contoh ini, pengguna yang membuat fungsi Lambda menamainya `SSMrestart-my-instances` dan membuat dua parameter yang disebut `instanceId` dan `targetType`.

> **Important**
>
> Kebijakan IAM untuk Maintenance Windows mengharuskan Anda menambahkan awalan ke nama fungsi `SSM` Lambda (atau alias). Sebelum melanjutkan untuk mendaftarkan jenis tugas ini, perbarui namanya di AWS Lambda untuk menyertakan `SSM`. Misalnya, jika nama fungsi Lambda Anda adalah `MyLambdaFunction`, ubah ke `SSMMyLambdaFunction`.

AWS CLIperintah:

Linux & macOS

> **Important**
>
> Jika Anda menggunakan versi 2 dariAWS CLI, Anda harus menyertakan opsi `--cli-binary-format raw-in-base64-out` dalam perintah berikut jika payload Lambda Anda tidak dikodekan base64. `cli_binary_format`Opsi ini hanya tersedia di versi 2. Untuk informasi tentang ini dan setelan AWS CLI `config` file lainnya, lihat Pengaturan `config` file yang didukung di Panduan AWS Command Line Interface Pengguna.

```
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**
>
> Jika Anda menggunakan versi 2 dariAWS CLI, Anda harus menyertakan opsi `--cli-binary-format raw-in-base64-out` dalam perintah berikut jika payload Lambda Anda tidak dikodekan base64. `cli_binary_format`Opsi ini hanya tersedia di versi 2. Untuk informasi tentang ini dan setelan AWS CLI `config` file lainnya, lihat Pengaturan `config` file yang didukung di Panduan AWS Command Line Interface Pengguna.

```
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\"}}'
```

Konten JSON untuk digunakan dengan opsi **`--cli-input-json`** file:

```
{
    "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
}
```

Mendaftarkan tugas Step Functions

Contoh berikut mendemonstrasikan cara untuk mendaftarkan tugas mesin tahapan Step Functions dengan jendela pemeliharaan menggunakan AWS CLI.

Untuk contoh ini, pengguna yang membuat mesin tahapan Step Functions membuat mesin tahapan bernama `SSMMyStateMachine` dengan parameter yang disebut `instanceId`.

> **Important**
>
> Kebijakan AWS Identity and Access Management (IAM) untuk Maintenance Windows mengharuskan Anda mengawali nama mesin status Step Functions. `SSM` Sebelum melanjutkan untuk mendaftarkan jenis tugas ini, Anda harus memperbarui namanya di AWS Step Functions untuk menyertakan `SSM`. Misalnya, jika nama mesin tahapan Anda adalah `MyStateMachine`, ubah ke `SSMMyStateMachine`.

AWS CLIperintah:

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"
```

Konten JSON untuk digunakan dengan opsi **`--cli-input-json`** file:

```
{
    "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
}
```

Tentang register-task-with-maintenance opsi -windows

Perintah register-task-with-maintenance-window menyediakan beberapa pilihan untuk pengonfigurasian tugas sesuai dengan kebutuhan Anda. Beberapa di antaranya diperlukan, beberapa di antaranya opsional, dan beberapa di antaranya berlaku untuk satu jenis tugas jendela pemeliharaan saja.

Topik ini memberikan informasi tentang beberapa pilihan ini untuk membantu Anda menggunakan sampel di bagian tutorial ini. Untuk informasi lebih lanjut tentang semua pilihan perintah, lihat register-task-with-maintenance-window di Referensi Perintah AWS CLI.

Tentang pilihan **`--task-arn`**

Opsi `--task-arn` ini digunakan untuk menentukan sumber daya tempat tugas beroperasi. Nilai yang Anda tentukan tergantung dari jenis tugas yang Anda daftarkan, seperti yang dijelaskan di tabel berikut.

TaskArn format untuk tugas jendela pemeliharaan

Jenis tugas jendela pemeliharaan	TaskArn nilai
RUN_COMMAND dan **AUTOMATION**	`TaskArn` adalah nama dokumen SSM atau Amazon Resource Name (ARN). Sebagai contoh: `AWS-RunBatchShellScript` -atau- `arn:aws:ssm:` *`region`*`:11112222 3333:document/My-Document` .
LAMBDA	`TaskArn` adalah nama fungsi atau ARN. Sebagai contoh: `SSMMy-Lambda-Function` -atau- `arn:aws:lambda:` *`region`*`:11112222 3333:function:SSMMyLambdaFu nction` . **Important** Kebijakan IAM untuk Maintenan ce Windows mengharuskan Anda menambahkan awalan ke nama fungsi `SSM` Lambda (atau alias). Sebelum melanjutkan untuk mendaftarkan jenis tugas ini, perbarui namanya di AWS Lambda untuk menyertakan `SSM`. Misalnya, jika nama fungsi Lambda Anda adalah `MyLambdaFunction` , ubah ke `SSMMyLambdaFunction` .

Jenis tugas jendela pemeliharaan	TaskArn nilai
STEP_FUNCTIONS	TaskArn adalah ARN mesin tahapan. Sebagai contoh: arn:aws:states:us-east-2:111122223333:stateMachine:SSMMyStateMachine . **Important** Kebijakan IAM untuk windows pemeliharaan mengharuskan Anda mengawali nama mesin tahapan Step Functions dengan SSM. Sebelum mendaftarkan jenis tugas ini, Anda harus memperbarui namanya di AWS Step Functions untuk menyertakan SSM. Misalnya, jika nama mesin tahapan Anda adalah MyStateMachine , ubah ke SSMMyStateMachine .

Tentang pilihan **--service-role-arn**

Peran untuk AWS Systems Manager untuk diasumsikan saat menjalankan tugas jendela pemeliharaan.

Untuk informasi selengkapnya, lihat Menyiapkan Maintenance Windows

Tentang pilihan **--task-invocation-parameters**

Pilihan --task-invocation-parameters digunakan untuk menentukan parameter yang bersifat unik untuk masing-masing dari keempat jenis tugas. Parameter yang didukung untuk masing-masing dari keempat jenis tugas dijelaskan di tabel berikut.

> **Note**
>
> Untuk informasi tentang penggunaan parameter semu di konten `--task-invocation-parameters`, seperti {{TARGET_ID}}, lihat Menggunakan parameter semu saat mendaftarkan tugas jendela pemeliharaan.

Pilihan parameter permintaan tugas untuk tugas jendela pemeliharaan

Jenis tugas jendela pemeliharaan	Parameter yang tersedia	Contoh
RUN_COMMAND	Komentar DocumentHash DocumentHashType NotificationConfig Keluaran3 BucketName OutPutS3 KeyPrefix Parameter 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" },`

Jenis tugas jendela pemeliharaan	Parameter yang tersedia	Contoh
		"OutputS3BucketName": " *DOC-EXAMPLE-BUCKET* ", "OutputS3KeyPrefix": " *DOC-EXAMPLE-FOLDER* ", "Parameters": { "commands": ["Get-ChildItem$env:temp-Recurse|Remove-Item-Recurse-force"] }, "ServiceRoleArn": "arn:aws:iam::123456789012:role/MyMaintenanceWindowServiceRole", "TimeoutSeconds": 3600 } }

Jenis tugas jendela pemeliharaan	Parameter yang tersedia	Contoh
otomatisasi	DocumentVersion Parameter	`"TaskInvocationParameters": { "Automation": { "DocumentVersion": "3", "Parameters": { "instanceid": [ "{{TARGET_ID}}" ] } } }`
LAMBDA	ClientContext Muatan Pengualifikasi	`"TaskInvocationParameters": { "Lambda": { "ClientContext": "ew0KICAi--truncated--0KIEXAMPLE", "Payload": "{ \"targetId\": \"{{TARGET_ID}}\", \"targetType\": \"{{TARGET_TYPE}}\" }", "Qualifier": "$LATEST" } }`

Jenis tugas jendela pemeliharaan	Parameter yang tersedia	Contoh
STEP_FUNCTIONS	Input Nama	`"TaskInvocationParameters": { "StepFunctions": { "Input": "{ \"targetId\": \"{{TARGET_ID}}\" }", "Name": "{{INVOCATION_ID}}" } }`

Tutorial: Melihat informasi tentang windows pemeliharaan (AWS CLI)

Tutorial ini menyertakan perintah untuk membantu Anda memperbarui atau mendapatkan informasi tentang windows pemeliharaan, tugas, eksekusi, dan permintaan Anda. Contoh diorganisasi oleh perintah untuk mendemonstrasikan cara menggunakan opsi perintah untuk memfilter jenis dari detail yang ingin Anda lihat.

Saat Anda mengikuti langkah di tutorial ini, ganti nilai di teks *merah* italik dengan pilihan dan ID Anda sendiri. Misalnya, ganti ID jendela pemeliharaan *mw-0c50858d01EXAMPLE* dan ID instans *i-02573cafcfEXAMPLE* dengan ID sumber daya yang Anda buat.

Untuk informasi tentang pengaturan dan konfigurasiAWS Command Line Interface(AWS CLI), lihatMenginstal, memperbarui, dan menghapus instalasiAWS CLIdanMengonfigurasiAWS CLI.

Contoh perintah

- Contoh untuk 'describe-maintenance-windows'
- Contoh untuk 'describe-maintenance-window-targets'
- Contoh untuk 'describe-maintenance-window-tasks'
- Contoh untuk 'describe-maintenance-windows-for-target'
- Contoh untuk 'describe-maintenance-window-executions'
- Contoh untuk 'describe-maintenance-window-schedule'

Contoh untuk 'describe-maintenance-windows'

Mencantumkan semua windows pemeliharaan di Akun AWS Anda

Jalankan perintah berikut.

```
aws ssm describe-maintenance-windows
```

Sistem mengembalikan informasi seperti berikut.

```
{
   "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"
      },
   ]
}
```

Daftar semua windows pemeliharaan yang diaktifkan

Jalankan perintah berikut.

```
aws ssm describe-maintenance-windows --filters "Key=Enabled,Values=true"
```

Sistem mengembalikan informasi seperti berikut.

```
{
   "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"
      },
   ]
}
```

Daftar semua windows pemeliharaan yang dinonaktifkan

Jalankan perintah berikut.

```
aws ssm describe-maintenance-windows --filters "Key=Enabled,Values=false"
```

Sistem mengembalikan informasi seperti berikut.

```
{
    "WindowIdentities": [
        {
            "WindowId": "mw-6e5c9d4b7cEXAMPLE",
            "Name": "My-Disabled-Maintenance-Window",
            "Enabled": false,
            "Duration": 2,
            "Cutoff": 1
        }
    ]
}
```

Daftar semua windows pemeliharaan yang memiliki nama yang dimulai dengan prefiks tertentu

Jalankan perintah berikut.

```
aws ssm describe-maintenance-windows --filters "Key=Name,Values=My"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "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
        }
    ]
}
```

Contoh untuk 'describe-maintenance-window-targets'

Menampilkan target untuk jendela pemeliharaan yang cocok dengan nilai informasi pemilik tertentu

Jalankan perintah berikut.

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**
>
> Kunci filter yang didukung adalah `Type`, `WindowTargetId` dan `OwnerInformation`.

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "Targets": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE",
            "ResourceType": "INSTANCE",
            "Targets": [
                {
                    "Key": "tag:Name",
                    "Values": [
                        "Production"
                    ]
                }
            ],
            "OwnerInformation": "CostCenter1",
            "Name": "Target1"
        }
    ]
}
```

Contoh untuk 'describe-maintenance-window-tasks'

Menunjukkan semua tugas terdaftar yang meminta dokumen perintah SSM **`AWS-RunPowerShellScript`**

Jalankan perintah berikut.

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"
```

Sistem mengembalikan informasi seperti berikut.

```
{
   "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"
            }
         ]
      }
   ]
}
```

Menunjukkan semua tugas terdaftar yang memiliki prioritas “3"

Jalankan perintah berikut.

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"
```

Sistem mengembalikan informasi seperti berikut.

```
{
```

```
   "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"
            }
         ]
      }
   ]
}
```

Menunjukkan semua tugas terdaftar yang memiliki prioritas “1" dan menggunakanRun Command

Jalankan perintah berikut.

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"
```

Sistem mengembalikan informasi seperti berikut.

```
{
    "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"
        }
    ]
}
```

Contoh untuk 'describe-maintenance-windows-for-target'

Mencantumkan informasi tentang target atau tugas jendela pemeliharaan yang dikaitkan dengan node tertentu

Jalankan perintah berikut.

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
```

Sistem mengembalikan informasi seperti berikut.

```
{
    "WindowIdentities": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "My-First-Maintenance-Window"
        },
        {
            "WindowId": "mw-9a8b7c6d5eEXAMPLE",
            "Name": "My-Second-Maintenance-Window"
        }
    ]
}
```

Contoh untuk 'describe-maintenance-window-executions'

Mencantumkan semua tugas yang berjalan sebelum tanggal tertentu

Jalankan perintah berikut.

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"
```

Sistem mengembalikan informasi seperti berikut.

```
{
    "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
        }
    ]
}
```

Mencantumkan semua tugas yang berjalan setelah tanggal tertentu

Jalankan perintah berikut.

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"
```

Sistem mengembalikan informasi seperti berikut.

```
{
    "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
        }
    ]
}
```

Contoh untuk 'describe-maintenance-window-schedule'

Menampilkan sepuluh jendela pemeliharaan terjadwal berikutnya yang berjalan untuk node tertentu

Jalankan perintah berikut.

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
```

Sistem mengembalikan informasi seperti berikut.

```
{
    "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"
}
```

Menampilkan jadwal jendela pemeliharaan untuk node yang ditandai dengan pasangan nilai kunci tertentu

Jalankan perintah berikut.

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"
```

Sistem mengembalikan informasi seperti berikut.

```
{
    "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"
}
```

Menampilkan waktu mulai untuk empat eksekusi berikutnya dari jendela pemeliharaan

Jalankan perintah berikut.

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"
```

Sistem mengembalikan informasi seperti berikut.

```
{
    "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: Melihat informasi tentang tugas dan eksekusi tugas (AWS CLI)

Tutorial ini mendemonstrasikan cara menggunakan AWS Command Line Interface (AWS CLI) untuk melihat detail tentang tugas jendela pemeliharaan Anda yang telah selesai.

Jika Anda melanjutkan langsung dari Tutorial: Membuat dan mengonfigurasi jendela pemeliharaan (AWS CLI), pastikan Anda telah memungkinkan cukup waktu bagi jendela pemeliharaan Anda untuk dijalankan setidaknya sekali untuk melihat hasil eksekusi.

Saat Anda mengikuti langkah di tutorial ini, ganti nilai di teks *merah* italik dengan pilihan dan ID Anda sendiri. Misalnya, ganti ID jendela pemeliharaan *mw-0c50858d01EXAMPLE* dan ID instans *i-02573cafcfEXAMPLE* dengan ID sumber daya yang Anda buat.

Untuk melihat informasi tentang tugas dan eksekusi tugas (AWS CLI)

1. Jalankan perintah berikut untuk melihat daftar eksekusi tugas untuk jendela pemeliharaan tertentu.

   Linux & macOS

   ```
   aws ssm describe-maintenance-window-executions \
       --window-id "mw-0c50858d01EXAMPLE"
   ```

   Windows

   ```
   aws ssm describe-maintenance-window-executions ^
       --window-id "mw-0c50858d01EXAMPLE"
   ```

Sistem mengembalikan informasi seperti berikut.

```
{
    "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. Jalankan perintah berikut untuk mendapatkan informasi tentang eksekusi tugas jendela pemeliharaan.

   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"
```

Sistem mengembalikan informasi seperti berikut.

```
{
    "WindowExecutionId": "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE",
    "TaskIds": [
        "c9b05aba-197f-4d8d-be34-e73fbEXAMPLE"
    ],
    "Status": "SUCCESS",
    "StartTime": 1557593493.096,
    "EndTime": 1557593498.611
}
```

3. Jalankan perintah berikut untuk mencantumkan tugas yang dijalankan sebagai bagian dari eksekusi jendela pemeliharaan.

   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"
   ```

   Sistem mengembalikan informasi seperti berikut.

   ```
   {
       "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. Jalankan perintah berikut untuk mendapatkan detail dari eksekusi tugas.

   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"
   ```

   Sistem mengembalikan informasi seperti berikut.

   ```
   {
       "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. Jalankan perintah berikut untuk mendapatkan permintaan tugas tertentu yang dilakukan untuk eksekusi tugas.

   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"
   ```

   Sistem mengembalikan informasi seperti berikut.

   ```
   {
       "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: Memperbarui jendela pemeliharaan (AWS CLI)

Tutorial ini mendemonstrasikan cara menggunakan AWS Command Line Interface (AWS CLI) untuk memperbarui jendela pemeliharaan. Di sini juga ditunjukkan cara memperbarui berbagai jenis tugas, termasuk untukAWS Systems Manager Run Commanddan Otomatisasi,AWS Lambda, danAWS Step Functions.

Contoh di bagian ini menggunakan tindakan Systems Manager berikut untuk memperbarui jendela pemeliharaan:

- UpdateMaintenanceWindow
- UpdateMaintenanceWindowTarget
- UpdateMaintenanceWindowTask
- DeregisterTargetFromMaintenanceWindow

Untuk informasi tentang penggunaan konsol Systems Manager untuk memperbarui jendela pemeliharaan, lihat Memperbarui atau menghapus sumber daya jendela pemeliharaan (konsol).

Saat Anda mengikuti langkah di tutorial ini, ganti nilai di teks *merah* italik dengan pilihan dan ID Anda sendiri. Misalnya, ganti ID jendela pemeliharaan *mw-0c50858d01EXAMPLE* dan ID instans *i-02573cafcfEXAMPLE* dengan ID sumber daya yang Anda buat.

Untuk memperbarui jendela pemeliharaan (AWS CLI)

1. Buka AWS CLI dan jalankan perintah berikut untuk memperbarui target agar menyertakan nama dan deskripsi.

   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"
```

Sistem mengembalikan informasi seperti berikut.

```
{
    "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. Jalankan perintah berikut untuk menggunakan pilihan `replace`untuk menghapus bidang deskripsi dan menambahkan target tambahan. Bidang deskripsi dihapus, karena pembaruan tidak mencakup bidang (nilai null). Pastikan untuk menentukan simpul tambahan yang telah dikonfigurasi untuk digunakan dengan 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
```

Sistem mengembalikan informasi seperti berikut.

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
    "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE",
    "Targets": [
        {
            "Key": "InstanceIds",
            "Values": [
                "i-02573cafcfEXAMPLE",
                "i-0471e04240EXAMPLE"
            ]
        }
    ],
    "Name": "My-Maintenance-Window-Target"
}
```

3. Pilihan `start-date` memungkinkan Anda untuk menunda aktivasi jendela pemeliharaan hingga tanggal yang ditentukan di masa depan. Pilihan `end-date` memungkinkan Anda untuk mengatur tanggal dan waktu di masa depan setelah jendela pemeliharaan tidak lagi berjalan. Tentukan pilihan dalam format ISO-8601 Extended.

   Jalankan perintah berikut untuk menentukan tanggal dan rentang waktu untuk eksekusi jendela pemeliharaan terjadwal secara reguler.

   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. Jalankan perintah berikut untuk memperbaruiRun Commandtugas.

> **Tip**
>
> Jika target Anda adalah instans Amazon Elastic Compute Cloud (Amazon EC2) untuk Windows Server, ubah `df` ke `ipconfig`, dan `AWS-RunShellScript` ke `AWS-RunPowerShellScript` di perintah berikut ini.

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"
```

Sistem mengembalikan informasi seperti berikut.

```
{
    "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. Adaptasi dan jalankan perintah berikut untuk memperbarui tugas 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"
```

Sistem mengembalikan informasi seperti berikut.

```
{
    "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. Jika Anda memperbarui tugas Step Functions, adaptasi dan jalankan perintah berikut untuk memperbarui tugas Step Functions task-invocation-parameters.

   Linux & macOS

   ```
   aws ssm update-maintenance-window-task \
       --window-id "mw-0c50858d01EXAMPLE" \
       --window-task-id "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE" \
       --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
    \
       --task-arn "arn:aws:states:region:execution:SSMStepFunctionTest" \
       --service-role-arn "arn:aws:iam:account-id:role/MaintenanceWindowsRole" \
       --task-invocation-parameters '{"StepFunctions":{"Input":"{\"InstanceId\":
   \"{{RESOURCE_ID}}\"}"}}' \
       --priority 0 --max-concurrency 10 --max-errors 5 \
       --name "My-Step-Functions-Task" \
       --description "A description for my Step Functions task"
   ```

   Windows

   ```
   aws ssm update-maintenance-window-task ^
       --window-id "mw-0c50858d01EXAMPLE" ^
       --window-task-id "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE" ^
       --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
    ^
       --task-arn "arn:aws:states:region:execution:SSMStepFunctionTest" ^
       --service-role-arn "arn:aws:iam:account-id:role/MaintenanceWindowsRole" ^
       --task-invocation-parameters '{"StepFunctions":{"Input":"{\"InstanceId\":
   \"{{RESOURCE_ID}}\"}"}}' ^
       --priority 0 --max-concurrency 10 --max-errors 5 ^
       --name "My-Step-Functions-Task" ^
       --description "A description for my Step Functions task"
   ```

   Sistem mengembalikan informasi seperti berikut.

```
{
    "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. Jalankan perintah berikut untuk membatalkan pendaftaran target dari jendela pemeliharaan. Contoh ini menggunakan parameter `safe` untuk menentukan apakah target direferensikan oleh tugas dan karenanya aman untuk dibatalkan pendaftarannya.

   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
```

Sistem mengembalikan informasi seperti berikut.

```
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. Jalankan perintah berikut untuk membatalkan pendaftaran target dari jendela pemeliharaan meski target direferensikan oleh tugas. Anda dapat memaksakan operasi pembatalan pendaftaran dengan menggunakan parameter `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
   ```

   Sistem mengembalikan informasi seperti berikut.

   ```
   {
       "WindowId": "mw-0c50858d01EXAMPLE",
       "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
   }
   ```

9. Jalankan perintah berikut untuk memperbaruiRun Commandtugas. Contoh ini menggunakan Systems ManagerParameter Storeparameter yang disebut`UpdateLevel`, yang diformat sebagai berikut: '`{{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}}'}}"
```

Sistem mengembalikan informasi seperti berikut.

```
{
    "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. Jalankan perintah berikut untuk memperbarui tugas Otomatisasi untuk menentukan parameter `WINDOW_ID` dan `WINDOW_TASK_ID` untuk parameter `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
    ```

Sistem mengembalikan informasi seperti berikut.

```
{
    "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: Menghapus jendela pemeliharaan (AWS CLI)

Untuk menghapus jendela pemeliharaan yang Anda buat di tutorial ini, jalankan perintah berikut.

```
aws ssm delete-maintenance-window --window-id "mw-0c50858d01EXAMPLE"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "WindowId":"mw-0c50858d01EXAMPLE"
}
```

Panduan jendela pemeliharaan

Panduan di bagian ini menunjukkan cara untuk membuat jendela pemeliharaan AWS Systems Manager menggunakan AWS Command Line Interface (AWS CLI) atau konsol Systems Manager kepada Anda. Jendela pemeliharaan yang Anda buat memperbaruiSSM Agentpada node terkelola.

Daftar Isi

Anda juga dapat melihat sampel perintah di Referensi AWS CLISystems Manager.

Panduan: Membuat jendela pemeliharaan untuk memperbaruiSSM Agent (AWS CLI)

Panduan berikut menunjukkan cara menggunakan AWS Command Line Interface (AWS CLI) untuk membuat jendela pemeliharaan AWS Systems Manager. Panduan juga menjelaskan cara mendaftarkan node terkelola Anda sebagai target dan mendaftarkanRun Command tugas Systems Manager untuk memperbaruiSSM Agent.

Sebelum Anda memulai

Sebelum menyelesaikan prosedur berikut, Anda harus memiliki izin administrator pada node yang ingin Anda konfigurasikan atau telah diberi izin yang sesuai diAWS Identity and Access Management (IAM). Sebelum Anda menyelesaikan prosedur berikut, pastikan Anda memiliki setidaknya satu node terkelola untuk Linux atauWindows Server yang dikonfigurasi untuk Systems Manager di lingkungan hibrid dan multicloud. Untuk informasi selengkapnya, lihat Menyiapkan AWS Systems Manager.

Topik

Langkah 1: Memulai

Untuk menjalankan perintah menggunakan AWS CLI

1. Instal dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaru dariAWS CLI.
2. Pastikan node siap untuk didaftarkan sebagai target untuk jendela pemeliharaan.

   Jalankan perintah berikut untuk melihat node mana yang online.

   ```
   aws ssm describe-instance-information --query "InstanceInformationList[*]"
   ```

   Jalankan perintah berikut untuk melihat detail tentang node tertentu.

   ```
   aws ssm describe-instance-information --instance-information-filter-list
    key=InstanceIds,valueSet=instance-id
   ```

Langkah 2: Membuat jendela pemeliharaan

Gunakan prosedur berikut untuk membuat jendela pemeliharaan dan menentukan pilihan dasarnya, seperti jadwal dan durasi.

Membuat jendela pemeliharaan (AWS CLI)

1. Buka AWS CLI dan jalankan perintah berikut untuk membuat jendela pemeliharaan yang berjalan setiap minggu pada hari Minggu pukul 02.00, di zona waktu Pasifik Amerika Serikat, dengan satu jam cutoff.

   Linux & macOS

   ```
   aws ssm create-maintenance-window \
   ```

```
    --name "My-First-Maintenance-Window" \
    --schedule "cron(0 2 ? * SUN *)" \
    --duration 2 \
    --schedule-timezone "America/Los_Angeles" \
    --cutoff 1 \
    --no-allow-unassociated-targets
```

Windows

```
aws ssm create-maintenance-window ^
    --name "My-First-Maintenance-Window" ^
    --schedule "cron(0 2 ? * SUN *)" ^
    --duration 2 ^
    --schedule-timezone "America/Los_Angeles" ^
    --cutoff 1 ^
    --no-allow-unassociated-targets
```

Untuk informasi lebih lanjut tentang cara membuat ekspresi cron untuk parameter `schedule`, lihat Referensi: Ekspresi cron dan rate untuk Systems Manager.

Untuk penjelasan tentang cara berbagai pilihan terkait jadwal untuk windows pemeliharaan berkaitan satu sama lain, lihat Penjadwalan jendela pemeliharaan dan pilihan periode aktif.

Untuk informasi lebih lanjut tentang penggunaan pilihan `--schedule`, lihat Referensi: Ekspresi cron dan rate untuk Systems Manager.

Sistem mengembalikan informasi seperti berikut.

```
{
   "WindowId":"mw-0c50858d01EXAMPLE"
}
```

2. Untuk mencantumkan ini dan setiap jendela pemeliharaan lainnya yang dibuat di Akun AWS di Wilayah AWS Anda saat ini, jalankan perintah berikut.

```
aws ssm describe-maintenance-windows
```

Sistem mengembalikan informasi seperti berikut.

```
{
```

```
    "WindowIdentities": [
        {
            "Cutoff": 1,
            "Name": "My-First-Maintenance-Window",
            "NextExecutionTime": "2019-02-03T02:00-08:00",
            "Enabled": true,
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Duration": 2
        }
    ]
}
```

Langkah 3: Mendaftarkan target jendela pemeliharaan (AWS CLI)

Gunakan prosedur berikut untuk mendaftarkan target dengan jendela pemeliharaan yang Anda buat di Langkah 2. Dengan mendaftarkan target, Anda menentukan node mana yang diperbarui.

Untuk mendaftarkan target jendela pemeliharaan (AWS CLI)

1. Jalankan perintah berikut. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

   Linux & macOS

   ```
   aws ssm register-target-with-maintenance-window \
       --window-id "mw-0c50858d01EXAMPLE" \
       --target "Key=InstanceIds,Values=i-02573cafcfEXAMPLE" \
       --resource-type "INSTANCE"
   ```

   Windows

   ```
   aws ssm register-target-with-maintenance-window ^
       --window-id "mw-0c50858d01EXAMPLE" ^
       --target "Key=InstanceIds,Values=i-02573cafcfEXAMPLE" ^
       --resource-type "INSTANCE"
   ```

   Sistem mengembalikan informasi seperti berikut, yang menyertakan ID target jendela pemeliharaan. Salin atau catat nilai `WindowTargetId`. Anda harus menentukan ID ini di langkah berikutnya untuk mendaftarkan tugas bagi jendela pemeliharaan ini.

```
{
   "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Perintah alternatif

Gunakan perintah berikut untuk mendaftarkan beberapa node terkelola.

Linux & macOS

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --targets "Key=InstanceIds,Values=i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE" \
    --resource-type "INSTANCE"
```

Windows

```
aws ssm register-target-with-maintenance-window ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --targets "Key=InstanceIds,Values=i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE" ^
    --resource-type "INSTANCE"
```

Gunakan perintah berikut untuk mendaftarkan node dengan menggunakan tag.

Linux & macOS

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --targets "Key=tag:Environment,Values=Prod" "Key=tag:Role,Values=Web" \
    --resource-type "INSTANCE"
```

Windows

```
aws ssm register-target-with-maintenance-window ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --targets "Key=tag:Environment,Values=Prod" "Key=tag:Role,Values=Web" ^
    --resource-type "INSTANCE"
```

2. Jalankan perintah berikut untuk menampilkan target untuk jendela pemeliharaan.

```
aws ssm describe-maintenance-window-targets --window-id "mw-0c50858d01EXAMPLE"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "Targets": [
        {
            "ResourceType": "INSTANCE",
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Targets": [
                {
                    "Values": [
                        "i-02573cafcfEXAMPLE"
                    ],
                    "Key": "InstanceIds"
                }
            ],
            "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
        },
        {
            "ResourceType": "INSTANCE",
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Targets": [
                {
                    "Values": [
                        "Prod"
                    ],
                    "Key": "tag:Environment"
                },
                {
                    "Values": [
                        "Web"
                    ],
                    "Key": "tag:Role"
                }
            ],
            "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
        }
    ]
}
```

Langkah 4: MendaftarkanRun Command tugas agar jendela pemeliharaan diperbaruiSSM Agent

Gunakan prosedur berikut untuk mendaftarkanRun Command tugas untuk jendela pemeliharaan yang Anda buat di Langkah 2. Run CommandTugas memperbaruiSSM Agent target terdaftar.

Untuk mendaftarkanRun Command tugas agar jendela pemeliharaan memperbaruiSSM Agent (AWS CLI)

1. Jalankan perintah berikut untuk mendaftarkanRun Command tugas untuk jendela pemeliharaan menggunakan`WindowTargetId` nilai di Langkah 3. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri. Pembaruan tugasSSM Agent dengan menggunakan`AWS-UpdateSSMAgent` dokumen.

   Linux & macOS

   ```
   aws ssm register-task-with-maintenance-window \
       --window-id "mw-0c50858d01EXAMPLE" \
       --task-arn "AWS-UpdateSSMAgent" \
       --name "UpdateSSMAgent" \
       --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
    \
       --service-role-arn "arn:aws:iam:account-id:role/MW-Role" \
       --task-type "RUN_COMMAND" \
       --max-concurrency 1 --max-errors 1 --priority 10
   ```

   Windows

   ```
   aws ssm register-task-with-maintenance-window ^
       --window-id "mw-0c50858d01EXAMPLE" ^
       --task-arn "AWS-UpdateSSMAgent" ^
       --name "UpdateSSMAgent" ^
       --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
    ^
       --service-role-arn "arn:aws:iam:account-id:role/MW-Role" ^
       --task-type "RUN_COMMAND" ^
       --max-concurrency 1 --max-errors 1 --priority 10
   ```

> **Note**
>
> Jika target yang Anda daftarkan pada langkah sebelumnya adalahWindows Server 2012 R2 atau yang lebih lama, Anda harus menggunakan dokumen `AWS-UpdateEC2Config`.

Sistem mengembalikan informasi seperti berikut.

```
{
   "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
}
```

2. Jalankan perintah berikut untuk mencantumkan semua tugas terdaftar untuk jendela pemeliharaan.

```
aws ssm describe-maintenance-window-tasks --window-id "mw-0c50858d01EXAMPLE"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "Tasks": [
        {
            "ServiceRoleArn": "arn:aws:iam::111122223333:role/MW-Role",
            "MaxErrors": "1",
            "TaskArn": "AWS-UpdateSSMAgent",
            "MaxConcurrency": "1",
            "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
            "TaskParameters": {},
            "Priority": 10,
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Type": "RUN_COMMAND",
            "Targets": [
                {
                    "Values": [
                        "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
                    ],
                    "Key": "WindowTargetIds"
                }
            ],
```

```
            "Name": "UpdateSSMAgent"
        }
    ]
}
```

Panduan: Membuat jendela pemeliharaan untuk memperbarui secara otomatisSSM Agent (konsol)

Panduan berikut menunjukkan cara menggunakan konsol AWS Systems Manager untuk membuat jendela pemeliharaan . Panduan juga menjelaskan cara mendaftarkan node terkelola Anda sebagai target dan mendaftarkanRun Command tugas Systems Manager untuk diperbaruiSSM Agent.

Sebelum Anda memulai

Sebelum menyelesaikan prosedur berikut, Anda harus memiliki izin administrator pada node yang ingin Anda konfigurasikan atau telah diberi izin yang sesuai diAWS Identity and Access Management (IAM). Sebelum Anda menyelesaikan prosedur berikut, pastikan Anda setidaknya memiliki satu node terkelola untuk Linux atauWindows Server di lingkungan hibrid dan multicloud yang dikonfigurasi untuk Systems Manager. Untuk informasi selengkapnya, lihat Menyiapkan AWS Systems Manager.

Topik

- Langkah 1: Membuat jendela pemeliharaan (konsol)
- Langkah 2: Mendaftarkan target jendela pemeliharaan (konsol)
- Langkah 3: MendaftarkanRun Command tugas Run un un un un un un un un un un un un un un un un un un un unSSM Agent un un un un un un

Langkah 1: Membuat jendela pemeliharaan (konsol)

Untuk membuat jendela pemeliharaan (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Maintenance Windows.
3. Pilih Buat jendela pemeliharaan.
4. Untuk Nama, masukkan nama deskriptif untuk membantu Anda mengidentifikasi jendela pemeliharaan ini.
5. (Opsional) Untuk Deskripsi, masukkan deskripsi.

6. Pilih Izinkan target tidak terdaftar jika Anda ingin mengizinkan tugas jendela pemeliharaan berjalan pada node terkelola, meski Anda belum mendaftarkan node tersebut sebagai target. Jika Anda memilih opsi ini, maka Anda dapat memilih node yang tidak terdaftar (menurut ID node) saat mendaftarkan tugas dengan jendela pemeliharaan.

   Jika Anda tidak memilih pilihan ini, maka Anda harus memilih target yang terdaftar sebelumnya saat mendaftarkan tugas dengan jendela pemeliharaan.

7. Tentukan jadwal untuk jendela pemeliharaan dengan menggunakan salah satu dari tiga pilihan penjadwalan.

   Untuk informasi tentang pembangunan ekspresi cron/rate, lihat Referensi: Ekspresi cron dan rate untuk Systems Manager.

8. Untuk Durasi, masukkan jumlah jam yang harus dijalankan oleh jendela pemeliharaan.

9. Untuk Berhenti memulai tugas, masukkan jumlah jam sebelum akhir jendela pemeliharaan di mana sistem harus berhenti menjadwalkan tugas baru untuk dijalankan.

10. (Opsional) Untuk Tanggal mulai jendela - opsional, tentukan tanggal dan waktu, dalam format ISO-8601 Extended, jika Anda ingin menjadikan jendela pemeliharaan aktif. Ini memungkinkan Anda menunda aktivasi jendela pemeliharaan hingga tanggal yang ditentukan di masa depan.

11. (Opsional) Untuk Tanggal selesai jendela - opsional, tentukan tanggal dan waktu, dalam format ISO-8601 Extended, jika Anda ingin menjadikan jendela pemeliharaan tidak aktif. Hal ini memungkinkan Anda untuk mengatur tanggal dan waktu di masa depan setelah jendela pemeliharaan tidak lagi berjalan.

12. (Opsional) Untuk Zona waktu jadwal - opsional, tentukan zona waktu untuk eksekusi jendela pemeliharaan terjadwal dasar, dalam format Internet Assigned Numbers Authority (IANA). Sebagai contoh: "Amerika/Los_Angeles", "etc/UTC", atau "Asia/Seoul".

    Untuk informasi lebih lanjut tentang format yang valid, lihat Basis Data Zona Waktu di situs web IANA.

13. (Opsional) Di area Kelola tanda, terapkan satu atau beberapa pasangan nama/nilai kunci tanda ke jendela pemeliharaan.

    Tanda adalah metadata opsional yang Anda tetapkan ke sumber daya. Tag memungkinkan Anda untuk mengkategorikan sumber daya dengan berbagai cara, seperti berdasarkan tujuan, pemilik, atau lingkungan. Misalnya, Anda mungkin ingin menandai jendela pemeliharaan untuk mengidentifikasi jenis tugas yang dijalankannya, jenis target, dan lingkungan tempat ia berjalan. Dalam hal ini, Anda dapat menentukan pasangan nama/nilai kunci berikut:

- `Key=TaskType,Value=AgentUpdate`
- `Key=OS,Value=Windows`
- `Key=Environment,Value=Production`

14. Pilih Buat jendela pemeliharaan. Sistem mengembalikan Anda ke halaman jendela pemeliharaan. Jendela pemeliharaan yang baru saja Anda buat berada di tahapan Diaktifkan.

Langkah 2: Mendaftarkan target jendela pemeliharaan (konsol)

Gunakan prosedur berikut untuk mendaftarkan target dengan jendela pemeliharaan yang Anda buat di Langkah 1. Dengan mendaftarkan target, Anda menentukan node mana yang diperbarui.

Untuk menetapkan target ke jendela pemeliharaan (konsol)

1. Di daftar windows pemeliharaan, pilih jendela pemeliharaan yang baru Anda buat.
2. Pilih Tindakan, lalu pilih Daftarkan target.
3. (Opsional) Untuk Nama target, masukkan nama untuk target.
4. (Opsional) Untuk Deskripsi, masukkan deskripsi.
5. (Opsional) Untuk Informasi pemilik, tentukan nama anda atau alias di tempat kerja. Informasi pemilik disertakan dalamEventBridge peristiwa Amazon yang dimunculkan selagi menjalankan tugas untuk target ini di jendela pemeliharaan ini.

   Untuk informasi tentang menggunakanEventBridge untuk memantau peristiwa Systems Manager, lihatPemantauan peristiwa Systems Manager dengan Amazon EventBridge.
6. Di area Target, pilih salah satu pilihan yang dijelaskan di tabel berikut.

Opsi	Deskripsi
Tentukan tag instance	Untuk kotak Tentukan instans tag, tentukan satu atau beberapa kunci tanda dan nilai (opsional) tanda yang telah atau akan ditambahkan ke node terkelola di akun Anda. Ketika jendela pemeliharaan berjalan, ia mencoba untuk melakukan tugas pada semua node terkelola yang telah ditambahi tanda ini.

Opsi	Deskripsi
	Jika Anda menentukan lebih dari satu kunci tanda, simpul harus ditandai dengan semua kunci tanda dan nilai yang Anda tentukan untuk disertakan dalam grup target.
Pilih node secara manual	Dari daftar, pilih kotak untuk setiap node yang ingin Anda sertakan di target jendela pemeliharaan. Daftar ini menyertakan semua node di akun Anda yang dikonfigurasi untuk digunakan dengan Systems Manager. Jika node terkelola yang Anda harapkan tidak tercantum, lihatMemecahkan masalah ketersediaan node terkelola untuk kiat pemecahan masalah. Untuk perangkat edge server on-premise, dan mesin virtual (VM), lihatMenyiapkan Manajer Sistem untuk lingkungan hybrid dan multicloud

Opsi	Deskripsi
Pilih grup sumber daya	Untuk Grup sumber daya, pilih nama grup sumber daya yang ada di akun Anda dari daftar. Untuk informasi tentang pembuatan dan penggunaan grup sumber daya, lihat topik berikut: • Apa itu kelompok sumber daya? di AWS Resource GroupsUser Guide • Resource Groups dan Penandaan untukAWS di Blog Berita AWS Untuk Jenis sumber daya, pilih hingga lima jenis sumber daya yang tersedia, atau pilih Semua jenis sumber daya. Jika tugas yang Anda tetapkan ke jendela pemeliharaan tidak berfungsi pada salah satu jenis sumber daya yang ditambahkan ke target, sistem mungkin akan melaporkan kesalahan. Tugas di mana jenis sumber daya yang didukung ditemukan akan terus berjalan meskipun ada kesalahan ini. Misalnya, anggap saja Anda menambahkan jenis sumber daya berikut ke target ini: • `AWS::S3::Bucket` • `AWS::DynamoDB::Table` • `AWS::EC2::Instance` Tetapi, nanti saat Anda menambahkan tugas ke jendela pemeliharaan, Anda menyertakan hanya tugas yang melakukan tindakan

Opsi	Deskripsi
	pada node, seperti penerapan dasar patch atau but ulang node. Di log jendela pemeliharaan, kesalahan mungkin dilaporkan karena tidak ada bucket Amazon Simple Storage Service (Amazon S3) atau tabel Amazon DynamoDB yang ditemukan. Akan tetapi, jendela pemeliharaan masih menjalankan tugas pada node di grup sumber daya Anda.

7. Pilih Daftarkan target.

Langkah 3: MendaftarkanRun Command tugas Run un un un un un un un un un un un un un un un un un un un unSSM Agent un un un un un un

Gunakan prosedur berikut untuk mendaftarkanRun Command tugas untuk jendela pemeliharaan yang Anda buat di Langkah 1. Run CommandTugas memperbaruiSSM Agent target terdaftar.

Untuk menetapkan tugas ke jendela pemeliharaan (konsol)

1. Di daftar windows pemeliharaan, pilih jendela pemeliharaan yang baru Anda buat.
2. Pilih Tindakan, lalu pilih Register Run command task.
3. (Opsional) Untuk Nama, masukkan nama untuk tugas, seperti UpdateSSMAgent.
4. (Opsional) Untuk Deskripsi, masukkan deskripsi.
5. Di area dokumen Command, pilih dokumen SSM Command`AWS-UpdateSSMAgent`.

> **Note**
>
> Jika target yang Anda daftarkan pada langkah sebelumnya adalahWindows Server 2012 R2 atau yang lebih lama, Anda harus menggunakan dokumen `AWS-UpdateEC2Config`.

6. Untuk Versi dokumen, pilih versi dokumen yang digunakan.
7. Untuk Prioritas tugas, tentukan prioritas untuk tugas ini. Nol (`0`) adalah prioritas tertinggi. Tugas di jendela pemeliharaan dijadwalkan dalam urutan prioritas dengan tugas yang memiliki prioritas yang sama dijadwalkan secara paralel.

8. Di bagian Target, identifikasi node tempat Anda ingin menjalankan operasi ini dengan memilih Memilih grup target terdaftar atau Memilih target yang tidak terdaftar.
9. Untuk Pengendalian rate:
   - Untuk Konkurensi, tentukan jumlah atau persentase node terkelola untuk menjalankan perintah pada waktu yang sama.

     > **Note**
     >
     > Jika Anda memilih target dengan menentukan tag diterapkan ke node terkelola atau dengan menentukanAWS sumber daya grup, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada waktu yang sama dengan menentukan persentase.

   - Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada node terkelola lainnya setelah gagal pada sejumlah atau persentase node. Misalnya, jika Anda menentukan tiga kesalahan, maka Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Simpul terkelola yang masih memroses perintah juga dapat mengirim kesalahan.
10. Untuk Peran layanan IAM, pilih peran untuk memberikan izin bagi Systems Manager untuk menjalankan tugas jendela pemeliharaan.

    Jika Anda perlu membuat peran layanan kustom untuk tugas jendela pemeliharaan, lihatGunakan konsol untuk mengonfigurasi izin untuk jendela pemeliharaan.
11. (Opsional) Untuk Opsi output, lakukan salah satu hal berikut:
    - Pilih kotak centang Aktifkan penulisan ke S3 untuk menyimpan output perintah ke file. Masukkan nama bucket dan prefiks (folder) di dalam kotak.

      > **Note**
      >
      > Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah dari profil instans yang ditetapkan ke node, bukan data pengguna yang melaksanakan tugas ini. Untuk informasi lebih lanjut, lihat Mengkonfigurasi izin instans untuk Systems Manager. Selain itu, jika bucket S3 yang ditentukan berada dalam yang berbedaAkun AWS, verifikasi bahwa profil instans yang terkait dengan node memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

- Pilih kotak centang CloudWatchkeluaran untuk menulis output lengkap ke AmazonCloudWatch Logs. Masukkan nama grupCloudWatch log log log.

12. Di bagian Notifikasi SNS, Anda dapat secara opsional mengizinkan Systems Manager untuk mengirimkan notifikasi tentang status perintah menggunakan Amazon Simple Notification Service (Amazon SNS). Jika Anda memilih untuk mengaktifkan opsi ini, Anda perlu menentukan hal berikut:

    a. IAM role untuk memulai notifikasi Amazon SNS.

    b. Topik Amazon SNS untuk digunakan.

    c. Jenis acara tertentu tentang hal mana yang ingin diberitahukan kepada Anda.

    d. Jenis notifikasi yang ingin Anda terima saat status perintah berubah. Untuk perintah yang dikirimkan ke beberapa node, pilih Permintaan untuk menerima notifikasi pada basis permintaan (per-node) saat status setiap permintaan berubah.

13. Di area Parameter, Anda dapat secara opsional menyediakan versi tertentuSSM Agent untuk menginstal, atau Anda dapat mengizinkanSSM Agent layanan untuk diturunkan ke versi yang lebih lama. Akan tetapi, untuk panduan ini kami tidak menyediakan versi. Oleh karenaSSM Agent itu, diperbarui ke versi terbaru.

14. Pilih Mendaftarkan tugas Run un command

Walkthrough: Membuat jendela pemeliharaan untuk patching (konsol)

> **Important**
>
> Anda dapat terus menggunakan topik warisan ini untuk membuat jendela pemeliharaan untuk patching. Namun, kami menyarankan Anda menggunakan kebijakan patch. Untuk informasi selengkapnya, lihat Menggunakan kebijakan Quick Setup tambalan dan Patch Managerkonfigurasi penambalan organisasi.

Untuk meminimalkan dampak pada ketersediaan server Anda, kami merekomendasikan Anda mengkonfigurasi jendela pemeliharaan untuk menjalankan patching pada waktu yang tidak akan mengganggu operasi bisnis Anda. Untuk informasi lebih lanjut tentang jendela pemeliharaan, lihat AWS Systems Manager Maintenance Windows.

Anda harus mengkonfigurasi peran dan izin untukMaintenance Windows, suatu kemampuanAWS Systems Manager, sebelum memulai prosedur ini. Untuk informasi selengkapnya, lihat Menyiapkan Maintenance Windows.

Untuk membuat jendela pemeliharaan untuk patching

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Maintenance Windows.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Maintenance Windows.
3. Pilih Buat jendela pemeliharaan.
4. Untuk Nama, masukkan nama yang menunjuk ini sebagai jendela pemeliharaan untuk melakukan patching pada pembaruan kritis dan penting.
5. Untuk Deskripsi, masukkan deskripsi.
6. Pilih Izinkan target tidak terdaftar jika Anda ingin mengizinkan tugas jendela pemeliharaan berjalan pada node terkelola, meski Anda belum mendaftarkan node tersebut sebagai target. Jika Anda memilih opsi ini, maka Anda dapat memilih node yang tidak terdaftar (menurut ID node) saat Anda mendaftarkan tugas dengan jendela pemeliharaan.

   Jika Anda tidak memilih pilihan ini, maka Anda harus memilih target yang terdaftar sebelumnya saat mendaftarkan tugas dengan jendela pemeliharaan.
7. Di bagian atas bagian Jadwal, tentukan jadwal untuk jendela pemeliharaan dengan menggunakan salah satu dari tiga opsi penjadwalan.

   Untuk informasi tentang pembangunan ekspresi cron/rate, lihat Referensi: Ekspresi cron dan rate untuk Systems Manager.
8. Untuk Durasi, masukkan jumlah jam yang akan dijalankan oleh jendela pemeliharaan. Nilai yang Anda tentukan menentukan waktu selesai tertentu untuk jendela pemeliharaan berdasarkan waktu dimulainya. Tidak ada tugas jendela pemeliharaan yang diizinkan untuk memulai setelah waktu selesai dikurangi jumlah jam yang Anda tentukan untuk Berhenti memulai tugas di langkah berikutnya.

Sebagai contoh, jika jendela pemeliharaan dimulai pada pukul 3 sore, durasinya tiga jam, dan nilai Berhenti memulai tugas adalah satu jam, tidak ada tugas jendela pemeliharaan yang dapat memulai setelah pukul 5 sore.

9. Untuk Berhenti memulai tugas, masukkan jumlah jam sebelum akhir jendela pemeliharaan dimana sistem harus berhenti menjadwalkan tugas baru untuk dijalankan.
10. (Opsional) Untuk Tanggal mulai (opsional), tentukan tanggal dan waktu, dalam format ISO-8601 Extended, jika Anda ingin menjadikan jendela pemeliharaan aktif. Ini memungkinkan Anda menunda aktivasi jendela pemeliharaan hingga tanggal yang ditentukan di masa depan.
11. (Opsional) Untuk Tanggal selesai (opsional), tentukan tanggal dan waktu, dalam format ISO-8601 Extended, jika Anda ingin menjadikan jendela pemeliharaan tidak aktif. Hal ini memungkinkan Anda untuk mengatur tanggal dan waktu di masa depan setelah jendela pemeliharaan tidak lagi berjalan.
12. (Opsional) Untuk Zona waktu (opsional), tentukan zona waktu untuk mendasari eksekusi jendela pemeliharaan terjadwal, dalam format Internet Assigned Numbers Authority (IANA). Sebagai contoh: "Amerika/Los_Angeles", "etc/UTC", atau "Asia/Seoul".

    Untuk informasi lebih lanjut tentang format yang valid, lihat Basis Data Zona Waktu di situs web IANA.
13. Pilih Buat jendela pemeliharaan.
14. Dalam daftar jendela pemeliharaan, pilih jendela pemeliharaan yang baru saja Anda buat, lalu pilih Tindakan, Daftarkan target.
15. (Opsional) Di bagian Detail target jendela pemeliharaan, berikan nama, deskripsi, dan informasi pemilik (nama atau alias Anda) untuk target ini.
16. Untuk Target, pilih Menentukan tag instans.
17. Untuk Tag instans, masukkan kunci tag dan nilai tag untuk mengidentifikasi node yang akan didaftarkan dengan jendela pemeliharaan, lalu pilih Tambahkan.
18. Pilih Daftarkan target. Sistem ini membuat target jendela pemeliharaan.
19. Di halaman detail jendela pemeliharaan yang Anda buat, pilih Tindakan, Daftarkan tugas Run Command.
20. (Opsional) Untuk Detail tugas jendela pemeliharaan, berikan nama dan deskripsi untuk tugas ini.
21. Untuk Dokumen perintah, pilih `AWS-RunPatchBaseline`.
22. Untuk Prioritas tugas, pilih prioritas. Nol (`0`) adalah prioritas tertinggi.

23. Untuk Target, di bawah Target berdasarkan, pilih target jendela pemeliharaan yang Anda buat sebelumnya dalam prosedur ini.
24. Untuk Pengendalian rate:
    - Untuk Konkurensi, tentukan jumlah atau persentase dari node terkelola untuk menjalankan perintah pada saat yang sama.

      > **Note**
      >
      > Jika Anda memilih target dengan menentukan tag diterapkan ke node terkelola atau dengan menentukanAWS sumber daya grup, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada waktu yang sama dengan menentukan persentase.

    - Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada node terkelola lainnya setelah gagal pada sejumlah atau persentase node. Misalnya, jika Anda menentukan tiga kesalahan, maka Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memroses perintah juga dapat mengirim kesalahan.
25. Untuk Peran layanan IAM, pilih peran untuk memberikan izin bagi Systems Manager untuk menjalankan tugas jendela pemeliharaan.

    Jika Anda perlu membuat peran layanan kustom untuk tugas jendela pemeliharaan, lihatGunakan konsol untuk mengonfigurasi izin untuk jendela pemeliharaan.
26. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Aktifkan output penulisan ke S3. Masukkan nama bucket dan prefiks (folder) di dalam kotak.

    > **Note**
    >
    > Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah dari profil instans yang ditetapkan ke node terkelola, bukan data pengguna IAM yang melaksanakan tugas ini. Untuk informasi selengkapnya, lihat Mengkonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berada dalam yang berbedaAkun AWS, pastikan profil instans atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

Untuk mengalirkan output ke grup CloudWatch log Amazon Logs, pilih kotak CloudWatch output. Masukkan nama grup log di kotak.

27. Di bagian Notifikasi SNS, jika Anda menginginkan notifikasi tentang status eksekusi perintah dikirimkan, pilih kotak centang Mengaktifkan notifikasi SNS.

    Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Amazon SNSRun Command, lihatPemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS.

28. Untuk Parameter:

    - Untuk Operasi, pilih Pindai untuk memindai patch yang hilang, atau pilih Instal untuk memindai dan menginstal patch yang hilang.
    - Anda tidak perlu memasukkan apa pun di bidang Snapshot Id. Sistem ini secara otomatis membuat dan menyediakan parameter ini.
    - Anda tidak perlu memasukkan apa pun di bidang Instal Override List kecuali jika AndaPatch Manager ingin menggunakan patch yang berbeda dari yang ditentukan untuk dasar patch. Untuk informasi, lihat Nama parameter: `InstallOverrideList`.
    - Untuk Opsi reboot, tentukan apakah Anda ingin node di-reboot jika patch diinstal selama`Install` operasi, atau jikaPatch Manager mendeteksi patch lain yang diinstal sejak reboot node terakhir. Untuk informasi, lihat Nama parameter: `RebootOption`.
    - (Opsional) Untuk Komentar, masukkan catatan pelacakan atau pengingat tentang perintah ini.
    - Untuk Batas waktu (detik), masukkan jumlah detik sistem harus menunggu operasi selesai sebelum dianggap tidak berhasil.

29. Pilih Mendaftarkan tugas Run Command.

Setelah tugas jendela pemeliharaan selesai, Anda dapat melihat detail kepatuhan patch di konsol Systems Manager pada halaman Instans Terkelola. Di bilah filter, gunakan filter `AWS:PatchSummary` dan `AWS:PatchCompliance`.

> **Note**
>
> Anda dapat menyimpan kueri Anda dengan membuat bookmark URL setelah menentukan filter.

Anda juga dapat menelusuri node tertentu dengan memilih node di halaman Instans Terkelola, lalu memilih tab Patch. Anda juga dapat menggunakan DescribePatchGroupStatedan DescribeInstancePatchStatesForPatchGroupAPI untuk melihat detail kepatuhan. Untuk informasi tentang data kepatuhan patch, lihat Tentang kepatuhan patch.

Tentang jadwal patching menggunakan jendela pemeliharaan

Setelah Anda mengkonfigurasi dasar patch (dan secara opsional, grup patch), Anda dapat menerapkan patch ke node Anda dengan menggunakan jendela pemeliharaan. Jendela pemeliharaan dapat mengurangi dampak pada ketersediaan server dengan membiarkan Anda menentukan waktu untuk melakukan proses patching yang tidak mengganggu operasi bisnis. Jendela pemeliharaan bekerja seperti ini:

1. Buat jendela pemeliharaan dengan sebuah jadwal untuk operasi patching Anda.
2. Pilih target untuk jendela pemeliharaan dengan menentukan`Patch Group` atau`PatchGroup` tag untuk nama tag, dan setiap nilai yang Anda telah berikan tag Amazon Elastic Compute Cloud (Amazon EC2), misalnya, “server web” atau “US-EAST-PROD. (Anda harus menggunakan`PatchGroup`, tanpa spasi, jika Anda telah mengizinkan tag dalam metadata instans EC2.
3. Buat tugas jendela pemeliharaan baru, dan tentukan dokumen `AWS-RunPatchBaseline`.

Ketika Anda mengkonfigurasi tugas, Anda dapat memilih untuk memindai instans atau memindai dan memindai dan memindai dan memindai dan memindai dan memindai dan menginstal patch pada node. Jika Anda memilih untuk memindai instansPatch Manager,AWS Systems Manager, suatu kemampuan, memindai setiap node dan membuat daftar patch yang hilang untuk Anda tinjau.

Jika Anda memilih untuk memindai dan menginstal patch,Patch Manager memindai setiap node dan membandingkan daftar patch yang diinstal terhadap daftar patch yang disetujui di baseline. Patch Managermengidentifikasi patch yang hilang, lalu mengunduh dan menginstal semua patch yang hilang, lalu mengunduh dan menginstal semua patch yang hilang dan disetujui.

Jika Anda ingin melakukan pemindaian satu kali atau menginstal untuk memperbaiki masalah, Anda dapat menggunakannyaRun Command untuk memanggil`AWS-RunPatchBaseline` dokumen secara langsung.

Important

Setelah menginstal patch, Systems Manager me-reboot setiap node. Reboot diperlukan untuk memastikan patch diinstal dengan benar dan untuk memastikan bahwa sistem tidak meninggalkan node dalam keadaan yang berpotensi buruk. (Pengecualian: Jika`RebootOption` parameter diatur ke`NoReboot` dalam`AWS-RunPatchBaseline` dokumen, node yang dikelola tidak di-reboot setelah Patch Manager dijalankan. Untuk informasi selengkapnya, lihatNama parameter: `RebootOption`.)

Menggunakan parameter semu saat mendaftarkan tugas jendela pemeliharaan

Ketika Anda mendaftarkan tugas di Maintenance Windows, kemampuan AWS Systems Manager, Anda menentukan parameter yang unik untuk masing-masing dari empat jenis tugas. (Dalam perintah CLI, ini disediakan menggunakan `--task-invocation-parameters` opsi.)

Anda juga dapat mereferensikan nilai tertentu menggunakan sintaks parameter semu, seperti `{{RESOURCE_ID}}`, `{{TARGET_TYPE}}`, dan `{{WINDOW_TARGET_ID}}`. Ketika tugas jendela pemeliharaan berjalan, ia meneruskan nilai yang benar dan bukan placeholder parameter semu. Daftar lengkap parameter semu yang dapat Anda gunakan disediakan nanti dalam topik ini diParameter semu yang didukung.

Important

Untuk jenis target `RESOURCE_GROUP`, tergantung dari format ID yang diperlukan untuk tugas, Anda dapat memilih antara penggunaan `{{TARGET_ID}}` dan `{{RESOURCE_ID}}`untuk mereferensikan sumber daya saat tugas berjalan. `{{TARGET_ID}}`menampilkan ARN lengkap dari sumber daya. `{{RESOURCE_ID}}` menampilkan hanya nama atau ID dari sumber daya yang lebih singkat, seperti yang ditunjukkan dalam contoh ini.

- Format `{{TARGET_ID}}`: `arn:aws:ec2:us-east-1:123456789012:instance/i-02573cafcfEXAMPLE`
- Format `{{RESOURCE_ID}}`: `i-02573cafcfEXAMPLE`

Untuk jenis target `INSTANCE`, baik parameter `{{TARGET_ID}}` maupun `{{RESOURCE_ID}}` menghasilkan ID instans saja. Untuk informasi selengkapnya, lihat Parameter semu yang didukung.

`{{TARGET_ID}}` dan `{{RESOURCE_ID}}` dapat digunakan untuk meneruskan ID dari sumber daya AWS ke tugas Otomatisasi, Lambda, dan Step Functions. Kedua parameter semu ini tidak dapat digunakan dengan Run Command tugas.

Contoh parameter semu

Anggap saja muatan Anda untuk tugas AWS Lambda perlu mereferensikan instans berdasarkan ID-nya.

Baik Anda menggunakan target jendela `RESOURCE_GROUP` pemeliharaan `INSTANCE` atau pemeliharaan, ini dapat dicapai dengan menggunakan parameter `{{RESOURCE_ID}}` semu. Sebagai contoh:

```
"TaskArn": "arn:aws:lambda:us-east-2:111122223333:function:SSMTestFunction",
    "TaskType": "LAMBDA",
    "TaskInvocationParameters": {
        "Lambda": {
            "ClientContext": "ew0KICAi--truncated--0KIEXAMPLE",
            "Payload": "{ \"instanceId\": \"{{RESOURCE_ID}}\" }",
            "Qualifier": "$LATEST"
        }
    }
```

Jika tugas Lambda Anda dimaksudkan untuk dijalankan terhadap jenis target yang didukung lainnya selain instans Amazon Elastic Compute Cloud (Amazon EC2), seperti tabel Amazon DynamoDB, sintaks yang sama dapat digunakan, dan `{{RESOURCE_ID}}` menghasilkan nama tabel saja. Akan tetapi, jika Anda memerlukan ARN lengkap dari tabel, gunakan `{{TARGET_ID}}`, seperti yang ditunjukkan dalam contoh berikut.

```
"TaskArn": "arn:aws:lambda:us-east-2:111122223333:function:SSMTestFunction",
    "TaskType": "LAMBDA",
    "TaskInvocationParameters": {
        "Lambda": {
            "ClientContext": "ew0KICAi--truncated--0KIEXAMPLE",
```

```
            "Payload": "{ \"tableArn\": \"{{TARGET_ID}}\" }",
            "Qualifier": "$LATEST"
        }
    }
```

Sintaks yang sama berfungsi untuk penargetan instans atau jenis sumber daya lainnya. Ketika beberapa jenis sumber daya telah ditambahkan ke grup sumber daya, tugas berjalan terhadap masing-masing sumber daya yang sesuai.

> **Important**
>
> Tidak semua jenis sumber daya yang mungkin disertakan di grup sumber daya menghasilkan nilai untuk parameter `{{RESOURCE_ID}}`. Untuk daftar jenis sumber daya yang didukung, lihat Parameter semu yang didukung.

Sebagai contoh lain, untuk menjalankan tugas Otomatisasi yang menghentikan instans EC2, Anda menentukan dokumen Systems Manager `AWS-StopEC2Instance` (dokumen SSM) sebagai nilai `TaskArn` dan menggunakan parameter semu `{{RESOURCE_ID}}`:

```
"TaskArn": "AWS-StopEC2Instance",
    "TaskType": "AUTOMATION"
    "TaskInvocationParameters": {
        "Automation": {
            "DocumentVersion": "1",
            "Parameters": {
                "instanceId": [
                    "{{RESOURCE_ID}}"
                ]
            }
        }
    }
```

Untuk menjalankan tugas Otomatisasi yang menyalin snapshot volume Amazon Elastic Block Store (Amazon EBS), Anda menentukan dokumen SSM `AWS-CopySnapshot` sebagai nilai `TaskArn` dan menggunakan parameter semu `{{RESOURCE_ID}}`.

```
"TaskArn": "AWS-CopySnapshot",
    "TaskType": "AUTOMATION"
    "TaskInvocationParameters": {
```

```
        "Automation": {
            "DocumentVersion": "1",
            "Parameters": {
                "SourceRegion": "us-east-2",
                "targetType":"RESOURCE_GROUP",
                "SnapshotId": [
                    "{{RESOURCE_ID}}"
                ]
            }
        }
    }
```

Parameter semu yang didukung

Daftar berikut ini menjelaskan parameter semu yang dapat Anda tentukan menggunakan sintaks {{*PSEUDO_PARAMETER*}} di pilihan `--task-invocation-parameters`.

- **`WINDOW_ID`**: ID dari jendela pemeliharaan target.
- **`WINDOW_TASK_ID`**: ID tugas jendela yang sedang berjalan.
- **`WINDOW_TARGET_ID`**: ID dari target jendela yang menyertakan target (ID target).
- **`WINDOW_EXECUTION_ID`**: ID dari eksekusi jendela saat ini.
- **`TASK_EXECUTION_ID`**: ID dari eksekusi tugas saat ini.
- **`INVOCATION_ID`**: ID dari permintaan saat ini.
- **`TARGET_TYPE`**: Jenis target. Jenis yang didukung termasuk `RESOURCE_GROUP` dan `INSTANCE`.
- **`TARGET_ID`**:

  Jika jenis target yang Anda tentukan adalah`INSTANCE`, parameter `TARGET_ID` semu diganti dengan ID instance. Misalnya, `i-078a280217EXAMPLE`.

  Jika jenis target yang Anda tentukan adalah`RESOURCE_GROUP`, nilai yang direferensikan untuk eksekusi tugas adalah ARN penuh dari sumber daya. Misalnya: `arn:aws:ec2:us-east-1:123456789012:instance/i-078a280217EXAMPLE`. Tabel berikut menyediakan sampel nilai `TARGET_ID` untuk jenis sumber daya tertentu dalam grup sumber daya.

> **Note**
>
> `TARGET_ID`tidak didukung untuk Run Command tugas.

Jenis sumber daya	Contoh TARGET_ID	
AWS::CloudWatch::Alarm	arn:aws:cloudwatch:us-east-1:123456789012:alarm:MyCloudWatchAlarm i-078a280217EXAMPLE	
AWS::EC2::Instance	arn:aws:ec2:us-east-1:123456789012:instance/ i-078a280217EXAMPLE	
AWS::EC2::Image	arn:aws:ec2:us-east-1:123456789012:image/ami-02250b3732EXAMPLE	
AWS::EC2::SecurityGroup	arn:aws:ec2:us-east-1:123456789012:security-group/sg-cEXAMPLE	
AWS::EC2::Snapshot	arn:aws:ec2:us-east-1:123456789012:snapshot/snap-03866bf003EXAMPLE	
AWS::EC2::Volume	arn:aws:ec2:us-east-1:123456789012:volume/vol-0912e04d78EXAMPLE	
AWS::DynamoDB::Table	arn:aws:dynamodb:us-east-1:123456789012:table/MyTable	

Jenis sumber daya	Contoh TARGET_ID
AWS::RDS::DBCluster	arn:aws:rds:us-east-2:123456789012:cluster:My-Cluster
AWS::RDS::DBInstance	arn:aws:rds:us-east-1:123456789012:db:My-SQL-Instance
AWS::S3::Bucket	arn:aws:s3:::DOC-EXAMPLE-BUCKET
AWS::SSM::ManagedInstance	arn:aws:ssm:us-east-1:123456789012:managed-instance/mi-0feadcf2d9EXAMPLE

- **RESOURCE_ID**: ID singkat dari jenis sumber daya yang terkandung dalam grup sumber daya. Tabel berikut menyediakan sampel nilai RESOURCE_ID untuk jenis sumber daya tertentu dalam grup sumber daya.

> **Note**
>
> RESOURCE_IDtidak didukung untuk Run Command tugas.

Jenis sumber daya	Contoh RESOURCE_ID
AWS::CloudWatch::Alarm	MyCloudWatchAlarm
AWS::EC2::Instance	i-078a280217EXAMPLE
AWS::EC2::Image	ami-02250b3732EXAMPLE

Jenis sumber daya	Contoh RESOURCE_ID	
AWS::EC2::Security Group	sg-cEXAMPLE	
AWS::EC2::Snapshot	snap-03866bf003EXAMPLE	
AWS::EC2::Volume	vol-0912e04d78EXAMPLE	
AWS::DynamoDB::Table	MyTable	
AWS::RDS::DBCluster	My-Cluster	
AWS::RDS::DBInstance	My-SQL-Instance	
AWS::S3::Bucket	DOC-EXAMPLE-BUCKET	
AWS::SSM::ManagedInstance	mi-0feadcf2d9EXAMPLE	

> **Note**
>
> Jika grup sumber daya AWS yang Anda tentukan mencakup jenis sumber daya yang tidak menghasilkan nilai `RESOURCE_ID`, dan tidak tercantum di tabel sebelumnya, maka parameter `RESOURCE_ID` tidak diisi. Permintaan eksekusi masih akan terjadi untuk sumber daya tersebut. Dalam hal ini, gunakan parameter semu `TARGET_ID` saja, yang akan diganti dengan ARN lengkap dari sumber daya.

Penjadwalan jendela pemeliharaan dan pilihan periode aktif

Ketika membuat jendela pemeliharaan, Anda harus menentukan seberapa sering jendela pemeliharaan berjalan dengan menggunakan Ekspresi cron atau tingkat. Secara opsional, Anda dapat menentukan rentang tanggal di mana jendela pemeliharaan dapat berjalan pada jadwal regulernya dan zona waktu yang mendasari jadwal reguler tersebut.

Akan tetapi, ingatlah bahwa pilihan zona waktu serta pilihan tanggal mulai dan tanggal akhir tidak saling memengaruhi. Waktu tanggal mulai dan tanggal akhir yang Anda tentukan (dengan atau tanpa offset untuk zona waktu Anda) menentukan hanya periode valid di mana jendela pemeliharaan dapat berjalan sesuai jadwalnya. Pilihan zona waktu menentukan zona waktu internasional di mana jadwal jendela pemeliharaan didasarkan selama periode validnya.

> **Note**
>
> Anda menentukan tanggal mulai dan akhir dalam format stempel waktu ISO-8601. Sebagai contoh: `2021-04-07T14:29:00-08:00`
> Anda menentukan zona waktu dalam format Internet Assigned Numbers Authority (IANA). Misalnya: `America/Chicago`, `Europe/Berlin` atau `Asia/Tokyo`

Contoh

- Contoh 1: Menentukan tanggal mulai jendela pemeliharaan
- Contoh 2: Menentukan tanggal mulai dan tanggal akhir jendela pemeliharaan
- Contoh 3: Membuat jendela pemeliharaan yang berjalan hanya sekali
- Contoh 4: Menentukan jumlah hari offset jadwal untuk jendela pemeliharaan

Contoh 1: Menentukan tanggal mulai jendela pemeliharaan

Anggap saja Anda menggunakan AWS Command Line Interface (AWS CLI) untuk membuat jendela pemeliharaan dengan pilihan berikut:

- `--start-date 2021-01-01T00:00:00-08:00`
- `--schedule-timezone "America/Los_Angeles"`
- `--schedule "cron(0 09 ? * WED *)"`

Sebagai contoh:

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 *)"
```

Ini artinya eksekusi pertama dari jendela pemeliharaan tidak akan terjadi sampai setelah tanggal dan waktu mulai yang ditentukan, yaitu pukul 00.00 Waktu Pasifik AS pada hari Jumat, 1 Januari 2021. (Zona waktu ini adalah delapan jam lebih lambat dari waktu UTC.) Dalam hal ini, tanggal dan waktu mulai dari periode jendela tidak mewakili kapan jendela pemeliharaan pertama kali berjalan. Jika digunakan bersama-sama, nilai `--schedule-timezone` dan `--schedule` artinya jendela pemeliharaan berjalan pada pukul 9.00 setiap hari Rabu di Zona Waktu Pasifik AS (diwakili oleh “America/Los Angeles” dalam format IANA). Eksekusi pertama dalam periode yang diizinkan yaitu pada hari Rabu, 4 Januari 2021, pukul 9.00 Waktu Pasifik AS.

Contoh 2: Menentukan tanggal mulai dan tanggal akhir jendela pemeliharaan

Anggap saja berikutnya Anda membuat jendela pemeliharaan dengan pilihan ini:

- `--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)"`

Sebagai contoh:

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)"
```

Periode yang diizinkan untuk jendela pemeliharaan ini dimulai pada pukul 3.15 Waktu Standar Jepang pada 1 Januari 2019. Periode valid untuk jendela pemeliharaan ini berakhir pada pukul 6.15 Waktu Standar Jepang pada hari Minggu, 30 Juni 2019. (Zona waktu ini sembilan jam lebih cepat dari waktu UTC.) Jika digunakan bersama-sama, nilai `--schedule-timezone` dan `--schedule` artinya jendela pemeliharaan berjalan pada pukul 3.15 setiap hari Selasa di Zona Waktu Standar Jepang (diwakili oleh “Asia/Tokyo” dalam format IANA). Ini karena jendela pemeliharaan berjalan setiap tujuh hari, dan menjadi aktif pada pukul 3.15 pada hari Selasa, 1 Januari. Eksekusi terakhir adalah pukul 3.15 Waktu Standar Jepang pada hari Selasa, 25 Juni 2019. Ini adalah hari Selasa terakhir sebelum periode pemeliharaan yang diizinkan berakhir lima hari kemudian.

Contoh 3: Membuat jendela pemeliharaan yang berjalan hanya sekali

Sekarang Anda membuat jendela pemeliharaan dengan pilihan ini:

- `--schedule "at(2020-07-07T15:55:00)"`

Sebagai contoh:

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
```

Jendela pemeliharaan ini hanya berjalan satu kali, pada pukul 15.55 waktu UTC pada tanggal 7 Juli 2020. Jendela pemeliharaan diizinkan untuk berjalan hingga lima jam, sesuai kebutuhan, tetapi tugas baru tidak dapat dimulai dua jam sebelum akhir periode pemeliharaan jendela.

Contoh 4: Menentukan jumlah hari offset jadwal untuk jendela pemeliharaan

Sekarang Anda membuat jendela pemeliharaan dengan pilihan ini:

```
--schedule-offset 2
```

Sebagai contoh:

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
```

Offset jadwal adalah jumlah hari untuk menunggu setelah tanggal dan waktu yang ditentukan oleh ekspresi CRON sebelum menjalankan jendela pemeliharaan.

Di contoh sebelumnya, ekspresi CRON menjadwalkan jendela pemeliharaan untuk berjalan pada hari Selasa ketiga setiap bulan pada pukul 23.30:

```
--schedule "cron(0 30 23 ? * TUE#3 *)
```

Akan tetapi, pencakupan `--schedule-offset 2` artinya jendela pemeliharaan tidak akan berjalan sampai pukul 23.30 dua hari setelah hari Selasa ketiga setiap bulan.

Offset jadwal didukung untuk ekspresi CRON saja.

Info lebih lanjut

- Referensi: Ekspresi cron dan rate untuk Systems Manager
- Membuat jendela pemeliharaan (konsol)
- Tutorial: Membuat dan mengonfigurasi jendela pemeliharaan (AWS CLI)
- CreateMaintenanceWindow di Referensi API AWS Systems Manager
- create-maintenance-windowdi AWS Systems Managerbagian Referensi AWS CLI Perintah
- Basis data Zona Waktu pada situs web IANA

Pendaftaran tugas jendela pemeliharaan tanpa target

Untuk setiap jendela pemeliharaan yang Anda buat, Anda dapat menentukan satu atau beberapa tugas yang akan dilakukan saat jendela pemeliharaan berjalan. Dalam kebanyakan kasus, Anda harus menentukan sumber daya, atau target, yang akan dijalankan oleh tugas. Akan tetapi, dalam beberapa kasus, Anda tidak perlu menentukan target secara eksplisit di tugas.

Satu atau beberapa target harus ditentukan untuk tugas Run Command jenis RCommand Systems Manager jendela pemeliharaan. Tergantung dari sifat tugas, target bersifat opsional untuk jenis tugas jendela pemeliharaan lainnya (Otomatisasi, AWS Lambda, dan AWS Step Functions Systems Manager).

Untuk jenis tugas Lambda dan Step Functions, apakah target diperlukan tergantung dari konten dari fungsi atau mesin tahapan yang telah Anda buat.

Dalam banyak kasus, Anda tidak perlu secara eksplisit menentukan target untuk tugas otomatisasi. Misalnya, katakanlah bahwa Anda membuat tugas jenis otomatisasi untuk memperbarui Amazon Machine Image (AMI) untuk Linux menggunakan `AWS-UpdateLinuxAmi` runbook. Ketika tugas berjalan, AMI diperbarui dengan paket distribusi Linux terbaru yang tersedia dan perangkat lunak Amazon. Contoh baru dibuat dari AMI telah menginstal pembaruan ini. Karena ID AMI yang akan diperbarui ditentukan dalam parameter input untuk runbook, tidak perlu untuk menentukan target lagi dalam tugas jendela pemeliharaan.

Demikian pula, anggap saja Anda menggunakan AWS Command Line Interface (AWS CLI) untuk mendaftarkan tugas Otomatisasi jendela pemeliharaanyang menggunakan `AWS-RestartEC2Instance` runbook. Karena node yang dimulai ulang ditentukan di `--task-invocation-parameters` argumen, Anda tidak perlu menentukan `--targets` pilihan juga.

> **Note**
>
> Untuk tugas jendela pemeliharaan tanpa target yang ditentukan, Anda tidak dapat memberikan nilai untuk `--max-errors` dan `--max-concurrency`. Sebaliknya, sistem menyisipkan nilai placeholder dari1, yang mungkin dilaporkan sesuai dengan perintah seperti describe-maintenance-window-tasksdan. get-maintenance-window-task Nilai-nilai ini tidak mempengaruhi tugas Anda yang sedang berjalan dan dapat diabaikan.

Contoh berikut mendemonstrasikan penghapusan `--targets--max-errors`, dan `--max-concurrency` pilihan untuk tugas jendela pemeliharaan tanpa target.

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**
>
> Untuk tugas jendela pemeliharaan yang terdaftar sebelum 23 Desember 2020: Jika Anda menentukan target untuk tugas dan salah satu tidak lagi diperlukan, Anda dapat memperbarui tugas tersebut untuk menghapus target menggunakan konsol Systems Manager atau perintah. update-maintenance-window-taskAWS CLI

Info lebih lanjut

- Pesan kesalahan: “Tugas jendela pemeliharaan tanpa target tidak mendukung MaxConcurrency nilai” dan “Tugas jendela pemeliharaan tanpa target tidak mendukung MaxErrors nilai”

Pemecahan masalah windows pemeliharaan

Gunakan informasi berikut untuk membantu Anda memecahkan masalah dengan windows pemeliharaan.

Topik

Mengedit kesalahan tugas: Pada halaman untuk pengeditan tugas jendela pemeliharaan, daftar IAM role menampilkan pesan kesalahan: “Kami tidak dapat menemukan peran jendela pemeliharaan IAM yang ditentukan untuk tugas ini. Ada kemungkinan telah dihapus, atau mungkin belum dibuat.”

Masalah 1: Peran jendela pemeliharaan AWS Identity and Access Management (IAM) yang awalnya Anda tentukan dihapus setelah Anda membuat tugas.

Kemungkinan perbaikan: 1) Pilih peran jendela pemeliharaan IAM yang berbeda, jika ada di akun Anda, atau buat yang baru dan pilih untuk tugas tersebut.

Masalah 2: Jika tugas dibuat menggunakan AWS Command Line Interface (AWS CLI), AWS Tools for Windows PowerShell, atau SDK AWS, nama peran jendela pemeliharaan IAM yang tidak ada bisa jadi telah ditentukan. Misalnya, peran jendela pemeliharaan IAM dapat dihapus sebelum Anda membuat tugas, atau nama peran mungkin salah ketik, seperti **`myrole`** dan bukan **`my-role`**.

Kemungkinan perbaikan: Pilih nama yang benar dari peran jendela pemeliharaan IAM yang ingin Anda gunakan, atau buat yang baru untuk menentukan tugas tersebut.

Tidak semua target jendela pemeliharaan diperbarui

Masalah: Anda menyadari bahwa tugas jendela pemeliharaan tidak berjalan pada semua sumber daya yang ditargetkan oleh jendela pemeliharaan. Misalnya, di hasil eksekusi jendela pemeliharaan, tugas untuk sumber daya yang ditandai sebagai gagal atau waktu habis.

Solusi: Alasan paling umum untuk tugas jendela pemeliharaan tidak berjalan pada sumber daya target meliputi konektivitas dan ketersediaan. Sebagai contoh:

- Systems Manager kehilangan koneksi ke sumber daya sebelum atau selama operasi jendela pemeliharaan.
- Sumber daya sedang offline atau dihentikan selama operasi jendela pemeliharaan.

Anda dapat menunggu waktu jendela pemeliharaan terjadwal berikutnya untuk menjalankan tugas pada sumber daya. Anda dapat secara manual menjalankan tugas jendela pemeliharaan pada sumber daya yang tidak tersedia atau sedang offline.

Tugas gagal dengan status pemanggilan tugas: “Peran yang disediakan tidak berisi izin SSM yang benar.”

Masalah: Anda telah menentukan peran layanan jendela pemeliharaan untuk tugas, tetapi tugas gagal berjalan dengan sukses dan status pemanggilan tugas melaporkan bahwa “Peran yang disediakan tidak berisi izin SSM yang benar.”

- Solusi: DiTugas 1: Membuat kebijakan untuk peran layanan jendela pemeliharaan kustom Anda, kami menyediakan kebijakan dasar yang dapat Anda lampirkan ke peran layanan jendela pemeliharaan kustom Anda. Kebijakan ini mencakup izin yang diperlukan untuk banyak skenario tugas. Namun, karena banyaknya tugas yang dapat Anda jalankan, Anda mungkin perlu memberikan izin tambahan dalam kebijakan untuk peran jendela pemeliharaan Anda.

  Misalnya, beberapa tindakan Otomatisasi menggunakan tumpukan AWS CloudFormation. Oleh karena itu, Anda mungkin perlu menambahkan izin tambahan `cloudformation:CreateStackcloudformation:DescribeStacks`, dan `cloudformation:DeleteStack` kebijakan untuk peran layanan jendela pemeliharaan Anda.

  Untuk contoh lain, runbook Otomasi `AWS-CopySnapshot` memerlukan izin untuk membuat snapshot Amazon Elastic Block Store (Amazon EBS). Oleh karena itu, Anda mungkin perlu menambahkan izin`ec2:CreateSnapshot`.

  Untuk informasi tentang izin peran yang diperlukan oleh runbook Otomasi AWS terkelola, lihat deskripsi buku runbook di referensi buku runbook OtomasiAWS Systems Manager.

  Untuk informasi tentang izin peran yang diperlukan oleh dokumen SSM AWS terkelola, tinjau konten dokumen di konsol Systems Manager bagian Documents.

Untuk informasi tentang izin peran yang diperlukan untuk tugas Step Functions, tugas Lambda, dan runbook Otomasi kustom dan dokumen SSM, verifikasi persyaratan izin dengan pembuat sumber daya tersebut.

Tugas gagal dengan pesan kesalahan: “Langkah gagal saat memvalidasi dan menyelesaikan masalah input langkah”

Masalah: Runbook Otomatisasi atau dokumen Systems Manager Command yang Anda gunakan di tugas mengharuskan Anda menentukan input seperti `InstanceId` atau `SnapshotId`, tetapi nilai tidak disediakan atau tidak disediakan dengan benar.

- Solusi 1: Jika tugas Anda menargetkan sumber daya tunggal, seperti satu node atau snapshot tunggal, masukkan ID-nya dalam parameter input untuk tugas tersebut.
- Solusi 2: Jika tugas Anda menargetkan beberapa sumber daya, seperti membuat gambar dari beberapa node saat Anda menggunakan runbook`AWS-CreateImage`, Anda dapat menggunakan salah satu parameter semu yang didukung untuk tugas jendela pemeliharaan dalam parameter input untuk mewakili ID node dalam perintah.

  Perintah berikut mendaftarkan tugas Otomatisasi Systems Manager dengan jendela pemeliharaan menggunakan AWS CLI. Nilai `--targets` menunjukkan ID target jendela pemeliharaan. Selain itu, meskipun `--targets` parameter menentukan ID target jendela, parameter runbook Otomasi mengharuskan ID node disediakan. Dalam hal ini, perintah menggunakan parameter semu `{{RESOURCE_ID}}` sebagai nilai `InstanceId`.

  AWS CLIperintah:

  Linux & macOS

  Contoh perintah berikut memulai ulang instans Amazon Elastic Compute Cloud (Amazon EC2) yang termasuk dalam grup target jendela pemeliharaan dengan ID E32eECB2-646C-4F4B-8ED1-205FBEXample.

```
aws ssm register-task-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --targets Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE \
    --task-arn "AWS-RestartEC2Instance" \
    --service-role-arn arn:aws:iam::123456789012:role/
MyMaintenanceWindowServiceRole \
    --task-type AUTOMATION \
```

```
    --task-invocation-parameters
 "Automation={DocumentVersion=5,Parameters={InstanceId='{{RESOURCE_ID}}'}}" \
    --priority 0 --max-concurrency 10 --max-errors 5 --name "My-Restart-EC2-
Instances-Automation-Task" \
    --description "Automation task to restart EC2 instances"
```

Windows

```
aws ssm register-task-with-maintenance-window ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --targets Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE ^
    --task-arn "AWS-RestartEC2Instance" ^
    --service-role-arn arn:aws:iam::123456789012:role/
MyMaintenanceWindowServiceRole ^
    --task-type AUTOMATION ^
    --task-invocation-parameters
 "Automation={DocumentVersion=5,Parameters={InstanceId='{{RESOURCE_ID}}'}}" ^
    --priority 0 --max-concurrency 10 --max-errors 5 --name "My-Restart-EC2-
Instances-Automation-Task" ^
    --description "Automation task to restart EC2 instances"
```

Untuk informasi lebih lanjut tentang menggunakan parameter semu untuk tugas jendela pemeliharaan, lihat Menggunakan parameter semu saat mendaftarkan tugas jendela pemeliharaan dan Contoh pendaftaran tugas.

Pesan kesalahan: “Tugas jendela pemeliharaan tanpa target tidak mendukung MaxConcurrency nilai” dan “Tugas jendela pemeliharaan tanpa target tidak mendukung MaxErrors nilai”

Masalah: Saat Anda mendaftarkan tugas Run Command -type, Anda harus menentukan setidaknya satu target untuk menjalankan tugas. Untuk jenis tugas lainnya (Otomatisasi, AWS Lambda, dan AWS Step Functions), tergantung dari sifat tugas, target bersifat opsional. Pilihan `MaxConcurrency` (jumlah sumber daya untuk menjalankan tugas pada waktu yang sama) dan `MaxErrors` (jumlah kegagalan untuk menjalankan tugas pada sumber daya target sebelum tugas gagal) tidak diperlukan atau didukung untuk tugas jendela pemeliharaan yang tidak menentukan target. Sistem menghasilkan pesan kesalahan ini jika nilai ditentukan untuk salah satu pilihan ini saat tidak ada target tugas yang ditentukan.

Solusi: Jika Anda menerima salah satu dari kesalahan ini, hapus nilai untuk konkurensi dan ambang kesalahan sebelum melanjutkan untuk mendaftarkan atau memperbarui tugas jendela pemeliharaan.

Untuk informasi lebih lanjut tentang menjalankan tugas yang tidak menentukan target, lihat Pendaftaran tugas jendela pemeliharaan tanpa target di Panduan Pengguna AWS Systems Manager.

AWS Systems Manager Manajemen Node

AWS Systems Manager menyediakan kemampuan berikut untuk mengakses, mengelola, dan mengonfigurasi node terkelola Anda. Node terkelola adalah mesin apa pun yang dikonfigurasi untuk digunakan dengan Systems Manager di lingkungan hybrid dan multicloud.

Topik

AWS Systems Manager Fleet Manager

Fleet Manager, kemampuan AWS Systems Manager, adalah pengalaman antarmuka pengguna terpadu (UI) yang membantu Anda mengelola node Anda dari jarak jauh yang berjalan di AWS atau di tempat. DenganFleet Manager, Anda dapat melihat status kesehatan dan kinerja seluruh armada server Anda dari satu konsol. Anda juga dapat mengumpulkan data dari masing-masing node untuk melakukan pemecahan masalah umum dan tugas manajemen dari konsol. Ini termasuk menghubungkan ke instance Windows menggunakan Remote Desktop Protocol (RDP), melihat folder dan konten file, manajemen registri Windows, manajemen pengguna sistem operasi, dan banyak lagi. Untuk memulaiFleet Manager, buka konsol Systems Manager. Di panel navigasi, pilih Fleet Manager.

Siapa yang harus menggunakanFleet Manager?

Setiap AWS pelanggan yang menginginkan cara terpusat untuk mengelola armada node mereka harus menggunakannyaFleet Manager.

Bagaimana bisa Fleet Manager menguntungkan organisasi saya?

Fleet Managermenawarkan manfaat ini:

- Lakukan berbagai tugas administrasi sistem umum tanpa harus terhubung secara manual ke node terkelola Anda.
- Kelola node yang berjalan di beberapa platform dari satu konsol terpadu.
- Kelola node yang menjalankan sistem operasi yang berbeda dari satu konsol terpadu.
- Meningkatkan efisiensi administrasi sistem Anda.

Apa saja fitur-fiturnyaFleet Manager?

Fitur utama Fleet Manager meliputi:

- Akses Portal Pangkalan Pengetahuan Red Hat

  Akses binari, berbagi pengetahuan, dan forum diskusi di Red Hat Knowledgebase Portal melalui () instans Anda. Red Hat Enterprise Linux RHEL
- Status simpul terkelola

  Lihat instance terkelola mana `running` dan manastopped. Untuk informasi selengkapnya tentang instans yang dihentikan, lihat Menghentikan dan memulai instans Anda di Panduan Pengguna Amazon EC2 untuk Instans Linux. Untuk perangkat AWS IoT Greengrass inti, Anda dapat melihat yang `onlineoffline`,, atau menampilkan status`Connection lost`.

  > **Note**
  >
  > Jika Anda menghentikan instans terkelola sebelum 12 Juli 2021, itu tidak akan menampilkan `stopped` penanda. Untuk menunjukkan penanda, mulai dan hentikan instance.

- Lihat informasi contoh

  Melihat informasi tentang folder dan data file yang disimpan pada volume yang dilampirkan pada instans terkelola, data kinerja tentang instans Anda secara real-time, dan data log yang disimpan pada instans Anda.
- Lihat informasi perangkat tepi

Lihat nama AWS IoT Greengrass Thing untuk perangkat, status SSM Agent ping dan versi, dan lainnya.

- Kelola akun dan registri

  Kelola akun pengguna sistem operasi (OS) pada instans dan registri Anda pada instance Windows Anda.

- Kontrol akses ke fitur

  Kontrol akses ke Fleet Manager fitur menggunakan kebijakan AWS Identity and Access Management (IAM). Dengan kebijakan ini, Anda dapat mengontrol pengguna atau grup individu mana di organisasi Anda yang dapat menggunakan berbagai Fleet Manager fitur, dan node terkelola mana yang dapat mereka kelola.

Topik

Memulai dengan Fleet Manager

Sebelum Anda dapat menggunakanFleet Manager, kemampuanAWS Systems Manager, untuk memantau dan mengelola node Anda yang dikelola, selesaikan langkah-langkah dalam topik berikut.

Topik

Langkah 1: Buat kebijakan IAM dengan izin Fleet Manager

Untuk menggunakanFleet Manager, kemampuan AWS Systems Manager, pengguna atau peran AWS Identity and Access Management (IAM) Anda harus memiliki izin yang diperlukan. Anda dapat membuat kebijakan IAM yang menyediakan akses ke semua Fleet Manager fitur, atau mengubah kebijakan Anda untuk memberikan akses ke fitur yang Anda pilih.

Contoh kebijakan di bawah ini memberikan izin yang diperlukan untuk semua Fleet Manager fitur dan izin yang diperlukan untuk subset fitur.

Untuk informasi selengkapnya tentang membuat dan mengedit kebijakan IAM, lihat Membuat Kebijakan IAM dalam Panduan Pengguna IAM.

Topik

- Contoh kebijakan untuk akses Fleet Manager administrator
- Contoh kebijakan untuk akses Fleet Manager hanya-baca

Contoh kebijakan untuk akses Fleet Manager administrator

Kebijakan berikut memberikan izin untuk semua Fleet Manager fitur. Ini berarti pengguna dapat membuat dan menghapus pengguna dan grup lokal, memodifikasi keanggotaan grup untuk grup lokal apa pun, dan memodifikasi kunci atau nilai Windows Server registri. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"EC2",
         "Effect":"Allow",
         "Action":[
            "ec2:CreateTags",
            "ec2:DeleteTags",
            "ec2:DescribeInstances",
            "ec2:DescribeTags"
         ],
         "Resource":"*"
      },
      {
         "Sid":"General",
         "Effect":"Allow",
         "Action":[
            "ssm:AddTagsToResource",
            "ssm:DescribeInstanceAssociationsStatus",
            "ssm:DescribeInstancePatches",
            "ssm:DescribeInstancePatchStates",
            "ssm:DescribeInstanceProperties",
            "ssm:GetCommandInvocation",
```

```
            "ssm:GetServiceSetting",
            "ssm:GetInventorySchema",
            "ssm:ListComplianceItems",
            "ssm:ListInventoryEntries",
            "ssm:ListTagsForResource",
            "ssm:ListCommandInvocations",
            "ssm:ListAssociations",
            "ssm:RemoveTagsFromResource"
         ],
         "Resource":"*"
      },
      {
            "Sid":"DefaultHostManagement",
            "Effect": "Allow",
            "Action": [
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "arn:aws:ssm:region:account-id:servicesetting/ssm/managed-
instance/default-ec2-instance-management-role"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::account-id:role/service-role/
AWSSystemsManagerDefaultEC2InstanceManagementRole",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
                        "ssm.amazonaws.com"
                    ]
                }
            }
      },
      {
         "Sid":"SendCommand",
         "Effect":"Allow",
         "Action":[
            "ssm:GetDocument",
            "ssm:SendCommand",
            "ssm:StartSession"
         ],
```

```
        "Resource":[
           "arn:aws:ec2:*:account-id:instance/*",
           "arn:aws:ssm:*:account-id:managed-instance/*",
           "arn:aws:ssm:*:account-id:document/SSM-SessionManagerRunShell",
           "arn:aws:ssm:*:*:document/AWS-PasswordReset",
           "arn:aws:ssm:*:*:document/AWSFleetManager-AddUsersToGroups",
           "arn:aws:ssm:*:*:document/AWSFleetManager-CopyFileSystemItem",
           "arn:aws:ssm:*:*:document/AWSFleetManager-CreateDirectory",
           "arn:aws:ssm:*:*:document/AWSFleetManager-CreateGroup",
           "arn:aws:ssm:*:*:document/AWSFleetManager-CreateUser",
           "arn:aws:ssm:*:*:document/AWSFleetManager-CreateUserInteractive",
           "arn:aws:ssm:*:*:document/AWSFleetManager-CreateWindowsRegistryKey",
           "arn:aws:ssm:*:*:document/AWSFleetManager-DeleteFileSystemItem",
           "arn:aws:ssm:*:*:document/AWSFleetManager-DeleteGroup",
           "arn:aws:ssm:*:*:document/AWSFleetManager-DeleteUser",
           "arn:aws:ssm:*:*:document/AWSFleetManager-DeleteWindowsRegistryKey",
           "arn:aws:ssm:*:*:document/AWSFleetManager-DeleteWindowsRegistryValue",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetDiskInformation",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetFileContent",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetFileSystemContent",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetGroups",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetPerformanceCounters",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetProcessDetails",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetUsers",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetWindowsEvents",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetWindowsRegistryContent",
           "arn:aws:ssm:*:*:document/AWSFleetManager-MountVolume",
           "arn:aws:ssm:*:*:document/AWSFleetManager-MoveFileSystemItem",
           "arn:aws:ssm:*:*:document/AWSFleetManager-RemoveUsersFromGroups",
           "arn:aws:ssm:*:*:document/AWSFleetManager-RenameFileSystemItem",
           "arn:aws:ssm:*:*:document/AWSFleetManager-SetWindowsRegistryValue",
           "arn:aws:ssm:*:*:document/AWSFleetManager-StartProcess",
           "arn:aws:ssm:*:*:document/AWSFleetManager-TerminateProcess"
        ],
        "Condition":{
           "BoolIfExists":{
              "ssm:SessionDocumentAccessCheck":"true"
           }
        }
     },
     {
        "Sid":"TerminateSession",
        "Effect":"Allow",
        "Action":[
```

```
            "ssm:TerminateSession"
         ],
         "Resource":"*",
         "Condition":{
            "StringLike":{
               "ssm:resourceTag/aws:ssmmessages:session-id":[
                  "${aws:userid}"
               ]
            }
         }
      },
      {
         "Sid":"KMS",
         "Effect":"Allow",
         "Action":[
            "kms:GenerateDataKey"
         ],
         "Resource":[
            "arn:aws:kms:region:account-id:key/key-name"
         ]
      }
   ]
}
```

Contoh kebijakan untuk akses Fleet Manager hanya-baca

Kebijakan berikut memberikan izin untuk fitur hanya-bacaFleet Manager. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"EC2",
         "Effect":"Allow",
         "Action":[
            "ec2:DescribeInstances",
            "ec2:DescribeTags"
         ],
         "Resource":"*"
      },
      {
         "Sid":"General",
```

```
         "Effect":"Allow",
         "Action":[
            "ssm:DescribeInstanceAssociationsStatus",
            "ssm:DescribeInstancePatches",
            "ssm:DescribeInstancePatchStates",
            "ssm:DescribeInstanceProperties",
            "ssm:GetCommandInvocation",
            "ssm:GetServiceSetting",
            "ssm:GetInventorySchema",
            "ssm:ListComplianceItems",
            "ssm:ListInventoryEntries",
            "ssm:ListTagsForResource",
            "ssm:ListCommandInvocations",
            "ssm:ListAssociations"
         ],
         "Resource":"*"
      },
      {
         "Sid":"SendCommand",
         "Effect":"Allow",
         "Action":[
            "ssm:GetDocument",
            "ssm:SendCommand",
            "ssm:StartSession"
         ],
         "Resource":[
            "arn:aws:ec2:*:account-id:instance/*",
            "arn:aws:ssm:*:account-id:managed-instance/*",
            "arn:aws:ssm:*:account-id:document/SSM-SessionManagerRunShell",
            "arn:aws:ssm:*:*:document/AWSFleetManager-GetDiskInformation",
            "arn:aws:ssm:*:*:document/AWSFleetManager-GetFileContent",
            "arn:aws:ssm:*:*:document/AWSFleetManager-GetFileSystemContent",
            "arn:aws:ssm:*:*:document/AWSFleetManager-GetGroups",
            "arn:aws:ssm:*:*:document/AWSFleetManager-GetPerformanceCounters",
            "arn:aws:ssm:*:*:document/AWSFleetManager-GetProcessDetails",
            "arn:aws:ssm:*:*:document/AWSFleetManager-GetUsers",
            "arn:aws:ssm:*:*:document/AWSFleetManager-GetWindowsEvents",
            "arn:aws:ssm:*:*:document/AWSFleetManager-GetWindowsRegistryContent"
         ],
         "Condition":{
            "BoolIfExists":{
               "ssm:SessionDocumentAccessCheck":"true"
            }
         }
```

```
      },
      {
         "Sid":"TerminateSession",
         "Effect":"Allow",
         "Action":[
            "ssm:TerminateSession"
         ],
         "Resource":"*",
         "Condition":{
            "StringLike":{
               "ssm:resourceTag/aws:ssmmessages:session-id":[
                  "${aws:userid}"
               ]
            }
         }
      },
      {
         "Sid":"KMS",
         "Effect":"Allow",
         "Action":[
            "kms:GenerateDataKey"
         ],
         "Resource":[
            "arn:aws:kms:region:account-id:key/key-name"
         ]
      }
   ]
}
```

Langkah 2: MemMemverifikasi dari Memverifikasi dari Systems Manager

Untuk instans Amazon Elastic Compute Cloud (Amazon EC2), dan server on-premise, perangkat edge, dan mesin virtual (VM)Fleet Manager, dan mesin virtual (VM) untuk diharuskan dengan Systems Manager.AWS IoT GreengrassAWS Systems Manager Ini berarti node Anda harus memenuhi prasyarat tertentu dan dikonfigurasi denganAWS Systems Manager Agen (SSM Agent). Untuk informasi selengkapnya, lihat Menyiapkan AWS Systems Manager.

Anda dapat menggunakanQuick SetupAWS Systems Manager, untuk membantu Anda dengan cepat mengonfigurasi instans Amazon EC2 sebagai instans terkelola dalam akun individual. Jika bisnis atau organisasi Anda menggunakanAWS Organizations, Anda juga dapat mengkonfigurasi instance di beberapa unit organisasi (oU) danWilayah AWS. Untuk informasi selengkapnya tentang

penggunaanQuick Setup untuk mengonfigurasi instance terkelola, lihatManajemen host Amazon EC2.

> **Note**
>
> Untuk mesin non-EC2 yang tidak berjalan diAWS, gunakan aktivasi hibrid untuk mengonfigurasi mesin untuk digunakan dengan Systems Manager di hibrid dan multicloud. Untuk informasi tentang aktivasi hibrid, lihat AWS Systems Manager Aktivasi Hibrid.

Bekerja dengan Fleet Manager

Anda dapat menggunakanFleet Manager, kemampuan AWS Systems Manager, untuk melakukan berbagai tugas pada node terkelola Anda dari AWS Systems Manager konsol. Topik berikut menjelaskan fitur yang disediakan olehFleet Manager.

> **Note**
>
> Satu-satunya fitur yang didukung untuk instans macOS adalah melihat sistem file.

Topik

- Bekerja dengan node terkelola
- Menggunakan pengaturan Konfigurasi Manajemen Host Default
- Menghubungkan ke instance Windows Server terkelola menggunakan Remote Desktop
- Mengelola volume Amazon EBS pada instans terkelola
- Bekerja dengan sistem file
- Memantau kinerja node terkelola
- Bekerja dengan proses
- Melihat log pada node terkelola
- Mengelola akun pengguna OS pada node terkelola
- Mengelola registri Windows pada node terkelola
- Mengakses portal Red Hat Knowledgebase

Bekerja dengan node terkelola

Node terkelola adalah mesin apa pun yang dikonfigurasi untuk AWS Systems Manager. Anda dapat mengonfigurasi jenis mesin berikut sebagai node terkelola:

- Instans Amazon Elastic Compute Cloud (Amazon EC2)
- Server di tempat Anda sendiri (server lokal)
- AWS IoT Greengrass perangkat inti
- AWS IoT dan perangkat AWS non-edge
- Mesin virtual (VM), termasuk VM di lingkungan cloud lainnya

> **Note**
>
> Di konsol Systems Manager, mesin apa pun yang diawali dengan “mi-” telah dikonfigurasi sebagai node terkelola menggunakan aktivasi hibrida. Perangkat Edge menampilkan nama AWS IoT Thing mereka.

AWS Systems Manager menawarkan tingkat instans standar dan tingkat instans lanjutan. Keduanya mendukung node terkelola di lingkungan hybrid dan multicloud Anda. Tingkat instans standar memungkinkan Anda mendaftarkan maksimum 1.000 mesin per per mesin. Akun AWS Wilayah AWS Jika Anda perlu mendaftarkan lebih dari 1.000 mesin dalam satu akun dan Wilayah, gunakan tingkat instance lanjutan. Anda dapat membuat node terkelola sebanyak yang Anda suka di tingkat instance lanjutan. Semua node terkelola yang dikonfigurasi untuk Systems Manager diberi harga pay-per-use berdasarkan. Untuk informasi selengkapnya tentang mengaktifkan tingkat instans lanjutan, lihat. Mengaktifkan tingkat instans lanjutan Untuk informasi lebih lanjut tentang harga, lihat AWS Systems Manager Harga.

> **Note**
>
> - Instans lanjutan juga memungkinkan Anda untuk terhubung ke node non-EC2 Anda di lingkungan hybrid dan multicloud dengan menggunakan. AWS Systems Manager Session Manager Session Managermenyediakan akses shell interaktif ke instance Anda. Untuk informasi selengkapnya, lihat AWS Systems Manager Session Manager.
> - Kuota instans standar juga berlaku untuk instans EC2 yang menggunakan aktivasi on-premise Systems Manager (yang bukan merupakan skenario umum).

- Untuk menambal aplikasi yang dirilis oleh Microsoft di instans on-premise mesin virtual (VM), aktifkan tingkat instans lanjutan. Biaya dikenakan untuk menggunakan tingkat instans lanjutan. Biaya tambahan dikenakan untuk menambal aplikasi yang dirilis oleh instans Microsoft Amazon Elastic Compute Cloud (Amazon EC2). Untuk informasi selengkapnya, lihat Mengenai aplikasi patching yang dikeluarkan oleh Microsoft pada Windows Server.

Tampilkan node terkelola

Jika Anda tidak melihat node terkelola yang terdaftar di konsol, lakukan hal berikut:

1. Verifikasi bahwa konsol terbuka di Wilayah AWS tempat Anda membuat node terkelola. Anda dapat beralih Wilayah dengan menggunakan daftar di sudut kanan atas konsol.
2. Verifikasi bahwa langkah penyiapan untuk node terkelola memenuhi persyaratan Systems Manager. Untuk informasi, lihat Menyiapkan AWS Systems Manager.
3. Untuk mesin non-EC2, verifikasi bahwa Anda menyelesaikan proses aktivasi hibrida. Untuk informasi selengkapnya, lihat Menyiapkan Manajer Sistem untuk lingkungan hybrid dan multicloud.

> **Note**
>
> Perhatikan informasi berikut.
>
> - Fleet ManagerKonsol tidak menampilkan node Amazon EC2 yang telah dihentikan.
> - Systems Manager memerlukan referensi waktu yang akurat untuk melakukan operasi pada mesin Anda. Jika tanggal dan waktu tidak disetel dengan benar pada node terkelola Anda, mesin mungkin tidak cocok dengan tanggal tanda tangan permintaan API Anda. Untuk informasi selengkapnya, lihat Kasus penggunaan dan praktik terbaik.
> - Saat Anda membuat atau mengedit tag, sistem dapat memakan waktu hingga satu jam untuk menampilkan perubahan dalam filter tabel.
> - Setelah status node terkelola setidaknya `Connection Lost` selama 30 hari, node mungkin tidak lagi terdaftar di Fleet Manager konsol. Untuk mengembalikannya ke daftar, masalah yang menyebabkan koneksi hilang harus diselesaikan. Untuk tips pemecahan masalah, lihat. Memecahkan masalah ketersediaan node terkelola

Verifikasi dukungan Systems Manager pada node terkelola

AWS Config menyediakan Aturan AWS Terkelola, yang merupakan aturan yang telah ditentukan sebelumnya dan dapat disesuaikan yang AWS Config digunakan untuk mengevaluasi apakah konfigurasi AWS sumber daya Anda mematuhi praktik terbaik umum. AWS Config Aturan Terkelola mencakup aturan ec2- instance-managed-by-systems -manager. Aturan ini memeriksa apakah instans Amazon EC2 di akun Anda dikelola oleh Systems Manager. Untuk informasi selengkapnya, lihat AWS Config Managed Rules.

Meningkatkan postur keamanan pada node terkelola

Untuk informasi tentang meningkatkan postur keamanan Anda terhadap perintah tingkat root yang tidak sah pada node terkelola Anda, lihat. Membatasi akses ke perintah tingkat root melaluiSSM Agent

Node terkelola deregister

Anda dapat membatalkan pendaftaran node terkelola kapan saja. Misalnya, jika Anda mengelola beberapa node dengan peran AWS Identity and Access Management (IAM) yang sama dan Anda melihat segala jenis perilaku jahat, Anda dapat membatalkan pendaftaran sejumlah mesin kapan saja. Untuk informasi tentang membatalkan pendaftaran node terkelola, lihat. Menderegistrasi node terkelola dalam lingkungan hybrid dan multicloud

Topik

- Mengonfigurasi tingkat instans
- Menyetel ulang kata sandi pada node terkelola
- Menderegistrasi node terkelola dalam lingkungan hybrid dan multicloud

Mengonfigurasi tingkat instans

Topik ini menjelaskan skenario ketika Anda harus mengaktifkan tingkat lanjutan.

AWS Systems Managermenawarkan tingkat instans standar dan tingkat instans standar instans standar untuk mesin non-EC2 dalam lingkungan hibrid dan multicloud.

Anda dapat mendaftarkan hingga 1.000 node standar yang diaktifkan hibrida per akun perWilayah AWS tanpa biaya tambahan. Namun, mendaftarkan lebih dari 1.000 node hibrid mengharuskan Anda mengaktifkan tingkatan instans lanjutan. Biaya dikenakan untuk menggunakan tingkat instans lanjutan. Untuk informasi selengkapnya, lihat AWS Systems Manager Harga.

Bahkan dengan kurang dari 1.000 node yang diaktifkan hibrida terdaftar, dua skenario lain memerlukan tingkat instans lanjutan:

- Anda ingin menggunakanSession Manager untuk terhubung ke node non-EC2.
- Anda ingin menambal aplikasi (bukan sistem operasi) yang dirilis oleh Microsoft pada node non-EC2.

> **Note**
>
> Biaya tambahan berlaku berlaku untuk menambal aplikasi yang dirilis oleh Microsoft instans Amazon EC2 EC2 Amazon EC2 berlaku.

Lanjutan-contoh tingkat skenario rinci

Informasi berikut memberikan detail tentang tiga skenario yang harus Anda aktifkan tingkatan instans lanjutan.

Skenario 1: Anda ingin mendaftarkan lebih dari 1.000 node yang diaktifkan hibrida

Dengan menggunakan tingkatan instans standar, Anda dapat mendaftarkan maksimum 1.000 node non-EC2 di lingkungan hybrid dan multicloud perWilayah AWS akun tertentu tanpa biaya tambahan. Jika Anda perlu mendaftarkan lebih dari 1.000 node non-EC2 di Wilayah, Anda harus menggunakan tingkat instans lanjutan. Anda kemudian dapat mengaktifkan mesin sebanyak mungkin untuk lingkungan hybrid dan multicloud Anda seperti yang Anda inginkan. Biaya untuk tingkatan instans lanjutan didasarkan pada jumlah node lanjutan yang diaktifkan sebagai node yang dikelola Systems Manager dan jam node tersebut berjalan.

Semua node terkelola Systems Manager yang menggunakan proses aktivasi yang dijelaskan dalam Membuat aktivasi node terkelola untuk lingkungan hibrid kemudian dikenakan biaya jika Anda melebihi 1.000 node lokal di Wilayah di akun tertentu.

> **Note**
>
> Anda juga dapat mengaktifkan instans Amazon Elastic Compute Cloud (Amazon EC2) menggunakan aktivasi hibrid Systems Manager dan bekerja dengannya sebagai instans non-EC2, misalnya untuk pengujian. Ini juga memenuhi syarat sebagai node hibrida. Ini bukan skenario umum.

Skenario 2: Menambal aplikasi yang dirilis Microsoft pada node yang diaktifkan hibrida

Tingkat instans lanjutan juga diperlukan jika Anda ingin menambal aplikasi yang dirilis Microsoft pada node non-EC2 di lingkungan hybrid dan multicloud. Jika Anda mengaktifkan tingkatan instans lanjutan untuk menambal aplikasi Microsoft pada node non-EC2, biaya akan dikeluarkan untuk semua node lokal, bahkan jika Anda memiliki kurang dari 1.000.

Biaya tambahan dikenakan untuk menambal aplikasi yang dirilis oleh instans Microsoft Amazon Elastic Compute Cloud (Amazon EC2). Untuk informasi selengkapnya, lihat Mengenai aplikasi patching yang dikeluarkan oleh Microsoft pada Windows Server.

Skenario 3: Menghubungkan ke node yang diaktifkan hibrida menggunakanSession Manager

Session Managermenyediakan akses shell interaktif ke instans Anda aktif. Untuk terhubung ke node terkelola yang diaktifkan hibrida menggunakanSession Manager, Anda harus mengaktifkan tingkat lanjutan-lanjutan. Biaya kemudian dikeluarkan untuk semua node yang diaktifkan hibrida, bahkan jika Anda memiliki kurang dari 1.000.

Ringkasan: Kapan saya membutuhkan tingkat instans lanjutan?

Gunakan tabel berikut untuk meninjau kapan Anda harus menggunakan tingkat instans lanjutan, dan untuk skenario mana biaya tambahan berlaku.

Skenario	Diperlukan tingkatan tingkat lanjutan?	Biaya tambahan berlaku berlaku berlaku tambahan berlaku berlaku berlaku berlaku berlaku
Jumlah node yang diaktifka n hibrida di Wilayah saya di akun tertentu lebih dari 1.000.	Ya	Ya
Saya ingin menggunak anPatch Manager untuk menambal aplikasi yang dirilis Microsoft pada sejumlah node yang diaktifkan hibrida, bahkan kurang dari 1.000.	Ya	Ya

Skenario	Diperlukan tingkatan tingkat lanjutan?	Biaya tambahan berlaku berlaku berlaku tambahan berlaku berlaku berlaku berlaku berlaku
Saya ingin menggunak anSession Manager untuk terhubung ke sejumlah node hibrida diaktifkan, bahkan kurang dari 1.000.	Ya	Ya
1. Jumlah node yang diaktifka n hibrida di suatu Wilayah dalam akun tertentu adalah 1.000 atau kurang; dan 2. Saya tidak menambal aplikasi Microsoft pada node yang diaktifkan hibrida; dan 3. Saya tidak terhubung ke node yang diaktifkan hibrida menggunakanSession Manager.	Tidak	Tidak

Topik

- Mengaktifkan tingkat instans lanjutan
- Mengembalikan dari tingkat instans lanjutan ke tingkat instans standar

Mengaktifkan tingkat instans lanjutan

AWS Systems Managermenawarkan tingkat instans standar dan tingkat instans lanjutan untuk mesin non-EC2 di lingkungan hybrid dan multicloud. Tingkat instans standar memungkinkan Anda mendaftarkan maksimum 1.000 mesin yang diaktifkan hibrida per per per mesin. Akun AWS Wilayah AWS Tingkat instance lanjutan juga diperlukan untuk digunakan untuk menambal aplikasi

yang dirilis Microsoft pada node non-EC2, dan Patch Manager untuk terhubung ke node non-EC2 menggunakan. Session Manager Untuk informasi selengkapnya, lihat Mengonfigurasi tingkat instans.

Bagian ini menjelaskan cara mengonfigurasi lingkungan hybrid dan multicloud Anda untuk menggunakan tingkat instance lanjutan.

Sebelum Anda memulai

Tinjau detail harga untuk instans lanjutan. Instans lanjutan tersedia di file. per-use-basis Untuk informasi selengkapnya, lihat AWS Systems Manager Harga.

Mengonfigurasi izin untuk mengaktifkan tingkat instans lanjutan

Verifikasi bahwa Anda memiliki izin di AWS Identity and Access Management(IAM) untuk mengubah lingkungan Anda dari tingkat instans standar ke tingkat instans lanjutan. Anda harus memiliki kebijakan `AdministratorAccess` IAM yang dilampirkan ke pengguna, grup, atau peran Anda, atau Anda harus memiliki izin untuk mengubah setelan layanan tingkat aktivasi Systems Manager. Pengaturan tingkat aktivasi menggunakan operasi API berikut:

- GetServiceSetting
- UpdateServiceSetting
- ResetServiceSetting

Gunakan prosedur berikut untuk menambahkan kebijakan IAM inline ke sebuah akun pengguna. Kebijakan ini memungkinkan pengguna untuk melihat setelan tingkat instans terkelola saat ini. Kebijakan ini juga memungkinkan pengguna untuk mengubah atau mengatur ulang pengaturan saat ini di Akun AWS dan Wilayah AWS yang ditentukan.

1. Masuk ke AWS Management Console dan buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Pengguna.
3. Dari daftar, pilih nama pengguna untuk menanamkan kebijakan ke dalamnya.
4. Pilih tab Izin.
5. Di sisi kanan halaman, di bawah Kebijakan izin, pilih Tambahkan kebijakan inline.
6. Pilih tab JSON.
7. Ganti konten default dengan yang berikut ini:

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"

            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "arn:aws:ssm:region:aws-account-id:servicesetting/ssm/
managed-instance/activation-tier"
        }
    ]
}
```

8. Pilih Tinjau kebijakan.
9. Di halaman Tinjau Kebijakan, untuk Nama, ketikkan nama untuk kebijakan inline tersebut. Sebagai contoh: **`Managed-Instances-Tier`**.
10. Pilih Buat kebijakan.

Administrator dapat menentukan izin hanya-baca dengan menetapkan kebijakan inline berikut kepada pengguna.

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

```
            "Action": [
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "*"
        }
    ]
}
```

Untuk informasi selengkapnya tentang membuat dan mengedit kebijakan IAM, lihat Membuat Kebijakan IAM dalam Panduan Pengguna IAM.

Mengaktifkan tingkat instans lanjutan (konsol)

Prosedur berikut menunjukkan cara menggunakan konsol Systems Manager untuk mengubah semua node non-EC2 yang ditambahkan menggunakan aktivasi instans terkelola, di tingkat yang ditentukan Akun AWS danWilayah AWS, untuk menggunakan tingkat instance lanjutan.

Sebelum Anda memulai

Verifikasi bahwa konsol tersebut terbuka di Wilayah AWS tempat Anda membuat instans terkelola. Anda dapat beralih Wilayah dengan menggunakan daftar di sudut kanan atas konsol.

Pastikan Anda telah menyelesaikan persyaratan penyiapan untuk instans Amazon Elastic Compute Cloud (Amazon EC2) dan mesin non-EC2 di lingkungan hybrid dan multicloud. Untuk informasi, lihat Menyiapkan AWS Systems Manager.

> **Important**
>
> Prosedur berikut menjelaskan cara mengubah pengaturan tingkat akun. Perubahan ini mengakibatkan tagihan yang ditagih ke akun Anda.

Untuk mengaktifkan tingkat instans lanjutan (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.

3. Pilih Pengaturan, Ubah pengaturan tingkat Instance.
4. Tinjau informasi di kotak dialog tentang mengubah pengaturan akun, dan kemudian, dan lanjutkan.
5. Jika Anda menyetujui, pilih opsi untuk menerima, lalu pilih Ubah pengaturan.

Sistem dapat memerlukan waktu beberapa menit untuk menyelesaikan proses pemindahan semua instans dari tingkat instans standar ke tingkat instans lanjutan.

> **Note**
>
> Untuk informasi tentang mengubah kembali ke tingkat instans standar, lihat Mengembalikan dari tingkat instans lanjutan ke tingkat instans standar.

Mengaktifkan tingkat instans lanjutan (AWS CLI)

Prosedur berikut menunjukkan cara menggunakan AWS Command Line Interface untuk mengubah semua server on-premise dan VM yang ditambahkan menggunakan aktivasi instans terkelola, dalam Akun AWS dan Wilayah AWS yang ditentukan, untuk menggunakan tingkat instans lanjutan.

> **Important**
>
> Prosedur berikut menjelaskan cara mengubah pengaturan tingkat akun. Perubahan ini mengakibatkan tagihan yang ditagih ke akun Anda.

Untuk mengaktifkan tingkat instans lanjutan menggunakan AWS CLI

1. Buka AWS CLI dan jalankan perintah berikut. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   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
```

Tidak ada output jika perintah berhasil.

2. Jalankan perintah berikut untuk melihat pengaturan layanan saat ini untuk node terkelola di saat ini Akun AWS danWilayah AWS.

   Linux & macOS

   ```
   aws ssm get-service-setting \
       --setting-id arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
   instance/activation-tier
   ```

   Windows

   ```
   aws ssm get-service-setting ^
       --setting-id arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
   instance/activation-tier
   ```

   Perintah tersebut mengembalikan informasi seperti berikut.

   ```
   {
       "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"
   ```

```
    }
}
```

Mengaktifkan tingkat instans lanjutan (PowerShell)

Prosedur berikut menunjukkan cara menggunakan AWS Tools for Windows PowerShell untuk mengubah semua server on-premise dan VM yang ditambahkan menggunakan aktivasi instans terkelola, dalam Akun AWS dan Wilayah AWS yang ditentukan, untuk menggunakan tingkat instans lanjutan.

> **Important**
>
> Prosedur berikut menjelaskan cara mengubah pengaturan tingkat akun. Perubahan ini mengakibatkan tagihan yang ditagih ke akun Anda.

Untuk mengaktifkan tingkat instance lanjutan menggunakan PowerShell

1. Buka AWS Tools for Windows PowerShell dan jalankan perintah berikut. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   ```
   Update-SSMServiceSetting `
       -SettingId "arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
   instance/activation-tier" `
       -SettingValue "advanced"
   ```

   Tidak ada output jika perintah berhasil.

2. Jalankan perintah berikut untuk melihat pengaturan layanan saat ini untuk node terkelola di saat ini Akun AWS danWilayah AWS.

   ```
   Get-SSMServiceSetting `
       -SettingId "arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
   instance/activation-tier"
   ```

   Perintah tersebut mengembalikan informasi seperti berikut.

   ```
   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
```

Sistem dapat memakan waktu beberapa menit untuk menyelesaikan proses pemindahan semua node dari tingkat instans standar ke tingkat instance lanjutan.

> **Note**
>
> Untuk informasi tentang mengubah kembali ke tingkat instans standar, lihat Mengembalikan dari tingkat instans lanjutan ke tingkat instans standar.

Mengembalikan dari tingkat instans lanjutan ke tingkat instans standar

Bagian ini menjelaskan cara mengubah node hibrid yang berjalan di tingkat instans lanjutan kembali ke tingkat instans standar. Konfigurasi ini berlaku untuk semua node hibrid dalamAkun AWS dan satuWilayah AWS.

Sebelum Anda memulai

Tinjau detail penting berikut.

> **Note**
>
> - Anda tidak dapat kembali ke tingkat instans standar jika menjalankan lebih dari 1.000 node hibrid di akun dan Wilayah. Anda harus terlebih dahulu membatalkan pendaftaran node hingga Anda memiliki 1.000 atau kurang. Ini juga berlaku untuk instans Amazon Elastic Compute Cloud (Amazon EC2) (yang bukan skenario umum). Untuk informasi selengkapnya, lihat Menderegistrasi node terkelola dalam lingkungan hybrid dan multicloud.
> - Setelah Anda kembali, Anda tidak akan dapat menggunakanSession Manager, kemampuan dariAWS Systems Manager, untuk secara interaktif mengakses node hibrid Anda.
> - Setelah Anda kembali, Anda tidak akan dapat menggunakanPatch Manager, kemampuan dariAWS Systems Manager, untuk menambal aplikasi yang dirilis oleh Microsoft di hibrid.

- Proses mengembalikan semua node hibrid ke tingkat instans standar dapat memakan waktu 30 menit atau lebih.

Bagian ini menjelaskan cara mengembalikan semua node hibrid dalamAkun AWS danWilayah AWS dari tingkat instans lanjutan.

Mengembalikan ke tingkat instans standar (konsol)

Prosedur berikut menunjukkan cara menggunakan konsol Systems Manager untuk mengubah semua node hibrid di lingkungan hibrid dan multicloud Anda untuk menggunakan tingkat instans standar diAkun AWS dan yang ditentukanWilayah AWS.

Untuk mengembalikan ke tingkat instans standar (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih dropdown Pengaturan akun, lalu pilih Pengaturan tingkat instans.
4. Pilih Mengubah pengaturan akun.
5. Tinjau informasi di pop-up tentang mengubah pengaturan akun, lalu jika Anda menyetujui, pilih opsi untuk menerima dan melanjutkan.

Mengembalikan ke tingkat instans standar (AWS CLI)

Prosedur berikut menunjukkan cara menggunakanAWS Command Line Interface untuk mengubah semua node hibrid di lingkungan hibrid dan lingkungan multicloud Anda untuk menggunakan tingkat instans standar diAkun AWS dan yang ditentukanWilayah AWS.

Untuk mengembalikan ke tingkat instans standar menggunakan AWS CLI

1. Buka AWS CLI dan jalankan perintah berikut. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

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
```

Tidak ada output jika perintah berhasil.

2. Jalankan perintah berikut 30 menit setelahnya untuk melihat pengaturan untuk instans terkelola di Akun AWS dan Wilayah AWS saat ini.

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
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
{
    "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"
    }
}
```

Status berubah menjadi Default setelah permintaan disetujui.

Mengembalikan ke tingkat instans standar (PowerShell)

Prosedur berikut menunjukkan cara menggunakanAWS Tools for Windows PowerShell untuk mengubah node hibrid di lingkungan hibrid dan multicloud Anda untuk menggunakan tingkat instans standar diAkun AWS dan yang ditentukanWilayah AWS.

Untuk mengembalikan ke tingkat instans standar menggunakanPowerShell

1. Buka AWS Tools for Windows PowerShell dan jalankan perintah berikut.

```
Update-SSMServiceSetting `
    -SettingId "arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
instance/activation-tier" `
    -SettingValue "standard"
```

Tidak ada output jika perintah berhasil.

2. Jalankan perintah berikut 30 menit setelahnya untuk melihat pengaturan untuk instans terkelola di Akun AWS dan Wilayah AWS saat ini.

```
Get-SSMServiceSetting `
    -SettingId "arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
instance/activation-tier"
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
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
```

Status berubah menjadi Default setelah permintaan disetujui.

Menyetel ulang kata sandi pada node terkelola

Anda dapat mengatur ulang kata sandi untuk setiap pengguna pada node terkelola. Ini termasuk instans Amazon Elastic Compute Cloud (Amazon EC2); perangkat inti AWS IoT Greengrass ; dan server lokal, perangkat edge, dan mesin virtual (VM) yang dikelola oleh. AWS Systems Manager Fungsionalitas pengaturan ulang kata sandi dibangun di atasSession Manager, kemampuan AWS Systems Manager. Anda dapat menggunakan fungsi ini untuk terhubung ke node terkelola tanpa membuka port masuk, memelihara host bastion, atau mengelola kunci SSH.

Reset kata sandi berguna ketika pengguna lupa kata sandi, atau ketika Anda ingin memperbarui kata sandi dengan cepat tanpa membuat koneksi RDP atau SSH ke node yang dikelola.

Prasyarat

Sebelum Anda dapat mengatur ulang kata sandi pada node yang dikelola, persyaratan berikut harus dipenuhi:

- Node terkelola tempat Anda ingin mengubah kata sandi harus berupa node terkelola Systems Manager. Juga, SSM Agent versi 2.3.668.0 atau yang lebih baru harus diinstal pada node yang dikelola.) Untuk informasi tentang menginstal atau memperbaruiSSM Agent, lihatBekerja dengan SSM Agent.
- Fungsionalitas pengaturan ulang kata sandi menggunakan Session Manager konfigurasi yang disiapkan agar akun Anda tersambung ke node terkelola. Oleh karena itu, prasyarat untuk menggunakan Session Manager harus telah diselesaikan untuk akun Anda saat ini. Wilayah AWS Untuk informasi selengkapnya, lihat Menyiapkan Session Manager.

  > **Note**
  >
  > Session Managerdukungan untuk node lokal disediakan hanya untuk tingkat instance lanjutan. Untuk informasi selengkapnya, lihat Mengaktifkan tingkat instans lanjutan.

- AWS Pengguna yang mengubah kata sandi harus memiliki `ssm:SendCommand` izin untuk node terkelola. Untuk informasi selengkapnya, lihat MembatasiRun Command akses akses berdasarkan tag.

Membatasi akses

Anda dapat membatasi kemampuan pengguna untuk mengatur ulang kata sandi ke node terkelola tertentu. Hal ini dilakukan dengan menggunakan kebijakan berbasis identitas untuk Session Manager `ssm:StartSession` operasi dengan dokumen SSM. `AWS-PasswordReset` Untuk informasi selengkapnya, lihat Mengontrol akses sesi pengguna ke instans.

Mengenkripsi data

Aktifkan AWS Key Management Service (AWS KMS) enkripsi lengkap untuk Session Manager data untuk menggunakan opsi reset kata sandi untuk node terkelola. Untuk informasi selengkapnya, lihat Aktifkan enkripsi kunci KMS data sesi (konsol).

Setel ulang kata sandi pada node terkelola

Anda dapat mengatur ulang kata sandi pada node yang dikelola Systems Manager menggunakan Fleet Managerkonsol Systems Manager atau AWS Command Line Interface (AWS CLI).

Untuk mengubah kata sandi pada node terkelola (konsol)

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tombol di sebelah node yang membutuhkan kata sandi baru.
4. Pilih Tindakan instans, Setel ulang kata sandi.
5. Untuk Nama pengguna, masukkan nama pengguna yang Anda gunakan untuk mengubah kata sandinya. Ini bisa berupa nama pengguna yang memiliki akun di node.
6. Pilih Kirim.
7. Ikuti petunjuk di jendela perintah Masukkan kata sandi baru untuk menentukan kata sandi baru.

> **Note**
>
> Jika versi SSM Agent pada node terkelola tidak mendukung penyetelan ulang kata sandi, Anda diminta untuk menginstal versi yang didukung menggunakanRun Command, kemampuan. AWS Systems Manager

Untuk mengatur ulang kata sandi pada node terkelola (AWS CLI)

1. Untuk mengatur ulang kata sandi untuk pengguna pada node terkelola, jalankan perintah berikut. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   > **Note**
   >
   > Untuk menggunakan AWS CLI untuk mengatur ulang kata sandi, Session Manager plugin harus diinstal pada mesin lokal Anda. Untuk informasi, lihat Instal Session Manager plugin untuk 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. Ikuti petunjuk di jendela perintah Masukkan kata sandi baru untuk menentukan kata sandi baru.

Memecahkan masalah pengaturan ulang kata sandi pada node terkelola

Banyak masalah pengaturan ulang kata sandi dapat diselesaikan dengan memastikan bahwa Anda telah melengkapi prasyarat pengaturan ulang kata sandi. Untuk masalah lain, gunakan informasi berikut untuk membantu Anda memecahkan masalah pengaturan ulang kata sandi.

Topik

Node terkelola tidak tersedia

Masalah: Anda ingin mengatur ulang kata sandi untuk node terkelola di halaman konsol instans terkelola, tetapi node tidak ada dalam daftar.

- Solusi: Node terkelola yang ingin Anda sambungkan mungkin tidak dikonfigurasi untuk Systems Manager. Untuk menggunakan instans EC2 dengan Systems Manager, profil instans AWS Identity and Access Management (IAM) yang memberikan izin Systems Manager untuk melakukan tindakan pada instans Anda harus dilampirkan ke instans. Untuk selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager.

  Untuk menggunakan mesin non-EC2 dengan Systems Manager, buat peran layanan IAM yang memberikan izin kepada Systems Manager untuk melakukan tindakan pada node terkelola Anda. Untuk informasi selengkapnya, lihat Membuat peran layanan IAM untuk lingkungan hibrid. (Session Managerdukungan untuk server lokal dan VM disediakan hanya untuk tingkat instance lanjutan. Untuk informasi lebih lanjut, lihatMengaktifkan tingkat instans lanjutan.)

SSM Agenttidak up-to-date (konsol)

Masalah: Pesan melaporkan bahwa versi SSM Agent tidak mendukung fungsi pengaturan ulang kata sandi.

- Solusi: Versi 2.3.668.0 atau yang lebih baru SSM Agent diperlukan untuk melakukan reset kata sandi. Di konsol, Anda dapat memperbarui agen pada node terkelola dengan memilih Perbarui SSM Agent.

Versi terbaru dirilis setiap kali kemampuan baru ditambahkan ke Systems Manager atau pembaruan dibuat untuk kemampuan yang ada. SSM Agent Gagal menggunakan agen versi terbaru dapat mencegah node terkelola Anda menggunakan berbagai kemampuan dan fitur Systems Manager. Untuk alasan itu, kami menyarankan Anda mengotomatiskan proses menjaga agar tetap SSM Agent up to date pada mesin Anda. Untuk informasi, lihat Mengotomatiskan pembaruan ke SSM Agent. Berlangganan halaman Catatan SSM Agent Rilis GitHub untuk mendapatkan pemberitahuan tentang SSM Agent pembaruan.

Opsi pengaturan ulang kata sandi tidak tersedia (AWS CLI)

Masalah: Anda berhasil terhubung ke node terkelola menggunakan AWS CLI `start-session` perintah. Anda menentukan `AWS-PasswordReset` Dokumen SSM dan memberikan nama pengguna yang valid, tetapi permintaan untuk mengubah kata sandi tidak muncul.

- Solusi: Versi SSM Agent pada node terkelola tidak up-to-date. Solusi: Versi 2.3.668.0 atau yang lebih baru diperlukan untuk melakukan pengaturan ulang kata sandi.

  Versi terbaru dirilis setiap kali kemampuan baru ditambahkan ke Systems Manager atau pembaruan dibuat untuk kemampuan yang ada. SSM Agent Gagal menggunakan agen versi terbaru dapat mencegah node terkelola Anda menggunakan berbagai kemampuan dan fitur Systems Manager. Untuk alasan itu, kami menyarankan Anda mengotomatiskan proses menjaga agar tetap SSM Agent up to date pada mesin Anda. Untuk informasi, lihat Mengotomatiskan pembaruan ke SSM Agent. Berlangganan halaman Catatan SSM Agent Rilis GitHub untuk mendapatkan pemberitahuan tentang SSM Agent pembaruan.

Tidak ada otorisasi untuk menjalankan **`ssm:SendCommand`**

Masalah: Anda mencoba menyambung ke node terkelola untuk mengubah kata sandi tetapi menerima pesan kesalahan yang mengatakan bahwa Anda tidak berwenang untuk berjalan `ssm:SendCommand` di node terkelola.

- Solusi: Kebijakan IAM Anda harus menyertakan izin untuk menjalankan `ssm:SendCommand` perintah. Untuk informasi, lihat MembatasiRun Command akses akses berdasarkan tag.

Session Managerpesan kesalahan

Masalah: Anda menerima pesan kesalahan yang terkait denganSession Manager.

- Solusi: Dukungan reset kata sandi mengharuskan yang Session Manager dikonfigurasi dengan benar. Untuk informasi selengkapnya, lihat Menyiapkan Session Manager dan Pemecahan Masalah Session Manager.

Menderegistrasi node terkelola dalam lingkungan hybrid dan multicloud

Jika Anda tidak lagi ingin mengelola server lokal, perangkat edge, atau mesin virtual (VM) dengan menggunakanAWS Systems Manager, Anda dapat membatalkan pendaftarannya. Membatalkan pendaftaran node yang diaktifkan hibrida menghapusnya dari daftar node terkelola di Systems Manager. AWS Systems Manager Agent (SSM Agent) yang berjalan pada node yang diaktifkan hibrida tidak akan dapat menyegarkan token otorisasi karena tidak lagi terdaftar. SSM Agenthibernasi dan mengurangi frekuensi ping ke Systems Manager di cloud menjadi sekali per jam.

Anda dapat mendaftarkan kembali server lokal, perangkat edge, atau VM kapan saja. Systems Manager menyimpan riwayat perintah untuk node terkelola yang dideregistrasi selama 30 hari.

Prosedur berikut menjelaskan cara membatalkan pendaftaran node yang diaktifkan hibrida dengan menggunakan konsol Systems Manager. Untuk informasi tentang cara melakukan ini dengan menggunakanAWS Command Line Interface, lihat deregister-managed-instance.

Untuk membatalkan pendaftaran node yang diaktifkan hibrida (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tombol di sebelah node terkelola yang ingin Anda deregister.
4. Pilih tindakan Instance, Deregister instance terkelola ini.
5. Tinjau informasi di pop-up Membatalkan pendaftaran instans terkelola ini, lalu jika Anda menyetujuinya, pilih Batalkan pendaftaran.

Menggunakan pengaturan Konfigurasi Manajemen Host Default

Pengaturan Konfigurasi Manajemen Host Default memungkinkan AWS Systems Manager untuk mengelola instans Amazon EC2 Anda secara otomatis sebagai instans terkelola. Instans terkelola adalah instans EC2 yang dikonfigurasi untuk digunakan dengan Systems Manager.

Manfaat mengelola instans Anda dengan Systems Manager meliputi hal-hal berikut:

- Connect ke instans EC2 Anda dengan aman menggunakan. Session Manager
- Lakukan pemindaian patch otomatis menggunakanPatch Manager.
- Lihat informasi terperinci tentang instans Anda menggunakan Systems Manager Inventory.
- Lacak dan kelola instance menggunakanFleet Manager.
- Tetap SSM Agent up to date secara otomatis.

Fleet Manager, InventarisPatch Manager,, dan Session Manager merupakan kemampuan Systems Manager.

Konfigurasi Manajemen Host Default memungkinkan untuk mengelola instans EC2 tanpa Anda harus membuat profil instans AWS Identity and Access Management (IAM) secara manual. Sebagai gantinya, Konfigurasi Manajemen Host Default membuat dan menerapkan peran IAM default untuk memastikan bahwa Systems Manager memiliki izin untuk mengelola semua instance di Akun AWS dan di Wilayah AWS mana itu diaktifkan.

Jika izin yang diberikan tidak cukup untuk kasus penggunaan, Anda juga dapat menambahkan kebijakan ke peran IAM default yang dibuat oleh Konfigurasi Manajemen Host Default. Atau, jika Anda tidak memerlukan izin untuk semua kemampuan yang disediakan oleh peran IAM default, Anda dapat membuat peran dan kebijakan kustom Anda sendiri. Setiap perubahan yang dibuat pada peran IAM yang Anda pilih untuk Konfigurasi Manajemen Host Default berlaku untuk semua instans Amazon EC2 yang dikelola di Wilayah dan akun.

Untuk informasi selengkapnya tentang kebijakan yang digunakan oleh Konfigurasi Manajemen Host Default, lihatAWS kebijakan terkelola: AmazonsSMManageDEC2 InstanceDefaultPolicy.

Terapkan akses hak akses paling rendah

Prosedur ini dalam topik ini dimaksudkan untuk dilakukan hanya oleh administrator. Oleh karena itu, kami menyarankan untuk menerapkan akses hak istimewa terkecil untuk mencegah pengguna non-

administratif mengonfigurasi atau memodifikasi Konfigurasi Manajemen Host Default. Untuk melihat contoh kebijakan yang membatasi akses ke Konfigurasi Manajemen Host Default, lihat Contoh kebijakan hak istimewa paling sedikit untuk Konfigurasi Manajemen Host Default nanti dalam topik ini.

> **Important**
>
> Informasi pendaftaran untuk instance yang terdaftar menggunakan Konfigurasi Manajemen Host Default disimpan secara lokal di direktori `var/lib/amazon/ssm` atau`C:\ProgramData\Amazon`. Menghapus direktori ini atau file-file mereka akan mencegah instance memperoleh kredensi yang diperlukan untuk terhubung ke Systems Manager menggunakan Default Host Management Configuration. Dalam kasus ini, Anda harus menggunakan profil instans IAM untuk memberikan izin yang diperlukan untuk instans Anda, atau membuat ulang instance.

Topik

Prasyarat

Untuk menggunakan Konfigurasi Manajemen Host Default di Wilayah AWS dan Akun AWS di mana Anda mengaktifkan pengaturan, persyaratan berikut harus dipenuhi.

- Instance yang akan dikelola harus menggunakan Instance Metadata Service Version 2 (IMDSv2).

  Konfigurasi Manajemen Host Default tidak mendukung Layanan Metadata Instans Versi 1. Untuk informasi tentang transisi ke IMDSv2, lihat Transisi menggunakan Layanan Metadata Instans Versi 2 di Panduan Pengguna Amazon EC2 untuk Instans Linux
- SSM Agentversi 3.2.582.0 atau yang lebih baru harus diinstal pada instance yang akan dikelola.

  Untuk informasi tentang memeriksa versi yang SSM Agent diinstal pada instans Anda, lihatMemeriksa nomor SSM Agent versi.

  Untuk informasi tentang memperbaruiSSM Agent, lihatMemperbarui secara otomatis SSM Agent.

- Anda, sebagai administrator yang melakukan tugas dalam topik ini, harus memiliki izin untuk operasi GetServiceSettingResetServiceSetting, dan UpdateServiceSettingAPI. Selain itu, Anda harus memiliki izin untuk `iam:PassRole` izin untuk peran `AWSSystemsManagerDefaultEC2InstanceManagementRole` IAM. Berikut ini adalah contoh kebijakan yang menyediakan izin ini. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri.

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

- Jika profil instans IAM sudah dilampirkan ke instans EC2 yang akan dikelola menggunakan Systems Manager, Anda harus menghapus izin apa pun darinya yang memungkinkan operasi. `ssm:UpdateInstanceInformation` SSM Agentmencoba menggunakan izin profil instans sebelum menggunakan izin Konfigurasi Manajemen Host Default. Jika Anda mengizinkan

`ssm:UpdateInstanceInformation` operasi di profil instans IAM Anda sendiri, instans tidak akan menggunakan izin Konfigurasi Manajemen Host Default.

Mengaktifkan pengaturan Konfigurasi Manajemen Host Default

Anda dapat mengaktifkan Konfigurasi Manajemen Host Default dari Fleet Manager konsol, atau dengan menggunakan AWS Command Line Interface atau AWS Tools for Windows PowerShell.

Anda harus mengaktifkan Konfigurasi Manajemen Host Default satu per satu di setiap Wilayah tempat Anda ingin instans Amazon EC2 dikelola oleh setelan ini.

Setelah mengaktifkan Konfigurasi Manajemen Host Default, mungkin diperlukan waktu hingga 30 menit agar instans Anda menggunakan kredensi peran yang Anda pilih pada langkah 5 dalam prosedur berikut.

Untuk mengaktifkan Konfigurasi Manajemen Host Default (konsol)

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih Manajemen akun, Konfigurasi Konfigurasi Manajemen Host Default.
4. Aktifkan Aktifkan Konfigurasi Manajemen Host Default.
5. Pilih peran AWS Identity and Access Management (IAM) yang digunakan untuk mengaktifkan kemampuan Systems Manager untuk instans Anda. Sebaiknya gunakan peran default yang disediakan oleh Konfigurasi Manajemen Host Default. Ini berisi set izin minimum yang diperlukan untuk mengelola instans Amazon EC2 Anda menggunakan Systems Manager. Jika Anda lebih suka menggunakan peran khusus, kebijakan kepercayaan peran tersebut harus mengizinkan Systems Manager sebagai entitas tepercaya.
6. Pilih Konfigurasi untuk menyelesaikan penyiapan.

Untuk mengaktifkan Konfigurasi Manajemen Host Default (baris perintah)

1. Buat file JSON di mesin lokal Anda yang berisi kebijakan hubungan kepercayaan berikut.

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

2. Buka AWS CLI atau Tools untuk Windows PowerShell dan jalankan salah satu perintah berikut, tergantung pada jenis sistem operasi mesin lokal Anda, untuk membuat peran layanan di akun Anda. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   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. Jalankan perintah berikut untuk melampirkan kebijakan `AmazonSSMManagedEC2InstanceDefaultPolicy` terkelola ke peran yang baru dibuat. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   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. Buka AWS CLI or Tools untuk Windows PowerShell dan jalankan perintah berikut. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   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"
```

Tidak ada output jika perintah berhasil.

5. Jalankan perintah berikut untuk melihat pengaturan layanan saat ini untuk Konfigurasi Manajemen Host Default di saat ini Akun AWS dan Wilayah AWS.

Linux & macOS

```
aws ssm get-service-setting \
--setting-id arn:aws:ssm:region:account-id:servicesetting/ssm/managed-instance/
default-ec2-instance-management-role
```

Windows

```
aws ssm get-service-setting ^
--setting-id arn:aws:ssm:region:account-id:servicesetting/ssm/managed-instance/
default-ec2-instance-management-role
```

PowerShell

```
Get-SSMServiceSetting `
-SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/managed-instance/
default-ec2-instance-management-role"
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
{
    "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"
    }
}
```

Menonaktifkan pengaturan Konfigurasi Manajemen Host Default

Anda dapat menonaktifkan Konfigurasi Manajemen Host Default dari Fleet Manager konsol, atau dengan menggunakan AWS Command Line Interface atau AWS Tools for Windows PowerShell.

Anda harus menonaktifkan pengaturan Konfigurasi Manajemen Host Default satu per satu di setiap Wilayah di mana Anda tidak lagi ingin instans Amazon EC2 dikelola oleh konfigurasi ini. Menonaktifkannya di satu Wilayah tidak menonaktifkannya di semua Wilayah.

Jika Anda menonaktifkan Konfigurasi Manajemen Host Default, dan Anda belum melampirkan profil instans ke instans Amazon EC2 yang memungkinkan akses ke Systems Manager, mereka tidak akan lagi dikelola oleh Systems Manager.

Untuk menonaktifkan Konfigurasi Manajemen Host Default (konsol)

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih Manajemen akun, Konfigurasi Manajemen Host Default.
4. Matikan Aktifkan Konfigurasi Manajemen Host Default.
5. Pilih Konfigurasi untuk menonaktifkan Konfigurasi Manajemen Host Default.

Untuk menonaktifkan Konfigurasi Manajemen Host Default (baris perintah)

- Buka AWS CLI or Tools untuk Windows PowerShell dan jalankan perintah berikut. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

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"
```

Contoh kebijakan hak istimewa paling sedikit untuk Konfigurasi Manajemen Host Default

Contoh kebijakan berikut menunjukkan cara mencegah anggota organisasi Anda membuat perubahan pada setelan Konfigurasi Manajemen Host Default di akun Anda.

Kebijakan kontrol layanan untuk AWS Organizations

Kebijakan berikut menunjukkan cara mencegah anggota non-administratif di Anda AWS Organizations memperbarui setelan Konfigurasi Manajemen Host Default Anda. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

```
{
    "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"
                ]
                }
            }
        }
    ]
```

```
}
```

Kebijakan untuk kepala sekolah IAM

Kebijakan berikut menunjukkan cara mencegah grup, peran, atau pengguna IAM dalam memperbarui AWS Organizations setelan Konfigurasi Manajemen Host Default. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "ssm:UpdateServiceSetting",
                "ssm:ResetServiceSetting"
            ],
            "Resource": "arn:aws:ssm:region:account-id:servicesetting/ssm/managed-
instance/default-ec2-instance-management-role"
        },
        {
            "Effect": "Deny",
            "Action": [
                "iam:AttachRolePolicy",
                "iam:DeleteRole",
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::account-id:role/service-role/
AWSSystemsManagerDefaultEC2InstanceManagementRole"
        }
    ]
}
```

Menghubungkan ke instance Windows Server terkelola menggunakan Remote Desktop

Anda dapat menggunakanFleet Manager, kemampuan AWS Systems Manager, untuk menyambung ke instans Windows Server Amazon Elastic Compute Cloud (Amazon EC2) menggunakan (RDP). Remote Desktop Protocol Fleet Manager Remote Desktop, yang didukung oleh NICE DCV, memberi Anda konektivitas aman ke Windows Server instans langsung dari konsol Systems Manager. Anda dapat memiliki hingga empat koneksi simultan dalam satu jendela browser.

Saat ini, Anda hanya dapat menggunakan Remote Desktop dengan instance yang menjalankan Windows Server 2012 RTM atau lebih tinggi. Remote Desktop hanya mendukung input bahasa Inggris.

> **Note**
>
> Fleet ManagerRemote Desktop adalah layanan khusus konsol dan tidak mendukung koneksi baris perintah ke instance terkelola Anda. Untuk terhubung ke instance Windows Server terkelola melalui shell, Anda dapat menggunakanSession Manager, kemampuan lain dari AWS Systems Manager. Untuk informasi selengkapnya, lihat AWS Systems Manager Session Manager.

Untuk informasi tentang mengkonfigurasi izin AWS Identity and Access Management (IAM) agar instans Anda berinteraksi dengan Systems Manager, lihat Mengonfigurasi izin instans untuk Systems Manager.

Topik

- Menyiapkan lingkungan Anda
- Mengkonfigurasi izin IAM untuk Remote Desktop
- Mengautentikasi koneksi Remote Desktop
- Durasi koneksi jarak jauh dan konkurensi
- Connect ke node terkelola menggunakan Remote Desktop

Menyiapkan lingkungan Anda

Sebelum menggunakan Remote Desktop, verifikasi bahwa lingkungan Anda memenuhi persyaratan berikut:

- Konfigurasi simpul terkelola

  Pastikan instans Amazon EC2 Anda dikonfigurasi sebagai node terkelola di Systems Manager.

- SSM Agentversi minimum

  Verifikasi bahwa node menjalankan SSM Agent versi 3.0.222.0 atau lebih tinggi. Untuk informasi tentang cara memeriksa versi agen mana yang berjalan pada node, lihatMemeriksa nomor SSM

Agent versi. Untuk informasi tentang menginstal atau memperbaruiSSM Agent, lihatBekerja dengan SSM Agent.

- Konfigurasi port RDP

  Untuk menerima koneksi jarak jauh, Remote Desktop Services layanan pada Windows Server node Anda harus menggunakan port RDP default 3389. Ini adalah konfigurasi default on Amazon Machine Images (AMIs) yang disediakan oleh AWS. Anda tidak secara eksplisit diharuskan untuk membuka port masuk apa pun untuk menggunakan Remote Desktop.

- PSReadLineversi modul untuk fungsionalitas keyboard

  Untuk memastikan keyboard Anda berfungsi dengan benarPowerShell, verifikasi bahwa node yang menjalankan Windows Server 2022 memiliki PSReadLine modul versi 2.2.2 atau lebih tinggi diinstal. Jika mereka menjalankan versi yang lebih lama, Anda dapat menginstal versi yang diperlukan menggunakan perintah berikut.

  ```
  Install-Module `
      -Name PSReadLine `
      -Repository PSGallery `-MinimumVersion 2.2.2
  ```

- Konfigurasi Manajer Sesi

  Sebelum Anda dapat menggunakan Remote Desktop, Anda harus menyelesaikan prasyarat untuk pengaturan Session Manager. Saat Anda terhubung ke instans menggunakan Remote Desktop, preferensi sesi apa pun yang ditentukan untuk Anda Akun AWS dan Wilayah AWS diterapkan. Untuk informasi selengkapnya, lihat Menyiapkan Session Manager.

  > **Note**
  >
  > Jika Anda mencatat aktivitas Session Manager menggunakan Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3), maka koneksi Remote Desktop Anda akan menghasilkan kesalahan berikut. `bucket_name/Port/stderr` Kesalahan ini adalah perilaku yang diharapkan dan dapat diabaikan dengan aman.
  >
  > ```
  > 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>
```

Mengkonfigurasi izin IAM untuk Remote Desktop

Selain izin IAM yang diperlukan untuk Systems Manager danSession Manager, pengguna atau peran yang Anda gunakan untuk mengakses konsol harus mengizinkan tindakan berikut:

- `ssm-guiconnect:CancelConnection`
- `ssm-guiconnect:GetConnection`
- `ssm-guiconnect:StartConnection`

Berikut ini adalah contoh kebijakan IAM yang dapat Anda lampirkan ke pengguna atau peran untuk memungkinkan berbagai jenis interaksi dengan Remote Desktop. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri.

Kebijakan standar untuk menghubungkan ke instans EC2

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "EC2",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:GetPasswordData"
            ],
            "Resource": "*"
        },
        {
            "Sid": "SSM",
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeInstanceProperties",
                "ssm:GetCommandInvocation",
                "ssm:GetInventorySchema"
            ],
            "Resource": "*"
        },
        {
```

```
            "Sid": "TerminateSession",
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/aws:ssmmessages:session-id": [
                        "${aws:userid}"
                    ]
                }
            }
        },
        {
            "Sid": "SSMStartSession",
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:*:account-id:instance/*",
                "arn:aws:ssm:*:account-id:managed-instance/*",
                "arn:aws:ssm:*::document/AWS-StartPortForwardingSession"
            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck": "true"
                },
                "ForAnyValue:StringEquals": {
                    "aws:CalledVia": "ssm-guiconnect.amazonaws.com"
                }
            }
        },
        {
            "Sid": "GuiConnect",
            "Effect": "Allow",
            "Action": [
                "ssm-guiconnect:CancelConnection",
                "ssm-guiconnect:GetConnection",
                "ssm-guiconnect:StartConnection"
            ],
            "Resource": "*"
        }
```

```
    ]
}
```

Kebijakan untuk menghubungkan ke instans EC2 dengan tag tertentu

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "EC2",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:GetPasswordData"
            ],
            "Resource": "*"
        },
        {
            "Sid": "SSM",
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeInstanceProperties",
                "ssm:GetCommandInvocation",
                "ssm:GetInventorySchema"
            ],
            "Resource": "*"
        },
        {
            "Sid": "SSMStartSession",
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ssm:*::document/AWS-StartPortForwardingSession"
            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck": "true"
                },
                "ForAnyValue:StringEquals": {
                    "aws:CalledVia": "ssm-guiconnect.amazonaws.com"
                }
```

```
            }
        },
        {
            "Sid": "AccessTaggedInstances",
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:*:account-id:instance/*",
                "arn:aws:ssm:*:account-id:managed-instance/*"
            ],
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/tag key": [
                        "tag value"
                    ]
                }
            }
        },
        {
            "Sid": "GuiConnect",
            "Effect": "Allow",
            "Action": [
                "ssm-guiconnect:CancelConnection",
                "ssm-guiconnect:GetConnection",
                "ssm-guiconnect:StartConnection"
            ],
            "Resource": "*"
        }
    ]
}
```

Kebijakan bagi AWS IAM Identity Center pengguna untuk terhubung ke instans EC2

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "SSO",
            "Effect": "Allow",
            "Action": [
                "sso:ListDirectoryAssociations*",
```

```
                "identitystore:DescribeUser"
            ],
            "Resource": "*"
        },
        {
            "Sid": "EC2",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:GetPasswordData"
            ],
            "Resource": "*"
        },
        {
            "Sid": "SSM",
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeInstanceProperties",
                "ssm:GetCommandInvocation",
                "ssm:GetInventorySchema"
            ],
            "Resource": "*"
        },
        {
            "Sid": "TerminateSession",
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/aws:ssmmessages:session-id": [
                        "${aws:userName}"
                    ]
                }
            }
        },
        {
            "Sid": "SSMStartSession",
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
```

```
            "Resource": [
                "arn:aws:ec2:*:*:instance/*",
                "arn:aws:ssm:*:*:managed-instance/*",
                "arn:aws:ssm:*:*:document/AWS-StartPortForwardingSession"
            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck": "true"
                },
                "ForAnyValue:StringEquals": {
                    "aws:CalledVia": "ssm-guiconnect.amazonaws.com"
                }
            }
        },
        {
            "Sid": "SSMSendCommand",
            "Effect": "Allow",
            "Action": [
                "ssm:SendCommand"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:instance/*",
                "arn:aws:ssm:*:*:managed-instance/*",
                "arn:aws:ssm:*:*:document/AWSSSO-CreateSSOUser"
            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck": "true"
                }
            }
        },
        {
            "Sid": "GuiConnect",
            "Effect": "Allow",
            "Action": [
                "ssm-guiconnect:CancelConnection",
                "ssm-guiconnect:GetConnection",
                "ssm-guiconnect:StartConnection"
            ],
            "Resource": "*"
        }
    ]
}
```

Mengautentikasi koneksi Remote Desktop

Saat membuat koneksi jarak jauh, Anda dapat mengautentikasi menggunakan Windows kredensional atau key pair `.pem` (file) Amazon EC2 yang terkait dengan instance. Untuk informasi tentang penggunaan pasangan kunci, lihat pasangan kunci Amazon EC2 dan Windows instans di Panduan Pengguna Amazon EC2 untuk Instans Windows.

Atau, jika Anda diautentikasi ke AWS Management Console penggunaan AWS IAM Identity Center, Anda dapat terhubung ke instans Anda tanpa memberikan kredensi tambahan. Untuk contoh kebijakan untuk mengizinkan otentikasi koneksi jarak jauh menggunakan IAM Identity Center, lihat. Mengkonfigurasi izin IAM untuk Remote Desktop

Sebelum Anda mulai

Perhatikan kondisi berikut untuk menggunakan autentikasi IAM Identity Center sebelum Anda mulai menghubungkan menggunakan Remote Desktop.

- Remote Desktop mendukung autentikasi IAM Identity Center untuk node yang sama di Wilayah AWS mana Anda mengaktifkan IAM Identity Center.
- Remote Desktop mendukung nama pengguna IAM Identity Center hingga 16 karakter.
- Remote Desktop mendukung nama pengguna IAM Identity Center yang terdiri dari karakter alfanumerik dan karakter khusus berikut: `. - _`

> **Important**
>
> Koneksi tidak akan berhasil untuk nama pengguna IAM Identity Center yang berisi karakter berikut: `+ = , @`
> IAM Identity Center mendukung karakter ini dalam nama pengguna, tetapi koneksi Fleet Manager RDP tidak.

- Ketika koneksi diautentikasi menggunakan IAM Identity Center, Remote Desktop membuat Windows pengguna lokal dalam grup Administrator Lokal instans. Pengguna ini bertahan setelah koneksi jarak jauh berakhir.
- Remote Desktop tidak mengizinkan autentikasi IAM Identity Center untuk node yang merupakan pengontrol Microsoft Active Directory domain.
- Meskipun Remote Desktop memungkinkan Anda untuk menggunakan autentikasi IAM Identity Center untuk node yang bergabung ke Active Directory domain, kami tidak menyarankan

melakukannya. Metode otentikasi ini memberikan izin administratif kepada pengguna yang mungkin mengganti izin yang lebih ketat yang diberikan oleh domain.

Wilayah yang Didukung untuk autentikasi Pusat Identitas IAM

Remote Desktopkoneksi menggunakan autentikasi IAM Identity Center didukung sebagai berikut: Wilayah AWS

- AS Timur (Ohio) (us-east-2)
- AS Timur (Virginia Utara) (us-east-1)
- AS Barat (California Utara) (us-west-1)
- AS Barat (Oregon) (us-west-2)
- Africa (Cape Town) (af-south-1)
- Asia Pacific (Hong Kong) (ap-east-1)
- Asia Pasifik (Mumbai) (ap-south-1)
- Asia Pasifik (Tokyo) (ap-northeast-1)
- Asia Pasifik (Seoul) (ap-northeast-2)
- Asia Pasifik (Osaka) (ap-northeast-3)
- Asia Pasifik (Singapura) (ap-southeast-1)
- Asia Pasifik (Sydney) (ap-southeast-2)
- Asia Pasifik (Jakarta) (ap-tenggara 3)
- Kanada (Pusat) (ca-central-1)
- Eropa (Frankfurt) (eu-central-1)
- Eropa (Stockholm) (eu-north-1)
- Eropa (Irlandia) (eu-west-1)
- Eropa (London) (eu-west-2)
- Eropa (Paris) (eu-west-3)
- Israel (Tel Aviv) (tengah-1)
- Amerika Selatan (São Paulo) (sa-east-1)
- Europe (Milan) (eu-south-1)
- Middle East (Bahrain) (me-south-1)

- AWS GovCloud (AS-Timur) (us-gov-east-1)
- AWS GovCloud (AS-Barat) (us-gov-west-1)

Durasi koneksi jarak jauh dan konkurensi

Ketentuan berikut berlaku untuk koneksi Remote Desktop yang aktif:

- Durasi koneksi

  Secara default, koneksi Remote Desktop terputus setelah 60 menit. Untuk mencegah koneksi terputus, Anda dapat memilih Perbarui sesi sebelum terputus untuk mengatur ulang pengatur waktu durasi.
- Batas waktu koneksi

  Koneksi Remote Desktop terputus setelah idle selama lebih dari 10 menit.
- Koneksi bersamaan

  Secara default, Anda dapat memiliki maksimum 5 koneksi Remote Desktop aktif pada satu waktu untuk yang sama Akun AWS dan Wilayah AWS. Untuk meminta peningkatan kuota layanan hingga 25 koneksi bersamaan, lihat Meminta peningkatan kuota dalam Panduan Pengguna Service Quotas.

Connect ke node terkelola menggunakan Remote Desktop

Dukungan salin/tempel browser untuk teks

Menggunakan browser Google Chrome dan Microsoft Edge, Anda dapat menyalin dan menempelkan teks dari node terkelola ke mesin lokal Anda, dan dari mesin lokal Anda ke node terkelola yang terhubung dengan Anda.

Menggunakan browser Mozilla Firefox, Anda dapat menyalin dan menempelkan teks dari node terkelola ke mesin lokal Anda saja. Menyalin dari mesin lokal Anda ke node terkelola tidak didukung.

Untuk terhubung ke node terkelola menggunakan Fleet Manager Remote Desktop

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.

3. Pilih node yang ingin Anda sambungkan. Anda dapat memilih kotak centang atau nama simpul.
4. Pada menu tindakan Node, pilih Connect with Remote Desktop.
5. Pilih jenis Otentikasi pilihan Anda. Jika Anda memilih kredensi pengguna, masukkan nama pengguna dan kata sandi untuk akun Windows pengguna pada node yang Anda sambungkan. Jika Anda memilih Pasangan kunci, Anda dapat memberikan otentikasi menggunakan salah satu metode berikut:

   a. Pilih Jelajahi mesin lokal jika Anda ingin memilih kunci PEM yang terkait dengan instance Anda dari sistem file lokal Anda.

      - atau -

   b. Pilih Tempel konten key pair jika Anda ingin menyalin konten file PEM dan menempelkannya ke bidang yang disediakan.
6. Pilih Connect.
7. Untuk memilih resolusi tampilan yang Anda inginkan, di menu Tindakan, pilih Resolusi, lalu pilih dari yang berikut ini:

   - Beradaptasi secara otomatis
   - 1920 x 1080
   - 1400 x 900
   - 1366 x 768
   - 800 x 600

   Opsi Adaptasi Secara Otomatis menetapkan resolusi berdasarkan ukuran layar yang terdeteksi.

Mengelola volume Amazon EBS pada instans terkelola

Amazon Elastic Block Store (Amazon EBS) menyediakan volume penyimpanan tingkat blok untuk digunakan dengan instans Amazon Elastic Compute Cloud (EC2). Volume EBS berfungsi seperti perangkat blok mentah yang tidak terformat. Anda dapat memasang volume ini sebagai perangkat di instans Anda.

Anda dapat menggunakanFleet Manager, kemampuan AWS Systems Manager, untuk mengelola volume Amazon EBS pada instans terkelola Anda. Misalnya, Anda dapat menginisialisasi volume EBS, memformat partisi, dan memasang volume agar tersedia untuk digunakan.

> **Note**
>
> Fleet Managersaat ini mendukung manajemen volume Amazon EBS hanya untuk Windows Server instans.

Lihat detail volume EBS

Untuk melihat detail volume EBS dengan Fleet Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tombol di sebelah instance terkelola yang ingin Anda lihat detail volume EBS.
4. Pilih View details (Lihat detail).
5. Pilih Alat, volume EBS.
6. Untuk melihat detail volume EBS, pilih ID-nya di kolom Volume ID.

Inisialisasi dan format volume EBS

Untuk menginisialisasi dan memformat volume EBS dengan Fleet Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.

3. Pilih tombol di sebelah instance terkelola yang ingin Anda inisialisasi, format, dan pasang volume EBS. Anda hanya dapat menginisialisasi volume EBS jika disknya kosong.
4. Pilih View details (Lihat detail).
5. Di menu Tools, pilih volume EBS.
6. Pilih tombol di sebelah volume EBS yang ingin Anda inisialisasi dan format.
7. Pilih Inisialisasi dan format.
8. Dalam gaya Partisi, pilih gaya partisi yang ingin Anda gunakan untuk volume EBS.
9. (Opsional) Pilih huruf Drive untuk partisi.
10. (Opsional) Masukkan nama Partisi untuk mengidentifikasi partisi.
11. Pilih sistem File yang akan digunakan untuk mengatur file dan data yang disimpan di partisi.
12. Pilih Konfirmasi untuk membuat volume EBS tersedia untuk digunakan. Anda tidak dapat mengubah konfigurasi partisi dari AWS Management Console setelah konfirmasi, namun, Anda dapat menggunakan SSH atau RDP untuk masuk ke instance untuk mengubah konfigurasi partisi.

Bekerja dengan sistem file

Anda dapat menggunakanFleet Manager, kemampuanAWS Systems Manager, untuk bekerja dengan sistem file pada node terkelola Anda. Dengan menggunakanFleet Manager, Anda dapat melihat informasi tentang direktori dan data file yang disimpan pada volume yang dilampirkan ke node terkelola Anda. Misalnya, Anda dapat melihat nama, ukuran, ekstensi, pemilik, dan izin untuk direktori dan file Anda. Hingga 10.000 baris data file dapat dipratinjau sebagai teks dari Fleet Manager konsol. Anda juga dapat menggunakan fitur ini untuk `tail` file. Saat menggunakan `tail` untuk melihat data file, 10 baris terakhir dari file ditampilkan pada awalnya. Saat baris data baru ditulis ke file, tampilan diperbarui secara real time. Maka, Anda dapat meninjau data log dari konsol tersebut, yang dapat meningkatkan efisiensi pemecahan masalah dan administrasi sistem Anda. Selain itu, Anda dapat membuat direktori dan menyalin, memotong, menempel, mengganti nama, atau menghapus file dan direktori.

Sebaiknya buat backup reguler, atau ambil snapshot dari volume Amazon Elastic Block Store (Amazon EBS) Elastic Block Store (Amazon EBS) yang dilampirkan ke node terkelola Anda. Saat

menyalin, atau memotong dan menempelkan file, file dan direktori yang ada di jalur tujuan dengan nama yang sama dengan file atau direktori baru diganti. Masalah serius dapat terjadi jika Anda mengganti atau memodifikasi file sistem dan direktori. AWStidak menjamin bahwa masalah ini dapat diselesaikan. Ubah file sistem dengan risiko Anda sendiri. Anda bertanggung jawab atas semua perubahan file dan direktori, dan memastikan Anda memiliki cadangan. Menghapus atau mengganti file dan direktori tidak dapat dibatalkan.

> **Note**
>
> Fleet ManagermenggunakanSession Manager, kemampuanAWS Systems Manager, untuk melihat pratinjau teks dan `tail` file. Untuk instans Amazon Elastic Compute Cloud (Amazon EC2), profil instans yang dilampirkan ke instans terkelola harus memberikan izin untuk menggunakan fitur ini. Session Manager Untuk informasi selengkapnya tentang menambahkan Session Manager izin ke profil instans, lihatMenambahkanSession Managerizin untuk peran IAM yang ada. Selain itu, enkripsi AWS Key Management Service (AWS KMS) harus diaktifkan di preferensi sesi Anda untuk menggunakan Fleet Manager fitur. Untuk informasi selengkapnya tentang mengaktifkan AWS KMS enkripsiSession Manager, lihatAktifkan enkripsi kunci KMS data sesi (konsol).

Untuk melihat sistem file dengan Fleet Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tautan node terkelola dengan sistem file yang ingin Anda lihat.
4. Pilih Alat, Sistem file.

Untuk melihat pratinjau teks file dengan Fleet Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

-atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.

3. Pilih tautan node terkelola dengan file yang ingin Anda pratinjau.
4. Pilih Alat, Sistem file.
5. Pilih nama File dari direktori yang berisi file yang ingin Anda pratinjau.
6. Pilih tombol di samping file yang kontennya ingin Anda pratinjau.
7. Pilih Tindakan, Pratinjau sebagai teks.

Untuk mengekor file dengan Fleet Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

-atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.

3. Pilih tautan node terkelola dengan file yang ingin Anda ekor.
4. Pilih Alat, Sistem file.
5. Pilih nama File dari direktori yang berisi file yang ingin Anda ekor.
6. Pilih tombol di samping file yang kontennya ingin Anda ikuti.
7. Pilih Actions, Tail file.

Untuk menyalin atau memotong dan menempelkan file atau direktori dengan Fleet Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

-atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.

3. Pilih tautan node terkelola dengan file yang ingin Anda salin, atau potong dan tempel.
4. Pilih Alat, Sistem file.
5. Untuk menyalin atau memotong file, pilih Nama file direktori yang berisi file yang ingin Anda salin atau potong. Untuk menyalin atau memotong direktori, pilih tombol di sebelah direktori yang ingin Anda salin atau potong dan kemudian lanjutkan ke langkah 8.
6. Pilih tombol di sebelah file yang ingin Anda salin atau potong.
7. Di menu Tindakan, pilih Salin atau Potong.
8. Dalam tampilan Sistem file, pilih tombol di sebelah direktori tempat Anda ingin menempelkan file.
9. Di menu Tindakan, pilih Tempel.

Untuk mengganti nama file atau direktori dengan Fleet Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tautan node terkelola dengan file atau direktori yang ingin Anda ganti namanya.
4. Pilih Alat, Sistem file.
5. Untuk mengganti nama file, pilih nama File dari direktori yang berisi file yang ingin Anda ganti nama. Untuk mengganti nama direktori, pilih tombol di sebelah direktori yang ingin Anda ganti nama dan kemudian lanjutkan ke langkah 8.
6. Pilih tombol di sebelah file yang kontennya ingin Anda ganti namanya.
7. Pilih Tindakan, Ganti Nama.
8. Untuk Nama file, masukkan nama baru untuk file tersebut dan pilih Ganti nama.

Untuk menghapus file atau direktori dengan Fleet Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tautan node terkelola dengan file atau direktori yang ingin Anda hapus.
4. Pilih Alat, Sistem file.
5. Untuk menghapus file, pilih Nama file direktori yang berisi file yang ingin Anda hapus. Untuk menghapus direktori, pilih tombol di sebelah direktori yang ingin Anda hapus dan kemudian lanjutkan ke langkah 7.
6. Pilih tombol di sebelah file dengan konten yang ingin Anda hapus.
7. Pilih Tindakan, Hapus.

Untuk membuat direktori dengan Fleet Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tautan node terkelola tempat Anda ingin membuat direktori.
4. Pilih Alat, Sistem file.
5. Pilih nama file direktori tempat Anda ingin membuat direktori baru.
6. Pilih Buat direktori.
7. Untuk nama Direktori, masukkan nama untuk direktori baru, lalu pilih Buat direktori.

Memantau kinerja node terkelola

Anda dapat menggunakanFleet Manager, kemampuan AWS Systems Manager, untuk melihat data kinerja tentang node terkelola Anda secara real time. Data performa diambil dari pengukur performa.

Penghitung kinerja berikut tersedia diFleet Manager:

- Penggunaan CPU
- Penggunaan input/output disk (I/O)
- Lalu lintas jaringan
- Penggunaan memori

> **Note**
>
> Fleet ManagermenggunakanSession Manager, kemampuan AWS Systems Manager, untuk mengambil data kinerja. Untuk instans Amazon Elastic Compute Cloud (Amazon EC2), profil instans yang dilampirkan ke instans terkelola harus memberikan izin untuk menggunakan fitur ini. Session Manager Untuk informasi selengkapnya tentang menambahkan Session Manager izin ke profil instans, lihatMenambahkanSession Managerizin untuk peran IAM yang ada. Selain itu, enkripsi AWS Key Management Service (AWS KMS) harus diaktifkan di preferensi sesi Anda untuk menggunakan Fleet Manager fitur. Untuk informasi selengkapnya tentang mengaktifkan AWS KMS enkripsiSession Manager, lihatAktifkan enkripsi kunci KMS data sesi (konsol).

Untuk melihat data kinerja dengan Fleet Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tombol di sebelah node terkelola yang kinerjanya ingin Anda pantau.
4. Pilih View details (Lihat detail).

5. Pilih Alat, Penghitung kinerja.

Bekerja dengan proses

Anda dapat menggunakanFleet Manager, kemampuanAWS Systems Manager, untuk bekerja dengan proses pada instans terkelola Anda. MenggunakanFleet Manager, Anda dapat melihat informasi tentang proses. Misalnya, Anda dapat melihat pemanfaatan CPU dan penggunaan memori proses selain pegangan dan utasnya. DenganFleet Manager, Anda dapat memulai dan menghentikan proses dari konsol.

> **Note**
>
> Fleet ManagermenggunakanSession Manager, kemampuanAWS Systems Manager, untuk mengambil data proses. Untuk instans Amazon Elastic Compute Cloud (Amazon EC2), profil instans yang dilampirkan ke instans terkelola harus memberikan izin untuk menggunakan fitur ini. Session Manager Untuk informasi selengkapnya tentang menambahkan Session Manager izin ke profil instans, lihatMenambahkanSession Managerizin untuk peran IAM yang ada. Selain itu, enkripsi AWS Key Management Service (AWS KMS) harus diaktifkan di preferensi sesi Anda untuk menggunakan Fleet Manager fitur. Untuk informasi selengkapnya tentang mengaktifkan AWS KMS enkripsiSession Manager, lihatAktifkan enkripsi kunci KMS data sesi (konsol).

Untuk melihat detail tentang proses dengan Fleet Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tautan instance yang prosesnya ingin Anda lihat.
4. Pilih Alat, Proses.

Untuk memulai proses dengan Fleet Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tautan instance tempat Anda ingin memulai proses.
4. Pilih Alat, Proses.
5. Pilih Mulai proses baru.
6. Untuk nama Proses atau jalur lengkap, masukkan nama proses atau jalur lengkap ke executable.
7. (Opsional) Untuk direktori Kerja, masukkan jalur direktori tempat Anda ingin proses dijalankan.

Untuk mengakhiri proses dengan Fleet Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tautan instance tempat Anda ingin memulai proses.
4. Pilih Alat, Proses.
5. Pilih tombol di sebelah proses yang ingin Anda hentikan.
6. Pilih Tindakan, Hentikan proses atau Tindakan, Hentikan pohon proses.

> **Note**
>
> Mengakhiri pohon proses juga mengakhiri semua proses dan aplikasi menggunakan proses itu.

Melihat log pada node terkelola

Anda dapat menggunakanFleet Manager, kemampuan AWS Systems Manager, untuk melihat data log yang disimpan di node terkelola Anda. Untuk node yang dikelola Windows, Anda dapat melihat log peristiwa Windows dan menyalin detailnya dari konsol. Untuk membantu Anda mencari peristiwa, filter log peristiwa Windows menurut Tingkat peristiwa, ID peristiwa, Sumber peristiwa, dan Waktu pembuatan. Anda juga dapat melihat data log lainnya menggunakan prosedur untuk melihat sistem file. Untuk informasi selengkapnya tentang melihat sistem file denganFleet Manager, lihatBekerja dengan sistem file.

Untuk melihat log peristiwa Windows dengan Fleet Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tombol di sebelah node terkelola yang log peristiwanya ingin Anda lihat.
4. Pilih View details (Lihat detail).
5. Pilih Alat, log peristiwa Windows.
6. Pilih Nama log yang berisi peristiwa yang ingin Anda lihat.
7. Pilih tombol di sebelah Nama log yang ingin Anda lihat, kemudian pilih Lihat peristiwa.
8. Pilih tombol di sebelah peristiwa yang ingin Anda lihat, kemudian pilih Lihat detail peristiwa.
9. (Opsional) Pilih Saling sebagai JSON untuk menyalin detail peristiwa ke clipboard Anda.

Mengelola akun pengguna OS pada node terkelola

Anda dapat menggunakanFleet Manager, kemampuan AWS Systems Manager, untuk mengelola akun pengguna sistem operasi (OS) pada node terkelola Anda. Misalnya, Anda dapat membuat dan menghapus pengguna dan grup. Selain itu, Anda dapat melihat detail seperti keanggotaan grup, peran pengguna, dan status.

Important

Fleet Managerpenggunaan Run Command danSession Manager, kemampuan AWS Systems Manager, untuk berbagai operasi manajemen pengguna. Sebagai hasilnya, pengguna dapat memberikan izin ke akun pengguna sistem operasi yang tidak dapat mereka berikan. Ini karena AWS Systems Manager Agent (SSM Agent) berjalan di Amazon Elastic Compute Cloud (Amazon EC2) instance menggunakan izin root (Linux) atau izin SYSTEM (Windows Server). Untuk informasi selengkapnya tentang membatasi akses ke perintah tingkat root, lihatSSM Agent. Membatasi akses ke perintah tingkat root melaluiSSM Agent Untuk membatasi akses ke fitur ini, sebaiknya buat kebijakan AWS Identity and Access Management (IAM) untuk pengguna Anda yang hanya mengizinkan akses ke tindakan yang Anda tentukan. Untuk informasi selengkapnya tentang membuat kebijakan IAMFleet Manager, lihatLangkah 1: Buat kebijakan IAM dengan izin Fleet Manager.

Membuat pengguna atau grup

Note

Fleet Managerdigunakan Session Manager untuk mengatur kata sandi untuk pengguna baru. Untuk instans Amazon EC2, profil instans yang dilampirkan ke instans terkelola Anda harus memberikan izin untuk Session Manager menggunakan fitur ini. Untuk informasi selengkapnya tentang menambahkan Session Manager izin ke profil instans, lihatMenambahkanSession Managerizin untuk peran IAM yang ada. Selain itu, enkripsi AWS Key Management Service (AWS KMS) harus diaktifkan di preferensi sesi Anda untuk menggunakan Fleet Manager fitur. Untuk informasi selengkapnya tentang mengaktifkan AWS KMS enkripsiSession Manager, lihatAktifkan enkripsi kunci KMS data sesi (konsol).

Untuk membuat akun pengguna OS dengan Fleet Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.

3. Pilih tombol di sebelah node terkelola yang ingin Anda buat pengguna baru.
4. Pilih View details (Lihat detail).
5. Pilih Alat, Pengguna, dan grup.
6. Pilih tab Pengguna, dan kemudian pilih Buat pengguna.
7. Masukkan nilai untuk Nama pengguna baru tersebut.
8. (Disarankan) Pilih kotak centang di samping Atur kata sandi. Anda akan diminta untuk memberikan kata sandi untuk pengguna baru di akhir prosedur.
9. Pilih Buat pengguna. Jika Anda memilih kotak centang untuk membuat kata sandi bagi pengguna baru, Anda akan diminta memasukkan nilai untuk kata sandi lalu pilih Selesai. Jika kata sandi yang Anda tentukan tidak memenuhi persyaratan yang ditentukan oleh kebijakan lokal atau domain node terkelola, kesalahan akan ditampilkan.

Untuk membuat grup OS dengan Fleet Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tombol di sebelah node terkelola tempat Anda ingin membuat grup.
4. Pilih View details (Lihat detail).
5. Pilih Alat, Pengguna, dan grup.
6. Pilih tab Grup, dan kemudian pilih Buat grup.
7. Masukkan nilai untuk Nama grup baru tersebut.
8. (Opsional) Masukkan nilai untuk Deskripsi grup baru tersebut.
9. (Opsional) Pilih pengguna untuk ditambahkan ke Anggota grup untuk grup baru tersebut.
10. Pilih Buat grup.

Memperbarui keanggotaan pengguna atau grup

Untuk menambahkan akun pengguna OS ke grup baru dengan Fleet Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tombol di sebelah node terkelola tempat akun pengguna ada yang ingin Anda perbarui.
4. Pilih View details (Lihat detail).
5. Pilih Alat, Pengguna, dan grup.
6. Pilih tab Pengguna.
7. Pilih tombol di samping pengguna yang ingin Anda perbarui.
8. Pilih Tindakan, Tambahkan pengguna ke grup.
9. Pilih grup yang ingin Anda tambahkan pengguna di bawah Tambahkan ke grup.
10. Pilih Tambahkan pengguna ke grup.

Untuk mengedit keanggotaan grup OS dengan Fleet Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tombol di sebelah node terkelola tempat grup ada yang ingin Anda perbarui.
4. Pilih View details (Lihat detail).
5. Pilih Alat, Pengguna, dan grup.
6. Pilih tab Grup.

7. Pilih tombol di samping grup yang ingin Anda perbarui.
8. Pilih Tindakan, Ubah grup.
9. Pilih pengguna yang ingin Anda tambahkan atau hapus di bawah Anggota grup.
10. Pilih Ubah grup.

Menghapus pengguna atau grup

Untuk menghapus akun pengguna OS dengan Fleet Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tombol di sebelah node terkelola tempat akun pengguna ada yang ingin Anda hapus.
4. Pilih View details (Lihat detail).
5. Pilih, Pengguna dan grup.
6. Pilih tab Pengguna.
7. Pilih tombol di samping pengguna yang ingin Anda hapus.
8. Pilih Tindakan, Hapus pengguna lokal.

Untuk menghapus grup OS dengan Fleet Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tombol di sebelah node terkelola tempat grup ada yang ingin Anda hapus.

4. Pilih View details (Lihat detail).
5. Pilih Alat, Pengguna, dan grup.
6. Pilih tab Grup.
7. Pilih tombol di samping grup yang ingin Anda perbarui.
8. Pilih Tindakan, Hapus grup lokal.

Mengelola registri Windows pada node terkelola

Anda dapat menggunakanFleet Manager, kemampuan AWS Systems Manager, untuk mengelola registri pada node Windows Server terkelola Anda. Dari Fleet Manager konsol Anda dapat membuat, menyalin, memperbarui, dan menghapus entri dan nilai registri.

> **Important**
>
> Sebaiknya buat cadangan registri, atau ambil snapshot dari volume root Amazon Elastic Block Store (Amazon EBS) Elastic Block Store (Amazon EBS) yang dilampirkan ke node terkelola Anda, sebelum Anda memodifikasi registri. Masalah serius dapat terjadi jika Anda salah mengubah registri. Masalah-masalah ini mungkin mengharuskan Anda untuk menginstal ulang sistem operasi, atau mengembalikan volume root node Anda dari snapshot. AWS tidak menjamin bahwa masalah ini dapat diselesaikan. Ubah registri dengan risiko Anda sendiri. Anda bertanggung jawab atas semua perubahan registri, dan memastikan Anda memiliki backup.

Membuat kunci atau entri registri Windows

Untuk membuat kunci registri Windows dengan Fleet Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tombol di sebelah node terkelola yang ingin Anda buat kunci registri.

4. Pilih View details (Lihat detail).
5. Pilih Alat, registri Windows.
6. Pilih hive tempat Anda ingin membuat kunci registri baru dengan memilih Nama registri.
7. Pilih Buat, Buat kunci registri.
8. Pilih tombol di samping entri registri yang ingin Anda buat kunci barunya.
9. Pilih Buat kunci registri.
10. Masukkan nilai untuk Nama kunci registri baru tersebut, lalu pilih Kirim.

Untuk membuat entri registri Windows dengan Fleet Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tombol di samping instans yang ingin Anda buat entri registrinya.
4. Pilih Lihat detail.
5. Pilih Alat, registri Windows.
6. Pilih hive dan kunci registri berikutnya yang ingin Anda buat entri registri barunya dengan memilih Nama registri.
7. Pilih Buat, Buat entri registri.
8. Masukkan nilai untuk Nama entri registri baru tersebut.
9. Pilih Jenis nilai yang ingin Anda buat untuk entri registri tersebut. Untuk informasi selengkapnya tentang jenis nilai registri, lihat Jenis nilai registri.
10. Masukkan nilai untuk Nilai entri registri baru tersebut.

Memperbarui entri registri Windows

Untuk memperbarui entri registri Windows dengan Fleet Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.

2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tombol di sebelah node terkelola yang ingin Anda perbarui entri registri.
4. Pilih View details (Lihat detail).
5. Pilih Alat, registri Windows.
6. Pilih hive dan kunci registri berikutnya yang ingin Anda perbarui dengan memilih Nama registri.
7. Pilih tombol di samping entri registri yang ingin Anda perbarui.
8. Pilih Tindakan, Perbarui entri registri.
9. Masukkan nilai baru untuk Nilai entri registri baru tersebut.
10. Pilih Perbarui.

Menghapus entri atau kunci registri Windows

Untuk menghapus kunci registri Windows dengan Fleet Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tombol di sebelah node terkelola yang ingin Anda hapus kunci registri.
4. Pilih Alat, registri Windows.
5. Pilih hive dan kunci registri berikutnya yang ingin Anda hapus dengan memilih Nama registri.
6. Pilih tombol di samping kunci registri yang ingin Anda hapus.
7. Pilih Tindakan, Hapus kunci registri.

Untuk menghapus entri registri Windows dengan Fleet Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih tombol di sebelah node terkelola tempat Anda ingin menghapus entri registri.
4. Pilih View details (Lihat detail).
5. Pilih Alat, registri Windows.
6. Pilih hive dan kunci registri berikutnya yang berisi entri yang ingin Anda hapus dengan memilih Nama registri.
7. Pilih tombol di samping entri registri yang ingin Anda hapus.
8. Pilih Tindakan, Hapus entri registri.

Mengakses portal Red Hat Knowledgebase

Anda dapat menggunakanFleet Manager, kemampuanAWS Systems Manager, untuk mengakses portal Knowledgebase jika Anda adalah pelanggan Red Hat. Anda dianggap sebagai pelanggan Red Hat jika Anda menjalankan Red Hat Enterprise Linux (RHEL) instans atau menggunakan RHEL layanan. AWS Portal Knowledgebase mencakup binari, dan forum berbagi pengetahuan dan diskusi untuk dukungan komunitas yang hanya tersedia untuk pelanggan berlisensi Red Hat.

Selain izin yang diperlukan AWS Identity and Access Management (IAM) untuk Systems Manager danFleet Manager, pengguna atau peran yang Anda gunakan untuk mengakses konsol harus mengizinkan `rhelkb:GetRhelURL` tindakan untuk mengakses portal Knowledgebase.

Untuk mengakses Portal Pangkalan Pengetahuan Red Hat

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.

3. Pilih RHEL contoh yang ingin Anda gunakan untuk terhubung ke Red Hat Knowledgebase Portal.
4. Pilih Manajemen akun, Akses Red Hat Knowledgebase untuk membuka halaman Red Hat Knowledgebase.

Jika Anda menggunakan RHEL on AWS untuk menjalankan RHEL beban kerja yang didukung penuh, Anda juga dapat mengakses Red Hat Knowledgebase melalui situs web Red Hat dengan menggunakan kredensialnya. AWS

Memecahkan masalah ketersediaan node terkelola

Untuk beberapa AWS Systems Manager kemampuan sepertiRun Command,Distributor, danSession Manager, Anda dapat memilih untuk secara manual memilih node terkelola tempat Anda ingin menjalankan operasi. Dalam kasus seperti ini, setelah Anda menentukan bahwa Anda ingin memilih node secara manual, sistem menampilkan daftar node terkelola tempat Anda dapat menjalankan operasi.

Topik ini memberikan informasi untuk membantu Anda mendiagnosis mengapa node terkelola yang telah Anda konfirmasi berjalan tidak disertakan dalam daftar node terkelola di Systems Manager.

Agar node dikelola oleh Systems Manager dan tersedia dalam daftar node terkelola, node harus memenuhi tiga persyaratan:

- SSM Agentharus diinstal dan berjalan pada node dengan sistem operasi yang didukung.

> **Note**
>
> Beberapa AWS managed Amazon Machine Images (AMIs) dikonfigurasi untuk meluncurkan instance dengan SSM Agentprainstal. (Anda juga dapat mengonfigurasi kustom AMI untuk pra-instalSSM Agent.) Untuk informasi selengkapnya, lihat Amazon Machine Images(AMIs) dengan SSM Agent prainstal.

- Untuk instans Amazon Elastic Compute Cloud (Amazon EC2), Anda harus melampirkan AWS Identity and Access Management profil instans (IAM) ke instans. Profil instans memungkinkan instance untuk berkomunikasi dengan layanan Systems Manager. Jika Anda tidak menetapkan

profil instance ke instance, Anda mendaftarkannya menggunakan aktivasi hibrida, yang bukan skenario umum.

- SSM Agentharus dapat terhubung ke titik akhir Systems Manager untuk mendaftarkan dirinya dengan layanan. Setelah itu, node yang dikelola harus tersedia untuk layanan, yang dikonfirmasi oleh layanan yang mengirimkan sinyal setiap lima menit untuk memeriksa kesehatan instans.
- Setelah status node terkelola setidaknya `Connection Lost` selama 30 hari, node mungkin tidak lagi terdaftar di Fleet Manager konsol. Untuk mengembalikannya ke daftar, masalah yang menyebabkan koneksi terputus harus diselesaikan.

Setelah memverifikasi bahwa node terkelola sedang berjalan, Anda dapat menggunakan perintah berikut untuk memeriksa apakah SSM Agent berhasil terdaftar dengan layanan Systems Manager. Perintah ini tidak mengeluarkan hasil hingga pendaftaran berhasil dilakukan.

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
```

Jika pendaftaran berhasil dan node terkelola sekarang tersedia untuk operasi Systems Manager, perintah mengembalikan hasil yang mirip dengan berikut ini.

```
{
    "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"
        }
    ]
}
```

Jika pendaftaran belum selesai atau gagal, perintah tersebut mengeluarkan hasil yang serupa dengan berikut ini:

```
{
    "InstanceAssociationStatusInfos": []
}
```

Jika perintah tidak mengembalikan hasil setelah 5 menit atau lebih, gunakan informasi berikut untuk membantu Anda memecahkan masalah dengan node terkelola Anda.

Topik

- Solusi 1: Verifikasi SSM Agent yang diinstal dan berjalan pada node terkelola
- Solusi 2: Verifikasi bahwa profil instans IAM telah ditentukan untuk instance (hanya instans EC2)
- Solusi 3: Verifikasi konektivitas titik akhir layanan
- Solusi 4: Verifikasi dukungan sistem operasi target
- Solusi 5: Pastikan Anda bekerja Wilayah AWS sama dengan instans Amazon EC2
- Solusi 6: Verifikasi konfigurasi proxy yang Anda terapkan SSM Agent pada node terkelola
- Solusi 7: Instal sertifikat TLS pada instans terkelola
- Memecahkan masalah ketersediaan node terkelola menggunakan ssm-cli

Solusi 1: Verifikasi SSM Agent yang diinstal dan berjalan pada node terkelola

Pastikan versi terbaru diinstal dan berjalan pada node terkelola. SSM Agent

Untuk menentukan SSM Agent apakah diinstal dan berjalan pada node terkelola, lihatMemeriksa SSM Agent status dan memulai agen.

Untuk menginstal atau menginstal ulang SSM Agent pada node terkelola, lihat topik berikut:

- Bekerja denganSSM Agent instans EC2 untuk Linux
- Instal SSM Agent untuk lingkungan hybrid (Linux)
- Bekerja dengan SSM Agent instans EC2 untuk Windows Server
- Instal SSM Agent untuk lingkungan hybrid (Windows)

Solusi 2: Verifikasi bahwa profil instans IAM telah ditentukan untuk instance (hanya instans EC2)

Untuk instans Amazon Elastic Compute Cloud (Amazon EC2), verifikasi bahwa instans dikonfigurasi dengan profil instans (IAM) AWS Identity and Access Management yang memungkinkan instans berkomunikasi dengan Systems Manager API. Juga verifikasi bahwa pengguna Anda memiliki kebijakan kepercayaan IAM yang memungkinkan pengguna Anda berkomunikasi dengan Systems Manager API.

> **Note**
>
> Server lokal, perangkat edge, dan mesin virtual (VM) menggunakan peran layanan IAM, bukan profil instance. Untuk informasi lebih lanjut, lihat Membuat peran layanan IAM untuk lingkungan hibrid.

Untuk menentukan apakah profil instans dengan izin yang diperlukan dilampirkan ke instans EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans untuk memeriksa profil instans.
4. Di tab Deskripsi di panel bawah, temukan IAM role dan pilih nama peran.

5. Di halaman Ringkasan peran untuk profil instans, di tab Izin, pastikan bahwa `AmazonSSMManagedInstanceCore` tercantum di bawah Kebijakan izin.

   Jika kebijakan kustom digunakan sebagai gantinya, pastikan bahwa kebijakan tersebut menyediakan izin yang sama seperti `AmazonSSMManagedInstanceCore`.

   Buka `AmazonSSMManagedInstanceCore` di konsol

   Untuk informasi tentang kebijakan lain yang dapat dilampirkan ke profil instans untuk Systems Manager, lihat Mengonfigurasi izin instans untuk Systems Manager.

Solusi 3: Verifikasi konektivitas titik akhir layanan

Verifikasi bahwa instans memiliki konektivitas ke titik akhir layanan Systems Manager. Konektivitas ini disediakan dengan membuat dan mengonfigurasi titik akhir VPC untuk Systems Manager, atau dengan mengizinkan lalu lintas keluar HTTPS (port 443) ke titik akhir layanan.

Untuk instans Amazon EC2, titik akhir layanan Systems Manager untuk instans digunakan untuk mendaftarkan instance jika konfigurasi virtual private cloud (VPC) memungkinkan lalu lintas keluar. Wilayah AWS Namun, jika konfigurasi VPC tempat instance diluncurkan tidak mengizinkan lalu lintas keluar dan Anda tidak dapat mengubah konfigurasi ini untuk mengizinkan konektivitas ke titik akhir layanan publik, Anda harus mengonfigurasi titik akhir antarmuka untuk VPC Anda sebagai gantinya.

Untuk informasi selengkapnya, lihat Membuat titik akhir VPC.

Solusi 4: Verifikasi dukungan sistem operasi target

Verifikasi bahwa operasi yang Anda pilih dapat dijalankan pada jenis node terkelola yang Anda harapkan untuk dilihat terdaftar. Beberapa operasi Systems Manager hanya dapat menargetkan instans Windows atau hanya instans Linux. Misalnya, dokumen Systems Manager (SSM) `AWS-InstallPowerShellModule` dan `AWS-ConfigureCloudWatch` hanya dapat dijalankan di instans Windows saja. Di halaman Jalankan perintah, jika Anda memilih salah satu dari dokumen ini dan memilih Pilih instans secara manual, hanya instans Windows Anda yang tercantum dan tersedia untuk dipilih.

Solusi 5: Pastikan Anda bekerja Wilayah AWS sama dengan instans Amazon EC2

Instans Amazon EC2 dibuat dan tersedia secara spesifik Wilayah AWS, seperti Wilayah Timur AS (Ohio) (us-timur-2) atau Wilayah Eropa (Irlandia) (eu-barat-1). Pastikan Anda bekerja Wilayah AWS

sama dengan instans Amazon EC2 yang ingin Anda gunakan. Untuk informasi selengkapnya, lihat Memilih Wilayah dalam Memulai dengan AWS Management Console.

Solusi 6: Verifikasi konfigurasi proxy yang Anda terapkan SSM Agent pada node terkelola

Verifikasi bahwa konfigurasi proxy yang Anda terapkan SSM Agent pada node terkelola sudah benar. Jika konfigurasi proxy salah, node tidak dapat terhubung ke titik akhir layanan yang diperlukan, atau Systems Manager mungkin mengidentifikasi sistem operasi node terkelola secara tidak benar. Lihat informasi yang lebih lengkap di Mengkonfigurasi SSM Agent untuk menggunakan proxy (Linux) dan SSM AgentKonfigurasikan untuk menggunakan proxy untuk Windows Server instance.

Solusi 7: Instal sertifikat TLS pada instans terkelola

Sertifikat Transport Layer Security (TLS) harus diinstal pada setiap instance terkelola yang Anda gunakan. AWS Systems Manager Layanan AWS gunakan sertifikat ini untuk mengenkripsi panggilan ke yang lain Layanan AWS.

Sertifikat TLS sudah diinstal secara default pada setiap instans Amazon EC2 dibuat dari Amazon Machine Image (AMI). Sebagian besar sistem operasi modern termasuk sertifikat TLS yang diperlukan dari Amazon Trust Services CAs berada di penyimpanan terpercayaan mereka.

Untuk memverifikasi apakah sertifikat yang diperlukan diinstal pada instans Anda, jalankan perintah berikut berdasarkan sistem operasi instance Anda. Pastikan untuk mengganti bagian *wilayah* URL dengan Wilayah AWS tempat instans terkelola Anda berada.

Linux & macOS

```
curl -L https://ssm.region.amazonaws.com
```

Windows

```
Invoke-WebRequest -Uri https://ssm.region.amazonaws.com
```

Perintah harus mengembalikan `Unknown0perationException` kesalahan. Jika Anda menerima pesan kesalahan SSL/TLS, maka sertifikat yang diperlukan mungkin tidak diinstal.

Jika Anda menemukan sertifikat Amazon Trust Services CA yang diperlukan tidak diinstal pada sistem operasi dasar Anda, pada instans yang dibuat dari AMIs yang tidak disediakan oleh Amazon,

atau di server lokal dan VM Anda sendiri, Anda harus menginstal dan mengizinkan sertifikat dari Amazon Trust Services, atau menggunakan AWS Certificate Manager (ACM) untuk membuat dan mengelola sertifikat untuk layanan terintegrasi yang didukung.

Setiap instans terkelola Anda harus memiliki salah satu sertifikat Transport Layer Security (TLS) berikut yang diinstal.

- Amazon Root CA 1
- Starfield Services Root Certificate Authority - G2
- Starfield Class 2 Certificate Authority

Untuk informasi tentang cara menggunakan ACM, lihat Panduan Pengguna AWS Certificate Manager.

Jika sertifikat di lingkungan komputasi dikelola oleh objek kebijakan grup (GPO), maka Anda mungkin perlu mengkonfigurasi kebijakan grup untuk menyertakan salah satu sertifikat ini.

Untuk informasi selengkapnya tentang sertifikat Amazon Root dan Starfield, lihat posting blog Cara AWS Mempersiapkan Pindah ke Otoritas Sertifikat Sendiri.

Memecahkan masalah ketersediaan node terkelola menggunakan `ssm-cli`

`ssm-cli`Ini adalah alat baris perintah mandiri yang termasuk dalam SSM Agent instalasi. Ketika Anda menginstal SSM Agent 3.1.501.0 atau yang lebih baru pada mesin, Anda dapat menjalankan `ssm-cli` perintah pada mesin itu. Output dari perintah tersebut membantu Anda menentukan apakah mesin memenuhi persyaratan minimum untuk instans Amazon EC2 atau mesin non-EC2 yang akan dikelola oleh AWS Systems Manager, dan oleh karena itu ditambahkan ke daftar node terkelola di Systems Manager. (SSM Agentversi 3.1.501.0 dirilis pada November 2021.)

Persyaratan minimum

Agar instans Amazon EC2 atau mesin non-EC2 dikelola oleh AWS Systems Manager, dan tersedia dalam daftar node terkelola, instans harus memenuhi tiga persyaratan utama:

- SSM Agentharus diinstal dan dijalankan pada mesin dengan sistem operasi yang didukung.

  Beberapa AWS managed Amazon Machine Images (AMIs) untuk EC2 dikonfigurasi untuk meluncurkan instance dengan SSM Agentprainstal. (Anda juga dapat mengonfigurasi kustom AMI untuk pra-instalSSM Agent.) Untuk informasi selengkapnya, lihat Amazon Machine Images(AMIs) dengan SSM Agent prainstal.

- Profil instans AWS Identity and Access Management (IAM) (untuk instans EC2) atau peran layanan IAM (untuk mesin non-EC2) yang menyediakan izin yang diperlukan untuk berkomunikasi dengan layanan Systems Manager harus dilampirkan ke mesin.
- SSM Agentharus dapat terhubung ke titik akhir Systems Manager untuk mendaftarkan dirinya dengan layanan. Setelah itu, node yang dikelola harus tersedia untuk layanan, yang dikonfirmasi oleh layanan yang mengirimkan sinyal setiap lima menit untuk memeriksa kesehatan node yang dikelola.

Perintah yang telah dikonfigurasi sebelumnya di **`ssm-cli`**

Perintah yang telah dikonfigurasi sebelumnya disertakan yang mengumpulkan informasi yang diperlukan untuk membantu Anda mendiagnosis mengapa mesin yang telah Anda konfirmasi berjalan tidak disertakan dalam daftar node terkelola di Systems Manager. Perintah ini dijalankan ketika Anda menentukan `get-diagnostics` opsi.

Pada mesin, jalankan perintah berikut untuk digunakan `ssm-cli` untuk membantu Anda memecahkan masalah ketersediaan node terkelola.

Linux & macOS

```
ssm-cli get-diagnostics --output table
```

Windows

Pada Windows Server mesin, Anda harus menavigasi ke `C:\Program Files\Amazon\SSM` direktori sebelum menjalankan perintah.

```
ssm-cli.exe get-diagnostics --output table
```

PowerShell

Pada Windows Server mesin, Anda harus menavigasi ke `C:\Program Files\Amazon\SSM` direktori sebelum menjalankan perintah.

```
.\ssm-cli.exe get-diagnostics --output table
```

Perintah mengembalikan output sebagai tabel yang mirip dengan berikut ini.

> **Note**
>
> Pemeriksaan konektivitas kessmmessages,,s3, kmslogs, dan monitoring titik akhir adalah untuk fitur opsional tambahan seperti Session Manager yang dapat masuk ke Amazon Simple Storage Service (Amazon S3) atau CloudWatch Amazon Logs, dan AWS Key Management Service menggunakan enkripsi ().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
                                    #
##############################################################################################
```

Tabel berikut memberikan rincian tambahan untuk setiap pemeriksaan yang dilakukan oleh`ssm-cli`.

`ssm-cli`pemeriksaan diagnostik

Memeriksa	Detail
Layanan metadata instans Amazon EC2	Menunjukkan apakah node terkelola dapat mencapai layanan metadata. Tes yang gagal menunjukkan masalah konektivitas `http://169.254.169.254` yang dapat disebabkan oleh konfigurasi firewall dan proxy rute lokal, proxy, atau sistem operasi (OS).
Pendaftaran instance hybrid	Menunjukkan SSM Agent apakah terdaftar menggunakan aktivasi hibrida.
Konektivitas ke `ssm` titik akhir	Menunjukkan apakah node dapat mencapai titik akhir layanan untuk Systems Manager pada port TCP 443. Tes yang gagal menunjukkan masalah konektivitas `https://ssm.` *`region`*`.amazonaws.com` tergantung pada Wilayah AWS tempat node berada. Masalah konektivitas dapat disebabkan oleh konfigurasi VPC termasuk grup keamanan, daftar kontrol akses jaringan, tabel rute, atau firewall dan proxy OS.
Konektivitas ke `ec2messages` titik akhir	Menunjukkan apakah node dapat mencapai titik akhir layanan untuk Systems Manager pada port TCP 443. Tes yang gagal menunjukkan masalah konektivitas `https://ec2messages.` *`region`*`.amazonaws.com` tergantung pada Wilayah AWS tempat node berada. Masalah konektivitas dapat disebabkan oleh konfigurasi VPC termasuk grup keamanan, daftar kontrol akses jaringan, tabel rute, atau firewall dan proxy OS.
Konektivitas ke `ssmmessages` titik akhir	Menunjukkan apakah node dapat mencapai titik akhir layanan untuk Systems Manager pada

Memeriksa	Detail
	port TCP 443. Tes yang gagal menunjukkan masalah konektivitas `https://ssmmessages.` *region*`.amazonaws.com` tergantung pada Wilayah AWS tempat node berada. Masalah konektivitas dapat disebabkan oleh konfigurasi VPC termasuk grup keamanan, daftar kontrol akses jaringan, tabel rute, atau firewall dan proxy OS.
Konektivitas ke `s3` titik akhir	Menunjukkan apakah node dapat mencapai titik akhir layanan untuk Amazon Simple Storage Service pada port TCP 443. Tes yang gagal menunjukkan masalah konektivitas `https://s3.` *region*`.amazonaws.com` tergantung pada Wilayah AWS tempat node berada. Konektivitas ke titik akhir ini tidak diperlukan agar node muncul di daftar node terkelola Anda.
Konektivitas ke `kms` titik akhir	Menunjukkan apakah node dapat mencapai titik akhir layanan untuk AWS Key Management Service pada port TCP 443. Tes yang gagal menunjukkan masalah konektivitas `https://kms.` *region*`.amazonaws.com` tergantung pada Wilayah AWS tempat node berada. Konektivitas ke titik akhir ini tidak diperlukan agar node muncul di daftar node terkelola Anda.

Memeriksa	Detail
Konektivitas ke `logs` titik akhir	Menunjukkan apakah node dapat mencapai titik akhir layanan untuk Amazon CloudWatch Logs pada port TCP 443. Tes yang gagal menunjukkan masalah konektivitas `https://logs.` *region* `.amazonaws.com` tergantung pada Wilayah AWS tempat node berada. Konektivitas ke titik akhir ini tidak diperlukan agar node muncul di daftar node terkelola Anda.
Konektivitas ke `monitoring` titik akhir	Menunjukkan apakah node dapat mencapai titik akhir layanan untuk Amazon CloudWatch pada port TCP 443. Tes yang gagal menunjukkan masalah konektivitas `https://monitoring.` *region* `.amazonaws.com` tergantung pada Wilayah AWS tempat node berada. Konektivitas ke titik akhir ini tidak diperlukan agar node muncul di daftar node terkelola Anda.
AWS Kredensil	Menunjukkan apakah SSM Agent memiliki kredensional yang diperlukan berdasarkan profil instans IAM (untuk instans EC2) atau peran layanan IAM (untuk mesin non-EC2) yang terpasang pada mesin. Pengujian yang gagal menunjukkan bahwa tidak ada profil instans IAM atau peran layanan IAM yang dilampirkan ke mesin, atau tidak berisi izin yang diperlukan untuk Systems Manager.

Memeriksa	Detail
Layanan agen	Menunjukkan apakah SSM Agent layanan sedang berjalan, dan apakah layanan berjalan sebagai root untuk Linux ataumacOS, atau SYSTEM untukWindows Server. Tes yang gagal menunjukkan SSM Agent layanan tidak berjalan atau tidak berjalan sebagai root atau SYSTEM.
Konfigurasi proxy	Menunjukkan SSM Agent apakah dikonfigurasi untuk menggunakan proxy.
Status gambar Sysprep (hanya Windows)	Menunjukkan keadaan `Sysprep` pada node. SSM Agenttidak akan dimulai pada node jika `Sysprep` status adalah nilai selain`IMAGE_STATE_COMPLETE` .
Versi SSM Agent	Menunjukkan apakah versi terbaru yang SSM Agent tersedia diinstal.

AWS Systems Manager Kepatuhan

Anda dapat menggunakan Kepatuhan, kemampuan AWS Systems Manager, untuk memindai armada node terkelola Anda untuk kepatuhan patch dan inkonsistensi konfigurasi. Anda dapat mengumpulkan dan mengumpulkan data dari beberapa Akun AWS dan Wilayah, lalu menelusuri sumber daya tertentu yang tidak sesuai. Secara default, Kepatuhan menampilkan data kepatuhan saat ini tentang penambalan Patch Manager dan asosiasi diState Manager. (Patch Managerdan State Manager juga keduanya kemampuan AWS Systems Manager.) Untuk memulai dengan Kepatuhan, buka konsol Systems Manager. Di panel navigasi, pilih Kepatuhan.

Data kepatuhan tambalan dari Patch Manager dapat dikirim ke AWS Security Hub. Security Hub memberi Anda pandangan komprehensif tentang pemberitahuan keamanan prioritas tinggi dan status kepatuhan Anda. Hub juga memantau status patching armada Anda. Untuk informasi selengkapnya, lihat Integrasi dengan Patch ManagerAWS Security Hub.

Kepatuhan menawarkan manfaat dan fitur tambahan berikut:

- Lihat riwayat kepatuhan dan pelacakan perubahan untuk Patch Manager menambal data dan State Manager asosiasi dengan menggunakan AWS Config.
- Sesuaikan layanan untuk membuat jenis kepatuhan Anda sendiri berdasarkan persyaratan IT atau bisnis Anda.
- Memperbaiki masalah dengan menggunakanRun Command, kemampuan lain dari AWS Systems ManagerState Manager, atau Amazon EventBridge.
- Port data ke Amazon Athena dan Amazon QuickSight untuk menghasilkan laporan di seluruh armada.

EventBridge dukungan

Kemampuan Systems Manager ini didukung sebagai jenis peristiwa dalam EventBridge aturan Amazon. Untuk informasi selengkapnya, lihat Pemantauan peristiwa Systems Manager dengan Amazon EventBridge dan Referensi: Pola dan jenis EventBridge acara Amazon untuk Systems Manager.

Integrasi Chef InSpec

Systems Manager terintegrasi dengan Chef InSpec. InSpec adalah kerangka kerja runtime open-source yang memungkinkan Anda membuat profil yang dapat dibaca manusia di atau GitHub Amazon Simple Storage Service (Amazon S3). Anda kemudian dapat menggunakan Systems Manager untuk menjalankan pemindaian kepatuhan dan melihat node terkelola yang sesuai dan tidak sesuai. Untuk informasi selengkapnya, lihat Menggunakan Chef InSpec profil dengan Kepatuhan Systems Manager.

Harga

Kepatuhan ditawarkan tanpa biaya tambahan. Anda hanya membayar untuk AWS sumber daya yang Anda gunakan.

Konten

- Panduan kepatuhan (AWS CLI)

Memulai dengan Kepatuhan

Untuk memulai Kepatuhan, kemampuan dari AWS Systems Manager, selesaikan tugas berikut.

Tugas	Untuk informasi selengkapnya
Kepatuhan bekerja dengan data patch diPatch Manager dan asosiasi diState Manager. (Patch ManagerdanState Manager juga keduanya kemampuanAWS Systems Manager.) Kepatuhan juga bekerja dengan jenis kepatuhan kustom kepatuhan kustom di node terkelola yang terkelola menggunak an Systems Manager. Verifikasi bahwa Anda telah menyelesaikan persyaratan pengaturan untuk instans Amazon Elastic Compute Cloud (Amazon EC2) Anda dan mesin non-EC2 di lingkungan hibrid dan multicloud.	Menyiapkan AWS Systems Manager
Perbarui Systems ManagerSSM Agent (SSM Agent) di node terkelola Anda ke versi terbaru.	Bekerja dengan SSM Agent
Jika Anda berencana untuk memantau kepatuhan patch patch patch patch, verifikas i bahwa Anda telah mengonfigurasi patchPatc h Manager. Anda harus melakukan operasi patching dengan menggunakanPatch Manager sebelum Kepatuhan patch dengan menggunak an sebelum Kepatuhan patch.	AWS Systems Manager Patch Manager
Jika Anda berencana untuk memantau kepatuhan asosiasi asosiasiState Manager asosiasi asosiasi asosiasi. Anda harus membuat asosiasi sebelum Kepatuhan dapat menampilkan data kepatuhan asosiasi.	AWS Systems Manager State Manager

Tugas	Untuk informasi selengkapnya
(Opsional) Konfigurasikan sistem untuk melihat riwayat kepatuhan dan pelacakan perubahan.	Melihat riwayat dan pelacakan perubahan konfigurasi kepatuhan
(Opsional) Buat jenis kepatuhan kustom.	Panduan kepatuhan (AWS CLI)
(Opsional) Buat sinkronisasi data sumber daya untuk menggabungkan semua data kepatuhan di bucket Amazon Simple Storage Service (Amazon S3).	Membuat sinkronisasi data sumber daya untuk Kepatuhan

Membuat sinkronisasi data sumber daya untuk Kepatuhan

Anda dapat menggunakan fitur sinkronisasi data sumber daya diAWS Systems Manager untuk mengirim data kepatuhan dari semua node terkelola Anda ke bucket Amazon Simple Storage Service (Amazon S3). Bila Anda membuat sinkronisasi, Anda dapat menentukan node terkelola dari beberapaAkun AWSWilayah AWS,, serta lingkungan hibrid dan multicloud Anda. Sinkronisasi data sumber daya kemudian secara otomatis memperbarui data terpusat saat data kepatuhan baru dikumpulkan. Dengan semua data kepatuhan disimpan di bucket S3 target, Anda dapat menggunakan layanan seperti Amazon Athena dan AmazonQuickSight untuk mengkueri dan menganalisis data agregat. Mengonfigurasi sinkronisasi data sumber daya untuk Kepatuhan adalah operasi satu kali.

Gunakan prosedur berikut untuk membuat sinkronisasi data sumber daya untuk Kepatuhan dengan menggunakan AWS Management Console.

Untuk membuat dan mengonfigurasi bucket S3 untuk sinkronisasi data sumber daya (konsol)

1. Buka konsol Amazon S3 di https://console.aws.amazon.com/s3/.
2. Buat bucket untuk menyimpan data kepatuhan agregat Anda. Untuk informasi selengkapnya, lihat Membuat Bucket di Panduan Pengguna Amazon Simple Storage Service. Catat nama bucket dan Wilayah AWS tempat Anda membuatnya.
3. Buka bucket, pilih tab Izin, dan kemudian pilih Kebijakan Bucket.
4. Salin dan tempelkan kebijakan bucket berikut ke dalam editor kebijakan. Ganti *DOC-EXAMPLE-BUCKET* dan *Account-ID* dengan nama bucket S3 yang Anda buat dan ID Akun AWS yang

valid. Secara opsional, ganti *Bucket-Prefix* dengan nama prefiks Amazon S3 (subdirektori). Jika Anda tidak membuat prefiks, hapus *Bucket-Prefix* dari ARN dalam kebijakan.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "SSMBucketPermissionsCheck",
            "Effect": "Allow",
            "Principal": {
                "Service": "ssm.amazonaws.com"
            },
            "Action": "s3:GetBucketAcl",
            "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
        },
        {
            "Sid": " SSMBucketDelivery",
            "Effect": "Allow",
            "Principal": {
                "Service": "ssm.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": ["arn:aws:s3:::DOC-EXAMPLE-BUCKET/Bucket-Prefix/*/
accountid=Account_ID_number/*"],
            "Condition": {
                "StringEquals": {
                    "s3:x-amz-acl": "bucket-owner-full-control"
                }
            }
        }
    ]
}
```

Untuk membuat sinkronisasi data sumber daya

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.

3. Pilih Manajemen akun, Sinkronisasi Data Sumber Daya, lalu pilih Buat sinkronisasi data sumber daya.
4. Di bidang Nama sinkronisasi, masukkan nama untuk konfigurasi sinkronisasi.
5. Di bidang Nama bucket, masukkan nama bucket Amazon S3 yang Anda buat pada awal prosedur ini.
6. (Opsional) Di bidang Prefiks bucket, masukkan nama prefiks bucket S3 (subdirektori).
7. Di bidang Wilayah Bucket, pilih Wilayah ini jika bucket S3 yang Anda buat berada di saat iniWilayah AWS. Jika bucket berada di Wilayah AWS yang berbeda, pilih Wilayah lain, dan masukkan nama Wilayah.

> **Note**
>
> Jika sinkronisasi dan bucket S3 target berada di Wilayah berbeda, Anda mungkin dikenakan harga transfer data. Untuk informasi selengkapnya, lihat Harga Amazon S3.

8. Pilih Buat.

Bekerja dengan Kepatuhan

Kepatuhan, kemampuan AWS Systems Manager, mengumpulkan, dan melaporkan data tentang status penambalan dalam penambalan dan Patch Manager asosiasi di. State Manager (Patch Managerdan State Manager juga keduanya kemampuan AWS Systems Manager.) Kepatuhan juga melaporkan jenis kepatuhan khusus yang telah Anda tentukan untuk node terkelola Anda. Bagian ini mencakup detail tentang masing-masing jenis kepatuhan ini dan bagaimana cara melihat data kepatuhan Systems Manager. Bagian ini juga mencakup informasi tentang cara melihat riwayat kepatuhan dan pelacakan perubahan.

> **Note**
>
> Systems Manager terintegrasi dengan Chef InSpec. InSpec adalah kerangka kerja runtime open-source yang memungkinkan Anda membuat profil yang dapat dibaca manusia di atau GitHub Amazon Simple Storage Service (Amazon S3). Kemudian Anda dapat menggunakan

Systems Manager untuk menjalankan pemindaian kepatuhan dan melihat instans yang patuh dan tidak patuh. Untuk informasi selengkapnya, lihat Menggunakan Chef InSpec profil dengan Kepatuhan Systems Manager.

Tentang kepatuhan patch

Setelah Anda menggunakan Patch Manager untuk menginstal tambalan pada instans Anda, informasi status kepatuhan segera tersedia untuk Anda di konsol atau sebagai respons terhadap AWS Command Line Interface (AWS CLI) perintah atau operasi API Systems Manager yang sesuai.

Untuk informasi tentang nilai status kepatuhan patch, lihat Memahami nilai keadaan kepatuhan patch.

Tentang kepatuhan State Manager asosiasi

Setelah Anda membuat satu atau beberapa State Manager asosiasi, informasi status kepatuhan akan segera tersedia untuk Anda di konsol atau sebagai respons terhadap AWS CLI perintah atau operasi API Systems Manager yang terkait. Untuk asosiasi, Kepatuhan menunjukkan status `Compliant` atau `Non-compliant` dan tingkat kepelikan ditetapkan untuk asosiasi, seperti `Critical` atau `Medium`.

Tentang kepatuhan kustom

Anda dapat menetapkan metadata kepatuhan ke node terkelola. Metadata ini kemudian dapat digabungkan dengan data kepatuhan lainnya untuk tujuan pelaporan kepatuhan. Misalnya, katakan bahwa bisnis Anda menjalankan versi 2.0, 3.0, dan 4.0 perangkat lunak X pada node terkelola Anda. Perusahaan ingin melakukan standarisasi pada versi 4.0, yang berarti bahwa instans yang menjalankan versi 2.0 dan 3.0 tidak sesuai. Anda dapat menggunakan operasi PutComplianceItemsAPI untuk secara eksplisit mencatat node terkelola mana yang menjalankan versi perangkat lunak X yang lebih lama. Anda hanya dapat menetapkan metadata kepatuhan dengan menggunakan,, atau SDK AWS CLI. AWS Tools for Windows PowerShell Perintah sampel CLI berikut memberikan metadata kepatuhan ke instans terkelola dan menentukan jenis kepatuhan dalam format `Custom:` yang diperlukan. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri.

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**
>
> `ResourceType`Parameter hanya mendukung`ManagedInstance`. Jika Anda menambahkan kepatuhan khusus ke perangkat AWS IoT Greengrass inti terkelola, Anda harus menentukan `ResourceType` dari`ManagedInstance`.

Manajer kepatuhan kemudian dapat melihat ringkasan atau membuat laporan tentang node terkelola mana yang sesuai atau tidak. Anda dapat menetapkan maksimal 10 jenis kepatuhan kustom yang berbeda ke node terkelola.

Untuk contoh cara membuat jenis kepatuhan kustom dan melihat data kepatuhan, lihat Panduan kepatuhan (AWS CLI).

Melihat data kepatuhan saat ini

Bagian ini menjelaskan cara melihat data kepatuhan di konsol Systems Manager dan dengan menggunakan AWS CLI. Untuk informasi tentang cara melihat riwayat kepatuhan patch dan asosiasi serta pelacakan perubahan, lihat Melihat riwayat dan pelacakan perubahan konfigurasi kepatuhan.

Topik

- Melihat data kepatuhan saat ini (konsol)
- Melihat data kepatuhan saat ini (AWS CLI)

Melihat data kepatuhan saat ini (konsol)

Gunakan prosedur berikut ini untuk melihat data kepatuhan di konsol Systems Manager.

Untuk melihat laporan kepatuhan saat ini di konsol Systems Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Kepatuhan.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Kepatuhan di panel navigasi.
3. Di bagian pemfilteran dasbor Kepatuhan, pilih opsi untuk memfilter data kepatuhan. Bagian Ringkasan sumber daya kepatuhan menampilkan jumlah data kepatuhan berdasarkan filter yang Anda pilih.
4. Untuk menelusuri sumber daya untuk informasi selengkapnya, gulir ke bawah ke ikhtisar Detail untuk area sumber daya dan pilih ID node terkelola.
5. Pada halaman Instance ID atau Name details, pilih tab Kepatuhan konfigurasi untuk melihat laporan kepatuhan konfigurasi terperinci untuk node terkelola.

> **Note**
>
> Untuk informasi tentang memperbaiki masalah kepatuhan, lihat Memperbaiki masalah kepatuhan menggunakan EventBridge.

Melihat data kepatuhan saat ini (AWS CLI)

Anda dapat melihat ringkasan data kepatuhan untuk jenis patching, asosiasi, dan kepatuhan kustom di bagian dalam menggunakan AWS CLI perintah berikut. AWS CLI

list-compliance-summaries

Mengembalikan jumlah ringkasan status asosiasi yang sesuai dan tidak sesuai dengan filter yang Anda tentukan. (API: ListComplianceSummaries)

list-resource-compliance-summaries

Mengembalikan jumlah ringkasan tingkat sumber daya. Ringkasan tersebut mencakup informasi tentang status patuh dan tidak patuh serta jumlah kepelikan item kepatuhan yang mendetail, sesuai dengan kriteria filter yang Anda tentukan. (API: ListResourceComplianceSummaries)

Anda dapat melihat data kepatuhan tambahan untuk patching dengan menggunakan perintah AWS CLI berikut.

describe-patch-group-state

Mengembalikan status kepatuhan patch agregat tingkat tinggi untuk grup patch. (API: DescribePatchGroupState)

describe-instance-patch-states-for-patch-group

Mengembalikan status patch tingkat tinggi untuk instans dalam grup patch yang ditentukan. (API: DescribeInstancePatchStatesForPatchGroup)

> **Note**
>
> Untuk ilustrasi tentang cara mengonfigurasi patching dan melihat detail kepatuhan tambalan dengan menggunakan AWS CLI, lihat. Tutorial: Menambal lingkungan server (AWS CLI)

Melihat riwayat dan pelacakan perubahan konfigurasi kepatuhan

Kepatuhan Systems Manager menampilkan data patching dan kepatuhan asosiasi saat ini untuk node terkelola Anda. Anda dapat melihat riwayat patching dan kepatuhan asosiasi serta mengubah pelacakan dengan menggunakan AWS Config. AWS Config memberikan tampilan rinci tentang konfigurasi AWS sumber daya di Anda Akun AWS. Ini mencakup bagaimana sumber daya terkait satu sama lain dan bagaimana sumber daya tersebut dikonfigurasi di masa lalu sehingga Anda dapat melihat bagaimana konfigurasi dan hubungan berubah dari waktu ke waktu. Untuk melihat riwayat kepatuhan dan pelacakan perubahan patching dan asosiasi, Anda harus mengaktifkan sumber daya berikut di AWS Config:

- `SSM:PatchCompliance`
- `SSM:AssociationCompliance`

Untuk informasi tentang cara memilih dan mengonfigurasi sumber daya ini di AWS Config, lihat Memilih Catatan AWS Config Sumber Daya Mana di Panduan Developer AWS Config .

> **Note**
>
> Untuk informasi tentang AWS Config harga, lihat Harga.

Menghapus sinkronisasi data sumber daya untuk Kepatuhan

Jika Anda tidak lagi ingin menggunakanAWS Systems Manager Kepatuhan untuk melihat data kepatuhan, sebaiknya hapus sinkronisasi data sumber daya yang digunakan untuk pengumpulan data Kepatuhan.

Untuk menghapus sinkronisasi data sumber daya Kepatuhan

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih Manajemen akun, Sinkronisasi data sumber daya.
4. Pilih sinkronisasi di daftar tersebut.

   > **Important**
   >
   > Pastikan Anda memilih sinkronisasi yang digunakan untuk Kepatuhan. Systems Manager mendukung sinkronisasi data sumber daya untuk beberapa kemampuan. Jika Anda memilih sinkronisasi yang salah, Anda dapat mengganggu agregasi data untuk Systems Manager Explorer atau Systems Manager Inventory.
5. Pilih Delete (Hapus).
6. Hapus bucket Amazon Simple Storage Service (Amazon S3) tempat data disimpan. Untuk informasi tentang menghapus sebuah bucket S3, lihat Menghapus sebuah bucket.

Memperbaiki masalah kepatuhan menggunakan EventBridge

Anda dapat dengan cepat memulihkan masalah kepatuhan patch dan asosiasi dengan menggunakanRun Command, kemampuan. AWS Systems Manager Anda dapat menargetkan instans atau ID perangkat AWS IoT Greengrass inti atau tag dan menjalankan `AWS-RunPatchBaseline` dokumen atau `AWS-RefreshAssociation` dokumen. Jika menyegarkan asosiasi atau menjalankan kembali baseline patch gagal menyelesaikan masalah kepatuhan, maka Anda perlu menyelidiki asosiasi, patch garis dasar, atau konfigurasi instans untuk memahami mengapa operasi tidak menyelesaikan masalah. Run Command

Untuk informasi selengkapnya tentang patching, lihat AWS Systems Manager Patch Manager dan Tentang dokumen SSM `AWS-RunPatchBaseline`.

Untuk informasi selengkapnya tentang asosiasi, lihat Bekerja dengan asosiasi di Systems Manager.

Untuk informasi selengkapnya tentang menjalankan perintah, lihat AWS Systems Manager Run Command.

Tentukan Kepatuhan sebagai target suatu EventBridge peristiwa

Anda juga dapat mengonfigurasi Amazon EventBridge untuk melakukan tindakan sebagai respons terhadap peristiwa Kepatuhan Manajer Sistem. Misalnya, jika satu atau beberapa node terkelola gagal menginstal pembaruan patch Kritis atau menjalankan asosiasi yang menginstal perangkat lunak anti-virus, maka Anda dapat mengkonfigurasi EventBridge untuk menjalankan `AWS-RunPatchBaseline` dokumen atau `AWS-RefreshAssocation` dokumen ketika peristiwa Kepatuhan terjadi.

Gunakan prosedur berikut untuk mengonfigurasi Kepatuhan sebagai target suatu EventBridge peristiwa.

Mengonfigurasi Kepatuhan sebagai target EventBridge peristiwa (konsol)

1. Buka konsol Amazon EventBridge di https://console.aws.amazon.com/events/.
2. Di panel navigasi, pilih Aturan.
3. Pilih Buat aturan.
4. Masukkan nama dan deskripsi untuk aturan.

   Aturan tidak boleh memiliki nama yang sama dengan aturan lain di Wilayah AWS yang sama dan di bus peristiwa yang sama.

5. Untuk bus Acara, pilih bus acara yang ingin Anda kaitkan dengan aturan ini. Jika Anda ingin aturan ini merespons peristiwa yang cocok yang berasal dari Anda sendiriAkun AWS, pilih default. Ketika Layanan AWS di akun Anda memancarkan suatu peristiwa, itu selalu masuk ke bus acara default akun Anda.
6. Untuk jenis Aturan, pilih Aturan dengan pola peristiwa.
7. Pilih Selanjutnya.
8. Untuk Sumber acara, pilih AWSacara atau acara EventBridge mitra.
9. Di bagian Pola acara, pilih Bentuk pola acara.
10. Untuk sumber acara, pilih AWSlayanan.
11. Untuk AWSlayanan, pilih Manajer Sistem.
12. Untuk Jenis peristiwa, pilih Kepatuhan Konfigurasi.
13. Untuk Jenis detail khusus, pilih Perubahan Status Kepatuhan Konfigurasi.
14. Pilih Selanjutnya.
15. Untuk jenis Target, pilih AWSlayanan.
16. Untuk Pilih target, pilih Manajer Sistem Run Command.
17. Di daftar Dokumen, pilih dokumen Systems Manager (dokumen SSM) untuk dijalankan ketika target Anda dipanggil. Misalnya, pilih `AWS-RunPatchBaseline` untuk peristiwa patch yang tidak sesuai, atau pilih `AWS-RefreshAssociation` untuk peristiwa asosiasi yang tidak sesuai.
18. Tentukan informasi untuk bidang dan parameter yang tersisa.

    > **Note**
    >
    > Bidang dan parameter yang wajib diisi memiliki tanda bintang (*) di samping namanya. Untuk membuat target, Anda harus menentukan nilai untuk setiap parameter atau bidang yang diperlukan. Jika tidak, sistem akan membuat aturannya, tetapi aturan tersebut tidak akan dijalankan.

19. Pilih Selanjutnya.
20. (Opsional) Masukkan satu atau lebih tanda untuk aturan. Untuk informasi selengkapnya, lihat Menandai EventBridge Sumber Daya Amazon Anda di Panduan EventBridge Pengguna Amazon.
21. Pilih Selanjutnya.
22. Tinjau detail aturan dan pilih Buat aturan.

Panduan kepatuhan (AWS CLI)

Prosedur berikut memandu Anda menjalani proses penggunaanAWS Command Line Interface (AWS CLI) untuk memanggil operasiAWS Systems Manager PutComplianceItemsAPI untuk menetapkan metadata kepatuhan kustom ke sumber daya. Anda juga dapat menggunakan operasi API ini untuk secara manual menetapkan metadata kepatuhan patch atau asosiasi ke node terkelola, seperti yang ditunjukkan dalam panduan berikut. Untuk informasi selengkapnya tentang kepatuhan kustom, lihat Tentang kepatuhan kustom.

Untuk menetapkan metadata kepatuhan kustom ke instans terkelola (AWS CLI)

1. Instal dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaru dariAWS CLI.

2. Jalankan perintah berikut untuk menetapkan metadata kepatuhan kustom ke node terkelola. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri. `ResourceType`Parameter hanya mendukung nilai`ManagedInstance`. Tentukan nilai ini meskipun Anda menetapkan metadata kepatuhan kustom ke perangkatAWS IoT Greengrass inti terkelola.

   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. Ulangi langkah sebelumnya untuk menetapkan metadata kepatuhan kustom tambahan ke satu atau beberapa node. Anda juga dapat secara manual menetapkan metadata kepatuhan patch atau asosiasi ke node terkelola dengan menggunakan perintah berikut:

   Metadata kepatuhan asosiasi

   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
   ```

   Metadata kepatuhan patch

   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. Jalankan perintah berikut untuk menampilkan daftar item kepatuhan untuk node terkelola tertentu. Gunakan filter untuk menelusuri data kepatuhan tertentu.

   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
   ```

   Contoh berikut menunjukkan kepada Anda cara menggunakan perintah ini dengan filter.

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. Jalankan perintah berikut untuk melihat ringkasan status kepatuhan. Gunakan filter untuk menelusuri data kepatuhan tertentu.

```
aws ssm list-resource-compliance-summaries --filters One or more filters.
```

Contoh berikut menunjukkan kepada Anda cara menggunakan perintah ini dengan filter.

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. Jalankan perintah berikut untuk melihat jumlah ringkasan sumber daya yang sesuai dan tidak sesuai untuk jenis kepatuhan. Gunakan filter untuk menelusuri data kepatuhan tertentu.

```
aws ssm list-compliance-summaries --filters One or more filters.
```

Contoh berikut menunjukkan kepada Anda cara menggunakan perintah ini dengan filter.

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 ManagerInventaris

AWS Systems ManagerInventaris memberikan visibilitas ke lingkunganAWS komputasi Anda. Anda dapat menggunakan Inventaris untuk mengumpulkan metadata dari node terkelola Anda. Anda dapat menyimpan metadata ini di bucket Amazon Simple Storage Service (Amazon S3) sentral, dan kemudian menggunakan alat bawaan untuk mengkueri data dan dengan cepat menentukan node mana yang menjalankan perangkat lunak dan konfigurasi yang diperlukan oleh kebijakan perangkat lunak Anda, dan node mana yang perlu diperbarui. Anda dapat mengkonfigurasi Inventaris pada

semua node terkelola dengan menggunakan prosedur satu klik. Anda juga dapat mengonfigurasi dan melihat data inventaris dari beberapa Wilayah AWS dan Akun AWS. Untuk memulai dengan Inventory, buka konsol Systems Manager. Di panel navigasi, pilih Inventaris.

Jika jenis metadata, yang dikonfigurasi sebelumnya, yang dikumpulkan oleh Inventaris Systems Manager tidak memenuhi kebutuhan Anda, maka Anda dapat membuat inventaris kustom. Inventaris kustom hanyalah sebuah file JSON dengan informasi yang Anda berikan dan tambahkan ke node terkelola di direktori tertentu. Ketika Inventaris Systems Manager mengumpulkan data, ia menangkap data inventaris kustom ini. Misalnya, jika Anda menjalankan pusat data yang besar, Anda dapat menentukan lokasi rak dari setiap server Anda sebagai inventaris kustom. Anda kemudian dapat melihat data ruang rak saat Anda melihat data inventaris lainnya.

> **Important**
>
> Inventaris Systems Manager hanya mengumpulkan metadata dari node terkelola Anda. Inventaris tidak mengakses informasi atau data kepemilikan.

Tabel berikut menjelaskan jenis data yang dapat Anda kumpulkan dengan Inventaris Systems Manager. Tabel juga menjelaskan penawaran yang berbeda untuk node penargetan dan interval pengumpulan yang dapat Anda tentukan.

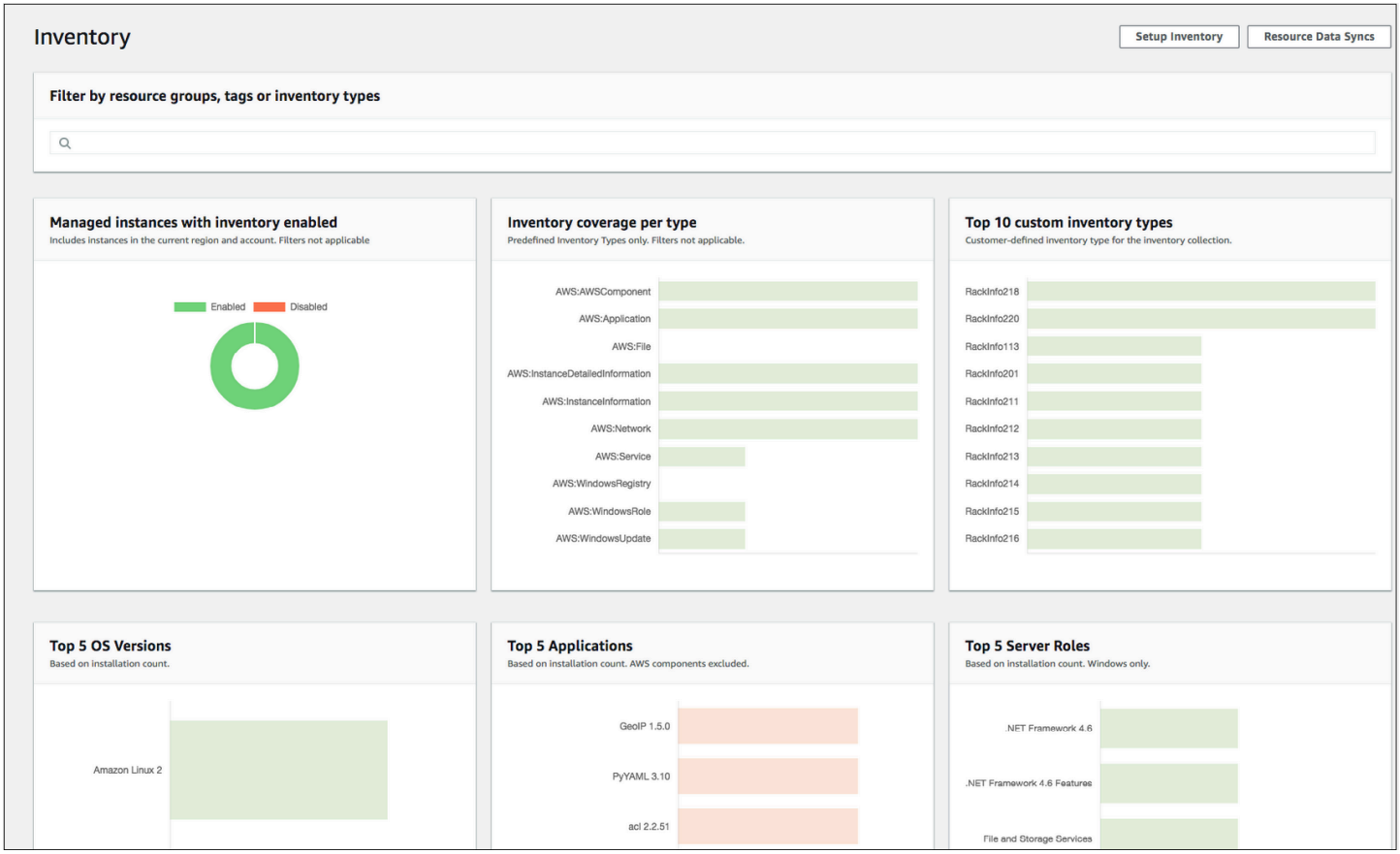
Konfigurasi	Detail
Jenis metadata	Anda dapat mengkonfigurasi Inventaris untuk mengumpulkan jenis data berikut: • Aplikasi: Nama, penerbit, versi aplikasi, dll. • Komponen AWS: driver, agen, versi EC2, dll. • File: Nama, ukuran, versi, tanggal terinstal, perubahan dan kapan terakhir kali diakses, dll • Konfigurasi jaringan: Alamat IP, alamat MAC, DNS, gateway, subnet mask, dll. • Pembaruan Windows: ID Hotfix, diinstal oleh, tanggal terinstal, dll.

Konfigurasi	Detail
	• Detail instans: Nama sistem, nama sistem operasi (OS), versi OS, DNS, domain, grup kerja, arsitektur OS, dll. • Layanan: Nama, nama tampilan, status, layanan dependen, jenis layanan, jenis mulai, dll • Tanda: Tanda yang ditetapkan ke node Anda. • Registri Windows: Jalur kunci registri, nama nilai, jenis nilai, dan nilai. • Peran Windows: Nama, nama tampilan, jalur, jenis fitur, tahapan terinstal, dll. • Inventaris kustom: Metadata yang ditetapkan ke node terkelola seperti yang dijelaskan dalamMenggunakan inventaris kustom. **Note** Untuk melihat daftar semua metadata yang dikumpulkan oleh Inventaris, lihat Metadata dikumpulkan oleh inventaris.
Node untuk ditargetkan	Anda dapat memilih untuk menginventarisasi semua node terkelola diAkun AWS, secara individual memilih node, atau grup target node dengan menggunakan tag. Untuk informasi lebih lanjut tentang pengumpulan data inventaris dari semua node terkelola Anda, lihatInventarisasi semua node terkelola di Akun AWS.

Konfigurasi	Detail
Waktu yang tepat untuk mengumpulkan informasi	Anda dapat menentukan interval pengumpulan dalam hitungan menit, jam, dan hari. Interval pengumpulan terpendek adalah setiap 30 menit.

> **Note**
>
> Tergantung dari jumlah data yang dikumpulkan, sistem dapat membutuhkan waktu beberapa menit untuk melaporkan data ke output yang Anda tentukan. Setelah informasi dikumpulkan, data dikirim melalui saluran HTTPS yang aman untukAWS penyimpanan teks murni yang hanya dapat diakses dari AndaAkun AWS.

Anda dapat melihat data di konsol Systems Manager pada halaman Inventaris, yang menyertakan beberapa kartu yang ditentukan sebelumnya untuk membantu Anda mengkueri data.



> **Note**
>
> Kartu Inventaris secara otomatis memfilter instans terkelola Amazon EC2 dengan tahapan Diakhiri dan Dihentikan. Untuk node inventaris dan perangkatAWS IoT Greengrass inti, kartu inventaris secara otomatis memfilter node dengan tahapan Diakhiri.

Jika Anda membuat sinkronisasi data sumber daya untuk menyinkronkan dan menyimpan semua data Anda dalam satu bucket Amazon S3, maka Anda dapat menelusuri data pada halaman Tampilan Detail Inventaris. Untuk informasi selengkapnya, lihat Mengkueri data inventaris dari beberapa Wilayah dan akun.

EventBridge dukungan

Kemampuan Systems Manager ini didukung sebagai jenis peristiwa di EventBridge aturan Amazon. Untuk informasi selengkapnya, lihat Pemantauan peristiwa Systems Manager dengan Amazon EventBridge dan Referensi: Pola dan jenis EventBridge acara Amazon untuk Systems Manager.

Konten

- Pelajari selengkapnya tentang Inventaris Systems Manager
- Menyiapkan Inventaris Systems Manager
- Pengonfigurasian pengumpulan inventaris
- Menggunakan data inventaris Systems Manager
- Menggunakan inventaris kustom
- Melihat riwayat inventaris dan pelacakan perubahan
- Menghentikan pengumpulan data dan menghapus data inventaris
- Panduan Inventaris Systems Manager
- Memecahkan masalah dengan Inventaris Systems Manager

Pelajari selengkapnya tentang Inventaris Systems Manager

Saat Anda mengkonfigurasiAWS Systems ManagerInventaris, Anda menentukan jenis metadata yang dikumpulkan, node yang dikelola sebagai tempat metadata sebaiknya dikumpulkan, dan jadwal untuk pengumpulan metadata. Konfigurasi ini disimpan denganAkun AWSsebagaiAWS Systems Manager State ManagerAsosiasi. Asosiasi hanya merupakan sebuah konfigurasi.

> **Note**
>
> Inventaris hanya mengumpulkan metadata. Ia tidak mengumpulkan data pribadi atau kepemilikan apa pun.

Topik

Metadata dikumpulkan oleh inventaris

Sampel berikut menunjukkan daftar lengkap metadata yang dikumpulkan oleh setiap plugin Inventaris AWS Systems Manager.

```
{
    "typeName": "AWS:InstanceInformation",
    "version": "1.0",
    "attributes":[
      { "name": "AgentType",                                "dataType" : "STRING"},
      { "name": "AgentVersion",                            "dataType" : "STRING"},
      { "name": "ComputerName",                            "dataType" : "STRING"},
      { "name": "InstanceId",                              "dataType" : "STRING"},
      { "name": "IpAddress",                               "dataType" : "STRING"},
      { "name": "PlatformName",                            "dataType" : "STRING"},
      { "name": "PlatformType",                            "dataType" : "STRING"},
      { "name": "PlatformVersion",                         "dataType" : "STRING"},
      { "name": "ResourceType",                            "dataType" : "STRING"},
      { "name": "AgentStatus",                             "dataType" : "STRING"},
      { "name": "InstanceStatus",                          "dataType" : "STRING"}
    ]
  },
  {
    "typeName" : "AWS:Application",
    "version": "1.1",
    "attributes":[
      { "name": "Name",               "dataType": "STRING"},
      { "name": "ApplicationType",    "dataType": "STRING"},
      { "name": "Publisher",          "dataType": "STRING"},
```

```
      { "name": "Version",           "dataType": "STRING"},
      { "name": "Release",           "dataType": "STRING"},
      { "name": "Epoch",             "dataType": "STRING"},
      { "name": "InstalledTime",     "dataType": "STRING"},
      { "name": "Architecture",      "dataType": "STRING"},
      { "name": "URL",               "dataType": "STRING"},
      { "name": "Summary",           "dataType": "STRING"},
      { "name": "PackageId",         "dataType": "STRING"}
    ]
  },
  {
    "typeName" : "AWS:File",
    "version": "1.0",
    "attributes":[
      { "name": "Name",              "dataType": "STRING"},
      { "name": "Size",           "dataType": "STRING"},
      { "name": "Description",       "dataType": "STRING"},
      { "name": "FileVersion",       "dataType": "STRING"},
      { "name": "InstalledDate",     "dataType": "STRING"},
      { "name": "ModificationTime",  "dataType": "STRING"},
      { "name": "LastAccessTime",    "dataType": "STRING"},
      { "name": "ProductName",       "dataType": "STRING"},
      { "name": "InstalledDir",      "dataType": "STRING"},
      { "name": "ProductLanguage",   "dataType": "STRING"},
      { "name": "CompanyName",       "dataType": "STRING"},
      { "name": "ProductVersion",      "dataType": "STRING"}
    ]
  },
  {
    "typeName" : "AWS:Process",
    "version": "1.0",
    "attributes":[
      { "name": "StartTime",         "dataType": "STRING"},
      { "name": "CommandLine",       "dataType": "STRING"},
      { "name": "User",              "dataType": "STRING"},
      { "name": "FileName",          "dataType": "STRING"},
      { "name": "FileVersion",       "dataType": "STRING"},
      { "name": "FileDescription",   "dataType": "STRING"},
      { "name": "FileSize",          "dataType": "STRING"},
      { "name": "CompanyName",       "dataType": "STRING"},
      { "name": "ProductName",       "dataType": "STRING"},
      { "name": "ProductVersion",    "dataType": "STRING"},
      { "name": "InstalledDate",     "dataType": "STRING"},
      { "name": "InstalledDir",      "dataType": "STRING"},
```

```
      { "name": "UsageId",            "dataType": "STRING"}
    ]
  },
  {
    "typeName": "AWS:AWSComponent",
    "version": "1.0",
    "attributes":[
      { "name": "Name",               "dataType": "STRING"},
      { "name": "ApplicationType",    "dataType": "STRING"},
      { "name": "Publisher",          "dataType": "STRING"},
      { "name": "Version",            "dataType": "STRING"},
      { "name": "InstalledTime",      "dataType": "STRING"},
      { "name": "Architecture",       "dataType": "STRING"},
      { "name": "URL",                "dataType": "STRING"}
    ]
  },
  {
    "typeName": "AWS:WindowsUpdate",
    "version":"1.0",
    "attributes":[
      { "name": "HotFixId",           "dataType": "STRING"},
      { "name": "Description",        "dataType": "STRING"},
      { "name": "InstalledTime",      "dataType": "STRING"},
      { "name": "InstalledBy",        "dataType": "STRING"}
    ]
  },
  {
    "typeName": "AWS:Network",
    "version":"1.0",
    "attributes":[
      { "name": "Name",               "dataType": "STRING"},
      { "name": "SubnetMask",         "dataType": "STRING"},
      { "name": "Gateway",            "dataType": "STRING"},
      { "name": "DHCPServer",         "dataType": "STRING"},
      { "name": "DNSServer",          "dataType": "STRING"},
      { "name": "MacAddress",         "dataType": "STRING"},
      { "name": "IPV4",               "dataType": "STRING"},
      { "name": "IPV6",               "dataType": "STRING"}
    ]
  },
  {
    "typeName": "AWS:PatchSummary",
    "version":"1.0",
    "attributes":[
```

```
      { "name": "PatchGroup",                        "dataType": "STRING"},
      { "name": "BaselineId",                       "dataType": "STRING"},
      { "name": "SnapshotId",                       "dataType": "STRING"},
      { "name": "OwnerInformation",                 "dataType": "STRING"},
      { "name": "InstalledCount",                   "dataType": "NUMBER"},
      { "name": "InstalledPendingRebootCount",      "dataType": "NUMBER"},
      { "name": "InstalledOtherCount",              "dataType": "NUMBER"},
      { "name": "InstalledRejectedCount",           "dataType": "NUMBER"},
      { "name": "NotApplicableCount",               "dataType": "NUMBER"},
      { "name": "UnreportedNotApplicableCount",     "dataType": "NUMBER"},
      { "name": "MissingCount",                     "dataType": "NUMBER"},
      { "name": "FailedCount",                      "dataType": "NUMBER"},
      { "name": "OperationType",                    "dataType": "STRING"},
      { "name": "OperationStartTime",               "dataType": "STRING"},
      { "name": "OperationEndTime",                 "dataType": "STRING"},
      { "name": "InstallOverrideList",              "dataType": "STRING"},
      { "name": "RebootOption",                     "dataType": "STRING"},
      { "name": "LastNoRebootInstallOperationTime", "dataType": "STRING"},
      { "name": "ExecutionId",                      "dataType": "STRING",
   "isOptional": "true"},
      { "name": "NonCompliantSeverity",             "dataType": "STRING",
   "isOptional": "true"},
      { "name": "SecurityNonCompliantCount",        "dataType": "NUMBER",
   "isOptional": "true"},
      { "name": "CriticalNonCompliantCount",        "dataType": "NUMBER",
   "isOptional": "true"},
      { "name": "OtherNonCompliantCount",           "dataType": "NUMBER",
   "isOptional": "true"}
    ]
  },
  {
    "typeName": "AWS:PatchCompliance",
    "version":"1.0",
    "attributes":[
      { "name": "Title",                      "dataType": "STRING"},
      { "name": "KBId",                       "dataType": "STRING"},
      { "name": "Classification",             "dataType": "STRING"},
      { "name": "Severity",                   "dataType": "STRING"},
      { "name": "State",                      "dataType": "STRING"},
      { "name": "InstalledTime",              "dataType": "STRING"}
    ]
  },
  {
    "typeName": "AWS:ComplianceItem",
```

```
    "version":"1.0",
    "attributes":[
      { "name": "ComplianceType",              "dataType": "STRING",
"isContext": "true"},
      { "name": "ExecutionId",                 "dataType": "STRING",
"isContext": "true"},
      { "name": "ExecutionType",               "dataType": "STRING",
"isContext": "true"},
      { "name": "ExecutionTime",               "dataType": "STRING",
"isContext": "true"},
      { "name": "Id",                          "dataType": "STRING"},
      { "name": "Title",                       "dataType": "STRING"},
      { "name": "Status",                      "dataType": "STRING"},
      { "name": "Severity",                    "dataType": "STRING"},
      { "name": "DocumentName",                "dataType": "STRING"},
      { "name": "DocumentVersion",             "dataType": "STRING"},
      { "name": "Classification",              "dataType": "STRING"},
      { "name": "PatchBaselineId",             "dataType": "STRING"},
      { "name": "PatchSeverity",               "dataType": "STRING"},
      { "name": "PatchState",                  "dataType": "STRING"},
      { "name": "PatchGroup",                  "dataType": "STRING"},
      { "name": "InstalledTime",               "dataType": "STRING"},
      { "name": "InstallOverrideList",         "dataType": "STRING",
"isOptional": "true"},
      { "name": "DetailedText",                "dataType": "STRING",
"isOptional": "true"},
      { "name": "DetailedLink",                "dataType": "STRING",
"isOptional": "true"},
      { "name": "CVEIds",                      "dataType": "STRING",
"isOptional": "true"}
    ]
 },
 {
    "typeName": "AWS:ComplianceSummary",
    "version":"1.0",
    "attributes":[
      { "name": "ComplianceType",                "dataType": "STRING"},
      { "name": "PatchGroup",                    "dataType": "STRING"},
      { "name": "PatchBaselineId",               "dataType": "STRING"},
      { "name": "Status",                        "dataType": "STRING"},
      { "name": "OverallSeverity",               "dataType": "STRING"},
      { "name": "ExecutionId",                   "dataType": "STRING"},
      { "name": "ExecutionType",                 "dataType": "STRING"},
      { "name": "ExecutionTime",                 "dataType": "STRING"},
```

```
      { "name": "CompliantCriticalCount",         "dataType": "NUMBER"},
      { "name": "CompliantHighCount",             "dataType": "NUMBER"},
      { "name": "CompliantMediumCount",           "dataType": "NUMBER"},
      { "name": "CompliantLowCount",              "dataType": "NUMBER"},
      { "name": "CompliantInformationalCount",    "dataType": "NUMBER"},
      { "name": "CompliantUnspecifiedCount",      "dataType": "NUMBER"},
      { "name": "NonCompliantCriticalCount",      "dataType": "NUMBER"},
      { "name": "NonCompliantHighCount",          "dataType": "NUMBER"},
      { "name": "NonCompliantMediumCount",        "dataType": "NUMBER"},
      { "name": "NonCompliantLowCount",           "dataType": "NUMBER"},
      { "name": "NonCompliantInformationalCount", "dataType": "NUMBER"},
      { "name": "NonCompliantUnspecifiedCount",   "dataType": "NUMBER"}
    ]
  },
  {
    "typeName": "AWS:InstanceDetailedInformation",
    "version":"1.0",
    "attributes":[
      { "name": "CPUModel",                   "dataType": "STRING"},
      { "name": "CPUCores",                   "dataType": "NUMBER"},
      { "name": "CPUs",                       "dataType": "NUMBER"},
      { "name": "CPUSpeedMHz",                "dataType": "NUMBER"},
      { "name": "CPUSockets",                 "dataType": "NUMBER"},
      { "name": "CPUHyperThreadEnabled",      "dataType": "STRING"},
      { "name": "OSServicePack",              "dataType": "STRING"}
    ]
   },
   {
     "typeName": "AWS:Service",
     "version":"1.0",
     "attributes":[
       { "name": "Name",                       "dataType": "STRING"},
       { "name": "DisplayName",                "dataType": "STRING"},
       { "name": "ServiceType",                "dataType": "STRING"},
       { "name": "Status",                     "dataType": "STRING"},
       { "name": "DependentServices",          "dataType": "STRING"},
       { "name": "ServicesDependedOn",         "dataType": "STRING"},
       { "name": "StartType",                  "dataType": "STRING"}
     ]
    },
    {
      "typeName": "AWS:WindowsRegistry",
      "version":"1.0",
      "attributes":[
```

```
        { "name": "KeyPath",                          "dataType": "STRING"},
        { "name": "ValueName",                        "dataType": "STRING"},
        { "name": "ValueType",                        "dataType": "STRING"},
        { "name": "Value",                            "dataType": "STRING"}
      ]
    },
    {
      "typeName": "AWS:WindowsRole",
      "version":"1.0",
      "attributes":[
        { "name": "Name",                         "dataType": "STRING"},
        { "name": "DisplayName",                  "dataType": "STRING"},
        { "name": "Path",                         "dataType": "STRING"},
        { "name": "FeatureType",                  "dataType": "STRING"},
        { "name": "DependsOn",                    "dataType": "STRING"},
        { "name": "Description",                  "dataType": "STRING"},
        { "name": "Installed",                    "dataType": "STRING"},
        { "name": "InstalledState",               "dataType": "STRING"},
        { "name": "SubFeatures",                  "dataType": "STRING"},
        { "name": "ServerComponentDescriptor",    "dataType": "STRING"},
        { "name": "Parent",                       "dataType": "STRING"}
      ]
    },
    {
      "typeName": "AWS:Tag",
      "version":"1.0",
      "attributes":[
        { "name": "Key",                     "dataType": "STRING"},
        { "name": "Value",                   "dataType": "STRING"}
      ]
    },
    {
      "typeName": "AWS:ResourceGroup",
      "version":"1.0",
      "attributes":[
        { "name": "Name",                  "dataType": "STRING"},
        { "name": "Arn",                   "dataType": "STRING"}
      ]
    },
    {
      "typeName": "AWS:BillingInfo",
      "version": "1.0",
      "attributes": [
        { "name": "BillingProductId",       "dataType": "STRING"}
```

```
    ]
}
```

> **Note**
>
> - Untuk`"typeName": "AWS:InstanceInformation"`, `InstanceStatus` bisa menjadi salah satu dari berikut ini: Active,, ConnectionLost Stopped, Terminated.
> - Dengan dirilisnya versi 2.5, RPM Package Manager mengganti atribut Serial dengan Jangka Waktu. Atribut Jangka Waktu adalah integer yang secara monoton meningkat seperti Serial. Ketika Anda menginventarisasi dengan menggunakan jenis `AWS:Application`, nilai Jangka Waktu yang semakin besar berarti versi yang semakin baru. Jika nilai Jangka Waktu sama atau kosong, maka gunakan nilai atribut Versi atau Rilisan untuk menentukan versi yang lebih baru.
> - Beberapa metadata tidak tersedia dari instance Linux. Secara khusus, untuk “TypeName”: “AWS:network”, jenis metadata berikut belum didukung untuk instance Linux. Mereka didukung untuk Windows.
>   - {“name”: "SubnetMask“, “DataType”: “STRING"},
>   - {“name”: “DHCPServer”, “DataType”: “STRING"},
>   - {“name”: “DNSServer”, “DataType”: “STRING"},
>   - {“name”: “Gateway”, “DataType”: “STRING"},

Menggunakan file dan inventaris registri Windows

Inventaris AWS Systems Manager memungkinkan Anda untuk mencari dan menginventarisasi file pada sistem operasi Windows, Linux, dan macOS. Anda juga dapat mencari dan menginventarisasi Registri Windows.

Berkas: Anda dapat mengumpulkan informasi metadata tentang file, termasuk di antaranya nama file, kapan file terakhir diubah dan diakses, serta ukuran file. Untuk memulai pengumpulan inventaris file, Anda menentukan jalur file tempat Anda ingin melakukan inventarisasi, satu atau beberapa pola yang menentukan jenis file yang ingin Anda inventarisasi, dan apakah jalur harus dilintasi secara berulang. Systems Manager menginventarisasi semua metadata file untuk file di jalur tertentu yang cocok dengan pola. Inventaris file menggunakan input parameter berikut.

```
{
```

```
"Path": string,
"Pattern": array[string],
"Recursive": true,
"DirScanLimit" : number // Optional
}
```

- Jalur: Jalur direktori tempat Anda ingin menginventarisasi file. Untuk Windows, Anda dapat menggunakan variabel lingkungan seperti % PROGRAMFILES% selama variabel memetakan ke jalur direktori tunggal. Misalnya, jika Anda menggunakan %PATH% yang memetakan ke beberapa jalur direktori, Inventaris menampilkan kesalahan.
- Pola: Sekumpulan pola untuk mengidentifikasi file.
- Berulang: Nilai Boolean yang mengindikasikan apakah Inventaris harus berulang kali melintasi direktori.
- DirScanLimit: Nilai opsional yang menentukan berapa banyak direktori yang dipindai. Gunakan parameter ini untuk meminimalkan dampak performa pada node yang dikelola. Secara default, Inventaris memindai maksimum 5.000 direktori.

> **Note**
>
> Inventaris mengumpulkan metadata sejumlah maksimum 500 file di semua jalur yang ditentukan.

Berikut adalah beberapa contoh cara menentukan parameter saat melakukan inventarisasi file.

- Pada Linux dan macOS, kumpulkan metadata dari file.sh di direktori `/home/ec2-user`, tanpa menyertakan semua subdirektori.

```
[{"Path":"/home/ec2-user","Pattern":["*.sh", "*.sh"],"Recursive":false}]
```

- Pada Windows, kumpulkan metadata dari semua file “.exe” di folder Program Files, termasuk subdirektori secara berulang.

```
[{"Path":"C:\Program Files","Pattern":["*.exe"],"Recursive":true}]
```

- Pada Windows, kumpulkan metadata dari pola log tertentu.

```
[{"Path":"C:\ProgramData\Amazon","Pattern":["*amazon*.log"],"Recursive":true}]
```

- Batasi jumlah direktori saat melakukan pengumpulan berulang.

```
[{"Path":"C:\Users","Pattern":["*.ps1"],"Recursive":true, "DirScanLimit": 1000}]
```

Registri Windows: Anda dapat mengumpulkan kunci dan nilai Registri Windows. Anda dapat memilih jalur kunci dan mengumpulkan semua kunci dan nilai secara berulang. Anda juga dapat mengumpulkan kunci registri tertentu dan nilainya untuk jalur tertentu. Inventaris mengumpulkan jalur, nama, jenis, dan nilai kunci.

```
{
"Path": string,
"Recursive": true,
"ValueNames": array[string] // optional
}
```

- Jalur: Jalur ke kunci Registri.
- Berulang: Nilai Boolean yang mengindikasikan apakah Inventaris harus berulang kali melintasi jalur Registri.
- ValueNames: Sekumpulan nama nilai untuk melakukan inventarisasi kunci Registri. Jika Anda menggunakan parameter ini, Systems Manager akan menginventarisasi nama nilai yang ditentukan untuk jalur tertentu saja.

> **Note**
>
> Inventaris mengumpulkan maksimum 250 nilai kunci Registri untuk semua jalur yang ditentukan.

Berikut adalah beberapa contoh cara menentukan parameter saat melakukan inventarisasi Registri Windows.

- Kumpulkan semua kunci dan nilai secara berulang untuk jalur tertentu.

```
[{"Path":"HKEY_LOCAL_MACHINE\SOFTWARE\Amazon","Recursive": true}]
```

- Kumpulkan semua kunci dan nilai untuk jalur tertentu (pencarian berulang dimatikan).

```
[{"Path":"HKEY_LOCAL_MACHINE\SOFTWARE\Intel\PSIS\PSIS_DECODER", "Recursive": false}]
```

- Kumpulkan kunci tertentu dengan menggunakan pilihan `ValueNames`.

```
{"Path":"HKEY_LOCAL_MACHINE\SOFTWARE\Amazon\MachineImage","ValueNames":["AMIName"]}
```

Terkait Layanan AWS

Inventaris AWS Systems Manager menyediakan snapshot dari inventaris Anda saat ini untuk membantu Anda mengelola kebijakan perangkat lunak dan meningkatkan postur keamanan keseluruhan armada Anda. Anda dapat memperluas kemampuan manajemen inventaris dan migrasi Anda menggunakan yang berikutLayanan AWS:

- AWS Config menyediakan catatan historis dari perubahan pada inventaris Anda, bersama dengan kemampuan untuk membuat aturan untuk menghasilkan notifikasi saat item konfigurasi diubah. Untuk informasi lebih lanjut, lihat, Mencatat inventaris instans terkelola Amazon EC2 di Panduan Developer AWS Config.
- Application Discovery Service AWS dirancang untuk mengumpulkan inventaris pada jenis OS, inventaris aplikasi, proses, koneksi, dan metrik performa server dari VM on-premise Anda untuk mendukung keberhasilan migrasi ke AWS. Untuk informasi lebih lanjut, lihat Panduan Pengguna Application Discovery Service.

Menyiapkan Inventaris Systems Manager

Sebelum Anda menggunakanAWS Systems ManagerInventaris untuk mengumpulkan metadata tentang aplikasi, layanan,AWSkomponen dan sebagainya yang berjalan pada Node terkelola Anda, kami menyarankan agar Anda mengonfigurasi sinkronisasi data sumber daya untuk memusatkan penyimpanan data inventaris Anda di satu bucket Amazon Simple Storage Service (Amazon S3). Kami juga menyarankan agar Anda mengonfigurasi Amazon EventBridge pemantauan peristiwa inventaris. Proses ini mempermudah untuk melihat dan mengelola data dan pengumpulan inventaris.

Topik

- Pengonfigurasian sinkronisasi data sumber daya untuk Inventaris
- Tentang EventBridgepemantauan peristiwa Inventaris

Pengonfigurasian sinkronisasi data sumber daya untuk Inventaris

Topik ini menjelaskan cara menyiapkan dan mengonfigurasi sinkronisasi data sumber daya untuk Inventaris AWS Systems Manager. Untuk informasi tentang sinkronisasi data sumber daya untuk Systems Manager Explorer, lihat Menyiapkan Systems Manager Explorer untuk menampilkan data dari beberapa akun dan Wilayah.

Tentang sinkronisasi data sumber daya

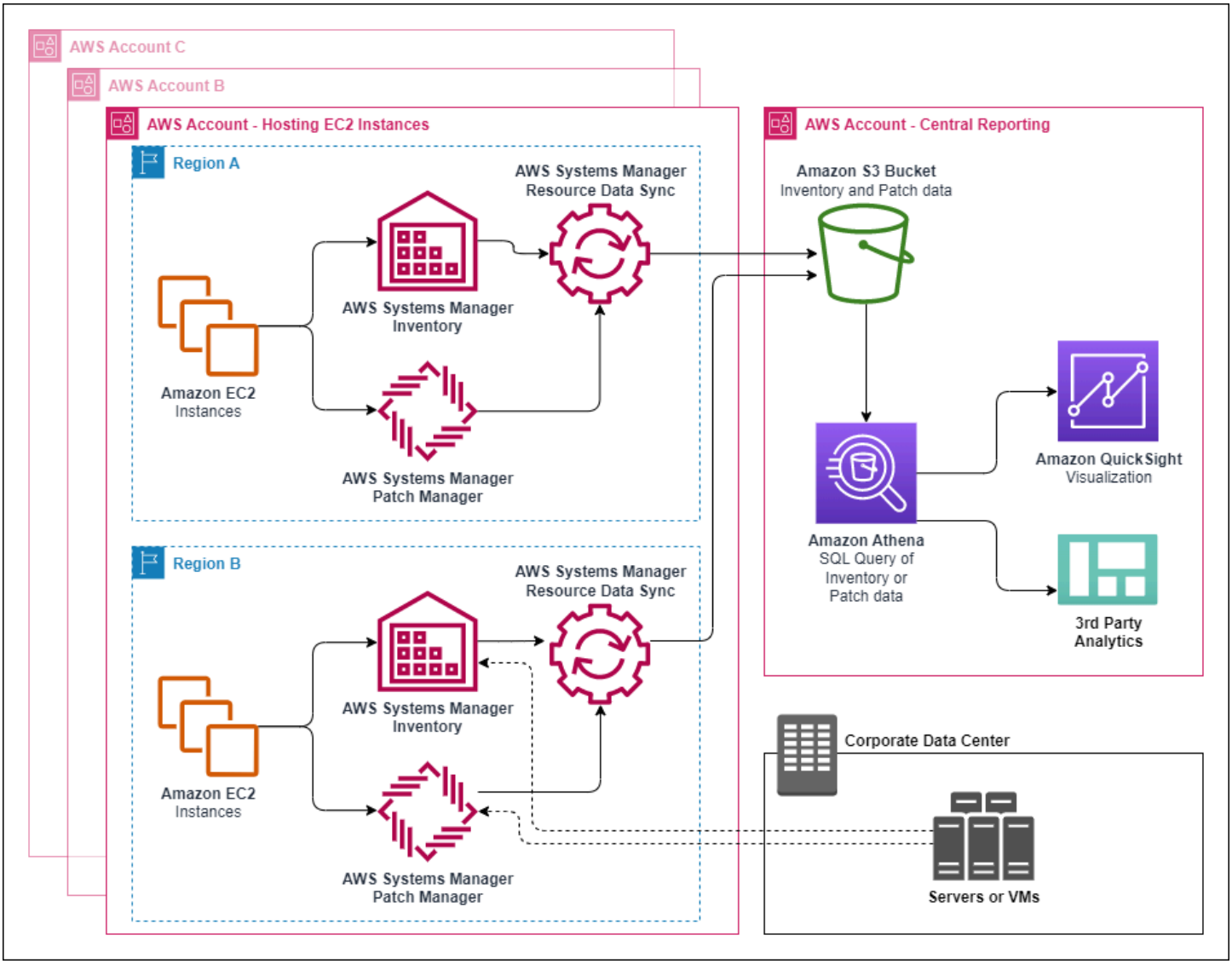
Anda dapat menggunakan sinkronisasi data sumber daya Systems Manager untuk mengirim data inventaris yang dikumpulkan dari semua node terkelola ke satu bucket Amazon Simple Storage Service (Amazon S3). Sinkronisasi data sumber daya kemudian secara otomatis memperbarui data terpusat saat data inventaris baru dikumpulkan. Dengan semua data inventaris yang disimpan dalam bucket Amazon S3 target, Anda dapat menggunakan layanan seperti Amazon Athena dan QuickSight Amazon untuk menanyakan dan menganalisis data gabungan.

Misalnya, Anda telah mengonfigurasi inventaris untuk mengumpulkan data tentang sistem operasi (OS) dan aplikasi yang berjalan pada armada 150 node terkelola. Beberapa node ini terletak di pusat data lokal, dan yang lainnya berjalan di Amazon Elastic Compute Cloud (Amazon EC2) di beberapa node. Wilayah AWS Jika Anda belum mengonfigurasi sinkronisasi data sumber daya, Anda perlu mengumpulkan data inventaris yang dikumpulkan secara manual untuk setiap node terkelola, atau Anda harus membuat skrip untuk mengumpulkan informasi ini. Anda kemudian harus mentransfer data ke dalam aplikasi sehingga Anda dapat menjalankan kueri dan menganalisisnya.

Dengan sinkronisasi data sumber daya, Anda melakukan operasi satu kali yang menyinkronkan semua data inventaris dari semua node terkelola Anda. Setelah sinkronisasi berhasil dibuat, Systems Manager membuat dasar dari semua data inventaris dan menyimpannya di bucket Amazon S3 target. Ketika data inventaris baru dikumpulkan, Systems Manager secara otomatis memperbarui data di bucket Amazon S3. Anda kemudian dapat dengan cepat dan hemat biaya mem-port data ke Amazon Athena dan Amazon. QuickSight

Diagram 1 menunjukkan bagaimana sinkronisasi data sumber daya mengumpulkan data inventaris dari Amazon EC2 dan jenis alat berat lainnya dalam lingkungan hybrid dan multicloud ke bucket Amazon S3 target. Diagram ini juga menunjukkan cara kerja sinkronisasi data sumber daya dengan beberapa Akun AWS dan Wilayah AWS.

Diagram 1: Sinkronisasi data sumber daya dengan beberapa Akun AWS dan Wilayah AWS



Jika Anda menghapus node terkelola, sinkronisasi data sumber daya akan mempertahankan file inventaris untuk node yang dihapus. Namun, untuk menjalankan node, sinkronisasi data sumber daya secara otomatis menimpa file inventaris lama saat file baru dibuat dan ditulis ke bucket Amazon S3. Jika Anda ingin melacak perubahan inventaris dari waktu ke waktu, Anda dapat menggunakan layanan AWS Config untuk melacak jenis sumber daya `SSM:ManagedInstanceInventory`. Untuk informasi lebih lanjut, lihat Memulai dengan AWS Config.

Gunakan prosedur di bagian ini untuk membuat sinkronisasi data sumber daya untuk Inventaris dengan menggunakan konsol Amazon S3 dan AWS Systems Manager. Anda juga dapat menggunakan AWS CloudFormation untuk membuat atau menghapus sinkronisasi data sumber daya. Untuk menggunakan AWS CloudFormation, tambahkan sumber daya

`AWS::SSM::ResourceDataSync` ke templat AWS CloudFormation Anda. Untuk informasi, lihat salah satu sumber daya dokumentasi berikut:

- Sumber daya AWS CloudFormation untuk sinkronisasi data sumber daya di AWS Systems Manager(blog)
- Menggunakan Templat AWS CloudFormation di Panduan Pengguna AWS CloudFormation

> **Note**
>
> Anda dapat menggunakan AWS Key Management Service (AWS KMS) untuk mengenkripsi data inventaris di bucket Amazon S3. Untuk contoh cara membuat sinkronisasi terenkripsi dengan menggunakan AWS Command Line Interface (AWS CLI) dan cara bekerja dengan data terpusat di Amazon Athena dan Amazon, lihat. QuickSight Panduan: Menggunakan sinkronisasi data sumber daya untuk mengumpulkan data inventaris

Sebelum Anda memulai

Sebelum Anda membuat sinkronisasi data sumber daya, gunakan prosedur berikut untuk membuat bucket Amazon S3 sentral untuk menyimpan data inventaris agregat. Prosedur ini menjelaskan cara menetapkan kebijakan bucket yang memungkinkan Systems Manager untuk menulis data inventaris ke bucket dari beberapa akun. Jika Anda sudah memiliki bucket Amazon S3 yang ingin Anda gunakan untuk mengumpulkan data inventaris untuk sinkronisasi data sumber daya, maka Anda harus mengonfigurasi bucket untuk menggunakan kebijakan dalam prosedur berikut.

> **Note**
>
> Inventaris Systems Manager tidak dapat menambahkan data ke bucket Amazon S3 tertentu jika bucket dikonfigurasi untuk menggunakan Object Lock. Pastikan bucket Amazon S3 yang Anda buat atau pilih untuk sinkronisasi data sumber daya tidak dikonfigurasi untuk menggunakan Amazon S3 Object Lock. Untuk informasi selengkapnya, lihat Cara Kerja Kunci Objek Amazon S3 di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

Untuk membuat dan mengonfigurasi bucket Amazon S3 untuk sinkronisasi data sumber daya

1. Buka konsol Amazon S3 di https://console.aws.amazon.com/s3/.

2. Buat bucket untuk menyimpan data Inventaris agregat Anda. Untuk informasi selengkapnya, lihat Membuat Bucket di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon. Catat nama bucket dan Wilayah AWS tempat Anda membuatnya.
3. Pilih tab Izin, dan kemudian pilih Kebijakan Bucket.
4. Salin dan tempelkan kebijakan bucket berikut ke dalam editor kebijakan. Ganti *DOC-EXAMPLE-BUCKET* dan *account-id* dengan nama bucket S3 yang Anda buat dan ID Akun AWS yang valid.

   Untuk memungkinkan beberapa Akun AWS untuk mengirim data inventaris ke bucket Amazon S3 sentral, tentukan setiap akun dalam kebijakan seperti yang ditunjukkan dalam sampel `Resource` berikut:

```
"Resource": [
    "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/accountid=123456789012/*",
    "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/accountid=444455556666/*",
    "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/accountid=777788889999/*"
],
"Condition": {
    "StringEquals": {
         "s3:x-amz-acl": "bucket-owner-full-control",
         "aws:SourceAccount": [
                  "123456789012",
                  "444455556666",
                  "777788889999"
         ]
     },
      "ArnLike": {
          "aws:SourceArn": [
            "arn:aws:ssm:*:123456789012:resource-data-sync/*",
            "arn:aws:ssm:*:444455556666:resource-data-sync/*",
            "arn:aws:ssm:*:777788889999:resource-data-sync/*"
          ]
     }
}
```

> **Note**
>
> Untuk informasi tentang cara melihat ID Akun AWS Anda, lihat ID Akun Amazon Web Services Anda dan Aliasnya di Panduan Pengguna IAM.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
        "Sid": "SSMBucketPermissionsCheck",
        "Effect": "Allow",
        "Principal": {
            "Service": "ssm.amazonaws.com"
        },
        "Action": "s3:GetBucketAcl",
        "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
        },
        {
        "Sid": " SSMBucketDelivery",
        "Effect": "Allow",
        "Principal": {
            "Service": "ssm.amazonaws.com"
        },
        "Action": "s3:PutObject",
        "Resource": [
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/accountid=ID_number/*",
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/accountid=ID_number/*",
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/accountid=ID_number/*",
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/accountid=ID_number/*"
        ],
        "Condition": {
            "StringEquals": {
                "s3:x-amz-acl": "bucket-owner-full-control",
                "aws:SourceAccount": "ID_number"
            },
            "ArnLike": {
                "aws:SourceArn": "arn:aws:ssm:*:ID_number:resource-data-sync/*"
            }
        }
      }
    ]
}
```

Membuat sinkronisasi data sumber daya untuk Inventaris

Gunakan prosedur berikut untuk membuat sinkronisasi data sumber daya untuk Inventaris Systems Manager dengan menggunakan konsol Systems Manager. Untuk informasi tentang cara membuat sinkronisasi data sumber daya dengan menggunakan AWS CLI, lihat Panduan: Mengonfigurasi node terkelola untuk Inventaris dengan menggunakan CLI.

Untuk membuat sinkronisasi data sumber daya

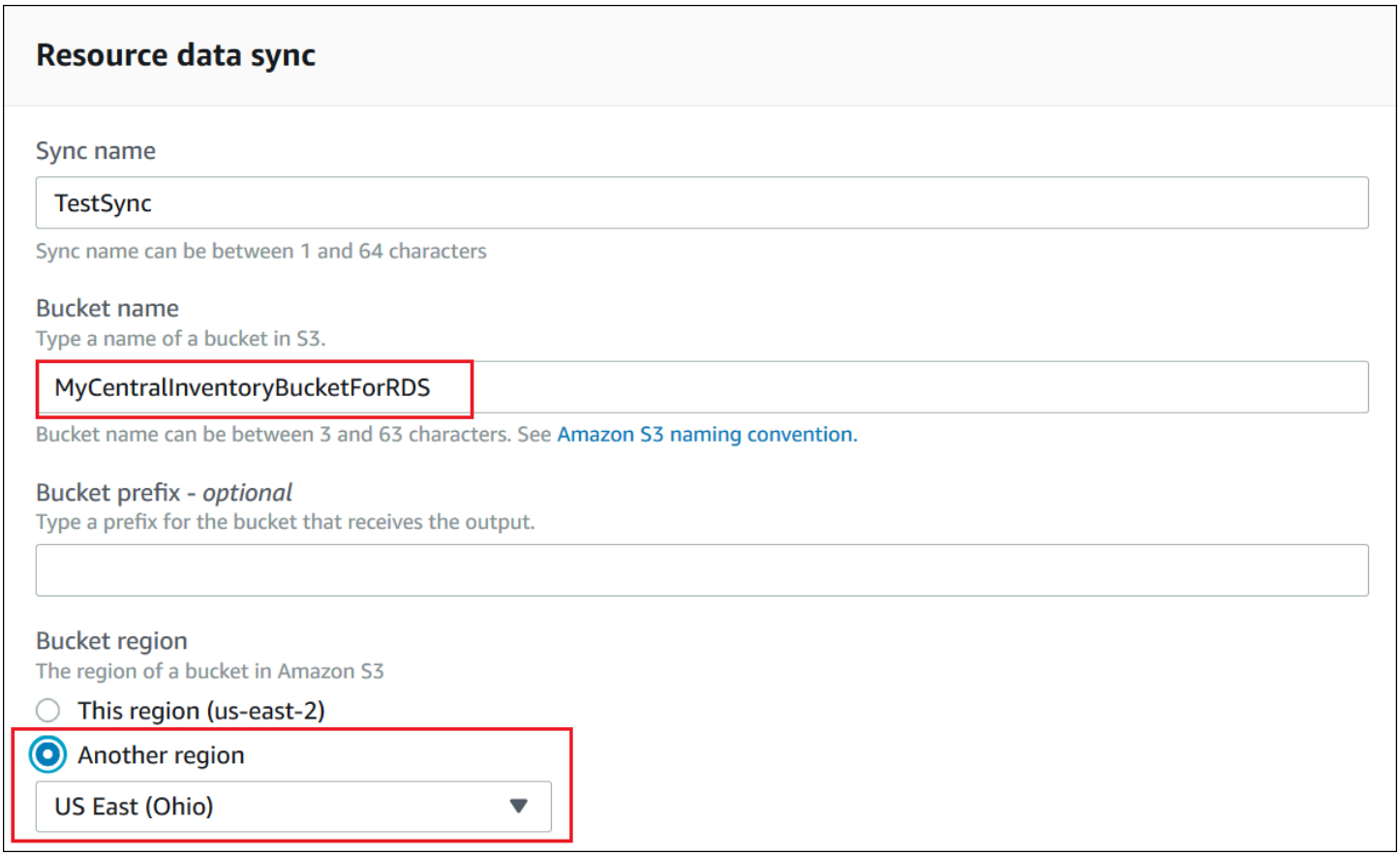
1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Di menu Pengelolaan akun, pilih Sinkronisasi data sumber daya.
4. Pilih Membuat sinkronisasi data sumber daya.
5. Di bidang Nama sinkronisasi, masukkan nama untuk konfigurasi sinkronisasi.
6. Di bidang Nama bucket, masukkan nama bucket Amazon S3 yang Anda buat menggunakan prosedur Untuk membuat dan mengonfigurasi bucket Amazon S3 untuk sinkronisasi data sumber daya.
7. (Opsional) Di bidang Prefiks bucket, masukkan nama prefiks bucket Amazon S3 (subdirektori).
8. Di bidang Wilayah Bucket, pilih Wilayah ini jika bucket Amazon S3 yang Anda buat berada di Wilayah AWS saat ini. Jika bucket berada di Wilayah AWS yang berbeda, pilih Wilayah lain, dan masukkan nama Wilayah.

   > **Note**
   >
   > Jika sinkronisasi dan bucket Amazon S3 target berada di berbagai wilayah, Anda mungkin akan dibebani harga transfer data. Untuk informasi lebih lanjut, lihat Harga Amazon S3.
9. (Opsional) Di bidang ARN Kunci KMS, ketik atau tempelkan ARN Kunci KMS untuk mengenkripsi data inventaris di Amazon S3.
10. Pilih Buat.

Untuk menyinkronkan data inventaris dari beberapa Wilayah AWS, Anda harus membuat sinkronisasi data sumber daya di masing-masing Wilayah. Ulangi prosedur ini di masing-masing Wilayah AWS tempat Anda ingin mengumpulkan data inventaris dan mengirimkannya ke bucket Amazon S3 sentral. Ketika Anda membuat sinkronisasi di setiap Wilayah, tentukan bucket Amazon S3 sentral di bidang Nama bucket. Kemudian gunakan pilihan Wilayah bucket untuk memilih Wilayah tempat Anda membuat bucket Amazon S3 sentral, seperti yang ditunjukkan pada cuplikan layar berikut. Di lain waktu ketika asosiasi berjalan untuk mengumpulkan data inventaris, Systems Manager menyimpan data di bucket Amazon S3 sentral.



Pembuatan sinkronisasi data sumber daya inventaris untuk akun yang ditentukan dalam AWS Organizations

Anda dapat menyinkronkan data inventaris dari Akun AWS yang ditentukan dalam AWS Organizations ke bucket Amazon S3 sentral. Setelah Anda menyelesaikan prosedur berikut, data inventaris disinkronkan ke prefiks kunci individu Amazon S3 di bucket sentral. Setiap prefiks kunci mewakili berbagai ID Akun AWS.

Sebelum Anda memulai

Sebelum memulai, pastikan Anda menyiapkan dan mengonfigurasi Akun AWS di AWS Organizations. Untuk informasi lebih lanjut, lihat di Panduan Pengguna AWS Organizations.

Selain itu, perhatikan bahwa Anda harus membuat sinkronisasi data sumber daya berbasis organisasi untuk setiap Wilayah AWS dan Akun AWS yang ditentukan dalam AWS Organizations.

Pembuatan bucket Amazon S3 sentral

Gunakan prosedur berikut untuk membuat bucket Amazon S3 sentral untuk menyimpan data inventaris agregat. Prosedur ini menjelaskan cara menetapkan kebijakan bucket yang memungkinkan Systems Manager untuk menulis data inventaris ke bucket dari ID akun AWS Organizations Anda. Jika Anda sudah memiliki bucket Amazon S3 yang ingin Anda gunakan untuk mengumpulkan data inventaris untuk sinkronisasi data sumber daya, maka Anda harus mengonfigurasi bucket untuk menggunakan kebijakan dalam prosedur berikut.

Untuk membuat dan mengonfigurasi bucket Amazon S3 untuk sinkronisasi data sumber daya untuk beberapa akun yang ditentukan dalam AWS Organizations

1. Buka konsol Amazon S3 di https://console.aws.amazon.com/s3/.
2. Buat bucket untuk menyimpan data inventaris agregat Anda. Untuk informasi selengkapnya, lihat Membuat Bucket di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon. Catat nama bucket dan Wilayah AWS tempat Anda membuatnya.
3. Pilih tab Izin, dan kemudian pilih Kebijakan Bucket.
4. Salin dan tempelkan kebijakan bucket berikut ke dalam editor kebijakan. Ganti *DOC-EXAMPLE-BUCKET* dan *organization-id* dengan nama bucket Amazon S3 yang Anda buat dan ID akun AWS Organizations yang valid.

   Secara opsional, ganti *bucket-prefix* dengan nama prefiks Amazon S3 (subdirektori). Jika Anda tidak membuat prefiks, hapus *bucket-prefix/* dari ARN dalam kebijakan berikut.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "SSMBucketPermissionsCheck",
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": "s3:GetBucketAcl",
```

```
      "Resource": "arn:aws:s3:::S3_bucket_name"
    },
    {
      "Sid": " SSMBucketDelivery",
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": "s3:PutObject",
      "Resource": [
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET/bucket-prefix/*/accountid=*/*"
      ],
      "Condition": {
        "StringEquals": {
          "s3:x-amz-acl": "bucket-owner-full-control",
          "s3:RequestObjectTag/OrgId": "organization-id"
                  }
      }
    },
    {
      "Sid": " SSMBucketDeliveryTagging",
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": "s3:PutObjectTagging",
      "Resource": [
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET/bucket-prefix/*/accountid=*/*"
      ]
    }
  ]
}
```

Membuat sinkronisasi data sumber daya inventaris untuk akun yang ditentukan dalam AWS Organizations

Prosedur berikut menjelaskan cara menggunakan AWS CLI untuk membuat sinkronisasi data sumber daya untuk akun yang ditentukan dalam AWS Organizations. Anda harus menggunakan AWS CLI untuk melakukan tugas ini. Anda juga harus melakukan prosedur ini untuk setiap Wilayah AWS dan Akun AWS yang ditentukan dalam AWS Organizations.

Untuk membuat sinkronisasi data sumber daya untuk akun yang ditentukan dalam AWS Organizations (AWS CLI)

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI.

2. Jalankan perintah berikut untuk memverifikasi bahwa Anda tidak memiliki sinkronisasi data sumber daya lainnya. Anda hanya dapat memiliki satu sinkronisasi data sumber daya berbasis organisasi.

   ```
   aws ssm list-resource-data-sync
   ```

   Jika perintah menampilkan sinkronisasi data sumber daya lainnya, Anda harus menghapusnya atau memilih untuk tidak membuat yang baru.

3. Jalankan perintah berikut untuk membuat sinkronisasi data sumber daya untuk akun yang ditentukan dalam AWS Organizations. Untuk *DOC-EXAMPLE-BUCKET*, tentukan nama bucket Amazon S3 yang Anda buat sebelumnya dalam topik ini. Jika Anda membuat prefiks (subdirektori) untuk bucket Anda, maka tentukan informasi ini untuk *prefix-name*.

   ```
   aws ssm create-resource-data-sync --sync-name name --s3-
   destination "BucketName=DOC-EXAMPLE-BUCKET,Prefix=prefix-
   name,SyncFormat=JsonSerDe,Region=Wilayah AWS, for example us-
   east-2,DestinationDataSharing={DestinationDataSharingType=Organization}"
   ```

4. Ulangi Langkah 2 dan 3 untuk setiap Wilayah AWS dan Akun AWS tempat Anda ingin menyinkronkan data ke bucket Amazon S3 sentral.

Mengelola sinkronisasi data sumber daya

Masing-masing Akun AWS dapat memiliki 5 sinkronisasi data sumber daya perWilayah AWS. Anda dapat menggunakan AWS Systems Manager Fleet Manager konsol untuk mengelola sinkronisasi data sumber daya Anda.

Untuk melihat sinkronisasi data sumber daya

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

-atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.

3. Di menu tarik-turun Manajemen akun, pilih Sinkronisasi data sumber daya.
4. Pilih sinkronisasi data sumber daya dari tabel, lalu pilih Lihat detail untuk melihat informasi tentang sinkronisasi data sumber daya Anda.

Untuk menghapus sinkronisasi data sumber daya

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Di menu tarik-turun Manajemen akun, pilih Sinkronisasi data sumber daya.
4. Pilih sinkronisasi data sumber daya dari tabel, lalu pilih Hapus.

Tentang EventBridgepemantauan peristiwa Inventaris

Anda dapat mengonfigurasi aturan di Amazon EventBridge untuk membuat sebuah acara dalam menanggapiAWS Systems ManagerPerubahan status sumber daya inventaris. EventBridge mendukung peristiwa untuk perubahan Tahapan Inventaris berikut. Semua peristiwa dikirimkan berdasarkan upaya terbaik.

Kejadian yang dihapus dengan jenis inventaris kustom pada instans tertentu: Jika aturan dikonfigurasi untuk memantau peristiwa ini, EventBridgemembuat peristiwa ketika jenis inventaris kustom pada dikelola tertentu dihapus. EventBridgemengirimkan satu peristiwa per node per jenis inventaris kustom. Berikut ini adalah sampel pola peristiwa.

```
{
    "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"
    }
}
```

Kejadian yang dihapus dengan jenis inventaris kustom pada semua instans: Jika aturan dikonfigurasi untuk memantau peristiwa ini, EventBridgemembuat acara ketika jenis inventaris khusus untuk semua node yang dikelola dihapus. Berikut ini adalah sampel pola peristiwa.

```
{
    "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"
    }
}
```

PutInventorypanggilan dengan peristiwa versi skema lama: Jika aturan dikonfigurasi untuk memantau peristiwa ini, EventBridge menciptakan sebuah acara ketika PutInventory panggilan dibuat yang menggunakan versi skema yang lebih rendah dari skema saat ini. Peristiwa ini berlaku untuk semua jenis inventaris. Berikut ini adalah sampel pola peristiwa.

```
{
    "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"
    }
}
```

Untuk informasi tentang cara mengkonfigurasi EventBridge untuk memantau peristiwa ini, lihat EventBridge Pengonfigurasan peristiwa Systems Manager.

Pengonfigurasian pengumpulan inventaris

Bagian ini menjelaskan cara mengonfigurasi koleksi AWS Systems Manager Inventaris pada satu atau beberapa node terkelola menggunakan konsol Systems Manager. Untuk contoh cara mengonfigurasi pengumpulan inventaris dengan menggunakan AWS Command Line Interface (AWS CLI), lihat Panduan Inventaris Systems Manager.

Saat Anda mengonfigurasi koleksi inventaris, Anda mulai dengan membuat AWS Systems Manager State Manager asosiasi. Systems Manager mengumpulkan data inventaris saat asosiasi dijalankan. Jika Anda tidak membuat asosiasi terlebih dahulu, dan mencoba memanggil

`aws:softwareInventory` plugin dengan menggunakan, misalnya AWS Systems ManagerRun Command, sistem mengembalikan kesalahan berikut: `The aws:softwareInventory plugin can only be invoked via ssm-associate.`

> **Note**
>
> Waspadai perilaku berikut jika Anda membuat beberapa asosiasi inventaris untuk node terkelola:
>
> - Setiap node dapat diberi asosiasi inventaris yang menargetkan semua node (--target “Key=InstanceIds, Values=*”).
> - Setiap node juga dapat diberi asosiasi tertentu yang menggunakan pasangan kunci tag/nilai atau grup AWS sumber daya.
> - Jika sebuah node diberi beberapa asosiasi inventaris, status akan ditampilkan Dilewati untuk asosiasi yang belum berjalan. Asosiasi yang berjalan akhir-akhir ini menampilkan status sebenarnya dari asosiasi inventaris.
> - Jika sebuah node diberi beberapa asosiasi inventaris dan masing-masing menggunakan pasangan kunci/nilai tag, maka asosiasi inventaris tersebut gagal berjalan di node karena konflik tag. Asosiasi masih berjalan pada node yang tidak memiliki konflik kunci tag/nilai.

Sebelum Anda Memulai

Sebelum Anda mengonfigurasi pengumpulan inventaris, selesaikan tugas berikut.

- Perbarui AWS Systems Manager SSM Agent pada node yang ingin Anda inventarisasi. Dengan menjalankan versi terbaruSSM Agent, Anda memastikan bahwa Anda dapat mengumpulkan metadata untuk semua jenis inventaris yang didukung. Untuk informasi tentang cara memperbarui SSM Agent dengan menggunakanState Manager, lihatWalkthrough: Perbarui secara otomatis (SSM AgentCLI).
- Pastikan Anda telah menyelesaikan persyaratan penyiapan untuk instans Amazon Elastic Compute Cloud (Amazon EC2) dan mesin non-EC2 di lingkungan hybrid dan multicloud. Untuk informasi, lihat Menyiapkan AWS Systems Manager.
- Untuk node Microsoft Windows, verifikasi bahwa node terkelola Anda dikonfigurasi dengan Windows PowerShell 3.0 (atau yang lebih baru). SSM Agentmenggunakan `ConvertTo-Json` cmdlet PowerShell untuk mengonversi data inventaris pembaruan Windows ke format yang diperlukan.

- (Opsional) Buat sinkronisasi data sumber daya untuk menyimpan data inventaris secara terpusat di bucket Amazon S3. Sinkronisasi data sumber daya kemudian secara otomatis memperbarui data terpusat saat data inventaris baru dikumpulkan. Untuk informasi selengkapnya, lihat Pengonfigurasian sinkronisasi data sumber daya untuk Inventaris.
- (Opsional) Buat file JSON untuk mengumpulkan inventaris kustom. Untuk informasi selengkapnya, lihat Menggunakan inventaris kustom.

Inventarisasi semua node terkelola di Akun AWS

Anda dapat menginventarisasi semua node terkelola di dalam Anda Akun AWS dengan membuat asosiasi inventaris global. Asosiasi inventaris global akan melakukan tindakan berikut:

- Secara otomatis menerapkan konfigurasi inventaris global (asosiasi) ke semua node terkelola yang ada di AndaAkun AWS. Node terkelola yang sudah memiliki asosiasi inventaris dilewati saat asosiasi inventaris global diterapkan dan dijalankan. Ketika sebuah node dilewati, pesan status terperinci menyatakan`0verridden By Explicit Inventory Association`. Node tersebut dilewati oleh asosiasi global, tetapi mereka masih akan melaporkan inventaris ketika mereka menjalankan asosiasi inventaris yang ditugaskan.
- Secara otomatis menambahkan node baru yang dibuat di asosiasi inventaris global AndaAkun AWS.

> **Note**
>
> - Jika node terkelola dikonfigurasi untuk asosiasi inventaris global, dan Anda menetapkan asosiasi tertentu ke node tersebut, maka Systems Manager Inventory akan menurunkan prioritas asosiasi global dan menerapkan asosiasi tertentu.
> - Asosiasi inventaris global tersedia dalam SSM Agent versi 2.0.790.0 atau yang lebih baru. Untuk informasi tentang cara memperbarui SSM Agent pada node Anda, lihatMemperbarui SSM Agent penggunaan Run Command.

Pengonfigurasian pengumpulan inventaris dengan satu klik (konsol)

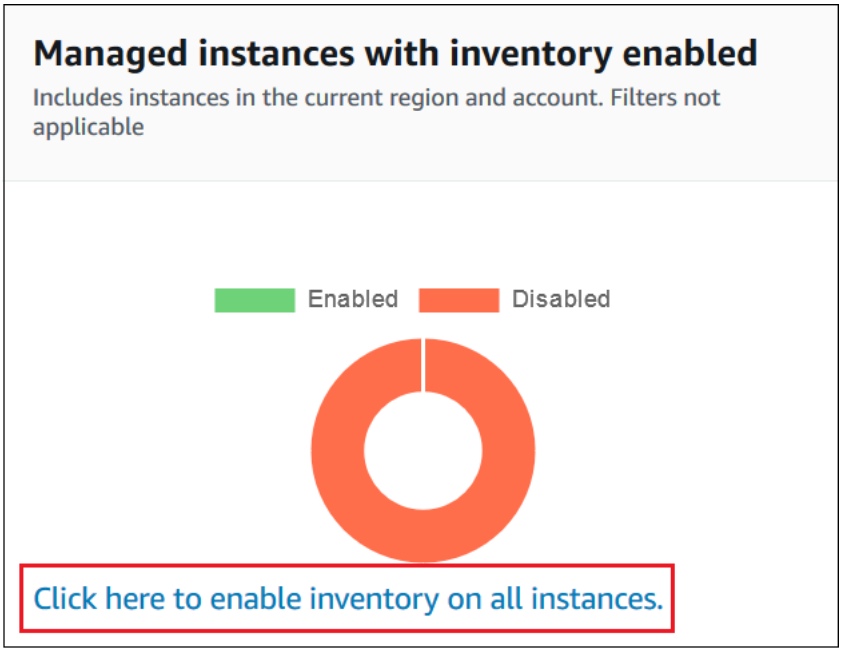
Gunakan prosedur berikut untuk mengonfigurasi Systems Manager Inventory untuk semua node terkelola dalam Anda Akun AWS dan dalam satu nodeWilayah AWS.

Untuk mengonfigurasi semua node terkelola di inventaris Region for Systems Manager saat ini

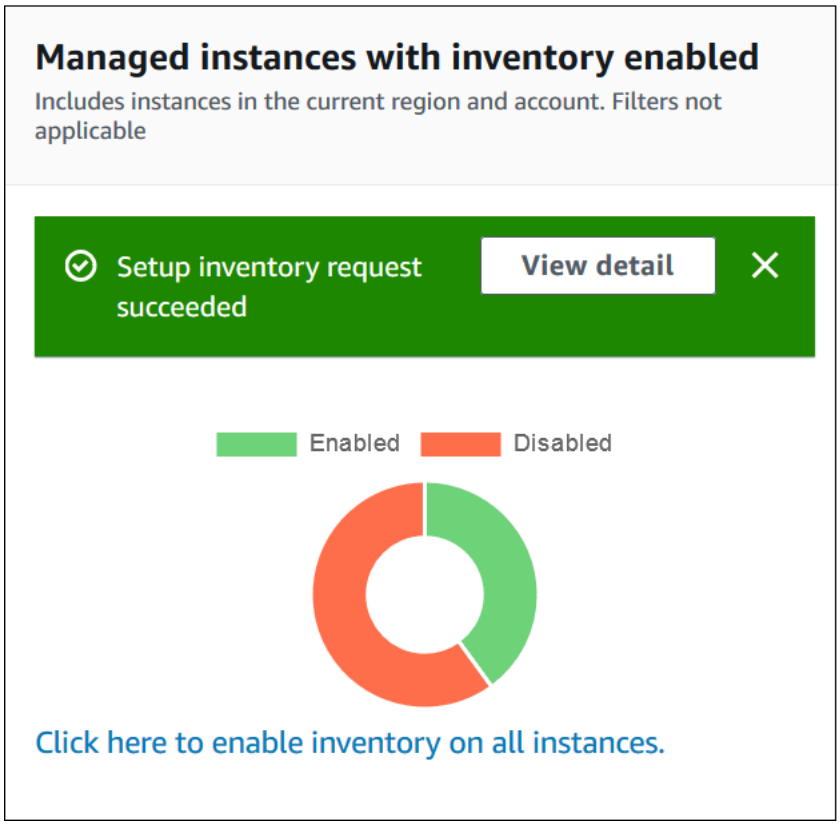
1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Inventaris.

   -atau-

   Jika halaman beranda AWS Systems Manager terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Inventaris di panel navigasi.
3. Di kartu Instans terkelola dengan inventaris diaktifkan, pilih Klik di sini untuk mengaktifkan inventaris pada semua instans.



   Jika berhasil, konsol menampilkan pesan berikut.



Bergantung pada jumlah node terkelola di akun Anda, diperlukan beberapa menit agar asosiasi inventaris global diterapkan. Tunggu beberapa menit lalu segarkan halaman. Verifikasi bahwa grafik berubah untuk mencerminkan bahwa inventaris dikonfigurasi pada semua node terkelola Anda.

Pengonfigurasian pengumpulan dengan menggunakan konsol

Bagian ini mencakup informasi tentang cara mengonfigurasi Inventaris Systems Manager untuk mengumpulkan metadata dari node terkelola menggunakan konsol Systems Manager. Anda dapat dengan cepat mengumpulkan metadata dari semua node dalam node tertentu Akun AWS (dan node masa depan apa pun yang mungkin dibuat di akun itu) atau Anda dapat mengumpulkan data inventaris secara selektif dengan menggunakan tag atau ID node.

> **Note**
>
> Sebelum menyelesaikan prosedur ini, periksa untuk melihat apakah asosiasi inventaris global sudah ada. Jika asosiasi inventaris global sudah ada, kapan pun Anda meluncurkan instance baru, asosiasi akan diterapkan padanya, dan instance baru akan diinventarisasi.

Untuk mengonfirgurasi pengumpulan inventaris

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Inventaris.

   -atau-

   Jika halaman beranda AWS Systems Manager terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Inventaris di panel navigasi.
3. Pilih Siapkan Inventaris.
4. Di bagian Target, identifikasi node tempat Anda ingin menjalankan operasi ini dengan memilih salah satu dari berikut ini.

   - Memilih semua instans terkelola di akun ini - Opsi ini memilih semua node terkelola yang tidak ada asosiasi inventaris yang ada. Jika Anda memilih opsi ini, node yang sudah memiliki asosiasi inventaris akan dilewati selama pengumpulan inventaris, dan ditampilkan dengan status Dilewati dalam hasil inventaris. Untuk informasi selengkapnya, lihat Inventarisasi semua node terkelola di Akun AWS.
   - Menentukan tag - Gunakan opsi ini untuk menentukan satu tag untuk mengidentifikasi node di akun Anda dari mana Anda ingin mengumpulkan inventaris. Jika Anda menggunakan tag, node apa pun yang dibuat di masa depan dengan tag yang sama juga akan melaporkan inventaris. Jika ada asosiasi inventaris yang ada dengan semua node, menggunakan tag untuk memilih node tertentu sebagai target untuk inventaris yang berbeda akan mengesampingkan keanggotaan node dalam grup target Semua instance terkelola. Node terkelola dengan tag yang ditentukan dilewati pada koleksi inventaris future dari Semua instance terkelola.
   - Memilih instance secara manual - Gunakan opsi ini untuk memilih node terkelola tertentu di akun Anda. Secara eksplisit memilih node tertentu dengan menggunakan opsi ini mengesampingkan asosiasi inventaris pada target Semua instance terkelola. Node dilewati pada koleksi inventaris masa depan dari Semua instance terkelola.

   > **Note**
   >
   > Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

5. Di bagian Jadwal, pilih seberapa sering Anda ingin sistem mengumpulkan metadata inventaris dari node Anda.
6. Di bagian Parameter, gunakan daftar untuk mengaktifkan atau menonaktifkan berbagai jenis pengumpulan inventaris. Lihat sampel berikut jika Anda ingin membuat pencarian inventaris untuk File atau Registri Windows.

   Berkas

   - Pada Linux dan macOS, kumpulkan metadata dari file.sh di direktori `/home/ec2-user`, tanpa menyertakan semua subdirektori.

     ```
     [{"Path":"/home/ec2-user","Pattern":["*.sh", "*.sh"],"Recursive":false}]
     ```

   - Pada Windows, kumpulkan metadata dari semua file “.exe” di folder Program Files, termasuk subdirektori secara berulang.

     ```
     [{"Path":"C:\Program Files","Pattern":["*.exe"],"Recursive":true}]
     ```

   - Pada Windows, kumpulkan metadata dari pola log tertentu.

     ```
     [{"Path":"C:\ProgramData\Amazon","Pattern":["*amazon*.log"],"Recursive":true}]
     ```

   - Batasi jumlah direktori saat melakukan pengumpulan berulang.

     ```
     [{"Path":"C:\Users","Pattern":["*.ps1"],"Recursive":true, "DirScanLimit": 1000}]
     ```

   Registri Windows

   - Kumpulkan semua kunci dan nilai secara berulang untuk jalur tertentu.

     ```
     [{"Path":"HKEY_LOCAL_MACHINE\SOFTWARE\Amazon","Recursive": true}]
     ```

   - Kumpulkan semua kunci dan nilai untuk jalur tertentu (pencarian berulang dimatikan).

     ```
     [{"Path":"HKEY_LOCAL_MACHINE\SOFTWARE\Intel\PSIS\PSIS_DECODER", "Recursive":
      false}]
     ```

   - Kumpulkan kunci tertentu dengan menggunakan pilihan `ValueNames`.

```
{"Path":"HKEY_LOCAL_MACHINE\SOFTWARE\Amazon\MachineImage","ValueNames":
["AMIName"]}
```

Untuk informasi lebih lanjut tentang pengumpulan inventaris File dan Registri Windows, lihat Menggunakan file dan inventaris registri Windows.

7. Di bagian Lanjutan, pilih Menyinkronkan log eksekusi inventaris ke bucket Amazon S3 jika Anda ingin menyimpan status eksekusi asosiasi di bucket Amazon S3.
8. Pilih Siapkan Inventaris. Systems Manager membuat State Manager asosiasi dan segera menjalankan Inventory pada node.
9. Di panel navigasi, pilih State Manager. Pastikan asosiasi baru dibuat dengan menggunakan dokumen **`AWS-GatherSoftwareInventory`**. Jadwal asosiasi menggunakan ekspresi tarif. Juga, pastikan bidang Status menunjukkan Berhasil. Jika Anda memilih pilihan untuk Menyinkronkan log eksekusi inventaris ke bucket Amazon S3, maka Anda dapat melihat data catatan di Amazon S3 setelah beberapa menit. Jika Anda ingin melihat data inventaris untuk node tertentu, pilih Instans Terkelola di panel navigasi.
10. Pilih simpul, lalu pilih Lihat detail.
11. Pada halaman detail simpul, pilih Inventaris. Gunakan daftar Jenis inventaris untuk memfilter inventaris.

Menggunakan data inventaris Systems Manager

Bagian ini mencakup topik yang menjelaskan cara mengkueri dan mengumpulkan data Inventaris AWS Systems Manager.

Topik

- Mengkueri data inventaris dari beberapa Wilayah dan akun
- Mengkueri pengumpulan inventaris dengan menggunakan filter
- Pengumpulan data inventaris

Mengkueri data inventaris dari beberapa Wilayah dan akun

Inventaris AWS Systems Manager berintegrasi dengan Amazon Athena untuk membantu Anda mengkueri data inventaris dari beberapa Wilayah AWS dan Akun AWS. Integrasi Athena

menggunakan sinkronisasi data sumber daya sehingga Anda dapat melihat data inventaris dari semua node terkelola pada halaman TampilanAWS Systems Manager Detail.

> **Important**
>
> Fitur ini menggunakan AWS Glue untuk merayapkan data di bucket Amazon Simple Storage Service (Amazon S3), dan Amazon Athena untuk mengkueri data. Tergantung dari berapa banyak data yang dirayapkan dan dikueri, Anda dapat dibebani biaya untuk penggunaan layanan ini. Dengan AWS Glue, Anda membayar tarif per jam, ditagih per detik, untuk perayap (menemukan data) dan pekerjaan ETL (pemrosesan dan pemuatan data). Dengan Athena, Anda dibebani biaya berdasarkan jumlah data yang dipindai oleh setiap kueri. Kami menganjurkan agar Anda melihat pedoman harga untuk layanan ini sebelum Anda menggunakan integrasi Amazon Athena dengan Inventaris Systems Manager. Untuk informasi lebih lanjut, lihat Harga Amazon Athena dan Harga AWS Glue.

Anda dapat melihat data inventaris pada halaman Tampilan Detail di semuaWilayah AWS tempat Amazon Athena tersedia. Untuk daftar Wilayah yang didukung, lihat Titik Akhir Layanan Amazon Athena di bagian Referensi Umum Amazon Web Services.

Sebelum Anda memulai

Integrasi Athena menggunakan sinkronisasi data sumber daya. Anda harus menyiapkan dan mengonfigurasi sinkronisasi data sumber daya untuk menggunakan fitur ini. Untuk informasi selengkapnya, lihat Pengonfigurasian sinkronisasi data sumber daya untuk Inventaris.

Juga, perhatikan bahwa halaman Tampilan Detail. Jika Anda bukan pemilik bucket Amazon S3 sentral, maka Anda tidak akan melihat data inventaris pada halaman Tampilan Detail.

Pengonfigurasian akses

Sebelum Anda dapat mengkueri dan melihat data dari beberapa account dan Wilayah pada Tampilan Detail di konsol Systems Manager, Anda harus mengonfigurasi identitas IAM.

Jika data inventaris disimpan dalam bucket Amazon S3 yang menggunakanAWS Key Management Service (AWS KMS) enkripsi, Anda juga harus mengonfigurasi entitas IAM dan peran`Amazon-GlueServiceRoleForSSM` layanan untukAWS KMS enkripsi.

Topik

- Mengkonfigurasi entitas IAM Anda untuk mengakses halaman Tampilan Terperinci

- (Opsional) Konfigurasikan izin untuk melihat dataAWS KMS terenkripsi

Mengkonfigurasi entitas IAM Anda untuk mengakses halaman Tampilan Terperinci

Berikut ini menjelaskan izin minimum yang diperlukan untuk melihat data inventaris pada halaman Tampilan Terperinci.

Kebijakan yang**`AWSQuicksightAthenaAccess`** dikelola

Blok izin berikut`PassRole` dan tambahan yang diperlukan

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowGlue",
            "Effect": "Allow",
            "Action": [
                "glue:GetCrawler",
                "glue:GetCrawlers",
                "glue:GetTables",
                "glue:StartCrawler",
                "glue:CreateCrawler"
            ],
            "Resource": "*"
        },
        {
            "Sid": "iamPassRole",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "glue.amazonaws.com"
                }
            }
        },
        {
            "Sid": "iamRoleCreation",
           "Effect": "Allow",
            "Action": [
                "iam:CreateRole",
                "iam:AttachRolePolicy"
```

```
            ],
            "Resource": "arn:aws:iam::account_ID:role/*"
        },
        {
            "Sid": "iamPolicyCreation",
            "Effect": "Allow",
            "Action": "iam:CreatePolicy",
            "Resource": "arn:aws:iam::account_ID:policy/*"
        }
    ]
}
```

(Opsional) Jika bucket Amazon S3 yang digunakan untuk menyimpan data inventaris dienkripsi dengan menggunakanAWS KMS, Anda juga harus menambahkan blok berikut ke kebijakan.

```
{
    "Effect": "Allow",
    "Action": [
        "kms:Decrypt"
    ],
    "Resource": [
        "arn:aws:kms:Region:account_ID:key/key_ARN"
    ]
}
```

Untuk menyediakan akses, tambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup diAWS IAM Identity Center:

  Buat set izin. Ikuti petunjuk di Buat set izin di PanduanAWS IAM Identity Center Pengguna.
- Pengguna yang dikelola dalam IAM melalui penyedia identitas:

  Membuat peran untuk Ikuti petunjuk dalam Membuat peran untuk penyedia identitas pihak ketiga (federasi) di Panduan Pengguna IAM.
- Pengguna IAM:
  - Buat peran yang dapat diasumsikan pengguna Anda. Ikuti petunjuk dalam Membuat peran untuk pengguna IAM di Panduan Pengguna IAM.
  - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk dalam Menambahkan izin ke pengguna (konsol) di Panduan Pengguna IAM.

(Opsional) Konfigurasikan izin untuk melihat dataAWS KMS terenkripsi

Jika bucket Amazon S3 yang digunakan untuk menyimpan data inventaris dienkripsi dengan menggunakanAWS Key Management Service (AWS KMS), Anda harus mengonfigurasi entitas IAM dan peran Amazon-GlueServiceRoleFor SSM dengan`kms:Decrypt` izin untukAWS KMS kunci tersebut.

Sebelum Anda memulai

Untuk memberikan`kms:Decrypt` izin untukAWS KMS kunci, tambahkan blok kebijakan berikut ke entitas IAM Anda:

```
{
    "Effect": "Allow",
    "Action": [
        "kms:Decrypt"
    ],
    "Resource": [
        "arn:aws:kms:Region:account_ID:key/key_ARN"
    ]
}
```

Jika Anda belum melakukannya, lengkapi prosedur tersebut dan tambahkan`kms:Decrypt` izin untukAWS KMS kunci.

Gunakan prosedur berikut untuk mengonfigurasi peran Amazon-GlueServiceRoleFor SSM dengan`kms:Decrypt` izin untukAWS KMS kunci.

Untuk mengonfigurasi peran Amazon-GlueServiceRoleFor SSM dengan**`kms:Decrypt`** izin

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran, lalu gunakan bidang pencarian untuk menemukan peran Amazon-GlueServiceRoleFor SSM. Halaman Ringkasan terbuka.
3. Gunakan kolom pencarian untuk menemukan peran Amazon-GlueServiceRoleFor SSM. Pilih nama peran. Halaman Ringkasan terbuka.
4. Pilih nama peran. Halaman Ringkasan terbuka.
5. Pilih Tambahkan kebijakan inline. Halaman Buat kebijakan terbuka.
6. Pilih tab JSON.
7. Hapus teks JSON yang ada di editor, lalu salin dan tempelkan kebijakan berikut ke editor JSON.

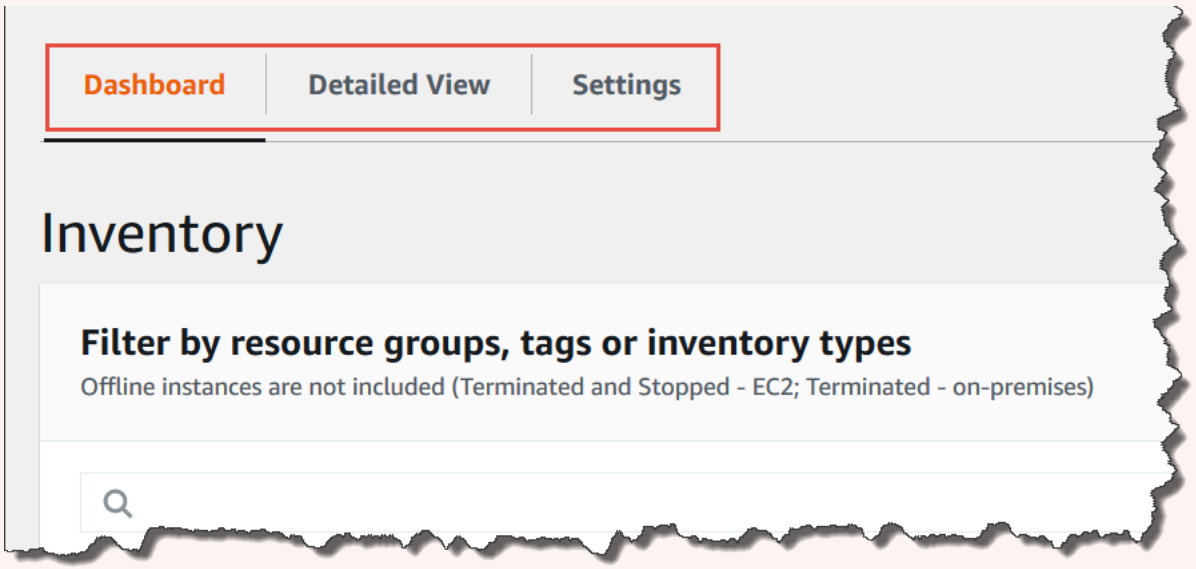
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:Region:account_ID:key/key_ARN"
            ]
        }
    ]
}
```

8. Pilih Tinjau kebijakan
9. Pada halaman Tinjau Kebijakan, masukkan nama di bidang Nama.
10. Pilih Buat kebijakan.

Mengkueri data pada halaman tampilan detail inventaris

Gunakan prosedur berikut untuk menampilkan data inventaris dari beberapa Wilayah AWS dan Akun AWS pada halaman Tampilan Detail Inventaris Systems Manager .

> **Important**
>
> Halaman Tampilan Detail Inventaris hanya tersedia di Wilayah AWS yang menawarkan Amazon Athena. Jika tab berikut tidak ditampilkan pada halaman Inventaris Systems Manager, artinya Athena tidak tersedia di Wilayah dan Anda tidak dapat menggunakan Tampilan Detail untuk mengkueri data.

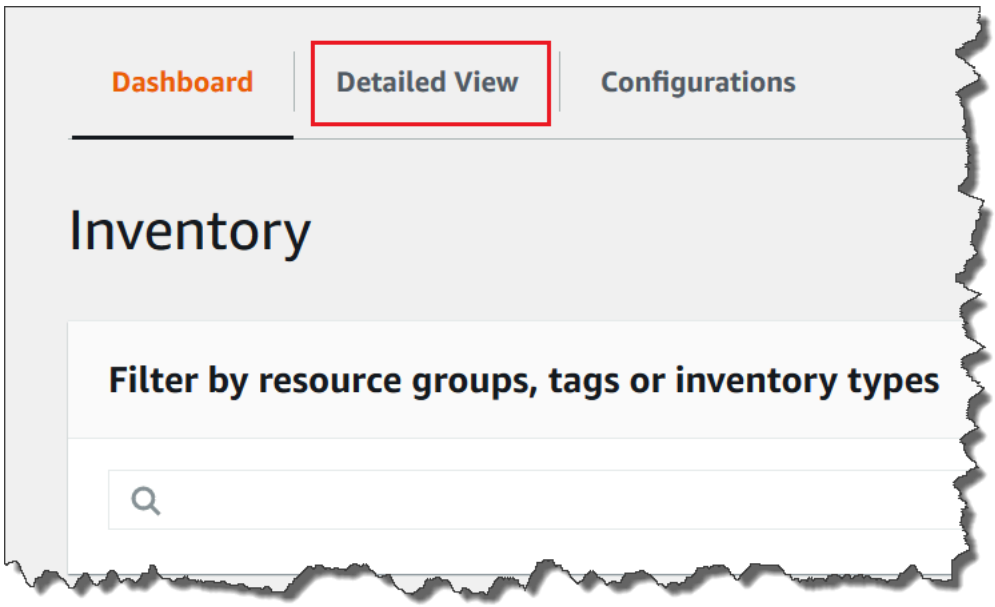


Untuk melihat data inventaris dari beberapa Wilayah dan akun di konsol AWS Systems Manager

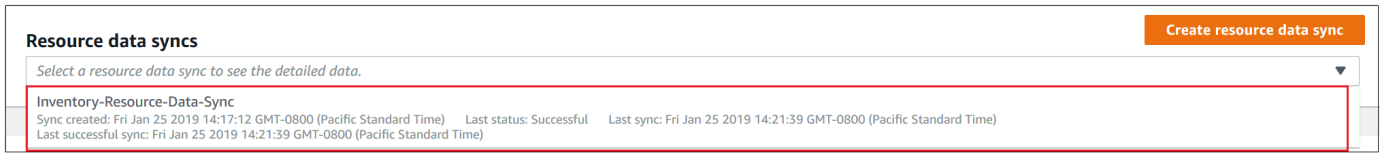
1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Inventaris.

   -atau-

   Jika halaman beranda AWS Systems Manager terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Inventaris di panel navigasi.
3. Pilih tab Tampilan Detail.

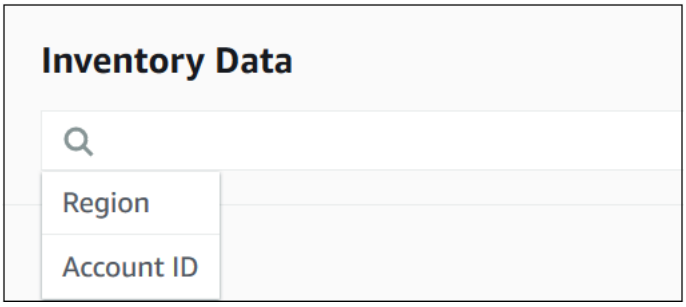


4. Pilih sinkronisasi data sumber daya yang datanya ingin Anda kueri.



5. Di daftar Jenis Inventaris, pilih jenis data inventaris yang ingin Anda kueri, lalu tekan Enter.

6. Untuk memfilter data, pilih batang Filter, lalu pilih pilihan filter.



Anda dapat menggunakan tombol Ekspor ke CSV untuk melihat himpunan kueri saat ini di aplikasi lembar kerja seperti Microsoft Excel. Anda juga dapat menggunakan Riwayat Kueri dan Jalankan Kueri Lanjutan untuk melihat detail riwayat dan berinteraksi dengan data Anda di Amazon Athena.

Pengeditan jadwal perayap AWS Glue

AWS Glue merayapkan data inventaris di bucket Amazon S3 sentral sebanyak dua kali sehari, secara default. Jika Anda sering mengubah jenis data yang dikumpulkan pada node maka Anda mungkin ingin merayapkan data lebih sering, seperti yang dijelaskan dalam prosedur berikut.

> **Important**
>
> AWS Glue membebani Akun AWS Anda berdasarkan tarif per jam, ditagih per detik, untuk perayap (menemukan data) dan pekerjaan ETL (pemrosesan dan pemuatan data). Sebelum Anda mengubah jadwal perayap, lihat halaman Harga AWS Glue.

Untuk mengubah jadwal perayap data inventaris

1. Buka konsol AWS Glue di https://console.aws.amazon.com/glue/.
2. Di panel navigasi, pilih Perayap.
3. Di daftar perayap, pilih pilihan di samping perayap data Inventaris Systems Manager. Nama perayap menggunakan format berikut:

   `AWSSystemsManager-DOC-EXAMPLE-BUCKET-Region-account_ID`
4. Pilih Tindakan, lalu pilih Edit perayap.
5. Di panel navigasi, pilih Jadwal.
6. Di bidang Ekspresi cron, tentukan jadwal baru dengan menggunakan format cron. Untuk informasi lebih lanjut tentang format cron, lihat Jadwal Berbasis Waktu untuk Pekerjaan dan Perayap di Panduan Developer AWS Glue.

> **Important**
>
> Anda dapat menjeda perayap untuk menghentikan pembebanan biaya dari AWS Glue. Jika Anda menjeda perayap, atau jika Anda mengubah frekuensi sehingga data dirayapkan lebih jarang, maka Tampilan Detail Inventaris mungkin menampilkan data yang bukan saat ini.

Mengkueri pengumpulan inventaris dengan menggunakan filter

Setelah mengumpulkan data inventaris, Anda dapat menggunakan kemampuan filter diAWS Systems Manager untuk mengkueri daftar node terkelola yang memenuhi kriteria tertentu.

Untuk mengkueri node berdasarkan filter inventaris

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Inventaris.

   -atau-

   Jika halaman beranda AWS Systems Manager terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Inventaris di panel navigasi.
3. Di Filter berdasarkan grup sumber daya, tag, atau jenis inventaris, pilih kotak filter. Daftar filter yang ditentukan sebelumnya akan ditampilkan.
4. Pilih atribut untuk difilter. Misalnya, pilih **`AWS:Application`**. Jika diminta, pilih atribut sekunder untuk difilter. Misalnya, pilih **`AWS:Application.Name`**.
5. Pilih pembatas dari daftar. Misalnya, pilih Mulailah dengan. Kotak teks ditampilkan dalam filter.
6. Masukkan nilai di kotak teks. Misalnya, masukkan Amazon (SSM Agentbernama AmazonSSM Agent).
7. Tekan Enter. Sistem menampilkan daftar node terkelola yang mencakup nama aplikasi yang dimulai dengan kata Amazon.

> **Note**
>
> Anda dapat menggabungkan beberapa filter untuk menyempurnakan pencarian Anda.

Pengumpulan data inventaris

Setelah mengonfigurasi node terkelola untukAWS Systems Manager Inventaris, Anda dapat melihat jumlah agregat data inventaris. Misalnya, anggap saja Anda mengonfigurasi puluhan atau ratusan node terkelola untuk mengumpulkan jenis`AWS:Application` inventaris. Dengan menggunakan informasi di bagian ini, Anda dapat melihat jumlah pasti dari berapa banyak node yang dikonfigurasi untuk mengumpulkan data ini.

Anda juga dapat melihat detail inventaris tertentu dengan mengumpulkan jenis data. Misalnya, jenis inventaris `AWS:InstanceInformation` mengumpulkan informasi platform sistem operasi dengan jenis data `Platform`. Dengan mengumpulkan data pada jenis`Platform` data, Anda dapat dengan cepat melihat berapa banyak node yang menjalankan Windows, berapa banyak yang menjalankan Linux, dan berapa banyak yang berjalanmacOS.

Prosedur di bagian ini menjelaskan cara melihat jumlah agregat dari data inventaris dengan menggunakan AWS Command Line Interface (AWS CLI). Anda juga dapat melihat jumlah agregat yang dikonfigurasi sebelumnya di konsol AWS Systems Manager pada halaman Inventaris. Dasbor yang dikonfigurasi sebelumnya ini disebut Wawasan Inventaris dan menawarkan remediasi satu klik untuk masalah konfigurasi inventaris Anda.

Perhatikan detail penting mengenai jumlah pengumpulan data inventaris berikut:

- Jika Anda mengakhiri node terkelola yang dikonfigurasi untuk mengumpulkan data inventaris, Systems Manager mempertahankan data inventaris selama 30 hari lalu menghapusnya. Untuk menjalankan node, sistem menghapus data inventaris yang berusia lebih dari 30 hari. Jika Anda harus menyimpan data inventaris lebih dari 30 hari, Anda dapat menggunakan AWS Config untuk mencatat riwayat atau secara berkala mengkueri dan mengunggah data ke bucket Amazon Simple Storage Service (Amazon S3).
- Jika node terlebih dahulu dikonfigurasi untuk melaporkan jenis data inventaris tertentu`AWS:Network`, dan nantinya Anda mengubah konfigurasi agar berhenti mengumpulkan jenis tersebut, jumlah pengumpulan akan tetap menunjukkan`AWS:Network` data sampai node dihentikan dan 30 hari telah berlalu.

Untuk informasi tentang cara mengonfigurasi dan mengumpulkan data inventaris dengan cepat dari semua node di tertentuAkun AWS (dan setiap node di future yang mungkin dibuat di akun tersebut), lihatPengonfigurasian pengumpulan dengan menggunakan konsol.

Topik

- Pengumpulan data inventaris untuk melihat jumlah node yang mengumpulkan jenis data tertentu
- Pengumpulan data inventaris dengan grup untuk melihat node mana yang dikonfigurasi dan yang tidak dikonfigurasi untuk mengumpulkan jenis inventaris

Pengumpulan data inventaris untuk melihat jumlah node yang mengumpulkan jenis data tertentu

Anda dapat menggunakan operasiAWS Systems Manager GetInventoryAPI untuk melihat jumlah agregat dari node yang mengumpulkan satu atau beberapa jenis inventaris dan jenis data. Misalnya, jenis`AWS:InstanceInformation` inventaris memungkinkan Anda untuk melihat agregat sistem operasi dengan menggunakan operasi GetInventory API dengan jenis`AWS:InstanceInformation.PlatformType` data. Berikut adalah contohAWS CLI perintah dan outputnya.

```
aws ssm get-inventory --aggregators "Expression=AWS:InstanceInformation.PlatformType"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "Entities":[
      {
         "Data":{
            "AWS:InstanceInformation":{
               "Content":[
                  {
                     "Count":"7",
                     "PlatformType":"windows"
                  },
                  {
                     "Count":"5",
                     "PlatformType":"linux"
                  }
               ]
            }
         }
      }
   ]
}
```

Memulai

Tentukan jenis inventaris dan jenis data yang Anda ingin lihat jumlahnya. Anda dapat melihat daftar jenis inventaris dan jenis data yang mendukung pengumpulan dengan menjalankan perintah berikut diAWS CLI.

```
aws ssm get-inventory-schema --aggregator
```

Perintah menampilkan daftar JSON dari jenis inventaris dan jenis data yang mendukung pengumpulan. TypeNameBidang menunjukkan jenis inventaris yang didukung. Sedangkan bidang Nama menunjukkan setiap jenis data. Misalnya, dalam daftar berikut, jenis`AWS:Application` inventaris mencakup jenis data untuk`Name` dan`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"
                }
            ]
        }
    ]
}
```

Anda dapat mengumpulkan data untuk jenis inventaris apa pun yang terdaftar dengan membuat perintah yang menggunakan sintaks berikut.

```
aws ssm get-inventory --aggregators "Expression=InventoryType.DataType"
```

Berikut ini adalah beberapa contoh.

Contoh 1

Contoh ini mengumpulkan jumlah peran Windows yang digunakan oleh node Anda.

```
aws ssm get-inventory --aggregators "Expression=AWS:WindowsRole.Name"
```

Contoh 2

Contoh ini mengumpulkan jumlah aplikasi yang diinstal pada node Anda.

```
aws ssm get-inventory --aggregators "Expression=AWS:Application.Name"
```

Penggabungan beberapa agregator

Anda juga dapat menggabungkan beberapa jenis inventaris dan jenis data dalam satu perintah untuk membantu Anda memahami data dengan lebih baik. Berikut ini adalah beberapa contoh.

Contoh 1

Contoh ini mengumpulkan jumlah jenis sistem operasi yang digunakan oleh node Anda. Ia juga menampilkan nama tertentu dari sistem operasi.

```
aws ssm get-inventory --aggregators '[{"Expression":
 "AWS:InstanceInformation.PlatformType", "Aggregators":[{"Expression":
 "AWS:InstanceInformation.PlatformName"}]}]'
```

Contoh 2

Contoh ini mengumpulkan jumlah aplikasi yang berjalan pada node Anda dan versi tertentu dari setiap aplikasi.

```
aws ssm get-inventory --aggregators '[{"Expression": "AWS:Application.Name",
 "Aggregators":[{"Expression": "AWS:Application.Version"}]}]'
```

Jika mau, Anda dapat membuat ekspresi pengumpulan dengan satu atau beberapa jenis inventaris dan jenis data dalam file JSON dan memanggil file dari AWS CLI. JSON dalam file harus menggunakan sintaks berikut.

```
[
       {
           "Expression": "string",
           "Aggregators": [
               {
                  "Expression": "string"
               }
           ]
       }
]
```

Anda harus menyimpan file dengan ekstensi file. json.

Berikut adalah contoh yang menggunakan beberapa jenis inventaris dan jenis data.

```
[
       {
           "Expression": "AWS:Application.Name",
           "Aggregators": [
               {
                   "Expression": "AWS:Application.Version",
```

```
                    "Aggregators": [
                      {
                      "Expression": "AWS:InstanceInformation.PlatformType"
                      }
                    ]
                }
            ]
        }
]
```

Gunakan perintah berikut untuk memanggil file dari AWS CLI.

```
aws ssm get-inventory --aggregators file://file_name.json
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
{"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"}
 ]
}
```

Pengumpulan data inventaris dengan grup untuk melihat node mana yang dikonfigurasi dan yang tidak dikonfigurasi untuk mengumpulkan jenis inventaris

Grup di Inventaris Systems Manager memungkinkan Anda dengan cepat melihat jumlah node terkelola yang dikonfigurasi dan yang tidak dikonfigurasi untuk mengumpulkan satu atau beberapa jenis inventaris. Dengan grup, Anda menentukan satu atau beberapa jenis inventaris dan filter yang menggunakan operator `exists`.

Misalnya, anggap saja Anda memiliki empat node terkelola yang dikonfigurasi untuk mengumpulkan jenis inventaris berikut:

- Simpul 1:`AWS:Application`
- Simpul 2:`AWS:File`
- Simpul 3:`AWS:Application`,`AWS:File`
- Simpul 4:`AWS:Network`

Anda dapat menjalankan perintah berikut dariAWS CLI untuk melihat berapa banyak node dikonfigurasi untuk mengumpulkan keduaAWS:Application danAWS:File inventory jenis. Tanggapan juga menampilkan jumlah dari berapa banyak node yang tidak dikonfigurasi untuk mengumpulkan kedua jenis inventaris ini.

```
aws ssm get-inventory --aggregators
 'Groups=[{Name=ApplicationAndFile,Filters=[{Key=TypeName,Values=[AWS:Application],Type=Exists}
{Key=TypeName,Values=[AWS:File],Type=Exists}]}]'
```

Tanggapan perintah menunjukkan bahwa hanya ada satu node terkelola yang dikonfigurasi untuk mengumpulkan jenisAWS:File inventarisAWS:Application dan inventaris.

```
{
```

```
   "Entities":[
      {
         "Data":{
            "ApplicationAndFile":{
               "Content":[
                  {
                     "notMatchingCount":"3"
                  },
                  {
                     "matchingCount":"1"
                  }
               ]
            }
         }
      }
   ]
}
```

> **Note**
>
> Grup tidak menampilkan jumlah jenis data. Juga, Anda tidak dapat menguraikan hasil untuk melihat ID node yang dikonfigurasi atau yang tidak dikonfigurasi untuk mengumpulkan jenis inventaris.

Jika mau, Anda dapat membuat ekspresi pengumpulan dengan satu atau beberapa jenis inventaris dalam file JSON dan memanggil file dari AWS CLI. JSON dalam file harus menggunakan sintaks berikut:

```
{
   "Aggregators":[
      {
         "Groups":[
            {
               "Name":"Name",
               "Filters":[
                  {
                     "Key":"TypeName",
                     "Values":[
                        "Inventory_type"
                     ],
                     "Type":"Exists"
```

```
                },
                {
                   "Key":"TypeName",
                   "Values":[
                      "Inventory_type"
                   ],
                   "Type":"Exists"
                }
             ]
          }
       ]
    }
 ]
}
```

Anda harus menyimpan file dengan ekstensi file. json.

Gunakan perintah berikut untuk memanggil file dari AWS CLI.

```
aws ssm get-inventory --cli-input-json file://file_name.json
```

Contoh tambahan

Contoh berikut menunjukkan cara mengumpulkan data inventaris untuk melihat node terkelola yang dikonfigurasi dan yang tidak dikonfigurasi untuk mengumpulkan jenis inventaris tertentu. Contoh-contoh ini menggunakan AWS CLI. Setiap contoh mencakup perintah lengkap dengan filter yang dapat Anda jalankan dari baris perintah dan sampel file input.json jika Anda lebih memilih untuk memasukkan informasi dalam file.

Contoh 1

Contoh ini mengumpulkan jumlah node yang dikonfigurasi dan yang tidak dikonfigurasi untuk mengumpulkan jenis inventaris`AWS:Application` atau jenis`AWS:File` inventaris.

Jalankan perintah berikut dari AWS CLI.

```
aws ssm get-inventory --aggregators
 'Groups=[{Name=ApplicationORFile,Filters=[{Key=TypeName,Values=[AWS:Application,
 AWS:File],Type=Exists}]}]'
```

Jika Anda lebih memilih untuk menggunakan file, salin dan tempelkan sampel berikut ke dalam file dan simpan sebagai input.json.

```
{
   "Aggregators":[
      {
         "Groups":[
            {
               "Name":"ApplicationORFile",
               "Filters":[
                  {
                     "Key":"TypeName",
                     "Values":[
                        "AWS:Application",
                        "AWS:File"
                     ],
                     "Type":"Exists"
                  }
               ]
            }
         ]
      }
   ]
}
```

Jalankan perintah berikut dari AWS CLI.

```
aws ssm get-inventory --cli-input-json file://input.json
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
{
   "Entities":[
      {
         "Data":{
            "ApplicationORFile":{
               "Content":[
                  {
                     "notMatchingCount":"1"
                  },
                  {
                     "matchingCount":"3"
                  }
               ]
            }
```

```
         }
      }
   ]
}
```

Contoh 2

Contoh ini mengumpulkan jumlah node yang dikonfigurasi dan yang tidak dikonfigurasi untuk mengumpulkan jenis`AWS:Network` inventaris`AWS:Application``AWS:File`, dan inventaris.

Jalankan perintah berikut dari 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}]}]'
```

Jika Anda lebih memilih untuk menggunakan file, salin dan tempelkan sampel berikut ke dalam file dan simpan sebagai 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"
                  }
               ]
            }
         ]
      }
   ]
}
```

Jalankan perintah berikut dari AWS CLI.

```
aws ssm get-inventory --cli-input-json file://input.json
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
{
   "Entities":[
      {
         "Data":{
            "Application":{
               "Content":[
                  {
                     "notMatchingCount":"2"
                  },
                  {
                     "matchingCount":"2"
                  }
               ]
            },
```

```
            "File":{
               "Content":[
                  {
                     "notMatchingCount":"2"
                  },
                  {
                     "matchingCount":"2"
                  }
               ]
            },
            "Network":{
               "Content":[
                  {
                     "notMatchingCount":"3"
                  },
                  {
                     "matchingCount":"1"
                  }
               ]
            }
         }
      }
   ]
}
```

Menggunakan inventaris kustom

Anda dapat menetapkan metadata apa pun yang Anda inginkan ke node Anda dengan membuat AWS Systems Manager inventaris kustom Inventaris. Misalnya, katakanlah Anda mengelola sejumlah besar server di rak di pusat data Anda, dan server ini telah dikonfigurasi sebagai node yang dikelola Systems Manager. Saat ini, Anda menyimpan informasi tentang lokasi rak server di lembar kerja. Dengan inventaris khusus, Anda dapat menentukan lokasi rak setiap node sebagai metadata pada node. Ketika Anda mengumpulkan inventaris dengan menggunakan Systems Manager, metadata dikumpulkan dengan metadata inventaris lainnya. Anda kemudian dapat mentransfer semua metadata inventaris ke bucket Amazon S3 sentral dengan menggunakan sinkronisasi data sumber daya dan mengkueri data.

> **Note**
>
> Systems Manager mendukung maksimum 20 jenis inventaris kustom per Akun AWS.

Untuk menetapkan inventaris kustom ke node, Anda dapat menggunakan operasi Systems Manager PutInventoryAPI, seperti yang dijelaskan dalamPanduan: Menetapkan metadata inventaris kustom ke node terkelola. Atau, Anda dapat membuat file JSON inventaris khusus dan mengunggahnya ke node. Bagian ini menjelaskan cara membuat file JSON.

Contoh file JSON dengan inventaris kustom berikut menentukan informasi rak tentang server on-premise. Contoh ini menentukan salah satu jenis dari data inventaris kustom (`"TypeName": "Custom:RackInformation"`), dengan beberapa entri di bawah `Content` yang menjelaskan data.

```
{
    "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"
      }
 }
```

Anda juga dapat menetapkan entri yang berbeda di bagian `Content`, seperti yang ditunjukkan dalam contoh berikut.

```
{
"SchemaVersion": "1.0",
"TypeName": "Custom:PuppetModuleInfo",
    "Content": [{
        "Name": "puppetlabs/aws",
        "Version": "1.0"
      },
      {
        "Name": "puppetlabs/dsc",
        "Version": "2.0"
      }
    ]
}
```

Skema JSON untuk inventaris kustom membutuhkan`SchemaVersion`,`TypeName`, dan `Content` bagian, tetapi Anda dapat menentukan informasi di bagian tersebut.

```
{
    "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"
      }
 }
```

Nilai `TypeName` dibatasi hingga 100 karakter. Juga, `TypeName` nilainya harus dimulai dengan kata `Custom` yang dikapitalisasi. Sebagai contoh, `Custom:PuppetModuleInfo`. Oleh karena itu, contoh-contoh berikut akan menghasilkan pengecualian:`CUSTOM:PuppetModuleInfo`,`custom:PuppetModuleInfo`.

`Content`Bagian ini mencakup atribut dan *`data`*. Item ini tidak sensitif terhadap huruf besar atau kecil. Namun, jika Anda mendefinisikan atribut (misalnya: "`Vendor`“: “DELL”), maka Anda harus secara konsisten mereferensikan atribut ini dalam file inventaris khusus Anda. Jika Anda menentukan "`Vendor`“: “DELL” (menggunakan huruf besar “V” dalam`vendor`) dalam satu file, dan kemudian Anda menentukan "`vendor`“: “DELL” (menggunakan huruf kecil “v” in`vendor`) di file lain, sistem mengembalikan kesalahan.

> **Note**
>
> Anda harus menyimpan file dengan `.json` ekstensi dan inventaris yang Anda tentukan harus hanya terdiri dari nilai string.

Setelah Anda membuat file, Anda harus menyimpannya di node. Tabel berikut menunjukkan lokasi di mana file JSON inventaris kustom harus disimpan pada node.

Sistem operasi	Jalur
Linux	*/var/lib/amazon/ssm/node-id/inventaris/kustom*
macOS	`/opt/aws/ssm/data/` *`node-id`*`/inventory/custom`

Sistem operasi	Jalur
Windows	%SystemDrive%\\ AmazonProgramData\ SSM\\ *node-idInstanceData\ inventaris* \ kustom

Untuk contoh cara menggunakan inventaris kustom, lihat Mendapatkan Manfaat Disk dari Armada Anda Menggunakan Jenis Inventaris Kustom Systems Manager EC2.

Penghapusan inventaris kustom

Anda dapat menggunakan operasi DeleteInventoryAPI untuk menghapus jenis inventaris khusus dan data yang terkait dengan jenis tersebut. Anda memanggil perintah delete-inventory dengan menggunakan AWS Command Line Interface (AWS CLI) untuk menghapus semua data untuk jenis inventaris. Anda memanggil perintah delete-inventory dengan `SchemaDeleteOption` untuk menghapus jenis inventaris kustom.

> **Note**
>
> Jenis inventaris juga disebut skema inventaris.

Parameter `SchemaDeleteOption` mencakup pilihan berikut:

- DeleteSchema: Opsi ini menghapus jenis kustom yang ditentukan dan semua data yang terkait dengannya. Anda dapat membuat ulang skema nantinya, jika Anda menginginkannya.
- DisableSchema: Jika Anda memilih opsi ini, sistem mematikan versi saat ini, menghapus semua data untuk itu, dan mengabaikan semua data baru jika versi kurang dari atau sama dengan versi yang dimatikan. Anda dapat mengizinkan jenis inventaris ini lagi dengan memanggil PutInventorytindakan untuk versi yang lebih besar dari versi yang dimatikan.

Untuk menghapus atau menonaktifkan inventaris kustom dengan menggunakan AWS CLI

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI.

2. Jalankan perintah berikut untuk menggunakan pilihan `dry-run` untuk melihat data mana yang akan dihapus dari sistem. Perintah ini tidak menghapus data apa pun.

```
aws ssm delete-inventory --type-name "Custom:custom_type_name" --dry-run
```

Sistem mengembalikan informasi seperti berikut.

```
{
   "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"
}
```

Untuk informasi tentang cara memahami ringkasan inventaris penghapusan, lihat Memahami ringkasan inventaris penghapusan.

3. Jalankan perintah berikut untuk menghapus semua data untuk jenis inventaris kustom.

```
aws ssm delete-inventory --type-name "Custom:custom_type_name"
```

> **Note**
>
> Output dari perintah ini tidak menunjukkan progres penghapusan. Untuk alasan ini, TotalCount dan Hitungan Sisa selalu sama karena sistem belum menghapus apa pun. Anda dapat menggunakan describe-inventory-deletions perintah untuk menunjukkan kemajuan penghapusan, seperti yang dijelaskan nanti dalam topik ini.

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "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"
}
```

Sistem menghapus semua data untuk jenis inventaris kustom tertentu dari layanan Inventaris Systems Manager.

4. Jalankan perintah berikut. Perintah melakukan tindakan berikut untuk versi saat ini dari jenis inventaris: menonaktifkan versi saat ini, menghapus semua data untuknya, dan mengabaikan semua data baru jika versinya kurang dari atau setara dengan versi yang dinonaktifkan.

```
aws ssm delete-inventory --type-name "Custom:custom_type_name" --schema-delete-
option "DisableSchema"
```

Sistem mengembalikan informasi seperti berikut.

```
{
   "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"
}
```

Anda dapat melihat jenis inventaris yang dinonaktifkan dengan menggunakan perintah berikut.

```
aws ssm get-inventory-schema --type-name Custom:custom_type_name
```

5. Jalankan perintah berikut untuk menghapus jenis inventaris.

```
aws ssm delete-inventory --type-name "Custom:custom_type_name" --schema-delete-
option "DeleteSchema"
```

Sistem menghapus skema dan semua data inventaris untuk jenis kustom tertentu.

Sistem mengembalikan informasi seperti berikut.

```
{
   "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"
}
```

Melihat status penghapusan

Anda dapat memeriksa status operasi hapus dengan menggunakan `describe-inventory-deletions` AWS CLI perintah. Anda dapat menentukan ID penghapusan untuk melihat status operasi penghapusan tertentu. Atau, Anda dapat menghilangkan ID penghapusan untuk melihat daftar semua penghapusan yang dijalankan dalam 30 hari terakhir.

1. Jalankan perintah berikut untuk melihat status operasi penghapusan. Sistem menampilkan ID penghapusan dalam ringkasan delete-inventory.

   ```
   aws ssm describe-inventory-deletions --deletion-id system_generated_deletion_ID
   ```

   Sistem menampilkan status terbaru. Operasi penghapusan mungkin belum selesai. Sistem mengembalikan informasi seperti berikut.

   ```
   {"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"}
   ```

```
  ]
}
```

Jika operasi penghapusan berhasil, `LastStatusMessage` menyatakan: Penghapusan berhasil.

```
{"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. Jalankan perintah berikut untuk melihat daftar dari semua penghapusan yang dijalankan dalam 30 hari terakhir.

```
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"
```

Memahami ringkasan inventaris penghapusan

Untuk membantu Anda memahami konten ringkasan inventaris penghapusan, pertimbangkan contoh berikut. Seorang pengguna menetapkan Custom: RackSpace inventaris ke tiga node. Item inventaris 1 dan 2 menggunakan tipe khusus versi 1.0 (" SchemaVersion ":"1.0"). Inventaris item 3 menggunakan tipe kustom versi 2.0 (" SchemaVersion ":"2.0").

RackSpace inventaris kustom 1

```
{
   "CaptureTime":"2018-02-19T10:48:55Z",
   "TypeName":"CustomType:RackSpace",
   "InstanceId":"i-1234567890",
   "SchemaVersion":"1.0"   "Content":[
      {
         content of custom type omitted
      }
   ]
}
```

RackSpace inventaris kustom 2

```
{
   "CaptureTime":"2018-02-19T10:48:55Z",
   "TypeName":"CustomType:RackSpace",
   "InstanceId":"i-1234567891",
   "SchemaVersion":"1.0"   "Content":[
      {
         content of custom type omitted
      }
   ]
}
```

RackSpace inventaris kustom 3

```
{
   "CaptureTime":"2018-02-19T10:48:55Z",
   "TypeName":"CustomType:RackSpace",
   "InstanceId":"i-1234567892",
   "SchemaVersion":"2.0"   "Content":[
      {
```

```
         content of custom type omitted
      }
   ]
}
```

Pengguna menjalankan perintah berikut untuk mem-pratinjau data yang akan dihapus.

```
aws ssm delete-inventory --type-name "Custom:RackSpace" --dry-run
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "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"
}
```

Pengguna menjalankan perintah berikut untuk menghapus Custom: RackSpace inventory.

> **Note**
>
> Output dari perintah ini tidak menunjukkan progres penghapusan. Untuk alasan ini, `TotalCount` dan `RemainingCount` selalu sama karena sistem belum menghapus apa pun. Anda dapat menggunakan `describe-inventory-deletions` perintah untuk menunjukkan kemajuan penghapusan.

```
aws ssm delete-inventory --type-name "Custom:RackSpace"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "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"
}
```

Melihat tindakan penghapusan inventaris di EventBridge

Anda dapat mengonfigurasi Amazon EventBridge untuk membuat acara kapan saja pengguna menghapus inventaris khusus. EventBridge menawarkan tiga jenis acara untuk operasi penghapusan inventaris kustom:

- Hapus tindakan untuk sebuah instance: Jika inventaris khusus untuk node terkelola tertentu berhasil dihapus atau tidak.
- Menghapus ringkasan tindakan: Ringkasan tindakan penghapusan.
- Peringatan untuk jenis inventaris kustom yang dimatikan: Peristiwa peringatan jika pengguna memanggil operasi PutInventoryAPI untuk versi tipe inventaris khusus yang sebelumnya dimatikan.

Berikut adalah contoh dari setiap acara.

Hapus tindakan untuk sebuah instance

```
{
   "version":"0",
   "id":"998c9cde-56c0-b38b-707f-0411b3ff9d11",
   "detail-type":"Inventory Resource State Change",
   "source":"aws.ssm",
   "account":"478678815555",
   "time":"2018-05-24T22:24:34Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:ssm:us-east-1:478678815555:managed-instance/i-0a5feb270fc3f0b97"
   ],
   "detail":{
      "action-status":"succeeded",
      "action":"delete",
      "resource-type":"managed-instance",
      "resource-id":"i-0a5feb270fc3f0b97",
      "action-reason":"",
      "type-name":"Custom:MyInfo"
   }
}
```

Hapus ringkasan tindakan

```
{
   "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"
   }
}
```

Peringatan untuk jenis inventaris khusus yang dimatikan

```
{
   "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"
   }
}
```

Gunakan prosedur berikut untuk membuat EventBridge aturan untuk operasi penghapusan inventaris kustom. Prosedur ini menunjukkan cara untuk membuat aturan yang mengirimkan notifikasi untuk operasi penghapusan inventaris kustom ke topik Amazon SNS. Sebelum memulai, pastikan Anda

memiliki topik Amazon SNS, atau buat yang baru. Untuk informasi lebih lanjut, lihat Memulai di Panduan Developer Amazon Simple Notification Service.

Mengkonfigurasi EventBridge untuk menghapus operasi inventaris

1. Buka EventBridge konsol Amazon di https://console.aws.amazon.com/events/.
2. Di panel navigasi, pilih Aturan.
3. Pilih Buat aturan.
4. Masukkan nama dan deskripsi untuk aturan.

   Aturan tidak boleh memiliki nama yang sama dengan aturan lain di Wilayah yang sama dan di bus kejadian yang sama.
5. Untuk bus acara, pilih bus acara yang ingin Anda kaitkan dengan aturan ini. Jika Anda ingin aturan ini merespons peristiwa pencocokan yang berasal dari Anda sendiriAkun AWS, pilih default. Ketika Layanan AWS di akun Anda memancarkan acara, itu selalu masuk ke bus acara default akun Anda.
6. Untuk jenis Aturan, pilih Aturan dengan pola acara.
7. Pilih Selanjutnya.
8. Untuk sumber Acara, pilih AWSacara atau acara EventBridge mitra.
9. Di bagian Pola acara, pilih Formulir pola acara.
10. Untuk sumber acara, pilih AWSlayanan.
11. Untuk AWSlayanan, pilih Systems Manager.
12. Untuk Jenis peristiwa, pilih Inventaris.
13. Untuk jenis detail spesifik, pilih Perubahan Status Sumber Daya Inventaris.
14. Pilih Selanjutnya.
15. Untuk jenis Target, pilih AWSlayanan.
16. Untuk Pilih target, pilih topik SNS, lalu untuk Topik, pilih topik Anda.
17. Di bagian Pengaturan tambahan, untuk Konfigurasi input target, verifikasi bahwa Peristiwa yang cocok dipilih.
18. Pilih Selanjutnya.
19. (Opsional) Masukkan satu atau lebih tanda untuk aturan. Untuk informasi selengkapnya, lihat Menandai EventBridge Sumber Daya Amazon Anda di Panduan EventBridge Pengguna Amazon.

20. Pilih Selanjutnya.
21. Tinjau detail aturan dan pilih Buat aturan.

Melihat riwayat inventaris dan pelacakan perubahan

Anda dapat melihat riwayatAWS Systems Manager Inventaris dan pelacakan perubahan untuk semua node terkelola Anda dengan menggunakan AWS Config. AWS Configmenyediakan tampilan rinci dari konfigurasiAWS sumber daya di AndaAkun AWS. Ini mencakup bagaimana sumber daya terkait satu sama lain dan bagaimana sumber daya tersebut dikonfigurasi di masa lalu sehingga Anda dapat melihat bagaimana konfigurasi dan hubungan berubah dari waktu ke waktu. Untuk melihat riwayat inventaris dan pelacakan perubahan, Anda harus mengaktifkan sumber daya berikut diAWS Config:

- SSM:ManagedInstanceInventory
- SSM:PatchCompliance
- SSM:AssociationCompliance
- SSM:FileData

> **Note**
>
> Perhatikan detail penting tentang riwayat Inventaris dan pelacakan perubahan berikut:
>
> - Jika Anda menggunakan AWS Config untuk melacak perubahan dalam sistem Anda, Anda harus mengonfigurasi Inventaris Systems Manager untuk mengumpulkan metadata `AWS:File` agar Anda dapat melihat perubahan file di AWS Config (`SSM:FileData`). Jika tidak, maka AWS Config tidak melacak perubahan file pada sistem Anda.
> - Dengan mengaktifkan SSM:PatchCompliance dan SSM:AssociationCompliance, Anda dapat melihatPatch Manager patching Systems Manager dan riwayat kepatuhanState Manager asosiasi Systems Manager dan pelacakan perubahan. Untuk informasi lebih lanjut tentang pengelolaan kepatuhan untuk sumber daya ini, lihat Bekerja dengan Kepatuhan.

Prosedur berikut menjelaskan cara mengaktifkan riwayat inventaris dan pencatatan pelacakan perubahan di AWS Config dengan menggunakan AWS Command Line Interface (AWS CLI). Untuk informasi lebih lanjut tentang cara memilih dan mengonfigurasi sumber daya ini di AWS Config, lihat

Memilih Catatan AWS Config Sumber Daya yang Mana di Panduan Developer AWS Config. Untuk informasi tentang harga AWS Config, lihat Harga.

Sebelum Anda memulai

AWS Config membutuhkan izin AWS Identity and Access Management (IAM) untuk mendapatkan detail konfigurasi tentang sumber daya Systems Manager. Dalam prosedur berikut, Anda harus menentukan Amazon Resource Name (ARN) untuk IAM role yang memberikan izin AWS Config untuk sumber daya Systems Manager. Anda dapat melampirkan kebijakan terkelola `AWS_ConfigRole` ke IAM role yang Anda tetapkan ke AWS Config. Untuk informasi selengkapnya tentang peran ini, lihat kebijakanAWS terkelola:AWS _ConfigRole di PanduanAWS Config Pengembang. Untuk informasi tentang cara membuat IAM role dan menetapkan kebijakan`AWS_ConfigRole` terkelola untuk peran tersebut, lihat Pembuatan untuk mendelegasikan izin ke Panduan Pengguna IAM.Layanan AWS

Untuk mengaktifkan riwayat inventaris dan catatan pelacakan perubahan di AWS Config

1. Instal dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaruAWS CLI.

2. Salin dan tempelkan sampel JSON berikut ke dalam file teks sederhana dan simpan sebagai recordingGroup.json.

```
{
   "allSupported":false,
   "includeGlobalResourceTypes":false,
   "resourceTypes":[
      "AWS::SSM::AssociationCompliance",
      "AWS::SSM::PatchCompliance",
      "AWS::SSM::ManagedInstanceInventory",
      "AWS::SSM::FileData"
   ]
}
```

3. Jalankan perintah berikut untuk memuat file recordingGroup.json ke dalam AWS Config.

```
aws configservice put-configuration-recorder --configuration-recorder
 name=myRecorder,roleARN=arn:aws:iam::123456789012:role/myConfigRole --recording-
group file://recordingGroup.json
```

4. Jalankan perintah berikut untuk memulai pencatatan riwayat inventaris dan pelacakan perubahan.

```
aws configservice start-configuration-recorder --configuration-recorder-
name myRecorder
```

Setelah Anda mengonfigurasi riwayat dan pelacakan perubahan, Anda dapat menguraikan riwayat untuk node terkelola tertentu dengan memilih AWS Configtombol di konsol Systems Manager. Anda dapat mengakses tombol AWS Config dari halaman Instans Terkelola atau halaman Inventaris. Tergantung dari ukuran monitor, Anda mungkin harus menggulir ke sisi kanan halaman untuk melihat tombol.

Menghentikan pengumpulan data dan menghapus data inventaris

Jika Anda tidak lagi ingin menggunakanAWS Systems Manager Inventory untuk melihat metadata tentangAWS sumber daya Anda, Anda dapat menghentikan pengumpulan data dan menghapus data yang telah dikumpulkan. Bagian ini mencakup informasi berikut.

Topik

- Menghentikan pengumpulan data data yang berhenti
- Menghapus sinkrondata sumber daya data sumber daya data sumber daya

Menghentikan pengumpulan data data yang berhenti

Ketika Anda awalnya mengkonfigurasi Systems Manager untuk mengumpulkan data inventaris, sistem membuatState Manager asosiasi yang mendefinisikan jadwal dan sumber daya untuk mengumpulkan metadata. Anda dapat menghentikan pengumpulan data dengan menghapusState Manager asosiasi apa pun yang menggunakan`AWS-GatherSoftwareInventory` dokumen.

Untuk menghapus keterkaitan Inventaris Inventaris

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih State Manager.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu () untuk membuka panel navigasi, lalu pilih ikon menu

(☰)
untuk membuka panel navigasi, lalu pilih ikon menu () untuk membuka panel navigasi, lalu pilih ikon menu () State Manager.

3. Pilih asosiasi yang menggunakan`AWS-GatherSoftwareInventory` dokumen dan kemudian pilih Hapus.
4. Ulangi langkah ketiga untuk setiap asosiasi yang tersisa yang menggunakan`AWS-GatherSoftwareInventory` dokumen.

Menghapus sinkrondata sumber daya data sumber daya data sumber daya

Jika Anda tidak lagi ingin menggunakanAWS Systems Manager Inventory untuk melihat metadata tentangAWS sumber daya Anda, sebaiknya hapus sinkronisasi data sumber daya yang digunakan untuk pengumpulan data inventaris.

Untuk menghapus sinkrondata sumber daya data sumber daya data sumber daya

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Inventaris.

   -atau-

   Jika halaman beranda AWS Systems Manager terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Inventaris di panel navigasi.
3. Pilih Sinkronisasi Data Sumber Daya.
4. Pilih sinkronisasi di daftar.

   > **Important**
   >
   > Pastikan Anda memilih sinkronisasi yang digunakan untuk Inventory. Systems Manager mendukung sinkronisasi data sumber daya untuk beberapa kemampuan. Jika Anda memilih sinkronisasi yang salah, Anda dapat mengganggu agregasi data untuk Systems Manager Explorer atau Kepatuhan Systems Manager.

5. Pilih Hapus
6. Ulangi langkah-langkah ini untuk sinkronisasi data sumber daya yang tersisa yang ingin Anda hapus.

7. Hapus bucket Amazon Simple Storage Service (Amazon S3) tempat data disimpan. Untuk informasi tentang menghapus bucket Amazon S3, lihat Menghapus bucket Amazon S3.

Panduan Inventaris Systems Manager

Gunakan panduan berikut untuk mengumpulkan dan mengelola data inventaris dengan menggunakan Inventaris AWS Systems Manager. Kami merekomendasikan Anda untuk Anda terlebih dahulu panduan Anda untuk Anda kelola Node di lingkungan pengujian.

Sebelum Anda memulai

Sebelum Anda memulai panduan panduan berikut:

- PerbaruiAWS Systems ManagerSSM Agent node yang ingin Anda inventarisasi. Dengan menjalankan versi terbaruSSM Agent, Anda dapat mengumpulkan metadata untuk semua jenis inventaris yang didukung. Untuk informasi tentang cara memperbaruiSSM Agent dengan menggunakanState Manager, lihatWalkthrough: Perbarui secara otomatis (SSM AgentCLI).
- Verifikasi bahwa Anda telah menyelesaikan persyaratan pengaturan instans Amazon Elastic Compute Cloud (Amazon EC2) Anda dan mesin non-EC2 di lingkungan hibrid dan multicloud. Untuk informasi, lihat Menyiapkan AWS Systems Manager.
- (Opsional) Buat file JSON untuk mengumpulkan inventaris kustom. Untuk informasi selengkapnya, lihat Menggunakan inventaris kustom.

Konten

Panduan: Menetapkan metadata inventaris kustom ke node terkelola

Prosedur berikut memandu Anda menjalani proses penggunaan operasiAWS Systems Manager PutInventoryAPI untuk menetapkan metadata inventaris kustom ke node terkelola. Contoh ini menetapkan informasi lokasi rak ke simpul. Untuk informasi lebih lanjut tentang inventaris kustom, lihat Menggunakan inventaris kustom.

Untuk menetapkan metadata inventaris kustom ke

1. Instal dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaru dariAWS CLI.

2. Jalankan perintah berikut untuk menetapkan informasi lokasi rak ke simpul.

   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"}]'
   ```

   Jendela

   ```
   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. Jalankan perintah berikut untuk melihat entri inventaris kustom untuk node ini.

   ```
   aws ssm list-inventory-entries --instance-id ID --type-name "Custom:RackInfo"
   ```

   Sistem menanggapi dengan informasi seperti berikut.

   ```
   {
       "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. Jalankan perintah berikut untuk melihat skema inventaris kustom.

```
aws ssm get-inventory-schema --type-name Custom:RackInfo
```

Sistem menanggapi dengan informasi seperti berikut.

```
{
    "Schemas": [
        {
            "TypeName": "Custom:RackInfo",
            "Version": "1.0",
            "Attributes": [
                {
                    "DataType": "STRING",
                    "Name": "RackLocation"
                }
            ]
        }
    ]
}
```

Panduan: Mengonfigurasi node terkelola untuk Inventaris dengan menggunakan CLI

Prosedur berikut memandu Anda menjalani proses pengonfigurasikanAWS Systems Manager Inventaris untuk mengumpulkan metadata dari node terkelola Anda. Ketika Anda mengonfigurasi pengumpulan inventaris, Anda memulai dengan membuatState Manager asosiasi Systems Manager. Systems Manager mengumpulkan data inventaris saat asosiasi dijalankan. Jika Anda tidak membuat asosiasi terlebih dahulu, dan mencoba untuk memintaaws:softwareInventory plugin dengan menggunakan, misalnya, Run.Run Command, sistem menampilkan kesalahan berikut:

`The aws:softwareInventory plugin can only be invoked via ssm-associate.`

> **Note**
>
> Node hanya dapat mengonfigurasi satu asosiasi inventaris dalam satu waktu. Jika Anda mengonfigurasi node dengan dua atau beberapa asosiasi inventaris, asosiasi tidak berjalan dan tidak ada data inventaris yang dikumpulkan.

Mengonfigurasi dengan cepat semua node terkelola Anda untuk Inventaris (CLI)

Anda dapat dengan cepat mengonfigurasi semua node terkelola di AndaAkun AWS dan di Wilayah saat ini untuk mengumpulkan data inventaris. Hal ini disebut pembuatan asosiasi inventaris global. Untuk membuat asosiasi inventaris global dengan menggunakan AWS CLI, gunakan pilihan wildcard untuk nilai `instanceIds`, seperti yang ditunjukkan dalam prosedur berikut.

Untuk mengonfigurasi inventaris untuk semua node terkelola di AndaAkun AWS dan di Wilayah saat ini (CLI)

1. InstalAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaruAWS CLI.

2. Jalankan perintah berikut.

   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**
>
> Perintah ini tidak mengizinkan Inventaris untuk mengumpulkan metadata untuk Registri Windows atau file. Untuk menginventarisasi jenis data ini, gunakan prosedur berikutnya.

Secara manual mengonfigurasi Inventaris pada node terkelola Anda (CLI)

Gunakan prosedur berikut untuk secara manual mengonfigurasiAWS Systems Manager Inventaris pada node terkelola Anda dengan menggunakan ID atau tanda node.

Untuk secara manual mengonfigurasi node terkelola untuk inventaris (CLI)

1. InstalAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaruAWS CLI.

2. Jalankan perintah berikut untuk membuatState Manager asosiasi yang menjalankan Inventaris Systems Manager pada node. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri. Perintah ini mengonfigurasi layanan untuk berjalan setiap enam jam dan untuk mengumpulkan konfigurasi jaringan, Pembaruan Windows, dan metadata aplikasi dari node.

   Linux & macOS

   ```
   aws ssm create-association \
   --name "AWS-GatherSoftwareInventory" \
   --targets "Key=instanceids,Values=an_instance_ID" \
   --schedule-expression "rate(240 minutes)" \
   --output-location "{ \"S3Location\": { \"OutputS3Region\": \"region_ID,
    for example us-east-2\", \"OutputS3BucketName\": \"DOC-EXAMPLE-BUCKET\",
    \"OutputS3KeyPrefix\": \"Test\" } }" \
   --parameters "networkConfig=Enabled,windowsUpdates=Enabled,applications=Enabled"
   ```

   Windows

   ```
   aws ssm create-association ^
   --name "AWS-GatherSoftwareInventory" ^
   --targets "Key=instanceids,Values=an_instance_ID" ^
   --schedule-expression "rate(240 minutes)" ^
   --output-location "{ \"S3Location\": { \"OutputS3Region\": \"region_ID,
    for example us-east-2\", \"OutputS3BucketName\": \"DOC-EXAMPLE-BUCKET\",
    \"OutputS3KeyPrefix\": \"Test\" } }" ^
   --parameters "networkConfig=Enabled,windowsUpdates=Enabled,applications=Enabled"
   ```

   Sistem menanggapi dengan informasi seperti berikut.

```
{
    "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"
            }
        ]
    }
}
```

Anda dapat menargetkan grup node yang besar dengan menggunakan`Targets` parameter dengan tanda EC2. Lihat contoh berikut.

Linux & macOS

```
aws ssm create-association \
--name "AWS-GatherSoftwareInventory" \
--targets "Key=tag:Environment,Values=Production" \
--schedule-expression "rate(240 minutes)" \
--output-location "{ \"S3Location\": { \"OutputS3Region\": \"us-east-2\",
 \"OutputS3BucketName\": \"DOC-EXAMPLE-BUCKET\", \"OutputS3KeyPrefix\": \"Test
\" } }" \
--parameters "networkConfig=Enabled,windowsUpdates=Enabled,applications=Enabled"
```

Windows

```
aws ssm create-association ^
--name "AWS-GatherSoftwareInventory" ^
--targets "Key=tag:Environment,Values=Production" ^
--schedule-expression "rate(240 minutes)" ^
--output-location "{ \"S3Location\": { \"OutputS3Region\": \"us-east-2\",
 \"OutputS3BucketName\": \"DOC-EXAMPLE-BUCKET\", \"OutputS3KeyPrefix\": \"Test
\" } }" ^
--parameters "networkConfig=Enabled,windowsUpdates=Enabled,applications=Enabled"
```

Anda juga dapat menginventarisasi kunci Registri padaWindows Server node dengan menggunakan jenis`windowsRegistry` inventaris dengan ekspresi.`files` Untuk informasi lebih lanjut tentang jenis inventaris ini, lihat Menggunakan file dan inventaris registri 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. Jalankan perintah berikut untuk melihat status asosiasi.

```
aws ssm describe-instance-associations-status --instance-id an_instance_ID
```

Sistem menanggapi dengan informasi seperti berikut.

```
{
"InstanceAssociationStatusInfos": [
         {
            "Status": "Pending",
            "DetailedStatus": "Associated",
            "Name": "reInvent2016PolicyDocumentTest",
            "InstanceId": "i-1a2b3c4d5e6f7g",
            "AssociationId": "1a2b3c4d5e6f7g-1a2b3c-1a2b3c-1a2b3c-1a2b3c4d5e6f7g",
            "DocumentVersion": "1"
        }
]
}
```

Panduan: Menggunakan sinkronisasi data sumber daya untuk mengumpulkan data inventaris

Panduan berikut menjelaskan cara membuat konfigurasi sinkronisasi data sumber daya untuk Inventaris AWS Systems Manager dengan menggunakan AWS Command Line Interface (AWS CLI). Sinkronisasi data sumber daya secara otomatis mem-port data inventaris dari semua node terkelola ke bucket Pusat Amazon Simple Storage Service (Amazon S3). Sinkronisasi secara otomatis memperbarui data di bucket Amazon S3 sentral setiap kali data inventaris baru ditemukan.

Panduan ini juga menjelaskan cara menggunakan Amazon Athena dan Amazon untuk menanyakan dan QuickSight menganalisis data gabungan. Untuk informasi tentang cara membuat sinkronisasi data sumber daya dengan menggunakan Systems Manager di AWS Management Console, lihat Pengonfigurasian sinkronisasi data sumber daya untuk Inventaris. Untuk informasi tentang menanyakan inventaris dari beberapa akun Wilayah AWS dan menggunakan Systems Manager di bagianAWS Management Console, lihatMengkueri data inventaris dari beberapa Wilayah dan akun.

> **Note**
>
> Panduan ini menyertakan informasi tentang cara mengenkripsi sinkronisasi dengan menggunakan AWS Key Management Service (AWS KMS). Inventaris tidak mengumpulkan data khusus pengguna, kepemilikan, atau sensitif sehingga enkripsi bersifat opsional. Untuk informasi lebih lanjut tentang AWS KMS, lihat Panduan Developer AWS Key Management Service.

Sebelum Anda memulai

Tinjau atau selesaikan tugas-tugas berikut sebelum Anda memulai penelusuran di bagian ini:

- Kumpulkan data inventaris dari node terkelola Anda. Untuk tujuan QuickSight bagian Amazon Athena dan Amazon dalam panduan ini, kami sarankan Anda mengumpulkan data Aplikasi. Untuk informasi selengkapnya tentang cara mengumpulkan data inventaris, lihat Pengonfigurasian pengumpulan inventaris atauPanduan: Mengonfigurasi node terkelola untuk Inventaris dengan menggunakan CLI.
- (Opsional) Jika data inventaris disimpan dalam bucket Amazon Simple Storage Service (Amazon S3) yang AWS Key Management Service menggunakan enkripsi AWS KMS (), Anda juga harus mengonfigurasi akun IAM dan `Amazon-GlueServiceRoleForSSM` peran layanan untuk enkripsi. AWS KMS Jika Anda tidak mengonfigurasi akun IAM dan peran ini, Systems Manager akan ditampilkan `Cannot load Glue tables` saat Anda memilih tab Tampilan Terperinci di konsol. Untuk informasi selengkapnya, lihat (Opsional) Konfigurasikan izin untuk melihat dataAWS KMS terenkripsi.
- (Opsional) Jika Anda ingin mengenkripsi sinkronisasi data sumber daya dengan menggunakanAWS KMS, Anda harus membuat kunci baru yang menyertakan kebijakan berikut, atau Anda harus memperbarui kunci yang ada dan menambahkan kebijakan ini ke dalamnya.

```
{
    "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/
*"
                }
            }
        }
    ]
}
```

Untuk membuat sinkronisasi data sumber daya untuk Inventaris

1. Buka konsol Amazon S3 di https://console.aws.amazon.com/s3/.
2. Buat bucket untuk menyimpan data inventaris agregat Anda. Untuk informasi selengkapnya, lihat Membuat Bucket di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon. Catat nama bucket dan Wilayah AWS tempat Anda membuatnya.
3. Setelah Anda membuat bucket, pilih tab Izin, dan kemudian pilih Kebijakan Bucket.
4. Salin dan tempelkan kebijakan bucket berikut ke dalam editor kebijakan. Ganti *DOC-EXAMPLE-BUCKET* dan *account-id* dengan nama bucket Amazon S3 yang Anda buat dan ID Akun AWS yang valid. Saat menambahkan beberapa akun, tambahkan string kondisi tambahan dan ARN untuk setiap akun. Hapus placeholder tambahan dari contoh saat menambahkan satu akun. Secara opsional, ganti *bucket-prefix* dengan nama prefiks Amazon S3 (subdirektori). Jika Anda tidak membuat prefiks, hapus *bucket-prefix/* dari ARN dalam kebijakan.

   ```
   {
   ```

```
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": " SSMBucketDelivery",
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": "s3:PutObject",
      "Resource": [
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET/bucket-prefix/*/accountid=account-id/*"
      ],
      "Condition": {
        "StringEquals": {
          "s3:x-amz-acl": "bucket-owner-full-control",
          "aws:SourceAccount": [
            "account-id1",
            "account-id2",
            "account-id3",
            "account-id4"
          ]
        },
        "ArnLike": {
          "aws:SourceArn": [
            "arn:aws:ssm:*:account-id1:resource-data-sync/*",
            "arn:aws:ssm:*:account-id2:resource-data-sync/*",
            "arn:aws:ssm:*:account-id3:resource-data-sync/*",
            "arn:aws:ssm:*:account-id4:resource-data-sync/*"
          ]
        }
      }
    }
  ]
}
```

5. (Opsional) Jika Anda ingin mengenkripsi sinkronisasi, Anda harus menambahkan kondisi berikut ke kebijakan yang tercantum pada langkah sebelumnya. Tambahkan ini di `StringEquals` bagian.

```
"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"
```

Inilah contohnya:

```
"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. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI.

7. (Opsional) Jika Anda ingin mengenkripsi sinkronisasi, jalankan perintah berikut untuk memverifikasi bahwa kebijakan bucket menegakkan persyaratan kunci AWS KMS. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   Linux & macOS

   ```
   aws s3 cp ./A_file_in_the_bucket s3://DOC-EXAMPLE-BUCKET/prefix/ \
   --sse aws:kms \
   --sse-kms-key-id "arn:aws:kms:region:account_ID:key/KMS_key_id" \
   --region region, for example, us-east-2
   ```

   Windows

   ```
   aws s3 cp ./A_file_in_the_bucket s3://DOC-EXAMPLE-BUCKET/prefix/ ^
       --sse aws:kms ^
       --sse-kms-key-id "arn:aws:kms:region:account_ID:key/KMS_key_id" ^
       --region region, for example, us-east-2
   ```

8. Jalankan perintah berikut untuk membuat konfigurasi sinkronisasi data sumber daya dengan bucket Amazon S3 yang Anda buat pada awal prosedur ini. Perintah ini membuat sinkronisasi dari Wilayah AWS yang Anda masuki.

> **Note**
>
> Jika sinkronisasi dan bucket Amazon S3 target berada di berbagai wilayah, Anda mungkin akan dibebani harga transfer data. Untuk informasi lebih lanjut, lihat Harga Amazon S3.

Linux & macOS

```
aws ssm create-resource-data-sync \
--sync-name a_name \
--s3-destination "BucketName=DOC-EXAMPLE-BUCKET,Prefix=prefix_name,
 if_specified,SyncFormat=JsonSerDe,Region=bucket_region"
```

Windows

```
aws ssm create-resource-data-sync ^
--sync-name a_name ^
--s3-destination "BucketName=DOC-EXAMPLE-BUCKET,Prefix=prefix_name,
 if_specified,SyncFormat=JsonSerDe,Region=bucket_region"
```

Anda dapat menggunakan parameter `region` untuk menentukan di mana konfigurasi sinkronisasi harus dibuat. Dalam contoh berikut, data inventaris dari Wilayah us-west-1, akan disinkronkan di bucket Amazon S3 di Wilayah us-west-2.

Linux & macOS

```
aws ssm create-resource-data-sync \
    --sync-name InventoryDataWest \
    --s3-destination "BucketName=DOC-EXAMPLE-
BUCKET,Prefix=HybridEnv,SyncFormat=JsonSerDe,Region=us-west-2"
    --region us-west-1
```

Windows

```
aws ssm create-resource-data-sync ^
--sync-name InventoryDataWest ^
```

```
--s3-destination "BucketName=DOC-EXAMPLE-
BUCKET,Prefix=HybridEnv,SyncFormat=JsonSerDe,Region=us-west-2" ^ --region us-
west-1
```

(Opsional) Jika Anda ingin mengenkripsi sinkronisasi dengan menggunakan AWS KMS, jalankan perintah berikut untuk membuat sinkronisasi. Jika Anda mengenkripsi sinkronisasi, maka kunci AWS KMS dan bucket Amazon S3 harus berada di Wilayah yang sama.

Linux & macOS

```
aws ssm create-resource-data-sync \
--sync-name sync_name \
--s3-destination "BucketName=DOC-EXAMPLE-BUCKET,Prefix=prefix_name,
 if_specified,SyncFormat=JsonSerDe,AWSKMSKeyARN=arn:aws:kms:region:account_ID:key/
KMS_key_ID,Region=bucket_region" \
--region region
```

Windows

```
aws ssm create-resource-data-sync ^
--sync-name sync_name ^
--s3-destination "BucketName=DOC-EXAMPLE-BUCKET,Prefix=prefix_name,
 if_specified,SyncFormat=JsonSerDe,AWSKMSKeyARN=arn:aws:kms:region:account_ID:key/
KMS_key_ID,Region=bucket_region" ^
--region region
```

9. Jalankan perintah berikut untuk menampilkan status konfigurasi sinkronisasi.

```
aws ssm list-resource-data-sync
```

Jika Anda membuat konfigurasi sinkronisasi di Wilayah yang berbeda, maka Anda harus menentukan parameter `region`, seperti yang ditunjukkan dalam contoh berikut.

```
aws ssm list-resource-data-sync --region us-west-1
```

10. Setelah konfigurasi sinkronisasi berhasil dibuat, periksa bucket target di Amazon S3. Data inventaris harus ditampilkan dalam beberapa menit.

Bekerja dengan Data di Amazon Athena

Bagian berikut menjelaskan cara melihat dan mengkueri data di Amazon Athena. Sebelum memulai, kami merekomendasikan agar Anda mempelajari tentang Athena. Untuk informasi lebih lanjut, lihat Apa itu Amazon Athena? dan Menggunakan Data di Panduan Pengguna Amazon Athena.

Untuk melihat dan mengkueri data di Amazon Athena

1. Buka konsol Athena di https://console.aws.amazon.com/athena/.
2. Salin dan tempelkan pernyataan berikut ke editor kueri lalu pilih Jalankan Kueri.

```
CREATE DATABASE ssminventory
```

   Sistem membuat basis data yang disebut ssminventory.

3. Salin dan tempelkan pernyataan berikut ke editor kueri lalu pilih Jalankan Kueri. Ganti *DOC-EXAMPLE-BUCKET* dan *bucket_prefix* dengan nama dan prefiks dari target Amazon S3.

```
CREATE EXTERNAL TABLE IF NOT EXISTS ssminventory.AWS_Application (
Name string,
ResourceId string,
ApplicationType string,
Publisher string,
Version string,
InstalledTime string,
Architecture string,
URL string,
Summary string,
PackageId string
)
PARTITIONED BY (AccountId string, Region string, ResourceType string)
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'
WITH SERDEPROPERTIES (
  'serialization.format' = '1'
) LOCATION 's3://DOC-EXAMPLE-BUCKET/bucket_prefix/AWS:Application/'
```

4. Salin dan tempelkan pernyataan berikut ke editor kueri lalu pilih Jalankan Kueri.

```
MSCK REPAIR TABLE ssminventory.AWS_Application
```

   Sistem mempartisi tabel.

> **Note**
>
> Jika Anda membuat sinkronisasi data sumber daya dari Wilayah AWS atau Akun AWS tambahan, maka Anda harus menjalankan perintah ini lagi untuk memperbarui partisi. Anda mungkin juga perlu memperbarui kebijakan bucket Amazon S3.

5. Untuk mem-pratinjau data Anda, pilih ikon tampilan di samping tabel `AWS_Application`.

6. Salin dan tempelkan pernyataan berikut ke editor kueri lalu pilih Jalankan Kueri.

```
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
```

Kueri menampilkan angka berbagai versi dari `aws-cfn-bootstrap`, yang merupakan aplikasi AWS yang ada pada instans Amazon Elastic Compute Cloud (Amazon EC2) untuk Linux, macOS, dan Windows Server.

7. Salin dan tempelkan satu per satu pernyataan berikut ke editor kueri, ganti *DOC-EXAMPLE-BUCKET* dan *bucket-prefix* dengan informasi untuk Amazon S3, lalu pilih Jalankan Kueri. Pernyataan ini menyiapkan tabel inventaris tambahan di Athena.

```
CREATE EXTERNAL TABLE IF NOT EXISTS ssminventory.AWS_AWSComponent (
 `ResourceId` string,
  `Name` string,
  `ApplicationType` string,
  `Publisher` string,
  `Version` string,
  `InstalledTime` string,
  `Architecture` string,
  `URL` string
)
PARTITIONED BY (AccountId string, Region string, ResourceType string)
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'
WITH SERDEPROPERTIES (
  'serialization.format' = '1'
```

```
) LOCATION 's3://DOC-EXAMPLE-BUCKET/bucket-prefix/AWS:AWSComponent/'
```

```
MSCK REPAIR TABLE ssminventory.AWS_AWSComponent
```

```
CREATE EXTERNAL TABLE IF NOT EXISTS ssminventory.AWS_WindowsUpdate (
  `ResourceId` string,
  `HotFixId` string,
  `Description` string,
  `InstalledTime` string,
  `InstalledBy` string
)
PARTITIONED BY (AccountId string, Region string, ResourceType string)
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'
WITH SERDEPROPERTIES (
  'serialization.format' = '1'
) LOCATION 's3://DOC-EXAMPLE-BUCKET/bucket-prefix/AWS:WindowsUpdate/'
```

```
MSCK REPAIR TABLE ssminventory.AWS_WindowsUpdate
```

```
CREATE EXTERNAL TABLE IF NOT EXISTS ssminventory.AWS_InstanceInformation (
  `AgentType` string,
  `AgentVersion` string,
  `ComputerName` string,
  `IamRole` string,
  `InstanceId` string,
  `IpAddress` string,
  `PlatformName` string,
  `PlatformType` string,
  `PlatformVersion` string
)
PARTITIONED BY (AccountId string, Region string, ResourceType string)
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'
WITH SERDEPROPERTIES (
  'serialization.format' = '1'
) LOCATION 's3://DOC-EXAMPLE-BUCKET/bucket-prefix/AWS:InstanceInformation/'
```

```
MSCK REPAIR TABLE ssminventory.AWS_InstanceInformation
```

```
CREATE EXTERNAL TABLE IF NOT EXISTS ssminventory.AWS_Network (
```

```
  `ResourceId` string,
  `Name` string,
  `SubnetMask` string,
  `Gateway` string,
  `DHCPServer` string,
  `DNSServer` string,
  `MacAddress` string,
  `IPV4` string,
  `IPV6` string
)
PARTITIONED BY (AccountId string, Region string, ResourceType string)
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'
WITH SERDEPROPERTIES (
  'serialization.format' = '1'
) LOCATION 's3://DOC-EXAMPLE-BUCKET/bucket-prefix/AWS:Network/'
```

```
MSCK REPAIR TABLE ssminventory.AWS_Network
```

```
CREATE EXTERNAL TABLE IF NOT EXISTS ssminventory.AWS_PatchSummary (
  `ResourceId` string,
  `PatchGroup` string,
  `BaselineId` string,
  `SnapshotId` string,
  `OwnerInformation` string,
  `InstalledCount` int,
  `InstalledOtherCount` int,
  `NotApplicableCount` int,
  `MissingCount` int,
  `FailedCount` int,
  `OperationType` string,
  `OperationStartTime` string,
  `OperationEndTime` string
)
PARTITIONED BY (AccountId string, Region string, ResourceType string)
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'
WITH SERDEPROPERTIES (
  'serialization.format' = '1'
) LOCATION 's3://DOC-EXAMPLE-BUCKET/bucket-prefix/AWS:PatchSummary/'
```

```
MSCK REPAIR TABLE ssminventory.AWS_PatchSummary
```

Bekerja dengan Data di Amazon QuickSight

Bagian berikut memberikan ikhtisar dengan tautan untuk membangun visualisasi di Amazon QuickSight.

Untuk membangun visualisasi di Amazon QuickSight

1. Mendaftar ke Amazon QuickSight dan kemudian masuk ke QuickSight konsol.
2. Buat himpunan data dari tabel `AWS_Application` dan tabel lainnya yang telah Anda buat. Untuk informasi lebih lanjut, lihat Membuat Himpunan Data Menggunakan Data Amazon Athena.
3. Menggabungkan tabel. Misalnya, Anda dapat menggabungkan kolom `instanceid` dari `AWS_InstanceInformation` karena cocok dengan kolom `resourceid` di tabel inventaris lainnya. Untuk informasi lebih lanjut tentang penggabungan tabel, lihat Penggabungan Tabel.
4. Membangun visualisasi. Untuk informasi selengkapnya, lihat Bekerja dengan QuickSight Visual Amazon.

Memecahkan masalah dengan Inventaris Systems Manager

Topik ini mencakup informasi tentang cara memecahkan masalah atau kesalahan umum dengan Inventaris AWS Systems Manager. Jika Anda mengalami kesulitan melihat node di Systems Manager, lihatMemecahkan masalah ketersediaan node terkelola.

Topik

- Beberapa penerapan semua asosiasi pada dokumen 'AWS-GatherSoftwareInventory' tidak didukung
- Status eksekusi inventaris tidak pernah menampilkan tertunda
- Dokumen AWS-ListWindowsInventory gagal dijalankan
- Konsol tidak menampilkan Dasbor Inventaris | Tampilan Detail | Tab pengaturan
- UnsupportedAgent
- Dilewati
- Gagal
- Kepatuhan inventaris gagal untuk instans Amazon EC2
- Objek bucket S3 berisi data lama

Beberapa penerapan semua asosiasi pada dokumen '**AWS-GatherSoftwareInventory**' tidak didukung

Kesalahan yang `Multiple apply all associations with document 'AWS-GatherSoftwareInventory' are not supported` berarti bahwa satu atau lebih Wilayah AWS tempat Anda mencoba mengonfigurasi asosiasi Inventaris untuk semua node sudah dikonfigurasi dengan asosiasi inventaris untuk semua node. Jika perlu, Anda dapat menghapus asosiasi inventaris yang ada untuk semua node dan kemudian membuat yang baru. Untuk melihat asosiasi inventaris yang ada, pilih State Managerdi konsol Systems Manager dan kemudian cari asosiasi yang menggunakan dokumen `AWS-GatherSoftwareInventory` SSM. Jika asosiasi inventaris yang ada untuk semua node dibuat di beberapa Wilayah, dan Anda ingin membuat yang baru, Anda harus menghapus asosiasi yang ada dari setiap Wilayah di mana ia ada.

Status eksekusi inventaris tidak pernah menampilkan tertunda

Ada dua alasan mengapa pengumpulan inventaris tidak pernah keluar dari `Pending` status:

- Tidak ada node dalam yang dipilihWilayah AWS:

  Jika Anda membuat asosiasi inventaris global menggunakan Systems ManagerQuick Setup, status asosiasi inventaris (`AWS-GatherSoftwareInventory`dokumen) menunjukkan `Pending` jika tidak ada node yang tersedia di Wilayah yang dipilih.
- Izin tidak mencukupi:

  Asosiasi inventaris menunjukkan `Pending` jika satu atau beberapa node tidak memiliki izin untuk menjalankan Systems Manager Inventory. Verifikasi bahwa profil instans AWS Identity and Access Management (IAM) menyertakan kebijakan terkelola ManagedInstanceCoreAmazonSSM. Untuk informasi tentang cara menambahkan kebijakan ini ke profil instans, lihat Konfigurasi alternatif.

  Minimum, profil instans harus memiliki izin IAM berikut.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeAssociation",
                "ssm:ListAssociations",
                "ssm:ListInstanceAssociations",
```

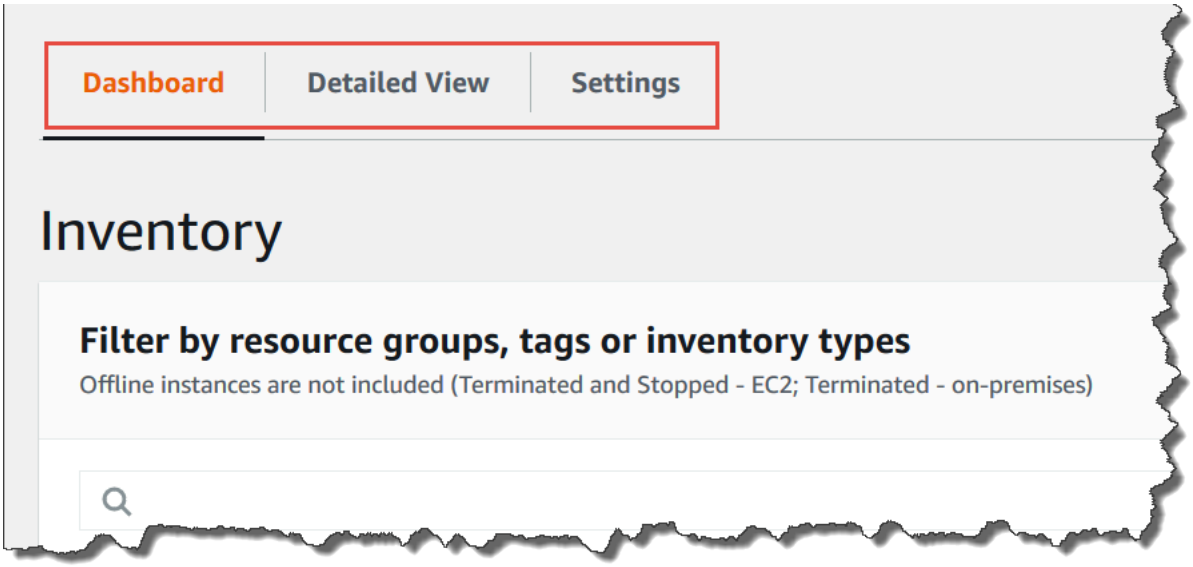
```
                "ssm:PutInventory",
                "ssm:PutComplianceItems",
                "ssm:UpdateAssociationStatus",
                "ssm:UpdateInstanceAssociationStatus",
                "ssm:UpdateInstanceInformation",
                "ssm:GetDocument",
                "ssm:DescribeDocument"
            ],
            "Resource": "*"
        }
    ]
}
```

Dokumen **`AWS-ListWindowsInventory`** gagal dijalankan

Dokumen `AWS-ListWindowsInventory` tidak lagi digunakan. Jangan gunakan dokumen ini untuk mengumpulkan inventaris. Sebagai gantinya, gunakan salah satu proses yang dijelaskan di Pengonfigurasian pengumpulan inventaris.

Konsol tidak menampilkan Dasbor Inventaris | Tampilan Detail | Tab pengaturan

Halaman Tampilan Detail Inventaris hanya tersedia di Wilayah AWS yang menawarkan Amazon Athena. Jika tab berikut tidak ditampilkan pada halaman Inventaris, artinya Athena tidak tersedia di Wilayah dan Anda tidak dapat menggunakan Tampilan Detail untuk mengkueri data.



UnsupportedAgent

Jika status rinci asosiasi inventaris ditampilkan UnsupportedAgent, dan status Asosiasi menunjukkan Gagal, maka versi AWS Systems Manager SSM Agent pada node terkelola tidak benar. Untuk membuat asosiasi inventaris global (untuk menginventarisasi semua node di AndaAkun AWS) misalnya, Anda harus menggunakan SSM Agent versi 2.0.790.0 atau yang lebih baru. Anda dapat melihat versi agen yang berjalan di setiap node di halaman Instans Terkelola di kolom versi Agen. Untuk informasi tentang cara memperbarui SSM Agent pada node Anda, lihatMemperbarui SSM Agent penggunaan Run Command.

Dilewati

Jika status asosiasi inventaris untuk node menunjukkan Skipped, ini berarti Anda membuat asosiasi inventaris global (untuk mengumpulkan inventaris dari semua node), tetapi node yang dilewati sudah memiliki asosiasi inventaris yang ditetapkan untuk itu. Asosiasi inventaris global tidak ditugaskan ke node ini, dan tidak ada inventaris yang dikumpulkan oleh asosiasi inventaris global. Namun, node masih akan melaporkan data inventaris saat asosiasi inventaris yang ada berjalan.

Jika Anda tidak ingin node dilewati oleh asosiasi inventaris global, Anda harus menghapus asosiasi inventaris yang ada. Untuk melihat asosiasi inventaris yang ada, pilih State Managerdi konsol Systems Manager dan kemudian cari asosiasi yang menggunakan dokumen `AWS-GatherSoftwareInventory` SSM.

Gagal

Jika status asosiasi inventaris untuk node menunjukkan Gagal, ini bisa berarti bahwa node memiliki beberapa asosiasi inventaris yang ditetapkan padanya. Sebuah node hanya dapat memiliki satu asosiasi inventaris yang ditetapkan pada satu waktu. Asosiasi inventaris menggunakan dokumen `AWS-GatherSoftwareInventory` AWS Systems Manager (dokumen SSM). Anda dapat menjalankan perintah berikut dengan menggunakan AWS Command Line Interface (AWS CLI) untuk melihat daftar asosiasi untuk node.

```
aws ssm describe-instance-associations-status
                --instance-id instance ID
```

Kepatuhan inventaris gagal untuk instans Amazon EC2

Kepatuhan inventaris untuk instans Amazon Elastic Compute Cloud (Amazon EC2) dapat gagal jika Anda menetapkan beberapa asosiasi inventaris ke instans.

Untuk mengatasi masalah ini, hapus satu atau beberapa asosiasi inventaris yang ditetapkan ke instance. Untuk informasi selengkapnya, lihat Menghapus asosiasi.

> **Note**
>
> Waspadai perilaku berikut jika Anda membuat beberapa asosiasi inventaris untuk node terkelola:
>
> - Setiap node dapat diberi asosiasi inventaris yang menargetkan semua node (--target “Key=InstanceIds, Values=*”).
> - Setiap node juga dapat diberi asosiasi tertentu yang menggunakan pasangan nilai kunci tag atau grup AWS sumber daya.
> - Jika sebuah node diberi beberapa asosiasi inventaris, status akan ditampilkan Dilewati untuk asosiasi yang belum berjalan. Asosiasi yang berjalan akhir-akhir ini menampilkan status sebenarnya dari asosiasi inventaris.
> - Jika sebuah node diberi beberapa asosiasi inventaris dan masing-masing menggunakan pasangan nilai kunci tag, maka asosiasi inventaris tersebut gagal berjalan di node karena konflik tag. Asosiasi masih berjalan pada node yang tidak memiliki konflik nilai kunci tag.

Objek bucket S3 berisi data lama

Data di dalam objek bucket Amazon S3 diperbarui saat asosiasi inventaris berhasil dan data baru ditemukan. Objek bucket Amazon S3 diperbarui untuk setiap node saat asosiasi berjalan dan gagal, tetapi data di dalam objek tidak diperbarui dalam kasus ini. Data di dalam objek bucket Amazon S3 hanya akan diperbarui jika asosiasi berjalan dengan sukses. Ketika asosiasi inventaris gagal, Anda akan melihat data lama di objek bucket Amazon S3.

AWS Systems Manager Aktivasi Hibrid

Untuk mengonfigurasi mesin non-EC2 untuk digunakan AWS Systems Manager di lingkungan hybrid dan multicloud, Anda membuat aktivasi hybrid. Jenis mesin non-EC2 yang didukung sebagai node terkelola meliputi:

- Server di tempat Anda sendiri (server lokal)
- AWS IoT GreengrassPerangkat inti
- AWS IoTdan perangkat AWS non-tepi

- Mesin virtual (VM), termasuk VM di lingkungan cloud lainnya

Ketika Anda menjalankan create-activationperintah untuk memulai proses aktivasi hybrid, Anda menerima kode aktivasi dan ID dalam respons perintah. Anda kemudian menyertakan kode aktivasi dan ID dengan perintah untuk menginstal SSM Agent pada mesin, seperti yang dijelaskan pada langkah 3 dariMenyiapkan Manajer Sistem untuk lingkungan hybrid dan multicloud. Proses aktivasi ini berlaku untuk semua jenis mesin non-EC2 kecuali perangkat AWS IoT Greengrass inti. Untuk informasi tentang pengaktifan perangkat AWS IoT Greengrass inti untuk Systems Manager, lihatAWS Systems ManagerMenyiapkan perangkat edge.

> **Note**
>
> Support saat ini tidak disediakan untuk mesin non-EC2macOS.

Tentang tingkatan instans Systems Manager

AWS Systems Managermenawarkan tingkat instans standar dan tingkat instans lanjutan. Keduanya mendukung node terkelola di lingkungan hybrid dan multicloud Anda. Tingkat instans standar memungkinkan Anda mendaftarkan maksimum 1.000 mesin per per. Akun AWS Wilayah AWS Jika Anda perlu mendaftarkan lebih dari 1.000 mesin dalam satu akun dan Wilayah, gunakan tingkat instans lanjutan. Anda dapat membuat sebanyak mungkin node terkelola yang Anda inginkan di tingkat instans lanjutan. Semua node terkelola yang dikonfigurasi untuk Systems Manager diberi harga pay-per-use berdasarkan harga. Untuk informasi lebih lanjut dalam mengaktifkan tingkat instans lanjutan, lihat. Mengaktifkan tingkat instans lanjutan Untuk informasi selengkapnya tentang harga, lihat AWS Systems Manager Harga.

> **Note**
>
> - Instans lanjutan juga memungkinkan Anda terhubung ke node non-EC2 Anda di lingkungan hibrid dan multicloud dengan menggunakan. AWS Systems Manager Session Manager Session Managermenyediakan akses shell interaktif ke instans Anda. Untuk informasi selengkapnya, lihat AWS Systems Manager Session Manager.
> - Kuota instans standar juga berlaku untuk instans EC2 yang menggunakan aktivasi on-premise Systems Manager (yang bukan merupakan skenario umum).
> - Untuk menambal aplikasi yang dirilis oleh Microsoft di instans on-premise mesin virtual (VM), aktifkan tingkat instans lanjutan. Biaya dikenakan untuk menggunakan tingkat instans

lanjutan. Biaya tambahan dikenakan untuk menambal aplikasi yang dirilis oleh instans Microsoft Amazon Elastic Compute Cloud (Amazon EC2). Untuk informasi selengkapnya, lihat Mengenai aplikasi patching yang dikeluarkan oleh Microsoft pada Windows Server.

AWS Systems Manager Session Manager

Session Manageradalah AWS Systems Manager kemampuan yang dikelola sepenuhnya. DenganSession Manager, Anda dapat mengelola instans Amazon Elastic Compute Cloud (Amazon EC2), perangkat edge, server lokal, dan mesin virtual (VM). Anda dapat menggunakan shell berbasis browser satu-klik interaktif atau (). AWS Command Line Interface AWS CLISession Managermenyediakan manajemen node yang aman dan dapat diaudit tanpa perlu membuka port masuk, memelihara host bastion, atau mengelola kunci SSH. Session Managerjuga memungkinkan Anda untuk mematuhi kebijakan perusahaan yang memerlukan akses terkontrol ke node terkelola, praktik keamanan yang ketat, dan log yang dapat diaudit sepenuhnya dengan detail akses node, sambil memberikan pengguna akhir akses lintas platform satu klik sederhana ke node terkelola Anda. Untuk memulaiSession Manager, buka konsol Systems Manager. Di panel navigasi, pilih Session Manager.

Bagaimana bisa Session Manager menguntungkan organisasi saya?

Session Managermenawarkan manfaat ini:

- Kontrol akses terpusat ke node terkelola menggunakan kebijakan IAM

  Administrator memiliki satu tempat untuk memberikan dan mencabut akses ke node terkelola. Dengan hanya menggunakan kebijakan AWS Identity and Access Management (IAM), Anda dapat mengontrol pengguna atau grup individu mana di organisasi Anda yang dapat digunakan Session Manager dan node terkelola mana yang dapat mereka akses.
- Tidak ada port masuk yang terbuka dan tidak perlu mengelola host bastion atau kunci SSH

  Membiarkan port SSH masuk dan PowerShell port jarak jauh terbuka di node terkelola Anda sangat meningkatkan risiko entitas menjalankan perintah yang tidak sah atau berbahaya pada node yang dikelola. Session Managermembantu Anda meningkatkan postur keamanan Anda dengan membiarkan Anda menutup port masuk ini, membebaskan Anda dari mengelola kunci dan sertifikat SSH, host benteng, dan kotak lompat.
- Akses sekali klik ke node terkelola dari konsol dan CLI

Menggunakan AWS Systems Manager konsol atau konsol Amazon EC2, Anda dapat memulai sesi dengan satu klik. Dengan menggunakan AWS CLI, Anda juga dapat memulai sesi yang menjalankan satu perintah atau urutan perintah. Karena izin untuk node terkelola disediakan melalui kebijakan IAM alih-alih kunci SSH atau mekanisme lainnya, waktu koneksi sangat berkurang.

- Connect ke instans Amazon EC2 dan node terkelola non-EC2 di lingkungan hybrid dan multicloud

  Anda dapat terhubung ke instans Amazon Elastic Compute Cloud (Amazon EC2) dan node non-EC2 di lingkungan hybrid dan multicloud Anda.

  Untuk terhubung ke node non-EC2 menggunakanSession Manager, Anda harus terlebih dahulu mengaktifkan tingkat instance lanjutan. Ada biaya untuk menggunakan tingkat instance lanjutan. Namun, tidak ada biaya tambahan untuk terhubung ke instans EC2 menggunakan. Session Manager Untuk informasi, lihat Mengonfigurasi tingkat instans.

- Penerusan port

  Arahkan ulang port apa pun di dalam node terkelola Anda ke port lokal pada klien. Setelah itu, sambungkan ke port lokal dan akses aplikasi server yang berjalan di dalam node.

- Dukungan lintas platform untukWindows,Linux, dan macOS

  Session Managermemberikan dukungan untukWindows,Linux, dan macOS dari satu alat. Misalnya, Anda tidak perlu menggunakan klien SSH untuk Linux dan macOS mengelola node atau koneksi RDP untuk Windows Server node terkelola.

- Aktivitas sesi logging dan audit

  Untuk memenuhi persyaratan operasional atau keamanan di organisasi Anda, Anda mungkin perlu memberikan catatan koneksi yang dibuat ke node terkelola dan perintah yang dijalankan pada node tersebut. Anda juga dapat menerima pemberitahuan jika pengguna di organisasi Anda memulai atau mengakhiri aktivitas sesi.

  Kemampuan logging dan audit disediakan melalui integrasi dengan hal-hal berikut: Layanan AWS

  - AWS CloudTrail— AWS CloudTrail menangkap informasi tentang panggilan Session Manager API yang dilakukan di Anda Akun AWS dan menuliskannya ke file log yang disimpan di bucket Amazon Simple Storage Service (Amazon S3) S3 yang Anda tentukan. Satu ember digunakan untuk semua CloudTrail log untuk akun Anda. Untuk informasi selengkapnya, lihat Pencatatan panggilan AWS Systems Manager API dengan AWS CloudTrail.

- Amazon Simple Storage Service— Anda dapat memilih untuk menyimpan data log sesi dalam bucket Amazon S3 menurut pilihan Anda untuk tujuan debugging dan pemecahan masalah. Data log dapat dikirim ke bucket Amazon S3 Anda dengan atau tanpa enkripsi menggunakan AWS KMS key Anda. Untuk informasi selengkapnya, lihat Log data sesi menggunakan Amazon S3 (konsol).
- Amazon CloudWatch Logs — CloudWatch Log memungkinkan Anda untuk memantau, menyimpan, dan mengakses file log dari berbagai Layanan AWS. Anda dapat mengirim data log sesi ke grup CloudWatch log Log untuk tujuan debugging dan pemecahan masalah. Data log dapat dikirim ke grup log Anda dengan atau tanpa AWS KMS enkripsi menggunakan kunci KMS Anda. Untuk informasi selengkapnya, lihat Data sesi logging menggunakan Amazon CloudWatch Logs (konsol).
- Amazon EventBridge dan Amazon Simple Notification Service - EventBridge memungkinkan Anda mengatur aturan untuk mendeteksi kapan perubahan terjadi pada AWS sumber daya yang Anda tentukan. Anda dapat membuat aturan untuk mendeteksi ketika pengguna di organisasi Anda memulai atau menghentikan sesi, dan kemudian menerima pemberitahuan melalui Amazon SNS (misalnya, pesan teks atau email) tentang acara tersebut. Anda juga dapat mengonfigurasi CloudWatch acara untuk memulai tanggapan lain. Untuk informasi selengkapnya, lihat Memantau aktivitas sesi sesi sesi menggunakan Amazon EventBridge (konsol) .

> **Note**
>
> Logging tidak tersedia untuk Session Manager sesi yang terhubung melalui port forwarding atau SSH. Ini karena SSH mengenkripsi semua data sesi, dan Session Manager hanya berfungsi sebagai terowongan untuk koneksi SSH.

Siapa yang harus menggunakanSession Manager?

- Setiap AWS pelanggan yang ingin meningkatkan keamanan dan postur audit mereka, mengurangi overhead operasional dengan memusatkan kontrol akses pada node yang dikelola, dan mengurangi akses node masuk.
- Pakar Keamanan Informasi yang ingin memantau dan melacak akses dan aktivitas node terkelola, menutup port masuk pada node terkelola, atau mengizinkan koneksi ke node terkelola yang tidak memiliki alamat IP publik.

- Administrator yang ingin memberikan dan mencabut akses dari satu lokasi, dan yang ingin memberikan satu solusi kepada pengguna untuk LinuxmacOS, dan Windows Server node terkelola.
- Pengguna yang ingin terhubung ke node terkelola hanya dengan satu klik dari browser atau AWS CLI tanpa harus memberikan kunci SSH.

Apa saja fitur utamaSession Manager?

- Support untukWindows Server, Linux dan node macOS terkelola

  Session Managermemungkinkan Anda membuat sambungan aman ke instans Amazon Elastic Compute Cloud (EC2), perangkat edge, server lokal, dan mesin virtual (VM) Amazon Elastic Compute Cloud (EC2). Untuk daftar tipe sistem operasi yang didukung, lihat Menyiapkan Session Manager.

  > **Note**
  >
  > Session Managerdukungan untuk mesin lokal disediakan hanya untuk tingkat instance lanjutan. Untuk informasi, lihat Mengaktifkan tingkat instans lanjutan.

- Konsol, CLI, dan akses SDK ke kemampuan Session Manager

  Anda dapat bekerja dengan Session Manager cara-cara berikut:

  AWS Systems Manager Konsol mencakup akses ke semua Session Manager kemampuan untuk administrator dan pengguna akhir. Anda dapat melakukan tugas apa pun yang berkaitan dengan sesi Anda dengan menggunakan konsol Systems Manager.

  Konsol Amazon EC2 menyediakan kemampuan bagi pengguna akhir untuk terhubung ke instans EC2 yang telah diberikan izin sesi.

  AWS CLITermasuk akses ke Session Manager kemampuan untuk pengguna akhir. Anda dapat memulai sesi, melihat daftar sesi, dan mengakhiri sesi secara permanen dengan menggunakan AWS CLI.

> **Note**
>
> Untuk menggunakan perintah AWS CLI to run session, Anda harus menggunakan versi 1.16.12 dari CLI (atau yang lebih baru), dan Anda harus menginstal Session Manager plugin di mesin lokal Anda. Untuk informasi, lihat Instal Session Manager plugin untuk AWS CLI. Untuk melihat pluginGitHub, lihat session-manager-plugin.

- Kontrol akses IAM

  Melalui penggunaan kebijakan IAM, Anda dapat mengontrol anggota organisasi mana yang dapat memulai sesi ke node terkelola dan node mana yang dapat mereka akses. Anda juga dapat memberikan akses sementara ke node terkelola Anda. Misalnya, Anda mungkin ingin memberikan insinyur on-call (atau sekelompok insinyur on-call) akses ke server produksi hanya selama durasi rotasi mereka.

- Dukungan kemampuan logging dan audit

  Session Managermemberi Anda opsi untuk mengaudit dan mencatat riwayat sesi di Anda Akun AWS melalui integrasi dengan sejumlah lainnya. Layanan AWS Untuk informasi lebih lanjut, lihat Mengaudit aktivitas sesi dan Mengaktifkan dan menonaktifkan pencatatan aktivitas sesi.

- Profil shell yang dapat dikonfigurasi

  Session Managermemberi Anda opsi untuk mengonfigurasi preferensi dalam sesi. Profil yang dapat disesuaikan ini mengizinkan Anda untuk menentukan preferensi seperti preferensi shell, variabel lingkungan, direktori kerja, dan menjalankan beberapa perintah ketika sesi dimulai.

- Dukungan enkripsi data kunci pelanggan

  Anda dapat mengonfigurasi Session Manager untuk mengenkripsi log data sesi yang Anda kirim ke bucket Amazon Simple Storage Service (Amazon S3) atau streaming ke grup log Log. CloudWatch Anda juga dapat mengonfigurasi Session Manager untuk mengenkripsi lebih lanjut data yang dikirimkan antara mesin klien dan node terkelola Anda selama sesi Anda. Untuk informasi, lihat Mengaktifkan dan menonaktifkan pencatatan aktivitas sesi dan Mengkonfigurasi preferensi sesi.

- AWS PrivateLink dukungan untuk node terkelola tanpa alamat IP publik

  Anda juga dapat mengatur Titik Akhir VPC untuk Systems Manager AWS PrivateLink untuk mengamankan sesi Anda lebih lanjut. AWS PrivateLink membatasi semua lalu lintas jaringan antara node terkelola, Systems Manager, dan Amazon EC2 ke jaringan Amazon. Untuk informasi selengkapnya, lihat Membuat titik akhir VPC.

- Terowongan

  Dalam sesi, gunakan dokumen Session-type AWS Systems Manager (SSM) untuk mengarahkan lalu lintas, seperti http atau protokol kustom, antara port lokal pada mesin klien dan port jarak jauh pada node terkelola.

- Perintah interaktif

  Buat dokumen SSM tipe sesi yang menggunakan sesi untuk menjalankan satu perintah secara interaktif, memberi Anda cara untuk mengelola apa yang dapat dilakukan pengguna pada node terkelola.

Apa itu sesi?

Sesi adalah koneksi yang dibuat ke node terkelola menggunakanSession Manager. Sesi didasarkan pada saluran komunikasi dua arah yang aman antara klien (Anda) dan node terkelola jarak jauh yang mengalirkan input dan output untuk perintah. Lalu lintas antara klien dan node terkelola dienkripsi menggunakan TLS 1.2, dan permintaan untuk membuat koneksi ditandatangani menggunakan Sigv4. Komunikasi dua arah ini memungkinkan bash interaktif dan PowerShell akses ke node terkelola. Anda juga dapat menggunakan kunci AWS Key Management Service (AWS KMS) untuk lebih mengenkripsi data di luar enkripsi TLS default.

Misalnya, katakanlah bahwa John adalah insinyur on-call di departemen IT Anda. Dia menerima pemberitahuan tentang masalah yang mengharuskannya terhubung dari jarak jauh ke node yang dikelola, seperti kegagalan yang memerlukan pemecahan masalah atau arahan untuk mengubah opsi konfigurasi sederhana pada node. Menggunakan AWS Systems Manager konsol, konsol Amazon EC2, atau AWS CLI, John memulai sesi yang menghubungkannya ke node terkelola, menjalankan perintah pada node yang diperlukan untuk menyelesaikan tugas, dan kemudian mengakhiri sesi.

Ketika John mengirim perintah pertama untuk memulai sesi, Session Manager layanan mengotentikasi ID-nya, memverifikasi izin yang diberikan kepadanya oleh kebijakan IAM, memeriksa pengaturan konfigurasi (seperti memverifikasi batas yang diizinkan untuk sesi), dan mengirim pesan SSM Agent ke untuk membuka koneksi dua arah. Setelah koneksi dibuat dan John mengetik perintah berikutnya, output perintah dari SSM Agent diunggah ke saluran komunikasi ini dan dikirim kembali ke mesin lokalnya.

Topik

- Menyiapkan Session Manager

- Bekerja dengan Session Manager
- Mengaudit aktivitas sesi
- Mengaktifkan dan menonaktifkan pencatatan aktivitas sesi
- Skema dokumen sesi
- Pemecahan Masalah Session Manager

Menyiapkan Session Manager

Sebelum Anda menggunakanAWS Systems Manager Session Manageruntuk terhubung ke node yang dikelola di akun Anda, selesaikan langkah-langkah dalam topik berikut.

Topik

- Langkah 1: Session Manager Prasyarat lengkap
- Langkah 2: Verifikasi atau tambahkan izin instans untuk Session Manager
- Langkah 3: Kontrol akses sesi ke node yang dikelola
- Langkah 4: Konfigurasi preferensi sesi
- Langkah 5: (Opsional) Batasi akses ke perintah dalam sesi
- Langkah 6: (Opsional) GunakanAWS PrivateLink untuk menyiapkan VPC endpoint untukSession Manager
- Langkah 7: (Opsional) Aktifkan atau nonaktifkan izin administratif akun ssm-user
- Langkah 8: (Opsional) Izinkan dan kontrol izin untuk koneksi SSH melalui Session Manager

Langkah 1: Session Manager Prasyarat lengkap

Sebelum menggunakanSession Manager, pastikan lingkungan Anda memenuhi persyaratan berikut.

Prasyarat Session Manager

Persyaratan	Deskripsi
Sistem operasi yang didukung	Session Managermendukung koneksi ke instans Amazon Elastic Compute Cloud (Amazon EC2), selain mesin non-EC2 di lingkungan hybrid dan multicloud Anda yang menggunakan tingkat instans lanjutan.

Persyaratan	Deskripsi
	Session Managermendukung versi sistem operasi berikut: **Note** Session Managermendukung instans EC2, perangkat edge, dan server lokal serta mesin virtual (VM) di lingkungan hybrid dan multicloud Anda yang menggunakan tingkat instans lanjutan. Untuk informasi selengkapnya tentang instans lanjutan, lihat Mengonfigurasi tingkat instans. Linux dan macOS Session Managermendukung semua versi Linux dan macOS yang didukung oleh AWS Systems Manager. Untuk informasi, lihat Sistem operasi dan jenis mesin yang didukung. Windows Session Managermendukung Windows Server 2012 hingga Windows Server 2022. **Note** Tidak mendukung Microsoft Windows Server 2016 Nano.

Persyaratan	Deskripsi
SSM Agent	Minimal, AWS Systems Manager SSM Agent versi 2.3.68.0 atau yang lebih baru harus diinstal pada node terkelola yang ingin Anda sambungkan melalui sesi. Untuk menggunakan opsi untuk mengenkripsi data sesi menggunakan kunci yang dibuat di AWS Key Management Service (AWS KMS), versi 2.3.539.0 atau yang lebih baru SSM Agent harus diinstal pada node terkelola. Untuk menggunakan profil shell dalam sesi, SSM Agent versi 3.0.161.0 atau yang lebih baru harus diinstal pada node terkelola. Untuk memulai Session Manager port forwardin g atau sesi SSH, SSM Agent versi 3.0.222.0 atau yang lebih baru harus diinstal pada node terkelola. Untuk melakukan streaming data sesi menggunakan Amazon CloudWatch Logs, SSM Agent versi 3.0.284.0 atau yang lebih baru harus diinstal pada node terkelola. Untuk informasi tentang cara menentukan nomor versi yang berjalan pada instance, lihatMemeriksa nomor SSM Agent versi. Untuk informasi tentang menginstal secara manual atau memperbarui secara otomatisSSM Agent, lihatBekerja dengan SSM Agent. Tentang akun ssm-user Dimulai dengan versi 2.3.50.0 dariSSM Agent, agen membuat akun pengguna pada node terkelola, dengan izin root atau administrator,

Persyaratan	Deskripsi
	dipanggil. `ssm-user` (Pada versi sebelum 2.3.612.0, akun dibuat saat SSM Agent memulai atau memulai ulang. Pada versi 2.3.612.0 dan yang lebih baru, `ssm-user` dibuat pertama kali sesi dimulai pada node terkelola.) Sesi diluncurkan menggunakan kredensial administratif akun pengguna ini. Untuk informasi tentang membatasi kontrol administratif untuk akun ini, lihat Menonaktifkan atau mengaktifkan izin administratif akun ssm-user. ssm-user pada pengendali domain Windows Server Dimulai dengan SSM Agent versi 2.3.612.0, `ssm-user` akun tidak dibuat secara otomatis pada node terkelola yang digunakan sebagai pengontrol domain. Windows Server Untuk digunakan Session Manager pada Windows Server mesin yang digunakan sebagai pengontrol domain, Anda harus membuat `ssm-user` akun secara manual jika belum ada, dan menetapkan izin Administrator Domain kepada pengguna. Windows ServerAktif, SSM Agent tetapkan kata sandi baru untuk `ssm-user` akun setiap kali sesi dimulai, jadi Anda tidak perlu menentukan kata sandi saat membuat akun.

Persyaratan	Deskripsi
Konektivitas ke titik akhir	Node terkelola yang Anda sambungkan juga harus mengizinkan lalu lintas keluar HTTPS (port 443) ke titik akhir berikut: • ec2pesan. *wilayah .amazonaws.com* • ssm. *wilayah .amazonaws.com* • ssmmessages. *wilayah .amazonaws.com* Untuk informasi selengkapnya, lihat topik berikut: • Referensi: ec2messages, ssmmessages, dan operasi API lainnya • Bagaimana cara membuat titik akhir VPC sehingga saya dapat menggunakan Systems Manager untuk mengelola instans EC2 pribadi tanpa akses internet? di pusat AWS re:Post pengetahuan. Atau, Anda dapat terhubung ke titik akhir yang diperlukan dengan menggunakan titik akhir antarmuka. Untuk informasi selengkapnya, lihat Langkah 6: (Opsional) GunakanAWS PrivateLink untuk menyiapkan VPC endpoint untukSession Manager.

Persyaratan	Deskripsi
AWS CLI	(Opsional) Jika Anda menggunakan AWS Command Line Interface (AWS CLI) untuk memulai sesi (alih-alih menggunakan AWS Systems Manager konsol atau konsol Amazon EC2), versi 1.16.12 atau yang lebih baru dari CLI harus diinstal pada mesin lokal Anda. Anda dapat menghubungi `aws --version` untuk memeriksa versi. Jika Anda perlu menginstal atau memutakhirkan CLI, lihat Menginstal AWS Command Line Interface di AWS Command Line Interface Panduan Pengguna. **Important** Versi terbaru dirilis setiap kali kemampuan baru ditambahkan ke Systems Manager atau pembaruan dibuat untuk kemampuan yang ada. SSM Agent Gagal menggunakan agen versi terbaru dapat mencegah node terkelola Anda menggunakan berbagai kemampuan dan fitur Systems Manager. Untuk alasan itu, kami menyarankan Anda mengotomatiskan proses menjaga agar tetap SSM Agent up to date pada mesin Anda. Untuk informasi, lihat Mengotomatiskan pembaruan ke SSM Agent. Berlangganan halaman Catatan SSM Agent Rilis GitHub untuk mendapatkan pemberitahuan tentang SSM Agent pembaruan.

Persyaratan	Deskripsi
	Selain itu, untuk menggunakan CLI untuk mengelola node AndaSession Manager, Anda harus terlebih dahulu menginstal Session Manager plugin di mesin lokal Anda. Untuk informasi, lihat Instal Session Manager plugin untuk AWS CLI.
Aktifkan tingkat instance lanjutan (lingkungan hybrid dan multicloud)	Untuk terhubung ke mesin non-EC2 menggunakanSession Manager, Anda harus mengaktifkan tingkat instance lanjutan di Akun AWS dan di Wilayah AWS mana Anda membuat aktivasi hibrida untuk mendaftarkan mesin non-EC2 sebagai node terkelola. Biaya dikenakan untuk menggunakan tingkat instans lanjutan. Untuk informasi selengkapnya tentang tingkat instance lanjutan, lihat. Mengonfigurasi tingkat instans

Persyaratan	Deskripsi
Verifikasi izin peran layanan IAM (lingkungan hybrid dan multicloud)	Node yang diaktifkan hibrida menggunakan peran layanan AWS Identity and Access Management (IAM) yang ditentukan dalam aktivasi hibrida untuk berkomunikasi dengan operasi Systems Manager API. Peran layanan ini harus berisi izin yang diperlukan untuk terhubung ke mesin hybrid dan multicloud Anda menggunakan. Session Manager Jika peran layanan Anda berisi kebijakan AWS terkelola `AmazonSSMManagedInstanceCore` , izin yang diperlukan untuk sudah Session Manager disediakan. Jika Anda menemukan bahwa peran layanan tidak berisi izin yang diperlukan, Anda harus membatalkan pendaftaran instance terkelola dan mendaftarkannya dengan aktivasi hibrida baru yang menggunakan peran layanan IAM dengan izin yang diperlukan. Untuk informasi selengkapnya tentang membatalkan pendaftaran instans terkelola, lihat. Menderegistrasi node terkelola dalam lingkungan hybrid dan multicloud Untuk informasi selengkapnya tentang membuat kebijakan IAM dengan Session Manager izin, lihat Langkah 2: Verifikasi atau tambahkan izin instans untuk. Session Manager

Langkah 2: Verifikasi atau tambahkan izin instans untuk Session Manager

Secara default, AWS Systems Manager tidak memiliki izin untuk melakukan tindakan pada instans Anda. Anda dapat memberikan izin instans di tingkat akun menggunakan peran AWS Identity and Access Management (IAM), atau pada tingkat instans menggunakan profil instans. Jika kasus penggunaan Anda memungkinkan, sebaiknya berikan akses di tingkat akun menggunakan

Konfigurasi Manajemen Host Default. Jika Anda telah menyiapkan Konfigurasi Manajemen Host Default untuk akun Anda menggunakan `AmazonSSMManagedEC2InstanceDefaultPolicy` kebijakan, Anda dapat melanjutkan ke langkah berikutnya. Untuk informasi selengkapnya tentang konfigurasi manajemen Host Default, lihatMenggunakan pengaturan Konfigurasi Manajemen Host Default.

Atau, Anda dapat menggunakan profil instans untuk memberikan izin yang diperlukan untuk instans Anda. Profil instans meneruskan peran IAM ke instans Amazon EC2. Anda dapat melampirkan profil instans IAM ke instans Amazon EC2 saat Anda meluncurkannya atau ke instans yang diluncurkan sebelumnya. Untuk informasi selengkapnya, lihat Menggunakan profil instans.

Untuk server lokal atau mesin virtual (VM), izin disediakan oleh peran layanan IAM yang terkait dengan aktivasi hibrid yang digunakan untuk mendaftarkan server lokal dan VM Anda ke Systems Manager. Server on-premise dan VM tidak menggunakan profil instans.

Jika Anda sudah menggunakan kemampuan Systems Manager lainnya, seperti Run Command atauParameter Store, profil instans dengan izin dasar yang diperlukan untuk Session Manager mungkin sudah terlampir ke instans Amazon EC2 Anda. Jika profil instans yang berisi kebijakan AWS terkelola `AmazonSSMManagedInstanceCore` sudah terlampir ke instans Anda, izin yang diperlukan untuk sudah Session Manager disediakan. Ini juga berlaku jika peran layanan IAM yang digunakan dalam aktivasi hibrid Anda berisi kebijakan `AmazonSSMManagedInstanceCore` terkelola.

> **Important**
>
> Anda tidak dapat mengubah peran layanan IAM yang terkait dengan aktivasi hibrid. Jika Anda menemukan bahwa peran layanan tidak mengandung izin yang diperlukan, Anda harus membatalkan pendaftaran instans terkelola dan mendaftarkannya dengan aktivasi hibrid baru yang menggunakan peran layanan dengan izin yang diperlukan. Untuk informasi selengkapnya tentang membatalkan pendaftaran instans terkelola, lihat. Menderegistrasi node terkelola dalam lingkungan hybrid dan multicloud Untuk informasi selengkapnya tentang membuat peran layanan IAM untuk mesin on-premise, lihat Membuat peran layanan IAM untuk lingkungan hibrida.

Namun, dalam beberapa kasus, Anda mungkin perlu memodifikasi izin yang terlampir ke profil instans Anda. Misalnya, Anda ingin memberikan serangkaian izin instans yang lebih sempit, Anda telah membuat kebijakan khusus untuk profil instans Anda, atau Anda ingin menggunakan enkripsi Amazon Simple Storage Service (Amazon S3) atau enkripsi AWS Key Management Service (AWS

KMS) opsi enkripsi untuk mengamankan data sesi. Untuk kasus ini, lakukan salah satu dari berikut ini untuk mengizinkan Session Manager tindakan dilakukan pada instans Anda:

- Sematkan izin untuk Session Manager tindakan dalam peran IAM khusus

  Untuk menambahkan izin untuk Session Manager tindakan ke peran IAM yang ada yang tidak bergantung pada kebijakan default AWS yang disediakan`AmazonSSMManagedInstanceCore`, ikuti langkah-langkah dalam. MenambahkanSession Managerizin untuk peran IAM yang ada

- Buat peran IAM kustom dengan Session Manager izin saja

  Untuk membuat peran IAM yang hanya berisi izin hanya berisi izin untuk Session Manager tindakan, ikuti langkah-langkah dalam. Buat peran IAM khusus untuk Session Manager

- Buat dan gunakan peran IAM baru dengan izin untuk semua tindakan Systems Manager

  Untuk membuat peran IAM untuk instans terkelola Systems Manager yang menggunakan kebijakan default yang disediakan oleh AWS untuk memberikan semua izin Systems Manager, ikuti langkah-langkah dalam mengkonfigurasi izin instans untuk Systems Manager.

Topik

MenambahkanSession Managerizin untuk peran IAM yang ada

Gunakan prosedur berikut untuk menambahkanSession Managerizin ke yang sudah adaAWS Identity and Access Management(IAM) peran. Dengan menambahkan izin ke peran yang ada, Anda dapat meningkatkan keamanan lingkungan komputasi Anda tanpa harus menggunakanAWS `AmazonSSMManagedInstanceCore`kebijakan untuk izin misalnya.

> **Note**
>
> Perhatikan informasi berikut:
>
> - Prosedur ini mengasumsikan bahwa peran Anda yang ada sudah termasuk Manajer Sistem lainnya`ssm`izin untuk tindakan yang ingin Anda izinkan akses. Kebijakan ini saja tidak cukup untuk digunakanSession Manager.

- Contoh kebijakan berikut mencakups3:GetEncryptionConfigurationtindakan. Tindakan ini diperlukan jika Anda memilihMenegakkan enkripsi log S3pilihan diSession Managerpreferensi logging.

Untuk menambahkanSession Managerizin untuk peran yang ada (konsol)

1. Masuk ke AWS Management Console dan buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran.
3. Pilih nama peran yang Anda tambahkan izin.
4. Pilih tab Izin.
5. PilihTambahkan izin, dan kemudian pilihBuat kebijakan inline.
6. Pilih tab JSON.
7. Ganti konten kebijakan default dengan konten berikut. Ganti*kunci-nama*dengan Amazon Resource Name (ARN) dariAWS Key Management Servicekunci (AWS KMS key) yang ingin Anda gunakan.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssmmessages:CreateControlChannel",
                "ssmmessages:CreateDataChannel",
                "ssmmessages:OpenControlChannel",
                "ssmmessages:OpenDataChannel"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetEncryptionConfiguration"
            ],
            "Resource": "*"
        },
        {
```

```
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "key-name"
        }
    ]
}
```

Untuk informasi tentang menggunakan kunci KMS untuk mengenkripsi data sesi, lihat Aktifkan enkripsi kunci KMS data sesi (konsol).

Jika Anda tidak akan menggunakan enkripsi AWS KMS untuk data sesi Anda, Anda dapat menghapus konten berikut dari kebijakan.

```
,
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "key-name"
        }
```

8. Pilih Next: Tags (Selanjutnya: Tanda).
9. (Opsional) Tambahkan tag dengan memilihTambahkan tag, dan memasukkan tag pilihan untuk kebijakan.
10. Pilih Next: Review (Selanjutnya: Tinjauan).
11. Pada halaman Tinjau kebijakan, untuk Nama, masukkan nama untuk kebijakan selaras, seperti **`SessionManagerPermissions`**.
12. (Opsional) Untuk Deskripsi, masukkan deskripsi untuk kebijakan.

    Pilih Buat kebijakan.

Untuk informasi tentang`ssmmessages`tindakan, lihatReferensi: ec2messages, ssmmessages, dan operasi API lainnya.

Buat peran IAM khusus untuk Session Manager

Anda dapat membuat peran AWS Identity and Access Management (IAM) yang memberikan izin untuk melakukan tindakan pada instans Session Manager terkelola Amazon EC2 Anda. Anda juga dapat menyertakan kebijakan untuk memberikan izin yang diperlukan agar log sesi yang akan dikirim ke Amazon Simple Storage Service (Amazon S3) dan Amazon CloudWatch Logs.

Setelah Anda membuat peran IAM, untuk informasi tentang cara melampirkan peran ke instans, lihat Melampirkan atau Mengganti Profil Instans di AWS re:Post situs web. Untuk informasi selengkapnya tentang profil dan peran instans IAM, lihat Menggunakan profil instans di Panduan Pengguna IAM dan peran IAM untuk Amazon EC2 di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux. Untuk informasi lebih lanjut tentang pembuatan peran layanan IAM untuk mesin on-premise, lihat Membuat peran layanan IAM untuk lingkungan hibrid.

Topik

- Membuat peran IAM dengan Session Manager izin (konsol) minimal
- Membuat peran IAM dengan izin untuk Session Manager dan Amazon S3 dan CloudWatch Log (konsol)

Membuat peran IAM dengan Session Manager izin (konsol) minimal

Gunakan prosedur berikut untuk membuat peran IAM kustom dengan kebijakan yang memberikan izin hanya untuk Session Manager tindakan pada instans Anda.

Untuk membuat profil instans dengan Session Manager izin (konsol) minimal

1. Masuk ke AWS Management Console dan buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Dalam panel navigasi, pilih Kebijakan, dan kemudian pilih Buat kebijakan. (Jika tombol Memulai ditampilkan, pilih tombol tersebut, dan kemudian pilih Buat kebijakan.)
3. Pilih tab JSON.
4. Ganti konten default dengan kebijakan berikut. Untuk mengenkripsi data sesi menggunakan AWS Key Management Service (AWS KMS), ganti *nama-kunci* dengan Amazon Resource Name (ARN) dari AWS KMS key yang ingin Anda gunakan.

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
        {
            "Effect": "Allow",
            "Action": [
                "ssm:UpdateInstanceInformation",
                "ssmmessages:CreateControlChannel",
                "ssmmessages:CreateDataChannel",
                "ssmmessages:OpenControlChannel",
                "ssmmessages:OpenDataChannel"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "key-name"
        }
    ]
}
```

Untuk informasi tentang menggunakan kunci KMS untuk mengenkripsi data sesi, lihat Aktifkan enkripsi kunci KMS data sesi (konsol).

Jika Anda tidak akan menggunakan enkripsi AWS KMS untuk data sesi Anda, Anda dapat menghapus konten berikut dari kebijakan.

```
,
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "key-name"
        }
```

5. Pilih Next: Tags (Selanjutnya: Tanda).
6. (Opsional) Tambahkan tag dengan memilih Tambahkan tag, dan masukkan tag pilihan untuk kebijakan.
7. Pilih Next: Review (Selanjutnya: Tinjauan).

8. Pada halaman Tinjau kebijakan, untuk Nama, masukkan nama untuk kebijakan selaras, seperti **`SessionManagerPermissions`**.
9. (Opsional) Untuk Deskripsi, masukkan deskripsi untuk kebijakan.
10. Pilih Buat kebijakan.
11. Di panel navigasi, pilih Peran, lalu pilih Buat peran.
12. Pada halaman Buat peran, pilih AWSlayanan, dan untuk Kasus penggunaan, pilih EC2.
13. Pilih Selanjutnya.
14. Pada halaman Tambahkan izin, pilih kotak centang di sebelah kiri nama kebijakan yang baru saja Anda buat, seperti**`SessionManagerPermissions`**.
15. Pilih Selanjutnya.
16. Pada halaman Nama, tinjau, dan buat, untuk Nama peran IAM, masukkan nama untuk peran IAM, seperti**`MySessionManagerRole`**.
17. (Opsional) Untuk Deskripsi peran, masukkan deskripsi untuk profil instans.
18. (Opsional) Tambahkan tag dengan memilih Tambahkan tag, dan masukkan tag pilihan untuk peran tersebut.

    Pilih Create role (Buat peran).

Untuk informasi tentang `ssmmessages` tindakan, lihatReferensi: ec2messages, ssmmessages, dan operasi API lainnya.

Membuat peran IAM dengan izin untuk Session Manager dan Amazon S3 dan CloudWatch Log (konsol)

Gunakan prosedur berikut untuk membuat peran IAM kustom dengan kebijakan yang memberikan izin untuk Session Manager tindakan pada instans Anda. Kebijakan ini juga memberikan izin yang diperlukan agar log sesi disimpan di bucket Amazon Simple Storage Service (Amazon S3) dan grup CloudWatch log Amazon Logs.

> **Important**
>
> Untuk mengeluarkan log sesi ke bucket Amazon S3 yang dimiliki oleh yang berbedaAkun AWS, Anda harus menambahkan `s3:PutObjectAcl` izin ke kebijakan peran IAM. Selain itu, Anda harus memastikan bahwa kebijakan bucket memberikan akses lintas akun ke peran IAM yang digunakan oleh akun pemilik untuk memberikan izin Systems Manager

untuk instans terkelola. Jika bucket menggunakan enkripsi Key Management Service (KMS), maka kebijakan KMS bucket juga harus memberikan akses lintas akun ini. Untuk informasi selengkapnya tentang mengonfigurasi izin bucket lintas akun di Amazon S3, lihat Memberikan izin bucket lintas akun di Panduan Pengguna Amazon Simple Storage Service. Jika izin lintas akun tidak ditambahkan, akun yang memiliki bucket Amazon S3 tidak dapat mengakses log output sesi.

Untuk informasi tentang menentukan preferensi untuk menyimpan log sesi, lihat Mengaktifkan dan menonaktifkan pencatatan aktivitas sesi.

Untuk membuat peran IAM dengan izin untuk Session Manager dan Amazon S3 dan CloudWatch Log (konsol)

1. Masuk ke AWS Management Console dan buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Dalam panel navigasi, pilih Kebijakan, dan kemudian pilih Buat kebijakan. (Jika tombol Memulai ditampilkan, pilih tombol tersebut, dan kemudian pilih Buat kebijakan.)
3. Pilih tab JSON.
4. Ganti konten default dengan kebijakan berikut. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssmmessages:CreateControlChannel",
                "ssmmessages:CreateDataChannel",
                "ssmmessages:OpenControlChannel",
                "ssmmessages:OpenDataChannel",
                "ssm:UpdateInstanceInformation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
```

```
                "logs:PutLogEvents",
                "logs:DescribeLogGroups",
                "logs:DescribeLogStreams"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:PutObject"
            ],
            "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/s3-bucket-prefix/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetEncryptionConfiguration"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "key-name"
        },
        {
            "Effect": "Allow",
            "Action": "kms:GenerateDataKey",
            "Resource": "*"
        }
    ]
}
```

5. Pilih Next: Tags (Selanjutnya: Tanda).
6. (Opsional) Tambahkan tag dengan memilih Tambahkan tag, dan masukkan tag pilihan untuk kebijakan.
7. Pilih Next: Review (Selanjutnya: Tinjauan).
8. Pada halaman Tinjau kebijakan, untuk Nama, masukkan nama untuk kebijakan selaras, seperti **`SessionManagerPermissions`**.
9. (Opsional) Untuk Deskripsi, masukkan deskripsi untuk kebijakan.

10. Pilih Buat kebijakan.
11. Di panel navigasi, pilih Peran, lalu pilih Buat peran.
12. Pada halaman Buat peran, pilih AWSlayanan, dan untuk Kasus penggunaan, pilih EC2.
13. Pilih Selanjutnya.
14. Pada halaman Tambahkan izin, pilih kotak centang di sebelah kiri nama kebijakan yang baru saja Anda buat, seperti**`SessionManagerPermissions`**.
15. Pilih Selanjutnya.
16. Pada halaman Nama, tinjau, dan buat, untuk Nama peran IAM, masukkan nama untuk peran IAM, seperti**`MySessionManagerRole`**.
17. (Opsional) Untuk Deskripsi peran, masukkan deskripsi untuk peran tersebut.
18. (Opsional) Tambahkan tag dengan memilih Tambahkan tag, dan masukkan tag pilihan untuk peran tersebut.
19. Pilih Create role (Buat peran).

Langkah 3: Kontrol akses sesi ke node yang dikelola

Anda memberikan atau mencabut Session Manager akses ke node terkelola dengan menggunakan kebijakan AWS Identity and Access Management (IAM). Anda dapat membuat kebijakan dan melampirkannya ke pengguna IAM atau grup yang menentukan node terkelola yang dapat disambungkan oleh pengguna atau grup. Anda juga dapat menentukan operasi Session Manager API yang dapat dilakukan pengguna atau grup pada node yang dikelola tersebut.

Untuk membantu Anda memulai dengan kebijakan izin IAMSession Manager, kami telah membuat contoh kebijakan untuk pengguna akhir dan pengguna administrator. Anda dapat menggunakan kebijakan ini hanya dengan perubahan kecil. Atau, gunakan sebagai panduan untuk membuat kebijakan IAM khusus. Untuk informasi selengkapnya, lihat Contoh kebijakan IAM untuk Session Manager. Untuk informasi tentang cara membuat kebijakan IAM dan melampirkannya ke pengguna atau grup, lihat Membuat Kebijakan IAM dan Menambahkan dan Menghapus Kebijakan IAM di Panduan Pengguna IAM.

Tentang format ARN ID sesi

Saat membuat kebijakan IAM untuk Session Manager akses, Anda menentukan ID sesi sebagai bagian dari Nama Sumber Daya Amazon (ARN). ID sesi menyertakan nama pengguna sebagai variabel. Untuk membantu mengilustrasikan hal ini, berikut adalah format Session Manager ARN dan contohnya:

```
arn:aws:ssm:region-id:account-id:session/session-id
```

Sebagai contoh:

```
arn:aws:ssm:us-east-2:123456789012:session/JohnDoe-1a2b3c4d5eEXAMPLE
```

Untuk informasi selengkapnya tentang penggunaan variabel dalam kebijakan IAM, lihat Elemen Kebijakan IAM: Variabel.

Topik

- Mulai sesi shell default dengan menentukan dokumen sesi default dalam kebijakan IAM
- Memulai sesi dengan dokumen dengan menentukan dokumen sesi dalam kebijakan IAM
- Contoh kebijakan IAM untuk Session Manager
- Contoh tambahan kebijakan IAM untuk Session Manager

Mulai sesi shell default dengan menentukan dokumen sesi default dalam kebijakan IAM

Saat Anda mengonfigurasi Session Manager untuk Akun AWS atau ketika Anda mengubah preferensi sesi di konsol Systems Manager, sistem akan membuat dokumen sesi SSM yang disebut`SSM-SessionManagerRunShell`. Ini adalah dokumen sesi default. Session Managermenggunakan dokumen ini untuk menyimpan preferensi sesi Anda, yang mencakup informasi seperti berikut:

- Lokasi tempat Anda ingin menyimpan data sesi, seperti bucket Amazon Simple Storage Service (Amazon S3) atau grup log CloudWatch Amazon Logs.
- ID kunci AWS Key Management Service (AWS KMS) untuk mengenkripsi data sesi.
- Apakah dukungan Run As diizinkan untuk sesi Anda.

Berikut adalah contoh informasi yang terkandung dalam dokumen preferensi `SSM-SessionManagerRunShell` sesi.

```
{
  "schemaVersion": "1.0",
  "description": "Document to hold regional settings for Session Manager",
  "sessionType": "Standard_Stream",
  "inputs": {
```

```
    "s3BucketName": "MyS3TestBucket",
    "s3KeyPrefix": "BucketPrefix",
    "s3EncryptionEnabled": true,
    "cloudWatchLogGroupName": "MyCWLogGroup",
    "cloudWatchEncryptionEnabled": false,
    "kmsKeyId": "1a2b3c4d",
    "runAsEnabled": true,
    "runAsDefaultUser": "RunAsUser"
  }
}
```

Secara default, Session Manager menggunakan dokumen sesi default ketika pengguna memulai sesi dariAWS Management Console. Ini berlaku untuk salah satu Fleet Manager atau Session Manager di konsol Systems Manager, atau EC2 Connect di konsol Amazon EC2. Session Managerjuga menggunakan dokumen sesi default ketika pengguna memulai sesi dengan menggunakan AWS CLI perintah seperti contoh berikut:

```
aws ssm start-session \
    --target i-02573cafcfEXAMPLE
```

Jika Anda ingin pengguna atau grup mengakses sesi shell default, kami sarankan Anda juga menentukan dokumen sesi default dalam kebijakan IAM, seperti yang ditunjukkan pada contoh berikut.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "EnableSSMSession",
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:us-west-2:123456789012:instance/i-02573cafcfEXAMPLE",
                "arn:aws:ssm:us-west-2:123456789012:document/SSM-
SessionManagerRunShell"
            ]
        }
    ]
}
```

Memulai sesi dengan dokumen dengan menentukan dokumen sesi dalam kebijakan IAM

Jika Anda menggunakan AWS CLI perintah start-session menggunakan dokumen sesi default, Anda dapat menghilangkan nama dokumen. Sistem secara otomatis memanggil dokumen `SSM-SessionManagerRunShell` sesi.

Dalam semua kasus lain, Anda harus menentukan nilai untuk `document-name` parameter. Ketika pengguna menentukan nama dokumen sesi dalam sebuah perintah, sistem memeriksa kebijakan IAM mereka untuk memverifikasi bahwa mereka memiliki izin untuk mengakses dokumen. Jika mereka tidak memiliki izin, permintaan koneksi gagal. Contoh berikut mencakup `document-name` parameter dengan dokumen `AWS-StartPortForwardingSession` sesi.

```
aws ssm start-session \
    --target i-02573cafcfEXAMPLE \
    --document-name AWS-StartPortForwardingSession \
    --parameters '{"portNumber":["80"], "localPortNumber":["56789"]}'
```

Menerapkan pemeriksaan izin dokumen sesi saat memulai sesi

Untuk membatasi akses ke dokumen `AWS-StartPortForwardingSession` sesi, Anda dapat menambahkan elemen kondisi ke kebijakan IAM pengguna yang memvalidasi apakah pengguna memiliki akses eksplisit ke dokumen sesi. Ketika kondisi ini diterapkan, pengguna harus menentukan nilai untuk `document-name` opsi `start-session` perintah. Elemen ketentuan berikut, ketika ditambahkan ke tindakan `ssm:StartSession` dalam kebijakan IAM, akan melakukan pemeriksaan akses dokumen sesi.

```
"Condition": {
    "BoolIfExists": {
        "ssm:SessionDocumentAccessCheck": "true"
    }
}
```

Dengan elemen kondisi ini disetel ke`true`, akses eksplisit ke dokumen sesi harus diberikan dalam kebijakan IAM agar pengguna dapat memulai sesi. Untuk memastikan elemen kondisi ditegakkan, elemen tersebut harus disertakan dalam semua pernyataan kebijakan yang memungkinkan `ssm:StartSession` tindakan. Inilah contohnya.

```
{
    "Version": "2012-10-17",
```

```
    "Statement": [
        {
            "Sid": "EnableSSMSession",
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:us-west-2:123456789012:instance/i-02573cafcfEXAMPLE",
                "arn:aws:ssm:us-west-2::document/AWS-StartPortForwardingSession"
            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck": "true"
                }
            }
        }
    ]
}
```

Dengan kebijakan IAM ini, jika elemen `SessionDocumentAccessCheck` kondisi disetel ke`true`, pengguna harus memasukkan `document-name` parameter dalam perintah mereka saat memulai sesi menggunakan. AWS CLI Nilai `document-name` harus dokumen yang ditentukan di `Resource` bagian kebijakan IAM. Jika pengguna memasukkan nama dokumen yang berbeda atau mereka tidak menentukan `document-name` parameter, permintaan gagal.

Jika elemen `SessionDocumentAccessCheck` kondisi disetel ke`false`, itu tidak mempengaruhi evaluasi kebijakan IAM.

Untuk contoh menentukan dokumen Session Manager sesi dalam kebijakan IAM, lihat. Kebijakan pengguna akhir Quickstart untuk Session Manager

Skenario lainnya

Untuk memulai sesi menggunakan SSH, langkah-langkah konfigurasi harus diselesaikan pada node terkelola target dan mesin lokal pengguna. Untuk selengkapnya, lihat (Opsional) Mengizinkan dan mengontrol izin untuk koneksi SSH melalui. Session Manager

Contoh kebijakan IAM untuk Session Manager

Gunakan sampel di bagian ini untuk membantu Anda membuat kebijakan AWS Identity and Access Management (IAM) yang memberikan izin akses yang paling umum diperlukan. Session Manager

> **Note**
>
> Anda juga dapat menggunakan AWS KMS key kebijakan untuk mengontrol entitas IAM (pengguna atau peran) mana dan Akun AWS diberi akses ke kunci KMS Anda. Untuk selengkapnya, lihat Ringkasan Mengelola Akses ke AWS KMS Sumber Daya Anda dan Menggunakan Kebijakan Utama AWS KMS di Panduan AWS Key Management Service Pengembang.

Topik

- Kebijakan pengguna akhir Quickstart untuk Session Manager
- Kebijakan administrator Quickstart untuk Session Manager

Kebijakan pengguna akhir Quickstart untuk Session Manager

Gunakan contoh berikut untuk membuat kebijakan pengguna akhir IAM. Session Manager

Anda dapat membuat kebijakan yang memungkinkan pengguna memulai sesi hanya dari Session Manager konsol dan AWS Command Line Interface (AWS CLI), hanya dari konsol Amazon Elastic Compute Cloud (Amazon EC2), atau dari ketiganya.

Kebijakan ini memberi pengguna akhir kemampuan untuk memulai sesi ke node terkelola tertentu dan kemampuan untuk mengakhiri hanya sesi mereka sendiri. Lihat Contoh tambahan kebijakan IAM untuk Session Manager untuk contoh kustomisasi yang mungkin ingin Anda buat pada kebijakan.

Dalam contoh kebijakan berikut, ganti setiap *placeholder sumber daya contoh dengan informasi* Anda sendiri.

Lihat bagian berikut untuk melihat kebijakan sampel untuk berbagai akses sesi yang ingin Anda berikan.

Manajer Sesi and CLI

Gunakan kebijakan sampel ini untuk memberi pengguna kemampuan untuk memulai dan melanjutkan sesi hanya dari Session Manager konsol dan AWS CLI. Kebijakan ini tidak memberikan semua izin yang diperlukan untuk memulai sesi dari konsol Amazon EC2.

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:region:account-id:instance/instance-id",
                "arn:aws:ssm:region:account-id:document/SSM-
SessionManagerRunShell" (1)
            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck":
 "true" (2)
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeSessions",
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceProperties",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeInstances"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [

 "kms:GenerateDataKey" (3)
```

```
            ],
            "Resource": "key-name"
        }
    ]
}
```

Amazon EC2

Gunakan kebijakan contoh ini untuk memberi pengguna kemampuan untuk memulai dan melanjutkan sesi hanya dari konsol Amazon EC2. Kebijakan ini tidak menyediakan semua izin yang diperlukan untuk memulai sesi dari Session Manager konsol dan. AWS CLI

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession",

 "ssm:SendCommand" 4
            ],
            "Resource": [
                "arn:aws:ec2:region:account-id:instance/instance-id",
                "arn:aws:ssm:region:account-id:document/SSM-
SessionManagerRunShell" 1
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceInformation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
```

```
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        }
    ]
}
```

Manajer Sesi, CLI, and Amazon EC2

Gunakan kebijakan contoh ini untuk memberi pengguna kemampuan memulai dan melanjutkan sesi dari Session Manager konsol, konsol AWS CLI, dan konsol Amazon EC2.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession",

 "ssm:SendCommand" 4
            ],
            "Resource": [
                "arn:aws:ec2:region:account-id:instance/instance-id",
                "arn:aws:ssm:region:account-id:document/SSM-
SessionManagerRunShell" 1
            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck":
 "true" 2
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeSessions",
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceInformation",
                "ssm:DescribeInstanceProperties",
                "ec2:DescribeInstances"
            ],
            "Resource": "*"
```

```
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [

 "kms:GenerateDataKey" (3)
            ],
            "Resource": "key-name"
        }
    ]
}
```

[1] `SSM-SessionManagerRunShell` adalah nama default dari dokumen SSM yang Session Manager dibuat untuk menyimpan preferensi konfigurasi sesi Anda. Anda dapat membuat dokumen Sesi khusus dan menetapkannya dalam kebijakan ini sebagai gantinya. Anda juga dapat menentukan dokumen AWS yang disediakan `AWS-StartSSHSession` untuk pengguna yang memulai sesi menggunakan SSH. Untuk informasi tentang langkah-langkah konfigurasi yang diperlukan untuk mendukung sesi menggunakan SSH, lihat (Opsional) Mengizinkan dan mengontrol izin untuk koneksi SSH melalui. Session Manager

[2] Jika Anda menentukan elemen ketentuan, `ssm:SessionDocumentAccessCheck`, seperti `true`, sistem memeriksa bahwa pengguna memiliki akses secara eksplisit ke dokumen Sesi yang ditentukan, dalam `SSM-SessionManagerRunShell` contoh ini, sebelum sesi dibuat. Untuk informasi selengkapnya, lihat Menerapkan pemeriksaan izin dokumen sesi saat memulai sesi.

[3] Izin `kms:GenerateDataKey` memungkinkan pembuatan kunci enkripsi data yang akan digunakan untuk mengenkripsi data sesi. Jika Anda akan menggunakan enkripsi AWS Key Management Service (AWS KMS) untuk data sesi Anda, ganti nama *`kunci dengan Nama`* Sumber Daya Amazon (ARN) dari kunci KMS yang ingin Anda gunakan, dalam format. `arn:aws:kms:us-`

`west-2:111122223333:key/1234abcd-12ab-34cd-56ef-12345EXAMPLE` Jika Anda tidak akan menggunakan enkripsi kunci KMS untuk data sesi Anda, hapus konten berikut dari kebijakan.

```
,
        {
            "Effect": "Allow",
            "Action": [
                "kms:GenerateDataKey"
            ],
            "Resource": "key-name"
        }
```

Untuk informasi tentang penggunaan AWS KMS untuk mengenkripsi data sesi, lihat. Aktifkan enkripsi kunci KMS data sesi (konsol)

[4] Izin untuk SendCommanddiperlukan untuk kasus di mana pengguna mencoba memulai sesi dari konsol Amazon EC2, tetapi perintah harus dikirim untuk memperbarui SSM Agent terlebih dahulu.

Kebijakan administrator Quickstart untuk Session Manager

Gunakan contoh berikut untuk membuat kebijakan administrator IAM untukSession Manager.

Kebijakan ini memberi administrator kemampuan untuk memulai sesi ke node terkelola yang ditandai dengan`Key=Finance,Value=WebServers`, izin untuk membuat, memperbarui, dan menghapus preferensi, dan izin untuk mengakhiri hanya sesi mereka sendiri. Lihat Contoh tambahan kebijakan IAM untuk Session Manager untuk contoh kustomisasi yang mungkin ingin Anda buat pada kebijakan.

Anda dapat membuat kebijakan yang memungkinkan administrator menjalankan tugas ini hanya dari Session Manager konsol dan AWS CLI, hanya dari konsol Amazon EC2, atau dari ketiganya.

Dalam contoh kebijakan berikut, ganti setiap *placeholder sumber daya contoh dengan informasi* Anda sendiri.

Lihat bagian berikut untuk melihat kebijakan sampel untuk tiga skenario izin.

Manajer Sesi and CLI

> Gunakan kebijakan contoh ini untuk memberi administrator kemampuan untuk melakukan tugas terkait sesi hanya dari Session Manager konsol dan perangkat. AWS CLI Kebijakan ini tidak

memberikan semua izin yang diperlukan untuk melakukan tugas terkait sesi dari konsol Amazon EC2.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:region:account-id:instance/*"
            ],
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/Finance": [
                        "WebServers"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeSessions",
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceProperties",
                "ec2:DescribeInstances"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:CreateDocument",
                "ssm:UpdateDocument",
                "ssm:GetDocument"
            ],
            "Resource": "arn:aws:ssm:region:account-id:document/SSM-
SessionManagerRunShell"
        },
        {
```

```
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        }
    ]
}
```

Amazon EC2

Gunakan kebijakan contoh ini untuk memberi administrator kemampuan untuk melakukan tugas terkait sesi hanya dari konsol Amazon EC2. Kebijakan ini tidak menyediakan semua izin yang diperlukan untuk melakukan tugas terkait sesi dari Session Manager konsol dan. AWS CLI

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession",

 "ssm:SendCommand" 1
            ],
            "Resource": [
                "arn:aws:ec2:region:account-id:instance/*"
            ],
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/tag-key": [
                        "tag-value"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetConnectionStatus",
```

```
                "ssm:DescribeInstanceInformation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        }
    ]
}
```

Manajer Sesi, CLI, and Amazon EC2

Gunakan kebijakan contoh ini untuk memberi administrator kemampuan untuk melakukan tugas terkait sesi dari Session Manager konsol, konsol AWS CLI, dan konsol Amazon EC2.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession",

 "ssm:SendCommand" (1)
            ],
            "Resource": [
                "arn:aws:ec2:region:account-id:instance/*"
            ],
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/tag-key": [
                        "tag-value"
                    ]
                }
            }
        },
```

```
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeSessions",
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceInformation",
                "ssm:DescribeInstanceProperties",
                "ec2:DescribeInstances"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:CreateDocument",
                "ssm:UpdateDocument",
                "ssm:GetDocument"
            ],
            "Resource": "arn:aws:ssm:region:account-id:document/SSM-
SessionManagerRunShell"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        }
    ]
}
```

[1] Izin untuk SendCommanddiperlukan untuk kasus di mana pengguna mencoba memulai sesi dari konsol Amazon EC2, tetapi perintah harus dikirim untuk memperbarui SSM Agent terlebih dahulu.

Contoh tambahan kebijakan IAM untuk Session Manager

Lihat contoh kebijakan berikut untuk membantu Anda membuat kebijakan kustom AWS Identity and Access Management (IAM) untuk setiap skenario akses Session Manager pengguna yang ingin Anda dukung.

Topik

Contoh 1: Berikan akses ke dokumen di konsol

Anda dapat mengizinkan pengguna menentukan dokumen kustom saat mereka meluncurkan sesi menggunakan konsol Pengelola Sesi. Contoh berikut kebijakan IAM memberikan izin untuk mengakses dokumen dengan nama yang dimulai dengan **`SessionDocument-`** yang ditentukan Wilayah AWS dan. Akun AWS

Untuk menggunakan kebijakan ini, ganti setiap *`placeholder sumber daya contoh dengan informasi`* Anda sendiri.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetDocument",
                "ssm:ListDocuments"
            ],
            "Resource": [
                "arn:aws:ssm:region:account-id:document/SessionDocument-*"

            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck": "true"
                }
            }
        }
    ]
}
```

> **Note**
>
> Konsol Session Manager hanya mendukung dokumen Session `sessionType` yang memiliki `Standard_Stream` yang digunakan untuk menentukan preferensi sesi. Untuk informasi selengkapnya, lihat Skema dokumen sesi.

Contoh 2: Batasi akses ke node terkelola tertentu

Anda dapat membuat kebijakan IAM yang menentukan node terkelola mana yang diizinkan untuk disambungkan oleh pengguna menggunakan Session Manager. Misalnya, kebijakan berikut memberi pengguna izin untuk memulai, mengakhiri, dan melanjutkan sesi mereka pada tiga node tertentu. Kebijakan membatasi pengguna untuk menyambung ke node selain yang ditentukan.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-2:123456789012:instance/i-1234567890EXAMPLE",
                "arn:aws:ec2:us-east-2:123456789012:instance/i-abcdefghijEXAMPLE",
                "arn:aws:ec2:us-east-2:123456789012:instance/i-0e9d8c7b6aEXAMPLE"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        }
    ]
}
```

Contoh 3: Batasi akses berdasarkan tag

Anda dapat membatasi akses ke node terkelola berdasarkan tag tertentu. Dalam contoh berikut, pengguna diizinkan untuk memulai dan melanjutkan sesi (`Effect: Allow, Action: ssm:StartSession, ssm:ResumeSession`) pada setiap node terkelola (`Resource: arn:aws:ec2:`*region*`:`*987654321098*`:instance/*`) dengan syarat bahwa node tersebut adalah Finance WebServer (`ssm:resourceTag/Finance: WebServer`). Jika pengguna mengirim perintah ke node terkelola yang tidak diberi tag atau yang memiliki tag selain`Finance: WebServer`, hasil perintah akan disertakan`AccessDenied`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-2:123456789012:instance/*"
            ],
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/Finance": [
                        "WebServers"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        }
    ]
}
```

Anda dapat membuat kebijakan IAM yang memungkinkan pengguna memulai sesi ke node terkelola yang ditandai dengan beberapa tag. Kebijakan berikut memungkinkan pengguna untuk memulai sesi ke node terkelola yang memiliki kedua tag yang ditentukan diterapkan padanya. Jika pengguna mengirim perintah ke node terkelola yang tidak ditandai dengan kedua tag ini, hasil perintah akan disertakan`AccessDenied`.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "ssm:StartSession"
         ],
         "Resource":"*",
         "Condition":{
            "StringLike":{
               "ssm:resourceTag/tag-key1":[
                  "tag-value1"
               ],
               "ssm:resourceTag/tag-key2":[
                  "tag-value2"
               ]
            }
         }
      }
   ]
}
```

Untuk informasi selengkapnya tentang membuat kebijakan IAM, lihat Kebijakan Terkelola dan Kebijakan Inline di Panduan Pengguna IAM. Untuk informasi selengkapnya tentang menandai node terkelola, lihat Menandai node terkelola dan Menandai resource Amazon EC2 Anda di Panduan Pengguna Amazon EC2 untuk Instans Linux (konten berlaku dan node terkelola). Windows Linux Untuk informasi selengkapnya tentang meningkatkan postur keamanan terhadap perintah tingkat root yang tidak sah pada node terkelola, lihat Membatasi akses ke perintah tingkat root melaluiSSM Agent

Contoh 4: Izinkan pengguna untuk mengakhiri hanya sesi yang mereka mulai

Session Managermenyediakan dua metode untuk mengontrol sesi mana pengguna federasi di Anda Akun AWS diizinkan untuk mengakhiri.

- Gunakan variabel `{aws:userid}` dalam kebijakan izin AWS Identity and Access Management (IAM). Pengguna federasi hanya dapat mengakhiri sesi yang mereka mulai. Untuk pengguna yang tidak terfederasi, gunakan variabel `{aws:username}` sebagai ganti. `{aws:userid}`
- Gunakan tag yang disediakan oleh AWS tag dalam kebijakan izin IAM. Dalam kebijakan, Anda memasukkan ketentuan yang mengizinkan pengguna untuk mengakhiri hanya sesi yang ditandai dengan tanda tertentu yang telah diberikan oleh AWS. Metode ini bekerja untuk semua akun, termasuk yang menggunakan ID federasi untuk memberikan akses ke AWS.

Metode 1: Berikan TerminateSession hak istimewa menggunakan variabel **`{aws:username}`**

Kebijakan IAM berikut mengizinkan pengguna untuk melihat ID semua sesi di akun Anda. Namun, pengguna dapat berinteraksi dengan node terkelola hanya melalui sesi yang mereka mulai. Pengguna yang ditetapkan kebijakan berikut tidak dapat terhubung ke atau mengakhiri sesi pengguna lain. Kebijakan menggunakan `{aws:username}` variabel untuk mencapai hal ini.

> **Note**
>
> Metode ini tidak bekerja untuk akun yang memberikan akses ke AWS menggunakan ID federasi.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "ssm:DescribeSessions"
            ],
            "Effect": "Allow",
            "Resource": [
                "*"
            ]
        },
        {
            "Action": [
                "ssm:TerminateSession"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:username}-*"
```

```
            ]
        }
    ]
}
```

Metode 2: Berikan TerminateSession hak istimewa menggunakan tag yang disediakan oleh AWS

Anda dapat mengontrol sesi mana yang dapat diakhiri pengguna dengan menyertakan variabel kunci tag bersyarat dalam kebijakan IAM. Ketentuan menetapkan bahwa pengguna hanya dapat mengakhiri sesi yang ditandai dengan salah satu atau kedua variabel kunci tanda tertentu dan nilai tertentu ini.

Saat pengguna Akun AWS memulai sesi, Session Manager terapkan dua tag sumber daya ke sesi tersebut. Tanda sumber daya pertama adalah `aws:ssmmessages:target-id`, yang dengannya Anda menentukan ID target yang diizinkan untuk diakhiri oleh pengguna. Tag sumber daya lainnya adalah `aws:ssmmessages:session-id`, dengan nilai dalam format *`role-id`*`:`*`caller-specified-role-name`*.

> **Note**
>
> Session Managertidak mendukung tag khusus untuk kebijakan kontrol akses IAM ini. Anda harus menggunakan tag sumber daya yang disediakan oleh AWS, dijelaskan di bawah ini.

`aws:ssmmessages:target-id`

Dengan kunci tag ini, Anda menyertakan ID node terkelola sebagai nilai dalam kebijakan. Dalam blok kebijakan berikut, pernyataan kondisi memungkinkan pengguna untuk mengakhiri hanya node I-02573CAFCFExample.

```
{
     "Version": "2012-10-17",
     "Statement": [
         {
             "Effect": "Allow",
             "Action": [
                "ssm:TerminateSession"
             ],
             "Resource": "*",
             "Condition": {
```

```
                "StringLike": {
                    "ssm:resourceTag/aws:ssmmessages:target-id": [
                       "i-02573cafcfEXAMPLE"
                    ]
                }
            }
        }
    ]
}
```

Jika pengguna mencoba untuk mengakhiri sesi yang belum diberi izin `TerminateSession` ini, mereka menerima kesalahan `AccessDeniedException`.

`aws:ssmmessages:session-id`

Kunci tanda ini mencakup variabel untuk ID sesi sebagai nilai dalam permintaan untuk memulai sesi.

Contoh berikut menunjukkan kebijakan untuk kasus di mana tipe pemanggil adalah `User`. Nilai yang Anda sediakan untuk `aws:ssmmessages:session-id` adalah ID pengguna. Dalam contoh ini, `AIDIODR4TAW7CSEXAMPLE` mewakili ID pengguna di Akun AWS Anda. Untuk mengambil ID untuk pengguna di Anda Akun AWS, gunakan perintah IAM,. `get-user` Untuk selengkapnya, lihat mendapatkan pengguna di AWS Identity and Access Management bagian Panduan Pengguna IAM.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
               "ssm:TerminateSession"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/aws:ssmmessages:session-id": [
                       "AIDIODR4TAW7CSEXAMPLE"
                    ]
                }
            }
        }
    ]
```

```
}
```

Contoh berikut menunjukkan kebijakan untuk kasus di mana tipe pemanggil adalah `AssumedRole`. Anda dapat menggunakan variabel `{aws:userid}` untuk nilai yang Anda sediakan untuk `aws:ssmmessages:session-id`. Atau, Anda dapat hardcode ID peran untuk nilai yang Anda sediakan untuk `aws:ssmmessages:session-id`. Jika Anda hardcode ID peran, Anda harus memberi nilai dalam format *`role-id`*`:`*`caller-specified-role-name`*. Sebagai contoh, `AIDIODR4TAW7CSEXAMPLE:MyRole`.

> **Important**
>
> Agar tanda sistem diterapkan, ID peran yang Anda sediakan hanya dapat berisi karakter berikut: huruf Unicode, 0-9, spasi, _, ., :, /, =, +, -, @, dan \.

Untuk mengambil ID peran untuk peran dalam Anda Akun AWS, gunakan `get-caller-identity` perintah. Untuk informasi, lihat get-caller-identitydi Referensi AWS CLI Perintah.

```
{
     "Version": "2012-10-17",
     "Statement": [
         {
             "Effect": "Allow",
             "Action": [
                "ssm:TerminateSession"
             ],
             "Resource": "*",
             "Condition": {
                 "StringLike": {
                     "ssm:resourceTag/aws:ssmmessages:session-id": [
                        "${aws:userid}*"
                     ]
                 }
             }
         }
     ]
}
```

Jika pengguna mencoba mengakhiri sesi yang belum diberi izin `TerminateSession` ini, mereka menerima kesalahan `AccessDeniedException`.

`aws:ssmmessages:target-id` dan **`aws:ssmmessages:session-id`**

Anda juga dapat membuat kebijakan IAM yang mengizinkan pengguna untuk mengakhiri sesi yang ditandai dengan kedua tanda sistem, seperti yang ditunjukkan dalam contoh ini.

```
{
   "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}*"
               ]
            }
         }
      }
   ]
}
```

Contoh 5: Izinkan akses penuh (administratif) ke semua sesi

Kebijakan IAM berikut memungkinkan pengguna untuk sepenuhnya berinteraksi dengan semua node terkelola dan semua sesi yang dibuat oleh semua pengguna untuk semua node. Ini harus diberikan hanya kepada Administrator yang membutuhkan kontrol penuh atas Session Manager aktivitas organisasi Anda.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "ssm:StartSession",
```

```
                "ssm:TerminateSession",
                "ssm:ResumeSession",
                "ssm:DescribeSessions",
                "ssm:GetConnectionStatus"
            ],
            "Effect": "Allow",
            "Resource": [
                "*"
            ]
        }
    ]
}
```

Langkah 4: Konfigurasi preferensi sesi

Pengguna yang telah diberi izin administratif dalam kebijakanAWS Identity and Access Management (IAM) mereka dapat mengonfigurasi preferensi sesi, termasuk yang berikut ini:

- Aktifkan dukungan Run As untuk nodeLinux terkelola. Hal ini memungkinkan untuk memulai sesi menggunakan kredensial pengguna sistem operasi tertentu alih-alih kredensial sistem yang dihasilkan`ssm-user` akun yangAWS Systems ManagerSession Manager dapat dibuat pada node terkelola.
- KonfigurasiSession Manager untuk menggunakanAWS KMS key enkripsi untuk memberikan perlindungan tambahan terhadap data yang dikirimkan antara mesin klien dan node terkelola.
- KonfigurasiSession Manager untuk membuat dan mengirim log riwayat sesi ke bucket Amazon Simple Storage Service (Amazon S3) atau grup CloudWatch log Amazon Logs. Data log yang disimpan kemudian dapat digunakan untuk mengaudit atau melaporkan koneksi sesi yang dibuat untuk node terkelola Anda dan pelaporan selama sesi.
- Konfigurasi batas waktu sesi. Anda dapat menggunakan pengaturan ini untuk menentukan kapan untuk mengakhiri sesi setelah periode tidak aktif.
- KonfigurasiSession Manager untuk menggunakan profil shell yang dapat dikonfigurasi. Profil yang dapat dikustomisasi ini mengizinkan Anda untuk menentukan preferensi dalam sesi seperti preferensi shell, variabel lingkungan, direktori kerja, dan menjalankan beberapa perintah ketika sesi dimulai.

Untuk informasi selengkapnya tentang izin yang diperlukan untuk mengonfigurasiSession Manager preferensi, lihatthe section called “Berikan atau Tolak izin pengguna untuk memperbarui atau Tolak izin penggunaSession Managerpilihan”.

Topik

- Berikan atau Tolak izin pengguna untuk memperbarui atau Tolak izin penggunaSession Managerpilihan
- Tentukan nilai waktu habis sesi idle
- Tentukan durasi sesi maksimum
- Izinkan profil shell yang dapat dikonfigurasi
- Aktifkan dukungan Run As untuk Linux dan node macOS terkelola
- Aktifkan enkripsi kunci KMS data sesi (konsol)
- Membuat dokumenSession Manager preferensi (baris perintah)
- PerbaruiSession Manager preferensi (baris perintah)

Untuk informasi tentang menggunakan konsol Systems Manager untuk mengkonfigurasi opsi data sesi log, lihat topik berikut:

- Log data sesi menggunakan Amazon S3 (konsol)
- Streaming data sesi menggunakan Amazon CloudWatch Logs (konsol)
- Data sesi logging menggunakan Amazon CloudWatch Logs (konsol)

Berikan atau Tolak izin pengguna untuk memperbarui atau Tolak izin penggunaSession Managerpilihan

Preferensi akun disimpan sebagai dokumen AWS Systems Manager (SSM) untuk setiap Wilayah AWS. Sebelum pengguna dapat memperbarui preferensi akun untuk sesi di akun Anda, mereka harus diberikan izin yang diperlukan untuk mengakses tipe dokumen SSM tempat preferensi ini disimpan. Izin ini diberikan melalui kebijakan AWS Identity and Access Management (IAM) .

Kebijakan administrator untuk mengizinkan preferensi akan dibuat dan diperbarui

Administrator dapat memiliki kebijakan berikut untuk membuat dan memperbarui preferensi kapan saja. Kebijakan berikut mengizinkan izin untuk mengakses dan memperbarui dokumen `SSM-SessionManagerRunShell` dalam akun us-east-2 123456789012.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
```

```
            "Action": [
                "ssm:CreateDocument",
                "ssm:GetDocument",
                "ssm:UpdateDocument",
                "ssm:DeleteDocument"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:ssm:us-east-2:123456789012:document/SSM-
SessionManagerRunShell"
            ]
        }
    ]
}
```

Kebijakan pengguna untuk mencegah preferensi diperbarui

Gunakan kebijakan berikut untuk mencegah pengguna akhir di akun Anda memperbarui atau menimpa apa punSession Managerpreferensi.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "ssm:CreateDocument",
                "ssm:GetDocument",
                "ssm:UpdateDocument",
                "ssm:DeleteDocument"
            ],
            "Effect": "Deny",
            "Resource": [
                "arn:aws:ssm:us-east-2:123456789012:document/SSM-
SessionManagerRunShell"
            ]
        }
    ]
}
```

Tentukan nilai waktu habis sesi idle

Session Manager, suatu kemampuanAWS Systems Manager, mengizinkan Anda untuk menentukan jumlah waktu untuk mengizinkan pengguna menjadi tidak aktif sebelum sistem mengakhiri sesi.

Secara default, waktu habis sesi setelah 20 menit tidak aktif. Anda dapat memodifikasi pengaturan ini untuk menentukan bahwa waktu habis sesi antara 1 dan 60 menit tidak aktif. Beberapa agen keamanan komputasi profesional merekomendasikan pengaturan waktu tunggu sesi idle hingga maksimum 15 menit.

Untuk mengizinkan waktu habis sesi idle (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Session Manager.
3. Pilih tab Preferensi, dan kemudian pilih Edit.
4. Tentukan jumlah waktu untuk mengizinkan pengguna menjadi tidak aktif sebelum sesi berakhir di bidang menit di bawah Waktu habis sesi idle.
5. Pilih Simpan.

Tentukan durasi sesi maksimum

Session Manager, kemampuanAWS Systems Manager, memungkinkan Anda untuk menentukan durasi maksimum sesi sebelum berakhir. Secara default, sesi tidak memiliki durasi maksimum. Nilai yang Anda tentukan untuk durasi sesi maksimum harus antara 1 dan 1.440 menit.

Untuk menentukan durasi sesi maksimum (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Session Manager.
3. Pilih tab Preferensi, dan kemudian pilih Edit.
4. Pilih kotak centang di samping Aktifkan durasi sesi maksimum.
5. Tentukan durasi maksimum sesi sebelum berakhir di bidang menit di bawah Durasi sesi maksimum.
6. Pilih Simpan.

Izinkan profil shell yang dapat dikonfigurasi

Secara default, sesi pada instans EC2 untukLinux mulai menggunakan shell Bourne (sh). Namun, Anda mungkin lebih suka menggunakan shell lain seperti bash. Dengan mengizinkan profil shell yang dapat dikonfigurasi, Anda dapat mengustomisasikan preferensi dalam sesi seperti preferensi shell, variabel lingkungan, direktori kerja, dan menjalankan beberapa perintah ketika sesi dimulai.

Important

Systems Manager tidak memeriksa perintah atau skrip di profil shell Anda untuk melihat perubahan apa yang akan dibuat untuk instans sebelum dijalankan. Untuk membatasi kemampuan pengguna memodifikasi perintah atau skrip yang dimasukkan dalam profil shell mereka, kami merekomendasikan hal berikut:

- Buat dokumen tipe sesi yang dikustomisasi untuk pengguna dan peran AWS Identity and Access Management (IAM) . Kemudian modifikasi kebijakan IAM untuk pengguna dan peran ini sehingga operasi API `StartSession` hanya dapat menggunakan dokumen tipe Sesi yang telah Anda buat untuk mereka. Untuk informasi, lihat Membuat dokumenSession Manager preferensi (baris perintah) dan Kebijakan pengguna akhir Quickstart untuk Session Manager.
- Modifikasi kebijakan IAM untuk pengguna dan peran IAM Anda untuk menolak akses ke operasi API `UpdateDocument` untuk sumber dokumen tipe Sesi yang Anda buat. Hal ini mengizinkan pengguna dan peran Anda untuk menggunakan dokumen yang Anda buat untuk preferensi sesi mereka tanpa mengizinkan mereka untuk memodifikasi salah satu pengaturan.

Untuk mengaktifkan profil shell yang dapat dikonfigurasi

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Session Manager.
3. Pilih tab Preferensi, dan kemudian pilih Edit.
4. Tentukan variabel lingkungan, preferensi shell, atau perintah yang ingin Anda jalankan ketika sesi Anda dimulai di bidang untuk sistem operasi yang berlaku.
5. Pilih Save (Simpan).

Berikut ini adalah beberapa perintah contoh yang dapat ditambahkan ke profil shell Anda.

Ubah ke shell bash dan ubah ke direktori /usr padaLinux instans.

```
exec /bin/bash
cd /usr
```

Keluarkan stempel waktu dan pesan selamat datang pada awal sesi.

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."
```

Lihat aktivitas sistem dinamis pada awal sesi.

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 "-------  ------    -----      ----- -----   ------     -- -----------"}
```

Aktifkan dukungan Run As untuk Linux dan node macOS terkelola

Secara default, Session Manager mengautentikasi koneksi menggunakan kredensil `ssm-user` akun yang dihasilkan sistem yang dibuat pada node terkelola. (Di Linux dan mesin macOS, akun ini ditambahkan ke `/etc/sudoers/`.) Jika Anda memilih, Anda dapat mengautentikasi sesi menggunakan kredensi akun pengguna sistem operasi (OS). Dalam hal ini, Session Manager

memverifikasi bahwa akun OS yang Anda tentukan ada di node sebelum memulai sesi. Jika Anda mencoba memulai sesi menggunakan akun OS yang tidak ada di node, koneksi gagal.

> **Note**
>
> Session Manager tidak mendukung penggunaan akun `root` pengguna sistem operasi untuk mengautentikasi koneksi. Untuk sesi yang diautentikasi menggunakan akun pengguna OS, kebijakan tingkat OS dan direktori node, seperti pembatasan login atau pembatasan penggunaan sumber daya sistem, mungkin tidak berlaku.

Cara kerjanya

Jika Anda mengaktifkan dukungan Run As untuk sesi, sistem memeriksa izin akses sebagai berikut:

1. Untuk pengguna yang memulai sesi, apakah entitas IAM mereka (pengguna atau peran) telah ditandai? `SSMSessionRunAs = os user account name`

   Jika Ya, apakah nama pengguna OS ada di node terkelola? Jika ya, mulai sesi. Jika tidak, jangan izinkan sesi dimulai.

   Jika entitas IAM belum diberi tag`SSMSessionRunAs = os user account name`, lanjutkan ke langkah 2.
2. Jika entitas IAM belum diberi tag`SSMSessionRunAs = os user account name`, apakah nama pengguna OS telah ditentukan dalam preferensiAkun AWS? Session Manager

   Jika Ya, apakah nama pengguna OS ada di node terkelola? Jika ya, mulai sesi. Jika tidak, jangan izinkan sesi dimulai.

> **Note**
>
> Saat Anda mengaktifkan dukungan Run As, ini mencegah Session Manager memulai sesi menggunakan `ssm-user` akun pada node terkelola. Ini berarti bahwa jika Session Manager gagal terhubung menggunakan akun pengguna OS yang ditentukan, itu tidak akan kembali ke koneksi menggunakan metode default.
> Jika Anda mengaktifkan Run As tanpa menentukan akun OS atau menandai entitas IAM, dan Anda belum menentukan akun OS dalam preferensi Session Manager, upaya koneksi sesi akan gagal.

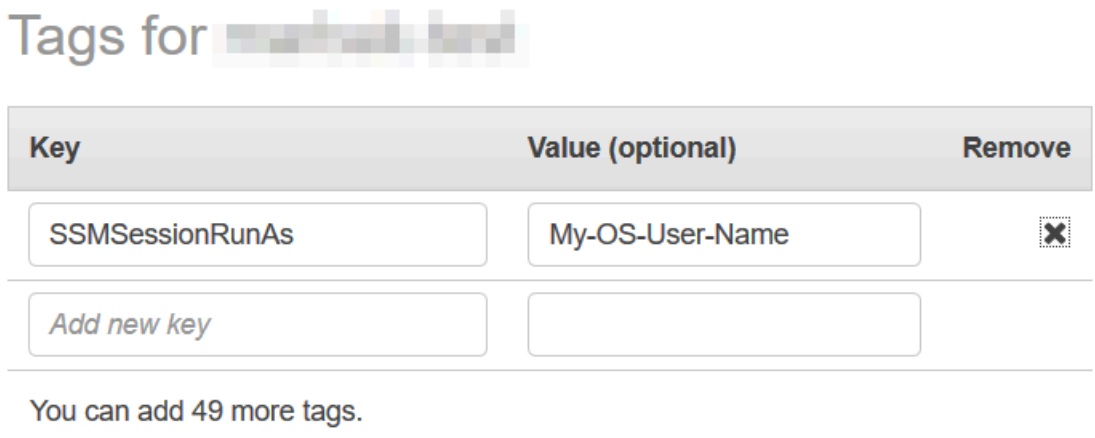
Untuk mengaktifkan dukungan Run As untuk Linux dan node macOS terkelola

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Session Manager.
3. Pilih tab Preferensi, dan kemudian pilih Edit.
4. Pilih kotak centang di samping Aktifkan dukungan Run As untuk Linux instance.
5. Lakukan salah satu dari cara berikut:

   - Opsi 1: Di bidang Nama pengguna sistem operasi, masukkan nama akun pengguna OS yang ingin Anda gunakan untuk memulai sesi. Menggunakan opsi ini, semua sesi dijalankan oleh pengguna OS yang sama untuk semua pengguna di Anda Akun AWS yang terhubung menggunakanSession Manager.
   - Opsi 2 (Disarankan): Pilih tautan konsol IAM. Di panel navigasi, pilih Pengguna atau Peran. Pilih entitas (pengguna atau peran) untuk menambahkan tanda, dan kemudian pilih tab Tanda. Masukkan `SSMSessionRunAs` untuk nama kunci. Masukkan nama akun pengguna OS untuk nilai kunci. Pilih Simpan perubahan.

     Dengan menggunakan opsi ini, Anda dapat menentukan pengguna OS unik untuk entitas IAM yang berbeda jika Anda memilih. Untuk informasi selengkapnya tentang menandai entitas IAM (pengguna atau peran), lihat Menandai sumber daya IAM di Panduan Pengguna IAM

     Berikut sebuah contoh.



6. Pilih Simpan.

Aktifkan enkripsi kunci KMS data sesi (konsol)

Gunakan AWS Key Management Service (AWS KMS) untuk membuat dan mengelola kunci enkripsi. DenganAWS KMS, Anda dapat mengontrol penggunaan enkripsi di berbagai Layanan AWS dan dalam aplikasi Anda. Anda dapat menentukan bahwa data sesi yang ditransmisikan antara node terkelola Anda dan mesin lokal pengguna di Anda Akun AWS dienkripsi menggunakan enkripsi kunci KMS. (Ini adalah tambahan untuk enkripsi TLS 1.2 yang telah diberikan oleh AWS secara default.) Untuk mengenkripsi data Session Manager sesi, buat kunci KMS simetris menggunakan. AWS KMS

AWS KMSenkripsi tersedia untuk`Standard_Stream`,`InteractiveCommands`, dan jenis `NonInteractiveCommands` sesi. Untuk menggunakan opsi untuk mengenkripsi data sesi menggunakan kunci yang dibuatAWS KMS, versi 2.3.539.0 atau yang lebih baru AWS Systems Manager SSM Agent harus diinstal pada node terkelola.

> **Note**
>
> Anda harus mengizinkan AWS KMS enkripsi untuk mengatur ulang kata sandi pada node terkelola Anda dari AWS Systems Manager konsol. Untuk informasi selengkapnya, lihat Setel ulang kata sandi pada node terkelola.

Anda dapat menggunakan kunci yang Anda buat di Akun AWS Anda. Anda juga dapat menggunakan kunci yang dibuat dalam Akun AWS yang berbeda. Pembuat kunci dalam Akun AWS yang berbeda harus memberikan Anda izin yang diperlukan untuk menggunakan kunci.

Setelah Anda mengaktifkan enkripsi kunci KMS untuk data sesi Anda, baik pengguna yang memulai sesi dan node terkelola yang mereka sambungkan harus memiliki izin untuk menggunakan kunci tersebut. Anda memberikan izin untuk menggunakan kunci KMS dengan kebijakan Session Manager through AWS Identity and Access Management (IAM). Untuk informasi lebih lanjut, lihat topik berikut:

- Tambahkan izin AWS KMS untuk pengguna di akun Anda: Contoh kebijakan IAM untuk Session Manager.
- Tambahkan AWS KMS izin untuk node terkelola di akun Anda:Langkah 2: Verifikasi atau tambahkan izin instans untuk Session Manager.

Untuk informasi lebih lanjut tentang membuat dan mengelola kunci KMS, lihat Panduan Developer AWS Key Management Service.

Untuk informasi tentang penggunaan AWS CLI untuk mengaktifkan enkripsi kunci KMS data sesi di akun Anda, lihat Membuat dokumenSession Manager preferensi (baris perintah) atau PerbaruiSession Manager preferensi (baris perintah).

> **Note**
>
> Ada biaya untuk menggunakan kunci KMS. Untuk informasi, lihat hargaAWS Key Management Service.

Untuk mengaktifkan enkripsi kunci KMS data sesi (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Session Manager.
3. Pilih tab Preferensi, dan kemudian pilih Edit.
4. Pilih kotak centang di sebelah Aktifkan enkripsi KMS.
5. Lakukan salah satu dari cara berikut:

   - Pilih tombol di samping Pilih kunci KMS di akun saya saat ini, lalu pilih kunci dari daftar.

     -atau-

     Pilih tombol di samping Masukkan alias kunci KMS atau ARN kunci KMS. Secara manual masukkan alias kunci KMS untuk kunci yang dibuat di akun Anda saat ini, atau masukkan Amazon Resource Name (ARN) kunci untuk kunci di akun lain. Berikut ini adalah contoh-contohnya:

     - Alias kunci: `alias/my-kms-key-alias`
     - ARN kunci: `arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-12345EXAMPLE`

     -atau-

     Pilih Buat kunci baru untuk membuat kunci KMS baru di akun Anda. Setelah Anda membuat kunci baru, kembali ke tab Preferensi dan pilih kunci untuk mengenkripsi data sesi di akun Anda.

   Untuk informasi lebih lanjut tentang berbagi kunci, lihat Mengizinkan Akun AWS Eksternal untuk Mengakses kunci di Panduan Developer AWS Key Management Service.

6. Pilih Simpan.

Membuat dokumenSession Manager preferensi (baris perintah)

Gunakan prosedur berikut untuk membuat dokumen SSM yang menentukan preferensi Anda untukAWS Systems ManagerSession Manager sesi. Anda dapat menggunakan dokumen untuk mengonfigurasi opsi sesi termasuk enkripsi data, durasi sesi, dan pencatatan. Misalnya, Anda dapat menentukan apakah akan menyimpan data log sesi di bucket Simple Storage Service (Amazon S3) atau grup CloudWatch log Amazon Logs. Anda dapat membuat dokumen yang menentukan preferensi umum untuk semua sesi untukAkun AWS danWilayah AWS, atau yang menentukan preferensi untuk setiap sesi.

> **Note**
>
> Anda juga dapat mengonfigurasi preferensi sesi umum dengan menggunakan konsol Pengelola Sesi.

Dokumen yang digunakan untuk mengatur preferensi Manajer Sesi harus memilikisessionType fileStandard_Stream. Untuk informasi selengkapnya tentang dokumen Session, lihatthe section called “Skema dokumen sesi”.

Untuk informasi tentang menggunakan baris perintah untuk memperbaruiSession Manager preferensi yang ada, lihatPerbaruiSession Manager preferensi (baris perintah).

Untuk contoh cara membuat preferensi sesi menggunakanAWS CloudFormation, lihat Membuat dokumen Systems Manager untukSession Manager preferensi di PanduanAWS CloudFormation Pengguna.

> **Note**
>
> Prosedur ini menjelaskan cara membuat dokumen untuk pengaturanSession Manager preferensi diAkun AWS tingkat. Untuk membuat dokumen yang akan digunakan untuk mengatur preferensi tingkat sesi, tentukan nilai selainSSM-SessionManagerRunShell untuk input perintah terkait nama file.
> Untuk menggunakan dokumen Anda untuk menetapkan preferensi sesi yang dimulai dariAWS Command Line Interface (AWS CLI), berikan nama dokumen sebagai nilai--

`document-name` parameter. Untuk mengatur preferensi sesi yang dimulai dari konsol Pengelola Sesi, Anda dapat mengetik atau memilih nama dokumen dari daftar.

Untuk membuatSession Manager preferensi (baris perintah)

1. Buat file JSON pada mesin lokal Anda dengan nama seperti `SessionManagerRunShell.json`, dan kemudian tempel konten berikut ke dalamnya.

```
{
    "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"
        }
    }
}
```

Anda juga dapat meluluskan nilai ke preferensi sesi Anda menggunakan parameter bukan hardcoding nilai seperti yang ditunjukkan dalam contoh berikut.

```
{
   "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. Tentukan di mana Anda ingin mengirim data sesi. Anda dapat menentukan nama bucket S3 (dengan prefiks opsional) atau nama grup CloudWatch log. Jika Anda ingin mengenkripsi data antara klien lokal dan node yang dikelola, berikan kunci KMS untuk digunakan enkripsi. Berikut adalah contohnya.

```
{
  "schemaVersion": "1.0",
  "description": "Document to hold regional settings for Session Manager",
  "sessionType": "Standard_Stream",
  "inputs": {
    "s3BucketName": "DOC-EXAMPLE-BUCKET",
    "s3KeyPrefix": "MyBucketPrefix",
    "s3EncryptionEnabled": true,
    "cloudWatchLogGroupName": "MyLogGroupName",
```

```
    "cloudWatchEncryptionEnabled": true,
    "cloudWatchStreamingEnabled": false,
    "kmsKeyId": "MyKMSKeyID",
    "runAsEnabled": true,
    "runAsDefaultUser": "MyDefaultRunAsUser",
    "idleSessionTimeout": "20",
    "maxSessionDuration": "60",
    "shellProfile": {
        "windows": "MyCommands",
        "linux": "MyCommands"
    }
  }
}
```

> **Note**
>
> Jika Anda tidak ingin mengenkripsi data log sesi, ubah `true` ke `false` untuk `s3EncryptionEnabled`.
> Jika Anda tidak mengirim log ke bucket Amazon S3 atau grup CloudWatch log Logs, tidak ingin mengenkripsi data sesi aktif, atau tidak ingin mengaktifkan dukungan Run As untuk sesi di akun Anda, Anda dapat menghapus baris untuk opsi tersebut. Pastikan baris terakhir di bagian `inputs` tidak berakhir dengan koma.
> Jika Anda menambahkan ID kunci KMS untuk mengenkripsi data sesi Anda, pengguna yang memulai sesi dan node terkelola yang terhubung harus memiliki izin untuk menggunakan kunci. Anda memberikan izin untuk menggunakan kunci KMS denganSession Manager melalui kebijakan IAM. Untuk informasi, lihat topik berikut:
>
> - Tambahkan izin AWS KMS untuk pengguna di akun Anda: Contoh kebijakan IAM untuk Session Manager
> - TambahkanAWS KMS izin untuk node terkelola di akun Anda:Langkah 2: Verifikasi atau tambahkan izin instans untuk Session Manager

3. Simpan file tersebut.
4. Dalam direktori tempat Anda membuat file JSON, jalankan perintah berikut.

   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
```

Jika berhasil, perintah mengembalikan output yang serupa dengan berikut.

```
{
    "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"
```

```
    }
}
```

PerbaruiSession Manager preferensi (baris perintah)

Prosedur berikut menjelaskan cara menggunakan alat baris perintah pilihan Anda untuk melakukan perubahan padaAWS Systems ManagerSession Manager preferensi untuk AndaAkun AWS pada yang dipilihWilayah AWS. GunakanSession Manager preferensi untuk menentukan opsi log data sesi di bucket Amazon Simple Storage Service (Amazon S3) atau grup CloudWatch log Amazon Logs. Anda juga dapat menggunakanSession Manager preferensi untuk mengenkripsi data sesi Anda.

Untuk memperbaruiSession Manager preferensi (baris perintah)

1. Buat file JSON pada mesin lokal Anda dengan nama seperti `SessionManagerRunShell.json`, dan kemudian tempel konten berikut ke dalamnya.

```
{
    "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. Tentukan di mana Anda ingin mengirim data sesi. Anda dapat menentukan nama bucket S3 (dengan prefiks opsional) atau nama grup CloudWatch log log. Jika Anda ingin mengenkripsi

data lebih lanjut antara klien lokal dan node yang dikelola, berikanAWS KMS key untuk digunakan enkripsi. Berikut adalah contohnya.

```
{
  "schemaVersion": "1.0",
  "description": "Document to hold regional settings for Session Manager",
  "sessionType": "Standard_Stream",
  "inputs": {
    "s3BucketName": "DOC-EXAMPLE-BUCKET",
    "s3KeyPrefix": "MyBucketPrefix",
    "s3EncryptionEnabled": true,
    "cloudWatchLogGroupName": "MyLogGroupName",
    "cloudWatchEncryptionEnabled": true,
    "cloudWatchStreamingEnabled": false,
    "kmsKeyId": "MyKMSKeyID",
    "runAsEnabled": true,
    "runAsDefaultUser": "MyDefaultRunAsUser",
    "idleSessionTimeout": "20",
    "maxSessionDuration": "60",
    "shellProfile": {
        "windows": "MyCommands",
        "linux": "MyCommands"
    }
  }
}
```

> **Note**
>
> Jika Anda tidak ingin mengenkripsi data log sesi, ubah `true` ke `false` untuk `s3EncryptionEnabled`.
>
> Jika Anda tidak mengirim log ke bucket Amazon S3 atau grup CloudWatch log, tidak ingin mengenkripsi data sesi aktif, atau tidak ingin mengaktifkan dukungan Run As untuk sesi di akun Anda, Anda dapat menghapus baris untuk opsi tersebut. Pastikan baris terakhir di bagian `inputs` tidak berakhir dengan koma.
>
> Jika Anda menambahkan ID kunci KMS untuk mengenkripsi data sesi, pengguna yang memulai sesi dan node terkelola yang terhubung harus memiliki izin untuk menggunakan kunci. Anda memberikan izin untuk menggunakan kunci KMS denganSession Manager melalui kebijakanAWS Identity and Access Management (IAM). Untuk informasi lebih lanjut, lihat topik berikut:

- Tambahkan izin AWS KMS untuk pengguna di akun Anda: Contoh kebijakan IAM untuk Session Manager.
- TambahkanAWS KMS izin untuk node terkelola di akun Anda:Langkah 2: Verifikasi atau tambahkan izin instans untuk Session Manager.

3. Simpan file tersebut.
4. Dalam direktori tempat Anda membuat file JSON, jalankan perintah berikut.

   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'
   ```

   Jika berhasil, perintah mengembalikan output yang serupa dengan berikut.

   ```
   {
       "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"
    }
}
```

Langkah 5: (Opsional) Batasi akses ke perintah dalam sesi

Anda dapat membatasi perintah yang dapat dijalankan oleh pengguna dalamAWS Systems ManagerSession Manager sesi dengan menggunakan dokumenAWS Systems Manager (SSM)`Session` tipe khusus. Dalam dokumen, Anda menentukan perintah yang dijalankan saat pengguna memulai sesi dan parameter yang dapat diberikan pengguna pada perintah. Dokumen `Session` dari `schemaVersion` harus 1.0, dan `sessionType` dokumen harus `InteractiveCommands`. Anda kemudian dapat membuat kebijakanAWS Identity and Access Management (IAM) yang mengizinkan pengguna untuk mengakses hanya`Session` dokumen yang Anda tentukan. Untuk informasi selengkapnya tentang cara menggunakan kebijakan IAM untuk membatasi akses ke perintah dalam sesi, lihat Contoh kebijakan IAM untuk perintah interaktif.

Dokumen dengan`sessionType` of hanya`InteractiveCommands` didukung untuk sesi yang dimulai dariAWS Command Line Interface (AWS CLI). Pengguna memberikan nama dokumen kustom sebagai nilai`--document-name` parameter dan memberikan nilai parameter perintah apa pun menggunakan`--parameters` opsi. Untuk informasi selengkapnya tentang menjalankan perintah interaktif, lihat Memulai sesi (perintah interaktif dan noninteraktif).

Gunakan prosedur untuk membuat dokumen SSM`Session` tipe khusus yang menentukan perintah yang boleh dijalankan pengguna.

Batasi akses ke perintah dalam sesi (konsol)

Untuk membatasi perintah yang dapat dijalankan oleh pengguna dalamSession Manager sesi (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.
3. Pilih Buat perintah atau sesi.
4. Untuk Nama, masukkan nama deskriptif untuk dokumen.
5. Untuk tipe dokumen, pilih Dokumen sesi.
6. Masukkan konten dokumen Anda yang menentukan perintah yang dapat dijalankan oleh pengguna dalamSession Manager sesi menggunakan JSON atau YAKL, seperti yang ditunjukkan pada contoh berikut.

   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. Pilih Buat dokumen.

Batasi akses ke perintah dalam sesi (baris perintah)

Sebelum Anda memulai

Jika Anda belum menjalankannya, Instal dan konfigurasikan AWS Command Line Interface (AWS CLI) atau AWS Tools for PowerShell. Untuk informasi, lihat Menginstal atau memperbarui versi terbaruAWS CLI dan MemasangAWS Tools for PowerShell.

Untuk membatasi perintah yang dapat dijalankan oleh pengguna dalamSession Manager sesi (baris perintah)

1. Buat file JSON atau YAKL untuk konten dokumen Anda yang menentukan perintah yang dapat dijalankan pengguna dalamSession Manager sesi, seperti yang ditunjukkan pada contoh berikut.

   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. Jalankan perintah berikut untuk membuat dokumen SSM menggunakan konten Anda yang menentukan perintah yang dapat dijalankan pengguna dalamSession Manager sesi.

   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"
```

Parameter perintah interaktif dan AWS CLI

Ada berbagai cara yang dapat Anda lakukan dalam memberikan parameter perintah interaktif saat menggunakan AWS CLI. Tergantung pada sistem operasi (OS) mesin klien Anda yang Anda gunakan untuk menghubungkan ke node yang dikelola denganAWS CLI, sintaks yang Anda berikan untuk perintah yang berisi karakter khusus atau escape mungkin berbeda. Contoh berikut menunjukkan beberapa cara yang berbeda yang dapat Anda lakukan dalam memberikan parameter perintah saat menggunakan AWS CLI, dan cara menangani karakter khusus atau escape.

Parameter yangParameter Store dapat direferensikan dalamAWS CLI untuk parameter perintah Anda seperti yang ditunjukkan dalam contoh berikut.

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}}"]}'
```

Contoh berikut menunjukkan bagaimana Anda dapat menggunakan sintaks singkat dengan AWS CLI untuk meneruskan parameter.

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"
```

Anda juga dapat memberikan parameter di JSON seperti yang ditunjukkan dalam contoh berikut.

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"]}'
```

Parameter juga dapat disimpan dalam file JSON dan diberikan untuk AWS CLI seperti yang ditunjukkan dalam contoh berikut. Untuk informasi lebih lanjut tentang penggunaan parameter AWS CLI dari file, lihat Memuat parameter AWS CLI dari file di Panduan Pengguna 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
```

Anda juga dapat membuat kerangka AWS CLI dari file input JSON seperti yang ditunjukkan dalam contoh berikut. Untuk informasi lebih lanjut tentang membuat kerangka AWS CLI dari file input JSON, lihat Membuat kerangka AWS CLI dan parameter input dari file input JSON atau YAML di Panduan Pengguna 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
```

Agar karakter escape di dalam tanda kutip, Anda harus menambahkan garis miring tambahan untuk karakter escape seperti yang ditunjukkan dalam contoh berikut.

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\""]}'
```

Untuk informasi tentang menggunakan tanda kutip dengan parameter perintah di AWS CLI lihat Menggunakan tanda kutip dengan string di AWS CLI pada AWS Command Line Interface Panduan Pengguna.

Contoh kebijakan IAM untuk perintah interaktif

Anda dapat membuat kebijakan IAM yang mengizinkan pengguna untuk mengakses hanya dokumen `Session` yang Anda tentukan. Hal ini membatasi perintah yang dapat dijalankan pengguna dalamSession Manager sesi hanya untuk perintah yang ditentukan dalam dokumen SSM`Session` tipe khusus Anda.

Izinkan pengguna untuk menjalankan perintah interaktif pada satu node yang dikelola

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"ssm:StartSession",
         "Resource":[
            "arn:aws:ec2:region:987654321098:instance/i-02573cafcfEXAMPLE",
```

```
            "arn:aws:ssm:region:987654321098:document/exampleAllowedSessionDocument"
         ],
         "Condition":{
            "BoolIfExists":{
               "ssm:SessionDocumentAccessCheck":"true"
            }
         }
      }
   ]
}
```

Izinkan pengguna untuk menjalankan perintah interaktif pada semua node yang dikelola

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"ssm:StartSession",
         "Resource":[
            "arn:aws:ec2:us-west-2:987654321098:instance/*",
            "arn:aws:ssm:us-
west-2:987654321098:document/exampleAllowedSessionDocument"
         ],
         "Condition":{
            "BoolIfExists":{
               "ssm:SessionDocumentAccessCheck":"true"
            }
         }
      }
   ]
}
```

Izinkan pengguna untuk menjalankan beberapa perintah interaktif pada semua node yang dikelola

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"ssm:StartSession",
         "Resource":[
            "arn:aws:ec2:us-west-2:987654321098:instance/*",
```

```
            "arn:aws:ssm:us-
west-2:987654321098:document/exampleAllowedSessionDocument",
            "arn:aws:ssm:us-
west-2:987654321098:document/exampleAllowedSessionDocument2"
         ],
         "Condition":{
            "BoolIfExists":{
               "ssm:SessionDocumentAccessCheck":"true"
            }
         }
      }
   ]
}
```

Langkah 6: (Opsional) GunakanAWS PrivateLink untuk menyiapkan VPC endpoint untukSession Manager

Anda dapat lebih meningkatkan postur keamanan node terkelola Anda dengan mengonfigurasiAWS Systems Manager untuk menggunakan virtual private cloud (VPC) endpoint antarmuka. Titik akhir antarmuka didukung oleh AWS PrivateLink, teknologi yang mengizinkan Anda mengakses Amazon Elastic Compute Cloud (Amazon EC2) dan API Systems Manager secara privat dengan menggunakan alamat IP privat.

AWS PrivateLinkmembatasi semua lalu lintas jaringan antara node terkelola Anda, Systems Manager, dan Amazon EC2 ke jaringan Amazon. (Node terkelola tidak memiliki akses ke internet.) Selain itu, Anda tidak memerlukan gateway internet, perangkat NAT, atau gateway privat virtual.

Untuk informasi tentang membuat endpoint VPC, lihat Membuat endpoint VPC.

Alternatif untuk menggunakan VPC endpoint adalah untuk memungkinkan akses internet keluar pada node terkelola Anda. Dalam kasus ini, node terkelola juga harus mengizinkan lalu lintas keluar HTTPS (port 443) ke titik akhir berikut:

- `ec2messages.region.amazonaws.com`
- `ssm.region.amazonaws.com`
- `ssmmessages.region.amazonaws.com`

Systems Manager menggunakan titik akhir akhir ini`ssmmessages.region.amazonaws.com`, untuk membuat panggilan dariSSM Agent keSession Manager layanan di cloud.

Untuk menggunakan fitur opsional sepertiAWS Key Management Service (AWS KMS) enkripsi, streaming log ke Amazon CloudWatch Logs (CloudWatch Log), dan mengirim log ke Amazon Simple Storage Service (Amazon S3), Anda harus mengizinkan lalu lintas keluar HTTPS (port 443) ke titik akhir berikut:

- `kms.region.amazonaws.com`
- `logs.region.amazonaws.com`
- `s3.region.amazonaws.com`

Untuk informasi lebih lanjut tentang titik akhir yang diperlukan untuk Systems Manager, lihat Referensi: ec2messages, ssmmessages, dan operasi API lainnya.

Langkah 7: (Opsional) Aktifkan atau nonaktifkan izin administratif akun ssm-user

Dimulai dengan versi 2.3.50.0 dariAWS Systems ManagerSSM Agent, agen membuat akun pengguna lokal yang dipanggil`ssm-user` dan menambahkannya ke`/etc/sudoers` (LinuxdanmacOS) atau ke grup Administrator (Windows). Pada versi agen lebih awal dari 2.612.0, akun dibuat saat pertama kaliSSM Agent dimulai atau dimulai ulang setelah instalasi. Pada versi 2.612.0 dan yang lebih baru,`ssm-user` akun dibuat saat pertama kali sesi dimulai pada sebuah node. Ini`ssm-user` adalah pengguna sistem operasi (OS) default ketikaAWS Systems ManagerSession Manager sesi dimulai. SSM Agentversi 2.3.612.0 dirilis pada 8 Mei 2019.

Jika Anda ingin mencegahSession Manager pengguna dari menjalankan perintah administratif pada sebuah node, Anda dapat memperbarui izin`ssm-user` akun. Anda juga dapat memulihkan izin ini setelah izin yang diperbarui telah dihapus.

Topik

- Mengelola izin akun sudo ssm-user padaLinux danmacOS
- Mengelola izin akun Administrator ssm-user pada Windows Server

Mengelola izin akun sudo ssm-user padaLinux danmacOS

Gunakan salah satu dari prosedur berikut ini untuk mengaktifkan atau menonaktifkan izin sudo akun ssm-user padaLinux dan nodemacOS terkelola.

GunakanRun Command untuk memodifikasi izin sudo ssm-user (konsol)

- Gunakan prosedur di Menjalankan perintah dari konsol dengan nilai berikut:

- Untuk Dokumen perintah, pilih `AWS-RunShellScript`.
- Untuk menghapus akses sudo, di area Parameter perintah, tempel berikut ini di kotak Perintah.

```
cd /etc/sudoers.d
echo "#User rules for ssm-user" > ssm-agent-users
```

-atau-

Untuk memulihkan akses sudo, di area Parameter perintah, tempel berikut ini di kotak Perintah.

```
cd /etc/sudoers.d
echo "ssm-user ALL=(ALL) NOPASSWD:ALL" > ssm-agent-users
```

Gunakan baris perintah untuk memodifikasi izin sudo ssm-user (AWS CLI)

1. Connect ke node yang dikelola dan jalankan perintah berikut.

```
sudo -s
```

2. Ganti direktori kerja menggunakan perintah berikut.

```
cd /etc/sudoers.d
```

3. Buka file bernama `ssm-agent-users` untuk mengedit.
4. Untuk menghapus akses sudo, hapus baris berikut.

```
ssm-user ALL=(ALL) NOPASSWD:ALL
```

-atau-

Untuk memulihkan akses sudo, tambahkan baris berikut.

```
ssm-user ALL=(ALL) NOPASSWD:ALL
```

5. Simpan file.

Mengelola izin akun Administrator ssm-user pada Windows Server

Gunakan salah satu dari prosedur berikut untuk mengaktifkan atau menonaktifkan izin Administrator akun ssm-user pada node yangWindows Server dikelola.

GunakanRun Command untuk memodifikasi izin Administrator (konsol)

- Gunakan prosedur di Menjalankan perintah dari konsol dengan nilai berikut:

  Untuk Dokumen perintah, pilih `AWS-RunPowerShellScript`.

  Untuk menghapus akses administratif, di area Parameter perintah, tempel berikut ini di kotak Perintah.

  ```
  net localgroup "Administrators" "ssm-user" /delete
  ```

  -atau-

  Untuk memulihkan akses administratif, di area Parameter perintah, tempel berikut ini di kotak Perintah.

  ```
  net localgroup "Administrators" "ssm-user" /add
  ```

Gunakan jendelaPowerShell atau prompt perintah untuk memodifikasi izin Administrator

1. Connect ke node terkelola dan buka jendelaPowerShell atau Command Prompt.
2. Untuk menghapus akses administratif, jalankan perintah berikut.

   ```
   net localgroup "Administrators" "ssm-user" /delete
   ```

   -atau-

   Untuk memulihkan akses administratif, jalankan perintah berikut.

   ```
   net localgroup "Administrators" "ssm-user" /add
   ```

GunakanWindows konsol untuk memodifikasi izin Administrator

1. Connect ke node terkelola dan buka jendelaPowerShell atau Command Prompt.
2. Dari baris perintah, jalankan `lusrmgr.msc` untuk membuka konsol Pengguna dan Grup Lokal.
3. Buka direktori Pengguna, dan kemudian buka ssm-user.
4. Pada tab Anggota Dari, lakukan salah satu hal berikut:

   - Untuk menghapus akses administratif, pilih Administrator, dan kemudian pilih Hapus.

     -atau-

     Untuk memulihkan akses administratif, masukkan **`Administrators`** di kotak teks, dan kemudian pilih Tambahkan.
5. Pilih OKE.

Langkah 8: (Opsional) Izinkan dan kontrol izin untuk koneksi SSH melalui Session Manager

Anda dapat mengizinkan pengguna Akun AWS untuk menggunakan AWS Command Line Interface (AWS CLI) untuk membuat koneksi Secure Shell (SSH) ke node terkelola menggunakan AWS Systems ManagerSession Manager. Pengguna yang terhubung menggunakan SSH juga dapat menyalin file antara mesin lokal mereka dan node yang dikelola menggunakan Secure Copy Protocol (SCP). Anda dapat menggunakan fungsi ini untuk terhubung ke node terkelola tanpa membuka port masuk atau mempertahankan host bastion.

Setelah mengizinkan koneksi SSH, Anda dapat menggunakan kebijakan AWS Identity and Access Management (IAM) untuk secara eksplisit mengizinkan atau menolak pengguna, grup, atau peran untuk membuat koneksi SSH menggunakan. Session Manager

> **Note**
>
> Logging tidak tersedia untuk Session Manager sesi yang terhubung melalui port forwarding atau SSH. Ini karena SSH mengenkripsi semua data sesi, dan Session Manager hanya berfungsi sebagai terowongan untuk koneksi SSH.

Topik

- Mengizinkan koneksi SSH untuk Session Manager
- Mengontrol izin pengguna untuk koneksi SSH melalui Session Manager

Mengizinkan koneksi SSH untuk Session Manager

Gunakan langkah-langkah berikut untuk mengizinkan koneksi SSH melalui Session Manager pada node terkelola.

Untuk memungkinkan koneksi SSH untuk Session Manager

1. Pada node terkelola yang ingin Anda izinkan koneksi SSH, lakukan hal berikut:

   - Pastikan SSH berjalan pada node yang dikelola. (Anda dapat menutup port masuk pada node.)
   - Pastikan SSM Agent versi 2.3.672.0 atau yang lebih baru diinstal pada node terkelola.

     Untuk informasi tentang menginstal atau memperbarui SSM Agent pada node terkelola, lihat topik berikut:

     - Bekerja dengan SSM Agent instans EC2 untuk Windows Server.
     - Bekerja denganSSM Agent instans EC2 untuk Linux
     - Bekerja dengan SSM Agent instans EC2 untuk macOS
     - Instal SSM Agent untuk lingkungan hybrid (Windows)
     - Instal SSM Agent untuk lingkungan hybrid (Linux)

     > **Note**
     >
     > Untuk menggunakan Session Manager server lokal, perangkat edge, dan mesin virtual (VM) yang diaktifkan sebagai node terkelola, Anda harus menggunakan tingkat instance lanjutan. Untuk informasi selengkapnya tentang instans lanjutan, lihat Mengonfigurasi tingkat instans.

2. Pada mesin lokal tempat Anda ingin terhubung ke node terkelola menggunakan SSH, lakukan hal berikut:

   - Pastikan bahwa versi 1.1.23.0 atau yang lebih baru dari Session Manager plugin diinstal.

     Untuk informasi tentang menginstal Session Manager plugin, lihatInstal Session Manager plugin untuk AWS CLI.

- Perbarui file konfigurasi SSH untuk memungkinkan menjalankan perintah proxy yang memulai Session Manager sesi dan mentransfer semua data melalui koneksi.

  Linux dan macOS

  > **Tip**
  >
  > File konfigurasi SSH biasanya terletak di `~/.ssh/config`.

  Tambahkan berikut ke file konfigurasi di mesin lokal.

  ```
  # SSH over Session Manager
  host i-* mi-*
      ProxyCommand sh -c "aws ssm start-session --target %h --document-name AWS-
  StartSSHSession --parameters 'portNumber=%p'"
  ```

  Windows

  > **Tip**
  >
  > File konfigurasi SSH biasanya terletak di `C:\Users\<username>\.ssh\config`.

  Tambahkan berikut ke file konfigurasi di mesin lokal.

  ```
  # 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"
  ```

- Buat atau verifikasi bahwa Anda memiliki sertifikat Privacy Enhanced Mail (file PEM), atau setidaknya kunci publik, untuk digunakan saat membuat koneksi ke node terkelola. Ini harus menjadi kunci yang sudah dikaitkan dengan node terkelola. Izin file kunci pribadi Anda harus diatur sehingga hanya Anda yang dapat membacanya. Anda dapat menggunakan perintah berikut untuk mengatur izin file kunci pribadi Anda sehingga hanya Anda yang dapat membacanya.

```
chmod 400 <my-key-pair>.pem
```

Misalnya, untuk instans Amazon Elastic Compute Cloud (Amazon EC2), file pasangan kunci yang Anda buat atau pilih saat Anda membuat instans. (Anda menentukan jalur ke sertifikat atau kunci sebagai bagian dari perintah untuk memulai sesi. Untuk informasi tentang memulai sesi menggunakan SSH, lihat Memulai sesi (SSH).)

Mengontrol izin pengguna untuk koneksi SSH melalui Session Manager

Setelah Anda mengaktifkan koneksi SSH melalui Session Manager pada node terkelola, Anda dapat menggunakan kebijakan IAM untuk mengizinkan atau menolak pengguna, grup, atau peran kemampuan untuk membuat koneksi SSH. Session Manager

Untuk menggunakan kebijakan IAM untuk mengizinkan koneksi SSH melalui Session Manager

- Gunakan salah satu opsi berikut:
  - Opsi 1: Buka konsol IAM di https://console.aws.amazon.com/iam/.

    Di panel navigasi, pilih Kebijakan, lalu perbarui kebijakan izin untuk pengguna atau peran yang ingin Anda izinkan untuk memulai koneksi SSH. Session Manager

    Misalnya, tambahkan elemen berikut ke kebijakan Quickstart yang Anda buat. Kebijakan pengguna akhir Quickstart untuk Session Manager Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ssm:StartSession",
            "Resource": [
                "arn:aws:ec2:region:account-id:instance/instance-id",
                "arn:aws:ssm:*:*:document/AWS-StartSSHSession"
            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck": "true"
                }
```

```
            }
        }
    ]
}
```

- Opsi 2: Lampirkan kebijakan inline ke kebijakan pengguna dengan menggunakan AWS Management Console, AWS CLI, atau API AWS.

  Dengan menggunakan metode pilihan Anda, lampirkan pernyataan kebijakan di Opsi 1 ke kebijakan untuk AWS pengguna, grup, atau peran.

  Untuk informasi, lihat Menambahkan dan Menghapus Izin Identitas IAM dalam Panduan Pengguna IAM.

Untuk menggunakan kebijakan IAM untuk menolak koneksi SSH melalui Session Manager

- Gunakan salah satu opsi berikut:

  - Opsi 1: Buka konsol IAM di https://console.aws.amazon.com/iam/. Di panel navigasi, pilih Kebijakan, lalu perbarui kebijakan izin untuk pengguna atau peran yang akan diblokir dari sesi awalSession Manager.

    Misalnya, tambahkan elemen berikut ke kebijakan Quickstart yang Anda buat. Kebijakan pengguna akhir Quickstart untuk Session Manager

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor1",
            "Effect": "Deny",
            "Action": "ssm:StartSession",
            "Resource": "arn:aws:ssm:*:*:document/AWS-StartSSHSession"
        }
    ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck": "true"
                }
             }
}
```

- Opsi 2: Lampirkan kebijakan inline ke kebijakan pengguna dengan menggunakan AWS Management Console, AWS CLI, atau API AWS.

  Dengan menggunakan metode pilihan Anda, lampirkan pernyataan kebijakan di Opsi 1 ke kebijakan untuk AWS pengguna, grup, atau peran.

  Untuk informasi, lihat Menambahkan dan Menghapus Izin Identitas IAM dalam Panduan Pengguna IAM.

Bekerja dengan Session Manager

Anda dapat menggunakanAWS Systems Managerkonsol Amazon Elastic Compute Cloud (Amazon EC2), atauAWS Command Line Interface(AWS CLI) untuk memulai sesi yang menghubungkan Anda ke node terkelola yang telah diberikan administrator sistem Anda akses untuk menggunakanAWS Identity and Access Management(IAM) kebijakan. Tergantung pada izin Anda, Anda juga dapat melihat informasi tentang sesi, melanjutkan sesi tidak aktif yang belum kehabisan waktu, dan mengakhiri sesi. Setelah sesi dibuat, itu tidak terpengaruh oleh durasi sesi peran IAM. Untuk informasi tentang membatasi durasi sesi denganSession ManagerLihatTentukan nilai waktu habis sesi idledanTentukan durasi sesi maksimum .

Untuk informasi lebih lanjut tentang sesi, lihat Apa itu sesi?

Topik

- Instal Session Manager plugin untuk AWS CLI
- Mulai sesi
- Mengakhiri sesi
- Lihat riwayat sesi

Instal Session Manager plugin untuk AWS CLI

Untuk memulai Session Manager sesi dengan node terkelola Anda dengan menggunakan AWS Command Line Interface (AWS CLI), Anda harus menginstal Session Managerplugin di mesin lokal Anda. Anda dapat menginstal plugin pada versi MicrosoftWindows Server, LinuxmacOS, dan versi yang didukungUbuntu Server.

> **Note**
>
> Untuk menggunakan Session Manager plugin, Anda harus memiliki AWS CLI versi 1.16.12 atau yang lebih baru diinstal pada mesin lokal Anda. Untuk informasi selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS Command Line Interface.

Topik

Session Managerplugin versi terbaru dan riwayat rilis

Mesin lokal Anda harus menjalankan versi Session Manager plugin yang didukung. Versi minimum yang didukung saat ini adalah 1.1.17.0. Jika Anda menjalankan versi yang lebih lama, Session Manager operasi Anda mungkin tidak berhasil.

Untuk melihat apakah Anda memiliki versi terbaru, jalankan perintah berikut di AWS CLI.

> **Note**
>
> Perintah mengembalikan hasil hanya jika plugin terletak di direktori pemasangan default untuk tipe sistem operasi Anda. Anda juga dapat memeriksa versi dalam isi file `VERSION` di direktori tempat Anda telah memasang plugin.

```
session-manager-plugin --version
```

Tabel berikut mencantumkan semua rilis Session Manager plugin dan fitur serta perangkat tambahan yang disertakan dengan setiap versi.

Versi	Tanggal rilis	Detail
1.2.553.0	10 Januari 2024	Peningkatan: Paket Golang yang ditingkatkan aws-sdk-go dan bergantung.
1.2.536.0	Desember 4, 2023	Peningkatan: Menambahkan dukungan untuk meneruskan respons StartSessionAPI sebagai variabel lingkungan ke session-manager-plugin.
1.2.497.0	1 Agustus 2023	Peningkatan: Upgrade Go SDK ke v1.44.302.
1.2.463.0	15 Maret 2023	Peningkatan: Menambahkan Mac with Apple silicon dukungan untuk Apple Mac (M1) di penginstal bundel macOS dan penginstal yang ditandatangani.
1.2.398.0	14 Oktober 2022	Enhancement: Support golang versi 1.17. Perbarui session-manager-plugin runner default untuk macOS untuk menggunakan python3. Perbarui jalur impor dari SSMCLI ke. session-manager-plugin
1.2.339.0	16 Juni 2022	Perbaikan bug: Perbaiki batas waktu sesi idle untuk sesi port.
1.2.331.0	Mei 27, 2022	Perbaikan bug: Perbaiki sesi port yang ditutup sebelum waktunya ketika server lokal tidak terhubung sebelum batas waktu.
1.2.323.0	Mei 19, 2022	Perbaikan bug: Nonaktifkan smux keep alive untuk menggunakan fitur batas waktu sesi idle.
1.2.312.0	31 Maret 2022	Peningkatan: Mendukung lebih banyak jenis payload pesan keluaran.
1.2.295.0	12 Januari 2022	Perbaikan bug: Sesi digantung yang disebabkan oleh klien mengirim ulang data aliran saat agen menjadi tidak aktif, dan log dan pesan yang salah. `start_publication` `pause_publication`

Versi	Tanggal rilis	Detail
1.2.279.0	27 Oktober 2021	Peningkatan: Kemasan zip untuk Windows platform.
1.2.245.0	19 Agustus 2021	Peningkatan: Tingkatkan `aws-sdk-go` ke versi terbaru (v1.40.17) untuk mendukung. AWS IAM Identity Center
1.2.234.0	26 Juli 2021	Perbaikan bug: Menangani sesi skenario yang tiba-tiba dihentikan dalam jenis sesi interaktif.
1.2.205.0	10 Juni 2021	Peningkatan: Dukungan tambahan untuk pemasang macOS yang ditandatangani.
1.2.54.0	29 Januari 2021	Peningkatan: Menambahkan dukungan untuk menjalankan sesi dalam mode NonInteractiveCommands eksekusi.
1.2.30.0	24 November 2020	Peningkatan: (Hanya sesi penerusan port) Meningkatkan kinerja secara keseluruhan.
1.2.7.0	15 Oktober 2020	Peningkatan: (Hanya sesi penerusan port) Mengurangi latensi dan meningkatkan kinerja secara keseluruhan.
1.1.61.0	17 April 2020	Peningkatan: Menambahkan dukungan ARM untuk Linux dan Ubuntu.
1.1.54.0	6 Januari 2020	Perbaikan bug: Menangani skenario kondisi balapan paket yang dijatuhkan saat Session Manager plugin belum siap.
1.1.50.0	19 November 2019	Peningkatan: Dukungan tambahan untuk meneruskan port ke soket unix lokal.
1.1.35.0	7 November 2019	Peningkatan: (Hanya sesi penerusan port) Kirim Terminate Session perintah ke SSM Agent saat pengguna lokal menekan. `Ctrl+C`
1.1.33.0	26 September 2019	Peningkatan: (Hanya sesi penerusan port) Mengirim sinyal pemutusan sambungan ke server ketika klien memutuskan koneksi TCP.

Versi	Tanggal rilis	Detail
1.1.31.0	6 September 2019	Peningkatan: Memperbarui untuk menjaga sesi penerusan port tetap terbuka hingga server jarak jauh menutup koneksi.
1.1.26.0	30 Juli 2019	Peningkatan: Memperbarui untuk membatasi tingkat transfer data selama sesi.
1.1.23.0	9 Juli 2019	Peningkatan: Menambahkan dukungan untuk menjalankan sesi SSH menggunakan. Session Manager
1.1.17.0	4 April 2019	Peningkatan: Dukungan tambahan untuk enkripsi data sesi lebih lanjut menggunakan AWS Key Management Service (AWS KMS).
1.0.37.0	20 September 2018	Peningkatan: Perbaikan bug untuk Windows versi.
1.0.0.0	11 September 2018	Rilis awal Session Manager plugin.

Instal Session Manager plugin pada Windows

Anda dapat menginstal Session Manager plugin pada Windows Vista atau nanti menggunakan installer mandiri.

Saat pembaruan dirilis, Anda harus mengulangi proses instalasi untuk mendapatkan versi terbaru Session Manager plugin.

> **Note**
>
> Untuk hasil terbaik, kami menyarankan Anda memulai sesi penggunaan Windows klienWindows PowerShell, versi 5 atau yang lebih baru. Atau, Anda dapat menggunakan shell Command diWindows 10. Session ManagerPlugin hanya mendukung PowerShell dan shell Command. Alat baris perintah pihak ketiga mungkin tidak kompatibel dengan plugin.

Untuk menginstal Session Manager plugin menggunakan installer EXE

1. Unduh pemasang menggunakan URL berikut.

   ```
   https://s3.amazonaws.com/session-manager-downloads/plugin/latest/windows/
   SessionManagerPluginSetup.exe
   ```

   Atau, Anda dapat mengunduh versi zip installer menggunakan URL berikut.

   ```
   https://s3.amazonaws.com/session-manager-downloads/plugin/latest/windows/
   SessionManagerPlugin.zip
   ```

2. Jalankan pemasang yang diunduh, dan ikuti petunjuk di layar. Jika Anda mengunduh versi zip installer, Anda harus unzip installer terlebih dahulu.

   Biarkan kotak lokasi pemasangan kosong untuk memasang plugin ke direktori default.

   - `%PROGRAMFILES%\Amazon\SessionManagerPlugin\bin\`

3. Verifikasi bahwa pemasangan berhasil. Untuk informasi, lihat VerifikasiSession Managerinstalasi Plugin.

   > **Note**
   >
   > Jika Windows tidak dapat menemukan executable, Anda mungkin perlu membuka kembali command prompt atau menambahkan direktori instalasi ke variabel PATH lingkungan Anda secara manual. Untuk informasi, lihat topik pemecahan masalah Session Managerplugin tidak secara otomatis ditambahkan ke jalur baris perintah (Windows).

Instal Session Manager plugin di macOS

Pilih salah satu topik berikut untuk menginstal Session Manager pluginmacOS. Penginstal yang dibundel menggunakan file ZIP. Setelah di-unzip, Anda dapat menginstal plugin menggunakan biner. Penginstal yang ditandatangani adalah file.pkg yang ditandatangani.

Topik

- Instal Session Manager plugin di macOS
- Instal Session Manager plugin macOS dengan installer yang ditandatangani

Instal Session Manager plugin di macOS

Bagian ini menjelaskan cara menginstal Session Manager plugin macOS menggunakan installer yang dibundel.

> **Important**
>
> Pemasang yang dibundel tidak mendukung pemasangan ke jalur yang berisi spasi.

Untuk menginstal Session Manager plugin menggunakan installer bundel () macOS

1. Unduh pemasang yang dibundel.

   x86_64

   ```
   curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/mac/
   sessionmanager-bundle.zip" -o "sessionmanager-bundle.zip"
   ```

   Mac dengan silikon Apple

   ```
   curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/
   mac_arm64/sessionmanager-bundle.zip" -o "sessionmanager-bundle.zip"
   ```

2. Buka zip paketnya.

   ```
   unzip sessionmanager-bundle.zip
   ```

3. Jalankan perintah pemasangan.

   ```
   sudo ./sessionmanager-bundle/install -i /usr/local/sessionmanagerplugin -b /usr/
   local/bin/session-manager-plugin
   ```

   > **Note**
   >
   > Plugin membutuhkan Python 2.6.5 atau yang lebih baru, atau Python 3.3 atau yang lebih baru. Secara default, skrip pemasangan berjalan berdasarkan versi default sistem Python. Jika Anda telah menginstal versi alternatif Python dan ingin menggunakannya untuk menginstal Session Manager plugin, jalankan skrip instal dengan versi itu dengan jalur absolut ke executable Python. Berikut adalah contohnya.

```
sudo /usr/local/bin/python3.8 sessionmanager-bundle/install -i /usr/local/
sessionmanagerplugin -b /usr/local/bin/session-manager-plugin
```

Installer menginstal Session Manager plugin di `/usr/local/sessionmanagerplugin` dan membuat symlink `session-manager-plugin` di direktori. `/usr/local/bin` Hal ini menghilangkan kebutuhan untuk menentukan direktori pemasangan di variabel `$PATH` pengguna.

Untuk melihat penjelasan opsi `-i` dan `-b`, gunakan opsi `-h`.

```
./sessionmanager-bundle/install -h
```

4. Verifikasi bahwa pemasangan berhasil. Untuk informasi, lihat VerifikasiSession Managerinstalasi Plugin.

> **Note**
>
> Untuk menghapus instalasi plugin, jalankan dua perintah berikut dalam urutan yang ditunjukkan.
>
> ```
> sudo rm -rf /usr/local/sessionmanagerplugin
> ```
>
> ```
> sudo rm /usr/local/bin/session-manager-plugin
> ```

Instal Session Manager plugin macOS dengan installer yang ditandatangani

Bagian ini menjelaskan cara menginstal Session Manager plugin saat macOS menggunakan penginstal yang ditandatangani.

Untuk menginstal Session Manager plugin menggunakan installer yang ditandatangani () macOS

1. Unduh pemasang yang ditandatangani.

x86_64

```
curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/mac/
session-manager-plugin.pkg" -o "session-manager-plugin.pkg"
```

Mac dengan silikon Apple

```
curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/
mac_arm64/session-manager-plugin.pkg" -o "session-manager-plugin.pkg"
```

2. Jalankan perintah instal.

```
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. Verifikasi bahwa pemasangan berhasil. Untuk informasi, lihat VerifikasiSession Managerinstalasi Plugin.

Instal Session Manager plugin pada Red Hat Enterprise Linux distribusi

Gunakan prosedur berikut untuk menginstal Session Manager plugin pada RHEL distribusi.

> **Note**
>
> Session ManagerPlugin ini tidak didukung di Amazon Linux 1. Ini didukung di Amazon Linux 2 dan distribusi yang lebih baru.

1. Unduh dan instal paket Session Manager plugin RPM.

   x86_64

   Pada RHEL 7, jalankan perintah berikut:

   ```
   sudo yum install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
   latest/linux_64bit/session-manager-plugin.rpm
   ```

   Pada RHEL 8 dan 9, jalankan perintah berikut:

```
sudo dnf install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
latest/linux_64bit/session-manager-plugin.rpm
```

x86

Pada RHEL 7, jalankan perintah berikut:

```
sudo yum install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
latest/linux_32bit/session-manager-plugin.rpm
```

Pada RHEL 8 dan 9, jalankan perintah berikut:

```
sudo dnf install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
latest/linux_32bit/session-manager-plugin.rpm
```

ARM64

Pada RHEL 7, jalankan perintah berikut:

```
sudo yum install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
latest/linux_arm64/session-manager-plugin.rpm
```

Pada RHEL 8 dan 9, jalankan perintah berikut:

```
sudo dnf install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
latest/linux_arm64/session-manager-plugin.rpm
```

2. Verifikasi bahwa pemasangan berhasil. Untuk informasi, lihat VerifikasiSession Managerinstalasi Plugin.

> **Note**
>
> Jika Anda ingin mencopot pemasangan plugin, jalankan `sudo yum erase session-manager-plugin -y`

Instal Session Manager plugin pada Debian Server dan Ubuntu Server

1. Unduh paket Session Manager plugin deb.

   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. Jalankan perintah pemasangan.

   ```
   sudo dpkg -i session-manager-plugin.deb
   ```

3. Verifikasi bahwa pemasangan berhasil. Untuk informasi, lihat VerifikasiSession Managerinstalasi Plugin.

> **Note**
>
> Jika Anda ingin mencopot pemasangan plugin, jalankan `sudo dpkg -r session-manager-plugin`

VerifikasiSession Managerinstalasi Plugin

Jalankan perintah berikut untuk memverifikasi bahwaSession ManagerPlugin berhasil diinstal.

```
session-manager-plugin
```

Jika pemasangan berhasil, pesan berikut dikembalikan.

```
The Session Manager plugin is installed successfully. Use the AWS CLI to start a
 session.
```

Anda juga dapat menguji pemasangan dengan menjalankan perintah berikut di AWS CLI. Dalam perintah berikut, ganti*instance-id*dengan informasi Anda sendiri.

```
aws ssm start-session --target instance-id
```

Perintah ini akan bekerja hanya jika AndaSession Manageradministrator telah memberi Anda izin IAM yang diperlukan untuk mengakses node yang dikelola target menggunakanSession Manager.

Session Managerplugin pada GitHub

Kode sumber untuk Session Manager plugin tersedia GitHubsehingga Anda dapat menyesuaikan plugin untuk memenuhi kebutuhan Anda. Kami mendorong Anda untuk mengirim permintaan tarik untuk perubahan yang ingin Anda sertakan. Namun, Amazon Web Services tidak menyediakan dukungan untuk menjalankan salinan yang dimodifikasi dari perangkat lunak ini.

(Opsional) Aktifkan logging Session Manager plugin

Session ManagerPlugin ini menyertakan opsi untuk mengizinkan pencatatan untuk sesi yang Anda jalankan. Secara default, log dimatikan.

Jika Anda mengizinkan logging, Session Manager plugin akan membuat file log untuk aktivitas aplikasi (`session-manager-plugin.log`) dan error (`errors.log`) pada mesin lokal Anda.

Topik

- Aktifkan logging untuk Session Manager plugin (Windows)
- Aktifkan logging untuk Session Manager plugin (LinuxdanmacOS)

Aktifkan logging untuk Session Manager plugin (Windows)

1. Temukan file `seelog.xml.template` untuk plugin.

   Lokasi default adalah `C:\Program Files\Amazon\SessionManagerPlugin\seelog.xml.template`.
2. Ubah nama file menjadi `seelog.xml`.
3. Buka file dan ubah `minlevel="off"` menjadi `minlevel="info"` atau `minlevel="debug"`.

> **Note**
>
> Secara default, entri log tentang membuka saluran data dan menghubungkan kembali sesi dicatat di tingkat INFO. Entri aliran data (paket dan pengakuan) dicatat pada tingkat DEBUG.

4. Ubah opsi konfigurasi lain yang ingin Anda modifikasi. Opsi yang dapat Anda ubah meliputi:

   - Tingkat debug: Anda dapat mengubah tingkat debug dari `formatid="fmtinfo"` menjadi `formatid="fmtdebug"`.
   - Opsi berkas log: Anda dapat membuat perubahan pada opsi file log, termasuk di mana log disimpan, dengan pengecualian nama berkas log.

   > **Important**
   >
   > Jangan mengubah nama file atau log tidak akan bekerja dengan benar.

   ```
   <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. Simpan file tersebut.

Aktifkan logging untuk Session Manager plugin (LinuxdanmacOS)

1. Temukan file `seelog.xml.template` untuk plugin.

   Lokasi default adalah `/usr/local/sessionmanagerplugin/seelog.xml.template`.

2. Ubah nama file menjadi `seelog.xml`.
3. Buka file dan ubah `minlevel="off"` menjadi `minlevel="info"` atau `minlevel="debug"`.

> **Note**
>
> Secara default, entri log tentang membuka saluran data dan menghubungkan kembali sesi dicatat di tingkat INFO. Entri aliran data (paket dan pengakuan) dicatat pada tingkat DEBUG.

4. Ubah opsi konfigurasi lain yang ingin Anda modifikasi. Opsi yang dapat Anda ubah meliputi:

   - Tingkat debug: Anda dapat mengubah tingkat debug dari `formatid="fmtinfo"` menjadi `outputs formatid="fmtdebug"`
   - Opsi berkas log: Anda dapat membuat perubahan pada opsi file log, termasuk di mana log disimpan, dengan pengecualian nama berkas log.

   > **Important**
   >
   > Jangan mengubah nama file atau log tidak akan bekerja dengan benar.

   ```
   <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**
   >
   > Jika Anda menggunakan direktori default yang ditentukan untuk menyimpan log, Anda harus menjalankan perintah sesi menggunakan sudo atau memberikan izin baca dan tulis penuh pada direktori tempat plugin dipasang. Untuk melewati pembatasan ini, ubah lokasi di mana log disimpan.

5. Simpan file tersebut.

Mulai sesi

Anda dapat menggunakan AWS Systems Manager konsol, konsol Amazon Elastic Compute Cloud (Amazon EC2), konsol AWS CLI(), atau SSH AWS Command Line Interface untuk memulai sesi.

Topik

- Memulai sesi (konsol Systems Manager)
- Memulai sesi (konsol Amazon EC2)
- Memulai sesi (AWS CLI)
- Memulai sesi (SSH)
- Memulai sesi (penerusan port)
- Memulai sesi (penerusan port ke host jarak jauh)
- Memulai sesi (perintah interaktif dan noninteraktif)

Memulai sesi (konsol Systems Manager)

Anda dapat menggunakan AWS Systems Manager konsol untuk memulai sesi dengan node terkelola di akun Anda.

> **Note**
>
> Sebelum Anda memulai sesi, pastikan bahwa Anda telah menyelesaikan langkah-langkah pengaturan untukSession Manager. Untuk informasi, lihat Menyiapkan Session Manager.

Untuk memulai sesi (konsol Systems Manager)

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Session Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Session Managerdi panel navigasi.
3. Pilih Mulai sesi.
4. (Opsional) Masukkan deskripsi sesi di bidang Alasan sesi.
5. Untuk contoh Target, pilih tombol opsi di sebelah kiri node terkelola yang ingin Anda sambungkan.

Jika node yang Anda inginkan tidak ada dalam daftar, atau jika Anda memilih node dan menerima kesalahan konfigurasi, lihat Node terkelola tidak tersedia atau tidak dikonfigurasi untuk Session Manager untuk langkah-langkah pemecahan masalah.

6. Pilih Mulai sesi untuk segera meluncurkan sesi.

   -atau-

   Pilih Berikutnya untuk opsi sesi.

7. (Opsional) Untuk dokumen Sesi, pilih dokumen yang ingin Anda jalankan saat sesi dimulai. Jika dokumen Anda mendukung parameter runtime, Anda dapat memasukkan satu atau beberapa nilai yang dipisahkan koma di setiap bidang parameter.
8. Pilih Berikutnya.
9. Pilih Mulai sesi.

Setelah koneksi dibuat, Anda dapat menjalankan perintah bash (LinuxandmacOS) atau PowerShell command (Windows) seperti yang Anda lakukan melalui jenis koneksi lainnya.

> **Important**
>
> Jika Anda ingin mengizinkan pengguna menentukan dokumen saat memulai sesi di konsol Pengelola Sesi, perhatikan hal berikut:
>
> - Anda harus memberikan izin `ssm:GetDocument` dan `ssm:ListDocuments` izin kepada pengguna dalam kebijakan IAM mereka. Untuk informasi selengkapnya, lihat Berikan akses ke dokumen Sesi kustom di konsol.
> - Konsol hanya mendukung dokumen Sesi yang `sessionType` didefinisikan sebagai`Standard_Stream`. Untuk informasi selengkapnya, lihat Skema dokumen sesi.

Memulai sesi (konsol Amazon EC2)

Anda dapat menggunakan konsol Amazon Elastic Compute Cloud (Amazon EC2) untuk memulai sesi dengan instans di akun Anda.

> **Note**
>
> Jika Anda menerima kesalahan bahwa Anda tidak berwenang untuk melakukan satu atau lebih tindakan Systems Manager, (`ssm:command-name`, selanjutnya, Anda harus menghubungi administrator Anda untuk mendapatkan bantuan. Administrator Anda adalah orang yang memberi Anda kredensial masuk. Minta orang tersebut untuk memperbarui kebijakan Anda agar Anda diizinkan untuk memulai sesi dari konsol Amazon EC2. Jika Anda administrator, lihat Contoh kebijakan IAM untuk Session Manager untuk informasi selengkapnya.

Untuk memulai sesi (konsol Amazon EC2)

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans, lalu pilih Hubungkan.
4. Untuk metode Koneksi, pilih Session Manager.
5. Pilih Hubungkan.

Setelah koneksi dibuat, Anda dapat menjalankan perintah bash (LinuxandmacOS) atau PowerShell command (Windows) seperti yang Anda lakukan melalui jenis koneksi lainnya.

Memulai sesi (AWS CLI)

Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI.

Sebelum Anda memulai sesi, pastikan bahwa Anda telah menyelesaikan langkah-langkah pengaturan untukSession Manager. Untuk informasi, lihat Menyiapkan Session Manager.

Untuk menggunakan perintah AWS CLI to run session, Session Manager plugin juga harus diinstal pada mesin lokal Anda. Untuk informasi, lihat Instal Session Manager plugin untuk AWS CLI.

Untuk memulai sesi menggunakan AWS CLI, jalankan perintah berikut menggantikan *instance-id* dengan informasi Anda sendiri.

```
aws ssm start-session \
```

```
    --target instance-id
```

Untuk informasi tentang opsi lain yang dapat Anda gunakan dengan start-session perintah, lihat start-sessiondi AWS Systems Manager bagian Referensi AWS CLI Perintah.

Memulai sesi (SSH)

Untuk memulai sesi Session Manager SSH, versi 2.3.672.0 atau yang lebih baru SSM Agent harus diinstal pada node terkelola.

Persyaratan koneksi SSH

Perhatikan persyaratan dan batasan berikut untuk koneksi sesi menggunakan SSH:

- Node terkelola target Anda harus dikonfigurasi untuk mendukung koneksi SSH. Untuk informasi selengkapnya, lihat (Opsional) Mengizinkan dan mengontrol izin untuk koneksi SSH melalui. Session Manager
- Anda harus terhubung menggunakan akun node terkelola yang terkait dengan sertifikat Privacy Enhanced Mail (PEM), bukan `ssm-user` akun yang digunakan untuk jenis koneksi sesi lainnya. Misalnya, pada instans EC2 untuk Linux danmacOS, pengguna default adalah. `ec2-user` Untuk informasi tentang mengidentifikasi pengguna default untuk setiap tipe instans, lihat Dapatkan Informasi Tentang Instans Anda di Panduan Pengguna Amazon EC2 untuk Instans Linux.
- Logging tidak tersedia untuk Session Manager sesi yang terhubung melalui port forwarding atau SSH. Ini karena SSH mengenkripsi semua data sesi, dan Session Manager hanya berfungsi sebagai terowongan untuk koneksi SSH.

> **Note**
>
> Sebelum Anda memulai sesi, pastikan bahwa Anda telah menyelesaikan langkah-langkah pengaturan untukSession Manager. Untuk informasi, lihat Menyiapkan Session Manager.

Untuk memulai sesi menggunakan SSH, jalankan perintah berikut. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

```
ssh -i /path/my-key-pair.pem username@instance-id
```

> **Tip**
>
> Saat memulai sesi menggunakan SSH, Anda dapat menyalin file lokal ke node terkelola target menggunakan format perintah berikut.
>
> ```
> scp -i /path/my-key-pair.pem /path/ExampleFile.txt username@instance-id:~
> ```

Untuk informasi tentang opsi lain yang dapat Anda gunakan dengan start-session perintah, lihat start-sessiondi AWS Systems Manager bagian Referensi AWS CLI Perintah.

Memulai sesi (penerusan port)

Untuk memulai sesi penerusan Session Manager port, versi 2.3.672.0 atau yang lebih baru SSM Agent harus diinstal pada node terkelola.

> **Note**
>
> Sebelum Anda memulai sesi, pastikan bahwa Anda telah menyelesaikan langkah-langkah pengaturan untukSession Manager. Untuk informasi, lihat Menyiapkan Session Manager. Untuk menggunakan perintah AWS CLI to run session, Anda harus menginstal Session Manager plugin di mesin lokal Anda. Untuk informasi, lihat Instal Session Manager plugin untuk AWS CLI.
>
> Tergantung pada sistem operasi dan alat baris perintah Anda, penempatan tanda kutip dapat berbeda dan karakter escape mungkin diperlukan.

Untuk memulai sesi penerusan port, jalankan perintah berikut dari CLI. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

Linux & macOS

```
aws ssm start-session \
    --target instance-id \
    --document-name AWS-StartPortForwardingSession \
    --parameters '{"portNumber":["80"], "localPortNumber":["56789"]}'
```

Windows

```
aws ssm start-session ^
    --target instance-id ^
    --document-name AWS-StartPortForwardingSession ^
    --parameters portNumber="3389",localPortNumber="56789"
```

`portNumber`adalah port jarak jauh pada node terkelola tempat Anda ingin lalu lintas sesi dialihkan. Misalnya, Anda dapat menentukan port `3389` untuk menghubungkan ke Windows node menggunakan Remote Desktop Protocol (RDP). Jika Anda tidak menentukan `portNumber` parameter, Session Manager gunakan `80` sebagai nilai default.

`localPortNumber`adalah port di komputer lokal Anda di mana lalu lintas dimulai, seperti`56789`. Nilai ini adalah apa yang Anda masukkan saat menghubungkan ke node terkelola menggunakan klien. Misalnya, **`localhost:56789`**.

Untuk informasi tentang opsi lain yang dapat Anda gunakan dengan start-session perintah, lihat start-sessiondi AWS Systems Manager bagian Referensi AWS CLI Perintah.

Untuk informasi selengkapnya tentang sesi penerusan port, lihat Port Forwarding Using AWS Systems ManagerSession Manager di Blog Berita.AWS

Memulai sesi (penerusan port ke host jarak jauh)

Untuk memulai sesi penerusan Session Manager port ke host jarak jauh, versi 3.1.1374.0 atau yang lebih baru SSM Agent harus diinstal pada node terkelola. Host jarak jauh tidak perlu dikelola oleh Systems Manager.

> **Note**
>
> Sebelum Anda memulai sesi, pastikan bahwa Anda telah menyelesaikan langkah-langkah pengaturan untukSession Manager. Untuk informasi, lihat Menyiapkan Session Manager. Untuk menggunakan perintah AWS CLI to run session, Anda harus menginstal Session Manager plugin di mesin lokal Anda. Untuk informasi, lihat Instal Session Manager plugin untuk AWS CLI.
>
> Tergantung pada sistem operasi dan alat baris perintah Anda, penempatan tanda kutip dapat berbeda dan karakter escape mungkin diperlukan.

Untuk memulai sesi penerusan port, jalankan perintah berikut dari file. AWS CLI Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

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"
```

`host`Nilai mewakili nama host atau alamat IP dari host jarak jauh yang ingin Anda sambungkan. Konektivitas umum dan persyaratan resolusi nama antara node terkelola dan host jarak jauh masih berlaku.

`portNumber`adalah port jarak jauh pada node terkelola tempat Anda ingin lalu lintas sesi dialihkan. Misalnya, Anda dapat menentukan port `3389` untuk menghubungkan ke Windows node menggunakan Remote Desktop Protocol (RDP). Jika Anda tidak menentukan `portNumber` parameter, Session Manager gunakan `80` sebagai nilai default.

`localPortNumber`adalah port di komputer lokal Anda di mana lalu lintas dimulai, seperti`56789`. Nilai ini adalah apa yang Anda masukkan saat menghubungkan ke node terkelola menggunakan klien. Misalnya, **`localhost:56789`**.

Untuk informasi tentang opsi lain yang dapat Anda gunakan dengan start-session perintah, lihat start-sessiondi AWS Systems Manager bagian Referensi AWS CLI Perintah.

Memulai sesi dengan tugas Amazon ECS

Session Managermendukung memulai sesi penerusan port dengan tugas di dalam cluster Amazon Elastic Container Service (Amazon ECS). Untuk melakukannya, Anda harus memperbarui peran tugas di IAM untuk menyertakan izin berikut:

```
{
   "Version": "2012-10-17",
   "Statement": [
       {
       "Effect": "Allow",
       "Action": [
            "ssmmessages:CreateControlChannel",
            "ssmmessages:CreateDataChannel",
            "ssmmessages:OpenControlChannel",
            "ssmmessages:OpenDataChannel"
       ],
      "Resource": "*"
      }
   ]
}
```

Untuk memulai sesi penerusan port dengan tugas Amazon ECS, jalankan perintah berikut dari file. AWS CLI Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

> **Note**
>
> Hapus < and > simbol dari `target` parameter. Simbol-simbol ini disediakan hanya untuk klarifikasi pembaca.

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"
```

Memulai sesi (perintah interaktif dan noninteraktif)

Sebelum Anda memulai sesi, pastikan bahwa Anda telah menyelesaikan langkah-langkah pengaturan untukSession Manager. Untuk informasi, lihat Menyiapkan Session Manager.

Untuk menggunakan perintah AWS CLI to run session, Session Manager plugin juga harus diinstal pada mesin lokal Anda. Untuk informasi, lihat Instal Session Manager plugin untuk AWS CLI.

Untuk memulai sesi perintah interaktif, jalankan perintah berikut. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

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"
```

Untuk informasi tentang opsi lain yang dapat Anda gunakan dengan start-session perintah, lihat start-sessiondi AWS Systems Manager bagian Referensi AWS CLI Perintah.

Info lebih lanjut

- Gunakan penerusan port AWS Systems ManagerSession Manager untuk terhubung ke host jarak jauh
- Penerusan port instans Amazon EC2 dengan AWS Systems Manager
- AWS Kelola sumber daya Microsoft AD yang Dikelola dengan penerusan Session Manager port
- Port Forwarding Menggunakan AWS Systems ManagerSession Manager di Blog AWS Berita.

Mengakhiri sesi

Anda dapat menggunakan AWS Systems Manager konsol atau AWS Command Line Interface (AWS CLI) untuk mengakhiri sesi yang Anda mulai di akun Anda. Jika tidak ada aktivitas pengguna setelah 20 menit, sesi akan berakhir. Setelah sesi berakhir, aktivitas tidak dapat dilanjutkan.

Topik

- Mengakhiri sesi (konsol)
- Mengakhiri sesi (AWS CLI)

Mengakhiri sesi (konsol)

Anda dapat menggunakan AWS Systems Manager konsol untuk mengakhiri sesi di akun Anda.

Untuk mengakhiri sesi (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Session Manager.
3. Untuk Sesi, pilih tombol opsi di sebelah kiri sesi yang ingin Anda akhiri.
4. Pilih Hentikan.

Mengakhiri sesi (AWS CLI)

Untuk mengakhiri sesi menggunakan AWS CLI, jalankan perintah berikut. Ganti *id sesi* dengan informasi Anda sendiri.

```
aws ssm terminate-session \
    --session-id session-id
```

Untuk informasi lebih lanjut tentang terminate-session perintah, lihat terminate-sessiondi AWS Systems Manager bagian Referensi AWS CLI Perintah.

Lihat riwayat sesi

Anda dapat menggunakan konsol AWS Systems Manager atau AWS Command Line Interface (AWS CLI) untuk melihat informasi tentang sesi di akun Anda. Di konsol, Anda dapat melihat detail sesi seperti berikut:

- ID sesi
- Pengguna yang terhubung ke node yang dikelola melalui sesi
- ID dari node yang dikelola
- Ketika sesi dimulai dan berakhir
- Status sesi
- Lokasi yang ditentukan untuk menyimpan log sesi (jika diaktifkan)

Dengan menggunakan AWS CLI, Anda dapat melihat daftar sesi di akun Anda, namun bukan detail tambahan yang tersedia di konsol.

Untuk informasi tentang informasi riwayat sesi log, lihat Mengaktifkan dan menonaktifkan pencatatan aktivitas sesi.

Topik

Melihat riwayat sesi (konsol)

Anda dapat menggunakan AWS Systems Manager untuk melihat detail tentang sesi di akun Anda.

Untuk melihat riwayat sesi (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Session Manager.
3. Pilih tab Riwayat sesi.

   -atau-

   Jika Session Manager halaman beranda terbuka terlebih dahulu, pilih Konfigurasikan Preferensi, lalu pilih tab Riwayat sesi.

Melihat riwayat sesi (AWS CLI)

Untuk melihat daftar sesi di akun Anda menggunakan AWS CLI, jalankan perintah berikut.

```
aws ssm describe-sessions \
```

```
    --state History
```

> **Note**
>
> Perintah ini hanya mengembalikan hasil untuk koneksi ke target yang dimulai menggunakan. Session Manager Perintah ini bukan mendaftarkan koneksi yang dibuat melalui cara lain, seperti Remote Desktop Protocol (RDP) atau Secure Shell Protocol (SSH).

Untuk informasi tentang pilihan lain yang dapat Anda gunakan dengan describe-sessions perintah, lihat describe-sessionsdi AWS Systems Manager bagian Referensi AWS CLI Perintah.

Mengaudit aktivitas sesi

Selain memberi informasi tentang sesi saat ini dan menyelesaikan sesi di konsol Systems Manager,Session Manager memberi Anda kemampuan untuk mengaudit aktivitas sesi di AndaAkun AWS menggunakanAWS CloudTrail.

CloudTrail menangkap panggilan API sesi melalui konsol Systems Manager,AWS Command Line Interface (AWS CLI), dan SDK Systems Manager. Anda dapat melihat informasi tersebut di CloudTrail konsol atau menyimpannya di bucket Amazon Simple Storage Service (Amazon S3) yang ditentukan. Satu bucket Amazon S3 digunakan untuk semua CloudTrail log untuk akun Anda. Untuk informasi selengkapnya, lihat Pencatatan panggilan AWS Systems Manager API dengan AWS CloudTrail.

Memantau aktivitas sesi sesi sesi menggunakan Amazon EventBridge (konsol)

Dengan EventBridge itu, Anda dapat mengatur aturan untuk mendeteksi ketika perubahan terjadi padaAWS sumber daya. Anda dapat membuat aturan untuk mendeteksi ketika pengguna di organisasi Anda memulai atau mengakhiri sesi, dan kemudian, misalnya, menerima pemberitahuan melalui Amazon SNS tentang acara tersebut.

EventBridge dukungan untukSession Manager bergantung pada log operasi API yang dicatat oleh CloudTrail. (Anda dapat menggunakan CloudTrail integrasi dengan EventBridge untuk menanggapi sebagian besarAWS Systems Manager kejadian.) Tindakan yang berlangsung dalam sesi, sepertiexit perintah, yang tidak membuat panggilan API tidak terdeteksi oleh EventBridge.

Langkah-langkah berikut menguraikan cara untuk memulai pemberitahuan melalui Amazon Simple Notification Service (Amazon SNS) ketika kejadianSession Manager API terjadi, sepertiStartSession.

Untuk memantau aktivitas sesi sesi menggunakan Amazon EventBridge (konsol)

1. Buat topik Amazon SNS untuk digunakan untuk mengirim pemberitahuan ketikaSession Manager kejadian terjadi bahwa Anda ingin melacak.

   Untuk informasi lebih lanjut, lihat Buat topik dalam Panduan Developer Amazon Simple Notification Service.
2. Buat EventBridge aturan untuk mengaktifkan target Amazon SNS untuk tipeSession Manager acara yang ingin Anda lacak.

   Untuk informasi tentang cara membuat aturan, lihat Membuat EventBridge aturan Amazon yang bereaksi terhadap kejadian di Panduan EventBridge Pengguna Amazon.

   Saat Anda mengikuti langkah-langkah untuk membuat aturan, buat pilihan berikut:

   - Untuk AWSlayanan, pilih Systems Manager.
   - Untuk jenis Event, pilih AWSAPI Call through CloudTrail.
   - Pilih Operasi khusus, dan kemudian masukkanSession Manager perintah atau perintah (satu per satu) yang Anda ingin menerima pemberitahuannya. Anda dapat memilih StartSession, ResumeSession, dan TerminateSession. (EventBridge tidak mendukung`Get*`, `List*`, dan`Describe*` perintah.)
   - Untuk Pilih target, pilih topik SNS. Untuk Topik, pilih nama topik Amazon SNS yang Anda buat di Langkah 1.

Untuk informasi selengkapnya, lihat Panduan EventBridge Pengguna Amazon dan Panduan Memulai Layanan Pemberitahuan Sederhana Amazon.

Mengaktifkan dan menonaktifkan pencatatan aktivitas sesi

Selain memberikan informasi tentang sesi saat ini dan yang telah selesai di konsol Systems Manager, Session Manager memberi Anda opsi untuk mencatat aktivitas sesi di konsol Anda Akun AWS. Hal ini mengizinkan Anda untuk melakukan hal berikut:

- Membuat dan menyimpan log sesi untuk tujuan arsip.
- Buat laporan yang menunjukkan detail setiap koneksi yang dibuat ke node terkelola yang Anda gunakan Session Manager selama 30 hari terakhir.
- Buat notifikasi aktivitas sesi di Anda Akun AWS, seperti notifikasi Amazon Simple Notification Service (Amazon SNS).

- Secara otomatis memulai tindakan lain pada AWS sumber daya sebagai hasil dari aktivitas sesi, seperti menjalankan AWS Lambda fungsi, memulai AWS CodePipeline pipeline, atau menjalankan AWS Systems Manager Run Command dokumen.

Important

Perhatikan persyaratan dan batasan berikut untukSession Manager:

- Session Managermencatat perintah yang Anda masukkan dan outputnya selama sesi tergantung pada preferensi sesi Anda. Untuk mencegah data sensitif, seperti kata sandi, dilihat di log sesi Anda, sebaiknya gunakan perintah berikut saat memasukkan data sensitif selama sesi.

  Linux & macOS

  ```
  stty -echo; read passwd; stty echo;
  ```

  Windows

  ```
  $Passwd = Read-Host -AsSecureString
  ```

- Jika Anda menggunakan Windows Server 2012 atau sebelumnya, data di log Anda mungkin tidak diformat secara optimal. Kami merekomendasikan menggunakan Windows Server 2012 R2 dan yang lebih baru untuk format log yang optimal.
- Jika Anda menggunakan Linux atau macOS mengelola node, pastikan bahwa utilitas layar diinstal. Jika tidak, data log Anda mungkin terpotong. Di Amazon Linux 1, Amazon Linux 2, AL2023 danUbuntu Server, utilitas layar diinstal secara default. Untuk menginstal layar secara manual, tergantung pada versi AndaLinux, jalankan salah satu `sudo yum install screen` atau`sudo apt-get install screen`.
- Logging tidak tersedia untuk Session Manager sesi yang terhubung melalui port forwarding atau SSH. Ini karena SSH mengenkripsi semua data sesi, dan Session Manager hanya berfungsi sebagai terowongan untuk koneksi SSH.

Untuk informasi selengkapnya tentang izin yang diperlukan untuk menggunakan Amazon S3 atau CloudWatch Amazon Log untuk mencatat data sesi, lihat. Membuat peran IAM dengan izin untuk Session Manager dan Amazon S3 dan CloudWatch Log (konsol)

Lihat topik berikut untuk informasi selengkapnya tentang opsi pencatatanSession Manager.

Topik

Streaming data sesi menggunakan Amazon CloudWatch Logs (konsol)

Anda dapat mengirim aliran log data sesi secara terus-menerus ke Amazon CloudWatch Logs. Detail penting, seperti perintah yang dijalankan pengguna dalam sesi, ID pengguna yang menjalankan perintah, dan stempel waktu saat data sesi dialirkan ke CloudWatch Log, disertakan saat streaming data sesi. Saat streaming data sesi, log diformat JSON untuk membantu Anda berintegrasi dengan solusi log Anda yang ada. Streaming data sesi tidak didukung untuk perintah interaktif.

> **Note**
>
> Untuk melakukan streaming data sesi dari node Windows Server terkelola, Anda harus menginstal PowerShell 5.1 atau yang lebih baru. Secara default, Windows Server 2016 dan yang lebih baru memiliki PowerShell versi yang diperlukan diinstal. Namun, Windows Server 2012 dan 2012 R2 tidak memiliki PowerShell versi yang diperlukan diinstal secara default. Jika Anda belum memperbarui PowerShell node terkelola Windows Server 2012 atau 2012 R2 Anda, Anda dapat melakukannya dengan menggunakanRun Command. Untuk informasi tentang memperbarui PowerShell penggunaanRun Command, lihatMemperbarui PowerShell menggunakan Run Command.

> **Important**
>
> Jika setelan kebijakan PowerShell Transkripsi dikonfigurasi pada node Windows Server terkelola, Anda tidak akan dapat mengalirkan data sesi.

Untuk melakukan streaming data sesi menggunakan Amazon CloudWatch Logs (konsol)

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.

2. Di panel navigasi, pilih Session Manager.
3. Pilih tab Preferensi, dan kemudian pilih Edit.
4. Pilih kotak centang di sebelah Aktifkan di bawah CloudWatch pencatatan.
5. Pilih opsi Streaming log sesi.
6. (Disarankan) Pilih kotak centang di sebelah Izinkan hanya grup CloudWatch log terenkripsi. Dengan opsi ini diaktifkan, data log dienkripsi menggunakan kunci enkripsi sisi server yang ditentukan untuk grup log. Jika Anda tidak ingin mengenkripsi data log yang dikirim ke CloudWatch Log, kosongkan kotak centang. Anda juga harus mengosongkan kotak centang jika enkripsi tidak diizinkan di grup log.
7. Untuk CloudWatch log, untuk menentukan grup CloudWatch log Log yang ada di log sesi Akun AWS untuk Anda unggah, pilih salah satu dari berikut ini:
   - Masukkan nama grup log di kotak teks yang telah dibuat di akun Anda untuk menyimpan data log sesi.
   - Jelajahi grup log: Pilih grup log yang telah dibuat di akun Anda untuk menyimpan data log sesi.
8. Pilih Simpan.

Log data sesi menggunakan Amazon S3 (konsol)

Anda dapat memilih untuk menyimpan data log sesi di bucket Amazon Simple Storage Service (Amazon S3) yang ditentukan untuk tujuan debugging dan pemecahan masalah. Opsi default adalah untuk log yang akan dikirim ke bucket Amazon S3 yang dienkripsi. Enkripsi dilakukan menggunakan kunci yang ditentukan untuk bucket, baik kunci Amazon S3 Server-Side Encryption (SSE) (AES-256) AWS KMS key atau Amazon S3.

> **Important**
>
> Saat Anda menggunakan bucket cara hosting virtual dengan Lapisan Soket Aman (SSL), sertifikat wildcard SSL hanya cocok dengan bucket yang tidak mengandung titik. Untuk menangani hal ini, gunakan HTTP atau tulis logika verifikasi sertifikat Anda sendiri. Kami menyarankan Anda untuk tidak menggunakan titik (".") dalam nama bucket saat menggunakan bucket cara hosting virtual.

Enkripsi bucket Amazon S3

Untuk mengirim log ke bucket Amazon S3 Anda dengan enkripsi, enkripsi harus diizinkan di bucket. Untuk informasi lebih lanjut tentang enkripsi bucket Amazon S3, lihat Enkripsi Default Amazon S3 untuk Bucket S3.

Kunci yang dikelola pelanggan

Jika Anda menggunakan kunci KMS yang Anda kelola sendiri untuk mengenkripsi bucket Anda, maka profil instans IAM yang terlampir pada instans Anda harus memiliki izin eksplisit untuk membaca kunci. Jika Anda menggunakan Kunci yang dikelola AWS, instance tidak memerlukan izin eksplisit ini. Untuk informasi lebih lanjut tentang memberikan profil instans dengan akses untuk menggunakan kunci, lihat Mengizinkan Pengguna Kunci untuk Menggunakan kunci di Panduan Developer AWS Key Management Service .

Ikuti langkah-langkah berikut Session Manager untuk mengonfigurasi penyimpanan log sesi di bucket Amazon S3.

> **Note**
>
> Anda juga dapat menggunakan AWS CLI untuk menentukan atau mengubah bucket Amazon S3 tempat data sesi dikirim. Untuk informasi, lihat PerbaruiSession Manager preferensi (baris perintah).

Untuk log data sesi menggunakan Amazon S3 (konsol)

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Session Manager.
3. Pilih tab Preferensi, dan kemudian pilih Edit.
4. Pilih kotak centang di samping Aktifkan di bawah log S3.
5. (Disarankan) Pilih kotak centang di sebelah Izinkan hanya bucket S3 yang dienkripsi. Dengan opsi ini diaktifkan, data log dienkripsi menggunakan kunci enkripsi sisi server yang ditentukan untuk bucket. Jika Anda tidak ingin mengenkripsi data log yang dikirim ke Amazon S3, kosongkan kotak centang. Anda juga harus mengosongkan kotak centang jika enkripsi tidak diizinkan di bucket S3.
6. Untuk nama bucket S3, pilih salah satu dari berikut ini:

> **Note**
>
> Kami menyarankan Anda untuk tidak menggunakan titik (".") dalam nama bucket saat menggunakan bucket cara hosting virtual. Untuk informasi selengkapnya tentang konvensi penamaan ember Amazon S3, lihat Pembatasan dan Batasan Bucket di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

- Pilih nama bucket dari daftar: Pilih bucket Amazon S3 yang telah dibuat di akun Anda untuk menyimpan data log sesi.
- Masukkan nama bucket di kotak teks: Masukkan nama bucket Amazon S3 yang telah dibuat di akun Anda untuk menyimpan data log sesi.

7. (Opsional) Untuk prefiks kunci S3, masukkan nama folder yang sudah ada atau baru untuk menyimpan log dalam bucket yang dipilih.
8. Pilih Simpan.

Untuk informasi selengkapnya tentang bekerja dengan bucket Amazon S3 dan Amazon S3, lihat Panduan Pengguna Layanan Penyimpanan Sederhana Amazon dan Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

Data sesi logging menggunakan Amazon CloudWatch Logs (konsol)

Dengan Amazon CloudWatch Logs, Anda dapat memantau, menyimpan, dan mengakses file log dari berbagai file Layanan AWS. Anda dapat mengirim data log sesi ke grup CloudWatch log Log untuk tujuan debugging dan pemecahan masalah. Opsi default adalah untuk data log yang akan dikirim dengan enkripsi menggunakan kunci KMS Anda, tetapi Anda dapat mengirim data ke grup log Anda dengan atau tanpa enkripsi.

Ikuti langkah-langkah berikut AWS Systems Manager Session Manager untuk mengonfigurasi pengiriman data log sesi ke grup CloudWatch log Log di akhir sesi Anda.

> **Note**
>
> Anda juga dapat menggunakan AWS CLI untuk menentukan atau mengubah grup CloudWatch log Log tempat data sesi dikirim. Untuk informasi, lihat PerbaruiSession Manager preferensi (baris perintah).

Untuk mencatat data sesi menggunakan Amazon CloudWatch Logs (konsol)

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Session Manager.
3. Pilih tab Preferensi, dan kemudian pilih Edit.
4. Pilih kotak centang di sebelah Aktifkan di bawah CloudWatch pencatatan.
5. Pilih opsi Unggah log sesi.
6. (Disarankan) Pilih kotak centang di sebelah Izinkan hanya grup CloudWatch log terenkripsi. Dengan opsi ini diaktifkan, data log dienkripsi menggunakan kunci enkripsi sisi server yang ditentukan untuk grup log. Jika Anda tidak ingin mengenkripsi data log yang dikirim ke CloudWatch Log, kosongkan kotak centang. Anda juga harus mengosongkan kotak centang jika enkripsi tidak diizinkan di grup log.
7. Untuk CloudWatch log, untuk menentukan grup CloudWatch log Log yang ada di log sesi Akun AWS untuk Anda unggah, pilih salah satu dari berikut ini:
   - Pilih grup log dari daftar: Pilih grup log yang telah dibuat di akun Anda untuk menyimpan data log sesi.
   - Masukkan nama grup log di kotak teks: Masukkan nama grup log yang telah dibuat di akun Anda untuk menyimpan data log sesi.
8. Pilih Simpan.

Untuk informasi selengkapnya tentang bekerja dengan CloudWatch Log, lihat Panduan Pengguna CloudWatch Log Amazon.

Menonaktifkan pencatatan Session Manager aktivitas di CloudWatch Log dan Amazon S3

Anda dapat menggunakan konsol Systems Manager atau AWS CLI untuk menonaktifkan aktivitas sesi masuk ke akun Anda.

Untuk menonaktifkan pencatatan aktivitas sesi (konsol)

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Session Manager.
3. Pilih tab Preferensi, dan kemudian pilih Edit.

4. Untuk menonaktifkan CloudWatch logging, di bagian CloudWatch logging, kosongkan kotak centang Aktifkan.
5. Untuk menonaktifkan logging S3, di bagian logging S3, kosongkan kotak centang Aktifkan.
6. Pilih Simpan.

Untuk menonaktifkan pencatatan aktivitas sesi (AWS CLI)

Untuk menonaktifkan pencatatan aktivitas sesi menggunakan AWS CLI, ikuti petunjuk diPerbaruiSession Manager preferensi (baris perintah).

Dalam file JSON Anda, pastikan bahwa `s3BucketName` dan `cloudWatchLogGroupName` input tidak mengandung nilai. Sebagai contoh:

```
"inputs": {
        "s3BucketName": "",
        ...
        "cloudWatchLogGroupName": "",
        ...
    }
```

Atau, Anda dapat menghapus semua `S3*` dan `cloudWatch*` input dari file JSON Anda untuk menonaktifkan logging.

Skema dokumen sesi

Informasi berikut menjelaskan elemen skema dari dokumen Sesi.AWS Systems Manager Session ManagerMenggunakan dokumen Sesi untuk menentukan jenis sesi yang akan dimulai, seperti sesi standar, sesi penerusan port, atau sesi untuk menjalankan perintah interaktif.

schemaVersion

Versi skema dokumen Sesi. Dokumen Sesi hanya mendukung versi 1.0.

Tipe: String

Wajib: Ya

description

Deskripsi yang Anda tentukan untuk dokumen Sesi. Misalnya, “Dokumen untuk memulai sesi penerusan port denganSession Manager“.

Tipe: String

Wajib: Tidak

sessionType

Tipe sesi dokumen Sesi digunakan untuk membuat.

Tipe: String

Wajib: Ya

Nilai valid: `InteractiveCommands` | `NonInteractiveCommands` | `Port` | `Standard_Stream`

inputs

Preferensi sesi yang akan digunakan untuk sesi yang dibuat menggunakan dokumen Sesi ini. Elemen ini diperlukan untuk dokumen Sesi yang digunakan untuk membuat sesi `Standard_Stream`.

Jenis: StringMap

Wajib: Tidak

s3BucketName

Bucket Amazon Simple Storage Service (Amazon S3) yang ingin Anda kirimkan log sesi di akhir sesi Anda.

Tipe: String

Wajib: Tidak

s3KeyPrefix

Prefiks yang digunakan saat mengirim log ke bucket Amazon S3 yang Anda tentukan di input `s3BucketName`. Untuk informasi selengkapnya tentang menggunakan awalan bersama dengan objek yang disimpan di Amazon S3, lihatBagaimana cara menggunakan folder dalam ember S3?diPanduan Pengguna Layanan Penyimpanan Sederhana Amazon.

Tipe: String

Wajib: Tidak

s3EncryptionEnabled

Jika diatur ke `true`, bucket Amazon S3 yang Anda tentukan di input `s3BucketName` harus dienkripsi.

Tipe: Boolean

Wajib: Ya

cloudWatchLogGroupName

Nama Amazon CloudWatch Log (CloudWatch Log) grup yang ingin Anda kirimi log sesi di akhir sesi Anda.

Tipe: String

Wajib: Tidak

cloudWatchEncryptionEnabled

Jika diatur ke `true`, grup log yang Anda tentukan dalam input `cloudWatchLogGroupName` harus dienkripsi.

Tipe: Boolean

Wajib: Ya

cloudWatchStreamingEnabled

Jika diatur ke `true`, aliran terus-menerus sesi data log dikirim ke grup log yang Anda tentukan di input `cloudWatchLogGroupName`. Jika diatur ke `false`, log sesi dikirim ke grup log yang Anda tentukan di input `cloudWatchLogGroupName` di akhir sesi Anda.

Tipe: Boolean

Wajib: Ya

kmsKeyId

ID dariAWS KMS keyAnda ingin menggunakan untuk mengenkripsi data lebih lanjut antara mesin klien lokal Anda dan node terkelola Amazon Elastic Compute Cloud (Amazon EC2) yang Anda sambungkan.

Tipe: String

Wajib: Tidak

runAsEnabled

Jika disetel ke`true`, Anda harus menentukan akun pengguna yang ada di node terkelola yang akan Anda sambungkan di`runAsDefaultUser`masukan. Jika tidak, sesi akan gagal untuk memulai. Secara default, sesi dimulai menggunakan`ssm-user`Akun yang dibuat olehAWS Systems Manager SSM Agent. Fitur Run As hanya didukung untuk menghubungkan keLinuxnode terkelola.

Jenis: Boolean

Wajib: Ya

runAsDefaultUser

Nama akun pengguna untuk memulai sesi dengan onLinuxnode terkelola saat`runAsEnabled`masukan diatur ke`true`. Akun pengguna yang Anda tentukan untuk input ini harus ada pada node terkelola yang akan Anda sambungkan; jika tidak, sesi akan gagal dimulai.

Tipe: String

Wajib: Tidak

idleSessionTimeout

Jumlah waktu tidak aktif yang ingin Anda izinkan sebelum sesi berakhir. Input ini diukur dalam hitungan menit.

Tipe: String

Nilai valid: 1–60

Wajib: Tidak

maxSessionDuration

Jumlah maksimum waktu yang ingin Anda izinkan sebelum sesi berakhir. Input ini diukur dalam hitungan menit.

Jenis: String

Nilai yang valid: 1-1440

Wajib: Tidak

shellProfile

Preferensi yang Anda tentukan per sistem operasi untuk diterapkan dalam sesi seperti preferensi shell, variabel lingkungan, direktori kerja, dan menjalankan beberapa perintah ketika sesi dimulai.

Jenis: StringMap

Wajib: Tidak

windows

Preferensi shell, variabel lingkungan, direktori kerja, dan perintah yang Anda tentukan untuk sesiWindowsnode terkelola.

Tipe: String

Wajib: Tidak

linux

Preferensi shell, variabel lingkungan, direktori kerja, dan perintah yang Anda tentukan untuk sesiLinuxnode terkelola.

Tipe: String

Wajib: Tidak

parameters

Objek yang menentukan parameter yang diterima dokumen. Untuk informasi lebih lanjut tentang menentukan parameter dokumen, lihat parameter di Elemen data tingkat atas. Untuk parameter yang sering Anda referensikan, kami sarankan Anda menyimpan parameter tersebut di Manajer SistemParameter Storedan kemudian mereferensikan mereka. Anda bisa referensi`String`dan`StringList` Parameter Storeparameter di bagian dokumen ini. Anda tidak bisa referensi`SecureString` Parameter Storeparameter di bagian dokumen ini. Anda dapat mereferensikanParameter Storeparameter menggunakan format berikut.

```
{{ssm:parameter-name}}
```

Untuk informasi selengkapnya tentang Parameter Store, lihat AWS Systems Manager Parameter Store.

Jenis: StringMap

Wajib: Tidak

properties

Objek yang nilainya Anda tentukan yang digunakan dalam operasi API `StartSession`.

Untuk dokumen Sesi yang digunakan untuk sesi `InteractiveCommands`, objek properti termasuk perintah untuk berjalan pada sistem operasi yang Anda tentukan. Anda juga dapat menentukan apakah perintah dijalankan sebagai`root`menggunakan`runAsElevated`properti boolean. Untuk informasi selengkapnya, lihat Membatasi akses ke perintah dalam sesi.

Untuk dokumen Sesi yang digunakan untuk sesi `Port`, objek properti berisi nomor port tempat lalu lintas harus diarahkan. Sebagai contoh, lihat contoh dokumen Sesi tipe `Port` nanti di topik ini.

Jenis: StringMap

Wajib: Tidak

Contoh dokumen sesi tipe `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"
        }
    }
}
```

Contoh dokumen sesi tipe `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
        }
    }
}
```

Contoh dokumen sesi tipe `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"
    }
}
```

Contoh dokumen sesi dengan karakter khusus

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
      }
   }
}
```

Pemecahan Masalah Session Manager

Gunakan informasi berikut untuk membantu Anda memecahkan masalah. AWS Systems Manager Session Manager

Topik

- Session Managertidak dapat terhubung dari konsol Amazon EC2
- Tidak ada izin untuk memulai sesi
- Tidak ada izin untuk mengubah preferensi sesi
- Node terkelola tidak tersedia atau tidak dikonfigurasi untuk Session Manager
- Session ManagerPlugin tidak ditemukan
- Session Managerplugin tidak secara otomatis ditambahkan ke jalur baris perintah (Windows)
- Session ManagerPlugin menjadi tidak responsif
- TargetNotConnected
- Layar kosong ditampilkan setelah memulai sesi
- Node terkelola menjadi tidak responsif selama sesi berjalan lama

- Terjadi kesalahan (InvalidDocument) saat memanggil StartSession operasi

Session Managertidak dapat terhubung dari konsol Amazon EC2

Masalah: Setelah membuat instance baru, tab Session Manager di konsol Amazon Elastic Compute Cloud (Amazon EC2) tidak memberi Anda opsi untuk terhubung.

Solusi A: Buat profil instance: Jika Anda belum melakukannya (seperti yang diinstruksikan oleh informasi pada tab Session Manager di konsol EC2), buat profil instans AWS Identity and Access Management (IAM) dengan menggunakan. Quick Setup Quick Setupadalah kemampuan AWS Systems Manager.

Session Managermemerlukan profil instans IAM untuk terhubung ke instans Anda. Anda dapat membuat profil instans dan menetapkannya ke instans Anda dengan membuat konfigurasi manajemen host denganQuick Setup. Konfigurasi manajemen host membuat profil instance dengan izin yang diperlukan dan menetapkannya ke instans Anda. Konfigurasi manajemen host juga memungkinkan kemampuan Systems Manager lainnya dan menciptakan peran IAM untuk menjalankan kemampuan tersebut. Tidak ada biaya untuk menggunakan Quick Setup atau kemampuan yang diaktifkan oleh konfigurasi manajemen host. Buka Quick Setup dan buat konfigurasi manajemen host.

> **Important**
>
> Setelah Anda membuat konfigurasi manajemen host, Amazon EC2 dapat mengambil beberapa menit untuk mendaftarkan perubahan dan menyegarkan tab Session Manager. Jika tab tidak menampilkan tombol Connect setelah dua menit, reboot instance Anda. Setelah reboot, jika Anda masih tidak melihat opsi untuk terhubung, buka Quick Setup dan verifikasi bahwa Anda hanya memiliki satu konfigurasi manajemen host. Jika ada dua, hapus konfigurasi yang lebih lama dan tunggu beberapa menit.

Jika Anda masih tidak dapat terhubung setelah membuat konfigurasi manajemen host, atau jika Anda menerima kesalahan, termasuk kesalahanSSM Agent, lihat salah satu solusi berikut:

- Solusi B: Tidak ada kesalahan, tetapi masih tidak dapat terhubung
- Solusi C: Kesalahan tentang hilang SSM Agent

Solusi B: Tidak ada kesalahan, tetapi masih tidak dapat terhubung

Jika Anda membuat konfigurasi manajemen host, menunggu beberapa menit sebelum mencoba terhubung, dan masih tidak dapat terhubung, maka Anda mungkin perlu menerapkan konfigurasi manajemen host secara manual ke instans Anda. Gunakan prosedur berikut untuk memperbarui konfigurasi manajemen Quick Setup host dan menerapkan perubahan pada instance.

Untuk memperbarui konfigurasi manajemen host menggunakan Quick Setup

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Quick Setup.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Quick Setupdi panel navigasi.
3. Dalam daftar Konfigurasi, pilih konfigurasi Manajemen Host yang Anda buat.
4. Pilih Tindakan, lalu pilih Edit konfigurasi.
5. Di bagian Target, pilih Manual.
6. Di bagian Instances, pilih instance yang Anda buat.
7. Pilih Perbarui.

Tunggu beberapa menit hingga EC2 menyegarkan tab Session Manager. Jika Anda masih tidak dapat terhubung atau jika Anda menerima kesalahan, tinjau solusi yang tersisa untuk masalah ini.

Solusi C: Kesalahan tentang hilang SSM Agent

Jika Anda tidak dapat membuat konfigurasi manajemen host dengan menggunakanQuick Setup, atau jika Anda menerima kesalahan tentang SSM Agent tidak diinstal, Anda mungkin perlu menginstal secara manual SSM Agent pada instans Anda. SSM Agentadalah perangkat lunak Amazon yang memungkinkan Systems Manager terhubung ke instans Anda dengan menggunakanSession Manager. SSM Agentdiinstal secara default di sebagian besar Gambar Mesin Amazon (AMI). Jika instans Anda dibuat dari AMI non-standar atau AMI yang lebih lama, Anda mungkin harus menginstal agen secara manual. Untuk prosedur yang akan diinstalSSM Agent, lihat topik berikut yang sesuai dengan sistem operasi instans Anda.

- Windows Server

- macOS
- AlmaLinux
- Amazon Linux 1
- Amazon Linux 2 dan AL2023
- CentOS
- CentOS Stream
- Debian Server
- Oracle Linux
- Red Hat Enterprise Linux
- Rocky Linux
- SUSE Linux Enterprise Server
- Ubuntu Server

Untuk masalah denganSSM Agent, lihatPemecahan Masalah SSM Agent.

Tidak ada izin untuk memulai sesi

Masalah: Anda mencoba untuk memulai sesi, tetapi sistem memberitahu Anda bahwa Anda tidak memiliki izin yang diperlukan.

- Solusi: Administrator sistem belum memberi Anda izin kebijakan AWS Identity and Access Management (IAM) untuk memulai Session Manager sesi. Untuk informasi, lihat Kontrol akses sesi pengguna ke instans.

Tidak ada izin untuk mengubah preferensi sesi

Masalah: Anda mencoba untuk memperbarui preferensi sesi global untuk organisasi Anda, tetapi sistem memberitahu Anda bahwa Anda tidak memiliki izin yang diperlukan.

- Solusi: Administrator sistem belum memberi Anda izin kebijakan IAM untuk menyetel Session Manager preferensi. Untuk informasi, lihat Berikan atau Tolak izin pengguna untuk memperbarui atau Tolak izin penggunaSession Managerpilihan.

Node terkelola tidak tersedia atau tidak dikonfigurasi untuk Session Manager

Masalah 1: Anda ingin memulai sesi di halaman Mulai konsol sesi, tetapi node terkelola tidak ada dalam daftar.

- Solusi A: Node terkelola yang ingin Anda sambungkan mungkin belum dikonfigurasi AWS Systems Manager. Untuk informasi selengkapnya, lihat Menyiapkan AWS Systems Manager.

  > **Note**
  >
  > Jika AWS Systems Manager SSM Agent sudah berjalan pada node terkelola saat Anda melampirkan profil instans IAM, Anda mungkin perlu memulai ulang agen sebelum instance terdaftar di halaman Mulai konsol sesi.

- Solusi B: Konfigurasi proxy yang Anda terapkan SSM Agent pada node terkelola Anda mungkin salah. Jika konfigurasi proxy salah, node terkelola tidak akan dapat mencapai titik akhir layanan yang diperlukan, atau node mungkin melaporkan sebagai sistem operasi yang berbeda dengan Systems Manager. Lihat informasi yang lebih lengkap di Mengkonfigurasi SSM Agent untuk menggunakan proxy (Linux) dan SSM AgentKonfigurasikan untuk menggunakan proxy untuk Windows Server instance.

Masalah 2: Node terkelola yang ingin Anda sambungkan ada dalam daftar di halaman Mulai konsol sesi, tetapi halaman tersebut melaporkan bahwa “Instance yang Anda pilih tidak dikonfigurasi untuk digunakanSession Manager.”

- Solusi A: Node terkelola telah dikonfigurasi untuk digunakan dengan layanan Systems Manager, tetapi profil instans IAM yang dilampirkan ke node mungkin tidak menyertakan izin untuk kemampuan tersebutSession Manager. Untuk selengkapnya, lihat Memverifikasi atau Membuat Profil Instans IAM dengan Session Manager Izin.
- Solusi B: Node terkelola tidak menjalankan versi SSM Agent yang mendukungSession Manager. Perbarui SSM Agent pada node ke versi 2.3.68.0 atau yang lebih baru.

  Perbarui SSM Agent secara manual pada node terkelola dengan mengikuti langkah-langkah dalam Menginstal dan menghapus instalasi secara manual SSM Agent pada instans EC2 untuk Windows ServerMenginstal secara manual SSM Agent pada instans EC2 untuk Linux,Bekerja dengan SSM Agent instans EC2 untuk macOS, atau, tergantung pada sistem operasi.

Atau, gunakan Run Command dokumen `AWS-UpdateSSMAgent` untuk memperbarui versi agen pada satu atau lebih node terkelola pada satu waktu. Untuk informasi, lihat Memperbarui SSM Agent penggunaan Run Command.

> **Tip**
>
> Untuk selalu memperbarui agen Anda, kami sarankan memperbarui SSM Agent ke versi terbaru pada jadwal otomatis yang Anda tentukan menggunakan salah satu metode berikut:
>
> - Jalankan `AWS-UpdateSSMAgent` sebagai bagian dari State Manager asosiasi. Untuk informasi, lihat Walkthrough: Perbarui secara otomatis (SSM AgentCLI).
> - Jalankan `AWS-UpdateSSMAgent` sebagai bagian dari jendela pemeliharaan. Untuk informasi tentang bekerja dengan jendela pemeliharaan, lihat Menggunakan windows pemeliharaan (konsol) dan Tutorial: Membuat dan mengonfigurasi jendela pemeliharaan (AWS CLI).

- Solusi C: Node terkelola tidak dapat mencapai titik akhir layanan yang diperlukan. Anda dapat meningkatkan postur keamanan node terkelola dengan menggunakan titik akhir antarmuka yang didukung oleh AWS PrivateLink untuk terhubung ke titik akhir Systems Manager. Alternatif untuk menggunakan titik akhir antarmuka adalah mengizinkan akses internet keluar pada node terkelola Anda. Untuk informasi selengkapnya, lihat Menggunakan PrivateLink untuk menyiapkan titik akhir VPC. Session Manager
- Solusi D: Node yang dikelola memiliki sumber daya CPU atau memori yang terbatas. Meskipun node terkelola Anda mungkin berfungsi, jika node tidak memiliki cukup sumber daya yang tersedia, Anda tidak dapat membuat sesi. Untuk informasi selengkapnya, lihat Memecahkan masalah instans yang tidak dapat dijangkau.

Session ManagerPlugin tidak ditemukan

Untuk menggunakan perintah AWS CLI to run session, Session Manager plugin juga harus diinstal pada mesin lokal Anda. Untuk informasi, lihat Instal Session Manager plugin untuk AWS CLI.

Session Managerplugin tidak secara otomatis ditambahkan ke jalur baris perintah (Windows)

Ketika Anda menginstal Session Manager pluginWindows, `session-manager-plugin` executable harus secara otomatis ditambahkan ke variabel PATH lingkungan sistem operasi Anda. Jika perintah gagal setelah Anda menjalankannya untuk memeriksa apakah Session Manager plugin diinstal dengan benar (`aws ssm start-session --target` *`instance-id`*), Anda mungkin perlu mengaturnya secara manual menggunakan prosedur berikut.

Untuk memodifikasi variabel PATH Anda (Windows)

1. Tekan Windows tombol dan masukkan**`environment variables`**.
2. Pilih Mengedit variabel lingkungan untuk akun Anda.
3. Pilih PATH dan kemudian pilih Edit.
4. Tambahkan jalur ke bidang Nilai variabel, dipisahkan oleh titik koma, seperti yang ditunjukkan dalam contoh ini: *`C:\existing\path`*;***`C:\new\path`***

   `C:\existing\path` mewakili nilai yang sudah ada di lapangan. ***`C:\new\path`*** mewakili jalur yang ingin Anda tambahkan, seperti yang ditunjukkan dalam contoh-contoh ini.

   - Mesin 64-bit: `C:\Program Files\Amazon\SessionManagerPlugin\bin\`
   - Mesin 32-bit: `C:\Program Files (x86)\Amazon\SessionManagerPlugin\bin\`
5. Pilih OK dua kali untuk menggunakan pengaturan baru.
6. Tutup semua prompt perintah yang berjalan dan buka kembali.

Session ManagerPlugin menjadi tidak responsif

Selama sesi penerusan port, lalu lintas mungkin berhenti meneruskan jika Anda memiliki perangkat lunak antivirus yang diinstal pada komputer lokal Anda. Dalam beberapa kasus, perangkat lunak antivirus mengganggu Session Manager plugin yang menyebabkan kebuntuan proses. Untuk mengatasi masalah ini, izinkan atau kecualikan Session Manager plugin dari perangkat lunak antivirus. Untuk informasi tentang jalur instalasi default untuk Session Manager plugin, lihatInstal Session Manager plugin untuk AWS CLI.

TargetNotConnected

Masalah: Anda mencoba memulai sesi, tetapi sistem mengembalikan pesan kesalahan, “Terjadi kesalahan (TargetNotConnected) saat memanggil StartSession operasi: *`InstanceId`* tidak terhubung.”

- Solusi A: Kesalahan ini dikembalikan ketika node terkelola target yang ditentukan untuk sesi tidak sepenuhnya dikonfigurasi untuk digunakan dengan Session Manager. Untuk informasi, lihat Menyiapkan Session Manager.
- Solusi B: Kesalahan ini juga dikembalikan jika Anda mencoba memulai sesi pada node terkelola yang terletak di node yang berbeda Akun AWS atau Wilayah AWS.

Layar kosong ditampilkan setelah memulai sesi

Masalah: Anda memulai sesi dan Session Manager menampilkan layar kosong.

- Solusi A: Masalah ini dapat terjadi ketika volume root pada node terkelola penuh. Karena kurangnya ruang disk, SSM Agent pada node berhenti bekerja. Untuk mengatasi masalah ini, gunakan Amazon CloudWatch untuk mengumpulkan metrik dan log dari sistem operasi. Untuk informasi, lihat Memantau memori dan metrik disk untuk instans Linux Amazon EC2 atau Memantau memori dan metrik disk untuk instans Windows Amazon EC2.
- Solusi B: Layar kosong mungkin tampil jika Anda mengakses konsol menggunakan tautan yang menyertakantitik akhir dan pasangan Wilayah yang tidak cocok. Sebagai contoh, dalam URL konsol berikut, `us-west-2` adalah titik akhir yang ditentukan, tapi `us-west-1` adalah Wilayah AWS yang ditentukan.

  ```
  https://us-west-2.console.aws.amazon.com/systems-manager/session-manager/sessions?
  region=us-west-1
  ```

- Solusi C: Node terkelola terhubung ke Systems Manager menggunakan titik akhir VPC, dan Session Manager preferensi Anda menulis output sesi ke bucket Amazon S3 atau grup log CloudWatch Amazon Logs, tetapi `s3` titik akhir gateway `logs` atau titik akhir antarmuka tidak ada di VPC. `s3`Titik akhir dalam format **`com.amazonaws.`*`region`*`.s3`**diperlukan jika node terkelola Anda terhubung ke Systems Manager menggunakan titik akhir VPC, dan preferensi Session Manager Anda menulis output sesi ke bucket Amazon S3. Atau, `logs` titik akhir dalam format **`com.amazonaws.`*`region`*`.logs`**diperlukan jika node terkelola Anda terhubung ke Systems Manager menggunakan titik akhir VPC, dan preferensi Session Manager Anda menulis output sesi

ke CloudWatch grup log Log. Untuk informasi selengkapnya, lihat Membuat VPC endpoint untuk Systems Manager.

- Solusi D: Grup log atau bucket Amazon S3 yang Anda tentukan dalam preferensi sesi Anda telah dihapus. Untuk mengatasi masalah ini, perbarui preferensi sesi Anda dengan grup log atau bucket S3 yang valid.
- Solusi E: Grup log atau bucket Amazon S3 yang Anda tentukan dalam preferensi sesi Anda tidak dienkripsi, tetapi Anda telah menetapkan `cloudWatchEncryptionEnabled` atau `s3EncryptionEnabled` input ke `true`. Untuk mengatasi masalah ini, perbarui preferensi sesi Anda dengan grup log atau bucket Amazon S3 yang dienkripsi, atau atur `cloudWatchEncryptionEnabled` atau `s3EncryptionEnabled` input ke `false`. Skenario ini hanya berlaku untuk pelanggan yang membuat preferensi sesi menggunakan alat baris perintah.

Node terkelola menjadi tidak responsif selama sesi berjalan lama

Masalah: Node terkelola Anda menjadi tidak responsif atau mogok selama sesi berjalan lama.

Solusi: Kurangi durasi retensi SSM Agent log untukSession Manager.

Untuk mengurangi durasi retensi SSM Agent log untuk sesi

1. Temukan `amazon-ssm-agent.json.template` di `/etc/amazon/ssm/` direktori untukLinux, atau `C:\Program Files\Amazon\SSM` untukWindows.
2. Salin isi `amazon-ssm-agent.json.template` ke file baru di direktori yang sama bernama `amazon-ssm-agent.json`.
3. Kurangi nilai default dari nilai `SessionLogsRetentionDurationHours` dalam properti `SSM`, dan simpan file.
4. Mulai ulangSSM Agent.

Terjadi kesalahan (InvalidDocument) saat memanggil StartSession operasi

Masalah: Anda menerima kesalahan berikut saat memulai sesi dengan menggunakan 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.
```

Solusi: Dokumen SSM yang Anda tentukan untuk `--document-name` parameter bukanlah dokumen Sesi. Gunakan prosedur berikut untuk melihat daftar dokumen Sesi di AWS Management Console.

Untuk melihat daftar dokumen Sesi

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.
3. Dalam daftar Kategori, pilih Dokumen sesi.

AWS Systems Manager Run Command

Dengan menggunakanRun Command, kemampuanAWS Systems Manager, Anda dapat mengelola konfigurasi node terkelola dari jarak jauh dan aman. Node terkelola adalah instans Amazon Elastic Compute Cloud (Amazon EC2) atau mesin non-EC2 di lingkungan hybrid dan multicloud Anda yang telah dikonfigurasi untuk Systems Manager. Run Commandmemungkinkan Anda untuk mengotomatiskan tugas administratif umum dan melakukan perubahan konfigurasi satu kali dalam skala besar. Anda dapat menggunakan Run Command dariAWS Management Console, the AWS Command Line Interface (AWS CLI)AWS Tools for Windows PowerShell, atau AWS SDK. Run Commandditawarkan tanpa biaya tambahan. Untuk memulaiRun Command, buka konsol Systems Manager. Di panel navigasi, pilih Run Command.

Administrator menggunakan Run Command untuk menginstal atau mem-bootstrap aplikasi, membangun pipeline penerapan, menangkap file log saat instance dihapus dari grup Auto Scaling, menggabungkan instance ke domain Windows, dan banyak lagi.

Memulai

Tabel berikut mencakup informasi untuk membantu Anda memulaiRun Command.

Topik	Detail
Menyiapkan AWS Systems Manager	Pastikan Anda telah menyelesaikan persyaratan penyiapan untuk instans Amazon Elastic Compute Cloud (Amazon EC2) dan mesin non-EC2 di lingkungan hybrid dan multicloud.

Topik	Detail
Menyiapkan Manajer Sistem untuk lingkungan hybrid dan multicloud	(Opsional) Daftarkan server lokal dan VM AWS sehingga Anda dapat mengelolanya menggunakan. Run Command
the section called “Menyiapkan perangkat tepi”	(Opsional) Konfigurasikan perangkat tepi sehingga Anda dapat mengelolanya menggunakanRun Command.
Menjalankan perintah pada node yang dikelola	Pelajari cara menjalankan perintah yang menargetkan satu atau lebih node terkelola dengan menggunakanAWS Management Console.
Run CommandPanduan	Pelajari cara menjalankan perintah menggunak an Alat untuk Windows PowerShell atauAWS CLI.

EventBridge dukungan

Kemampuan Systems Manager ini didukung sebagai jenis peristiwa dan tipe target dalam EventBridge aturan Amazon. Untuk informasi, lihat Pemantauan peristiwa Systems Manager dengan Amazon EventBridge dan Referensi: Pola dan jenis EventBridge acara Amazon untuk Systems Manager.

Info lebih lanjut

- Jarak jauh Run Command pada Instans EC2 (tutorial 10 menit)
- Kuota layanan Systems Manager di Referensi Umum Amazon Web Services
- AWS Systems ManagerReferensi API

Topik

- Run CommandPanduan
- Memecahkan masalah Run Command Systems Manager

Menyiapkan Run Command

Sebelum Anda dapat mengelola node dengan menggunakanRun Command, suatu kemampuanAWS Systems Manager, konfigurasikan kebijakanAWS Identity and Access Management (IAM) untuk setiap pengguna yang akan menjalankan perintah.

Anda juga harus mengkonfigurasi node Anda untuk Systems Manager. Untuk informasi selengkapnya, lihat Menyiapkan AWS Systems Manager.

Kami merekomendasikan untuk melengkapi tugas penyiapan opsional berikut untuk membantu meminimalkan postur keamanan danday-to-day pengelolaan Node terkelola Anda.

Memantau eksekusi perintah menggunakan AmazonEventBridge

Anda dapat menggunakanEventBridge untuk membuat log perubahan status eksekusi perintah. Anda dapat membuat aturan yang berjalan setiap kali terjadi status transisi, atau saat ada transisi ke satu atau beberapa status yang penting. Anda juga dapat menentukanRun Command sebagai tindakan target ketika suatuEventBridge peristiwa terjadi. Untuk informasi selengkapnya, lihat EventBridge Pengonfigurasan peristiwa Systems Manager.

Memantau eksekusi perintah menggunakan AmazonCloudWatch Logs

Anda dapat mengkonfigurasiRun Command untuk secara berkala mengirim semua output perintah dan log kesalahan ke grup AmazonCloudWatch. Anda dapat memantau log output ini secara hampir waktu nyata, mencari frasa, nilai, atau pola tertentu, dan membuat alarm berdasarkan pencarian. Untuk informasi selengkapnya, lihat MengonfigurasiCloudWatch Log Amazon untukRun Command.

MembatasiRun Command akses ke node terkelola tertentu

Anda dapat membatasi kemampuan pengguna untuk menjalankan perintah pada node yang dikelola dengan menggunakanAWS Identity and Access Management (IAM). Secara khusus, Anda dapat membuat kebijakan IAM dengan syarat bahwa pengguna hanya dapat menjalankan perintah pada node terkelola yang ditandai dengan tag tertentu. Untuk informasi selengkapnya, lihat MembatasiRun Command akses akses berdasarkan tag.

MembatasiRun Command akses akses berdasarkan tag

Bagian ini menjelaskan cara membatasi kemampuan pengguna untuk menjalankan perintah pada node yang dikelola dengan menentukan kondisi tag dalam kebijakan IAM. Node yang dikelola menyertakan instans Amazon EC2 dan node non-EC2 dalam lingkungan hybrid dan multicloud yang dikonfigurasi untuk Systems Manager. Meskipun informasi tersebut tidak disajikan secara eksplisit, Anda juga dapat membatasi akses ke perangkatAWS IoT Greengrass inti terkelola. Untuk memulai, Anda harus menandaiAWS IoT Greengrass perangkat Anda. Untuk informasi selengkapnya, lihat MenandaiAWS IoT Greengrass Version 2 sumber daya Anda di PanduanAWS IoT Greengrass Version 2 Developer.

Anda dapat membatasi eksekusi perintah ke node terkelola tertentu dengan membuat kebijakan IAM yang mencakup syarat bahwa pengguna hanya dapat menjalankan perintah pada node dengan tag tertentu. Pada contoh berikut, pengguna diperbolehkan untuk menggunakanRun Command (`Effect: Allow, Action: ssm:SendCommand`) dengan menggunakan dokumen SSM (`Resource: arn:aws:ssm:*:*:document/*`) pada setiap node (`Resource: arn:aws:ec2:*:*:instance/*`) dengan kondisi bahwa node adalah FinanceWebServer (`ssm:resourceTag/Finance: WebServer`). Jika pengguna mengirimkan perintah ke node yang tidak ditandai atau yang memiliki tag selain`Finance: WebServer`, hasil eksekusi menunjukkan`AccessDenied`.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "ssm:SendCommand"
         ],
         "Resource":[
            "arn:aws:ssm:*:*:document/*"
         ]
      },
      {
         "Effect":"Allow",
         "Action":[
            "ssm:SendCommand"
         ],
         "Resource":[
            "arn:aws:ec2:*:*:instance/*"
         ],
```

```
         "Condition":{
            "StringLike":{
               "ssm:resourceTag/Finance":[
                  "WebServers"
               ]
            }
         }
      }
   ]
}
```

Anda dapat membuat kebijakan IAM yang mengizinkan pengguna untuk menjalankan perintah pada node terkelola yang ditandai beberapa tanda. Kebijakan berikut ini mengizinkan pengguna untuk menjalankan perintah pada node terkelola yang memiliki dua tag. Jika pengguna mengirimkan perintah ke node yang tidak ditandai dengan kedua tag tersebut, hasil eksekusi menunjukkan`AccessDenied`.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "ssm:SendCommand"
         ],
         "Resource":"*",
         "Condition":{
            "StringLike":{
               "ssm:resourceTag/tag_key1":[
                  "tag_value1"
               ],
               "ssm:resourceTag/tag_key2":[
                  "tag_value2"
               ]
            }
         }
      },
      {
         "Effect":"Allow",
         "Action":[
            "ssm:SendCommand"
         ],
         "Resource":[
```

```
            "arn:aws:ssm:us-west-1::document/AWS-*",
            "arn:aws:ssm:us-east-2::document/AWS-*"
         ]
      },
      {
         "Effect":"Allow",
         "Action":[
            "ssm:UpdateInstanceInformation",
            "ssm:ListCommands",
            "ssm:ListCommandInvocations",
            "ssm:GetDocument"
         ],
         "Resource":"*"
      }
   ]
}
```

Anda juga dapat membuat kebijakan IAM yang mengizinkan pengguna untuk menjalankan perintah pada beberapa grup Node terkelola yang ditandai. Kebijakan contoh berikut ini mengizinkan pengguna untuk menjalankan perintah pada salah satu grup yang ditandai, atau kedua grup.

```
{
   "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":"*"
         }
      ]
}
```

Untuk informasi selengkapnya tentang membuat kebijakan IAM, lihat Kebijakan Terkelola dan kebijakan inline dalam Panduan Pengguna IAM. Untuk informasi selengkapnya tentang menandai node yang dikelola, lihat Editor Tag di PanduanAWS Resource Groups Pengguna.

Menjalankan perintah pada node yang dikelola

Bagian ini mencakup informasi tentang cara mengirim perintah dariAWS Systems Manager konsol ke node yang dikelola. Bagian ini juga mencakup informasi tentang cara membatalkan perintah.

Untuk informasi tentang cara mengirim perintah menggunakan Windows PowerShell, lihatWalkthrough: GunakanAWS Tools for Windows PowerShell denganRun Command atau contoh

di AWS Systems Managerbagian ReferensiAWS Tools for PowerShell Cmdlet. Untuk informasi tentang cara mengirim perintah menggunakan AWS Command Line Interface (AWS CLI), lihat Walkthrough: Gunakan dengan AWS CLIRun Command atau contoh di Referensi CLI SSM.

> **Important**
>
> Ketika Anda mengirim perintah menggunakan perintahRun Command, jangan menyertakan informasi sensitif yang diformat sebagai teks biasa, seperti kata sandi, data konfigurasi, atau rahasia lainnya. Semua aktivitas API Systems Manager di akun Anda dicatat dalam bucket S3 untukAWS CloudTrail log. Ini berarti bahwa setiap pengguna dengan akses ke bucket S3 dapat melihat nilai-nilai teks biasa dari rahasia tersebut. Untuk alasan ini, kami merekomendasikan untuk membuat dan menggunakan`SecureString` parameter untuk mengenkripsi data sensitif yang Anda gunakan dalam operasi Systems Manager. Untuk informasi selengkapnya, lihat Membatasi akses ke parameter Systems Manager menggunakan kebijakan IAM.

Konten

Menjalankan perintah dari konsol

Anda dapat menggunakanRun Command, kemampuanAWS Systems Manager, dari AWS Management Console untuk mengkonfigurasi node terkelola tanpa harus masuk ke dalamnya. Topik ini mencakup contoh yang menunjukkan cara memperbarui SSM Agent pada node terkelola dengan menggunakanRun Command.

Sebelum Anda memulai

Sebelum Anda mengirim perintah menggunakanRun Command, verifikasi bahwa node terkelola Anda memenuhi semua persyaratan penyiapan Systems Manager.

Untuk mengirim perintah menggunakan Run Command

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.

2. Di panel navigasi, pilih Run Command.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Run Command.

3. Pilih Run Command.
4. Di daftar Dokumen perintah, pilih dokumen Systems Manager.
5. Di bagian Parameter perintah, tentukan nilai untuk parameter yang diperlukan.
6. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

   > **Tip**
   >
   > Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

7. Untuk Parameter lainnya:

   - Untuk Komentar, ketik informasi tentang perintah ini.
   - Untuk Waktu habis (detik), tentukan jumlah detik untuk menunggu sistem sebelum gagal menjalankan perintah keseluruhan.

8. Untuk Pengendalian rate:

   - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

     > **Note**
     >
     > Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

- Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

9. (Opsional) Pilih CloudWatch alarm untuk diterapkan pada perintah Anda untuk pemantauan. Untuk melampirkan CloudWatch alarm ke perintah Anda, prinsipal IAM yang menjalankan perintah harus memiliki izin untuk `iam:createServiceLinkedRole` tindakan tersebut. Untuk informasi selengkapnya tentang CloudWatch alarm, lihat Menggunakan CloudWatch alarm Amazon. Perhatikan bahwa jika alarm Anda aktif, pemanggilan perintah yang tertunda tidak berjalan.
10. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan milik pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbedaAkun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

11. Di bagian Notifikasi SNS, jika Anda ingin notifikasi dikirim tentang status eksekusi perintah, pilih kotak centang Aktifkan notifikasi SNS.

    Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Run Command Amazon SNS, lihat. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS
12. Pilih Jalankan.

Untuk informasi tentang membatalkan perintah, lihat the section called “Membatalkan perintah”.

Menjalankan kembali perintah

Systems Manager mencakup dua pilihan untuk membantu Anda menjalankan kembali perintah dari halaman Run Command dalam konsol Systems Manager.

- Jalankan kembali: Tombol ini memungkinkan Anda untuk menjalankan perintah yang sama tanpa membuat perubahan.
- Salin ke baru: Tombol ini menyalin pengaturan dari satu perintah ke perintah baru dan memberikan Anda pilihan untuk mengedit pengaturan tersebut sebelum Anda menjalankannya.

Untuk menjalankan kembali perintah

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Run Command.
3. Pilih sebuah perintah untuk dijalankan kembali. Anda dapat menjalankan kembali perintah segera setelah menjalankannya dari halaman detail perintah. Atau, Anda dapat memilih perintah yang sebelumnya Anda jalankan dari tab Riwayat perintah.
4. Pilih Jalankan kembali untuk menjalankan perintah yang sama tanpa perubahan, atau pilih Salin ke baru untuk mengedit pengaturan perintah sebelum Anda menjalankannya.

Menjalankan perintah menggunakan versi dokumen tertentu

Anda dapat menggunakan parameter versi dokumen untuk menentukan versi dokumen AWS Systems Manager apa yang digunakan ketika perintah berjalan. Anda dapat menentukan salah satu opsi berikut untuk parameter ini:

- $DEFAULT
- $LATEST
- Nomor versi

Jalankan prosedur berikut ini untuk menjalankan perintah menggunakan parameter versi dokumen.

Linux

Untuk menjalankan perintah menggunakan AWS CLI pada mesin Linux lokal

1. Instal dan konfigurasikan dan konfigurasikan dan konfigurasikan dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaru dariAWS CLI.
2. Cantumkan semua dokumen yang tersedia

   Perintah ini mencantumkan semua dokumen yang tersedia untuk akun Anda berdasarkan izin AWS Identity and Access Management(IAM).

   ```
   aws ssm list-documents
   ```
3. Jalankan perintah berikut ini untuk menampilkan versi yang berbeda dari dokumen. Ganti *nama dokumen* dengan informasi Anda sendiri.

   ```
   aws ssm list-document-versions \
       --name "document name"
   ```
4. Jalankan perintah berikut ini untuk menjalankan perintah yang menggunakan sebuah versi dokumen SSM. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

   ```
   aws ssm send-command \
       --document-name "AWS-RunShellScript" \
       --parameters commands="echo Hello" \
       --instance-ids instance-ID \
       --document-version '$LATEST'
   ```

Windows

Untuk menjalankan perintah menggunakan AWS CLI pada mesin Windows lokal

1. Instal dan konfigurasikan dan konfigurasikan dan konfigurasikan dan konfigurasikanAWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk informasi, lihat Menginstal atau memperbarui versi terbaru dariAWS CLI.
2. Cantumkan semua dokumen yang tersedia

Perintah ini mencantumkan semua dokumen yang tersedia untuk akun Anda berdasarkan izin AWS Identity and Access Management(IAM).

```
aws ssm list-documents
```

3. Jalankan perintah berikut ini untuk menampilkan versi yang berbeda dari dokumen. Ganti *nama dokumen* dengan informasi Anda sendiri.

```
aws ssm list-document-versions ^
    --name "document name"
```

4. Jalankan perintah berikut ini untuk menjalankan perintah yang menggunakan sebuah versi dokumen SSM. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

```
aws ssm send-command ^
    --document-name "AWS-RunShellScript" ^
    --parameters commands="echo Hello" ^
    --instance-ids instance-ID ^
    --document-version "$LATEST"
```

PowerShell

Untuk menjalankan perintah menggunakan Tools for PowerShell

1. Instal dan konfigurasikan dan konfigurasikan dan konfigurasikan dan konfigurasikan dan konfigurasikan dan konfigurasiAWS Tools for PowerShell (Tools for Windows PowerShell), jika Anda belum melakukannya.

   Untuk informasi, lihat MenginstalAWS Tools for PowerShell.

2. Cantumkan semua dokumen yang tersedia

   Perintah ini mencantumkan semua dokumen yang tersedia untuk akun Anda berdasarkan izin AWS Identity and Access Management(IAM).

```
Get-SSMDocumentList
```

3. Jalankan perintah berikut ini untuk menampilkan versi yang berbeda dari dokumen. Ganti *nama dokumen* dengan informasi Anda sendiri.

```
Get-SSMDocumentVersionList `
    -Name "document name"
```

4. Jalankan perintah berikut ini untuk menjalankan perintah yang menggunakan sebuah versi dokumen SSM. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

```
Send-SSMCommand `
    -DocumentName "AWS-RunShellScript" `
    -Parameter @{commands = "echo helloWorld"} `
    -InstanceIds "instance-ID" `
    -DocumentVersion $LATEST
```

Menjalankan perintah saat skala

Anda dapat menggunakanRun Command, kemampuanAWS Systems Manager, untuk menjalankan perintah pada armada node dikelola dengan menggunakan`targets`. `targets`Parameter menerima`Key,Value` kombinasi berdasarkan tag yang Anda tentukan untuk node yang dikelola. Ketika Anda menjalankan perintah, sistem akan mencari dan mencoba untuk menjalankan perintah pada semua node terkelola yang sesuai dengan tag yang ditentukan. Untuk informasi selengkapnya tentang memberi tag pada instans terkelola, lihat MenandaiAWS sumber daya Anda di Panduan Pengguna TaggingAWS Resources. Untuk informasi tentang menandai perangkat IoT terkelola Anda, lihat MenandaiAWS IoT Greengrass Version 2 sumber daya Anda di PanduanAWS IoT Greengrass Version 2 Developer.

Anda juga dapat menggunakan`targets` parameter untuk menargetkan daftar ID node terkelola tertentu, seperti yang dijelaskan di bagian berikutnya.

Untuk mengendalikan cara perintah berjalan di ratusan atau ribuan node terkelola,Run Command juga menyertakan parameter untuk membatasi berapa banyak node yang secara bersamaan dapat memproses permintaan dan berapa banyak kesalahan yang dapat dilemparkan oleh perintah sebelum perintah dibatalkan.

Daftar Isi

Menargetkan beberapa node terkelola

Anda dapat menjalankan perintah dan menargetkan node yang dikelola dengan menentukan tag, namaAWS resource group, atau ID node terkelola.

Contoh berikut menunjukkan format perintah saat menggunakanRun Command from theAWS Command Line Interface (AWS CLI). Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri. Contoh perintah di bagian ini disingkat menggunakan `[...]`.

Contoh 1: Menargetkan tag

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 ^
    [...]
```

Contoh 2: Menargetkan sebuahAWS resource group berdasarkan nama

Anda dapat menentukan maksimal satu nama resource group per perintah. Saat Anda membuat resource group, kami merekomendasikan untuk menyertakan `AWS::SSM:ManagedInstance` dan `AWS::EC2::Instance` sebagai jenis sumber daya dalam kriteria pengelompokan Anda.

> **Note**
>
> Untuk mengirim perintah yang menargetkan sebuah resource group, Anda harus telah diberikan izinAWS Identity and Access Management (IAM) untuk mencantumkan atau melihat sumber daya milik grup tersebut. Untuk informasi selengkapnya, lihat Menyiapkan izin di PanduanAWS Resource Groups Pengguna.

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 ^
    [...]
```

Contoh 3: Menargetkan sebuahAWS resource group berdasarkan jenis sumber daya

Anda dapat menentukan maksimal lima jenis resource group per perintah. Saat Anda membuat resource group, kami merekomendasikan untuk menyertakan `AWS::SSM:ManagedInstance` dan `AWS::EC2::Instance` sebagai jenis sumber daya dalam kriteria pengelompokan Anda.

> **Note**
>
> Untuk mengirim perintah yang menargetkan sebuah resource group, Anda harus telah diberikan izin IAM untuk mencantumkan, atau melihat, sumber daya milik grup tersebut. Untuk informasi selengkapnya, lihat Menyiapkan izin di PanduanAWS Resource Groups Pengguna.

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 ^
    [...]
```

Contoh 4: Menargetkan ID instans

Contoh berikut menunjukkan bagaimana menargetkan node yang dikelola dengan menggunakan`instanceids` kunci dengan`targets` parameter. Anda dapat menggunakan kunci ini untuk menargetkan perangkatAWS IoT Greengrass inti terkelola karena setiap perangkat diberi *ID_Number* mi. Anda dapat melihat ID perangkat diFleet Manager, kemampuanAWS 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 ^
    [...]
```

Jika Anda menandai node terkelola untuk lingkungan yang berbeda menggunakan`Key` nama`Environment` dan`Values` dari`Development`,`Test`,`Production`,`Pre-production` dan, maka Anda dapat mengirim perintah ke semua node yang dikelola di salah satu lingkungan ini dengan menggunakan`targets` parameter dengan sintaks berikut.

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 ^
    [...]
```

Anda dapat menargetkan node terkelola tambahan di lingkungan lain dengan menambahkan ke`Values` daftar. Pisahkan item menggunakan koma.

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 ^
    [...]
```

Variasi: Menyempurnakan target Anda menggunakan beberapa kriteria `Key`

Anda dapat menyempurnakan jumlah target untuk perintah Anda dengan menyertakan beberapa kriteria `Key`. Jika Anda menyertakan lebih dari satu`Key` kriteria, sistem menargetkan node terkelola yang memenuhi semua kriteria tersebut. Perintah berikut ini menargetkan semua node terkelola yang ditandai untuk Departemen Keuangan dan ditandai untuk peran server basis data.

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 ^
    [...]
```

Variasi: Menggunakan beberapa `Key` dan kriteria `Value`

Berdasarkan contoh sebelumnya, Anda dapat menargetkan beberapa departemen dan beberapa peran server dengan memasukkan item tambahan di kriteria `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 ^
    [...]
```

Variasi: Menargetkan node terkelola yang ditandai menggunakan beberapa`Values` kriteria

Jika Anda menandai node terkelola untuk lingkungan yang berbeda menggunakan`Key` nama`DepartmentSales` dan`Values` dari dan`Finance`, maka Anda dapat mengirim perintah ke semua node di lingkungan ini dengan menggunakan`targets` parameter dengan sintaks berikut.

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 ^
    [...]
```

Anda dapat menentukan maksimum lima kunci, dan lima nilai untuk setiap kunci.

Jika salah satu kunci tag (nama tag) atau nilai tag termasuk spasi, lampirkan kunci tag atau nilai dalam tanda kutip, seperti yang ditunjukkan dalam contoh berikut.

Contoh: Spasi di tag `Value`

Linux & macOS

```
aws ssm send-command \
    --document-name document-name \
    --targets Key=tag:OS,Values="Windows Server 2016 Nano" \
    [...]
```

Windows

```
aws ssm send-command ^
    --document-name document-name ^
    --targets Key=tag:OS,Values="Windows Server 2016 Nano" ^
    [...]
```

Contoh: Spasi di kunci `tag` dan `Value`

Linux & macOS

```
aws ssm send-command \
    --document-name document-name \
    --targets Key="tag:Operating System",Values="Windows Server 2016 Nano" \
    [...]
```

Windows

```
aws ssm send-command ^
    --document-name document-name ^
    --targets Key="tag:Operating System",Values="Windows Server 2016 Nano" ^
    [...]
```

Contoh: Spasi dalam satu item dalam daftar `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" ^
    [...]
```

Menggunakan pengendali rate

Anda dapat mengendalikan rate perintah dikirimkan ke node terkelola dalam sebuah grup dengan menggunakan pengendali konkurensi dan pengendali kesalahan.

Topik

- Menggunakan pengendali konkurensi
- Menggunakan pengendali kesalahan

Menggunakan pengendali konkurensi

Anda dapat mengendalikan jumlah node terkelola yang menjalankan perintah secara bersamaan menggunakan`max-concurrency` parameter (opsi konkurensi di halaman Jalankan perintah). Anda dapat menentukan jumlah absolut node terkelola, misalnya**10**, atau persentase target yang diatur, misalnya**10%**. Sistem antrean mengirimkan perintah untuk satu node dan menunggu sampai

sistem mengakui pemanggilan awal tersebut sebelum mengirim perintah untuk dua node lagi. Sistem secara eksponensial mengirimkan perintah ke lebih banyak node sampai sistem memenuhi nilai`max-concurrency`. default untuk nilai `max-concurrency` adalah 50. Contoh berikut ini menunjukkan kepada Anda cara menentukan nilai untuk parameter `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 ^
    [...]
```

Menggunakan pengendali kesalahan

Anda juga dapat mengendalikan eksekusi perintah untuk ratusan atau ribuan node terkelola dengan mengatur batas kesalahan menggunakan`max-errors` parameter (bidang Ambang Kesalahan

di halaman Jalankan perintah). parameter menentukan berapa banyak kesalahan yang diizinkan sebelum sistem berhenti mengirim perintah ke node terkelola tambahan. Anda dapat menentukan jumlah kesalahan mutlak, misalnya **10**, atau persentase target yang ditentukan, misalnya **10%**. Jika Anda menentukan **3**, misalnya, maka sistem akan berhenti mengirim perintah ketika kesalahan keempat diterima. Jika Anda menentukan**0**, maka sistem berhenti mengirim perintah ke node terkelola tambahan setelah hasil kesalahan pertama dikembalikan. Jika Anda mengirim perintah ke 50 node terkelola dan mengatur`max-errors` ke**10%**, maka sistem berhenti mengirim perintah ke node tambahan ketika kesalahan keenam diterima.

Pemanggilan yang sudah menjalankan perintah saat `max-errors` tercapai diizinkan untuk diselesaikan, tetapi beberapa pemanggilan tersebut mungkin gagal juga. Jika Anda perlu memastikan bahwa tidak akan ada lebih dari `max-errors` pemanggilan yang gagal, atur `max-concurrency` ke **1** sehingga pemanggilan akan dilanjutkan satu per satu. default untuk maksimal kesalahan adalah 0. Contoh berikut ini menunjukkan kepada Anda cara menentukan nilai untuk parameter `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 ^
    [...]
```

Membatalkan perintah

Anda dapat mencoba untuk membatalkan perintah selama layanan menunjukkan bahwa itu dalam status Tertunda atau Mengeksekusi. Namun, bahkan jika perintah masih berada dalam salah satu status tersebut, kami tidak dapat menjamin bahwa perintah akan dibatalkan dan proses yang mendasarinya berhenti.

Untuk menjalankan perintah menggunakan konsol

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Run Command.
3. Pilih pemanggilan perintah yang ingin Anda batalkan.
4. Pilih Batalkan perintah.

Untuk membatalkan perintah menggunakan AWS CLI

Jalankan perintah berikut. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

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"
```

Untuk informasi tentang status perintah yang dibatalkan, lihat Memahami status perintah.

Menggunakan kode keluar dalam perintah

Dalam beberapa kasus, Anda mungkin perlu mengelola cara perintah Anda ditangani dengan menggunakan kode keluar.

Tentukan kode keluar dalam perintah

MenggunakanRun Command, kemampuanAWS Systems Manager, Anda dapat menentukan kode keluar untuk menentukan bagaimana perintah ditangani. Secara default, kode keluar dari perintah terakhir yang dijalankan dalam skrip dilaporkan sebagai kode keluar untuk seluruh skrip. Misalnya, Anda memiliki skrip yang berisi tiga perintah. Yang pertama gagal tapi yang berikutnya berhasil. Karena perintah terakhir berhasil, status eksekusi dilaporkan sebagai `succeeded`.

skrip shell

Untuk menggagalkan seluruh skrip pada kegagalan perintah pertama, Anda dapat menyertakan pernyataan bersyarat shell untuk keluar dari skrip jika perintah sebelum yang terakhir gagal. Gunakan pendekatan berikut.

```
<command 1>
    if [ $? != 0 ]
    then
        exit <N>
```

```
    fi
    <command 2>
    <command 3>
```

Pada contoh berikut, seluruh skrip gagal jika perintah pertama gagal.

```
cd /test
    if [ $? != 0 ]
    then
        echo "Failed"
        exit 1
    fi
    date
```

PowerShell skrip

PowerShell mengharuskan Anda memanggil`exit` secara eksplisit dalam skrip AndaRun Command agar berhasil menangkap kode keluar.

```
<command 1>
    if ($?) {<do something>}
    else {exit <N>}
    <command 2>
    <command 3>
    exit <N>
```

Berikut ini contohnya:

```
cd C:\
    if ($?) {echo "Success"}
    else {exit 1}
    date
```

Menangani reboot saat menjalankan perintah

Jika Anda menggunakanRun Command, kemampuan AWS Systems Manager, untuk menjalankan skrip yang me-reboot node terkelola, kami sarankan Anda menentukan kode keluar dalam skrip Anda. Jika Anda mencoba me-reboot node dari skrip dengan menggunakan beberapa mekanisme lain, status eksekusi skrip mungkin tidak diperbarui dengan benar, bahkan jika reboot adalah langkah terakhir dalam skrip Anda. Untuk node terkelola Windows, Anda tentukan `exit 3010`

dalam skrip Anda. Untuk Linux dan node macOS terkelola, Anda tentukan`exit 194`. Kode keluar menginstruksikan AWS Systems Manager Agent (SSM Agent) untuk me-reboot node yang dikelola, dan kemudian memulai ulang skrip setelah reboot selesai. Sebelum memulai reboot, SSM Agent menginformasikan layanan Systems Manager di cloud bahwa komunikasi akan terganggu selama reboot server.

> **Note**
>
> Skrip reboot tidak dapat menjadi bagian dari `aws:runDocument` plugin. Jika dokumen berisi skrip reboot dan dokumen lain mencoba menjalankan dokumen itu melalui `aws:runDocument` plugin, SSM Agent mengembalikan kesalahan.

Buat skrip idempoten

Saat mengembangkan skrip yang me-reboot node terkelola, buat skrip idempoten sehingga eksekusi skrip berlanjut di tempat yang ditinggalkannya setelah reboot. Skrip idempoten mengelola status dan memvalidasi apakah tindakan dilakukan atau tidak. Hal ini mencegah langkah untuk dijalankan beberapa kali ketika hanya dimaksudkan untuk dijalankan satu kali.

Berikut adalah contoh garis besar skrip idempoten yang me-reboot node terkelola beberapa kali.

```
$name = Get current computer name
If ($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
    }
```

Contoh

Contoh skrip berikut menggunakan kode keluar untuk memulai ulang node terkelola. Contoh Linux menginstal pembaruan paket di Amazon Linux, dan kemudian memulai ulang node. Windows ServerContoh menginstal Telnet-Client pada node, dan kemudian restart.

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
    }
```

Memahami status perintah

Run CommandSebuah kemampuan dariAWS Systems Manager, melaporkan informasi status terperinci tentang berbagai status yang dialami perintah selama pemrosesan dan untuk setiap node terkelola yang memproses perintah. Anda dapat memantau status perintah menggunakan metode berikut:

- PilihSegarkanikon padaPerintahtab diRun Commandantarmuka konsol.
- Panggilandaftar-perintahataulist-command-invocationsmenggunakanAWS Command Line Interface(AWS CLI). Atau hubungiDapatkan-ssmCommandatauDapatkan-SSMCommandInvocationmemakaiAWS Tools for Windows PowerShell.

- Konfigurasikan Amazon EventBridge untuk menanggapi perubahan status atau status.
- Konfigurasikan Amazon Simple Notification Service (Amazon SNS) untuk mengirim notifikasi untuk semua perubahan status atau status tertentu seperti`Failed`atau`TimedOut`.

Run Commandstatus

Run Commandmelaporkan rincian status untuk tiga area: plugin, pemanggilan, dan status perintah keseluruhan. plugin adalah blok eksekusi kode yang ditetapkan dalam dokumen SSM perintah Anda. Untuk informasi selengkapnya tentang plugin, lihat Referensi plugin dokumen perintah.

Ketika Anda mengirim perintah ke beberapa node terkelola pada saat yang sama, setiap salinan perintah yang menargetkan setiap node adalahpemanggilan perintah. Misalnya, jika Anda menggunakan dokumen `AWS-RunShellScript` dan mengirimkan perintah `ifconfig` ke 20 instans Linux, perintah tersebut memiliki 20 pemanggilan. Setiap pemanggilan perintah secara individual melaporkan status. Plugin untuk pemanggilan perintah tertentu juga secara individual melaporkan status.

Terakhir,Run Commandtermasuk status perintah agregat untuk semua plugin dan pemanggilan. Status perintah agregat dapat berbeda dari status yang dilaporkan oleh plugin atau pemanggilan, seperti yang tercantum dalam tabel berikut.

> **Note**
>
> Jika Anda menjalankan perintah ke sejumlah besar node terkelola menggunakan`max-concurrency`atau`max-errors`parameter, status perintah mencerminkan batas yang dikenakan oleh parameter tersebut, seperti yang dijelaskan dalam tabel berikut. Untuk informasi selengkapnya tentang parameter ini, lihat Menjalankan perintah saat skala.

Status detail untuk plugin dan pemanggilan perintah

Status	Detail
Tertunda	Perintah belum dikirim ke node terkelola atau belum diterima olehSSM Agent. Jika perintah tidak diterima oleh agen sebelum panjang waktu berlalu yang sama dengan jumlah parameter Waktu habis (detik) dan parameter

Status	Detail
	Waktu eksekusi, status berubah ke `Delivery Timed Out`.
InProgress	Systems Manager mencoba mengirim perintah ke node yang dikelola, atau perintah diterima olehSSM Agentdan sudah mulai berjalan pada instance. Tergantung pada hasil dari semua plugin perintah, status berubah ke `Success`, `Failed`, `Delivery Timed Out`, atau `Execution Timed Out`. Pengecualian: Jika agen tidak berjalan atau tersedia di node, status perintah tetap di`In Progress`sampai agen tersedia lagi, atau sampai batas waktu eksekusi tercapai. Status kemudian berubah ke status terakhir.
Terlambat	Sistem mencoba mengirim perintah ke node yang dikelola tetapi tidak berhasil. Sistem mencoba lagi.
Berhasil	Perintah tersebut diterima olehSSM Agentpada node yang dikelola dan mengembalikan kode keluar nol. Status initidakberarti perintah diproses pada node. Ini adalah status terakhir. **Note** Untuk memecahkan masalah kesalahan atau mendapatkan informasi selengkapnya tentang eksekusi perintah, kirim perintah yang menangani kesalahan atau pengecual ian dengan mengembalikan kode keluar yang sesuai (kode keluar bukan nol untuk kegagalan perintah).

Status	Detail
DeliveryTimedOut	Perintah tidak dikirim ke node terkelola sebelum batas waktu total berakhir. Total timeout tidak dihitung terhadap perintah indukmax-errors limit, tetapi mereka berkontribusi pada apakah status perintah induk adalahSuccess,Incomplete , atauDelivery Timed Out. Ini adalah status terakhir.
ExecutionTimedOut	Otomatisasi perintah dimulai pada node terkelola, tetapi perintah tidak selesai sebelum batas waktu eksekusi berakhir. Batas waktu eksekusi dihitung sebagai kegagalan, yang akan mengirim balasan bukan nol dan Manajer Sistem akan keluar dari upaya untuk menjalank an otomatisasi perintah, dan melaporkan status kegagalan.
Gagal	Perintah tidak berhasil pada node terkelola . Untuk plugin, ini menunjukkan bahwa kode hasil tidak nol. Untuk pemanggilan perintah, ini menunjukkan bahwa kode hasil untuk satu atau lebih plugin tidak nol. Kegagalan pemanggil an dihitung terhadap batas max-errors perintah induk. Ini adalah status terakhir.
Dibatalkan	Perintah dibatalkan sebelum selesai. Ini adalah keadaan akhir.

Status	Detail
Tidak terkirim	Perintah tidak dapat dikirim ke node terkelola . Node mungkin tidak ada atau mungkin tidak merespons. Pemanggilan tidak terkirim tidak dihitung terhadap batas `max-errors` perintah induk, tetapi berkontribusi pada apakah status perintah induk adalah `Success` atau `Incomplete` . Sebagai contoh, jika semua pemanggilan dalam perintah memiliki status `Undeliverable` , maka status perintah yang dikembalikan adalah `Failed`. Namun, jika perintah memiliki lima pemanggilan, empat di antaranya mengembalikan status `Undeliverable` dan salah satunya mengembalikan status `Success`, maka status perintah induk adalah `Success`. Ini adalah keadaan akhir.
Diakhiri	Perintah induk melebihi batas `max-errors` dan pemanggilan perintah berikutnya dibatalkan oleh sistem. Ini adalah status terakhir.

Status	Detail
InvalidPlatform	Perintah dikirim ke node terkelola yang tidak cocok dengan platform yang diperlukan yang ditentukan oleh dokumen yang dipilih.`Invalid Platform`tidak dihitung terhadap batas maks-error perintah induk, tetapi itu berkontribusi pada apakah status perintah induk adalah Sukses atau Gagal. Sebagai contoh, jika semua pemanggilan dalam perintah memiliki status `Invalid Platform`, maka status perintah yang dikembalikan adalah `Failed`. Namun, jika perintah memiliki lima pemanggil an, empat di antaranya mengembalikan status `Invalid Platform` dan salah satunya mengembalikan status `Success`, maka status perintah induk adalah `Success`. Ini adalah keadaan akhir.
AccessDenied	TheAWS Identity and Access Managemen t(IAM) pengguna atau peran yang memulai perintah tidak memiliki akses ke node terkelola yang ditargetkan.`Access Denied`tidak dihitung terhadap perintah induk`max-errors` batas, tetapi itu berkontri busi pada apakah status perintah induk adalah`Success`atau`Failed`. Sebagai contoh, jika semua pemanggilan dalam perintah memiliki status `Access Denied`, maka status perintah yang dikembalikan adalah `Failed`. Namun, jika perintah memiliki lima pemanggil an, empat di antaranya mengembalikan status `Access Denied` dan salah satunya mengembalikan status `Success`, maka status perintah induk adalah `Success`. Ini adalah keadaan akhir.

Status detail untuk perintah

Status	Detail
Tertunda	Perintah belum diterima oleh agen pada node yang dikelola.
InProgress	Perintah telah dikirim ke setidaknya satu node terkelola tetapi belum mencapai status akhir pada semua node.
Terlambat	Sistem mencoba mengirim perintah ke node tetapi tidak berhasil. Sistem mencoba lagi.
Berhasil	Perintah tersebut diterima olehSSM Agentpada semua node terkelola yang ditentukan atau ditargetkan dan mengembalikan kode keluar nol. Semua pemanggilan perintah telah mencapai keadaan akhir, dan nilai `max-errors` tidak tercapai. Status initidakberarti perintah berhasil diproses pada semua node terkelola yang ditentukan atau ditargetkan. Ini adalah status terakhir. **Note** Untuk memecahkan masalah kesalahan atau mendapatkan informasi selengkapnya tentang eksekusi perintah, kirim perintah yang menangani kesalahan atau pengecualian dengan mengembalikan kode keluar yang sesuai (kode keluar bukan nol untuk kegagalan perintah).
DeliveryTimedOut	Perintah tidak dikirim ke node terkelola sebelum batas waktu total berakhir. Nilai `max-errors` atau lebih pemanggilan perintah

Status	Detail
	menunjukkan status `Delivery Timed Out`. Ini adalah keadaan akhir.
Gagal	Perintah tidak berhasil pada node terkelola . Nilai `max-errors` atau lebih pemanggil an perintah menunjukkan status `Failed`. Ini adalah keadaan akhir.
Tidak lengkap	Perintah dicoba pada semua node terkelola dan satu atau lebih pemanggilan tidak memiliki nilai`Success`. Namun, tidak terjadi cukup pemanggilan yang gagal untuk status menjadi `Failed`. Ini adalah status terakhir.
Dibatalkan	Perintah dibatalkan sebelum selesai. Ini adalah status terakhir.
RateExceeded	Jumlah node terkelola yang ditargetkan oleh perintah melebihi kuota akun untuk pemanggil an yang tertunda. Sistem telah membatalkan perintah sebelum menjalankannya pada node apa pun. Ini adalah status terakhir.

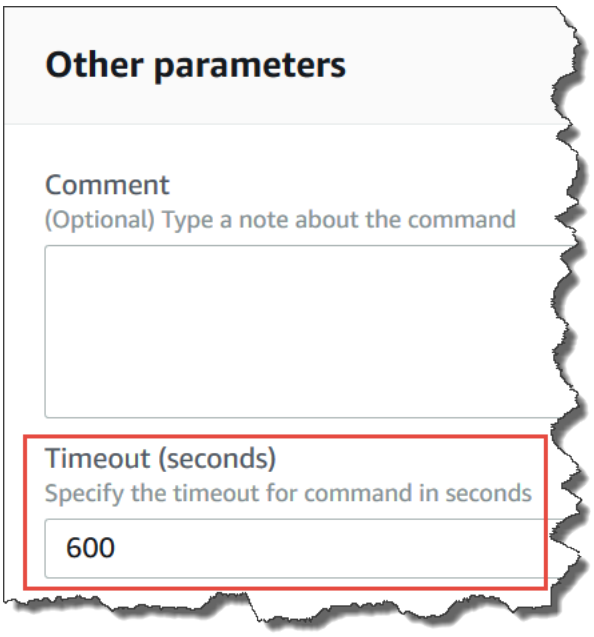
Status	Detail
AccessDenied	Pengguna atau peran yang memulai perintah tidak memiliki akses ke grup sumber daya yang ditargetkan.`AccessDenied` tidak dihitung terhadap perintah induk`max-errors` limit, tetapi berkontribusi pada apakah status perintah induk adalah`Success`atau`Failed`. (Sebagai contoh, jika semua pemanggilan dalam perintah memiliki status `AccessDenied` , maka status perintah yang dikembalikan adalah `Failed`. Namun, jika perintah memiliki 5 pemanggilan, 4 di antaranya mengembalikan status `AccessDenied` dan 1 mengembalikan status `Success`, maka status perintah induk adalah `Success`.) Ini adalah keadaan akhir.
Tidak Ada Instans Dalam Tag	Nilai pasangan kunci tag atau grup sumber daya yang ditargetkan oleh perintah tidak cocok dengan node terkelola apa pun. Ini adalah status terakhir.

Memahami nilai batas waktu perintah

Systems Manager memberlakukan nilai batas waktu berikut saat menjalankan perintah.

Total Batas Waktu

Di konsol Manajer Sistem, Anda menentukan nilai batas waktu diBatas waktu (detik)lapangan. Setelah perintah dikirim,Run Commandmemeriksa apakah perintah telah kedaluwarsa atau tidak. Jika sebuah perintah mencapai batas kedaluwarsa perintah (total batas waktu), statusnya berubah ke `DeliveryTimedOut` untuk semua pemanggilan yang memiliki status `InProgress`, `Pending` atau `Delayed`.



Pada tingkat yang lebih teknis, total batas waktu (Batas waktu (detik)) adalah kombinasi dari dua nilai batas waktu, seperti yang ditunjukkan di sini:

```
Total timeout = "Timeout(seconds)" from the console + "timeoutSeconds":
"{{ executionTimeout }}" from your SSM document
```

Misalnya, nilai default Waktu habis (detik) di konsol Systems Manager adalah 600 detik. Jika Anda menjalankan perintah dengan menggunakan dokumen SSM `AWS-RunShellScript`, nilai default "timeoutSeconds": "{{ executionTimeout }}" adalah 3600 detik, seperti yang ditunjukkan dalam contoh dokumen berikut:

```
"executionTimeout": {
    "type": "String",
    "default": "3600",

"runtimeConfig": {
  "aws:runShellScript": {
    "properties": [
      {
        "timeoutSeconds": "{{ executionTimeout }}"
```

Ini berarti perintah berjalan selama 4.200 detik (70 menit) sebelum sistem menetapkan status perintah`DeliveryTimedOut`.

Batas Waktu Eksekusi

Di konsol Systems Manager, Anda menentukan nilai batas waktu eksekusi di bidang Batas Waktu Eksekusi, jika tersedia. Tidak semua dokumen SSM mengharuskan Anda menentukan batas waktu eksekusi. TheBatas Waktu Eksekusibidang hanya ditampilkan ketika parameter input yang sesuai telah ditentukan dalam dokumen SSM. Jika ditentukan, perintah harus selesai dalam periode waktu ini.

> **Note**
>
> Run Commandbergantung padaSSM Agentmendokumentasikan respon terminal untuk menentukan apakah perintah dikirim ke agen atau tidak.SSM Agentharus mengirim`ExecutionTimedOut`sinyal untuk pemanggilan atau perintah yang akan ditandai sebagai`ExecutionTimedOut`.

Batas Waktu Eksekusi Default

Jika dokumen SSM tidak mengharuskan Anda secara eksplisit menentukan nilai batas waktu eksekusi, maka Systems Manager memberlakukan batas waktu eksekusi default hard-coded.

Cara Systems Manager melaporkan waktu habis

Jika Manajer Sistem menerima`execution timeout`balasan dariSSM Agentpada target, kemudian Manajer Sistem menandai pemanggilan perintah sebagai`executionTimeout`.

JikaRun Commandtidak menerima respons terminal dokumen dariSSM Agent, pemanggilan perintah ditandai sebagai`deliveryTimeout`.

Untuk menentukan status batas waktu pada target,SSM Agentmenggabungkan semua parameter dan isi dokumen SSM untuk dihitung`executionTimeout`. KapanSSM Agentmenentukan bahwa perintah telah habis waktunya, ia mengirimkan`executionTimeout`ke layanan.

default untuk Waktu habis (detik) adalah 3600 detik. default untuk Batas Waktu Eksekusi juga 3600 detik. Oleh karena itu, total batas waktu default untuk perintah adalah 7200 detik.

> **Note**
>
> SSM Agentproses`executionTimeout`berbeda tergantung pada jenis dokumen SSM dan versi dokumen.

Run CommandPanduan

Panduan di bagian ini menunjukkan cara menjalankan perintah denganRun Command, suatu kemampuanAWS Systems Manager, menggunakanAWS Command Line Interface (AWS CLI) atauAWS Tools for Windows PowerShell.

Daftar Isi

Anda juga dapat melihat contoh perintah di referensi berikut ini.

- AWS CLIReferensi Systems Manager
- AWS Tools for Windows PowerShell - AWS Systems Manager

Memperbarui perangkat lunak menggunakan Run Command

Prosedur berikut menjelaskan cara memperbarui perangkat lunak pada node terkelola Anda.

Memperbarui SSM Agent penggunaan Run Command

Prosedur berikut menjelaskan cara memperbarui yang SSM Agent berjalan pada node terkelola Anda. Anda dapat memperbarui ke versi terbaru SSM Agent atau menurunkan versi ke versi yang lebih lama. Ketika Anda menjalankan perintah, sistem mengunduh versi dari AWS, menginstalnya, dan kemudian menghapus instalan versi yang ada sebelum perintah dijalankan. Jika terjadi kesalahan selama proses ini, sistem akan kembali ke versi di server sebelum perintah dijalankan dan status perintah menunjukkan bahwa perintah gagal.

> **Note**
>
> Jika sebuah instance menjalankan macOS versi 11.0 (Big Sur) atau yang lebih baru, instance harus memiliki SSM Agent versi 3.1.941.0 atau lebih tinggi untuk menjalankan dokumen. AWS-UpdateSSMAgent Jika instance menjalankan versi SSM Agent rilis sebelum 3.1.941.0, Anda dapat memperbarui SSM Agent untuk menjalankan AWS-UpdateSSMAgent dokumen dengan menjalankan dan perintah. `brew update` `brew upgrade amazon-ssm-agent`

Untuk diberi tahu tentang SSM Agent pembaruan, berlangganan halaman Catatan SSM Agent Rilis diGitHub.

Untuk memperbarui SSM Agent menggunakan Run Command

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Run Command.
3. Pilih Run Command.
4. Di daftar Dokumen perintah, pilih **`AWS-UpdateSSMAgent`**.
5. Di bagian Parameter perintah, tentukan nilai untuk parameter berikut, jika Anda ingin:

   a. (Opsional) Untuk Versi, masukkan versi SSM Agent untuk menginstal. Anda dapat menginstal agen versi lama. Jika Anda tidak menentukan versinya, layanan akan menginstal versi terbaru.

   b. (Opsional) Untuk Izinkan Downgrade, pilih true untuk menginstal versi sebelumnya. SSM Agent Jika Anda memilih opsi ini, tentukan nomor versi sebelumnya. Pilih salah untuk menginstal hanya versi layanan terbaru.
6. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

> **Tip**
>
> Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

7. Untuk Parameter lainnya:

   - Untuk Komentar, ketik informasi tentang perintah ini.
   - Untuk Waktu habis (detik), tentukan jumlah detik untuk menunggu sistem sebelum gagal menjalankan perintah keseluruhan.

8. Untuk Pengendalian rate:

   - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

     > **Note**
     >
     > Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

   - Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

9. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan milik pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang

ditentukan berbeda Akun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

10. Di bagian Notifikasi SNS, jika Anda ingin notifikasi dikirim tentang status eksekusi perintah, pilih kotak centang Aktifkan notifikasi SNS.

    Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Run Command Amazon SNS, lihat. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

11. Pilih Jalankan.

Memperbarui PowerShell menggunakan Run Command

Prosedur berikut menjelaskan cara memperbarui PowerShell ke versi 5.1 pada node terkelola Windows Server 2012 dan 2012 R2 Anda. Skrip yang disediakan dalam prosedur ini mengunduh pembaruan Windows Management Framework (WMF) versi 5.1, dan memulai instalasi pembaruan. Node reboot selama proses ini karena ini diperlukan saat menginstal WMF 5.1. Unduhan dan instalasi pembaruan memakan waktu sekitar lima menit untuk diselesaikan.

Untuk memperbarui PowerShell menggunakan Run Command

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Run Command.
3. Pilih Run Command.
4. Di daftar Dokumen perintah, pilih **`AWS-RunPowerShellScript`**.
5. Di bagian Perintah, tempelkan perintah berikut untuk sistem operasi Anda.

   Server Windows 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. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

> **Tip**
>
> Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

7. Untuk Parameter lainnya:

   - Untuk Komentar, ketik informasi tentang perintah ini.
   - Untuk Waktu habis (detik), tentukan jumlah detik untuk menunggu sistem sebelum gagal menjalankan perintah keseluruhan.

8. Untuk Pengendalian rate:

   - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

> **Note**
>
> Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa

banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

- Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

9. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan milik pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbeda Akun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

10. Di bagian Notifikasi SNS, jika Anda ingin notifikasi dikirim tentang status eksekusi perintah, pilih kotak centang Aktifkan notifikasi SNS.

    Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Run Command Amazon SNS, lihat. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

11. Pilih Jalankan.

Setelah node terkelola reboot dan instalasi pembaruan selesai, sambungkan ke node Anda untuk mengonfirmasi bahwa PowerShell berhasil ditingkatkan ke versi 5.1. Untuk memeriksa versi PowerShell pada node Anda, buka PowerShell dan masukkan`$PSVersionTable`. Nilai `PSVersion` dalam tabel output menunjukkan 5.1 jika pemutakhiran berhasil.

Jika nilai `PSVersion` berbeda dari 5.1, misalnya 3.0 atau 4.0, tinjau log Pengaturan di Penampil Kejadian pada Log Windows. Log ini menunjukkan mengapa instalasi pembaruan gagal.

Walkthrough: Gunakan dengan AWS CLIRun Command

Contoh panduan berikut ini menunjukkan cara menggunakan AWS Command Line Interface (AWS CLI) untuk melihat informasi tentang perintah dan parameter perintah, cara menjalankan perintah, dan cara melihat status perintah tersebut.

> **Important**
>
> Hanya administrator tepercaya yang boleh diizinkan untuk menggunakan dokumen AWS Systems Manager yang telah dikonfigurasikan yang ditampilkan dalam topik ini. Perintah atau skrip yang ditentukan dalam dokumen Systems Manager berjalan dengan izin administratif pada node terkelola Anda. Jika pengguna memiliki izin untuk menjalankan dokumen Systems Manager yang telah ditentukan sebelumnya (dokumen apa pun yang dimulai dengan`AWS-`), maka pengguna tersebut juga memiliki akses administrator ke node. Untuk semua pengguna lain, Anda harus membuat dokumen yang restriktif dan membagikannya dengan pengguna tertentu.

Topik

Langkah 1: Memulai

Anda harus memiliki izin administrator pada node terkelola yang ingin Anda konfigurasi atau Anda harus telah diberikan izin yang sesuai di AWS Identity and Access Management (IAM). Perhatikan juga, contoh ini menggunakan Wilayah Timur AS (Ohio) (us-timur-2). Run Commandtersedia di titik akhir layanan Systems Manager yang Wilayah AWS tercantum di. Referensi Umum Amazon Web Services Untuk informasi selengkapnya, lihat Menyiapkan AWS Systems Manager.

Untuk menjalankan perintah menggunakan AWS CLI

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI.

2. Cantumkan semua dokumen yang tersedia.

   Perintah ini mencantumkan semua dokumen yang tersedia untuk akun Anda berdasarkan izin IAM.

   ```
   aws ssm list-documents
   ```

3. Verifikasi bahwa node terkelola siap menerima perintah.

   Output dari perintah berikut menunjukkan jika node terkelola sedang online.

   Linux & macOS

   ```
   aws ssm describe-instance-information \
       --output text --query "InstanceInformationList[*]"
   ```

   Windows

   ```
   aws ssm describe-instance-information ^
       --output text --query "InstanceInformationList[*]"
   ```

4. Jalankan perintah berikut untuk melihat detail tentang node terkelola tertentu.

   > **Note**
   >
   > Untuk menjalankan perintah dalam panduan ini, ganti ID instans dan perintah. Untuk perangkat AWS IoT Greengrass inti terkelola, gunakan mi- *ID_Number* misalnya ID. ID perintah dikembalikan sebagai respon terhadap send-command. ID Instance tersedia dariFleet Manager, kemampuanAWS 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
```

Langkah 2: Jalankan skrip shell untuk melihat detail sumber daya

Menggunakan Run Command dan `AWS-RunShellScript` dokumen, Anda dapat menjalankan perintah atau skrip apa pun pada node terkelola seolah-olah Anda masuk secara lokal.

Lihat deskripsi dan parameter yang tersedia

Jalankan perintah berikut untuk melihat deskripsi dokumen JSON Systems Manager.

Linux & macOS

```
aws ssm describe-document \
    --name "AWS-RunShellScript" \
    --query "[Document.Name,Document.Description]"
```

Windows

```
aws ssm describe-document ^
    --name "AWS-RunShellScript" ^
    --query "[Document.Name,Document.Description]"
```

Jalankan perintah berikut untuk melihat parameter yang tersedia dan detail tentang parameter tersebut.

Linux & macOS

```
aws ssm describe-document \
    --name "AWS-RunShellScript" \
    --query "Document.Parameters[*]"
```

Windows

```
aws ssm describe-document ^
    --name "AWS-RunShellScript" ^
```

```
    --query "Document.Parameters[*]"
```

Langkah 3: Kirim perintah sederhana menggunakan dokumen **`AWS-RunShellScript`**

Jalankan perintah berikut untuk mendapatkan informasi IP untuk node yang dikelola Linux.

Jika Anda menargetkan node Windows Server terkelola, ubah `document-name` ke `AWS-RunPowerShellScript` dan ubah `command` dari `ifconfig` keipconfig.

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
```

Dapatkan informasi perintah dengan data respon

Perintah berikut ini menggunakan Command ID yang dikembalikan dari perintah sebelumnya untuk mendapatkan detail dan data respon eksekusi perintah. Sistem mengembalikan data respon jika perintah selesai. Jika eksekusi perintah menunjukkan `"Pending"` atau `"InProgress"` Anda menjalankan perintah ini lagi untuk melihat data respon.

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
```

Identifikasi pengguna

Perintah berikut menampilkan pengguna default yang menjalankan perintah.

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")
```

Dapatkan status perintah

Perintah berikut menggunakan Command ID untuk mendapatkan status eksekusi perintah pada node yang dikelola. Contoh ini menggunakan Command ID yang dikembalikan pada perintah sebelumnya.

Linux & macOS

```
aws ssm list-commands \
    --command-id "command-ID"
```

Windows

```
aws ssm list-commands ^
    --command-id "command-ID"
```

Dapatkan detail perintah

Perintah berikut menggunakan Command ID dari perintah sebelumnya untuk mendapatkan status eksekusi perintah pada basis per node yang dikelola.

Linux & macOS

```
aws ssm list-command-invocations \
    --command-id "command-ID" \
    --details
```

Windows

```
aws ssm list-command-invocations ^
    --command-id "command-ID" ^
    --details
```

Dapatkan informasi perintah dengan data respons untuk node terkelola tertentu

Perintah berikut mengembalikan output dari `aws ssm send-command` permintaan asli untuk node terkelola tertentu.

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
```

Tampilkan versi Python

Perintah berikut mengembalikan versi Python berjalan pada node.

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}"'
```

Langkah 4: Jalankan skrip Python sederhana menggunakan Run Command

Perintah berikut menjalankan Python sederhana “Hello World” script menggunakan. 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}"'
```

Langkah 5: Jalankan skrip Bash menggunakan Run Command

Contoh di bagian ini menunjukkan bagaimana menjalankan skrip bash berikut menggunakanRun Command.

Untuk contoh penggunaan Run Command untuk menjalankan skrip yang disimpan di lokasi terpencil, lihat Menjalankan skrip dari Amazon S3 danMenjalankan skrip dari 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
```

Skrip ini menginstal AWS CodeDeploy agen di Amazon Linux dan Red Hat Enterprise Linux (RHEL) instans, seperti yang dijelaskan dalam Buat instans Amazon EC2 CodeDeploy untuk dalam AWS CodeDeploy Panduan Pengguna.

Skrip menginstal CodeDeploy agen dari ember S3 AWS terkelola di Wilayah Timur AS (Ohio) Anda (us-timur-2),. `aws-codedeploy-us-east-2`

Jalankan skrip bash dalam sebuah perintah AWS CLI

Contoh berikut ini menunjukkan cara menyertakan skrip bash dalam perintah CLI menggunakan opsi `--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"]}'
```

Jalankan skrip bash dalam file JSON

Pada contoh berikut ini, konten dari skrip bash disimpan dalam file JSON, dan file tersebut disertakan dalam perintah menggunakan opsi `--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
```

Konten dari file `installCodeDeployAgent.json` yang direferensikan ini ditampilkan dalam contoh berikut.

```
{
    "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"
        ]
    }
}
```

Walkthrough: GunakanAWS Tools for Windows PowerShell denganRun Command

Contoh berikut ini menunjukkan cara menggunakan AWS Tools for Windows PowerShell untuk melihat informasi tentang perintah dan parameter perintah, cara menjalankan perintah, dan cara melihat status perintah tersebut. Panduan ini mencakup contoh untuk masing-masing dokumen AWS Systems Manager yang telah ditetapkan sebelumnya.

> **Important**
>
> Hanya administrator tepercaya yang boleh diizinkan untuk menggunakan dokumen Systems Manager yang telah dikonfigurasikan yang ditampilkan dalam topik ini. Perintah atau skrip yang ditentukan dalam dokumen Systems Manager dijalankan dengan izin administratif pada node yang dikelola Anda. Jika pengguna memiliki izin untuk menjalankan salah satu dokumen Systems Manager yang telah ditentukan sebelumnya (dokumen yang dimulai denganAWS), maka pengguna tersebut juga memiliki akses administrator ke node. Untuk semua pengguna lain, Anda harus membuat dokumen yang restriktif dan membagikannya dengan pengguna tertentu.

Topik

- Konfigurasikan pengaturan sesi AWS Tools for Windows PowerShell
- Cantumkan semua dokumen yang tersedia
- JalankanPowerShell perintah atau skrip
- Instal aplikasi menggunakan dokumen AWS-InstallApplication
- InstalPowerShell modul menggunakan dokumenAWS-InstallPowerShellModule JSON
- Bergabunglah dengan node yang dikelola ke Domain menggunakan dokumenAWS-JoinDirectoryServiceDomain JSON
- Kirim metrik Windows ke AmazonCloudWatch Logs menggunakanAWS-ConfigureCloudWatch dokumen
- Memperbarui EC2Config menggunakan dokumen AWS-UpdateEC2Config
- Mengaktifkan atau menonaktifkan pembaruan otomatis Windows menggunakan dokumen AWS-ConfigureWindowsUpdate
- Mengelola pembaruan Windows menggunakanRun Command

Konfigurasikan pengaturan sesi AWS Tools for Windows PowerShell

Tentukan kredensial Anda

Buka Tools for WindowsPowerShell pada komputer lokal Anda dan jalankan perintah berikut ini untuk menentukan kredensials Anda. Anda harus memiliki izin administrator pada node yang dikelola yang ingin Anda konfigurasikan atau telah diberi izin yang sesuai diAWS Identity and Access Management (IAM). Untuk informasi selengkapnya, lihat Menyiapkan AWS Systems Manager.

```
Set-AWSCredentials –AccessKey key-name –SecretKey key-name
```

Tetapkan Wilayah AWS default

Jalankan perintah berikut ini untuk mengatur Region untukPowerShell sesi Anda. Contoh ini menggunakan Region US East (Ohio) (us-east-2). Run Commandtersedia dalam endpoint layanan Systems Manager yangWilayah AWS tercantum di Referensi Umum Amazon Web Services.

```
Set-DefaultAWSRegion `
    -Region us-east-2
```

Cantumkan semua dokumen yang tersedia

Perintah ini mencantumkan semua dokumen yang tersedia untuk akun Anda.

```
Get-SSMDocumentList
```

JalankanPowerShell perintah atau skrip

MenggunakanRun Command dan`AWS-RunPowerShell` dokumen, Anda dapat menjalankan perintah atau skrip pada node yang dikelola seolah-olah Anda masuk secara lokal. Anda dapat mengeluarkan perintah atau memasukkan jalur ke skrip lokal untuk menjalankan perintah.

> **Note**
>
> Untuk informasi tentang me-reboot node yang dikelola saat menggunakanRun Command untuk memanggil skrip, lihatMenangani reboot saat menjalankan perintah.

Lihat deskripsi dan parameter yang tersedia

```
Get-SSMDocumentDescription `
    -Name "AWS-RunPowerShellScript"
```

Lihat informasi selengkapnya tentang parameter

```
Get-SSMDocumentDescription `
    -Name "AWS-RunPowerShellScript" | Select -ExpandProperty Parameters
```

Kirim perintah menggunakan dokumen **`AWS-RunPowerShellScript`**

Perintah berikut ini menunjukkan konten`"C:\Users"` direktori dan konten`"C:\"` direktori pada dua node yang dikelola.

```
$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:\')}
```

Dapatkan detail permintaan perintah

Perintah berikut ini menggunakan`CommandId` untuk mendapatkan status eksekusi perintah pada kedua node yang dikelola. Contoh ini menggunakan `CommandId` yang dikembalikan pada perintah sebelumnya.

```
Get-SSMCommand `
    -CommandId $runPSCommand.CommandId
```

Status perintah dalam contoh ini dapat berupa Sukses, Tertunda, ataulnProgress.

Dapatkan informasi perintah per node yang dikelola

Perintah berikut ini menggunakanCommandId dari perintah sebelumnya untuk mendapatkan status eksekusi perintah pada basis node per dikelola.

```
Get-SSMCommandInvocation `
    -CommandId $runPSCommand.CommandId
```

Dapatkan informasi perintah dengan data respon untuk node yang dikelola tertentu

Perintah berikut ini mengembalikan output dari aslinyaSend-SSMCommand untuk node dikelola tertentu.

```
Get-SSMCommandInvocation `
    -CommandId $runPSCommand.CommandId `
    -Details $true `
    -InstanceId instance-ID | Select -ExpandProperty CommandPlugins
```

Batalkan perintah

Perintah berikut ini membatalkan Send-SSMCommand untuk dokumen 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
```

Periksa status perintah

Perintah berikut ini memeriksa status perintah Cancel.

```
Get-SSMCommand `
```

```
    -CommandId $cancelCommand.CommandId
```

Instal aplikasi menggunakan dokumen **`AWS-InstallApplication`**

MenggunakanRun Command dan`AWS-InstallApplication` dokumen, Anda dapat menginstal, atau menghapus aplikasi pada node yang dikelola. Perintah ini memerlukan jalur atau alamat ke sebuah MSI.

> **Note**
>
> Untuk informasi tentang me-reboot node yang dikelola saat menggunakanRun Command untuk memanggil skrip, lihatMenangani reboot saat menjalankan perintah.

Lihat deskripsi dan parameter yang tersedia

```
Get-SSMDocumentDescription `
    -Name "AWS-InstallApplication"
```

Lihat informasi selengkapnya tentang parameter

```
Get-SSMDocumentDescription `
    -Name "AWS-InstallApplication" | Select -ExpandProperty Parameters
```

Kirim perintah menggunakan dokumen **`AWS-InstallApplication`**

Perintah berikut ini menginstal versi Python pada node yang dikelola Anda dalam mode tanpa pengawasan, dan mencatat output ke file teks lokal pada`C:` drive Anda.

```
$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'}
```

Dapatkan informasi perintah per node yang dikelola

Perintah berikut ini menggunakan `CommandId` untuk mendapatkan status eksekusi perintah.

```
Get-SSMCommandInvocation `
    -CommandId $installAppCommand.CommandId `
```

```
    -Details $true
```

Dapatkan informasi perintah dengan data respon untuk node yang dikelola tertentu

Perintah berikut ini mengembalikan hasil instalasi Python.

```
Get-SSMCommandInvocation `
    -CommandId $installAppCommand.CommandId `
    -Details $true `
    -InstanceId instance-ID | Select -ExpandProperty CommandPlugins
```

InstalPowerShell modul menggunakan dokumen**AWS-InstallPowerShellModule** JSON

Anda dapat menggunakanRun Command untuk menginstalPowerShell modul pada node yang dikelola. Untuk informasi selengkapnya tentangPowerShell modul, lihat PowerShellModul Windows.

Lihat deskripsi dan parameter yang tersedia

```
Get-SSMDocumentDescription `
    -Name "AWS-InstallPowerShellModule"
```

Lihat informasi selengkapnya tentang parameter

```
Get-SSMDocumentDescription `
    -Name "AWS-InstallPowerShellModule" | Select -ExpandProperty Parameters
```

InstalPowerShell modul

Perintah berikut ini mengunduh file EZOut.zip, menginstalnya, lalu menjalankan perintah tambahan untuk menginstal penampil XPS. Terakhir, output dari perintah ini diunggah ke bucket S3 bernama "demo-ssm-output-bucket”.

```
$installPSCommand = Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-InstallPowerShellModule" `
    -Parameter @{'source'='https://gallery.technet.microsoft.com/EZOut-33ae0fb7/
file/110351/1/EZOut.zip';'commands'=@('Add-WindowsFeature -name XPS-Viewer -restart')}
 `
    -OutputS3BucketName demo-ssm-output-bucket
```

Dapatkan informasi perintah per node yang dikelola

Perintah berikut ini menggunakan `CommandId` untuk mendapatkan status eksekusi perintah.

```
Get-SSMCommandInvocation `
    -CommandId $installPSCommand.CommandId `
    -Details $true
```

Dapatkan informasi perintah dengan data respon untuk node yang dikelola

Perintah berikut ini mengembalikan output dari `Send-SSMCommand` asli untuk `CommandId` tertentu.

```
Get-SSMCommandInvocation `
    -CommandId $installPSCommand.CommandId `
    -Details $true | Select -ExpandProperty CommandPlugins
```

Bergabunglah dengan node yang dikelola ke Domain menggunakan dokumen**`AWS-JoinDirectoryServiceDomain`** JSON

MenggunakanRun Command, Anda dapat dengan cepat menggabungkan node yang dikelola keAWS Directory Service domain. Sebelum menjalankan perintah ini, buat direktori. Kami juga merekomendasikan agar Anda mempelajari selengkapnya tentang AWS Directory Service. Untuk informasi selengkapnya, lihat Panduan Administrasi AWS Directory Service.

Anda hanya dapat menggabungkan node yang dikelola ke sebuah domain. Anda tidak dapat menghapus sebuah node dari sebuah domain.

> **Note**
>
> Untuk informasi tentang node terkelola saat menggunakanRun Command untuk memanggil skrip, lihatMenangani reboot saat menjalankan perintah.

Lihat deskripsi dan parameter yang tersedia

```
Get-SSMDocumentDescription `
    -Name "AWS-JoinDirectoryServiceDomain"
```

Lihat informasi selengkapnya tentang parameter

```
Get-SSMDocumentDescription `
    -Name "AWS-JoinDirectoryServiceDomain" | Select -ExpandProperty Parameters
```

Bergabung dengan node yang dikelola ke domain

Perintah berikut ini menggabungkan node yang dikelola keAWS Directory Service domain yang diberikan dan mengunggah output yang dihasilkan ke bucket Amazon Simple Storage Service (Amazon S3) contoh.

```
$domainJoinCommand = Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-JoinDirectoryServiceDomain" `
    -Parameter @{'directoryId'='d-example01'; 'directoryName'='ssm.example.com';
 'dnsIpAddresses'=@('192.168.10.195', '192.168.20.97')} `
    -OutputS3BucketName demo-ssm-output-bucket
```

Dapatkan informasi perintah per node yang dikelola

Perintah berikut ini menggunakan `CommandId` untuk mendapatkan status eksekusi perintah.

```
Get-SSMCommandInvocation `
    -CommandId $domainJoinCommand.CommandId `
    -Details $true
```

Dapatkan informasi perintah dengan data respon untuk node yang dikelola

Perintah ini mengembalikan output dari `Send-SSMCommand` asli untuk `CommandId` tertentu.

```
Get-SSMCommandInvocation `
    -CommandId $domainJoinCommand.CommandId `
    -Details $true | Select -ExpandProperty CommandPlugins
```

Kirim metrik Windows ke AmazonCloudWatch Logs menggunakan**`AWS-ConfigureCloudWatch`** dokumen

Anda dapat mengirimWindows Server pesan di log aplikasi, sistem, keamanan, dan Event Tracing for Windows (ETW) ke AmazonCloudWatch Logs. Ketika Anda mengizinkan pencatatan untuk pertama kalinya, Systems Manager mengirimkan semua log yang dihasilkan dalam satu (1) menit dari waktu Anda mulai mengunggah log untuk log aplikasi, sistem, keamanan, dan ETW. Log yang terjadi sebelum waktu ini tidak disertakan. Jika Anda menonaktifkan pencatatan dan kemudian mengaktifkan log kembali, Systems Manager mengirimkan log dari waktu yang tersisa. Untuk setiap file log kustom dan log Internet Information Services (IIS), Systems Manager membaca berkas log dari awal. Selain itu, Systems Manager juga dapat mengirim data pengukur performa keCloudWatch Logs.

Jika sebelumnya Anda mengaktifkanCloudWatch integrasi di EC2Config, pengaturan Systems Manager akan mengganti pengaturan apa pun yang disimpan secara lokal pada node yang dikelola dalam`C:\Program Files\Amazon\EC2ConfigService\Settings\AWS.EC2.Windows.CloudWatch.json` file. Untuk informasi selengkapnya tentang penggunaan EC2Config untuk mengelola penghitung kinerja dan log pada satu node terkelola, lihat Mengumpulkan metrik dan log dari instans Amazon EC2 dan server lokal denganCloudWatch agen di PanduanCloudWatch Pengguna Amazon.

Lihat deskripsi dan parameter yang tersedia

```
Get-SSMDocumentDescription `
    -Name "AWS-ConfigureCloudWatch"
```

Lihat informasi selengkapnya tentang parameter

```
Get-SSMDocumentDescription `
    -Name "AWS-ConfigureCloudWatch" | Select -ExpandProperty Parameters
```

Kirim log aplikasi keCloudWatch

Perintah berikut ini mengkonfigurasi node yang dikelola dan memindahkan log Aplikasi Windows keCloudWatch.

```
$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"]}}}'}
```

Dapatkan informasi perintah per node yang dikelola

Perintah berikut ini menggunakan `CommandId` untuk mendapatkan status eksekusi perintah.

```
Get-SSMCommandInvocation `
    -CommandId $cloudWatchCommand.CommandId `
    -Details $true
```

Dapatkan informasi perintah dengan data respon untuk node yang dikelola tertentu

Perintah berikut ini mengembalikan hasilCloudWatch konfigurasi Amazon.

```
Get-SSMCommandInvocation `
    -CommandId $cloudWatchCommand.CommandId `
    -Details $true `
    -InstanceId instance-ID | Select -ExpandProperty CommandPlugins
```

Kirim penghitung kinerja untukCloudWatch menggunakan**AWS-ConfigureCloudWatch** dokumen

Perintah demonstrasi berikut ini mengunggah pengukur performa keCloudWatch. Untuk informasi lebih lanjut, lihat Panduan Pengguna 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"]}}}'}
```

Memperbarui EC2Config menggunakan dokumen **AWS-UpdateEC2Config**

MenggunakanRun Command danAWS-EC2ConfigUpdate dokumen, Anda dapat memperbarui layanan EC2Config yang berjalan pada node yangWindows Server dikelola. Perintah ini dapat memperbarui layanan EC2Config ke versi terbaru atau versi yang Anda tentukan.

Lihat deskripsi dan parameter yang tersedia

```
Get-SSMDocumentDescription `
    -Name "AWS-UpdateEC2Config"
```

Lihat informasi selengkapnya tentang parameter

```
Get-SSMDocumentDescription `
```

```
    -Name "AWS-UpdateEC2Config" | Select -ExpandProperty Parameters
```

Memperbarui EC2Config ke versi terbaru

```
$ec2ConfigCommand = Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-UpdateEC2Config"
```

Dapatkan informasi perintah dengan data respon untuk node yang dikelola

Perintah ini mengembalikan output dari perintah yang ditentukan dari `Send-SSMCommand` sebelumnya.

```
Get-SSMCommandInvocation `
    -CommandId $ec2ConfigCommand.CommandId `
    -Details $true `
    -InstanceId instance-ID | Select -ExpandProperty CommandPlugins
```

Memperbarui EC2Config ke versi tertentu

Perintah berikut ini menurunkan EC2Config ke versi yang lebih lama.

```
Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-UpdateEC2Config" `
    -Parameter @{'version'='4.9.3519'; 'allowDowngrade'='true'}
```

Mengaktifkan atau menonaktifkan pembaruan otomatis Windows menggunakan dokumen **`AWS-ConfigureWindowsUpdate`**

MenggunakanRun Command dan`AWS-ConfigureWindowsUpdate` dokumen, Anda dapat mengaktifkan atau menonaktifkan pembaruan Windows otomatis pada node yangWindows Server dikelola. Perintah ini mengkonfigurasi Windows Update Agent untuk mengunduh dan menginstal pembaruan Windows pada hari dan jam yang Anda tentukan. Jika pembaruan memerlukan reboot, node yang dikelola me-reboot secara otomatis 15 menit setelah pembaruan telah diinstal. Dengan perintah ini Anda juga dapat mengkonfigurasi Windows Update untuk memeriksa pembaruan tetapi tidak menginstalnya. Dokumen `AWS-ConfigureWindowsUpdate` kompatibel dengan Windows Server 2008, 2008 R2, 2012, 2012 R2, dan 2016.

Lihat deskripsi dan parameter yang tersedia

```
Get-SSMDocumentDescription `
    -Name "AWS-ConfigureWindowsUpdate"
```

Lihat informasi selengkapnya tentang parameter

```
Get-SSMDocumentDescription `
    -Name "AWS-ConfigureWindowsUpdate" | Select -ExpandProperty Parameters
```

Aktifkan pembaruan otomatis Windows

Perintah berikut ini mengkonfigurasi Windows Update untuk secara otomatis mengunduh dan menginstal pembaruan setiap hari pada pukul 10:00 malam.

```
$configureWindowsUpdateCommand = Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-ConfigureWindowsUpdate" `
    -Parameters @{'updateLevel'='InstallUpdatesAutomatically';
 'scheduledInstallDay'='Daily'; 'scheduledInstallTime'='22:00'}
```

Lihat status perintah untuk mengizinkan pembaruan otomatis Windows

Perintah berikut ini menggunakan `CommandId` untuk mendapatkan status eksekusi perintah untuk mengizinkan pembaruan otomatis Windows.

```
Get-SSMCommandInvocation `
    -Details $true `
    -CommandId $configureWindowsUpdateCommand.CommandId | Select -ExpandProperty
 CommandPlugins
```

Nonaktifkan pembaruan otomatis Windows

Perintah berikut ini menurunkan tingkat notifikasi Windows Update sehingga sistem memeriksa pembaruan tetapi tidak secara otomatis memperbarui node yang dikelola.

```
$configureWindowsUpdateCommand = Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-ConfigureWindowsUpdate" `
    -Parameters @{'updateLevel'='NeverCheckForUpdates'}
```

Lihat status perintah untuk menonaktifkan pembaruan otomatis Windows

Perintah berikut ini menggunakan `CommandId` untuk mendapatkan status eksekusi perintah untuk menonaktifkan pembaruan otomatis Windows.

```
Get-SSMCommandInvocation `
    -Details $true `
    -CommandId $configureWindowsUpdateCommand.CommandId | Select -ExpandProperty
 CommandPlugins
```

Mengelola pembaruan Windows menggunakanRun Command

MenggunakanRun Command dan`AWS-InstallWindowsUpdates` dokumen, Anda dapat mengelola pembaruan untuk node yangWindows Server dikelola. Perintah ini memindai atau menginstal pembaruan yang hilang pada node yang dikelola dan secara opsional me-reboot setelah instalasi. Anda juga dapat menentukan klasifikasi dan tingkat kepelikan yang sesuai agar pembaruan diinstal di lingkungan Anda.

> **Note**
>
> Untuk informasi tentang me-reboot node yang dikelola saat menggunakanRun Command untuk memanggil skrip, lihatMenangani reboot saat menjalankan perintah.

Contoh berikut ini mendemonstrasikan cara melakukan tugas manajemen Windows Update yang ditentukan.

Cari semua pembaruan Windows yang hilang

```
Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-InstallWindowsUpdates" `
    -Parameters @{'Action'='Scan'}
```

Instal pembaruan Windows tertentu

```
Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-InstallWindowsUpdates" `
    -Parameters @{'Action'='Install';'IncludeKbs'='kb-ID-1,kb-ID-2,kb-
ID-3';'AllowReboot'='True'}
```

Instal pembaruan Windows penting yang hilang

```
Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-InstallWindowsUpdates" `
    -Parameters @{'Action'='Install';'SeverityLevels'='Important';'AllowReboot'='True'}
```

Instal pembaruan Windows yang hilang dengan pengecualian khusus

```
Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-InstallWindowsUpdates" `
    -Parameters @{'Action'='Install';'ExcludeKbs'='kb-ID-1,kb-
ID-2';'AllowReboot'='True'}
```

Memecahkan masalah Run Command Systems Manager

Run Command, kemampuanAWS Systems Manager, memberikan detail status dengan setiap eksekusi perintah. Untuk informasi lebih lanjut tentang detail status perintah, lihat Memahami status perintah. Anda juga dapat menggunakan informasi dalam topik ini untuk membantu memecahkan masalah denganRun Command.

Topik

- Beberapa node terkelola saya hilang
- Sebuah langkah dalam skrip saya gagal, tetapi status keseluruhan adalah 'berhasil'
- SSM Agenttidak berjalan dengan benar

Beberapa node terkelola saya hilang

DiPerintah Runhalaman, setelah Anda memilih dokumen SSM untuk dijalankan dan pilihInstans memilih secara manualdi dalamTargetbagian, sebuah daftar ditampilkan dari node terkelola yang dapat Anda pilih untuk menjalankan perintah.

Jika sebuah node terkelola yang Anda harapkan tidak tercantum, lihatMemecahkan masalah ketersediaan node terkelolauntuk kiat pemecahan masalah.

Setelah Anda membuat, mengaktifkan, reboot, atau memulai ulang node yang dikelola, instalRun Commandpada node, atau melampirkanAWS Identity and Access Management(IAM) instans profil ke node, mungkin diperlukan waktu beberapa menit sampai node terkelola ditambahkan ke daftar.

Sebuah langkah dalam skrip saya gagal, tetapi status keseluruhan adalah 'berhasil'

MenggunakanRun Command, Anda dapat menentukan cara skrip Anda menangani kode keluar. Secara default, kode keluar dari perintah terakhir yang dijalankan dalam skrip dilaporkan sebagai kode keluar untuk seluruh skrip. Namun, Anda dapat menyertakan pernyataan bersyarat untuk keluar dari skrip jika perintah sebelum yang terakhir gagal. Untuk informasi dan contoh, lihat Tentukan kode keluar dalam perintah.

SSM Agenttidak berjalan dengan benar

Jika Anda mengalami masalah menjalankan perintah menggunakanRun Command, mungkin ada masalah denganSSM Agent. Untuk informasi tentang menyelidiki masalah denganSSM Agent, LihatPemecahan Masalah SSM Agent.

AWS Systems Manager State Manager

State Manager, kemampuanAWS Systems Manager, adalah layanan manajemen konfigurasi yang aman dan dapat diskalakan yang mengotomatiskan proses menjaga node terkelola dan AWS sumber daya lainnya dalam keadaan yang Anda tentukan. Untuk memulaiState Manager, buka konsol Manajer Sistem. Di panel navigasi, pilih State Manager.

> **Note**
>
> State Managerdan Maintenance Windows dapat melakukan beberapa jenis pembaruan serupa pada node terkelola Anda. Pilihan mana yang Anda pilih bergantung pada apakah Anda perlu mengotomatisasi kepatuhan sistem atau melakukan tugas prioritas tinggi dan sensitif terhadap waktu selama periode yang ditentukan.
> Untuk informasi selengkapnya, lihat Memilih antara State Manager dan Maintenance Windows.

Bagaimana State Manager manfaat organisasi saya?

Dengan menggunakan dokumen Manajer Sistem (dokumen SSM) yang telah dikonfigurasi sebelumnya, State Manager menawarkan manfaat berikut untuk mengelola node Anda:

- Node bootstrap dengan perangkat lunak tertentu saat start-up.
- Download dan perbarui agen pada jadwal yang ditentukan, termasukSSM Agent.

- Mengkonfigurasi pengaturan jaringan.
- Bergabung node ke domain Microsoft Active Directory.
- Jalankan skrip di Linux,macOS, dan node yang dikelola Windows sepanjang siklus hidupnya.

Untuk mengelola drift konfigurasi di seluruh AWS sumber daya lain, Anda dapat menggunakan Otomasi, kemampuan Manajer Sistem, dengan State Manager untuk melakukan jenis tugas berikut:

- Lampirkan peran Manajer Sistem ke instans Amazon Elastic Compute Cloud (Amazon EC2) untuk menjadikannya node terkelola.
- Tegakkan masuk dan keluarnya aturan yang diinginkan untuk grup keamanan.
- Buat atau hapus backup Amazon DynamoDB.
- Buat atau hapus snapshot Amazon Elastic Block Store (Amazon EBS).
- Matikan izin baca dan tulis di bucket Amazon Simple Storage Service (Amazon S3).
- Mulai, mulai ulang, atau hentikan node terkelola dan instans Amazon Relational Database Service (Amazon RDS).
- Terapkan patch ke Linux, macOS, dan Window AMIs.

Untuk informasi tentang penggunaan State Manager dengan runbook Otomasi, lihatMenjadwalkan otomatisasi denganState Manager asosiasi.

Siapa yang harus menggunakanState Manager?

State Managersesuai untuk setiap AWS pelanggan yang ingin meningkatkan manajemen dan tata kelola AWS sumber daya mereka dan mengurangi penyimpangan konfigurasi.

Apa saja fitur dariState Manager?

Fitur utama dari State Manager meliputi:

- State Managerasosiasi

  State ManagerAsosiasi adalah konfigurasi yang Anda tetapkan ke AWS sumber daya Anda. Konfigurasi mendefinisikan status yang ingin Anda pertahankan pada sumber daya Anda. Misalnya, asosiasi dapat menentukan bahwa perangkat lunak antivirus harus diinstal dan berjalan pada node yang dikelola, atau bahwa port tertentu harus ditutup.

Asosiasi menentukan jadwal kapan harus menerapkan konfigurasi dan target untuk asosiasi. Misalnya, asosiasi untuk perangkat lunak antivirus mungkin berjalan sekali sehari pada semua node yang dikelola dalam sebuahAkun AWS. Jika perangkat lunak tidak diinstal pada node, maka asosiasi dapat menginstruksikan State Manager untuk menginstalnya. Jika perangkat lunak diinstal, tetapi layanan tidak berjalan, maka asosiasi dapat menginstruksikan State Manager untuk memulai layanan.

- Opsi penjadwalan yang fleksibel

  State Managermenawarkan opsi berikut untuk penjadwalan saat asosiasi berjalan:

  - Pemrosesan segera atau tertunda

    Ketika Anda membuat asosiasi, secara default, sistem segera menjalankannya pada sumber daya yang ditentukan. Setelah menjalankan awal, asosiasi berjalan dalam interval sesuai dengan jadwal yang Anda tetapkan.

    Anda dapat menginstruksikan untuk State Manager tidak segera menjalankan asosiasi dengan menggunakan asosiasi Terapkan hanya pada opsi interval Cron yang ditentukan berikutnya di konsol atau `ApplyOnlyAtCronInterval` parameter dari baris perintah.

  - Cron dan tingkat ekspresi

    Saat membuat asosiasi, Anda menentukan jadwal kapan State Manager menerapkan konfigurasi. State Managermendukung sebagian besar ekspresi cron dan tingkat standar untuk penjadwalan saat asosiasi berjalan. State Managerjuga mendukung ekspresi cron yang mencakup hari dalam seminggu dan tanda nomor (#) untuk menunjuk n th hari bulan untuk menjalankan asosiasi dan (L) tanda untuk menunjukkan X hari terakhir bulan.

    > **Note**
    >
    > State Managersaat ini tidak mendukung menentukan bulan dalam ekspresi cron untuk asosiasi.

    Untuk mengontrol lebih lanjut ketika asosiasi berjalan, misalnya jika Anda ingin menjalankan asosiasi dua hari setelah patch Selasa, Anda dapat menentukan offset. Offset mendefinisikan berapa hari untuk menunggu setelah hari yang dijadwalkan untuk menjalankan asosiasi.

Untuk informasi tentang membangun cron dan tingkat ekspresi, lihat. Referensi: Ekspresi cron dan rate untuk Systems Manager

- Beberapa opsi penargetan

  Sebuah asosiasi juga menentukan target untuk asosiasi. State Managermendukung penargetan AWS sumber daya dengan menggunakan tag,AWS Resource Groups, ID node individu, atau semua node yang dikelola di saat ini Wilayah AWS dan. Akun AWS

- Dukungan Amazon S3

  Simpan output perintah dari asosiasi yang berjalan di bucket Amazon S3 pilihan Anda. Untuk informasi selengkapnya, lihat Bekerja dengan asosiasi di Systems Manager.

- Dukungan EventBridge

  Kemampuan Manajer Sistem ini didukung sebagai jenis peristiwa dan jenis target dalam EventBridge aturan Amazon. Untuk informasi, lihat Pemantauan peristiwa Systems Manager dengan Amazon EventBridge dan Referensi: Pola dan jenis EventBridge acara Amazon untuk Systems Manager.

Apakah ada biaya untuk digunakanState Manager?

State Managertersedia tanpa biaya tambahan.

Bagaimana cara memulainyaState Manager?

Selesaikan tugas-tugas berikut untuk memulaiState Manager.

Tugas	Untuk Informasi Selengkapnya
Mengatur Manajer Sistem	Menyiapkan AWS Systems Manager
Pelajari selengkapnya tentang State Manager	Tentang State Manager
Buat dan tetapkan State Manager asosiasi ke node Anda	Bekerja dengan asosiasi di Systems Manager

Info lebih lanjut

- Memerangi Drift Konfigurasi Menggunakan Amazon EC2 Systems Manager dan Windows DSC PowerShell
- Mengonfigurasi Instans Amazon EC2 dalam Grup Penskalaan Otomatis Menggunakan State Manager

Topik

Tentang State Manager

State ManagerSebuah kemampuan dariAWS Systems Manager, adalah layanan yang aman dan terukur yang mengotomatiskan proses menjaga node terkelola dalamhybrid dan multicloudinfrastruktur dalam keadaan yang Anda definisikan.

Berikut bagaimana State Manager cara kerjanya:

1. Tentukan status yang ingin Anda terapkanAWSsumber daya.

   Apakah Anda ingin menjamin bahwa node terkelola Anda dikonfigurasi dengan aplikasi tertentu, seperti aplikasi antivirus atau malware? Apakah Anda ingin mengotomatiskan proses memperbaruiSSM Agentatau lainnyaAWSpaket-paket sepertiAWSPVDriver? Apakah Anda perlu menjamin bahwa port tertentu ditutup atau dibuka? Untuk memulai denganState Manager, tentukan keadaan yang ingin Anda terapkan padaAWSsumber daya. Status yang ingin Anda terapkan menentukan dokumen SSM mana yang Anda gunakan untuk membuatState Managerasosiasi.

   SEBUAHState Manager asosiasiadalah konfigurasi yang Anda tetapkan keAWSsumber daya. Konfigurasi mendefinisikan status yang ingin Anda pertahankan pada sumber daya Anda. Misalnya, asosiasi dapat menentukan bahwa perangkat lunak antivirus harus diinstal dan berjalan pada node yang dikelola, atau port tertentu harus ditutup.

   Asosiasi menentukan jadwal kapan harus menerapkan konfigurasi dan target untuk asosiasi. Misalnya, asosiasi untuk perangkat lunak antivirus dapat berjalan sekali sehari pada semua node yang dikelola dalamAkun AWS. Jika perangkat lunak tidak diinstal pada node, maka asosiasi

dapat menginstruksikanState Manageruntuk menginstalnya. Jika perangkat lunak diinstal, tetapi layanan tidak berjalan, maka asosiasi dapat menginstruksikanState Manageruntuk memulai layanan.

2. Tentukan apakah dokumen SSM yang telah dikonfigurasi dapat membantu Anda membuat status yang diinginkan padaAWSsumber daya.

   Systems Manager menyertakan puluhan dokumen SSM yang telah dikonfigurasi yang dapat Anda gunakan untuk membuat asosiasi. Dokumen yang telah dikonfigurasi siap untuk melakukan tugas-tugas umum seperti menginstal aplikasi, mengonfigurasi Amazon CloudWatch, berlariAWS Systems Managerotomatisasi, berjalan PowerShell dan skrip Shell, dan menggabungkan node terkelola ke domain layanan direktori untuk Active Directory.

   Anda dapat melihat semua dokumen SSM di konsol Systems Manager. Pilih nama dokumen untuk mempelajari lebih lanjut tentang masing-masing dokumen. Berikut ini adalah dua contoh: `AWS-ConfigureAWSPackage` dan `AWS-InstallApplication`.

3. Buat asosiasi.

   Anda dapat membuat asosiasi dengan menggunakan konsol Manajer Sistem,AWS Command Line Interface(AWS CLI),AWS Tools for Windows PowerShell(Alat untuk Windows PowerShell), atau API Manajer Sistem. Saat Anda membuat asosiasi, Anda menentukan informasi berikut:

   - Sebuah nama untuk asosiasi.
   - Parameter untuk dokumen SSM (misalnya, jalur ke aplikasi untuk menginstal atau skrip untuk berjalan pada node).
   - Menargetkan untuk asosiasi. Anda dapat menargetkan node terkelola dengan menentukan tag, dengan memilih ID node individual, atau dengan memilih grup diAWS Resource Groups. Anda juga bisa menargetkansegalanode terkelola di saat iniWilayah AWSdanAkun AWS.
   - Jadwal untuk kapan atau seberapa sering menerapkan status. Anda dapat menentukan ekspresi cron atau rate. Untuk informasi selengkapnya tentang membuat jadwal menggunakan ekspresi cron dan rate, lihat Ekspresi cron dan rate untuk associate.

     > **Note**
     >
     > State Managersaat ini tidak mendukung menentukan bulan dalam ekspresi cron untuk asosiasi.

   Saat Anda menjalankan perintah untuk membuat asosiasi, Manajer Sistem mengikat informasi yang Anda tentukan (jadwal, target, dokumen SSM, dan parameter) ke sumber daya yang

ditargetkan. Status asosiasi awalnya menunjukkan "Tertunda" saat sistem mencoba untuk mencapai semua target dan segera menerapkan status yang ditentukan dalam asosiasi.

> **Note**
>
> Jika Anda membuat asosiasi baru yang dijadwalkan untuk berjalan sementara asosiasi sebelumnya masih berjalan, asosiasi sebelumnya akan habis waktu dan asosiasi baru berjalan.

Manajer Sistem melaporkan status permintaan untuk membuat asosiasi pada sumber daya. Anda dapat melihat detail status di konsol atau (untuk node terkelola) dengan menggunakanDescribeInstanceAssociationsStatusOperasi API. Jika Anda memilih untuk menulis output perintah ke Amazon Simple Storage Service (Amazon S3) saat membuat asosiasi, Anda juga dapat melihat output di bucket Amazon S3 yang Anda tentukan.

Untuk informasi selengkapnya, lihat Bekerja dengan asosiasi di Systems Manager.

> **Note**
>
> Operasi API yang diprakarsai oleh dokumen SSM selama proses asosiasi tidak masukAWS CloudTrail.

4. Memantau dan memperbarui.

Setelah Anda membuat asosiasi,State Managermenerapkan kembali konfigurasi sesuai dengan jadwal yang Anda tentukan dalam asosiasi. Anda dapat melihat status asosiasi Anda diState Managerhalamandi konsol atau dengan langsung memanggil ID asosiasi yang dihasilkan oleh Manajer Sistem saat Anda membuat asosiasi. Untuk informasi selengkapnya, lihat Melihat riwayat asosiasi. Anda dapat memperbarui dokumen asosiasi Anda dan mengajukan permohonan kembali sesuai kebutuhan. Anda juga dapat membuat beberapa versi asosiasi. Untuk informasi selengkapnya, lihat Mengedit dan membuat versi baru asosiasi.

Kapan asosiasi diterapkan pada sumber daya?

Saat Anda membuat asosiasi, Anda menentukan dokumen SSM yang mendefinisikan konfigurasi, daftar sumber daya target, dan jadwal untuk menerapkan konfigurasi. Secara default,State

Managermenjalankan asosiasi saat Anda membuatnya dan kemudian sesuai dengan jadwal Anda.State Managerjuga mencoba untuk menjalankan asosiasi dalam situasi berikut:

- Suntingan asosiasi—State Managermenjalankan asosiasi setelah pengguna mengedit dan menyimpan perubahannya ke salah satu bidang asosiasi berikut:`DOCUMENT_VERSION`,`PARAMETERS`,`SCHEDULE_EXPRESSION`,`OUTPUT_S3_LOCATION`.
- Suntingan dokumen—State Managermenjalankan asosiasi setelah pengguna mengedit dan menyimpan perubahan pada dokumen SSM yang menentukan status konfigurasi asosiasi. Secara khusus, asosiasi berjalan setelah pengeditan dokumen berikut:
  - Seorang pengguna menentukan versi dokumen $DEFAULT baru dan asosiasi dibuat menggunakan versi $DEFAULT.
  - Seorang pengguna memperbarui dokumen dan asosiasi dibuat menggunakan versi $LATEST.
  - Pengguna menghapus dokumen yang ditentukan saat asosiasi dibuat.
- Parameter Storeperubahan nilai parameter—State Managermenjalankan asosiasi setelah pengguna mengedit nilai parameter yang ditentukan dalam asosiasi.
- Mulai manual—State Managermenjalankan asosiasi saat diprakarsai oleh pengguna baik dari konsol Manajer Sistem atau secara terprogram.
- Perubahan target—State Managermenjalankan asosiasi setelah salah satu aktivitas berikut terjadi pada instance target:
  - Sebuah contoh datang online untuk pertama kalinya.
  - Sebuah contoh datang online setelah melewatkan lari asosiasi terjadwal.
  - Sebuah contoh datang online setelah dihentikan selama lebih dari 30 hari.

> **Note**
>
> Pembaruan target tidak memengaruhi asosiasi yang dibuat menggunakan Otomatisasi Manajer Sistem.

Bekerja dengan asosiasi di Systems Manager

Bagian ini menjelaskan cara membuat dan mengelolaState Managerasosiasi dengan menggunakanAWS Systems Managerkonsol,AWS Command Line Interface(AWS CLI), danAWS Tools for PowerShell.

Topik

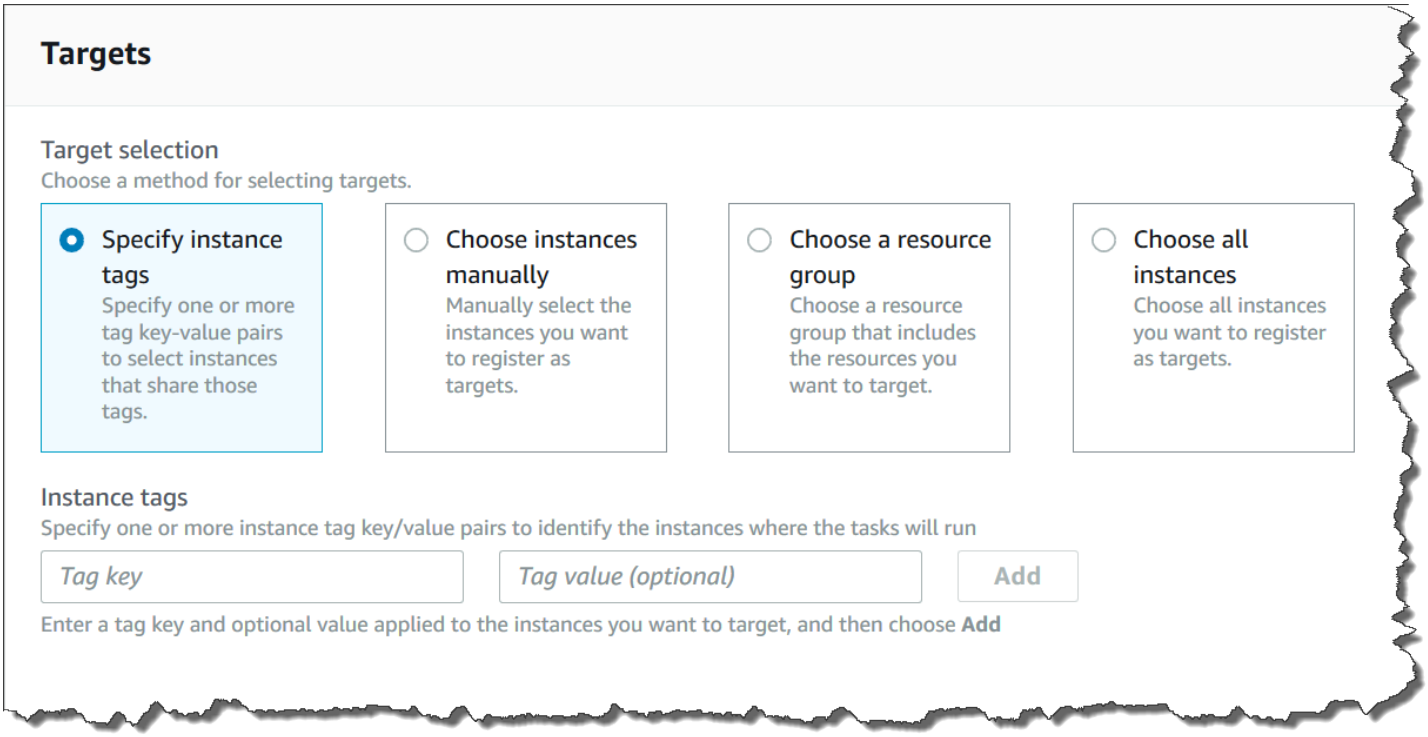
- Tentang target dan kontrol tingkat dalam State Manager asosiasi
- Membuat asosiasi
- Mengedit dan membuat versi baru asosiasi
- Menghapus asosiasi
- Menjalankan grup Auto Scaling dengan asosiasi
- Melihat riwayat asosiasi
- Bekerja dengan asosiasi menggunakan IAM

Tentang target dan kontrol tingkat dalam State Manager asosiasi

Topik ini menjelaskanState Manager, kemampuanAWS Systems Manager, fitur yang membantu Anda menyebarkan asosiasi ke lusinan atau ratusan node sambil mengontrol jumlah node yang menjalankan asosiasi pada waktu yang dijadwalkan.

Target

Saat membuat State Manager asosiasi, Anda memilih node mana yang akan dikonfigurasi dengan asosiasi di bagian Target pada konsol Systems Manager, seperti yang ditunjukkan di sini.



Jika Anda membuat asosiasi dengan menggunakan alat baris perintah seperti AWS Command Line Interface (AWS CLI), maka Anda menentukan parameter `targets`. Menargetkan node

memungkinkan Anda untuk mengkonfigurasi puluhan, ratusan, atau ribuan node dengan asosiasi tanpa harus menentukan atau memilih ID node individual.

Setiap node yang dikelola dapat ditargetkan oleh maksimal 20 asosiasi.

State Managertermasuk opsi target berikut saat membuat asosiasi.

Tentukan tag

Gunakan opsi ini untuk menentukan kunci tag dan (opsional) nilai tag yang ditetapkan ke node Anda. Saat Anda menjalankan permintaan, sistem akan menemukan dan mencoba membuat asosiasi pada semua node yang cocok dengan kunci dan nilai tag yang ditentukan. Jika Anda menentukan beberapa nilai tag, asosiasi menargetkan setiap node dengan setidaknya satu dari nilai tag tersebut. Ketika sistem awalnya menciptakan asosiasi, ia menjalankan asosiasi. Setelah menjalankan awal ini, sistem menjalankan asosiasi yang sesuai dengan jadwal yang Anda tentukan.

Jika Anda membuat node baru dan menetapkan kunci tag dan nilai yang ditentukan ke node tersebut, sistem secara otomatis menerapkan asosiasi, menjalankannya segera, dan kemudian menjalankannya sesuai dengan jadwal. Hal ini berlaku ketika asosiasi menggunakan dokumen Perintah atau Kebijakan dan tidak berlaku jika asosiasi menggunakan runbook Otomatisasi. Jika Anda menghapus tag yang ditentukan dari node, sistem tidak lagi menjalankan asosiasi pada node tersebut.

> **Note**
>
> Jika Anda menggunakan runbook Otomasi dengan State Manager dan batasan penandaan mencegah Anda mencapai tujuan tertentu, pertimbangkan untuk menggunakan runbook Otomasi dengan Amazon. EventBridge Untuk informasi selengkapnya, lihat Jalankan otomatisasi berdasarkan peristiwa. Untuk informasi selengkapnya tentang menggunakan runbook denganState Manager, lihatMenjadwalkan otomatisasi denganState Manager asosiasi.

Sebagai praktik terbaik, sebaiknya gunakan tag saat membuat asosiasi yang menggunakan dokumen Command atau Policy. Kami juga merekomendasikan penggunaan tag saat membuat asosiasi untuk menjalankan grup Auto Scaling. Untuk informasi selengkapnya, lihat Menjalankan grup Auto Scaling dengan asosiasi.

> **Note**
>
> Perhatikan informasi berikut.
>
> - Saat membuat asosiasi di konsol, saat menargetkan node dengan menggunakan tag, Anda hanya dapat menentukan satu kunci tag. Jika Anda ingin menggunakan konsol dan Anda ingin menargetkan node Anda dengan menggunakan lebih dari satu kunci tag, tetapkan kunci tag ke AWS Resource Groups grup dan tambahkan node ke dalamnya. Anda kemudian dapat memilih opsi Grup Sumber Daya dalam daftar Target saat Anda membuat State Manager asosiasi.
> - Anda dapat menentukan maksimal lima kunci tag dengan menggunakanAWS CLI. Jika Anda menggunakanAWS CLI, semua kunci tag yang ditentukan dalam `create-association` perintah harus saat ini ditetapkan ke node. Jika tidak, State Manager gagal menargetkan node untuk asosiasi. Untuk informasi tentang menetapkan tag ke node Anda, lihatPenandaan sumber daya Systems Manager.

Pilih node secara manual

Gunakan opsi ini untuk secara manual memilih node tempat Anda ingin membuat asosiasi. Panel Instances menampilkan semua node terkelola Systems Manager di saat ini Akun AWS dan. Wilayah AWS Anda dapat secara manual memilih node sebanyak yang Anda inginkan. Ketika sistem awalnya menciptakan asosiasi, ia menjalankan asosiasi. Setelah menjalankan awal ini, sistem menjalankan asosiasi yang sesuai dengan jadwal yang Anda tentukan.

> **Note**
>
> Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

Pilih grup sumber daya

Gunakan opsi ini untuk membuat asosiasi pada semua node yang dikembalikan oleh kueri AWS Resource Groups berbasis tag atau berbasis AWS CloudFormation tumpukan.

Di bawah ini adalah detail tentang penargetan kelompok sumber daya untuk asosiasi.

- Jika Anda menambahkan node baru ke grup, sistem secara otomatis memetakan node ke asosiasi yang menargetkan grup sumber daya. Sistem menerapkan asosiasi ke node ketika menemukan perubahan. Setelah menjalankan awal ini, sistem menjalankan asosiasi yang sesuai dengan jadwal yang Anda tentukan.
- Jika Anda membuat asosiasi yang menargetkan grup sumber daya dan jenis `AWS::SSM::ManagedInstance` sumber daya ditentukan untuk grup tersebut, maka menurut desain, asosiasi tersebut berjalan di instans Amazon Elastic Compute Cloud (Amazon EC2) dan node non-EC2 di lingkungan hybrid dan multicloud.
- Jika Anda membuat asosiasi yang menargetkan grup sumber daya, grup sumber daya tidak boleh memiliki lebih dari lima kunci tag yang ditetapkan padanya atau lebih dari lima nilai yang ditentukan untuk salah satu kunci tag. Jika salah satu dari kondisi ini berlaku untuk tag dan kunci yang ditetapkan ke grup sumber daya Anda, asosiasi gagal dijalankan dan mengembalikan `InvalidTarget` kesalahan.
- Jika Anda menghapus grup sumber daya, semua instans dalam grup tersebut tidak lagi menjalankan asosiasi. Sebagai praktik terbaik, hapus asosiasi yang menargetkan grup.
- Paling banyak Anda dapat menargetkan satu grup sumber daya untuk asosiasi. Grup beberapa atau bertingkat tidak didukung.
- Setelah Anda membuat asosiasi, State Manager secara berkala memperbarui asosiasi dengan informasi tentang sumber daya di Grup Sumber Daya. Jika Anda menambahkan sumber daya baru ke Resource Group, jadwal saat sistem menerapkan asosiasi untuk sumber daya baru tergantung pada beberapa faktor. Anda dapat menentukan status asosiasi di State Manager halaman konsol Systems Manager.

> **Warning**
>
> Pengguna, grup, atau peran AWS Identity and Access Management (IAM) dengan izin untuk membuat asosiasi yang menargetkan grup sumber daya dari instans Amazon EC2 secara otomatis memiliki kontrol tingkat akar dari semua instans dalam grup. Hanya administrator tepercaya harus diizinkan untuk membuat asosiasi.

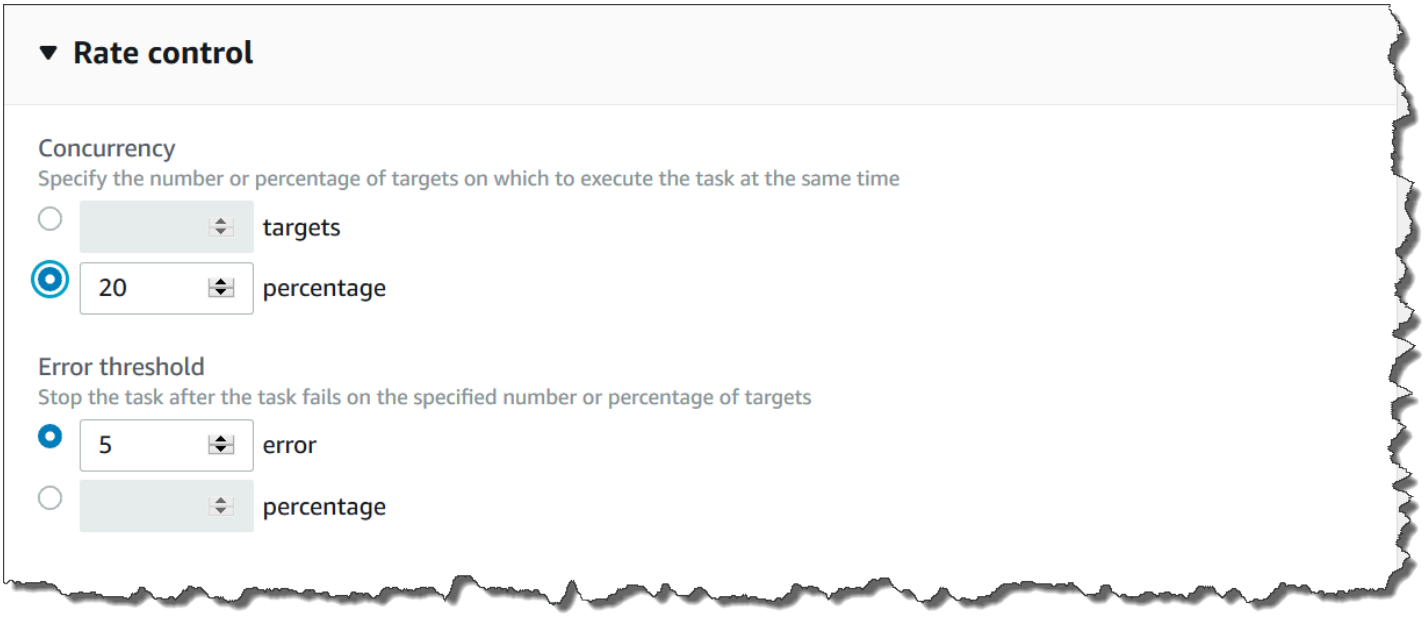
Untuk informasi lebih lanjut tentang Resource Groups, lihat Apa Itu AWS Resource Groups? di Panduan Pengguna AWS Resource Groups.

Pilih semua node

Gunakan opsi ini untuk menargetkan semua node di saat ini Akun AWS danWilayah AWS. Ketika Anda menjalankan permintaan, sistem menemukan dan mencoba untuk membuat asosiasi pada semua node di saat ini Akun AWS danWilayah AWS. Ketika sistem awalnya menciptakan asosiasi, ia menjalankan asosiasi. Setelah menjalankan awal ini, sistem menjalankan asosiasi yang sesuai dengan jadwal yang Anda tentukan. Jika Anda membuat node baru, sistem secara otomatis menerapkan asosiasi, menjalankannya segera, dan kemudian menjalankannya sesuai jadwal.

Kontrol rate

Anda dapat mengontrol eksekusi asosiasi pada node Anda dengan menentukan nilai konkurensi dan ambang kesalahan. Nilai konkurensi menentukan berapa banyak node dapat menjalankan asosiasi secara bersamaan. Ambang kesalahan menentukan berapa banyak eksekusi asosiasi yang dapat gagal sebelum Systems Manager mengirimkan perintah ke setiap node yang dikonfigurasi dengan asosiasi tersebut untuk berhenti menjalankan asosiasi. Perintah menghentikan asosiasi berjalan sampai eksekusi yang dijadwalkan berikutnya. Fitur konkurensi dan batas kesalahan secara kolektif disebut pengendalian rate.



Bersamaan

Concurrency membantu membatasi dampak pada node Anda dengan memungkinkan Anda menentukan bahwa hanya sejumlah node tertentu yang dapat memproses asosiasi pada satu waktu. Anda dapat menentukan jumlah absolut node, misalnya 20, atau persentase dari set target node, misalnya 10%.

State Managerkonkurensi memiliki batasan dan batasan berikut:

- Jika Anda memilih untuk membuat asosiasi dengan menggunakan target, tetapi Anda tidak menentukan nilai konkurensi, maka State Manager secara otomatis menerapkan konkurensi maksimum 50 node.
- Jika node baru yang cocok dengan kriteria target online saat asosiasi yang menggunakan konkurensi sedang berjalan, maka node baru menjalankan asosiasi jika nilai konkurensi tidak terlampaui. Jika nilai konkurensi terlampaui, maka node diabaikan selama interval eksekusi asosiasi saat ini. Node menjalankan asosiasi selama interval terjadwal berikutnya sambil menyesuaikan dengan persyaratan konkurensi.
- Jika Anda memperbarui asosiasi yang menggunakan konkurensi, dan satu atau beberapa node memproses asosiasi itu saat diperbarui, maka node apa pun yang menjalankan asosiasi diizinkan untuk diselesaikan. Asosiasi yang belum dimulai dihentikan. Setelah menjalankan asosiasi selesai, semua node target segera menjalankan asosiasi lagi karena telah diperbarui. Ketika asosiasi berjalan lagi, nilai konkurensi diterapkan.

Batas kesalahan

Ambang kesalahan menentukan berapa banyak eksekusi asosiasi yang diizinkan gagal sebelum Systems Manager mengirimkan perintah ke setiap node yang dikonfigurasi dengan asosiasi tersebut. Perintah menghentikan asosiasi berjalan sampai eksekusi yang dijadwalkan berikutnya. Anda dapat menentukan jumlah kesalahan absolut, misalnya 10, atau persentase target yang ditetapkan, misalnya 10%.

Jika Anda menentukan jumlah absolut dari tiga kesalahan, misalnya, State Manager mengirimkan perintah berhenti ketika kesalahan keempat dikembalikan. Jika Anda menentukan 0, maka State Manager kirimkan perintah stop setelah hasil kesalahan pertama dikembalikan.

Jika Anda menentukan ambang kesalahan 10% untuk 50 asosiasi, maka State Manager kirimkan perintah berhenti ketika kesalahan keenam dikembalikan. Asosiasi yang sudah berjalan ketika batas kesalahan tercapai diperbolehkan untuk menyelesaikan, tetapi beberapa asosiasi ini mungkin gagal. Untuk memastikan bahwa tidak ada lebih banyak kesalahan daripada jumlah yang ditentukan untuk batas kesalahan, atur nilai Konkurensi ke 1 sehingga asosiasi melanjutkan satu per satu.

State Managerambang kesalahan memiliki batasan dan batasan berikut:

- Batas kesalahan diterapkan untuk interval saat ini.
- Informasi tentang setiap kesalahan, termasuk detail tingkat langkah, dicatat dalam riwayat asosiasi.

- Jika Anda memilih untuk membuat asosiasi dengan menggunakan target, tetapi Anda tidak menentukan ambang kesalahan, maka State Manager secara otomatis memberlakukan ambang kegagalan 100%.

Membuat asosiasi

State Manager, kemampuanAWS Systems Manager, membantu Anda menjaga AWS sumber daya Anda dalam keadaan yang Anda tentukan dan mengurangi penyimpangan konfigurasi. Untuk melakukan ini, State Manager gunakan asosiasi. Asosiasi adalah konfigurasi yang Anda tetapkan ke AWS sumber daya Anda. Konfigurasi mendefinisikan status yang ingin Anda pertahankan pada sumber daya Anda. Misalnya, asosiasi dapat menentukan bahwa perangkat lunak antivirus harus diinstal dan berjalan pada node terkelola, atau port tertentu harus ditutup.

Asosiasi menentukan jadwal kapan harus menerapkan konfigurasi dan target untuk asosiasi. Misalnya, asosiasi untuk perangkat lunak antivirus dapat berjalan sekali sehari pada semua node yang dikelola dalam fileAkun AWS. Jika perangkat lunak tidak diinstal pada node, maka asosiasi dapat menginstruksikan State Manager untuk menginstalnya. Jika perangkat lunak diinstal, tetapi layanan tidak berjalan, maka asosiasi dapat menginstruksikan State Manager untuk memulai layanan.

> **Note**
>
> Anda dapat menetapkan tag ke asosiasi saat Anda membuatnya dengan menggunakan alat baris perintah seperti AWS CLI orAWS Tools for PowerShell. Menambahkan tag ke asosiasi menggunakan konsol Systems Manager tidak didukung. Untuk informasi selengkapnya tentang tanda, lihat Penandaan sumber daya Systems Manager.

Prosedur berikut menjelaskan cara membuat asosiasi yang menggunakan salah satu `Command` atau `Policy` dokumen untuk menargetkan node terkelola. Untuk informasi tentang membuat asosiasi yang menggunakan runbook Otomasi untuk menargetkan node atau jenis AWS sumber daya lainnya, lihatMenjadwalkan otomatisasi denganState Manager asosiasi.

Target asosiasi dan kontrol tarif

Asosiasi menentukan node terkelola, atau target, yang harus menerima asosiasi. State Managermenyertakan beberapa fitur untuk membantu Anda menargetkan node terkelola dan

mengontrol bagaimana asosiasi diterapkan ke target tersebut. Untuk informasi selengkapnya tentang pengendalian target dan rate, lihat Tentang target dan kontrol tingkat dalam State Manager asosiasi.

Menjalankan asosiasi

Secara default, State Manager jalankan asosiasi segera setelah Anda membuatnya, dan kemudian sesuai dengan jadwal yang telah Anda tentukan.

Sistem ini juga menjalankan asosiasi sesuai dengan aturan berikut:

- State Managermencoba untuk menjalankan asosiasi pada semua node yang ditentukan atau ditargetkan selama interval.
- Jika asosiasi tidak berjalan selama interval (karena, misalnya, nilai konkurensi membatasi jumlah node yang dapat memproses asosiasi pada satu waktu), maka State Manager mencoba untuk menjalankan asosiasi selama interval berikutnya.
- State Managermenjalankan asosiasi setelah perubahan pada konfigurasi asosiasi, node target, dokumen, atau parameter. Untuk informasi selengkapnya, lihat Kapan asosiasi diterapkan pada sumber daya?
- State Managermencatat riwayat untuk semua interval yang dilewati. Anda dapat melihat riwayat di tab Riwayat Eksekusi.

Asosiasi penjadwalan

Anda dapat menjadwalkan asosiasi untuk dijalankan pada interval dasar seperti setiap 10 jam, atau Anda dapat membuat jadwal yang lebih maju menggunakan ekspresi cron dan rate kustom. Anda juga dapat mencegah asosiasi berjalan saat pertama kali membuatnya.

Menggunakan ekspresi cron dan rate untuk menjadwalkan proses asosiasi

Selain ekspresi cron dan rate standar, State Manager juga mendukung ekspresi cron yang mencakup hari dalam seminggu dan tanda angka (#) untuk menunjuk hari ke-n dalam sebulan untuk menjalankan asosiasi. Berikut adalah contoh yang menjalankan jadwal cron pada hari Selasa ketiga setiap bulan pukul 23:30 UTC:

```
cron(30 23 ? * TUE#3 *)
```

Berikut adalah contoh yang berjalan pada hari Kamis kedua setiap bulan pada tengah malam UTC:

```
cron(0 0 ? * THU#2 *)
```

State Managerjuga mendukung tanda (L) untuk menunjukkan hari X terakhir setiap bulan. Berikut adalah contoh yang menjalankan jadwal cron pada hari Selasa terakhir setiap bulan pada tengah malam UTC:

```
cron(0 0 ? * 3L *)
```

Untuk mengontrol lebih lanjut saat asosiasi berjalan, misalnya jika Anda ingin menjalankan asosiasi dua hari setelah patch Selasa, Anda dapat menentukan offset. Offset mendefinisikan berapa hari untuk menunggu setelah hari yang dijadwalkan untuk menjalankan asosiasi. Misalnya, jika Anda menentukan jadwal cron`cron(0 0 ? * THU#2 *)`, Anda dapat menentukan angka 3 di bidang Offset Jadwal untuk menjalankan asosiasi setiap hari Minggu setelah Kamis kedua setiap bulan.

> **Note**
>
> Untuk menggunakan offset, Anda harus memilih Terapkan asosiasi hanya pada interval Cron yang ditentukan berikutnya di konsol atau tentukan `ApplyOnlyAtCronInterval` parameter dari baris perintah. Ketika salah satu dari opsi ini diaktifkan, State Manager tidak menjalankan asosiasi segera setelah Anda membuatnya.

Untuk informasi selengkapnya tentang ekspresi cron dan rate, lihat Referensi: Ekspresi cron dan rate untuk Systems Manager.

Membuat asosiasi (konsol)

Prosedur berikut menjelaskan cara menggunakan konsol Systems Manager untuk membuat State Manager asosiasi.

> **Warning**
>
> Saat membuat asosiasi, Anda dapat memilih grup AWS sumber daya dari node terkelola sebagai target untuk asosiasi. Jika pengguna, grup, atau peran AWS Identity and Access Management (IAM) memiliki izin untuk membuat asosiasi yang menargetkan grup sumber daya dari node terkelola, maka pengguna, grup, atau peran tersebut secara otomatis memiliki kontrol tingkat root dari semua node dalam grup. Izinkan hanya administrator tepercaya untuk membuat asosiasi.

Untuk membuat State Manager asosiasi

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih State Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih State Manager.
3. Pilih Buat asosiasi.
4. Di bidang Nama, tentukan nama.
5. Di daftar Dokumen, pilih opsi di samping nama dokumen. Perhatikan tipe dokumen. Prosedur ini berlaku untuk dokumen `Command` dan `Policy`. Untuk informasi tentang membuat asosiasi yang menggunakan runbook Otomatisasi, lihat Menjadwalkan otomatisasi denganState Manager asosiasi.

   > **Important**
   >
   > State Managertidak mendukung asosiasi berjalan yang menggunakan versi baru dokumen jika dokumen tersebut dibagikan dari akun lain. State Manager selalu menjalankan `default` versi dokumen jika dibagikan dari akun lain, meskipun konsol Systems Manager menunjukkan bahwa versi baru telah diproses. Jika Anda ingin menjalankan asosiasi menggunakan versi baru dokumen yang dibagikan dari akun lain, Anda harus menyetel versi dokumen ke`default`.
6. Untuk Parameter, tentukan parameter input yang diperlukan.
7. (Opsional) Pilih CloudWatch alarm untuk diterapkan ke asosiasi Anda untuk pemantauan.

   > **Note**
   >
   > Perhatikan informasi berikut tentang langkah ini.
   >
   > - Daftar alarm menampilkan maksimal 100 alarm. Jika Anda tidak melihat alarm dalam daftar, gunakan tombol AWS Command Line Interface untuk membuat asosiasi. Untuk informasi selengkapnya, lihat Membuat asosiasi (baris perintah).

- Untuk melampirkan CloudWatch alarm ke perintah Anda, kepala sekolah IAM yang membuat asosiasi harus memiliki izin untuk `iam:createServiceLinkedRole` tindakan tersebut. Untuk informasi selengkapnya tentang CloudWatch alarm, lihat Menggunakan CloudWatch alarm Amazon.
- Jika alarm Anda aktif, pemanggilan atau otomatisasi perintah yang tertunda tidak berjalan.

8. Untuk Target, pilih satu opsi. Untuk informasi tentang menggunakan target, lihat Tentang target dan kontrol tingkat dalam State Manager asosiasi.
9. Di bagian Tentukan jadwal, pilih Sesuai Jadwal atau Tidak ada jadwal. Jika Anda memilih Sesuai Jadwal, gunakan tombol yang disediakan untuk membuat cron atau jadwal rate untuk asosiasi.

   Jika Anda tidak ingin asosiasi segera berjalan setelah Anda membuatnya, pilih Terapkan asosiasi hanya pada interval Cron yang ditentukan berikutnya.
10. (Opsional) Di bidang Offset jadwal, tentukan angka antara 1 dan 6.
11. Di bagian Opsi lanjutan gunakan Keparahan Kepatuhan untuk memilih tingkat keparahan untuk asosiasi dan gunakan Ubah Kalender untuk memilih perubahan kalender untuk asosiasi.

    Pelaporan kepatuhan menunjukkan apakah status asosiasi sesuai atau tidak, bersama dengan tingkat keparahan yang Anda tunjukkan di sini. Untuk informasi selengkapnya, lihat Tentang kepatuhan State Manager asosiasi.

    Kalender perubahan menentukan kapan asosiasi berjalan. Jika kalender ditutup, asosiasi tidak akan diterapkan. Jika kalendar dibuka, asosiasi berjalan dengan sesuai. Untuk informasi selengkapnya, lihat AWS Systems Manager Change Calendar.
12. Di bagian Rate control, pilih opsi untuk mengontrol bagaimana asosiasi berjalan pada beberapa node. Untuk informasi lebih lanjut tentang menggunakan kontrol rate, lihat Tentang target dan kontrol tingkat dalam State Manager asosiasi.

    Di bagian Konkurensi, pilih satu opsi:

    - Pilih target untuk memasukkan jumlah absolut dari target yang dapat menjalankan asosiasi secara bersamaan.
    - Pilih persentase untuk memasukkan persentase dari kumpulan target yang dapat menjalankan asosiasi secara bersamaan.

Di bagian Batas kesalahan, pilih opsi:

- Pilih kesalahan untuk memasukkan jumlah absolut kesalahan yang diizinkan sebelum State Manager berhenti menjalankan asosiasi pada target tambahan.
- Pilih persentase untuk memasukkan persentase kesalahan yang diizinkan sebelum State Manager berhenti menjalankan asosiasi pada target tambahan.

13. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Aktifkan output penulisan ke S3. Masukkan nama bucket dan prefiks (folder) di dalam kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin dari profil instance yang ditetapkan ke node terkelola, bukan izin pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbedaAkun AWS, verifikasi bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

Berikut ini adalah izin minimal yang diperlukan untuk mengaktifkan output Amazon S3 untuk asosiasi. Anda dapat membatasi akses lebih lanjut dengan melampirkan kebijakan IAM ke pengguna atau peran dalam akun. Minimal, profil instans Amazon EC2 harus memiliki IAM role dengan kebijakan yang dikelola `AmazonSSMManagedInstanceCore` dan kebijakan yang sejalan berikut.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:PutObjectAcl"
            ],
            "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
```

```
        }
    ]
}
```

Untuk izin minimal, bucket Amazon S3 yang Anda ekspor harus memiliki pengaturan default yang ditentukan oleh konsol Amazon S3. Untuk informasi selengkapnya tentang pembuatan bucket Amazon S3, lihat Membuat bucket dalam Panduan Pengguna Amazon S3.

> **Note**
>
> Operasi API yang diprakarsai oleh dokumen SSM selama proses asosiasi tidak masuk. AWS CloudTrail

14. Pilih Buat Asosiasi.

> **Note**
>
> Jika Anda menghapus asosiasi yang Anda buat, asosiasi tidak lagi berjalan pada setiap target dari asosiasi tersebut.

Membuat asosiasi (baris perintah)

Prosedur berikut menjelaskan cara menggunakan AWS CLI (di Linux atau Windows) atau Alat PowerShell untuk membuat State Manager asosiasi. Bagian ini mencakup beberapa contoh yang menunjukkan bagaimana menggunakan target dan kontrol rate. Target dan kontrol tingkat memungkinkan Anda menetapkan asosiasi ke lusinan atau ratusan node sambil mengontrol eksekusi asosiasi tersebut. Untuk informasi selengkapnya tentang pengendalian target dan rate, lihat Tentang target dan kontrol tingkat dalam State Manager asosiasi.

Sebelum Anda memulai

`targets`Parameter adalah array kriteria pencarian yang menargetkan node menggunakan `Value` kombinasi`Key`, yang Anda tentukan. Jika Anda berencana untuk membuat asosiasi pada lusinan atau ratusan node dengan menggunakan `targets` parameter, tinjau opsi penargetan berikut sebelum memulai prosedur.

Targetkan node tertentu dengan menentukan ID

```
--targets Key=InstanceIds,Values=instance-id-1,instance-id-2,instance-id-3
```

```
--targets
 Key=InstanceIds,Values=i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE,i-07782c72faEXAMPLE
```

Instance target dengan menggunakan tag

```
--targets Key=tag:tag-key,Values=tag-value-1,tag-value-2,tag-value-3
```

```
--targets Key=tag:Environment,Values=Development,Test,Pre-production
```

Target node dengan menggunakan AWS Resource Groups

```
--targets Key=resource-groups:Name,Values=resource-group-name
```

```
--targets Key=resource-groups:Name,Values=WindowsInstancesGroup
```

Menargetkan semua instans di Akun AWS dan Wilayah AWS saat ini

```
--targets Key=InstanceIds,Values=*
```

Note

Perhatikan informasi berikut.

- State Managertidak mendukung asosiasi berjalan yang menggunakan versi baru dokumen jika dokumen tersebut dibagikan dari akun lain. State Manager selalu menjalankan `default` versi dokumen jika dibagikan dari akun lain, meskipun konsol Systems Manager menunjukkan bahwa versi baru telah diproses. Jika Anda ingin menjalankan asosiasi menggunakan versi baru dokumen yang dibagikan dari akun lain, Anda harus menyetel versi dokumen ke`default`.
- Anda dapat menentukan maksimal lima kunci tag dengan menggunakanAWS CLI. Jika Anda menggunakanAWS CLI, semua kunci tag yang ditentukan dalam `create-association` perintah harus saat ini ditetapkan ke node. Jika tidak, State Manager gagal menargetkan node untuk asosiasi. Untuk informasi tentang menetapkan tag ke node Anda, lihatPenandaan sumber daya Systems Manager.

- Saat Anda membuat asosiasi, Anda menentukan kapan jadwal berjalan. Tentukan jadwal dengan menggunakan ekspresi cron atau rate. Untuk informasi selengkapnya tentang ekspresi cron dan rate, lihat Ekspresi cron dan rate untuk associate.

Untuk membuat asosiasi

1. Instal dan konfigurasikan AWS CLI atau AWS Tools for PowerShell, jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI dan Menginstal AWS Tools for PowerShell.
2. Gunakan format berikut untuk membuat perintah yang membuat State Manager asosiasi. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   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 "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 "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"
```

Contoh berikut membuat asosiasi pada node yang ditandai dengan`"Environment,Linux"`. Asosiasi menggunakan `AWS-UpdateSSMAgent` dokumen untuk memperbarui SSM Agent pada node yang ditargetkan pada 2:00 UTC setiap Minggu pagi. Asosiasi ini berjalan secara bersamaan pada 10 node maksimum pada waktu tertentu. Juga, asosiasi ini berhenti berjalan pada lebih banyak node untuk interval eksekusi tertentu jika jumlah kesalahan melebihi 5. Untuk pelaporan kepatuhan, asosiasi ini ditetapkan tingkat keparahan Medium.

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 "cron(0 2 ? * SUN *)" \
  --max-errors "5" \
  --max-concurrency "10"
```

Windows

```
aws ssm create-association ^
  --association-name Update_SSM_Agent_Linux ^
  --targets Key=tag:Environment,Values=Linux ^
  --name AWS-UpdateSSMAgent  ^
  --compliance-severity "MEDIUM" ^
  --schedule "cron(0 2 ? * SUN *)" ^
  --max-errors "5" ^
  --max-concurrency "10"
```

PowerShell

```
New-SSMAssociation `
  -AssociationName Update_SSM_Agent_Linux `
  -Name AWS-UpdateSSMAgent `
  -Target @{
      "Key"="tag:Environment"
      "Values"="Linux"
    } `
  -ComplianceSeverity MEDIUM `
  -ScheduleExpression "cron(0 2 ? * SUN *)" `
  -MaxConcurrency 10 `
  -MaxError 5
```

Contoh berikut menargetkan ID node dengan menentukan nilai wildcard (*). Hal ini memungkinkan Systems Manager untuk membuat asosiasi pada semua node di saat ini Akun AWS danWilayah AWS. Asosiasi ini berjalan secara bersamaan pada 10 node maksimum

pada waktu tertentu. Juga, asosiasi ini berhenti berjalan pada lebih banyak node untuk interval eksekusi tertentu jika jumlah kesalahan melebihi 5. Untuk pelaporan kepatuhan, asosiasi ini ditetapkan tingkat keparahan Medium. Asosiasi ini menggunakan offset jadwal, yang berarti berjalan dua hari setelah jadwal cron yang ditentukan. Ini juga mencakup `ApplyOnlyAtCronInterval` parameter, yang diperlukan untuk menggunakan offset jadwal dan yang berarti asosiasi tidak akan berjalan segera setelah dibuat.

Linux & macOS

```
aws ssm create-association \
  --association-name Update_SSM_Agent_Linux \
  --name "AWS-UpdateSSMAgent" \
  --targets "Key=instanceids,Values=*" \
  --compliance-severity "MEDIUM" \
  --schedule "cron(0 2 ? * SUN#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 "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
```

Contoh berikut membuat asosiasi pada node di Resource Groups. Grup ini diberi nama "HR-Department". Asosiasi menggunakan `AWS-UpdateSSMAgent` dokumen untuk memperbarui SSM Agent node yang ditargetkan pada pukul 2:00 UTC setiap Minggu pagi. Asosiasi ini berjalan secara bersamaan pada 10 node maksimum pada waktu tertentu. Juga, asosiasi ini berhenti berjalan pada lebih banyak node untuk interval eksekusi tertentu jika jumlah kesalahan melebihi 5. Untuk pelaporan kepatuhan, asosiasi ini ditetapkan tingkat keparahan Medium. Asosiasi ini berjalan pada jadwal cron yang ditentukan. Ini tidak segera berjalan setelah asosiasi dibuat.

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 "cron(0 2 ? * SUN *)" \
  --max-errors "5" \
  --max-concurrency "10" \
  --apply-only-at-cron-interval
```

Windows

```
aws ssm create-association ^
  --association-name Update_SSM_Agent_Linux ^
  --targets Key=resource-groups:Name,Values=HR-Department ^
  --name AWS-UpdateSSMAgent  ^
  --compliance-severity "MEDIUM" ^
  --schedule "cron(0 2 ? * SUN *)" ^
  --max-errors "5" ^
```

```
  --max-concurrency "10" ^
  --apply-only-at-cron-interval
```

PowerShell

```
New-SSMAssociation `
  -AssociationName Update_SSM_Agent_Linux `
  -Name AWS-UpdateSSMAgent `
  -Target @{
      "Key"="resource-groups:Name"
      "Values"="HR-Department"
    } `
  -ScheduleExpression "cron(0 2 ? * SUN *)" `
  -MaxConcurrency 10 `
  -MaxError 5 `
  -ComplianceSeverity MEDIUM `
  -ApplyOnlyAtCronInterval
```

Contoh berikut membuat asosiasi yang berjalan pada node ditandai dengan ID node tertentu. Asosiasi menggunakan SSM Agent dokumen untuk memperbarui SSM Agent pada node yang ditargetkan sekali ketika kalender perubahan terbuka. Asosiasi memeriksa status kalender saat dijalankan. Jika kalender ditutup pada waktu peluncuran dan asosiasi hanya dijalankan sekali, asosiasi tidak akan berjalan lagi karena jendela berjalan asosiasi telah berlalu. Jika kalendar dibuka, asosiasi berjalan dengan sesuai.

> **Note**
>
> Jika Anda menambahkan node baru ke tag atau grup sumber daya yang ditindaklanjuti asosiasi saat kalender perubahan ditutup, asosiasi akan diterapkan ke node tersebut setelah kalender perubahan terbuka.

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 "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 "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)"
```

Contoh berikut membuat asosiasi yang berjalan pada node ditandai dengan ID node tertentu. Asosiasi menggunakan SSM Agent dokumen untuk memperbarui node yang ditargetkan SSM Agent pada node yang ditargetkan pada pukul 2:00 pagi setiap hari Minggu. Asosiasi ini hanya berjalan pada jadwal cron yang ditentukan saat kalender perubahan terbuka. Ketika asosiasi dibuat, asosisasi memeriksa status kalender. Jika kalender ditutup, asosiasi tidak akan diterapkan. Ketika interval untuk menerapkan asosiasi dimulai pada 2:00 AM pada hari Minggu, asosiasi memeriksa untuk melihat apakah kalender terbuka. Jika kalendar dibuka, asosiasi berjalan dengan sesuai.

> **Note**
>
> Jika Anda menambahkan node baru ke tag atau grup sumber daya yang ditindaklanjuti asosiasi saat kalender perubahan ditutup, asosiasi akan diterapkan ke node tersebut setelah kalender perubahan terbuka.

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 "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 "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

Jika Anda menghapus asosiasi yang Anda buat, asosiasi tidak lagi berjalan pada setiap target dari asosiasi tersebut. Juga, jika Anda menentukan parameter `apply-only-at-cron-interval`, Anda dapat mengatur ulang opsi ini. Untuk melakukannya, tentukan

parameter `no-apply-only-at-cron-interval` saat Anda memperbarui asosiasi dari baris perintah. Parameter ini memaksa asosiasi untuk berjalan segera setelah memperbarui asosiasi dan sesuai dengan interval yang ditentukan.

Mengedit dan membuat versi baru asosiasi

Anda dapat mengeditState Managerasosiasi untuk menentukan nama baru, jadwal, tingkat keparahan, atau target. Anda juga dapat memilih untuk menulis output perintah ke bucket Amazon Simple Storage Service (Amazon S3). Setelah Anda mengedit asosiasi,State Managermembuat versi baru. Anda dapat melihat versi yang berbeda setelah mengedit, seperti yang dijelaskan dalam prosedur berikut.

Prosedur berikut menjelaskan cara mengedit dan membuat versi baru asosiasi menggunakan konsol Manajer Sistem,AWS Command Line Interface(AWS CLI), danAWS Tools for PowerShell(Alat untuk PowerShell).

Important

State Managertidak mendukung asosiasi berjalan yang menggunakan versi baru dokumen jika dokumen tersebut dibagikan dari akun lain.State Managerselalu menjalankan`default`versi dokumen jika dibagikan dari akun lain, meskipun konsol Manajer Sistem menunjukkan bahwa versi baru telah diproses. Jika Anda ingin menjalankan asosiasi menggunakan versi baru dokumen yang dibagikan dari akun lain, Anda harus menyetel versi dokumen ke`default`.

Mengedit asosiasi (konsol)

Prosedur berikut menjelaskan cara menggunakan konsol Systems Manager untuk mengedit dan membuat versi baru asosiasi.

Note

Prosedur ini mengharuskan Anda memiliki akses tulis ke bucket Amazon S3 yang ada. Jika Anda belum pernah menggunakan Amazon S3 sebelumnya, perhatikan bahwa Anda akan dikenakan biaya untuk menggunakan Amazon S3. Untuk informasi tentang cara membuat bucket, lihat Buat Bucket.

Untuk mengeditState Managerasosiasi

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih State Manager.

   -atau-

   JikaAWS Systems Managerhalaman rumah terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih State Manager.
3. Pilih asosiasi yang Anda buat di Membuat asosiasi (baris perintah) lalu pilih Edit.
4. Di bidang Nama, masukkan nama baru.
5. Di bagian Tentukan jadwal, pilih opsi baru.
6. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Aktifkan output penulisan ke S3. Masukkan nama bucket dan prefiks (folder) di dalam kotak.

   > **Note**
   >
   > Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin dari profil instance yang ditetapkan ke node terkelola, bukan izin pengguna IAM yang melakukan tugas ini. Untuk informasi lebih lanjut, lihatKonfigurasikan izin instans untuk Manajer SistemanatauBuat peran layanan IAM untuk lingkungan hybrid. Selain itu, jika bucket S3 yang ditentukan berbedaAkun AWS, verifikasi bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.
7. Pilih Edit asosiasi. Konfigurasikan asosiasi untuk memenuhi kebutuhan Anda saat ini.
8. Di halaman Asosiasi, pilih nama asosiasi yang Anda edit, lalu pilih tab Versi. Sistem mencantumkan setiap versi asosiasi yang Anda buat dan edit.
9. Buka konsol Amazon S3 di https://console.aws.amazon.com/s3/.
10. Pilih nama bucket Amazon S3 yang Anda tentukan untuk menyimpan output perintah, lalu pilih folder yang diberi nama dengan ID node yang menjalankan asosiasi. (Jika anda memilih untuk menyimpan output dalam folder dalam bucket, buka terlebih dahulu.)
11. Menelusuri beberapa tingkat, melalui folder `awsrunPowerShell`, ke file `stdout`.
12. Pilih Buka atau Unduh untuk melihat nama host.

Mengedit asosiasi (baris perintah)

Prosedur berikut menjelaskan cara menggunakan AWS CLI (di Linux atau Windows) atau AWS Tools for PowerShell untuk mengedit dan membuat versi baru asosiasi.

Untuk mengeditState Managerasosiasi

1. Instal dan konfigurasikan AWS CLI atau AWS Tools for PowerShell, jika Anda belum melakukannya.

   Untuk informasi, lihatMenginstal atau memperbarui versi terbaruAWS CLIdanInstalasiAWS Tools for PowerShell.

2. Gunakan format berikut untuk membuat perintah untuk mengedit dan membuat versi baru dari yang sudah adaState Managerasosiasi. Ganti masing-masing*contoh placeholder sumber daya*dengan informasi Anda sendiri.

   > **Important**
   >
   > Saat Anda meneleponUpdateAssociation, sistem menjatuhkan semua parameter opsional dari permintaan dan menimpa asosiasi dengan nilai nol untuk parameter tersebut. Ini dengan desain. Anda harus menentukan semua parameter opsional dalam panggilan, bahkan jika Anda tidak mengubah parameter. Ini termasukNameparameter. Sebelum memanggil tindakan API ini, kami sarankan Anda memanggilDescribeAssociationOperasi API dan catat semua parameter opsional yang diperlukan untukUpdateAssociationpanggilan.

   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 "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 "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
```

Contoh berikut memperbarui asosiasi yang ada untuk mengubah nama ke `TestHostnameAssociation2`. Versi asosiasi baru berjalan setiap jam dan menulis output perintah ke bucket Amazon S3 yang ditentukan.

Linux & macOS

```
aws ssm update-association \
  --association-id 8dfe3659-4309-493a-8755-01234EXAMPLE \
  --association-name TestHostnameAssociation2 \
  --parameters commands="echo Association" \
  --output-location S3Location='{OutputS3Region=us-
east-1,OutputS3BucketName=DOC-EXAMPLE-BUCKET,OutputS3KeyPrefix=logs}' \
  --schedule-expression "cron(0 */1 * * ? *)"
```

Windows

```
aws ssm update-association ^
  --association-id 8dfe3659-4309-493a-8755-01234EXAMPLE ^
  --association-name TestHostnameAssociation2 ^
  --parameters commands="echo Association" ^
  --output-location S3Location='{OutputS3Region=us-
east-1,OutputS3BucketName=DOC-EXAMPLE-BUCKET,OutputS3KeyPrefix=logs}' ^
  --schedule-expression "cron(0 */1 * * ? *)"
```

PowerShell

```
Update-SSMAssociation `
  -AssociationId b85ccafe-9f02-4812-9b81-01234EXAMPLE `
  -AssociationName TestHostnameAssociation2 `
  -Parameter @{"commands"="echo Association"} `
  -S3Location_OutputS3BucketName DOC-EXAMPLE-BUCKET `
  -S3Location_OutputS3KeyPrefix logs `
  -S3Location_OutputS3Region us-east-1 `
  -ScheduleExpression "cron(0 */1 * * ? *)"
```

Contoh berikut memperbarui asosiasi yang ada untuk mengubah nama ke `CalendarAssociation`. Asosiasi baru berjalan ketika kalender terbuka dan menulis output perintah ke bucket Amazon S3 yang ditentukan.

Linux & macOS

```
aws ssm update-association \
  --association-id 8dfe3659-4309-493a-8755-01234EXAMPLE \
  --association-name CalendarAssociation \
  --parameters commands="echo Association" \
  --output-location S3Location='{OutputS3Region=us-
east-1,OutputS3BucketName=DOC-EXAMPLE-BUCKET,OutputS3KeyPrefix=logs}' \
  --calendar-names "arn:aws:ssm:us-east-1:123456789012:document/testCalendar2"
```

Windows

```
aws ssm update-association ^
  --association-id 8dfe3659-4309-493a-8755-01234EXAMPLE ^
  --association-name CalendarAssociation ^
  --parameters commands="echo Association" ^
  --output-location S3Location='{OutputS3Region=us-
east-1,OutputS3BucketName=DOC-EXAMPLE-BUCKET,OutputS3KeyPrefix=logs}' ^
  --calendar-names "arn:aws:ssm:us-east-1:123456789012:document/testCalendar2"
```

PowerShell

```
Update-SSMAssociation `
  -AssociationId b85ccafe-9f02-4812-9b81-01234EXAMPLE `
  -AssociationName CalendarAssociation `
  -AssociationName OneTimeAssociation `
  -Parameter @{"commands"="echo Association"} `
  -S3Location_OutputS3BucketName DOC-EXAMPLE-BUCKET `
  -CalendarNames "arn:aws:ssm:us-east-1:123456789012:document/testCalendar2"
```

Contoh berikut memperbarui asosiasi yang ada untuk mengubah nama ke `MultiCalendarAssociation`. Asosiasi baru berjalan ketika kalender terbuka dan menulis output perintah ke bucket Amazon S3 yang ditentukan.

Linux & macOS

```
aws ssm update-association \
  --association-id 8dfe3659-4309-493a-8755-01234EXAMPLE \
  --association-name MultiCalendarAssociation \
```

```
  --parameters commands="echo Association" \
  --output-location S3Location='{OutputS3Region=us-
east-1,OutputS3BucketName=DOC-EXAMPLE-BUCKET,OutputS3KeyPrefix=logs}' \
  --calendar-names "arn:aws:ssm:us-east-1:123456789012:document/testCalendar1"
 "arn:aws:ssm:us-east-2:123456789012:document/testCalendar2"
```

Windows

```
aws ssm update-association ^
  --association-id 8dfe3659-4309-493a-8755-01234EXAMPLE ^
  --association-name MultiCalendarAssociation ^
  --parameters commands="echo Association" ^
  --output-location S3Location='{OutputS3Region=us-
east-1,OutputS3BucketName=DOC-EXAMPLE-BUCKET,OutputS3KeyPrefix=logs}' ^
  --calendar-names "arn:aws:ssm:us-east-1:123456789012:document/testCalendar1"
 "arn:aws:ssm:us-east-2:123456789012:document/testCalendar2"
```

PowerShell

```
Update-SSMAssociation `
  -AssociationId b85ccafe-9f02-4812-9b81-01234EXAMPLE `
  -AssociationName MultiCalendarAssociation `
  -Parameter @{"commands"="echo Association"} `
  -S3Location_OutputS3BucketName DOC-EXAMPLE-BUCKET `
  -CalendarNames "arn:aws:ssm:us-east-1:123456789012:document/testCalendar1"
 "arn:aws:ssm:us-east-2:123456789012:document/testCalendar2"
```

3. Untuk melihat versi baru asosiasi, jalankan perintah berikut.

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 *
```

Sistem mengembalikan informasi seperti berikut.

Linux & macOS

```
{
    "AssociationDescription": {
        "ScheduleExpression": "cron(0 */1 * * ? *)",
        "OutputLocation": {
            "S3Location": {
                "OutputS3KeyPrefix": "logs",
                "OutputS3BucketName": "DOC-EXAMPLE-BUCKET",
                "OutputS3Region": "us-east-1"
            }
        },
        "Name": "AWS-RunPowerShellScript",
        "Parameters": {
            "commands": [
                "echo Association"
            ]
        },
        "LastExecutionDate": 1559316400.338,
        "Overview": {
            "Status": "Success",
            "DetailedStatus": "Success",
            "AssociationStatusAggregatedCount": {}
        },
        "AssociationId": "b85ccafe-9f02-4812-9b81-01234EXAMPLE",
        "DocumentVersion": "$DEFAULT",
        "LastSuccessfulExecutionDate": 1559316400.338,
        "LastUpdateAssociationDate": 1559316389.753,
        "Date": 1559314038.532,
        "AssociationVersion": "2",
        "AssociationName": "TestHostnameAssociation2",
        "Targets": [
            {
                "Values": [
```

```
                    "Windows"
                ],
                "Key": "tag:Environment"
            }
        ]
    }
}
```

Windows

```
{
    "AssociationDescription": {
        "ScheduleExpression": "cron(0 */1 * * ? *)",
        "OutputLocation": {
            "S3Location": {
                "OutputS3KeyPrefix": "logs",
                "OutputS3BucketName": "DOC-EXAMPLE-BUCKET",
                "OutputS3Region": "us-east-1"
            }
        },
        "Name": "AWS-RunPowerShellScript",
        "Parameters": {
            "commands": [
                "echo Association"
            ]
        },
        "LastExecutionDate": 1559316400.338,
        "Overview": {
            "Status": "Success",
            "DetailedStatus": "Success",
            "AssociationStatusAggregatedCount": {}
        },
        "AssociationId": "b85ccafe-9f02-4812-9b81-01234EXAMPLE",
        "DocumentVersion": "$DEFAULT",
        "LastSuccessfulExecutionDate": 1559316400.338,
        "LastUpdateAssociationDate": 1559316389.753,
        "Date": 1559314038.532,
        "AssociationVersion": "2",
        "AssociationName": "TestHostnameAssociation2",
        "Targets": [
            {
                "Values": [
                    "Windows"
```

```
                ],
                "Key": "tag:Environment"
            }
        ]
    }
}
```

PowerShell

```
AssociationId                : b85ccafe-9f02-4812-9b81-01234EXAMPLE
AssociationName              : TestHostnameAssociation2
AssociationVersion           : 2
AutomationTargetParameterName :
ComplianceSeverity           :
Date                         : 5/31/2019 2:47:18 PM
DocumentVersion              : $DEFAULT
InstanceId                   :
LastExecutionDate            : 5/31/2019 3:26:40 PM
LastSuccessfulExecutionDate  : 5/31/2019 3:26:40 PM
LastUpdateAssociationDate    : 5/31/2019 3:26:29 PM
MaxConcurrency               :
MaxErrors                    :
Name                         : AWS-RunPowerShellScript
OutputLocation               :
 Amazon.SimpleSystemsManagement.Model.InstanceAssociationOutputLocation
Overview                     :
 Amazon.SimpleSystemsManagement.Model.AssociationOverview
Parameters                   : {[commands,
 Amazon.Runtime.Internal.Util.AlwaysSendList`1[System.String]]}
ScheduleExpression           : cron(0 */1 * * ? *)
Status                       :
Targets                      : {tag:Environment}
```

Menghapus asosiasi

Prosedur berikut menjelaskan cara menghapus State Manager asosiasi dengan menggunakan AWS Systems Manager konsol.

Menghapus asosiasi

Gunakan prosedur berikut untuk menghapus asosiasi dengan menggunakan AWS Systems Manager konsol.

Untuk menghapus asosiasi

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih State Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih State Manager.
3. Pilih asosiasi dan kemudian pilih Hapus.

Menjalankan grup Auto Scaling dengan asosiasi

Praktik terbaik saat menggunakan asosiasi untuk menjalankan grup Auto Scaling adalah dengan menggunakan target tag. Tidak menggunakan tag dapat menyebabkan Anda mencapai batas asosiasi.

Jika semua node ditandai dengan kunci dan nilai yang sama, Anda hanya perlu satu asosiasi untuk menjalankan grup Auto Scaling Anda. Prosedur berikut menjelaskan cara membuat asosiasi semacam itu.

Untuk membuat asosiasi yang menjalankan grup Auto Scaling

1. Pastikan semua node dalam grup Auto Scaling ditandai dengan kunci dan nilai yang sama. Untuk petunjuk selengkapnya tentang menandai node, lihat Menandai grup dan instance Penskalaan Otomatis di Panduan Pengguna. AWS Auto Scaling
2. Membuat asosiasi dengan menggunakan prosedur di Bekerja dengan asosiasi di Systems Manager.

   Jika Anda bekerja di konsol, pilih Tentukan tag instans dalam bidang Target. Untuk Tag instans, masukkan kunci Tag dan nilai untuk grup Auto Scaling Anda.

   Jika Anda menggunakan AWS Command Line Interface (AWS CLI), tentukan `--targets Key=tag:tag-key,Values=tag-value` di mana kunci dan nilai cocok dengan apa yang Anda beri tag pada node Anda.

Melihat riwayat asosiasi

Anda dapat melihat semua eksekusi untuk ID asosiasi tertentu dengan menggunakan operasi DescribeAssociationExecutionsAPI. Gunakan operasi ini untuk melihat status, status terperinci, hasil, waktu eksekusi terakhir, dan informasi lebih lanjut untuk State Manager asosiasi. State Manageradalah kemampuanAWS Systems Manager. Operasi API ini juga menyertakan filter untuk membantu Anda menemukan asosiasi yang sesuai dengan kriteria yang Anda tentukan. Misalnya, Anda dapat menentukan tanggal dan waktu yang tepat, dan menggunakan filter GREATER_THAN untuk melihat eksekusi yang diproses setelah tanggal dan waktu yang ditentukan.

Jika, misalnya, eksekusi asosiasi gagal, Anda dapat menelusuri detail eksekusi tertentu dengan menggunakan operasi DescribeAssociationExecutionTargetsAPI. Operasi ini menunjukkan sumber daya, seperti ID node, tempat asosiasi berjalan dan berbagai status asosiasi. Anda kemudian dapat melihat sumber daya atau node mana yang gagal menjalankan asosiasi. Dengan ID sumber daya, Anda dapat melihat rincian eksekusi perintah untuk melihat langkah mana dalam perintah yang gagal.

Contoh di bagian ini juga mencakup informasi tentang cara menggunakan operasi StartAssociationsOnceAPI untuk menjalankan asosiasi sekali pada saat pembuatan. Anda dapat menggunakan operasi API ini ketika Anda menyelidiki eksekusi asosiasi yang gagal. Jika Anda melihat asosiasi yang gagal, Anda dapat membuat perubahan pada sumber daya, dan kemudian segera menjalankan asosiasi untuk melihat apakah perubahan pada sumber daya mengizinkan asosiasi berjalan dengan sukses.

> **Note**
>
> Operasi API yang diprakarsai oleh dokumen SSM selama proses asosiasi tidak masuk. AWS CloudTrail

Melihat riwayat asosiasi (konsol)

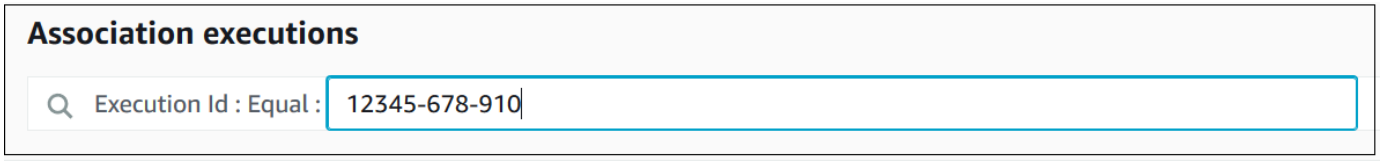
Gunakan prosedur berikut untuk melihat riwayat eksekusi untuk ID asosiasi tertentu lalu melihat rincian eksekusi untuk satu sumber daya atau lebih.

Untuk melihat riwayat eksekusi untuk ID asosiasi tertentu

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Pilih State Manager.

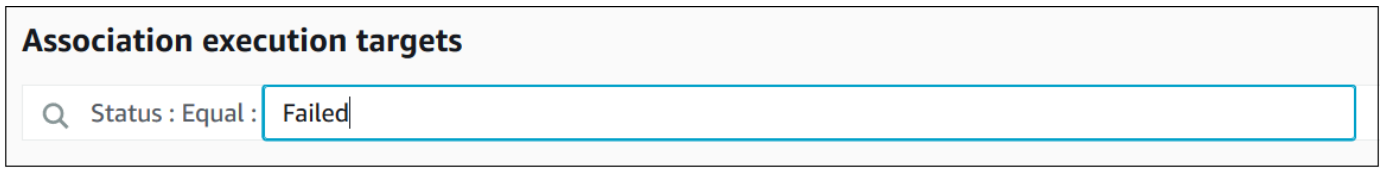
3. Di bidang Id asosiasi, pilih asosiasi yang ingin Anda lihat riwayatnya.
4. Pilih tombol Melihat detail.
5. Pilih tab Riwayat eksekusi.
6. Pilih asosiasi yang Anda inginkan untuk melihat detail eksekusi tingkat sumber daya. Misalnya, pilih asosiasi yang menunjukkan status Gagal. Anda kemudian dapat melihat detail eksekusi untuk node yang gagal menjalankan asosiasi.

   Gunakan filter kotak pencarian untuk menemukan eksekusi yang ingin Anda lihat rinciannya.

7. Pilih ID eksekusi. Halaman Target eksekusi asosiasi terbuka. Halaman ini menunjukkan semua sumber daya yang menjalankan asosiasi.
8. Pilih ID sumber daya untuk melihat informasi spesifik tentang sumber daya tersebut.

   Gunakan filter kotak pencarian untuk menemukan sumber daya yang ingin Anda lihat rinciannya.

9. Jika Anda sedang menyelidiki asosiasi yang gagal dijalankan, Anda dapat menggunakan tombol Terapkan asosiasi sekarang untuk menjalankan asosiasi sekali setelah pembuatan. Setelah Anda membuat perubahan pada sumber daya di mana asosiasi gagal untuk dijalankan, pilih tautan ID Asosiasi di breadcrumb navigasi.
10. Pilih tombol Terapkan asosiasi sekarang. Setelah eksekusi selesai, verifikasi bahwa eksekusi asosiasi telah berhasil.

Melihat riwayat asosiasi (baris perintah)

Prosedur berikut menjelaskan cara menggunakan AWS Command Line Interface (AWS CLI) (pada Linux atau Windows) atau AWS Tools for PowerShell untuk melihat riwayat eksekusi untuk ID asosiasi tertentu. Setelah ini, prosedur menjelaskan cara untuk melihat rincian eksekusi untuk satu sumber daya atau lebih.

Untuk melihat riwayat eksekusi untuk ID asosiasi tertentu

1. Instal dan konfigurasikan AWS CLI atau AWS Tools for PowerShell, jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI dan Menginstal AWS Tools for PowerShell.

2. Jalankan perintah berikut untuk menampilkan daftar eksekusi untuk ID asosiasi tertentu.

   Linux & macOS

   ```
   aws ssm describe-association-executions \
     --association-id ID \
     --filters Key=CreatedTime,Value="2018-04-10T19:15:38.372Z",Type=GREATER_THAN
   ```

   > **Note**
   >
   > Perintah ini menyertakan filter untuk membatasi hasil hanya untuk eksekusi yang terjadi setelah tanggal dan waktu tertentu. Jika Anda ingin melihat semua eksekusi untuk ID asosiasi tertentu, hapus parameter `--filters` dan nilai `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**
   >
   > Perintah ini menyertakan filter untuk membatasi hasil hanya untuk eksekusi yang terjadi setelah tanggal dan waktu tertentu. Jika Anda ingin melihat semua eksekusi untuk ID asosiasi tertentu, hapus parameter `--filters` dan nilai `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**
>
> Perintah ini menyertakan filter untuk membatasi hasil hanya untuk eksekusi yang terjadi setelah tanggal dan waktu tertentu. Jika Anda ingin melihat semua eksekusi untuk ID asosiasi tertentu, hapus parameter `-Filter` dan nilai `@{"Key"="CreatedTime";"Value"="2019-06-01T19:15:38.372Z";"Type"="GREAT`

Sistem mengembalikan informasi seperti berikut.

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
```

Anda dapat membatasi hasil dengan menggunakan satu filter atau lebih. Contoh berikut mengembalikan semua asosiasi yang dijalankan sebelum tanggal dan waktu tertentu.

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"}
```

Berikut mengembalikan semua asosiasi yang berhasil dijalankan setelah tanggal dan waktu tertentu.

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. Jalankan perintah berikut untuk menampilkan semua target di mana eksekusi spesifik berjalan.

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
```

Anda dapat membatasi hasil dengan menggunakan satu filter atau lebih. Contoh berikut mengembalikan informasi tentang semua target di mana asosiasi tertentu gagal untuk dijalankan.

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"
    }
```

Contoh berikut mengembalikan informasi tentang node terkelola tertentu di mana asosiasi gagal dijalankan.

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. Jika Anda menyelidiki asosiasi yang gagal dijalankan, Anda dapat menggunakan operasi StartAssociationsOnceAPI untuk menjalankan asosiasi segera dan hanya satu kali. Setelah Anda mengubah sumber daya di mana asosiasi gagal untuk dijalankan, jalankan perintah berikut untuk segera menjalankan asosiasi dan hanya satu kali.

   Linux & macOS

   ```
   aws ssm start-associations-once \
     --association-id ID
   ```

   Windows

   ```
   aws ssm start-associations-once ^
     --association-id ID
   ```

   PowerShell

   ```
   Start-SSMAssociationsOnce `
     -AssociationId ID
   ```

Bekerja dengan asosiasi menggunakan IAM

State Manager, kemampuanAWS Systems Manager, Menggunakansasaranuntuk memilih instans mana yang Anda konfigurasikan dengan asosiasi Anda. Awalnya, asosiasi dibuat dengan menentukan nama dokumen (`Name`) dan ID instans (`InstanceId`). Hal ini menciptakan asosiasi antara dokumen dan instans atau instans terkelola misalnya. Asosiasi digunakan untuk diidentifikasi oleh parameter ini. Parameter ini sekarang tidak lagi digunakan, tetapi mereka masih didukung. Sumber daya `instance` dan `managed-instance` ditambahkan sebagai sumber daya untuk tindakan dengan `Name` dan `InstanceId`.

Pemberlakuan kebijakan (IAM) AWS Identity and Access Management tergantung pada jenis sumber daya yang ditentukan. Sumber daya untukState Manageroperasi hanya diterapkan berdasarkan permintaan yang diterapkan.State Managertidak melakukan pemeriksaan mendalam untuk properti sumber daya di akun Anda. Permintaan hanya divalidasi terhadap sumber daya kebijakan jika parameter permintaan berisi sumber daya kebijakan tertentu. Misalnya, jika Anda menentukan

instans di blok sumber daya, kebijakan diberlakukan jika permintaan menggunakan parameter `InstanceId`. Parameter `Targets` untuk setiap sumber daya di akun tidak diperiksa untuk `InstanceId` itu.

Berikut ini adalah beberapa kasus dengan perilaku yang membingungkan:

- DescribeAssociation,DeleteAssociation, danUpdateAssociationmenggunakan`instance`,`managed-instance`, dan`document`sumber daya untuk menentukan cara usang mengacu pada asosiasi. Hal ini mencakup semua asosiasi dibuat dengan parameter `InstanceId` yang tidak lagi digunakan.
- CreateAssociation,CreateAssociationBatch, danUpdateAssociationmenggunakan`instance`dan`managed-instance`sumber daya untuk menentukan cara usang mengacu pada asosiasi. Hal ini mencakup semua asosiasi dibuat dengan parameter `InstanceId` yang tidak lagi digunakan. Jenis sumber daya `document` adalah bagian dari cara yang tidak lagi digunakan dalam merujuk ke asosiasi dan merupakan properti sebenarnya dari sebuah asosiasi. Ini berarti Anda dapat membangun kebijakan IAM dengan Izinkan atau Tolak izin untuk tindakan Buat dan Perbarui berdasarkan nama dokumen.

Untuk informasi selengkapnya mengenai menggunakan kebijakan IAM dengan Systems Manager, lihat Identity and access management untuk AWS Systems Manager atau Tindakan, sumber daya, dan kunci kondisi untuk AWS Systems Manager dalam Referensi Otorisasi Layanan.

AWS Systems ManagerState Managerpenelusuran

Panduan berikut menunjukkan cara membuat dan mengkonfigurasiState Manager asosiasi dengan menggunakan konsol Systems Manager atauAWS Command Line Interface (AWS CLI). Panduan tersebut juga menunjukkan bagaimana untuk secara otomatis melakukan tugas administratif umum dengan menggunakanState Manager, kemampuanAWS Systems Manager.

Topik

- Panduan: Membuat asosiasi yang menjalankan file MOF
- Walkthrough: Membuat asosiasi yang menjalankan buku pedoman Ansible
- Walkthrough: Membuat asosiasi yang menjalankan resep Chef
- Walkthrough: Perbarui secara otomatis (SSM AgentCLI)
- Panduan: Secara otomatis memperbarui driver PV pada instans EC2 untuk Windows Server (konsol)

Panduan: Membuat asosiasi yang menjalankan file MOF

Anda dapat menjalankan file Managed Object Format (MOF) untuk memberlakukan status yang diinginkan pada node terkelola Windows Server denganState ManagerAWS Systems Manager, kemampuan, dengan menggunakan dokumen`AWS-ApplyDSCMofs` SSM. Dokumen `AWS-ApplyDSCMofs` memiliki dua mode eksekusi. Dengan mode pertama, Anda dapat mengkonfigurasi asosiasi untuk memindai dan melaporkan jika node terkelola dalam status yang diinginkan ditentukan dalam file MOF yang ditentukan. Dalam modus kedua, Anda dapat menjalankan file MOF dan mengubah konfigurasi node Anda berdasarkan sumber daya dan nilai-nilai mereka yang didefinisikan dalam file MOF. Dokumen `AWS-ApplyDSCMofs` mengizinkan Anda untuk mengunduh dan menjalankan file konfigurasi MOF dari Amazon Simple Storage Service (Amazon S3), berbagi lokal, atau dari situs web aman dengan domain HTTPS.

State Managerlog dan melaporkan status setiap eksekusi file MOF selama asosiasi berjalan. State Managerjuga melaporkan output dari setiap eksekusi file MOF sebagai peristiwa kepatuhan yang dapat Anda lihat pada halaman AWS Systems ManagerKepatuhan.

Eksekusi file MOF dibuat pada Windows PowerShell Desired State Configuration (PowerShell DSC). PowerShell DSC adalah platform deklaratif yang digunakan untuk konfigurasi, deployment, dan pengelolaan sistem Windows. PowerShell DSC mengizinkan administrator untuk menggambarkan, dalam dokumen teks sederhana yang disebut konfigurasi DSC, bagaimana mereka ingin server dikonfigurasi. Konfigurasi PowerShell DSC adalah PowerShell skrip khusus yang menyatakan apa yang harus dilakukan, tetapi bukan cara melakukannya. Menjalankan konfigurasi menghasilkan file MOF. File MOF dapat diterapkan ke satu server atau lebih untuk mencapai konfigurasi yang diinginkan untuk server tersebut. PowerShell Sumber daya DSC melakukan pekerjaan yang sebenarnya dalam memberlakukan konfigurasi. Untuk informasi selengkapnya, lihat Tinjauan Windows PowerShell Desired State Configuration.

Topik

Menggunakan Amazon S3 untuk menyimpan artifact

Jika Anda menggunakan Amazon S3 untuk menyimpan PowerShell modul, file MOF, laporan kepatuhan, atau laporan status, makaAWS Identity and Access Management (IAM) role yang digunakanAWS Systems ManagerSSM Agent harus memilikiGetObject danListBucket izin pada bucket. Jika Anda tidak memberikan izin ini, sistem akan menampilkan kesalahan Akses Ditolak. Di bawah ini adalah informasi penting tentang menyimpan artifact di Amazon S3.

- Jika bucket berada dalam Akun AWS yang berbeda, buat kebijakan sumber daya bucket yang memberikan izin akun (atau IAM role) `GetObject` dan `ListBucket`.
- Jika Anda ingin menggunakan sumber daya DSC khusus, Anda dapat mengunduh sumber daya ini dari bucket Amazon S3. Anda juga dapat menginstalnya secara otomatis dari PowerShell galeri.
- Jika Anda menggunakan Amazon S3 sebagai sumber modul, unggah modul sebagai file Zip dalam format sensitif huruf kapital berikut: *ModuleName*_*ModuleVersion*.zip. Contohnya: MyModule _1.0.0.zip.
- Semua file harus berada dalam root bucket. Struktur folder tidak didukung.

Menyelesaikan kredensial dalam file MOF

Kredensial diselesaikan dengan menggunakan AWS Secrets Manager atau AWS Systems Manager Parameter Store. Hal ini mengizinkan Anda untuk mengatur rotasi kredensial otomatis. Hal ini juga mengizinkan DSC untuk secara otomatis menyebarkan kredensial ke server Anda tanpa memindahkan MOF.

Untuk menggunakanAWS Secrets Manager rahasia dalam konfigurasi, buat objek PSCredential di mana nama pengguna adalah SecretId atau SecretARN dari rahasia yang berisi kredensil. Anda dapat menentukan nilai apa pun untuk kata sandi. Nilai diabaikan. Berikut adalah contohnya.

```
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
        }
    }
}
```

Kompilasi MOF Anda menggunakan PsAllowPlaintextPassword pengaturan dalam data konfigurasi. Hal ini boleh karena kredensialnya hanya berisi label.

Di Secrets Manager, pastikan bahwa node memiliki GetSecretValue akses dalam Kebijakan yang Dikelola IAM, dan secara opsional dalam Kebijakan Sumber Daya Rahasia jika ada. Untuk bekerja dengan DSC, rahasianya harus dalam format berikut.

```
{ 'Username': 'a_name', 'Password': 'a_password' }
```

Rahasianya dapat memiliki properti lain (misalnya, properti yang digunakan untuk rotasi), tetapi setidaknya harus memiliki properti nama pengguna dan kata sandi.

Kami sarankan Anda menggunakan metode rotasi multi-user, di mana Anda memiliki dua nama pengguna dan kata sandi yang berbeda, dan rotasi fungsi AWS Lambda membalik di antara keduanya. Metode ini mengizinkan Anda untuk memiliki beberapa akun aktif sekaligus menghilangkan risiko mengunci pengguna selama rotasi.

Menggunakan token dalam file MOF

Token memberi Anda kemampuan untuk memodifikasi nilai atribut sumber daya setelah MOF dikompilasi. Hal ini mengizinkan Anda untuk menggunakan kembali file MOF umum pada beberapa server yang memerlukan konfigurasi yang sama.

Substitusi token hanya bekerja untuk Atribut Sumber Daya jenis `String`. Namun, jika sumber daya Anda memiliki properti simpul CIM yang nested, hal itu juga menyelesaikan token dari`String` properti dalam node CIM. Anda tidak dapat menggunakan substitusi token untuk angka atau larik.

Contohnya, pertimbangkan skenario di mana Anda menggunakan xComputerManagement sumber daya dan Anda ingin mengubah nama komputer menggunakan DSC. Biasanya Anda akan membutuhkan file MOF khusus untuk mesin itu. Namun, dengan dukungan token, Anda dapat membuat satu file MOF dan menerapkannya ke semua node Anda. Di properti `ComputerName`, alih-alih mengkodekan nama komputer ke MOF, Anda dapat menggunakan token tipe Tag Instans. Nilai diselesaikan selama penguraian MOF. Lihat contoh berikut ini.

```
Configuration MyConfig
```

```
{
    xComputer Computer
    {
        ComputerName = '{tag:ComputerName}'
    }
}
```

Anda kemudian menetapkan tag pada node terkelola di konsol Systems Manager, atau tag Amazon Elastic Compute Cloud (Amazon EC2) di konsol Amazon EC2. Ketika Anda menjalankan dokumen, skrip menggantikan token {tag:ComputerName} token untuk nilai dari tag instans.

Anda juga dapat menggabungkan beberapa tag ke properti tunggal, seperti yang ditunjukkan dalam contoh berikut.

```
Configuration MyConfig
{
    File MyFile
    {
        DestinationPath = '{env:TMP}\{tag:ComputerName}'
        Type = 'Directory'
    }
}
```

Ada lima jenis token yang dapat Anda gunakan:

- tag: Amazon EC2 atau tag node terkelola.
- tagb64: Ini sama dengan tag, tetapi sistem menggunakan base64 untuk memecahkan kode nilainya. Hal ini memungkinkan Anda untuk menggunakan karakter khusus dalam nilai tag.
- env: Variabel Resolves Environment.
- ssm:Parameter Store nilai. Hanya tipe String dan Secure String yang didukung.
- tagssm: Ini sama seperti tag, tetapi jika tag tidak diatur pada node, sistem mencoba untuk menyelesaikan nilai dari parameter Systems Manager dengan nama yang sama. Hal ini berguna dalam situasi ketika Anda ingin 'nilai global default' tetapi Anda ingin bisa menimpanya pada satu node (misalnya, deployment one-box).

Berikut ini adalahParameter Store contoh yang menggunakan tipe`ssm` token.

```
File MyFile
```

```
{
    DestinationPath = "C:\ProgramData\ConnectionData.txt"
    Content = "{ssm:%servicePath%/ConnectionData}"
}
```

Token memainkan peran penting dalam mengurangi kode berlebihan dengan membuat file MOF generik dan dapat digunakan kembali. Jika Anda dapat menghindari file MOF server tertentu, maka tidak perlu layanan bangunan MOF. Layanan bangunan MOF meningkatkan biaya, memperlambat penyediaan waktu, dan meningkatkan risiko penyimpangan konfigurasi antara node yang dikelompokkan karena versi modul yang berbeda yang diinstal pada server pembuatan ketika MOF-nya dikompilasi.

Prasyarat

Sebelum Anda membuat asosiasi yang menjalankan file MOF, verifikasi bahwa node terkelola Anda memiliki prasyarat berikut yang terinstal:

- Windows PowerShell versi 5.0 atau lebih baru. Untuk informasi selengkapnya, lihat Persyaratan PowerShell Sistem Windows di Microsoft.com.
- AWS Tools for Windows PowerShell versi 3.3.261.0 atau lebih baru.
- SSM Agentversi 2.2 atau lebih baru.

Membuat asosiasi yang menjalankan file MOF

Untuk membuat asosiasi yang menjalankan file MOF

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih State Manager.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih State Manager.
3. Pilih State Manager, lalu pilih Buat asosiasi.
4. Di bidang Nama, tentukan nama. Ini opsional, tetapi direkomendasikan. Sebuah nama dapat membantu Anda memahami tujuan dari asosiasi ketika Anda membuatnya. Spasi tidak diperbolehkan dalam nama.

5. Di daftar Dokumen, pilih **AWS-ApplyDSCMofs**.
6. Di bagian Parameter, tentukan pilihan Anda untuk parameter input yang diperlukan dan opsional.

   a. Mofs Untuk Terapkan: Tentukan satu file MOF atau lebih untuk dijalankan ketika asosiasi ini berjalan. Gunakan koma untuk memisahkan daftar file MOF. Anda dapat menentukan opsi berikut untuk menemukan file MOF.

   - Nama bucket Amazon S3. Nama bucket harus menggunakan huruf kecil. Tentukan informasi ini dengan menggunakan format berikut.

     ```
     s3:doc-example-bucket:MOF_file_name.mof
     ```

     Jika Anda ingin menentukan sebuah Wilayah AWS, maka gunakan format berikut.

     ```
     s3:bucket_Region:doc-example-bucket:MOF_file_name.mof
     ```

   - Situs web yang aman. Tentukan informasi ini dengan menggunakan format berikut.

     ```
     https://domain_name/MOF_file_name.mof
     ```

     Ini contohnya.

     ```
     https://www.example.com/TestMOF.mof
     ```

   - Sistem file pada berbagi lokal. Tentukan informasi ini dengan menggunakan format berikut.

     ```
     \server_name\shared_folder_name\MOF_file_name.mof
     ```

     Ini contohnya.

     ```
     \StateManagerAssociationsBox\MOFs_folder\MyMof.mof
     ```

   b. Jalur Layanan: (Opsional) Jalur layanan adalah prefiks bucket Amazon S3 di mana Anda ingin menulis laporan dan informasi status. Atau, jalur layanan adalah jalur untuk tagParameter Store berbasis parameter. Ketika menyelesaikan tag berbasis parameter, sistem menggunakan {ssm:%servicePath%/*parameter_name*} untuk memasukkan nilai servicePath ke nama parameter. Misalnya, jika jalur layanan Anda adalah "WebServers/

Production" maka sistem menyelesaikan parameter sebagai: WebServers /Production/ *parameter_name*. Hal ini berguna saat Anda menjalankan beberapa lingkungan di akun yang sama.

c. Laporkan Nama Bucket: (Opsional) Masukkan nama bucket Amazon S3 di mana Anda ingin menulis data kepatuhan. Laporan disimpan dalam bucket ini dalam format JSON.

> **Note**
>
> Anda dapat mengawali nama bucket dengan Wilayah tempat bucket berada. Berikut ini adalah contohnya: us-west-2:MyMOFBucket. Jika Anda menggunakan proxy untuk titik akhir Amazon S3 di Wilayah tertentu yang tidak menyertakan us-east-1, awali nama bucket dengan Wilayah. Jika nama bucket tidak diawali, maka secara otomatis menemukan Wilayah bucket dengan menggunakan titik akhir us-east-1.

d. Mode Operasi Mof: PilihState Manager perilaku saat menjalankan **`AWS-ApplyDSCMofs`**asosiasi:

- Terapkan: Perbaiki konfigurasi node yang tidak sesuai.
- ReportOnly: Jangan perbaiki konfigurasi node, tetapi sebaliknya log semua data kepatuhan dan laporan node yang tidak sesuai.

e. Nama Bucket Status: (Opsional) Masukkan nama bucket Amazon S3 di mana Anda ingin menulis informasi status eksekusi MOF. Laporan status ini adalah ringkasan tunggal dari kepatuhan terbaru dari sebuah node. Ini berarti bahwa laporan ditimpa saat berikutnya asosiasi menjalankan file MOF.

> **Note**
>
> Anda dapat mengawali nama bucket dengan Wilayah tempat bucket berada. Inilah contohnya: `us-west-2:doc-example-bucket`. Jika Anda menggunakan proxy untuk titik akhir Amazon S3 di Wilayah tertentu yang tidak menyertakan us-east-1, awali nama bucket dengan Wilayah. Jika nama bucket tidak diawali, maka bucket secara otomatis menemukan Wilayah bucket menggunakan titik akhir us-east-1.

f. Nama Bucket Sumber Modul: (Opsional) Masukkan nama bucket Amazon S3 yang berisi file PowerShell modul. Jika Anda menentukan Tidak ada, pilih Benar untuk opsi berikutnya, Izinkan Sumber Modul Galeri PS.

> **Note**
>
> Anda dapat mengawali nama bucket dengan Wilayah tempat bucket berada. Inilah contohnya: `us-west-2:doc-example-bucket`. Jika Anda menggunakan proxy untuk titik akhir Amazon S3 di Wilayah tertentu yang tidak menyertakan us-east-1, awali nama bucket dengan Wilayah. Jika nama bucket tidak diawali, maka bucket secara otomatis menemukan Wilayah bucket menggunakan titik akhir us-east-1.

g. Izinkan Sumber Modul Galeri PS: (Opsional) Pilih Benar untuk mengunduh PowerShell modul dari https://www.powershellgallery.com/. Jika Anda memilih Salah, tentukan sumber untuk opsi sebelumnya, ModuleSourceBucketName.

h. Proxy Uri: (Opsional) Gunakan opsi ini untuk mengunduh file MOF dari server proxy.

i. Perilaku Reboot: (Opsional) Tentukan salah satu perilaku reboot berikut jika eksekusi file MOF Anda memerlukan reboot:

- AfterMof: Reboot node setelah semua eksekusi MOF selesai. Bahkan jika beberapa eksekusi MOF meminta reboot, sistem menunggu sampai semua eksekusi MOF selesai untuk reboot.
- Segera: Reboot node setiap kali eksekusi MOF memintanya. Jika menjalankan beberapa file MOF yang meminta reboot, maka node di-reboot beberapa kali.
- Tidak pernah: Nodes tidak di-reboot, bahkan jika eksekusi MOF secara eksplisit meminta reboot.

j. Menggunakan Nama Komputer Untuk Melapor: (Opsional) Aktifkan opsi ini untuk menggunakan nama komputer saat melaporkan informasi kepatuhan. Nilai default adalah salah, yang berarti bahwa sistem menggunakan ID node ketika melaporkan informasi kepatuhan.

k. Mengaktifkan Pencatatan Verbose: (Opsional) Kami merekomendasikan bahwa Anda mengaktifkan pencatatan verbose saat men-deploy file MOF untuk pertama kalinya.

> **Important**
>
> Bila diizinkan, pencatatan verbose menulis lebih banyak data ke bucket Amazon S3 Anda daripada pencatatan eksekusi asosiasi standar. Hal ini mungkin mengakibatkan performa yang lebih lambat dan biaya penyimpanan yang lebih tinggi untuk Amazon S3. Untuk mengurangi masalah ukuran penyimpanan, kami

sarankan Anda mengaktifkan kebijakan siklus hidup pada bucket Amazon S3 Anda. Untuk informasi selengkapnya, lihat Bagaimana Cara Membuat Kebijakan Siklus Hidup untuk Bucket S3? dalam Panduan Pengguna Amazon Simple Storage Service.

l. Aktifkan Pencatatan Debug: (Opsional) Kami merekomendasikan bahwa Anda mengaktifkan pencatatan debug untuk memecahkan masalah kegagalan MOF. Kami juga menyarankan Anda untuk menonaktifkan opsi ini untuk penggunaan normal.

> **Important**
>
> Bila diizinkan, pencatatan debug menulis lebih banyak data ke bucket Amazon S3 Anda daripada pencatatan eksekusi asosiasi standar. Hal ini mungkin mengakibatkan performa yang lebih lambat dan biaya penyimpanan yang lebih tinggi untuk Amazon S3. Untuk mengurangi masalah ukuran penyimpanan, kami sarankan Anda mengaktifkan kebijakan siklus hidup pada bucket Amazon S3 Anda. Untuk informasi selengkapnya, lihat Bagaimana Cara Membuat Kebijakan Siklus Hidup untuk Bucket S3? dalam Panduan Pengguna Amazon Simple Storage Service.

m. Jenis Kepatuhan: (Opsional) Tentukan jenis kepatuhan yang akan digunakan saat melaporkan informasi kepatuhan. Jenis kepatuhan default adalah Custom:DSC. Jika Anda membuat beberapa asosiasi yang menjalankan file MOF, maka pastikan untuk menentukan jenis kepatuhan yang berbeda untuk setiap asosiasi. Jika tidak, setiap asosiasi tambahan yang menggunakan Custom:DSC menimpa data kepatuhan yang ada.

n. Skrip Pre Reboot: (Opsional) Tentukan skrip untuk dijalankan jika konfigurasi telah menunjukkan bahwa reboot diperlukan. Skrip berjalan sebelum reboot. Skrip harus satu baris. Pisahkan baris tambahan dengan menggunakan titik koma.

7. Di bagian Target, pilih Menentukan tag atau Memilih Instans secara Manual. Jika Anda memilih untuk menargetkan sumber daya dengan menggunakan tag, masukkan kunci tag dan nilai tag di bidang yang disediakan. Untuk informasi selengkapnya tentang menggunakan target, lihat Tentang target dan kontrol tingkat dalam State Manager asosiasi.
8. Di bagian Tentukan jadwal, pilih Sesuai Jadwal atau Tidak ada jadwal. Jika Anda memilih Sesuai Jadwal, gunakan tombol yang disediakan untuk membuat jadwal cron atau rate untuk asosiasi.
9. Di bagian Opsi lanjutan:

- Di Keparahan kepatuhan, pilih tingkat keparahan untuk asosiasi. Pelaporan kepatuhan menunjukkan apakah status asosiasi sesuai atau tidak, bersama dengan tingkat keparahan yang Anda tunjukkan di sini. Untuk informasi selengkapnya, lihat Tentang kepatuhan State Manager asosiasi.

10. Di bagian Pengendalian rate, konfigurasikan opsi untuk menjalankanState Manager asosiasi di seluruh armada node terkelola. Untuk informasi selengkapnya tentang opsi ini, lihat Tentang target dan kontrol tingkat dalam State Manager asosiasi.

    Di bagian Konkurensi, pilih satu opsi:

    - Pilih target untuk memasukkan jumlah absolut dari target yang dapat menjalankan asosiasi secara bersamaan.
    - Pilih persentase untuk memasukkan persentase dari kumpulan target yang dapat menjalankan asosiasi secara bersamaan.

    Di bagian Batas kesalahan, pilih opsi:

    - Pilih kesalahan untuk memasukkan jumlah absolut kesalahan yang diizinkan sebelumState Manager berhenti menjalankan asosiasi pada target tambahan.
    - Pilih persentase untuk memasukkan persentase kesalahan yang diizinkan sebelumState Manager berhenti menjalankan asosiasi pada target tambahan.

11. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Aktifkan output penulisan ke S3. Masukkan nama bucket dan prefiks (folder) di dalam kotak.

    > **Note**
    >
    > Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah dari profil instans yang ditetapkan ke node terkelola, bukan data pengguna IAM yang melaksanakan tugas ini. Untuk informasi lebih lanjut, lihat Mengkonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berada di yang berbedaAkun AWS, pastikan bahwa profil instans atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

12. Pilih Buat Asosiasi.

State Managermembuat dan segera menjalankan asosiasi pada node atau target tertentu. Setelah eksekusi awal, asosiasi berjalan dalam interval yang sesuai dengan jadwal yang Anda tetapkan dan sesuai dengan aturan berikut:

- State Managermenjalankan asosiasi pada node yang online ketika interval dimulai dan melewati node offline.
- State Managermencoba untuk menjalankan asosiasi pada semua node yang dikonfigurasi selama interval.
- Jika asosiasi tidak berjalan selama interval (karena, misalnya, nilai konkurensi terbatas jumlah node yang dapat memproses asosiasi pada satu waktu), makaState Manager mencoba untuk menjalankan asosiasi selama interval berikutnya.
- State Managermencatat sejarah untuk semua interval yang dilewati. Anda dapat melihat riwayat di tab Riwayat Eksekusi.

> **Note**
>
> `AWS-ApplyDSCMofs` adalah dokumen Perintah Systems Manager. Ini berarti Anda juga dapat menjalankan dokumen ini dengan menggunakanRun Command, kemampuanAWS Systems Manager. Untuk informasi selengkapnya, lihat AWS Systems Manager Run Command.

Pemecahan Masalah

Bagian ini berisi informasi untuk membantu Anda memecahkan masalah yang membuat asosiasi menjalankan file MOF.

Mengaktifkan pencatatan yang ditingkatkan

Sebagai langkah pertama untuk pemecahan masalah, aktifkan pencatatan yang ditingkatkan. Lebih khusus lagi, lakukan hal berikut:

1. Verifikasi bahwa asosiasi dikonfigurasi untuk menulis output perintah ke Amazon S3 atau Amazon CloudWatch Logs (CloudWatch).
2. Atur parameter Aktifkan Pencatatan Verbose ke Betul.
3. Atur parameter Aktifkan Pencatatan Debug ke Betul.

Dengan pencatatan debug dan verbose diaktifkan, file output Stdout menyertakan rincian tentang eksekusi skrip. File output ini dapat membantu Anda mengidentifikasi di mana skrip gagal. File output Stderr berisi kesalahan yang terjadi selama eksekusi skrip.

Masalah umum

Bagian ini mencakup informasi tentang masalah umum yang dapat terjadi saat membuat asosiasi yang menjalankan file MOF dan langkah-langkah untuk memecahkan masalah ini.

MOF saya tidak diterapkan

JikaState Manager gagal untuk menerapkan asosiasi untuk node Anda, maka mulai dengan meninjau file output Stderr. File ini dapat membantu Anda memahami akar dari masalah tersebut. Juga, verifikasikan hal berikut:

- Node ini memiliki izin akses yang diperlukan untuk semua bucket Amazon S3 yang terkait dengan MOF. Secara khusus:
  - s3:GetObject Izin: Ini diperlukan untuk file MOF dalam bucket Amazon S3 privat dan modul khusus dalam bucket Amazon S3.
  - s3:PutObject Izin: Ini diperlukan untuk menulis laporan kepatuhan dan status kepatuhan untuk bucket Amazon S3.
- Jika Anda menggunakan tag, pastikan bahwa node memiliki kebijakan IAM yang diperlukan. Menggunakan tag memerlukan IAM role instans untuk memiliki kebijakan yang mengizinkan tindakan `ec2:DescribeInstances` dan `ssm:ListTagsForResource`.
- Pastikan bahwa node memiliki tag yang diharapkan atau parameter SSM yang ditetapkan.
- Pastikan bahwa tag atau parameter SSM tidak salah eja.
- Coba aplikasikan MOF secara lokal pada node untuk memastikan tidak ada masalah dengan file MOF itu sendiri.

MOF saya sepertinya gagal, tapi eksekusi Systems Manager berhasil

Jika dokumen `AWS-ApplyDSCMofs` berhasil dijalankan, maka status eksekusi Systems Manager menunjukkan Sukses. Status ini tidak mencerminkan status kepatuhan node Anda terhadap persyaratan konfigurasi dalam file MOF. Untuk melihat status kepatuhan node Anda, lihat laporan kepatuhan. Anda dapat melihat laporan JSON di Bucket Laporan Amazon S3. Hal ini berlaku untukRun Command danState Manager eksekusi. Selain ituState Manager, Anda dapat melihat rincian kepatuhan pada halaman Kepatuhan Systems Manager.

Stderr menyatakan: Kegagalan resolusi nama mencoba mencapai layanan

Kesalahan ini menunjukkan bahwa skrip tidak dapat mencapai layanan jarak jauh. Kemungkinan besar, skrip tidak bisa mencapai Amazon S3. Masalah ini paling sering terjadi ketika skrip mencoba untuk menulis laporan kepatuhan atau status kepatuhan untuk bucket Amazon S3 yang disediakan dalam parameter dokumen. Biasanya, kesalahan ini terjadi ketika lingkungan komputasi menggunakan firewall atau proxy transparan yang menyertakan daftar yang diizinkan. Untuk mengatasi masalah ini:

- Gunakan sintaks bucket khusus Wilayah untuk semua parameter bucket Amazon S3. Misalnya, Mofs untuk Terapkan parameter harus diformat sebagai berikut:

  s3: *ember-wilayah*: *ember-nama*: *mof-file-name*.mof.

  Ini contohnya: `s3:us-west-2:doc-example-bucket:my-mof.mof`

  Nama bucket Laporan, Status, dan Modul Sumber harus diformat sebagai berikut.

  bucket-region:*bucket-name*. Ini contohnya: `us-west-1:doc-example-bucket`
- Jika sintaks khusus Wilayah tidak memperbaiki masalah, maka pastikan bahwa node yang ditargetkan dapat mengakses Amazon S3 di Wilayah yang diinginkan. Untuk memverifikasi ini:
  1. Cari nama titik akhir untuk Amazon S3 di Wilayah Amazon S3 yang sesuai. Untuk informasi, lihat Endpoint Layanan Amazon S3 di Referensi Umum Amazon Web Services.
  2. Log on ke node target dan jalankan perintah ping berikut.

     ```
     ping s3.s3-region.amazonaws.com
     ```

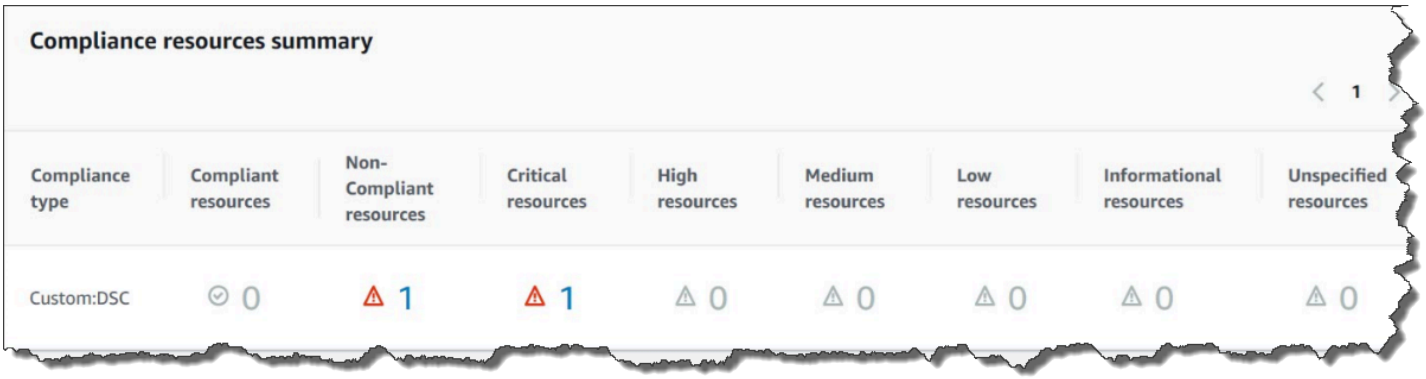
     Jika ping gagal, itu berarti Amazon S3 mati, atau proxy firewall/transparan memblokir akses ke Wilayah Amazon S3, atau node tidak dapat mengakses internet.

Melihat rincian kepatuhan sumber daya DSC

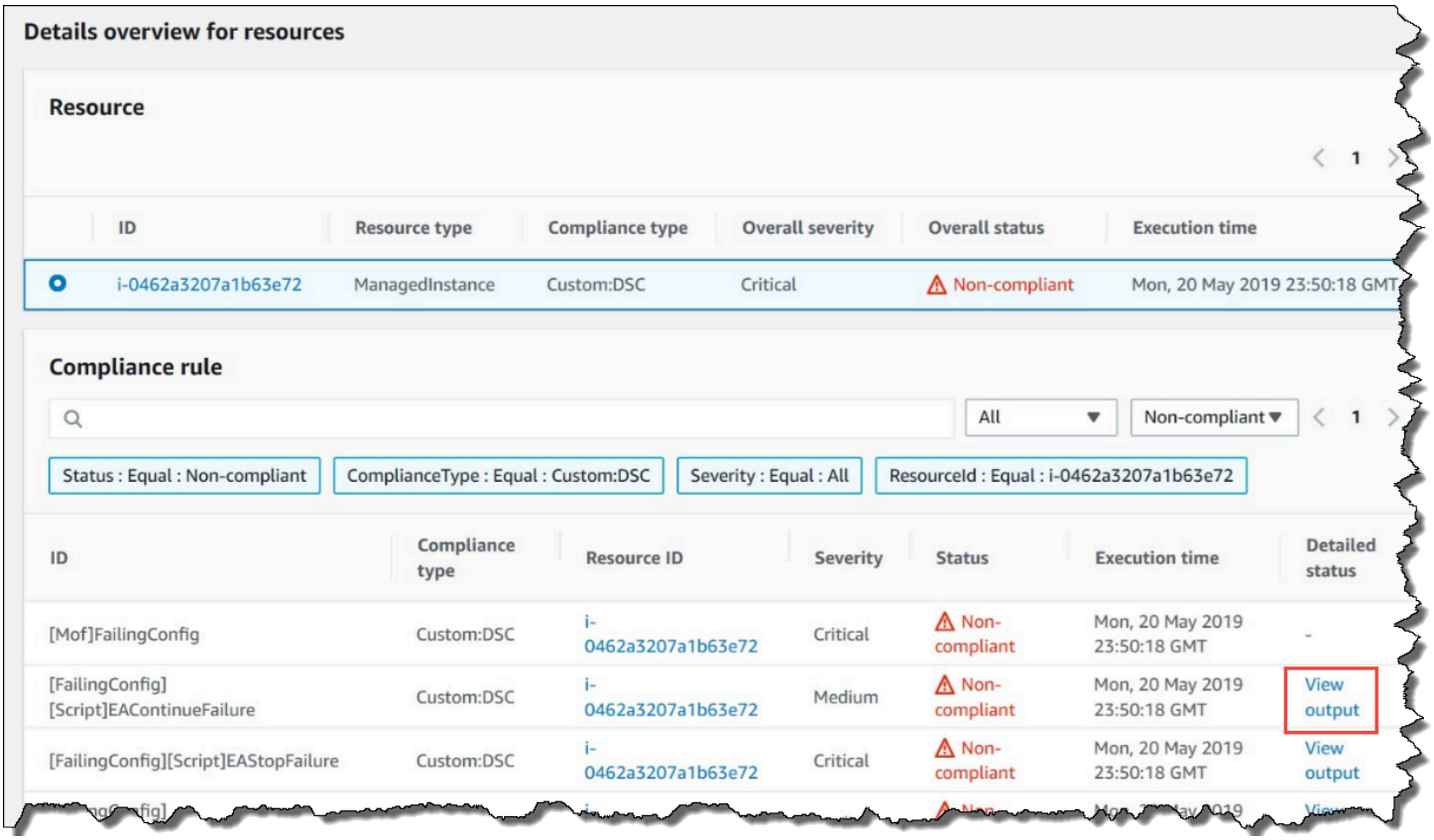
Systems Manager menangkap informasi kepatuhan tentang kegagalan sumber daya DSC di Amazon S3 Bucket Status yang Anda tentukan ketika Anda menjalankan dokumen `AWS-ApplyDSCMofs`. Mencari informasi tentang kegagalan sumber daya DSC dalam bucket Amazon S3 dapat memakan waktu. Sebaliknya, Anda dapat melihat informasi ini di halaman Kepatuhan Systems Manager.

Bagian Ringkasan sumber daya kepatuhan menampilkan jumlah sumber daya yang gagal. Pada contoh berikut, ComplianceTypeadalah Custom:DSC dan satu sumber daya tidak sesuai.

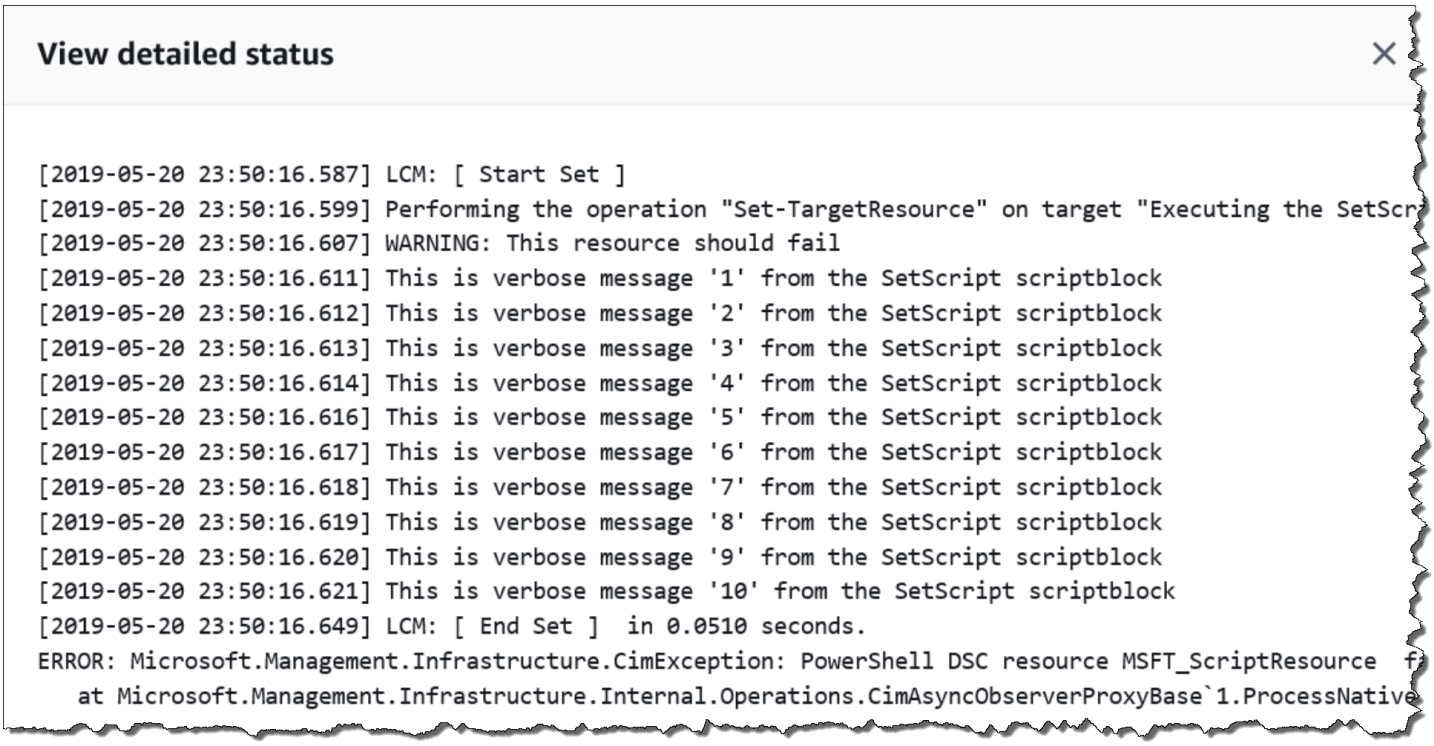
> **Note**
>
> Custom:DSC adalah ComplianceTypenilai default dalam`AWS-ApplyDSCMofs` dokumen. Nilai ini dapat disesuaikan.



Bagian Gambaran umum detail untuk sumber daya menampilkan informasi tentang sumber daya AWS sumber daya DSC yang tidak sesuai. Bagian ini juga mencakup nama MOF, langkah-langkah eksekusi skrip, dan (bila berlaku) tautan Lihat outputuntuk melihat informasi status yang terperinci.



Tautan Lihat output menampilkan 4.000 karakter terakhir dari status yang terperinci. Systems Manager dimulai dengan pengecualian sebagai elemen pertama, dan kemudian memindai kembali melalui pesan verbose dan menambahkan sebanyak mungkin sampai mencapai kuota karakter 4.000. Proses ini menampilkan pesan log yang dikeluarkan sebelum pengecualian diluncurkan, yang merupakan pesan yang paling relevan untuk pemecahan masalah.



Untuk informasi tentang cara melihat informasi kepatuhan, lihat AWS Systems Manager Kepatuhan.

Situasi yang mempengaruhi pelaporan kepatuhan

JikaState Manager asosiasi gagal, maka tidak ada data kepatuhan yang dilaporkan. Lebih khusus lagi, jika MOF gagal untuk memproses, maka Systems Manager tidak melaporkan item kepatuhan apa pun karena asosiasi gagal. Misalnya, jika Systems Manager mencoba untuk mengunduh MOF dari bucket Amazon S3 di mana node tidak memiliki izin untuk mengakses, maka asosiasi gagal dan tidak ada data kepatuhan yang dilaporkan.

Jika sumber daya dalam MOF kedua gagal, maka Systems Manager akan melaporkan data kepatuhan. Sebagai contoh, jika MOF mencoba untuk membuat file pada drive yang tidak ada, kemudian Systems Manager melaporkan kepatuhan karena dokumen `AWS-ApplyDSCMofs` mampu memproses sepenuhnya, yang berarti asosiasi berhasil berjalan.

Walkthrough: Membuat asosiasi yang menjalankan buku pedoman Ansible

Anda dapat membuat State Manager asosiasi yang menjalankan Ansible buku pedoman dengan menggunakan dokumen `AWS-ApplyAnsiblePlaybooks` SSM. State Manageradalah kemampuan AWS Systems Manager. Dokumen ini menawarkan manfaat berikut untuk menjalankan playbook:

- Support untuk menjalankan playbook yang kompleks
- Support untuk mengunduh pedoman dari GitHub dan Amazon Simple Storage Service (Amazon S3)
- Support untuk struktur playbook yang terkompresi
- Pencatatan yang ditingkatkan
- Kemampuan untuk menentukan playbook mana yang akan dijalankan saat playbook dipaketkan

> **Note**
>
> Systems Manager menyertakan dua dokumen SSM yang memungkinkan Anda membuat State Manager asosiasi yang menjalankan Ansible playbook: `AWS-RunAnsiblePlaybook` dan. `AWS-ApplyAnsiblePlaybooks` Dokumen `AWS-RunAnsiblePlaybook` tidak lagi digunakan. Ini tetap tersedia di Systems Manager untuk tujuan warisan. Kami menyarankan agar Anda menggunakan dokumen `AWS-ApplyAnsiblePlaybooks` karena peningkatan yang dijelaskan di sini.
>
> Asosiasi yang menjalankan Ansible buku pedoman tidak didukung. macOS

Support untuk menjalankan playbook yang kompleks

Dokumen `AWS-ApplyAnsiblePlaybooks` mendukung playbook kompleks yang dipaketkan karena akan menyalin keseluruhan struktur file ke direktori lokal sebelum mengeksekusi playbook utama yang ditentukan. Anda dapat memberikan playbook sumber dalam file Zip atau dalam struktur direktori. File atau direktori Zip dapat disimpan di GitHub atau Amazon S3.

Support untuk mengunduh pedoman dari GitHub

Dokumen `AWS-ApplyAnsiblePlaybooks` menggunakan plugin `aws:downloadContent` untuk mengunduh file playbook. File dapat disimpan GitHub dalam satu file atau sebagai kumpulan file playbook gabungan. Untuk mengunduh kontenGitHub, tentukan informasi tentang GitHub repositori Anda dalam format JSON. Inilah contohnya.

```
{
   "owner":"TestUser",
   "repository":"GitHubTest",
   "path":"scripts/python/test-script",
   "getOptions":"branch:master",
   "tokenInfo":"{{ssm-secure:secure-string-token}}"
}
```

Support untuk mengunduh pedoman dari Amazon S3

Anda juga dapat menyimpan dan mengunduh Ansible buku pedoman di Amazon S3 sebagai satu file.zip atau struktur direktori. Untuk mengunduh konten dari Amazon S3, tentukan jalur ke file. Berikut ini adalah dua contoh.

Contoh 1: Unduh file playbook tertentu

```
{
   "path":"https://s3.amazonaws.com/doc-example-bucket/playbook.yml"
}
```

Contoh 2: Unduh isi direktori

```
{
   "path":"https://s3.amazonaws.com/doc-example-bucket/ansible/webservers/"
}
```

> **Important**
>
> Jika Anda menentukan Amazon S3, maka profil instance AWS Identity and Access Management (IAM) pada node terkelola harus dikonfigurasi dengan kebijakan tersebut. `AmazonS3ReadOnlyAccess` Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager.

Support untuk struktur playbook terkompresi

Dokumen `AWS-ApplyAnsiblePlaybooks` mengizinkan Anda untuk menjalankan file .zip yang dikompresi dalam paket yang diunduh. Dokumen memeriksa apakah file yang diunduh berisi file yang dikompresi dalam format .zip. Jika .zip ditemukan, dokumen secara otomatis mendekompresi file dan kemudian menjalankan otomatisasi yang ditentukan. Ansible

Penebangan yang ditingkatkan

Dokumen `AWS-ApplyAnsiblePlaybooks` termasuk parameter opsional untuk menentukan tingkat pencatatan yang berbeda. Spesifikasikan -v untuk verbositas rendah, -vv atau –vvv untuk verbositas medium, dan -vvvv untuk pencatatan tingkat debug. Opsi ini langsung dipetakan ke opsi Ansible verbositas.

Kemampuan untuk menentukan pedoman mana yang akan dijalankan saat pedoman dibundel

Dokumen `AWS-ApplyAnsiblePlaybooks` menyertakan parameter yang diperlukan untuk menentukan playbook mana yang akan dijalankan ketika beberapa playbook dipaketkan. Opsi ini memberikan fleksibilitas untuk menjalankan playbook untuk men-support kasus penggunaan yang berbeda.

Dependensi yang terinstal

Jika Anda menentukan True untuk InstallDependenciesparameter, maka Systems Manager memverifikasi bahwa node Anda memiliki dependensi berikut diinstal:

- Ubuntu Server/Debian Server: Apt-get (Manajemen Package), Python 3,, Unzip Ansible
- Amazon Linux: Ansible
- RHEL: Python 3Ansible,, Unzip

Jika satu atau lebih dari dependensi ini tidak ditemukan, maka Systems Manager secara otomatis menginstalnya.

Buat asosiasi yang menjalankan Ansible playbook (konsol)

Prosedur berikut menjelaskan cara menggunakan konsol Systems Manager untuk membuat State Manager asosiasi yang menjalankan Ansible playbook menggunakan `AWS-ApplyAnsiblePlaybooks` dokumen.

Untuk membuat asosiasi yang menjalankan Ansible playbooks (konsol)

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih State Manager.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih State Manager.

3. Pilih State Manager, lalu pilih Buat asosiasi.
4. Untuk Nama, tentukan nama yang membantu anda mengingat tujuan asosiasi.
5. Di daftar Dokumen, pilih **`AWS-ApplyAnsiblePlaybooks`**.
6. Di bagian Parameter, untuk Jenis Sumber, pilih salah satu GitHubatau S3.

   GitHub

   Jika Anda memilih GitHub, masukkan informasi repositori dalam format berikut.

   ```
   {
      "owner":"user_name",
      "repository":"name",
      "path":"path_to_directory_or_playbook_to_download",
      "getOptions":"branch:branch_name",
      "tokenInfo":"{{(Optional)_token_information}}"
   }
   ```

   S3

   Jika Anda memilih S3, masukkan informasi jalur dalam format berikut.

   ```
   {
      "path":"https://s3.amazonaws.com/path_to_directory_or_playbook_to_download"
   }
   ```

7. Untuk Instal Dependensi, pilih opsi.
8. (Opsional) Untuk File Playbook, masukkan nama file. Jika file Zip berisi buku playbook, tentukan jalur relatif ke file Zip.
9. (Opsional) Untuk Variabel Ekstra, masukkan variabel yang State Manager ingin Anda kirim Ansible saat runtime.
10. (Opsional) Untuk Periksa, pilih opsi.
11. (Opsional) Untuk Verbose, pilih opsi.
12. Untuk Target, pilih satu opsi. Untuk informasi tentang menggunakan target, lihat Tentang target dan kontrol tingkat dalam State Manager asosiasi.

13. Di bagian Tentukan jadwal, pilih Sesuai jadwal atau Tidak ada jadwal. Jika Anda memilih Sesuai jadwal, maka gunakan tombol yang disediakan untuk membuat jadwal cron atau rate untuk asosiasi.
14. Di bagian Opsi lanjutan, untuk Keparahan kepatuhan, pilih tingkat keparahan untuk asosiasi. Pelaporan kepatuhan menunjukkan apakah status asosiasi sesuai atau tidak, bersama dengan tingkat keparahan yang Anda tunjukkan di sini. Untuk informasi selengkapnya, lihat Tentang kepatuhan State Manager asosiasi.
15. Di bagian Rate control, konfigurasikan opsi untuk menjalankan State Manager asosiasi di seluruh armada node terkelola. Untuk informasi tentang menggunakan kontrol rate, lihat Tentang target dan kontrol tingkat dalam State Manager asosiasi.

    Di bagian Konkurensi, pilih satu opsi:

    - Pilih target untuk memasukkan jumlah absolut dari target yang dapat menjalankan asosiasi secara bersamaan.
    - Pilih persentase untuk memasukkan persentase dari kumpulan target yang dapat menjalankan asosiasi secara bersamaan.

    Di bagian Batas kesalahan, pilih opsi:

    - Pilih kesalahan untuk memasukkan jumlah absolut kesalahan yang diizinkan sebelum State Manager berhenti menjalankan asosiasi pada target tambahan.
    - Pilih persentase untuk memasukkan persentase kesalahan yang diizinkan sebelum State Manager berhenti menjalankan asosiasi pada target tambahan.
16. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Aktifkan output penulisan ke S3. Masukkan nama bucket dan prefiks (folder) di dalam kotak.

    > **Note**
    >
    > Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin dari profil instance yang ditetapkan ke node terkelola, bukan izin pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbeda Akun AWS, verifikasi bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

17. Pilih Buat Asosiasi.

> **Note**
>
> Jika Anda menggunakan tag untuk membuat asosiasi pada satu atau lebih node target, dan kemudian Anda menghapus tag dari node, node itu tidak lagi menjalankan asosiasi. Node terlepas dari State Manager dokumen.

Buat asosiasi yang menjalankan Ansible buku pedoman (CLI)

Prosedur berikut menjelaskan cara menggunakan AWS Command Line Interface (AWS CLI) untuk membuat State Manager asosiasi yang menjalankan Ansible buku pedoman dengan menggunakan `AWS-ApplyAnsiblePlaybooks` dokumen.

Untuk membuat asosiasi yang menjalankan Ansible buku pedoman (CLI)

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI.

2. Jalankan salah satu perintah berikut untuk membuat asosiasi yang menjalankan Ansible buku pedoman dengan menargetkan node menggunakan tag. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri. Command (A) menentukan GitHub sebagai tipe sumber. Command (B) menentukan Amazon S3 sebagai jenis sumber.

   (A) GitHub sumber

   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"
```

Ini contohnya.

```
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) Sumber 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"
```

Ini contohnya.

```
aws ssm create-association --name "AWS-ApplyAnsiblePlaybooks" \
    --targets "Key=tag:OS,Values=Linux" \
    --parameters '{"SourceType":["S3"],"SourceInfo":["{\"path\":\"https://
s3.amazonaws.com/DOC-EXAMPLE-BUCKET/playbook.yml\"}"],"InstallDependencies":
["True"],"PlaybookFile":["playbook.yml"],"ExtraVariables":["SSM=True"],"Check":
["False"],"Verbose":["-v"]}' \
    --association-name "AnsibleAssociation" \
    --schedule-expression "cron(0 2 ? * SUN *)"
```

> **Note**
>
> State Managerasosiasi tidak mendukung semua ekspresi cron dan rate. Untuk informasi selengkapnya tentang membuat ekspresi cron dan rate untuk asosiasi, lihat Referensi: Ekspresi cron dan rate untuk Systems Manager.

Sistem mencoba untuk membuat asosiasi pada node dan segera menerapkan status.

3. Jalankan perintah berikut untuk menampilkan status terbaru dari asosiasi yang baru saja Anda buat.

```
aws ssm describe-association --association-id "ID"
```

Walkthrough: Membuat asosiasi yang menjalankan resep Chef

Anda dapat membuat State Manager asosiasi yang menjalankan Chef resep dengan menggunakan dokumen `AWS-ApplyChefRecipes` SSM. State Manageradalah kemampuan AWS Systems Manager. Anda dapat menargetkan node terkelola Systems Manager berbasis Linux dengan dokumen SSM `AWS-ApplyChefRecipes`. Dokumen ini menawarkan manfaat berikut untuk menjalankan Chef resep:

- Mendukung beberapa rilis Chef (Chef11 hingga Chef 18).
- Secara otomatis menginstal perangkat lunak Chef klien pada node target.
- Secara opsional menjalankan pemeriksaan kepatuhan Systems Manager pada node target, dan menyimpan hasil pemeriksaan kepatuhan di bucket Amazon Simple Storage Service (Amazon S3).
- Menjalankan beberapa buku masak dan resep dalam satu dokumen.
- Secara opsional menjalankan resep dalam `why-run` mode, untuk menunjukkan resep mana yang berubah pada node target tanpa membuat perubahan.
- Secara opsional menerapkan atribut JSON khusus untuk `chef-client` dijalankan.
- Opsional menerapkan atribut JSON kustom dari file sumber yang disimpan di lokasi yang Anda tentukan.

Anda dapat menggunakan bucket Git GitHub,, HTTP, atau Amazon S3 sebagai sumber unduhan untuk Chef buku masak dan resep yang Anda tentukan dalam dokumen. `AWS-ApplyChefRecipes`

> **Note**
>
> Asosiasi yang menjalankan Chef resep tidak didukung pada macOS atauWindows Server.

Prasyarat: Menyiapkan asosiasi, repositori, dan buku masak Anda

Sebelum Anda membuat `AWS-ApplyChefRecipes` dokumen, siapkan buku masak dan Chef repositori buku masak Anda. Jika Anda belum memiliki Chef buku masak yang ingin Anda gunakan, Anda dapat memulai dengan menggunakan `HelloWorld` buku masak tes yang AWS telah disiapkan untuk Anda. Dokumen `AWS-ApplyChefRecipes` sudah menunjuk ke buku masak ini secara default. Buku masak Anda harus diatur serupa dengan struktur direktori berikut. Dalam contoh berikut, `jenkins` dan `nginx` merupakan contoh Chef buku masak yang tersedia di Chef Supermarketsitus Chef web.

Meskipun tidak AWS dapat secara resmi mendukung buku masak di Chef Supermarketsitus web, banyak dari mereka bekerja dengan `AWS-ApplyChefRecipes` dokumen tersebut. Berikut ini adalah contoh kriteria untuk menentukan kapan Anda menguji buku masak komunitas:

- Buku masak harus men-support sistem operasi berbasis Linux dari node terkelola Systems Manager yang Anda targetkan.
- Buku masak harus valid untuk versi Chef klien (Chef11 hingga Chef 18) yang Anda gunakan.
- Buku masak kompatibel denganChef Infra Client, dan, tidak memerlukan server Chef.

Verifikasi bahwa Anda dapat mencapai `Chef.io` situs web, sehingga buku masak apa pun yang Anda tentukan dalam daftar jalankan dapat diinstal saat dokumen Systems Manager (dokumen SSM) berjalan. Menggunakan folder `cookbooks` yang di-nest disupport, tetapi tidak diperlukan; Anda dapat menyimpan buku masak langsung di tingkat root.

```
<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

Sebelum Anda membuat State Manager asosiasi yang menjalankan Chef resep, ketahuilah bahwa dokumen yang dijalankan menginstal perangkat lunak Chef klien pada node terkelola Systems Manager Anda, kecuali jika Anda menetapkan nilai versi Chef klien. None Operasi ini menggunakan skrip instalasi dari Chef untuk menginstal Chef komponen atas nama Anda. Sebelum Anda menjalankan `AWS-ApplyChefRecipes` dokumen, pastikan perusahaan Anda dapat mematuhi persyaratan hukum yang berlaku, termasuk ketentuan lisensi yang berlaku untuk penggunaan Chef perangkat lunak. Untuk informasi lebih lanjut, lihat situs Chefweb.

Systems Manager dapat mengirimkan laporan kepatuhan ke bucket S3, konsol Systems Manager, atau membuat hasil kepatuhan tersedia sebagai respons terhadap perintah API Systems Manager. Untuk menjalankan laporan kepatuhan Systems Manager, profil instans yang dilampirkan ke node

terkelola Systems Manager harus memiliki izin untuk menulis ke bucket S3. Profil instans harus memiliki izin untuk menggunakan API `PutComplianceItem` Systems Manager. Untuk informasi selengkapnya tentang kepatuhan Systems Manager, lihat AWS Systems Manager Kepatuhan.

Mencatat dokumen yang berjalan

Saat menjalankan dokumen Systems Manager (dokumen SSM) menggunakan State Manager asosiasi, Anda dapat mengonfigurasi asosiasi untuk memilih output dokumen yang dijalankan, dan Anda dapat mengirim output ke Amazon S3 atau CloudWatch Amazon Logs CloudWatch (Log). Untuk membantu memudahkan pemecahan masalah saat asosiasi selesai berjalan, verifikasi bahwa asosiasi dikonfigurasi untuk menulis output perintah ke bucket Amazon S3 atau Log. CloudWatch Untuk informasi selengkapnya, lihat Bekerja dengan asosiasi di Systems Manager.

Menerapkan atribut JSON ke target saat menjalankan resep

Anda dapat menentukan atribut JSON untuk Chef klien Anda untuk diterapkan ke node target selama menjalankan asosiasi. Saat menyiapkan asosiasi, Anda dapat memberikan JSON mentah atau memberikan jalur ke file JSON yang disimpan di Amazon S3.

Gunakan atribut JSON ketika Anda ingin menyesuaikan bagaimana resep dijalankan tanpa harus memodifikasi resep itu sendiri, misalnya:

- Mengesampingkan sejumlah kecil atribut

  Gunakan JSON khusus untuk menghindari keharusan mempertahankan beberapa versi resep untuk mengakomodasi perbedaan kecil.
- Memberikan nilai variabel

  Gunakan JSON kustom untuk menentukan nilai yang mungkin berubah dari run-to-run. Misalnya, jika Chef buku masak Anda mengonfigurasi aplikasi pihak ketiga yang menerima pembayaran, Anda dapat menggunakan JSON khusus untuk menentukan URL titik akhir pembayaran.

Menentukan atribut dalam JSON mentah

Berikut ini adalah contoh format yang dapat Anda gunakan untuk menentukan atribut JSON kustom untuk Chef resep Anda.

```
{"filepath":"/tmp/example.txt", "content":"Hello, World!"}
```

Menentukan jalur ke file JSON

Berikut ini adalah contoh format yang dapat Anda gunakan untuk menentukan jalur ke atribut JSON kustom untuk Chef resep Anda.

```
{"sourceType":"s3", "sourceInfo":"someS3URL1"}, {"sourceType":"s3",
 "sourceInfo":"someS3URL2"}
```

Gunakan Git sebagai sumber buku masak

`AWS-ApplyChefRecipes`Dokumen menggunakan `aws:downloadContent`plugin untuk mengunduh Chef buku masak. Untuk mengunduh konten dari Git, tentukan informasi tentang repositori Git Anda dalam format JSON seperti pada contoh berikut. Ganti masing-masing *example-resource-placeholder* dengan informasi Anda sendiri.

```
{
   "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}}"
}
```

Gunakan GitHub sebagai sumber buku masak

Dokumen `AWS-ApplyChefRecipes` menggunakan plugin `aws:downloadContent` untuk mengunduh buku masak. Untuk mengunduh konten dariGitHub, tentukan informasi tentang GitHub repositori Anda dalam format JSON seperti pada contoh berikut. Ganti masing-masing *example-resource-placeholder* dengan informasi Anda sendiri.

```
{
   "owner":"TestUser",
   "repository":"GitHubCookbookRepository",
   "path":"cookbooks/HelloWorld",
   "getOptions":"branch:refs/head/main",
   "tokenInfo":"{{ssm-secure:token-secure-string-parameter}}"
}
```

Gunakan HTTP sebagai sumber buku masak

Anda dapat menyimpan Chef buku masak di lokasi HTTP kustom baik sebagai satu `.zip` atau `tar.gz` file, atau struktur direktori. Untuk mengunduh konten dari HTTP, tentukan jalur ke file atau direktori dalam format JSON seperti pada contoh berikut. Ganti masing-masing *example-resource-placeholder* dengan informasi Anda sendiri.

```
{
   "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}}"
}
```

Menggunakan Amazon S3 sebagai sumber buku masak

Anda juga dapat menyimpan dan mengunduh Chef buku masak di Amazon S3 sebagai `.zip` satu `tar.gz` atau file, atau struktur direktori. Untuk mengunduh konten dari Amazon S3, tentukan jalur ke file dalam format JSON seperti pada contoh berikut. Ganti masing-masing *example-resource-placeholder* dengan informasi Anda sendiri.

Contoh 1: Unduh buku masak tertentu

```
{
   "path":"https://s3.amazonaws.com/chef-cookbooks/HelloWorld.zip"
}
```

Contoh 2: Unduh isi direktori

```
{
   "path":"https://s3.amazonaws.com/chef-cookbooks-test/HelloWorld"
}
```

> **Important**
>
> Jika Anda menentukan Amazon S3, profil instans AWS Identity and Access Management (IAM) pada node terkelola harus dikonfigurasi dengan kebijakan. `AmazonS3ReadOnlyAccess` Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager.

Topik

- Buat asosiasi yang menjalankan Chef resep (konsol)
- Buat asosiasi yang menjalankan Chef resep (CLI)
- Melihat detail kepatuhan sumber daya Chef

Buat asosiasi yang menjalankan Chef resep (konsol)

Prosedur berikut menjelaskan cara menggunakan konsol Systems Manager untuk membuat State Manager asosiasi yang menjalankan Chef buku masak dengan menggunakan `AWS-ApplyChefRecipes` dokumen.

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih State Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih State Manager.
3. Pilih State Manager, lalu pilih Buat asosiasi.
4. Untuk Nama, masukkan nama yang membantu Anda mengingat tujuan asosiasi.
5. Di daftar Dokumen, pilih **`AWS-ApplyChefRecipes`**.
6. Di Parameter, untuk Jenis Sumber, pilih Git GitHub, HTTP, atau S3.
7. Untuk info Sumber, masukkan informasi sumber buku masak menggunakan format yang sesuai untuk Jenis Sumber yang Anda pilih pada langkah 6. Untuk informasi selengkapnya, lihat topik berikut:

   - the section called “Gunakan Git sebagai sumber buku masak”
   - the section called “Gunakan GitHub sebagai sumber buku masak”
   - the section called “Gunakan HTTP sebagai sumber buku masak”
   - the section called “Menggunakan Amazon S3 sebagai sumber buku masak”
8. Di Jalankan daftar, buat daftar resep yang ingin Anda jalankan dalam format berikut, memisahkan setiap resep dengan koma seperti yang ditunjukkan. Jangan masukkan spasi setelah koma. Ganti masing-masing *`example-resource-placeholder`* dengan informasi Anda sendiri.

```
recipe[cookbook-name1::recipe-name],recipe[cookbook-name2::recipe-name]
```

9. (Opsional) Tentukan atribut JSON kustom yang Anda ingin Chef klien berikan ke node target Anda.

   a. Dalam konten atribut JSON, tambahkan atribut apa pun yang Anda ingin Chef klien berikan ke node target Anda.

   b. Di sumber atribut JSON, tambahkan jalur ke atribut apa pun yang Anda ingin Chef klien lewatkan ke node target Anda.

   Untuk informasi selengkapnya, lihat the section called “Menerapkan atribut JSON ke target saat menjalankan resep”.

10. Untuk versi Chef klien, tentukan Chef versi. Nilai yang valid adalah `11` melalui`18`, atau`None`. Jika Anda menentukan nomor antara `11` `18` (inklusif), Systems Manager menginstal versi Chef klien yang benar pada node target Anda. Jika Anda menentukan`None`, Systems Manager tidak menginstal Chef klien pada node target sebelum menjalankan resep dokumen.

11. (Opsional) Untuk argumen Chef klien, tentukan argumen tambahan yang didukung untuk versi yang Chef Anda gunakan. Untuk mempelajari lebih lanjut tentang argumen yang didukung, jalankan `chef-client -h` pada node yang menjalankan Chef klien.

12. (Opsional) Aktifkan Why-run untuk menampilkan perubahan yang dibuat pada node target jika resep dijalankan, tanpa benar-benar mengubah node target.

13. Untuk Keparahan kepatuhan, pilih keparahan dari hasil Kepatuhan Systems Manager yang ingin Anda laporkan. Pelaporan kepatuhan menunjukkan apakah status asosiasi sesuai atau tidak sesuai, bersama dengan tingkat keparahan yang Anda tentukan. Laporan kepatuhan disimpan di bucket S3 yang Anda tentukan sebagai nilai dari parameter Bucket laporan kepatuhan (langkah 14). Untuk informasi selengkapnya tentang Kepatuhan, lihat Bekerja dengan Kepatuhan dalam panduan ini.

    Pemindaian kepatuhan mengukur penyimpangan antara konfigurasi yang ditentukan dalam Chef resep dan sumber daya node Anda. Nilai yang valid adalah `Critical`, `High`, `Medium`, `Low`, `Informational`, `Unspecified`, atau `None`. Untuk melewati pelaporan kepatuhan, pilih `None`.

14. Untuk Jenis kepatuhan, tentukan jenis kepatuhan yang hasilnya ingin Anda laporkan. Nilai yang valid adalah `Association` untuk State Manager asosiasi, atau `Custom: tipe khusus`. Nilai default-nya adalah `Custom:Chef`.

15. Untuk bucket laporan Kepatuhan, masukkan nama bucket S3 untuk menyimpan informasi tentang setiap Chef proses yang dilakukan oleh dokumen ini, termasuk konfigurasi sumber daya dan hasil Kepatuhan.
16. Dalam Rate control, konfigurasikan opsi untuk menjalankan State Manager asosiasi di seluruh armada node terkelola. Untuk informasi tentang menggunakan kontrol rate, lihat Tentang target dan kontrol tingkat dalam State Manager asosiasi.

    Di Konkurensi, pilih opsi:

    - Pilih target untuk memasukkan jumlah absolut dari target yang dapat menjalankan asosiasi secara bersamaan.
    - Pilih persentase untuk memasukkan persentase dari kumpulan target yang dapat menjalankan asosiasi secara bersamaan.

    Di Batas kesalahan, pilih opsi:

    - Pilih kesalahan untuk memasukkan jumlah absolut kesalahan yang diizinkan sebelum State Manager berhenti menjalankan asosiasi pada target tambahan.
    - Pilih persentase untuk memasukkan persentase kesalahan yang diizinkan sebelum State Manager berhenti menjalankan asosiasi pada target tambahan.
17. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Aktifkan output penulisan ke S3. Masukkan nama bucket dan prefiks (folder) di dalam kotak.

    > **Note**
    >
    > Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin dari profil instance yang ditetapkan ke node terkelola, bukan izin pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbeda Akun AWS, verifikasi bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.
18. Pilih Buat Asosiasi.

Buat asosiasi yang menjalankan Chef resep (CLI)

Prosedur berikut menjelaskan cara menggunakan AWS Command Line Interface (AWS CLI) untuk membuat State Manager asosiasi yang menjalankan buku masak Chef dengan menggunakan `AWS-ApplyChefRecipes` dokumen.

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI.

2. Jalankan salah satu perintah berikut untuk membuat asosiasi yang menjalankan Chef buku masak pada node target yang memiliki tag yang ditentukan. Gunakan perintah yang sesuai untuk jenis sumber buku masak dan sistem operasi Anda. Ganti masing-masing *example-resource-placeholder* dengan informasi Anda sendiri.

   a. Sumber Git

      Linux & macOS

```
aws ssm create-association --name "AWS-ApplyChefRecipes" \
    --targets Key=tag:TagKey,Values=TagValue \
    --parameters '{"SourceType":["Git"],"SourceInfo":["{\"repository\":
\"repository-name\", \"getOptions\": \"branch:branch-name\", \"username
\": \"{{ ssm-secure:username-secure-string-parameter }}\", \"password\":
 \"{{ ssm-secure:password-secure-string-parameter }}\"}"], "RunList":
["{\"recipe[cookbook-name-1::recipe-name]\", \"recipe[cookbook-
name-2::recipe-name]\"}"], "JsonAttributesContent": ["{custom-json-
content}"], "JsonAttributesSources": "{\"sourceType\":\"s3\", \"sourceInfo
\":\"s3-bucket-endpoint-1\"}, {\"sourceType\":\"s3\", \"sourceInfo\":
\"s3-bucket-endpoint-2\"}", "ChefClientVersion": ["version-number"],
 "ChefClientArguments":["{chef-client-arguments}"], "WhyRun": boolean,
 "ComplianceSeverity": ["severity-value"], "ComplianceType":
 ["Custom:Chef"], "ComplianceReportBucket": ["s3-bucket-name"]}' \
    --association-name "name" \
    --schedule-expression "cron-or-rate-expression"
```

      Windows

```
aws ssm create-association --name "AWS-ApplyChefRecipes" ^
    --targets Key=tag:TagKey,Values=TagValue ^
```

```
    --parameters '{"SourceType":["Git"],"SourceInfo":["{\"repository\":
\"repository-name\", \"getOptions\": \"branch:branch-name\", \"username
\": \"{{ ssm-secure:username-secure-string-parameter }}\", \"password\":
 \"{{ ssm-secure:password-secure-string-parameter }}\"}"], "RunList":
["{\"recipe[cookbook-name-1::recipe-name]\", \"recipe[cookbook-
name-2::recipe-name]\"}"], "JsonAttributesContent": ["{custom-json}"],
 "JsonAttributesSources": "{\"sourceType\":\"s3\", \"sourceInfo\":
\"s3-bucket-endpoint-1\"}, {\"sourceType\":\"s3\", \"sourceInfo\":
\"s3-bucket-endpoint-2\"}", "ChefClientVersion": ["version-number"],
 "ChefClientArguments":["{chef-client-arguments}"], "WhyRun": boolean,
 "ComplianceSeverity": ["severity-value"], "ComplianceType":
 ["Custom:Chef"], "ComplianceReportBucket": ["s3-bucket-name"]}' ^
    --association-name "name" ^
    --schedule-expression "cron-or-rate-expression"
```

b. GitHub sumber

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"
```

Ini contohnya.

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. Sumber 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. Sumber Amazon S3

Linux & macOS

```
aws ssm create-association --name "AWS-ApplyChefRecipes" \
    --targets Key=tag:TagKey,Values=TagValue \
    --parameters '{"SourceType":["S3"],"SourceInfo":["{\"path\":\"https://
s3.amazonaws.com/path_to_Zip_file,_directory,_or_cookbook_to_download\"}"],
 "RunList":["{\"recipe[cookbook_name1::recipe_name]\",
 \"recipe[cookbook_name2::recipe_name]\"}"], "JsonAttributesContent":
 ["{Custom_JSON}"], "ChefClientVersion": ["version_number"],
 "ChefClientArguments":["{chef_client_arguments}"], "WhyRun": true_or_false,
 "ComplianceSeverity": ["severity_value"], "ComplianceType":
 ["Custom:Chef"], "ComplianceReportBucket": ["DOC-EXAMPLE-BUCKET"]}' \
    --association-name "name" \
    --schedule-expression "cron_or_rate_expression"
```

Windows

```
aws ssm create-association --name "AWS-ApplyChefRecipes" ^
    --targets Key=tag:TagKey,Values=TagValue ^
    --parameters '{"SourceType":["S3"],"SourceInfo":["{\"path\":\"https://
s3.amazonaws.com/path_to_Zip_file,_directory,_or_cookbook_to_download\"}"],
 "RunList":["{\"recipe[cookbook_name1::recipe_name]\",
 \"recipe[cookbook_name2::recipe_name]\"}"], "JsonAttributesContent":
 ["{Custom_JSON}"], "ChefClientVersion": ["version_number"],
 "ChefClientArguments":["{chef_client_arguments}"], "WhyRun": true_or_false,
 "ComplianceSeverity": ["severity_value"], "ComplianceType":
 ["Custom:Chef"], "ComplianceReportBucket": ["DOC-EXAMPLE-BUCKET"]}' ^
    --association-name "name" ^
    --schedule-expression "cron_or_rate_expression"
```

Ini contohnya.

Linux & macOS

```
aws ssm create-association --name "AWS-ApplyChefRecipes" \
    --targets "Key=tag:OS,Values= Linux" \
    --parameters '{"SourceType":["S3"],"SourceInfo":["{\"path
\":\"https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/HelloWorld
\"}"], "RunList":["{\"recipe[HelloWorld::HelloWorldRecipe]\",
 \"recipe[HelloWorld::InstallApp]\"}"], "JsonAttributesContent":
```

```
 ["{\"state\": \"visible\",\"colors\": {\"foreground\": \"light-blue
\",\"background\": \"dark-gray\"}}"], "ChefClientVersion": ["14"],
 "ChefClientArguments":["{--fips}"], "WhyRun": false, "ComplianceSeverity":
 ["Medium"], "ComplianceType": ["Custom:Chef"], "ComplianceReportBucket":
 ["ChefComplianceResultsBucket"]}' \
    --association-name "name" \
    --schedule-expression "cron(0 2 ? * SUN *)"
```

Windows

```
aws ssm create-association --name "AWS-ApplyChefRecipes" ^
    --targets "Key=tag:OS,Values= Linux" ^
    --parameters '{"SourceType":["S3"],"SourceInfo":["{\"path
\":\"https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/HelloWorld
\"}"], "RunList":["{\"recipe[HelloWorld::HelloWorldRecipe]\",
 \"recipe[HelloWorld::InstallApp]\"}"], "JsonAttributesContent":
 ["{\"state\": \"visible\",\"colors\": {\"foreground\": \"light-blue
\",\"background\": \"dark-gray\"}}"], "ChefClientVersion": ["14"],
 "ChefClientArguments":["{--fips}"], "WhyRun": false, "ComplianceSeverity":
 ["Medium"], "ComplianceType": ["Custom:Chef"], "ComplianceReportBucket":
 ["ChefComplianceResultsBucket"]}' ^
    --association-name "name" ^
    --schedule-expression "cron(0 2 ? * SUN *)"
```

Sistem membuat asosiasi, dan kecuali ekspresi cron atau rate yang Anda tentukan mencegahnya, sistem menjalankan asosiasi pada node target.

> **Note**
>
> State Managerasosiasi tidak mendukung semua ekspresi cron dan rate. Untuk informasi selengkapnya tentang membuat ekspresi cron dan rate untuk asosiasi, lihat Referensi: Ekspresi cron dan rate untuk Systems Manager.

3. Jalankan perintah berikut untuk melihat status asosiasi yang baru saja Anda buat.

```
aws ssm describe-association --association-id "ID"
```

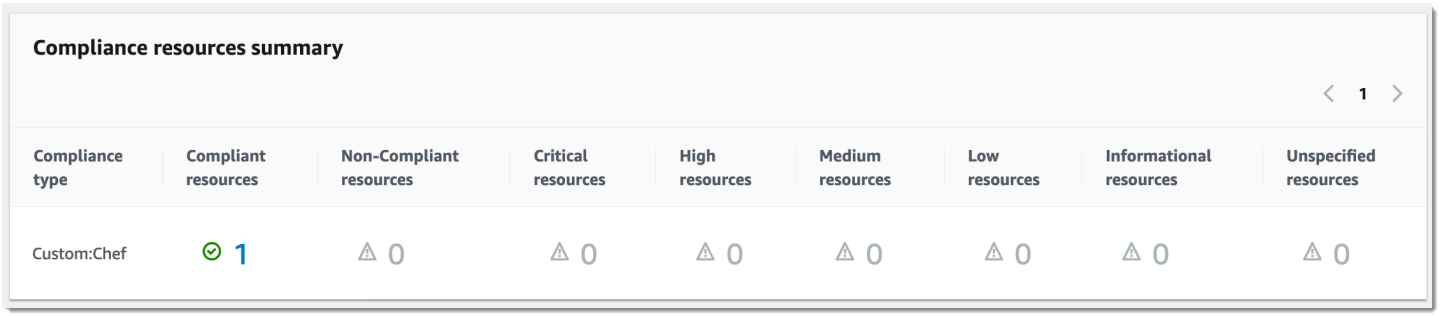
Melihat detail kepatuhan sumber daya Chef

Systems Manager menangkap informasi kepatuhan tentang sumber daya yang Chef dikelola dalam nilai bucket laporan Kepatuhan Amazon S3 yang ditentukan saat menjalankan dokumen. `AWS-ApplyChefRecipes` Mencari informasi tentang kegagalan Chef sumber daya dalam bucket S3 dapat memakan waktu. Sebaliknya, Anda dapat melihat informasi ini pada halaman Kepatuhan Systems Manager.

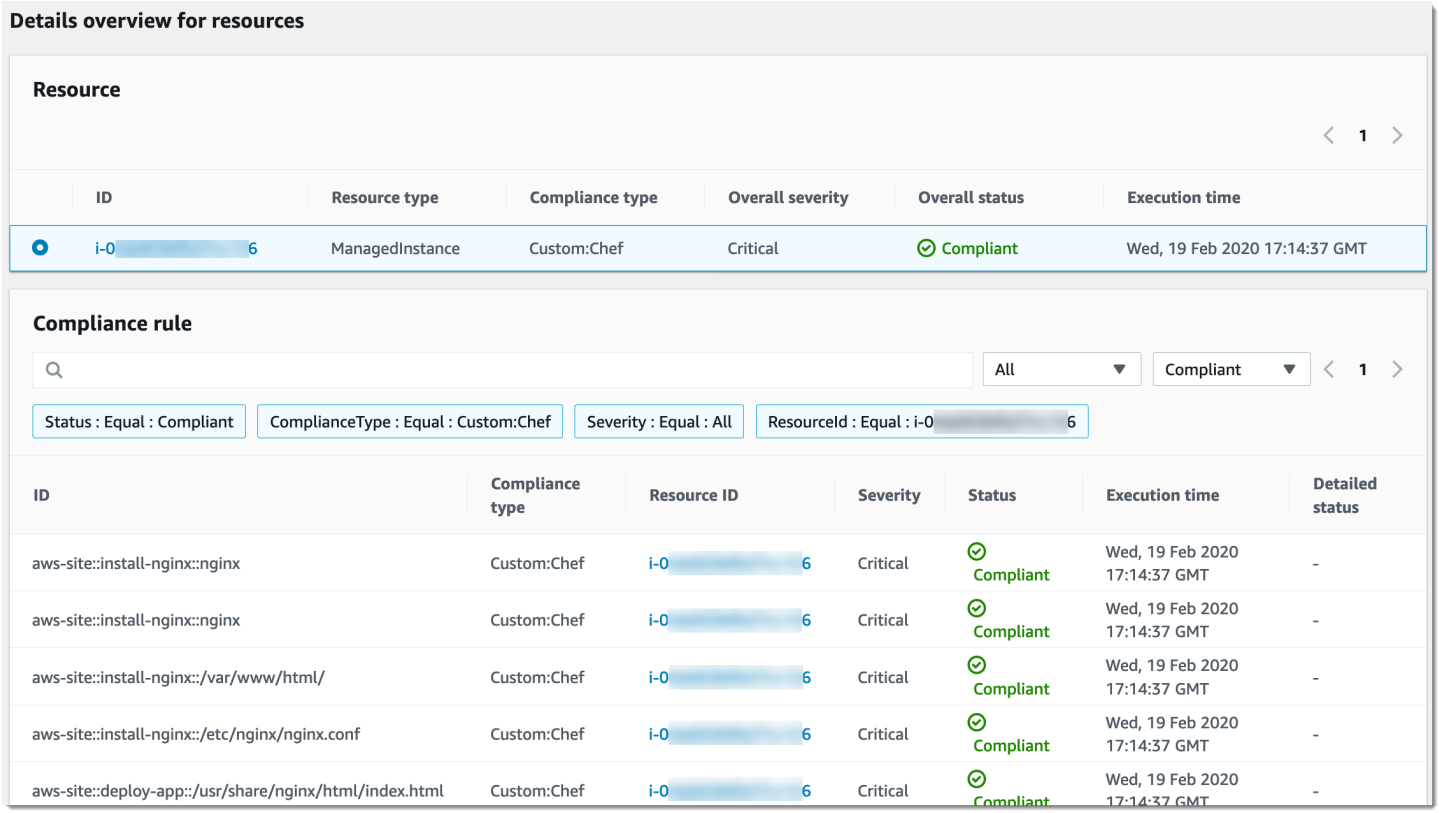
Pemindaian Kepatuhan Systems Manager mengumpulkan informasi tentang sumber daya pada node terkelola yang dibuat atau diperiksa dalam proses terbaruChef. Sumber daya dapat mencakup file, direktori, layanan `systemd`, paket `yum`, file templat, paket `gem`, dan buku masak dependen, di antaranya.

Bagian Ringkasan sumber daya kepatuhan menampilkan jumlah sumber daya yang gagal. Dalam contoh berikut, ComplianceTypeadalah Custom: Chef dan satu sumber daya tidak sesuai.

> **Note**
>
> `Custom:Chef`adalah ComplianceTypenilai default dalam `AWS-ApplyChefRecipes` dokumen. Nilai ini dapat disesuaikan.



Bagian Ikhtisar detail untuk sumber daya menunjukkan informasi tentang AWS sumber daya yang tidak sesuai. Bagian ini juga mencakup jenis Chef sumber daya yang digunakan untuk menjalankan kepatuhan, tingkat keparahan masalah, status kepatuhan, dan tautan ke informasi lebih lanjut bila berlaku.



Melihat output menunjukkan 4.000 karakter terakhir dari status terperinci. Systems Manager dimulai dengan pengecualian sebagai elemen pertama, menemukan pesan verbose, dan menunjukkannya sampai mencapai kuota karakter 4.000. Proses ini menampilkan pesan log yang dikeluarkan sebelum pengecualian diluncurkan, yang merupakan pesan yang paling relevan untuk pemecahan masalah.

Untuk informasi tentang cara melihat informasi kepatuhan, lihat AWS Systems Manager Kepatuhan.

Kegagalan asosiasi mempengaruhi pelaporan kepatuhan

Jika State Manager asosiasi gagal, tidak ada data kepatuhan yang dilaporkan. Misalnya, jika Systems Manager mencoba mengunduh Chef buku masak dari bucket S3 yang node tidak memiliki izin untuk mengaksesnya, asosiasi gagal, dan Systems Manager tidak melaporkan data kepatuhan.

Walkthrough: Perbarui secara otomatis (SSM AgentCLI)

Prosedur berikut memandu Anda melalui proses menciptakan State Manager asosiasi menggunakan AWS Command Line Interface. Asosiasi secara otomatis memperbarui SSM Agent sesuai dengan jadwal yang Anda tentukan. Untuk informasi selengkapnya tentang SSM Agent, lihat Bekerja dengan SSM Agent. Untuk menyesuaikan jadwal pembaruan untuk SSM Agent menggunakan konsol, lihatMemperbarui secara otomatis SSM Agent.

Untuk diberi tahu tentang SSM Agent pembaruan, berlangganan halaman Catatan SSM Agent Rilis diGitHub.

Sebelum Anda mulai

Sebelum Anda menyelesaikan prosedur berikut, verifikasi bahwa Anda setidaknya memiliki satu instans Amazon Elastic Compute Cloud (Amazon EC2) untuk Linux, macOS, atau Windows Server yang dikonfigurasi untuk Systems Manager. Untuk informasi selengkapnya, lihat Menyiapkan AWS Systems Manager.

Jika Anda membuat asosiasi dengan menggunakan salah satu AWS CLI atau AWS Tools for Windows PowerShell, gunakan `--Targets` parameter untuk menargetkan instance, seperti yang ditunjukkan pada contoh berikut. Jangan gunakan `--InstanceID` parameter. Parameter `--InstanceID` adalah parameter warisan.

Untuk membuat asosiasi untuk memperbarui secara otomatis SSM Agent

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI.
2. Jalankan perintah berikut untuk membuat asosiasi dengan menargetkan instans menggunakan tag Amazon Elastic Compute Cloud (Amazon EC2). Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri. Parameter `Schedule` menetapkan jadwal untuk menjalankan asosiasi setiap Minggu pagi pukul 2:00 pagi. (UTC).

   State Managerasosiasi tidak mendukung semua ekspresi cron dan rate. Untuk informasi selengkapnya tentang membuat ekspresi cron dan rate untuk asosiasi, lihat Referensi: Ekspresi cron dan rate untuk 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 *)"
```

Anda dapat menargetkan beberapa instance dengan menentukan ID instance dalam daftar yang dipisahkan koma.

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 *)"
```

Anda dapat menentukan versi yang ingin SSM Agent Anda perbarui.

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
```

Sistem mengembalikan informasi seperti berikut.

```
{
    "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"
            }
        ]
    }
}
```

Sistem mencoba untuk membuat asosiasi pada instans dan menerapkan status setelah pembuatan. Status asosiasi menunjukkan `Pending`.

3. Jalankan perintah berikut untuk menampilkan status terbaru dari asosiasi yang Anda buat.

```
aws ssm list-associations
```

Jika instans Anda tidak menjalankan versi terbaruSSM Agent, status akan ditampilkan`Failed`. Ketika versi baru diterbitkan, asosiasi secara otomatis menginstal agen baru, dan status akan ditampilkan`Success`. SSM Agent

Panduan: Secara otomatis memperbarui driver PV pada instans EC2 untuk Windows Server (konsol)

Amazon Windows Amazon Machine Images (AMIs) berisi seperangkat driver untuk mengizinkan akses ke perangkat keras virtual. Driver ini digunakan oleh Amazon Elastic Compute Cloud (Amazon EC2) untuk memetakan penyimpanan instans dan volume Amazon Elastic Block Store (Amazon EBS) ke perangkat mereka. Kami menyarankan Anda menginstal driver terbaru untuk meningkatkan stabilitas dan performa instans EC2 untuk Windows Server. Untuk informasi selengkapnya tentang driver PV, lihat Driver PV AWS.

Panduan berikut menunjukkan kepada Anda cara mengonfigurasi State Manager asosiasi untuk mengunduh dan menginstal driver AWS PV baru secara otomatis saat driver tersedia. State Manageradalah kemampuanAWS Systems Manager.

Sebelum Anda memulai

Sebelum Anda menyelesaikan prosedur berikut, verifikasi bahwa Anda memiliki setidaknya satu instans Amazon EC2 untuk Windows Server dijalankan yang dikonfigurasi untuk Systems Manager. Untuk informasi selengkapnya, lihat Menyiapkan AWS Systems Manager.

Untuk membuat State Manager asosiasi yang secara otomatis memperbarui driver PV

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih State Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih State Manager.
3. Pilih Buat asosiasi.
4. Di bidang Nama, masukkan nama deskriptif untuk asosiasi.
5. Di daftar Dokumen, pilih `AWS-ConfigureAWSPackage`.
6. Di area Parameter, lakukan hal berikut:
   - Untuk Tindakan, pilih Instal.
   - Untuk Jenis penginstalan, pilih Hapus instalan dan instal ulang.

> **Note**
>
> Upgrade di tempat tidak didukung untuk paket ini. Itu harus dihapus dan diinstal ulang.

- Untuk Nama, masukkan **`AWSPVDriver`**.

Anda tidak perlu memasukkan apa pun untuk Versi dan Argumen Tambahan.

7. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

> **Tip**
>
> Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

> **Note**
>
> Jika Anda memilih untuk menargetkan instans dengan menggunakan tanda, dan Anda menentukan tanda yang memetakan ke instans Linux, kaitan berhasil pada instans Windows tetapi gagal pada instans Linux. Status keseluruhan asosiasi menunjukkan Gagal.

8. Di area Tentukan jadwal, pilih apakah akan menjalankan asosiasi pada jadwal yang Anda konfigurasikan, atau hanya sekali. Driver PV yang diperbarui dirilis beberapa kali dalam setahun, sehingga Anda dapat menjadwalkan asosiasi untuk dijalankan sebulan sekali, jika Anda mau.
9. Di area Opsi lanjutan, untuk tingkat keparahan Kepatuhan, pilih tingkat keparahan untuk asosiasi. Pelaporan kepatuhan menunjukkan apakah status asosiasi sesuai atau tidak, bersama dengan tingkat keparahan yang Anda tunjukkan di sini. Untuk informasi selengkapnya, lihat Tentang kepatuhan State Manager asosiasi.
10. Untuk Pengendalian rate:

- Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

> **Note**
>
> Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

- Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

11. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Aktifkan output penulisan ke S3. Masukkan nama bucket dan prefiks (folder) di dalam kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin dari profil instance yang ditetapkan ke node terkelola, bukan izin pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbedaAkun AWS, verifikasi bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

12. (Opsional) Di bagian CloudWatch alarm, untuk nama Alarm, pilih CloudWatch alarm untuk diterapkan ke asosiasi Anda untuk pemantauan.

> **Note**
>
> Perhatikan informasi berikut tentang langkah ini.
>
> - Daftar alarm menampilkan maksimal 100 alarm. Jika Anda tidak melihat alarm dalam daftar, gunakan tombol AWS Command Line Interface untuk membuat asosiasi. Untuk informasi selengkapnya, lihat Membuat asosiasi (baris perintah).
> - Untuk melampirkan CloudWatch alarm ke perintah Anda, kepala sekolah IAM yang membuat asosiasi harus memiliki izin untuk `iam:createServiceLinkedRole`

tindakan tersebut. Untuk informasi selengkapnya tentang CloudWatch alarm, lihat Menggunakan CloudWatch alarm Amazon.

- Jika alarm Anda aktif, pemanggilan atau otomatisasi perintah yang tertunda tidak berjalan.

13. Pilih Buat asosiasi, lalu pilih Tutup. Sistem ini mencoba untuk membuat asosiasi pada instans dan segera menerapkan status.

    Jika Anda membuat asosiasi pada satu atau lebih instans Amazon EC2 untuk Windows Server, status berubah ke Sukses. Jika instans Anda tidak dikonfigurasi untuk Systems Manager, atau jika Anda secara tidak sengaja menargetkan instans Linux, status akan menampilkan Gagal.

    Jika statusnya Gagal, pilih ID asosiasi, pilih tab Sumber Daya, lalu verifikasi bahwa asosiasi berhasil dibuat pada instans EC2 Anda untuk Windows Server. Jika instans EC2 untuk Windows Server menampilkan status Gagal, verifikasi bahwa instans berjalan pada instance, dan verifikasi bahwa instance dikonfigurasi dengan peran AWS Identity and Access Management (IAM) untuk Systems Manager. SSM Agent Lihat informasi yang lebih lengkap di Menyiapkan AWS Systems Manager.

AWS Systems Manager Patch Manager

Patch Manager, suatu kemampuanAWS Systems Manager, mengotomatisasi proses patching node terkelola dengan pembaruan terkait dan jenis pembaruan lainnya.

> **Important**
>
> Mulai 22 Desember 2022, Systems Manager menyediakan dukungan untuk kebijakan patch, yang merupakan metode baru dan direkomendasikan untuk mengonfigurasi operasi patching Anda. Dengan menggunakan konfigurasi kebijakan tambalan tunggal, Anda dapat menentukan patching untuk semua akun di semua Wilayah di organisasi Anda, hanya untuk akun dan Wilayah yang Anda pilih, atau untuk satu pasangan Wilayah Akun. Untuk informasi selengkapnya, lihat Menggunakan kebijakan Quick Setup tambalan.

Anda dapat menggunakanPatch Manager untuk menerapkan untuk menerapkan patch untuk kedua sistem operasi dan aplikasi. (Pada Windows Server, dukungan aplikasi terbatas pada pembaruan untuk aplikasi yang dirilis oleh Microsoft.) Anda dapat menggunakanPatch Manager untuk memasang

Service pada node Windows dan melakukan pemutakhiran versi minor pada node Linux. Anda dapat melakukan patch pada armada instans Amazon Elastic Compute Cloud (Amazon EC2), perangkat edge, server on-premise, dan mesin virtual (VM) berdasarkan jenis sistem operasi. Ini termasuk versi yang didukung dari beberapa sistem operasi, seperti yang tercantum dalamPrasyarat Patch Manager. Anda dapat memindai instans untuk melihat laporan patch yang hilang saja, atau Anda dapat memindai dan secara otomatis menginstal semua patch yang hilang. Untuk memulaiPatch Manager, buka konsol Systems Manager. Di panel navigasi, pilih Patch Manager.

> **Note**
>
> AWStidak menguji tambalan sebelum membuatnya tersedia diPatch Manager. SelainPatch Manager itu, tidak mendukung peningkatan versi utama sistem operasi, sepertiWindows Server 2016 hinggaWindows Server 2019, atauSUSE Linux Enterprise Server (SLES) 12.0 hinggaSLES 15.0.
>
> Untuk jenis sistem operasi berbasis Linux yang melaporkan tingkat keparahan patch,Patch Manager gunakan tingkat keparahan yang dilaporkan oleh penerbit perangkat lunak untuk pemberitahuan pembaruan atau patch individual. Patch Managertidak memperoleh tingkat keparahan dari sumber pihak ketiga, seperti Common Vulnerability Scoring System (CVSS), atau dari metrik yang dirilis oleh National Vulnerability Database (NVD).

Garis dasar patch

Patch Managermenggunakan dasar, yang mencakup aturan untuk persetujuan patch dalam beberapa hari sejak rilis, selain daftar opsional untuk disetujui dan ditolak dari daftar patch yang disetujui dan ditolak. Ketika operasi patching berjalan,Patch Manager bandingkan patch yang saat ini diterapkan ke node yang dikelola dengan yang harus diterapkan sesuai dengan aturan yang ditetapkan di dasar patch. Anda dapat memilihPatch Manager untuk menampilkan hanya laporan patch yang hilang (`Scan`operasi), atau Anda dapat memilihPatch Manager untuk secara otomatis menginstal semua patch yang ditemukan hilang dari node yang dikelola (`Scan and install`operasi).

Menambal metode operasi

Patch Managersaat ini menawarkan empat metode untuk menjalankan`Scan` dan`Scan and install` operasi:

- (Disarankan) Kebijakan tambalan yang dikonfigurasi diQuick Setup — Berdasarkan integrasi denganAWS Organizations, kebijakan patch tunggal dapat menentukan jadwal patch dan

menambal garis dasar untuk seluruh organisasi, termasuk beberapaAkun AWS dan semua akunWilayah AWS tersebut beroperasi. Kebijakan patch juga hanya dapat menargetkan beberapa unit organisasi (oU) dalam suatu organisasi. Anda dapat menggunakan kebijakan tambalan tunggal untuk memindai dan menginstal pada jadwal yang berbeda. Untuk informasi selengkapnya, lihat Patch Managerkonfigurasi penambalan organisasi dan Menggunakan kebijakan Quick Setup tambalan.

- Opsi Manajemen Host yang dikonfigurasi diQuick Setup - Konfigurasi Manajemen Host juga didukung oleh integrasi denganAWS Organizations, sehingga memungkinkan untuk menjalankan operasi penambalan hingga seluruh Organisasi. Namun, opsi ini terbatas pada pemindaian patch yang hilang menggunakan garis dasar patch default saat ini dan memberikan hasil dalam laporan kepatuhan. Metode operasi ini tidak dapat menginstal patch. Untuk informasi selengkapnya, lihat Manajemen host Amazon EC2.
- Jendela pemeliharaan untuk menjalankan tambalan**`Scan`** atau**`Install`** tugas - Jendela pemeliharaan, yang Anda siapkan dalam kemampuan Systems Manager yang dipanggilMaintenance Windows, dapat dikonfigurasi untuk menjalankan berbagai jenis tugas pada jadwal yang Anda tentukan. TugasRun Command -type dapat digunakan untuk menjalankan`Scan` atau`Scan and install` mengerjakan serangkaian node terkelola yang Anda pilih. Setiap tugas jendela pemeliharaan dapat menargetkan node yang dikelola hanya dalam satuAkun AWSWilayah AWS pasangan. Untuk informasi selengkapnya, lihat Walkthrough: Membuat jendela pemeliharaan untuk patching (konsol).
- Operasi 'Patch now' sesuai permintaan diPatch Manager - Opsi Patch now memungkinkan Anda melewati pengaturan jadwal saat Anda perlu menambal node yang dikelola secepat mungkin. Menggunakan Patch sekarang, Anda menentukan apakah akan menjalankan`Scan` atau`Scan and install` operasi dan node yang dikelola untuk menjalankan operasi. Anda juga dapat memilih untuk menjalankan dokumen Systems Manager (dokumen SSM) sebagai pengait siklus hidup selama menambal operasi penambalan. Setiap operasi Patch sekarang dapat menargetkan node yang dikelola hanya dalam satuAkun AWSWilayah AWS pasangan. Untuk informasi selengkapnya, lihat Menambal node terkelola sesuai permintaan.

Pelaporan kepatuhan

SetelahScan operasi, Anda dapat menggunakan konsol Systems Manager untuk melihat informasi tentang node terkelola mana yang tidak sesuai dengan patch, dan patch mana yang hilang dari masing-masing node tersebut. Anda juga dapat membuat laporan kepatuhan dalam format.csv yang dikirim ke bucket Amazon Simple Storage Service (Amazon S3) yang Anda pilih. Anda dapat membuat laporan satu kali, atau membuat laporan pada jadwal rutin. Untuk satu node terkelola,

laporan mencakup rincian semua patch untuk node. Untuk laporan pada semua node terkelola, hanya ringkasan tentang berapa banyak patch yang hilang disediakan. Setelah laporan dibuat, Anda dapat menggunakan alat seperti Amazon QuickSight untuk mengimpor dan menganalisis data. Untuk informasi selengkapnya, lihat Mengerjakan laporan kepatuhan patch.

> **Note**
>
> Item kepatuhan yang dihasilkan melalui penggunaan kebijakan tambalan memiliki jenis eksekusi`PatchPolicy`. Item kepatuhan yang tidak dihasilkan dalam operasi kebijakan tambalan memiliki jenis eksekusi`Command`.

Integrasi

Patch Managerterintegrasi dengan yang lain berikutLayanan AWS:

- AWS Identity and Access Management(IAM) - Gunakan IAM untuk mengontrol pengguna, grup, dan peran mana yang memiliki akses kePatch Manager operasi. Untuk informasi selengkapnya, lihatCara kerja AWS Systems Manager dengan IAM dan Konfigurasikan izin instans untuk Systems Manager.
- AWS CloudTrail- Gunakan CloudTrail untuk merekam riwayat peristiwa operasi patching yang dapat diaudit yang diprakarsai oleh pengguna, peran, atau grup. Untuk informasi selengkapnya, lihat Pencatatan panggilan AWS Systems Manager API dengan AWS CloudTrail.
- AWS Security Hub- Data kepatuhan dariPatch ManagerAWS Security Hub. Security Hub memberi Anda pandangan komprehensif tentang pemberitahuan keamanan prioritas tinggi dan status kepatuhan Anda. Hub juga memantau status patching armada Anda. Untuk informasi selengkapnya, lihat Integrasi dengan Patch ManagerAWS Security Hub.
- AWS Config- Siapkan perekamanAWS Config untuk melihat data manajemen instans Amazon EC2 diPatch Manager Dasbor. Untuk informasi selengkapnya, lihat Melihat ringkasan Patch Dasbor.

Topik

- Menggunakan kebijakan Quick Setup tambalan
- Prasyarat Patch Manager
- BagaimanaPatch Manager operasi bekerja
- Tentang dokumen SSM untuk patching node terkelola
- Tentang dasar patch

- MenggunakanKernel Live Patchingdi node terkelola Amazon Linux 2
- Bekerja denganPatch Manager (konsol)
- Bekerja denganPatch Manager(AWS CLI)
- Tutorial AWS Systems ManagerPatch Manager
- Pemecahan Masalah Patch Manager

Menggunakan kebijakan Quick Setup tambalan

Mulai 22 Desember 2022, Patch Manager menawarkan metode baru yang direkomendasikan untuk mengonfigurasi penambalan untuk organisasi Anda dan Akun AWS melalui penggunaan kebijakan tambalan.

Kebijakan tambalan adalah konfigurasi yang Anda atur menggunakanQuick Setup, kemampuan AWS Systems Manager. Kebijakan tambalan memberikan kontrol yang lebih luas dan lebih terpusat atas operasi patching Anda daripada yang tersedia dengan metode konfigurasi patching sebelumnya. Kebijakan patch dapat digunakan dengan semua sistem operasi yang didukung oleh Patch Manager, termasuk versi Linux yang didukungmacOS, danWindows Server. Untuk informasi tentang membuat kebijakan tambalan, lihatPatch Managerkonfigurasi penambalan organisasi.

Fitur utama kebijakan tambalan

Alih-alih menggunakan metode lain untuk menambal node Anda, gunakan kebijakan tambalan untuk memanfaatkan fitur-fitur utama ini:

- Penyiapan tunggal — Menyiapkan operasi penambalan menggunakan jendela pemeliharaan atau State Manager asosiasi dapat memerlukan banyak tugas di berbagai bagian konsol Systems Manager. Menggunakan kebijakan tambalan, semua operasi penambalan Anda dapat diatur dalam satu wizard.
- Dukungan multi-akun/multi-wilayah — Menggunakan jendela pemeliharaan, State Manager asosiasi, atau fitur Patch now diPatch Manager, Anda dibatasi untuk menargetkan node terkelola dalam satu pasangan. Akun AWS Wilayah AWS Jika Anda menggunakan beberapa akun dan beberapa Wilayah, tugas penyiapan dan pemeliharaan Anda dapat memerlukan banyak waktu, karena Anda harus melakukan tugas penyiapan di setiap pasangan Account-region. Namun, jika Anda menggunakannya AWS Organizations, Anda dapat menyiapkan satu kebijakan tambalan yang berlaku untuk semua node terkelola Wilayah AWS di semua node Anda Akun AWS. Atau, jika Anda memilih, kebijakan tambalan hanya dapat diterapkan pada beberapa unit organisasi (OU) di

akun dan Wilayah yang Anda pilih. Kebijakan tambalan juga dapat berlaku untuk satu akun lokal, jika Anda mau.

- Dukungan instalasi di tingkat organisasi — Opsi konfigurasi Manajemen Host yang ada di Quick Setup menyediakan dukungan untuk pemindaian harian node terkelola Anda untuk kepatuhan patch. Namun, pemindaian ini dilakukan pada waktu yang telah ditentukan dan hanya menghasilkan informasi kepatuhan tambalan. Tidak ada instalasi patch yang dilakukan. Menggunakan kebijakan tambalan, Anda dapat menentukan jadwal yang berbeda untuk pemindaian dan pemasangan. Anda juga dapat memilih frekuensi dan waktu operasi ini dengan menggunakan ekspresi CRON atau Rate kustom. Misalnya, Anda dapat memindai tambalan yang hilang setiap hari untuk memberi Anda informasi kepatuhan yang diperbarui secara berkala. Tapi, jadwal instalasi Anda bisa hanya seminggu sekali untuk menghindari downtime yang tidak diinginkan.
- Pemilihan baseline patch yang disederhanakan — Kebijakan patch masih menggabungkan baseline patch, dan tidak ada perubahan pada cara baseline patch dikonfigurasi. Namun, saat Anda membuat atau memperbarui kebijakan tambalan, Anda dapat memilih baseline AWS terkelola atau kustom yang ingin Anda gunakan untuk setiap jenis sistem operasi (OS) dalam satu daftar. Tidak perlu menentukan baseline default untuk setiap jenis OS dalam tugas terpisah.

> **Note**
>
> Saat menambal operasi berdasarkan kebijakan patch dijalankan, mereka menggunakan dokumen `AWS-RunPatchBaseline` SSM. Untuk informasi selengkapnya, lihat Tentang dokumen SSM `AWS-RunPatchBaseline`.

Informasi terkait

Terapkan operasi patching secara terpusat di seluruh AWS Organisasi Anda menggunakan Systems Manager Quick Setup (AWS Cloud Operations and Migrations Blog)

Perbedaan lain dengan kebijakan tambalan

Berikut adalah beberapa perbedaan lain yang perlu diperhatikan saat menggunakan kebijakan tambalan alih-alih metode konfigurasi tambalan sebelumnya:

- Tidak diperlukan grup tambalan — Dalam operasi penambalan sebelumnya, Anda dapat menandai beberapa node untuk menjadi milik grup tambalan, dan kemudian menentukan garis dasar

tambalan yang akan digunakan untuk grup tambalan itu. Jika tidak ada grup tambalan yang ditentukan, instance yang Patch Manager ditambal dengan baseline patch default saat ini untuk jenis OS. Menggunakan kebijakan tambalan, tidak perlu lagi menyiapkan dan memelihara grup tambalan.

- Halaman 'Configure patching' dihapus — Sebelum rilis kebijakan patch, Anda dapat menentukan default node mana yang akan ditambal, jadwal patching, dan operasi patching pada halaman Patching Configure. Halaman ini telah dihapus dariPatch Manager. Opsi ini sekarang ditentukan dalam kebijakan tambalan.
- Tidak ada dukungan 'Patch now' — Kemampuan untuk menambal node sesuai permintaan masih terbatas pada satu Akun AWS Wilayah AWS pasangan pada satu waktu. Untuk informasi, lihat Menambal node terkelola sesuai permintaan.
- Kebijakan tambalan dan informasi kepatuhan — Saat node terkelola dipindai untuk kepatuhan sesuai dengan konfigurasi kebijakan penambalan, data kepatuhan akan tersedia untuk Anda. Anda dapat melihat dan bekerja dengan data dengan cara yang sama seperti metode pemindaian kepatuhan lainnya. Meskipun Anda dapat menyiapkan kebijakan tambalan untuk seluruh organisasi atau beberapa unit organisasi, informasi kepatuhan dilaporkan secara individual untuk setiap Akun AWS Wilayah AWS pasangan. Untuk informasi selengkapnya, lihat Mengerjakan laporan kepatuhan patch.
- Status kepatuhan asosiasi dan kebijakan tambalan — Status patching untuk node terkelola yang berada di bawah kebijakan Quick Setup tambalan cocok dengan status eksekusi State Manager asosiasi untuk node tersebut. Jika status eksekusi asosiasi adalah`Compliant`, status patching untuk node terkelola juga ditandai`Compliant`. Jika status eksekusi asosiasi adalah`Non-Compliant`, status patching untuk node terkelola juga ditandai`Non-Compliant`.

Wilayah AWS didukung untuk kebijakan tambalan

Konfigurasi kebijakan tambalan di Quick Setup saat ini didukung di Wilayah berikut:

- AS Timur (Ohio) (us-east-2)
- AS Timur (Virginia Utara) (us-east-1)
- AS Barat (California Utara) (us-west-1)
- AS Barat (Oregon) (us-west-2)
- Asia Pacific (Mumbai) (ap-south-1)
- Asia Pacific (Seoul) (ap-northeast-2)
- Asia Pasifik (Singapura) (ap-southeast-1)

- Asia Pacific (Sydney) (ap-southeast-2)
- Asia Pacific (Tokyo) (ap-northeast-1)
- Kanada (Pusat) (ca-central-1)
- Eropa (Frankfurt) (eu-central-1)
- Eropa (Irlandia) (eu-west-1)
- Eropa (London) (eu-west-2)
- Eropa (Paris) (eu-west-3)
- Eropa (Stockholm) (eu-north-1)
- Amerika Selatan (São Paulo) (sa-east-1)

Prasyarat Patch Manager

Pastikan bahwa Anda telah memenuhi prasyarat yang diperlukan sebelum menggunakanPatch Manager, kemampuan. AWS Systems Manager

Topik

- Versi SSM Agent
- Versi Python
- Konektivitas ke sumber patch
- Akses titik akhir S3
- Sistem operasi yang didukung untuk Patch Manager

Versi SSM Agent

Versi 2.0.834.0 atau yang lebih baru berjalan pada node SSM Agent terkelola yang ingin Anda kelola. Patch Manager

> **Note**
>
> Versi terbaru dirilis setiap kali kemampuan baru ditambahkan ke Systems Manager atau pembaruan dibuat untuk kemampuan yang ada. SSM Agent Gagal menggunakan agen versi terbaru dapat mencegah node terkelola Anda menggunakan berbagai kemampuan dan fitur Systems Manager. Untuk alasan itu, kami menyarankan Anda mengotomatiskan proses menjaga agar tetap SSM Agent up to date pada mesin Anda. Untuk informasi, lihat

Mengotomatiskan pembaruan ke SSM Agent. Berlangganan halaman Catatan SSM Agent Rilis GitHub untuk mendapatkan pemberitahuan tentang SSM Agent pembaruan.

Versi Python

Untuk macOS dan sebagian besar sistem operasi Linux (OS), Patch Manager saat ini mendukung Python versi 2.6 - 3.10. OS AlmaLinux, Debian ServerRaspberry Pi OS, dan Ubuntu Server OS memerlukan versi Python 3 yang didukung (3.0 - 3.10).

Konektivitas ke sumber patch

Jika node terkelola Anda tidak memiliki koneksi langsung ke Internet dan Anda menggunakan Amazon Virtual Private Cloud (Amazon VPC) dengan titik akhir VPC, Anda harus memastikan bahwa node memiliki akses ke repositori patch sumber (repo). Pada node Linux, pembaruan patch biasanya diunduh dari repo jarak jauh yang dikonfigurasi pada node. Oleh karena itu, node harus dapat terhubung ke repo sehingga penambalan dapat dilakukan. Untuk informasi selengkapnya, lihat Cara pemilihan patch keamanan.

Windows Servernode yang dikelola harus dapat terhubung ke Katalog Pembaruan Windows atau Layanan Pembaruan Server Windows (WSUS). Konfirmasikan bahwa node Anda memiliki konektivitas ke Katalog Pembaruan Microsoft melalui gateway internet, gateway NAT, atau instance NAT. Jika Anda menggunakan WSUS, konfirmasikan bahwa node memiliki konektivitas ke server WSUS di lingkungan Anda. Untuk informasi selengkapnya, lihat Masalah: node terkelola tidak memiliki akses ke Katalog Pembaruan Windows atau WSUS.

Akses titik akhir S3

Baik node terkelola Anda beroperasi di jaringan pribadi atau publik, tanpa akses ke bucket Amazon Simple Storage Service (Amazon S3) AWS terkelola yang diperlukan, operasi penambalan gagal. Untuk informasi tentang bucket S3 node terkelola Anda harus dapat mengakses, lihat SSM Agentkomunikasi dengan bucket S3 AWS terkelola dan. Langkah 2: Buat titik akhir VPC

Sistem operasi yang didukung untuk Patch Manager

Patch ManagerKemampuan ini tidak mendukung semua versi sistem operasi yang sama yang didukung oleh kemampuan Systems Manager lainnya. Misalnya, Patch Manager tidak mendukung CentOS 6.3 atau Raspberry Pi OS 8 (Jessie). (Untuk daftar lengkap sistem operasi yang didukung oleh Systems Manager, lihat Sistem operasi yang didukung untuk Systems Manager.) Oleh karena

itu, pastikan bahwa node terkelola yang Patch Manager ingin Anda gunakan menjalankan salah satu sistem operasi yang tercantum dalam tabel berikut.

Sistem operasi	Detail
Linux	• AlmaLinux 8.3—8.7, 9.0—9.2 • Amazon Linux 2012.03—2018.03 • Amazon Linux 2 versi 2.0 dan semua versi yang lebih baru • Amazon Linux 2022 • Amazon Linux 2023 • CentOS 6.5—7.9, 8.0—8.5 • CentOS Stream8 • Debian Server8.x, 9.x, 10.x, 11.x, dan 12.x • Oracle Linux7,5—8,7, 9,0 - 9,2 • Raspberry Pi OS(sebelumnya Raspbian) 9 (Stretch) • Red Hat Enterprise Linux(RHEL) 6.5—8.8, 9.0—9.2 • Rocky Linux8.4—8.7, 9.0—9.2 • SUSE Linux Enterprise Server(SLES) 12.0 dan kemudian 12. x versi; 15.0 - 15.5 • Ubuntu Server14.04 LTS, 16.04 LTS, 18.04 LTS, 20.04 LTS, 20.10 STR, 22.04 LTS, dan 23.04 **Note** Node terkelola yang dibuat dari Amazon Linux 1 AMI yang menggunakan proxy harus menjalankan versi Python `requests` modul saat ini untuk mendukung Patch Manager operasi. Untuk informasi selengkapnya,

Sistem operasi	Detail
	lihat Memutakhirkan modul permintaan Python di Amazon Linux 1 instance yang menggunakan server proxy.

Sistem operasi	Detail
macOS	11.3.1; 11.4—11.7 (Big Sur) 12.0—12.6 (Monterey) 13.0—13.5 (Ventura) 14.0 (Sonoma) macOSPembaruan OS Patch Managertidak mendukung pembaruan atau peningkatan sistem operasi (OS) untukmacOS, seperti dari 12.x ke 13.x atau 13.1 hingga 13.2. Untuk melakukan pembaruan versi OSmacOS, kami sarankan untuk menggunakan mekanisme peningkatan OS bawaan Apple. Untuk informasi selengkap nya, lihat Manajemen Perangkat di situs web Dokumentasi Pengembang Apple. Dukungan Homebrew Sistem manajemen paket perangkat lunak sumber terbuka Homebrew telah menghenti kan dukungan untuk macOS 10.14.x (Mojave) dan 10.15.x (Catalina). Akibatnya, operasi penambalan pada versi ini saat ini tidak didukung. Dukungan Wilayah macOStidak didukung sama sekali Wilayah AWS. Untuk informasi selengkapnya tentang dukungan Amazon EC2macOS, lihat instans Amazon EC2 Mac di Panduan Pengguna Amazon EC2 untuk Instans Linux. macOSperangkat tepi

Sistem operasi	Detail
	SSM Agentuntuk perangkat AWS IoT Greengrass inti tidak didukung padamacOS. Anda tidak dapat menggunakan Patch Manager untuk menambal perangkat macOS tepi.
Windows	Windows Server2008 hingga Windows Server 2022, termasuk versi R2. **Note** SSM Agentuntuk perangkat AWS IoT Greengrass inti tidak didukung pada Windows 10. Anda tidak dapat menggunakan Patch Manager untuk menambal perangkat Windows 10 edge. Per 14 Januari 2020, Windows Server 2008 tidak lagi didukung untuk pembaruan fitur atau keamanan dari Microsoft. Legacy Amazon Machine Images (AMIs) untuk Windows Server 2008 dan 2008 R2 masih menyertakan versi 2 dari SSM Agent prainstal, tetapi Systems Manager tidak lagi secara resmi mendukung versi 2008 dan tidak lagi memperbarui agen untuk versi ini. Windows Server Selain itu, SSM Agent versi 3 mungkin tidak kompatibel dengan semua operasi pada Windows Server 2008 dan 2008 R2. Versi final yang didukung secara resmi SSM Agent untuk versi Windows Server 2008 adalah 2.3.1644.0.

BagaimanaPatch Manager operasi bekerja

Bagian ini menyediakan detail teknis yang menjelaskan caraPatch Manager, suatu kemampuanAWS Systems Manager, menentukan patch apa yang akan diinstal dan cara menginstalnya pada setiap sistem operasi yang didukung. Untuk sistem operasi Linux, bagian ini juga menyediakan informasi tentang menentukan repositori sumber, dalam dasar patch kustom, untuk patch selain patch default yang dikonfigurasi pada sebuah node yang dikelola. Bagian ini juga menyediakan detail tentang cara aturan dasar patch bekerja pada distribusi sistem operasi Linux yang berbeda.

> **Note**
>
> Informasi dalam topik berikut berlaku apa pun metode atau jenis konfigurasi yang Anda gunakan untuk operasi patching Anda:
>
> - Kebijakan tambalan yang dikonfigurasi diQuick Setup
> - Opsi Manajemen Host yang dikonfigurasi diQuick Setup
> - Jendela pemeliharaan untuk menjalankan patch`Scan` atau`Install` tugas
> - Patch on-demand sekarang beroperasi

Topik

Bagaimana tanggal rilis paket dan tanggal pembaruan dihitung

> **Important**
>
> Informasi di halaman ini berlaku untuk sistem operasi (OS) Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022, dan Amazon Linux 2023 untuk instans Amazon Elastic Compute Cloud (Amazon EC2). Paket untuk jenis OS ini dibuat dan dikelola oleh Amazon Web Services. Bagaimana produsen sistem operasi lain mengelola paket dan repositori mereka

memengaruhi bagaimana tanggal rilis dan tanggal pembaruan mereka dihitung. Untuk OS selain Amazon Linux, Amazon Linux 2, Amazon Linux 2022, dan Amazon Linux 2023, seperti Red Hat Enterprise Linux (RHEL) dan SUSE Linux Enterprise Server (SLES), lihat dokumentasi pabrikan untuk informasi tentang bagaimana paket mereka diperbarui dan dipelihara.

Dalam pengaturan untuk baseline patch kustom yang Anda buat, untuk sebagian besar jenis OS, Anda dapat menentukan bahwa patch disetujui secara otomatis untuk instalasi setelah beberapa hari tertentu. AWS menyediakan beberapa baseline patch yang telah ditentukan yang mencakup tanggal persetujuan otomatis 7 hari.

Penundaan persetujuan otomatis adalah jumlah hari untuk menunggu setelah tambalan dirilis, sebelum tambalan secara otomatis disetujui untuk ditambal. Misalnya, Anda membuat aturan menggunakan `CriticalUpdates` klasifikasi dan mengonfigurasinya selama 7 hari penundaan persetujuan otomatis. Akibatnya, tambalan kritis baru dengan tanggal rilis atau tanggal pembaruan terakhir 7 Juli secara otomatis disetujui pada 14 Juli.

Untuk menghindari hasil yang tidak terduga dengan penundaan persetujuan otomatis di Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022, dan Amazon Linux 2023, penting untuk memahami bagaimana tanggal rilis dan tanggal pembaruan mereka dihitung.

Dalam kebanyakan kasus, waktu tunggu persetujuan otomatis sebelum tambalan diinstal dihitung dari `Updated Date` nilai dalam`updateinfo.xml`, bukan nilai. `Release Date` Berikut ini adalah detail penting tentang perhitungan tanggal ini:

- `Release Date`Ini adalah tanggal pemberitahuan dirilis. Ini tidak berarti paket tersebut harus tersedia di repositori terkait.
- `Update Date`Ini adalah tanggal terakhir pemberitahuan diperbarui. Pembaruan pemberitahuan dapat mewakili sesuatu yang sekecil pembaruan teks atau deskripsi. Ini tidak berarti paket dirilis sejak tanggal tersebut atau harus tersedia di repositori terkait.

  Ini berarti bahwa sebuah paket dapat memiliki `Update Date` nilai 7 Juli tetapi tidak tersedia untuk instalasi sampai (misalnya) 13 Juli. Misalkan untuk kasus ini bahwa baseline patch yang menentukan penundaan persetujuan otomatis 7 hari berjalan dalam operasi pada 14 Juli. `Install` Karena `Update Date` nilainya 7 hari sebelum tanggal berjalan, tambalan dan pembaruan dalam paket diinstal pada 14 Juli. Instalasi terjadi meskipun hanya 1 hari telah berlalu sejak paket tersedia untuk instalasi yang sebenarnya.

- Paket yang berisi sistem operasi atau patch aplikasi dapat diperbarui lebih dari satu kali setelah rilis awal.
- Sebuah paket dapat dirilis ke repositori AWS terkelola tetapi kemudian diputar kembali jika masalah kemudian ditemukan dengannya.

Dalam beberapa operasi penambalan, faktor-faktor ini mungkin tidak penting. Misalnya, jika baseline patch dikonfigurasi untuk menginstal patch dengan nilai keparahan `Low` dan`Medium`, dan klasifikasi, penundaan persetujuan otomatis apa pun mungkin berdampak kecil pada operasi Anda. `Recommended`

Namun, dalam kasus di mana waktu tambalan kritis atau tingkat keparahan tinggi lebih penting, Anda mungkin ingin melakukan kontrol lebih besar saat tambalan dipasang. Metode yang disarankan untuk melakukan ini adalah dengan menggunakan repositori sumber patch alternatif alih-alih repositori default untuk menambal operasi pada node yang dikelola.

Anda dapat menentukan repositori sumber patch alternatif ketika membuat dasar patch kustom. Di setiap dasar patch kustom, Anda dapat menentukan konfigurasi sumber patch hingga 20 versi sistem operasi Linux yang didukung. Lihat informasi yang lebih lengkap di Cara menentukan repositori sumber patch alternatif (Linux).

Cara pemilihan patch keamanan

Fokus utamaPatch Manager, kemampuan AWS Systems Manager, adalah menginstal pembaruan terkait keamanan sistem operasi pada node yang dikelola. Secara default, Patch Manager tidak menginstal semua tambalan yang tersedia, melainkan serangkaian tambalan yang lebih kecil yang berfokus pada keamanan.

Untuk jenis sistem operasi berbasis Linux yang melaporkan tingkat keparahan patch, Patch Manager gunakan tingkat keparahan yang dilaporkan oleh penerbit perangkat lunak untuk pemberitahuan pembaruan atau tambalan individual. Patch Managertidak memperoleh tingkat keparahan dari sumber pihak ketiga, seperti Common Vulnerability Scoring System (CVSS), atau dari metrik yang dirilis oleh National Vulnerability Database (NVD).

> **Note**
>
> Pada semua sistem berbasis Linux yang didukung olehPatch Manager, Anda dapat memilih repositori sumber berbeda yang dikonfigurasi untuk node terkelola, biasanya untuk

menginstal pembaruan nonsecurity. Untuk informasi, lihat Cara menentukan repositori sumber patch alternatif (Linux).

Sisa bagian ini menjelaskan bagaimana Patch Manager memilih patch keamanan untuk berbagai sistem operasi yang didukung.

Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022, and Amazon Linux 2023

Repositori yang telah dikonfigurasi sebelumnya ditangani secara berbeda di Amazon Linux 1 dan Amazon Linux 2 daripada di Amazon Linux 2022 dan Amazon Linux 2023.

Di Amazon Linux 1 dan Amazon Linux 2, layanan baseline patch Systems Manager menggunakan repositori yang telah dikonfigurasi sebelumnya pada node terkelola. Biasanya ada dua repositori (repo) yang telah dikonfigurasi sebelumnya pada sebuah node:

- ID repo: `amzn-main/latest`

  Nama repo: `amzn-main-Base`
- ID repo: `amzn-updates/latest`

  Nama repo: `amzn-updates-Base`

Instans Amazon Linux 2023 (AL2023) awalnya berisi pembaruan yang tersedia dalam versi AL2023 dan yang dipilih. AMI Secara default, instans AL2023 Anda tidak secara otomatis menerima pembaruan keamanan penting dan penting tambahan saat diluncurkan. Sebagai gantinya, dengan peningkatan deterministik melalui fitur repositori berversi di AL2023, yang diaktifkan secara default, Anda dapat menerapkan pembaruan berdasarkan jadwal yang memenuhi kebutuhan spesifik Anda. Untuk informasi selengkapnya, lihat Peningkatan deterministik melalui repositori berversi di Panduan Pengguna Amazon Linux 2023.

Di Amazon Linux 2022, repositori yang telah dikonfigurasi sebelumnya terkait dengan versi pembaruan paket yang terkunci. Ketika new Amazon Machine Images (AMIs) untuk Amazon Linux 2022 dirilis, mereka dikunci ke versi tertentu. Untuk pembaruan tambalan, Patch Manager ambil versi terkunci terbaru dari repositori pembaruan tambalan dan kemudian memperbarui paket pada node terkelola berdasarkan konten versi terkunci tersebut.

Pada AL2023, repositori yang telah dikonfigurasi adalah sebagai berikut:

- ID repo: `amazonlinux`

Nama repo: repositori Amazon Linux 2023

Di Amazon Linux 2022 (rilis pratinjau), repositori yang telah dikonfigurasi sebelumnya terkait dengan versi pembaruan paket yang terkunci. Ketika new Amazon Machine Images (AMIs) untuk Amazon Linux 2022 dirilis, mereka dikunci ke versi tertentu. Untuk pembaruan tambalan, Patch Manager ambil versi terkunci terbaru dari repositori pembaruan tambalan dan kemudian memperbarui paket pada node terkelola berdasarkan konten versi terkunci tersebut.

Di Amazon Linux 2022, repositori yang telah dikonfigurasi sebelumnya adalah sebagai berikut:

- ID repo: `amazonlinux`

  Nama repo: repositori Amazon Linux 2022

> **Note**
>
> Semua pembaruan diunduh dari repo jarak jauh yang dikonfigurasi pada node terkelola. Oleh karena itu, node harus memiliki akses keluar ke internet untuk terhubung ke repo sehingga penambalan dapat dilakukan.

Node terkelola Amazon Linux 1 dan Amazon Linux 2 menggunakan Yum sebagai manajer paket. Amazon Linux 2022 dan Amazon Linux 2023 menggunakan DNF sebagai pengelola paket.

Kedua manajer paket menggunakan konsep pemberitahuan pembaruan sebagai file bernama`updateinfo.xml`. Pemberitahuan pembaruan hanyalah sebuah kumpulan paket yang memperbaiki masalah tertentu. Semua paket yang ada dalam pemberitahuan pembaruan dianggap Keamanan olehPatch Manager. Paket individu tidak ditetapkan klasifikasi atau tingkat kepelikan. Untuk alasan ini, Patch Manager tetapkan atribut pemberitahuan pembaruan ke paket terkait.

> **Note**
>
> Jika Anda memilih kotak centang Sertakan pembaruan non-keamanan di halaman dasar Buat tambalan, maka paket yang tidak diklasifikasikan dalam `updateinfo.xml` file (atau paket yang berisi file tanpa nilai Klasifikasi, Tingkat Keparahan, dan Tanggal yang diformat dengan benar) dapat disertakan dalam daftar patch yang telah difilter

sebelumnya. Namun, agar patch dapat diterapkan, patch harus tetap memenuhi aturan dasar patch yang ditentukan pengguna.

CentOS and CentOS Aliran

Pada CentOS danCentOS Stream, layanan baseline patch Systems Manager menggunakan repositori (repo) yang telah dikonfigurasi sebelumnya pada node yang dikelola. Daftar berikut memberikan contoh untuk CentOS 8.2 fiktif Amazon Machine Image (AMI):

- ID repo: `example-centos-8.2-base`

  Nama repo: `Example CentOS-8.2 - Base`
- ID repo: `example-centos-8.2-extras`

  Nama repo: `Example CentOS-8.2 - Extras`
- ID repo: `example-centos-8.2-updates`

  Nama repo: `Example CentOS-8.2 - Updates`
- ID repo: `example-centos-8.x-examplerepo`

  Nama repo: `Example CentOS-8.x – Example Repo Packages`

> **Note**
>
> Semua pembaruan diunduh dari repo jarak jauh yang dikonfigurasi pada node terkelola. Oleh karena itu, node harus memiliki akses keluar ke internet untuk terhubung ke repo sehingga penambalan dapat dilakukan.

CentOS 6 dan 7 node yang dikelola menggunakan Yum sebagai manajer paket. CentOS 8 dan CentOS Stream node menggunakan DNF sebagai manajer paket. Kedua pengelola paket menggunakan konsep pemberitahuan pembaruan. Pemberitahuan pembaruan hanyalah sebuah kumpulan paket yang memperbaiki masalah tertentu.

Namun, CentOS dan repo CentOS Stream default tidak dikonfigurasi dengan pemberitahuan pembaruan. Ini berarti itu Patch Manager tidak mendeteksi paket pada CentOS dan CentOS Stream repo default. Patch ManagerUntuk memungkinkan memproses paket yang tidak

terkandung dalam pemberitahuan pembaruan, Anda harus mengaktifkan `EnableNonSecurity` bendera dalam aturan dasar tambalan.

> **Note**
>
> CentOS dan pemberitahuan CentOS Stream pembaruan didukung. Repo dengan pemberitahuan pembaruan dapat diunduh setelah peluncuran.

Debian Server and Raspberry Pi OS

Pada Debian Server dan Raspberry Pi OS (sebelumnya Raspbian), layanan baseline patch Systems Manager menggunakan repositori (repo) yang telah dikonfigurasi sebelumnya pada instance. Repo yang telah dikonfigurasi ini digunakan untuk menarik daftar terbaru dari pemutakhiran paket yang tersedia. Untuk ini, Systems Manager melakukan perintah setara `sudo apt-get update`.

Paket kemudian di-filter dari repo `debian-security` *`codename`*. Ini berarti bahwa pada setiap versiDebian Server, Patch Manager hanya mengidentifikasi peningkatan yang merupakan bagian dari repo terkait untuk versi tersebut, sebagai berikut:

- Debian Server8: `debian-security jessie`
- Debian Server9: `debian-security stretch`
- Debian Server10: `debian-security buster`
- Debian Server11: `debian-security bullseye`
- Debian Server12: `debian-security bookworm`

> **Note**
>
> Hanya pada Debian Server 8: Karena beberapa node terkelola Debian Server 8.* merujuk ke repositori paket usang (`jessie-backports`), Patch Manager melakukan langkah-langkah tambahan untuk memastikan bahwa operasi penambalan berhasil. Untuk informasi selengkapnya, lihat Cara menginstal patch.

Oracle Linux

PadaOracle Linux, layanan baseline patch Systems Manager menggunakan repositori (repo) yang telah dikonfigurasi sebelumnya pada node terkelola. Biasanya ada dua repo yang telah dikonfigurasi sebelumnya pada sebuah node.

Oracle Linux7:

- ID repo: `ol7_UEKR5/x86_64`

  Nama repo: `Latest Unbreakable Enterprise Kernel Release 5 for Oracle Linux 7Server (x86_64)`
- ID repo: `ol7_latest/x86_64`

  Nama repo: `Oracle Linux 7Server Latest (x86_64)`

Oracle Linux8:

- ID repo: `ol8_baseos_latest`

  Nama repo: `Oracle Linux 8 BaseOS Latest (x86_64)`
- ID repo: `ol8_appstream`

  Nama repo: `Oracle Linux 8 Application Stream (x86_64)`
- ID repo: `ol8_UEKR6`

  Nama repo: `Latest Unbreakable Enterprise Kernel Release 6 for Oracle Linux 8 (x86_64)`

Oracle Linux9:

- ID repo: `ol9_baseos_latest`

  Nama repo: `Oracle Linux 9 BaseOS Latest (x86_64)`
- ID repo: `ol9_appstream`

  Nama repo: `Oracle Linux 9 Application Stream Packages(x86_64)`
- ID repo: `ol9_UEKR7`

Nama repo: `Oracle Linux UEK Release 7 (x86_64)`

> **Note**
>
> Semua pembaruan diunduh dari repo jarak jauh yang dikonfigurasi pada node terkelola. Oleh karena itu, node harus memiliki akses keluar ke internet untuk terhubung ke repo sehingga penambalan dapat dilakukan.

Oracle Linuxnode terkelola menggunakan Yum sebagai manajer paket, dan Yum menggunakan konsep pemberitahuan pembaruan sebagai file bernama. `updateinfo.xml` Pemberitahuan pembaruan hanyalah sebuah kumpulan paket yang memperbaiki masalah tertentu. Paket individu tidak ditetapkan klasifikasi atau tingkat kepelikan. Untuk alasan ini, Patch Manager tetapkan atribut pemberitahuan pembaruan ke paket terkait dan menginstal paket berdasarkan filter Klasifikasi yang ditentukan dalam baseline patch.

> **Note**
>
> Jika Anda memilih kotak centang Sertakan pembaruan non-keamanan di halaman dasar Buat tambalan, maka paket yang tidak diklasifikasikan dalam `updateinfo.xml` file (atau paket yang berisi file tanpa nilai Klasifikasi, Tingkat Keparahan, dan Tanggal yang diformat dengan benar) dapat disertakan dalam daftar patch yang telah difilter sebelumnya. Namun, agar patch dapat diterapkan, patch harus tetap memenuhi aturan dasar patch yang ditentukan pengguna.

AlmaLinux, RHEL, and Linux Rocky

Pada AlmaLinux,Red Hat Enterprise Linux, dan Rocky Linux layanan baseline patch Systems Manager menggunakan repositori (repo) yang telah dikonfigurasi sebelumnya pada node terkelola. Biasanya ada tiga repo yang telah dikonfigurasi sebelumnya pada sebuah node.

Semua pembaruan diunduh dari repo jarak jauh yang dikonfigurasi pada node terkelola. Oleh karena itu, node harus memiliki akses keluar ke internet untuk terhubung ke repo sehingga penambalan dapat dilakukan.

> **Note**
>
> Jika Anda memilih kotak centang Sertakan pembaruan non-keamanan di halaman dasar Buat tambalan, maka paket yang tidak diklasifikasikan dalam `updateinfo.xml` file (atau paket yang berisi file tanpa nilai Klasifikasi, Tingkat Keparahan, dan Tanggal yang diformat dengan benar) dapat disertakan dalam daftar patch yang telah difilter sebelumnya. Namun, agar patch dapat diterapkan, patch harus tetap memenuhi aturan dasar patch yang ditentukan pengguna.

Red Hat Enterprise Linux7 node terkelola menggunakan Yum sebagai manajer paket. AlmaLinux, Red Hat Enterprise Linux 8, dan node Rocky Linux terkelola menggunakan DNF sebagai manajer paket. Kedua pengelola paket menggunakan konsep pemberitahuan pembaruan sebagai file bernama `updateinfo.xml`. Pemberitahuan pembaruan hanyalah sebuah kumpulan paket yang memperbaiki masalah tertentu. Paket individu tidak ditetapkan klasifikasi atau tingkat kepelikan. Untuk alasan ini, Patch Manager tetapkan atribut pemberitahuan pembaruan ke paket terkait dan menginstal paket berdasarkan filter Klasifikasi yang ditentukan dalam baseline patch.

RHEL7

> **Note**
>
> ID repo berikut dikaitkan dengan RHUI 2. RHUI 3 diluncurkan pada bulan Desember 2019 dan memperkenalkan skema penamaan yang berbeda untuk ID repositori Yum. Bergantung pada RHEL-7 tempat AMI Anda membuat node terkelola, Anda mungkin perlu memperbarui perintah Anda. Untuk informasi selengkapnya, lihat ID Repositori untuk RHEL 7 di AWS Telah Berubah di Portal Pelanggan Red Hat.

- ID repo: `rhui-REGION-client-config-server-7/x86_64`

  Nama repo: `Red Hat Update Infrastructure 2.0 Client Configuration Server 7`
- ID repo: `rhui-REGION-rhel-server-releases/7Server/x86_64`

  Nama repo: `Red Hat Enterprise Linux Server 7 (RPMs)`
- ID repo: `rhui-REGION-rhel-server-rh-common/7Server/x86_64`

Nama repo: `Red Hat Enterprise Linux Server 7 RH Common (RPMs)`

AlmaLinux, 8 RHEL 8, dan Rocky Linux 8

- ID repo: `rhel-8-appstream-rhui-rpms`

  Nama repo: `Red Hat Enterprise Linux 8 for x86_64 - AppStream from RHUI (RPMs)`

- ID repo: `rhel-8-baseos-rhui-rpms`

  Nama repo: `Red Hat Enterprise Linux 8 for x86_64 - BaseOS from RHUI (RPMs)`

- ID repo: `rhui-client-config-server-8`

  Nama repo: `Red Hat Update Infrastructure 3 Client Configuration Server 8`

AlmaLinux 9, RHEL 9, dan Rocky Linux 9

- ID repo: `rhel-9-appstream-rhui-rpms`

  Nama repo: `Red Hat Enterprise Linux 9 for x86_64 - AppStream from RHUI (RPMs)`

- ID repo: `rhel-9-baseos-rhui-rpms`

  Nama repo: `Red Hat Enterprise Linux 9 for x86_64 - BaseOS from RHUI (RPMs)`

- ID repo: `rhui-client-config-server-9`

  Nama repo: `Red Hat Enterprise Linux 9 Client Configuration`

SLES

Pada node terkelola SUSE Linux Enterprise Server (SLES), pustaka ZYPP mendapatkan daftar tambalan yang tersedia (kumpulan paket) dari lokasi berikut:

- Daftar repositori: `etc/zypp/repos.d/*`
- Informasi paket: `/var/cache/zypp/raw/*`

SLESnode terkelola menggunakan Zypper sebagai manajer paket, dan Zypper menggunakan konsep tambalan. Patch hanyalah kumpulan paket yang memperbaiki masalah tertentu. Patch Managermenangani semua paket yang direferensikan dalam tambalan sebagai terkait keamanan. Karena paket individual tidak diberi klasifikasi atau tingkat keparahan, Patch Manager berikan paket atribut tambalan yang menjadi miliknya.

Ubuntu Server

PadaUbuntu Server, layanan baseline patch Systems Manager menggunakan repositori (repo) yang telah dikonfigurasi sebelumnya pada node terkelola. Repo yang telah dikonfigurasi ini digunakan untuk menarik daftar terbaru dari pemutakhiran paket yang tersedia. Untuk ini, Systems Manager melakukan perintah setara `sudo apt-get update`.

Paket kemudian disaring dari *`codename`*`-security` repo, di mana nama kode unik untuk versi rilis, seperti `trusty` untuk 14. Ubuntu Server Patch Managerhanya mengidentifikasi peningkatan yang merupakan bagian dari repo ini:

- Ubuntu Server14.04 LTS: `trusty-security`
- Ubuntu Server16.04 LTS: `xenial-security`
- Ubuntu Server18.04 LTS: `bionic-security`
- Ubuntu Server20.04 LTS: `focal-security`
- Ubuntu Server20.10 STR: `groovy-security`
- Ubuntu Server22,04 LTS () `jammy-security`
- Ubuntu Server23.04 () `lunar-security`

Windows Server

Pada sistem operasi Microsoft Windows, Patch Manager mengambil daftar pembaruan yang tersedia yang diterbitkan Microsoft ke Microsoft Update dan secara otomatis tersedia untuk Windows Server Update Services (WSUS).

Patch Managerterus memantau pembaruan baru di setiap Wilayah AWS. Daftar pembaruan yang tersedia disegarkan di setiap Region setidaknya sekali per hari. Ketika informasi tambalan dari Microsoft diproses, Patch Manager menghapus pembaruan yang digantikan oleh pembaruan selanjutnya dari daftar tambalannya. Oleh karena itu, hanya pembaruan terbaru yang ditampilkan dan tersedia untuk instalasi. Misalnya, jika `KB4012214` menggantikan`KB3135456`, hanya `KB4012214` tersedia sebagai pembaruan diPatch Manager.

Patch Managerhanya membuat patch yang tersedia untuk versi sistem Windows Server operasi yang didukung untukPatch Manager. Misalnya, tidak Patch Manager dapat digunakan untuk menambal Windows RT.

> **Note**
>
> Dalam beberapa kasus, Microsoft merilis patch untuk aplikasi yang tidak menentukan tanggal dan waktu yang diperbarui. Dalam kasus ini, tanggal dan waktu yang diperbarui `01/01/1970` disediakan secara default.

Cara menentukan repositori sumber patch alternatif (Linux)

Bila Anda menggunakan repositori default yang dikonfigurasi pada node terkelola untuk operasi patching, kemampuan Patch Manager AWS Systems Manager, memindai atau menginstal patch terkait keamanan. Ini adalah perilaku default untukPatch Manager. Untuk informasi selengkapnya tentang cara Patch Manager memilih dan menginstal patch keamanan, lihat. Cara pemilihan patch keamanan

Namun, pada sistem Linux, Anda juga dapat menggunakan Patch Manager untuk menginstal tambalan yang tidak terkait dengan keamanan, atau yang berada di repositori sumber yang berbeda dari yang default yang dikonfigurasi pada node terkelola. Anda dapat menentukan repositori sumber patch alternatif ketika membuat dasar patch kustom. Di setiap dasar patch kustom, Anda dapat menentukan konfigurasi sumber patch hingga 20 versi sistem operasi Linux yang didukung.

Misalnya, misalkan Ubuntu Server armada Anda menyertakan node terkelola Ubuntu Server 14,04 dan Ubuntu Server 16,04. Dalam kasus ini, Anda dapat menentukan repositori alternatif untuk setiap versi dalam dasar patch kustom yang sama. Untuk setiap versi, Anda memberikan nama, menentukan jenis versi sistem operasi (produk), dan menyediakan konfigurasi repositori. Anda juga dapat menentukan satu repositori sumber alternatif yang berlaku untuk semua versi sistem operasi yang didukung.

> **Note**
>
> Menjalankan baseline patch khusus yang menentukan repositori patch alternatif untuk node terkelola tidak menjadikannya repositori default baru pada sistem operasi. Setelah operasi patching selesai, repositori yang sebelumnya dikonfigurasi sebagai default untuk sistem operasi node tetap default.

Untuk daftar contoh skenario penggunaan opsi ini, lihat Contoh penggunaan untuk repositori sumber patch alternatif yang tersedia nanti dalam topik ini.

Untuk informasi tentang dasar patch default dan kustom, lihat Tentang dasar patch yang telah ditetapkan dan kustom.

Contoh: Menggunakan konsol

Untuk menentukan repositori sumber patch alternatif ketika Anda bekerja di konsol Systems Manager, gunakan bagian Sumber patch pada halaman Buat dasar patch. Untuk informasi tentang penggunaan opsi Sumber patch, lihat Membuat dasar patch kustom (Linux).

Contoh: Menggunakan AWS CLI

Untuk contoh penggunaan opsi `--sources` dengan AWS Command Line Interface (AWS CLI), lihat Buat dasar patch dengan repositori kustom untuk versi OS yang berbeda.

Topik

- Pertimbangan penting untuk repositori alternatif
- Contoh penggunaan untuk repositori sumber patch alternatif

Pertimbangan penting untuk repositori alternatif

Ingatlah poin-poin berikut saat Anda merencanakan strategi patching Anda menggunakan repositori patch alternatif.

Hanya repositori yang ditentukan yang digunakan untuk patching

Menentukan repositori alternatif tidak berarti menentukan repositori tambahan. Anda dapat memilih untuk menentukan repositori selain yang dikonfigurasi sebagai default pada node terkelola. Namun, Anda juga harus menentukan repositori default sebagai bagian dari konfigurasi sumber patch alternatif jika Anda ingin pembaruan mereka diterapkan.

Misalnya, pada node terkelola Amazon Linux 2, repositori default adalah `amzn2-core` dan. `amzn2extra-docker` Jika Anda ingin menyertakan repositori Extra Packages for Enterprise Linux (EPEL) di operasi patching Anda, Anda harus menentukan ketiga repositori tersebut sebagai repositori alternatif.

> **Note**
>
> Menjalankan baseline patch khusus yang menentukan repositori patch alternatif untuk node terkelola tidak menjadikannya repositori default baru pada sistem operasi. Setelah operasi patching selesai, repositori yang sebelumnya dikonfigurasi sebagai default untuk sistem operasi node tetap default.

Perilaku patching untuk distribusi berbasis YUM bergantung pada manifes updateinfo.xml

Saat Anda menentukan repositori patch alternatif untuk distribusi berbasis Yum, seperti Amazon Linux 1 atau Amazon Linux 2, Red Hat Enterprise Linux atau CentOS, perilaku patching bergantung pada apakah repositori menyertakan manifes pembaruan dalam bentuk file yang lengkap dan diformat dengan benar. `updateinfo.xml` file ini menentukan tanggal rilis, klasifikasi, dan tingkat kepelikan dari berbagai paket. Salah satu dari hal berikut ini akan mempengaruhi perilaku patching:

- Jika Anda memfilter Klasifikasi dan Tingkat kepelikan, tetapi keduanya tidak ditentukan dalam `updateinfo.xml`, paket tidak akan disertakan oleh filter. Ini juga berarti bahwa paket tanpa file `updateinfo.xml` tidak akan disertakan dalam patching.
- Jika Anda memfilter ApprovalAfterDays, tetapi tanggal rilis paket tidak dalam format Unix Epoch (atau tidak memiliki tanggal rilis yang ditentukan), paket tidak akan disertakan oleh filter.
- Ada pengecualian jika Anda memilih kotak centang Sertakan pembaruan non-keamanan di halaman dasar Buat tambalan. Dalam hal ini, paket tanpa file `updateinfo.xml` (atau yang berisi file ini tanpa format Klasifikasi, Tingkat kepelikan, dan nilai Tanggal yang benar) akan disertakan ke dalam daftar patch pra-filter. (Mereka tetap harus memenuhi persyaratan aturan dasar patch lainnya agar dapat diinstal.)

Contoh penggunaan untuk repositori sumber patch alternatif

Contoh 1 - Pembaruan Nonsecurity untuk Ubuntu Server

Anda sudah menggunakan Patch Manager untuk menginstal patch keamanan pada armada node Ubuntu Server terkelola menggunakan baseline patch yang telah ditentukan AWS -provided. `AWS-UbuntuDefaultPatchBaseline` Anda dapat membuat dasar patch yang didasarkan pada default ini, tapi dalam aturan persetujuan Anda menentukan bahwa Anda ingin turut menginstal pembaruan non-keamanan yang merupakan bagian dari distribusi default. Ketika baseline patch ini dijalankan terhadap node Anda, patch untuk masalah keamanan dan nonsecurity diterapkan. Anda juga dapat

memilih untuk menyetujui patch non-keamanan di pengecualian patch yang Anda tentukan untuk dasar.

Contoh 2 - Personal Package Archives (PPA) untuk Ubuntu Server

Node Ubuntu Server terkelola Anda menjalankan perangkat lunak yang didistribusikan melalui Personal Package Archives (PPA) untuk Ubuntu. Dalam hal ini, Anda membuat baseline patch yang menentukan repositori PPA yang telah Anda konfigurasikan pada node terkelola sebagai repositori sumber untuk operasi patching. Kemudian gunakan Run Command untuk menjalankan dokumen baseline patch pada node.

Contoh 3 – Aplikasi Perusahaan Internal di Amazon Linux

Anda perlu menjalankan beberapa aplikasi yang diperlukan untuk kepatuhan peraturan industri pada node yang dikelola Amazon Linux Anda. Anda dapat mengkonfigurasi repositori untuk aplikasi ini pada node, menggunakan YUM untuk menginstal aplikasi pada awalnya, dan kemudian memperbarui atau membuat baseline patch baru untuk memasukkan repositori perusahaan baru ini. Setelah ini Anda dapat menggunakan Run Command untuk menjalankan `AWS-RunPatchBaseline` dokumen dengan `Scan` opsi untuk melihat apakah paket perusahaan terdaftar di antara paket yang diinstal dan up to date pada node yang dikelola. Jika belum diperbarui, Anda dapat menjalankan dokumen lagi menggunakan opsi `Install` untuk memperbarui aplikasi.

Cara menginstal patch

Patch Manager, kemampuan AWS Systems Manager, menggunakan mekanisme bawaan yang sesuai untuk jenis sistem operasi untuk menginstal pembaruan pada node terkelola. Misalnya, padaWindows Server, Windows Update API digunakan, dan di Amazon Linux 2 manajer `yum` paket digunakan.

Sisa bagian ini menjelaskan bagaimana Patch Manager menginstal patch pada sistem operasi.

Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022, and Amazon Linux 2023

Di node yang dikelola Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022, dan Amazon Linux 2023, alur kerja penginstalan tambalan adalah sebagai berikut:

1. Jika daftar patch ditentukan menggunakan URL https atau URL ala jalur Amazon Simple Storage Service (Amazon S3) menggunakan parameter `InstallOverrideList` untuk dokumen `AWS-RunPatchBaseline` atau `AWS-RunPatchBaselineAssociation`, patch yang terdaftar diinstal dan langkah 2-7 dilewati.

2. Terapkan GlobalFiltersseperti yang ditentukan dalam baseline patch, hanya menyimpan paket yang memenuhi syarat untuk diproses lebih lanjut.
3. Terapkan ApprovalRulesseperti yang ditentukan dalam baseline patch. Setiap aturan persetujuan dapat menentukan paket sebagai disetujui.

   Namun, aturan persetujuan, juga tunduk pada apakah kontak centang Sertakan pembaruan non-keamanan dipilih saat membuat atau terakhir memperbarui dasar patch.

   Jika pembaruan non-keamanan dikecualikan, diterapkan suatu aturan implisit untuk memilih hanya paket dengan pemutakhiran dalam repo keamanan. Untuk setiap paket, versi kandidat paket (yang biasanya versi terbaru) harus merupakan bagian dari repo keamanan.

   Jika pembaruan non-keamanan disertakan, patch dari repositori lain juga dipertimbangkan.
4. Terapkan ApprovedPatchesseperti yang ditentukan dalam baseline patch. Tambalan yang disetujui disetujui untuk diperbarui meskipun dibuang oleh GlobalFiltersatau jika tidak ada aturan persetujuan yang ditentukan dalam ApprovalRulesmemberikan persetujuan.
5. Terapkan RejectedPatchesseperti yang ditentukan dalam baseline patch. Patch yang ditolak akan dihapus dari daftar patch yang disetujui dan tidak akan diterapkan.
6. Jika beberapa versi patch disetujui, versi yang terbaru diterapkan.
7. API pembaruan YUM (Amazon Linux 1, Amazon Linux 2) atau API pembaruan DNF (Amazon Linux 2022, Amazon Linux 2023) diterapkan ke tambalan yang disetujui sebagai berikut:
   - Untuk garis dasar patch default yang telah ditentukan yang disediakan oleh AWS, hanya tambalan yang ditentukan dalam yang `updateinfo.xml` diterapkan (hanya pembaruan keamanan). Ini karena kotak centang Sertakan pembaruan nonkeamanan tidak dipilih. Garis dasar yang telah ditentukan setara dengan baseline kustom dengan yang berikut:
     - Kotak centang Sertakan pembaruan nonkeamanan tidak dipilih
     - Daftar KEPARAHAN `[Critical, Important]`
     - Daftar KLASIFIKASI `[Security, Bugfix]`

     Untuk Amazon Linux 1 dan Amazon Linux 2, perintah yum yang setara untuk alur kerja ini adalah:

     ```
     sudo yum update-minimal --sec-severity=Critical,Important --bugfix -y
     ```

     Untuk Amazon Linux 2022 dan Amazon Linux 2023, perintah dnf yang setara untuk alur kerja ini adalah:

```
sudo dnf upgrade-minimal --sec-severity=Critical --sec-severity=Important --
bugfix -y
```

Jika kotak centang Sertakan pembaruan nonkeamanan dipilih, semua tambalan `updateinfo.xml` dan yang tidak masuk `updateinfo.xml` akan diterapkan (pembaruan keamanan dan nonkeamanan).

Untuk Amazon Linux 1 dan Amazon Linux 2, jika baseline dengan Include pembaruan nonsecurity dipilih, memiliki daftar KEPARAHAN `[Critical, Important]` dan daftar KLASIFIKASI`[Security, Bugfix]`, perintah yum yang setara adalah:

```
sudo yum update --security --sec-severity=Critical,Important --bugfix -y
```

Untuk Amazon Linux 2022, dan Amazon Linux 2023, perintah dnf yang setara adalah:

```
sudo dnf upgrade --security --sec-severity=Critical --sec-severity=Important --
bugfix -y
```

> **Note**
>
> Untuk Amazon Linux 2022 dan Amazon Linux 2023, tingkat keparahan patch `Medium` setara dengan tingkat keparahan `Moderate` yang mungkin ditentukan di beberapa repositori eksternal. Jika Anda menyertakan patch `Medium` keparahan di baseline patch, patch `Moderate` keparahan dari patch eksternal juga diinstal pada instance.
>
> Saat Anda menanyakan data kepatuhan menggunakan tindakan API DescribeInstancePatches, pemfilteran untuk tingkat keparahan akan `Medium` melaporkan tambalan dengan tingkat keparahan keduanya `Medium` dan. `Moderate` Amazon Linux 2022 dan Amazon Linux 2023 juga mendukung tingkat keparahan patchNone, yang diakui oleh manajer paket DNF.

8. Node terkelola di-boot ulang jika ada pembaruan yang diinstal. (Pengecualian: Jika `RebootOption` parameter disetel ke `NoReboot` dalam `AWS-RunPatchBaseline` dokumen, node terkelola tidak di-boot ulang setelah Patch Manager dijalankan. Untuk informasi lebih lanjut, lihatNama parameter: `RebootOption`.)

CentOS and CentOS Aliran

Pada CentOS dan node CentOS Stream terkelola, alur kerja instalasi patch adalah sebagai berikut:

1. Jika daftar patch ditentukan menggunakan URL https atau URL ala jalur Amazon Simple Storage Service (Amazon S3) menggunakan parameter `InstallOverrideList` untuk dokumen `AWS-RunPatchBaseline` atau `AWS-RunPatchBaselineAssociation`, patch yang terdaftar diinstal dan langkah 2-7 dilewati.

   Terapkan GlobalFiltersseperti yang ditentukan dalam baseline patch, hanya menyimpan paket yang memenuhi syarat untuk diproses lebih lanjut.
2. Terapkan ApprovalRulesseperti yang ditentukan dalam baseline patch. Setiap aturan persetujuan dapat menentukan paket sebagai disetujui.

   Namun, aturan persetujuan, juga tunduk pada apakah kontak centang Sertakan pembaruan non-keamanan dipilih saat membuat atau terakhir memperbarui dasar patch.

   Jika pembaruan non-keamanan dikecualikan, diterapkan suatu aturan implisit untuk memilih hanya paket dengan pemutakhiran dalam repo keamanan. Untuk setiap paket, versi kandidat paket (yang biasanya versi terbaru) harus merupakan bagian dari repo keamanan.

   Jika pembaruan non-keamanan disertakan, patch dari repositori lain juga dipertimbangkan.
3. Terapkan ApprovedPatchesseperti yang ditentukan dalam baseline patch. Tambalan yang disetujui disetujui untuk diperbarui meskipun dibuang oleh GlobalFiltersatau jika tidak ada aturan persetujuan yang ditentukan dalam ApprovalRulesmemberikan persetujuan.
4. Terapkan RejectedPatchesseperti yang ditentukan dalam baseline patch. Patch yang ditolak akan dihapus dari daftar patch yang disetujui dan tidak akan diterapkan.
5. Jika beberapa versi patch disetujui, versi yang terbaru diterapkan.
6. API pembaruan YUM (pada versi CentOS 6.x dan 7.x) atau pembaruan DNF (pada CentOS CentOS Stream 8 dan) diterapkan ke tambalan yang disetujui.
7. Node terkelola di-boot ulang jika ada pembaruan yang diinstal. (Pengecualian: Jika `RebootOption` parameter disetel ke `NoReboot` dalam `AWS-RunPatchBaseline` dokumen, node terkelola tidak di-boot ulang setelah Patch Manager dijalankan. Untuk informasi lebih lanjut, lihatNama parameter: `RebootOption`.)

Debian Server and Raspberry Pi OS

Pada Debian Server dan Raspberry Pi OS (sebelumnya Raspbian) instance, alur kerja instalasi patch adalah sebagai berikut:

1. Jika daftar patch ditentukan menggunakan URL https atau URL ala jalur Amazon Simple Storage Service (Amazon S3) menggunakan parameter `InstallOverrideList` untuk dokumen `AWS-RunPatchBaseline` atau `AWS-RunPatchBaselineAssociation`, patch yang terdaftar diinstal dan langkah 2-7 dilewati.
2. Jika pembaruan tersedia untuk `python3-apt` (antarmuka pustaka Python `kelibapt`), itu ditingkatkan ke versi terbaru. (Paket nonsecurity ini ditingkatkan meskipun Anda tidak memilih opsi Sertakan pembaruan nonsecurity.)

   > **Important**
   >
   > Hanya pada Debian Server 8: Karena beberapa node terkelola Debian Server 8.* merujuk ke repositori paket usang (`jessie-backports`), Patch Manager lakukan langkah-langkah tambahan berikut untuk memastikan bahwa operasi penambalan berhasil:
   >
   > a. Pada node terkelola Anda, referensi ke `jessie-backports` repositori dikomentari dari daftar lokasi sumber (). `/etc/apt/sources.list.d/jessie-backports` Akibatnya, tidak ada upaya yang dilakukan untuk mengunduh patch dari lokasi tersebut.
   >
   > b. Kunci penandatanganan pembaruan keamanan Stretch diimpor. Kunci ini memberikan izin yang diperlukan untuk operasi pembaruan dan penginstalan pada distribusi Debian Server 8.*.
   >
   > c. `apt-get`Operasi dijalankan pada titik ini untuk memastikan bahwa versi terbaru diinstal sebelum proses penambalan dimulai. `python3-apt`
   >
   > d. Setelah proses instalasi selesai, referensi ke repositori `jessie-backports` dipulihkan dan kunci penandatanganan dihapus dari keyring sumber apt. Hal ini dilakukan untuk mempertahankan konfigurasi sistem seperti keadaan sebelum operasi patching.
   >
   > Lain kali Patch Manager memperbarui sistem, proses yang sama diulang.
3. Terapkan GlobalFilterseperti yang ditentukan dalam baseline patch, hanya menyimpan paket yang memenuhi syarat untuk diproses lebih lanjut.

4. Terapkan ApprovalRulesseperti yang ditentukan dalam baseline patch. Setiap aturan persetujuan dapat menentukan paket sebagai disetujui.

   > **Note**
   >
   > Karena tidak mungkin menentukan tanggal rilis paket pembaruan secara andalDebian Server, opsi persetujuan otomatis tidak didukung untuk sistem operasi ini.

   Namun, aturan persetujuan, juga tunduk pada apakah kontak centang Sertakan pembaruan non-keamanan dipilih saat membuat atau terakhir memperbarui dasar patch.

   Jika pembaruan non-keamanan dikecualikan, diterapkan suatu aturan implisit untuk memilih hanya paket dengan pemutakhiran dalam repo keamanan. Untuk setiap paket, versi kandidat paket (yang biasanya versi terbaru) harus merupakan bagian dari repo keamanan.

   Jika pembaruan non-keamanan disertakan, patch dari repositori lain juga dipertimbangkan.

   > **Note**
   >
   > Untuk Debian Server danRaspberry Pi OS, versi kandidat patch terbatas pada tambalan yang disertakan di `debian-security` dalamnya.

5. Terapkan ApprovedPatchesseperti yang ditentukan dalam baseline patch. Tambalan yang disetujui disetujui untuk diperbarui meskipun dibuang oleh GlobalFiltersatau jika tidak ada aturan persetujuan yang ditentukan dalam ApprovalRulesmemberikan persetujuan.
6. Terapkan RejectedPatchesseperti yang ditentukan dalam baseline patch. Patch yang ditolak akan dihapus dari daftar patch yang disetujui dan tidak akan diterapkan.
7. Pustaka APT digunakan untuk memutakhirkan paket.
8. Node terkelola di-boot ulang jika ada pembaruan yang diinstal. (Pengecualian: Jika `RebootOption` parameter disetel ke `NoReboot` dalam `AWS-RunPatchBaseline` dokumen, node terkelola tidak di-boot ulang setelah Patch Manager dijalankan. Untuk informasi lebih lanjut, lihatNama parameter: `RebootOption`.)

macOS

Pada node macOS terkelola, alur kerja instalasi patch adalah sebagai berikut:

1. Daftar properti `/Library/Receipts/InstallHistory.plist` adalah catatan perangkat lunak yang telah diinstal dan dimutakhirkan menggunakan pengelola paket `softwareupdate` dan `installer`. Menggunakan alat baris perintah `pkgutil` (untuk `installer`) dan pengelola paket `softwareupdate`, perintah CLI dijalankan untuk mengurai daftar ini.

   Untuk`installer`, respons terhadap perintah CLI mencakup`package name`,,`version`, `volumelocation`, dan `install-time` detail, tetapi hanya `package name` dan `version` digunakan oleh. Patch Manager

   Untuk `softwareupdate`, respon terhadap perintah CLI meliputi nama paket (`display name`), `version`, dan `date`, tetapi hanya nama paket dan versi yang digunakan oleh Patch Manager.

   Untuk Brew dan Brew Cask, Homebrew tidak mendukung perintahnya berjalan dalam kendali pengguna root. Akibatnya, Patch Manager kueri untuk dan menjalankan perintah Homebrew baik sebagai pemilik direktori Homebrew atau sebagai pengguna valid milik grup pemilik direktori Homebrew. Perintahnya mirip dengan `softwareupdate` dan `installer` dan dijalankan melalui sub-proses Python untuk mengumpulkan data paket, dan outputnya diurai untuk mengidentifikasi nama dan versi paket.
2. Terapkan GlobalFiltersseperti yang ditentukan dalam baseline patch, hanya menyimpan paket yang memenuhi syarat untuk diproses lebih lanjut.
3. Terapkan ApprovalRulesseperti yang ditentukan dalam baseline patch. Setiap aturan persetujuan dapat menentukan paket sebagai disetujui.
4. Terapkan ApprovedPatchesseperti yang ditentukan dalam baseline patch. Tambalan yang disetujui disetujui untuk diperbarui meskipun dibuang oleh GlobalFiltersatau jika tidak ada aturan persetujuan yang ditentukan dalam ApprovalRulesmemberikan persetujuan.
5. Terapkan RejectedPatchesseperti yang ditentukan dalam baseline patch. Patch yang ditolak akan dihapus dari daftar patch yang disetujui dan tidak akan diterapkan.
6. Jika beberapa versi patch disetujui, versi yang terbaru diterapkan.
7. Memanggil CLI paket yang sesuai pada node terkelola untuk memproses tambalan yang disetujui sebagai berikut:

> **Note**
>
> `installer` tidak memiliki fungsi untuk memeriksa dan menginstal pembaruan. Oleh karena ituinstaller, untuk, Patch Manager hanya melaporkan paket mana yang diinstal. Akibatnya, paket `installer` tidak pernah dilaporkan sebagai `Missing`.

- Untuk baseline patch default yang telah ditentukan sebelumnya yang disediakan oleh AWS, dan untuk baseline patch kustom di mana kotak centang Sertakan pembaruan non-keamanan tidak dipilih, hanya pembaruan keamanan yang diterapkan.
- Untuk garis dasar tambalan khusus di mana kotak centang Sertakan pembaruan non-keamanan dipilih, pembaruan keamanan dan nonkeamanan diterapkan.

8. Node terkelola di-boot ulang jika ada pembaruan yang diinstal. (Pengecualian: Jika `RebootOption` parameter disetel ke `NoReboot` dalam `AWS-RunPatchBaseline` dokumen, node terkelola tidak di-boot ulang setelah Patch Manager dijalankan. Untuk informasi lebih lanjut, lihatNama parameter: `RebootOption`.)

Oracle Linux

Pada node Oracle Linux terkelola, alur kerja instalasi patch adalah sebagai berikut:

1. Jika daftar patch ditentukan menggunakan URL https atau URL ala jalur Amazon Simple Storage Service (Amazon S3) menggunakan parameter `InstallOverrideList` untuk dokumen `AWS-RunPatchBaseline` atau `AWS-RunPatchBaselineAssociation`, patch yang terdaftar diinstal dan langkah 2-7 dilewati.
2. Terapkan GlobalFiltersseperti yang ditentukan dalam baseline patch, hanya menyimpan paket yang memenuhi syarat untuk diproses lebih lanjut.
3. Terapkan ApprovalRulesseperti yang ditentukan dalam baseline patch. Setiap aturan persetujuan dapat menentukan paket sebagai disetujui.

   Namun, aturan persetujuan, juga tunduk pada apakah kontak centang Sertakan pembaruan non-keamanan dipilih saat membuat atau terakhir memperbarui dasar patch.

   Jika pembaruan non-keamanan dikecualikan, diterapkan suatu aturan implisit untuk memilih hanya paket dengan pemutakhiran dalam repo keamanan. Untuk setiap paket, versi kandidat paket (yang biasanya versi terbaru) harus merupakan bagian dari repo keamanan.

Jika pembaruan non-keamanan disertakan, patch dari repositori lain juga dipertimbangkan.

4. Terapkan ApprovedPatchesseperti yang ditentukan dalam baseline patch. Tambalan yang disetujui disetujui untuk diperbarui meskipun dibuang oleh GlobalFiltersatau jika tidak ada aturan persetujuan yang ditentukan dalam ApprovalRulesmemberikan persetujuan.
5. Terapkan RejectedPatchesseperti yang ditentukan dalam baseline patch. Patch yang ditolak akan dihapus dari daftar patch yang disetujui dan tidak akan diterapkan.
6. Jika beberapa versi patch disetujui, versi yang terbaru diterapkan.
7. Pada node terkelola versi 7, API pembaruan YUM diterapkan ke tambalan yang disetujui sebagai berikut:
   - Untuk baseline patch default yang telah ditentukan yang disediakan oleh AWS, dan untuk baseline patch kustom di mana kotak centang Sertakan pembaruan non-keamanan tidak dipilih, hanya tambalan yang ditentukan dalam `updateinfo.xml` yang diterapkan (hanya pembaruan keamanan).

     Perintah setara yum untuk alur kerja ini adalah:

     ```
     sudo yum update-minimal --sec-severity=Important,Moderate --bugfix -y
     ```

   - Untuk baseline patch kustom di mana kotak centang Sertakan pembaruan non-keamanan dipilih, tambalan yang masuk `updateinfo.xml` dan yang tidak `updateinfo.xml` ada diterapkan (pembaruan keamanan dan nonkeamanan).

     Perintah setara yum untuk alur kerja ini adalah:

     ```
     sudo yum update --security --bugfix -y
     ```

     Pada node terkelola versi 8 dan 9, API pembaruan DNF diterapkan ke tambalan yang disetujui sebagai berikut:

     - Untuk baseline patch default yang telah ditentukan yang disediakan oleh AWS, dan untuk baseline patch kustom di mana kotak centang Sertakan pembaruan non-keamanan tidak dipilih, hanya tambalan yang ditentukan dalam `updateinfo.xml` yang diterapkan (hanya pembaruan keamanan).

       Perintah setara yum untuk alur kerja ini adalah:

```
sudo dnf upgrade-minimal --security --sec-severity Moderate --sec-severity
 Important
```

- Untuk baseline patch kustom di mana kotak centang Sertakan pembaruan non-keamanan dipilih, tambalan yang masuk `updateinfo.xml` dan yang tidak `updateinfo.xml` ada diterapkan (pembaruan keamanan dan nonkeamanan).

  Perintah setara yum untuk alur kerja ini adalah:

```
sudo dnf upgrade --security --bugfix
```

8. Node terkelola di-boot ulang jika ada pembaruan yang diinstal. (Pengecualian: Jika `RebootOption` parameter disetel ke `NoReboot` dalam `AWS-RunPatchBaseline` dokumen, node terkelola tidak di-boot ulang setelah Patch Manager dijalankan. Untuk informasi lebih lanjut, lihatNama parameter: `RebootOption`.)

AlmaLinux, RHEL, and Linux Rocky

Pada AlmaLinux,Red Hat Enterprise Linux, dan node Rocky Linux terkelola, alur kerja instalasi patch adalah sebagai berikut:

1. Jika daftar patch ditentukan menggunakan URL https atau URL ala jalur Amazon Simple Storage Service (Amazon S3) menggunakan parameter `InstallOverrideList` untuk dokumen `AWS-RunPatchBaseline` atau `AWS-RunPatchBaselineAssociation`, patch yang terdaftar diinstal dan langkah 2-7 dilewati.
2. Terapkan GlobalFiltersseperti yang ditentukan dalam baseline patch, hanya menyimpan paket yang memenuhi syarat untuk diproses lebih lanjut.
3. Terapkan ApprovalRulesseperti yang ditentukan dalam baseline patch. Setiap aturan persetujuan dapat menentukan paket sebagai disetujui.

   Namun, aturan persetujuan, juga tunduk pada apakah kontak centang Sertakan pembaruan non-keamanan dipilih saat membuat atau terakhir memperbarui dasar patch.

   Jika pembaruan non-keamanan dikecualikan, diterapkan suatu aturan implisit untuk memilih hanya paket dengan pemutakhiran dalam repo keamanan. Untuk setiap paket, versi kandidat paket (yang biasanya versi terbaru) harus merupakan bagian dari repo keamanan.

   Jika pembaruan non-keamanan disertakan, patch dari repositori lain juga dipertimbangkan.

4. Terapkan ApprovedPatchesseperti yang ditentukan dalam baseline patch. Tambalan yang disetujui disetujui untuk diperbarui meskipun dibuang oleh GlobalFiltersatau jika tidak ada aturan persetujuan yang ditentukan dalam ApprovalRulesmemberikan persetujuan.
5. Terapkan RejectedPatchesseperti yang ditentukan dalam baseline patch. Patch yang ditolak akan dihapus dari daftar patch yang disetujui dan tidak akan diterapkan.
6. Jika beberapa versi patch disetujui, versi yang terbaru diterapkan.
7. API pembaruan YUM (pada RHEL 7) atau API pembaruan DNF (pada AlmaLinux 8 dan 9, 8 dan 9, dan RHEL Rocky Linux 8 dan 9) diterapkan pada tambalan yang disetujui sebagai berikut:
   - Untuk baseline patch default yang telah ditentukan yang disediakan oleh AWS, dan untuk baseline patch kustom di mana kotak centang Sertakan pembaruan non-keamanan tidak dipilih, hanya tambalan yang ditentukan dalam `updateinfo.xml` yang diterapkan (hanya pembaruan keamanan).

     Untuk RHEL 7, perintah yum setara untuk alur kerja ini adalah:

     ```
     sudo yum update-minimal --sec-severity=Critical,Important --bugfix -y
     ```

     Untuk AlmaLinux, RHEL 8, danRocky Linux, perintah dnf yang setara untuk alur kerja ini adalah:

     ```
     sudo dnf update-minimal --sec-severity=Critical --bugfix -y ; \
     sudo dnf update-minimal --sec-severity=Important --bugfix -y
     ```

   - Untuk baseline patch kustom di mana kotak centang Sertakan pembaruan non-keamanan dipilih, tambalan yang masuk `updateinfo.xml` dan yang tidak `updateinfo.xml` ada diterapkan (pembaruan keamanan dan nonkeamanan).

     Untuk RHEL 7, perintah yum setara untuk alur kerja ini adalah:

     ```
     sudo yum update --security --bugfix -y
     ```

     Untuk AlmaLinux 8 dan 9, RHEL 8 dan 9, dan Rocky Linux 8 dan 9, perintah dnf setara untuk alur kerja ini adalah:

     ```
     sudo dnf update --security --bugfix -y
     ```

8. Node terkelola di-boot ulang jika ada pembaruan yang diinstal. (Pengecualian: Jika `RebootOption` parameter disetel ke `NoReboot` dalam `AWS-RunPatchBaseline` dokumen, node terkelola tidak di-boot ulang setelah Patch Manager dijalankan. Untuk informasi lebih lanjut, lihatNama parameter: `RebootOption`.)

SLES

Pada node terkelola SUSE Linux Enterprise Server (SLES), alur kerja instalasi patch adalah sebagai berikut:

1. Jika daftar patch ditentukan menggunakan URL https atau URL ala jalur Amazon Simple Storage Service (Amazon S3) menggunakan parameter `InstallOverrideList` untuk dokumen `AWS-RunPatchBaseline` atau `AWS-RunPatchBaselineAssociation`, patch yang terdaftar diinstal dan langkah 2-7 dilewati.
2. Terapkan GlobalFiltersseperti yang ditentukan dalam baseline patch, hanya menyimpan paket yang memenuhi syarat untuk diproses lebih lanjut.
3. Terapkan ApprovalRulesseperti yang ditentukan dalam baseline patch. Setiap aturan persetujuan dapat menentukan paket sebagai disetujui.

   Namun, aturan persetujuan, juga tunduk pada apakah kontak centang Sertakan pembaruan non-keamanan dipilih saat membuat atau terakhir memperbarui dasar patch.

   Jika pembaruan non-keamanan dikecualikan, diterapkan suatu aturan implisit untuk memilih hanya paket dengan pemutakhiran dalam repo keamanan. Untuk setiap paket, versi kandidat paket (yang biasanya versi terbaru) harus merupakan bagian dari repo keamanan.

   Jika pembaruan non-keamanan disertakan, patch dari repositori lain juga dipertimbangkan.
4. Terapkan ApprovedPatchesseperti yang ditentukan dalam baseline patch. Tambalan yang disetujui disetujui untuk diperbarui meskipun dibuang oleh GlobalFiltersatau jika tidak ada aturan persetujuan yang ditentukan dalam ApprovalRulesmemberikan persetujuan.
5. Terapkan RejectedPatchesseperti yang ditentukan dalam baseline patch. Patch yang ditolak akan dihapus dari daftar patch yang disetujui dan tidak akan diterapkan.
6. Jika beberapa versi patch disetujui, versi yang terbaru diterapkan.
7. API pembaruan Zypper diterapkan untuk patch yang disetujui.
8. Node terkelola di-boot ulang jika ada pembaruan yang diinstal. (Pengecualian: Jika `RebootOption` parameter disetel ke `NoReboot` dalam `AWS-RunPatchBaseline` dokumen,

node terkelola tidak di-boot ulang setelah Patch Manager dijalankan. Untuk informasi lebih lanjut, lihatNama parameter: `RebootOption`.)

Ubuntu Server

Pada node Ubuntu Server terkelola, alur kerja instalasi patch adalah sebagai berikut:

1. Jika daftar patch ditentukan menggunakan URL https atau URL ala jalur Amazon Simple Storage Service (Amazon S3) menggunakan parameter `InstallOverrideList` untuk dokumen `AWS-RunPatchBaseline` atau `AWS-RunPatchBaselineAssociation`, patch yang terdaftar diinstal dan langkah 2-7 dilewati.
2. Jika pembaruan tersedia untuk `python3-apt` (antarmuka pustaka Python `kelibapt`), itu ditingkatkan ke versi terbaru. (Paket nonsecurity ini ditingkatkan meskipun Anda tidak memilih opsi Sertakan pembaruan nonsecurity.)
3. Terapkan GlobalFiltersseperti yang ditentukan dalam baseline patch, hanya menyimpan paket yang memenuhi syarat untuk diproses lebih lanjut.
4. Terapkan ApprovalRulesseperti yang ditentukan dalam baseline patch. Setiap aturan persetujuan dapat menentukan paket sebagai disetujui.

> **Note**
>
> Karena tidak mungkin menentukan tanggal rilis paket pembaruan secara andalUbuntu Server, opsi persetujuan otomatis tidak didukung untuk sistem operasi ini.

Namun, aturan persetujuan, juga tunduk pada apakah kontak centang Sertakan pembaruan non-keamanan dipilih saat membuat atau terakhir memperbarui dasar patch.

Jika pembaruan non-keamanan dikecualikan, diterapkan suatu aturan implisit untuk memilih hanya paket dengan pemutakhiran dalam repo keamanan. Untuk setiap paket, versi kandidat paket (yang biasanya versi terbaru) harus merupakan bagian dari repo keamanan.

Jika pembaruan non-keamanan disertakan, patch dari repositori lain juga dipertimbangkan.

Namun, aturan persetujuan juga tunduk pada apakah kontak centang Sertakan pembaruan non-keamanan dipilih saat membuat atau terakhir memperbarui dasar patch.

> **Note**
>
> Untuk setiap versiUbuntu Server, versi kandidat tambalan terbatas pada tambalan yang merupakan bagian dari repo terkait untuk versi tersebut, sebagai berikut:
>
> - Ubuntu Server14.04 LTS: `trusty-security`
> - Ubuntu Server16.04 LTS: `xenial-security`
> - Ubuntu Server18.04 LTS: `bionic-security`
> - Ubuntu Server20.04 LTS): `focal-security`
> - Ubuntu Server20.10 STR: `groovy-security`
> - Ubuntu Server22.04 LTS: `jammy-security`
> - Ubuntu Server23.04: `lunar-lobster`

5. Terapkan ApprovedPatchesseperti yang ditentukan dalam baseline patch. Tambalan yang disetujui disetujui untuk diperbarui meskipun dibuang oleh GlobalFiltersatau jika tidak ada aturan persetujuan yang ditentukan dalam ApprovalRulesmemberikan persetujuan.
6. Terapkan RejectedPatchesseperti yang ditentukan dalam baseline patch. Patch yang ditolak akan dihapus dari daftar patch yang disetujui dan tidak akan diterapkan.
7. Pustaka APT digunakan untuk memutakhirkan paket.
8. Node terkelola di-boot ulang jika ada pembaruan yang diinstal. (Pengecualian: Jika `RebootOption` parameter disetel ke `NoReboot` dalam `AWS-RunPatchBaseline` dokumen, node terkelola tidak di-boot ulang setelah Patch Manager dijalankan. Untuk informasi lebih lanjut, lihatNama parameter: `RebootOption`.)

Windows Server

Ketika operasi patching dilakukan pada node Windows Server terkelola, node meminta snapshot dari baseline patch yang sesuai dari Systems Manager. snapshot ini berisi daftar semua pembaruan yang tersedia di dasar patch yang disetujui untuk deployment. Daftar pembaruan ini dikirim ke Windows Update API, yang menentukan pembaruan mana yang berlaku untuk node terkelola dan menginstalnya sesuai kebutuhan. Jika ada pembaruan yang diinstal, node terkelola akan di-boot ulang setelahnya, sebanyak yang diperlukan untuk menyelesaikan semua tambalan yang diperlukan. (Pengecualian: Jika `RebootOption` parameter disetel ke `NoReboot` dalam `AWS-RunPatchBaseline` dokumen, node terkelola tidak di-boot ulang setelah Patch Manager dijalankan. Untuk informasi lebih lanjut, lihatNama parameter: `RebootOption`.) Ringkasan

operasi penambalan dapat ditemukan di output Run Command permintaan. Log tambahan dapat ditemukan pada node terkelola di `%PROGRAMDATA%\Amazon\PatchBaselineOperations\Logs` folder.

Karena API Pembaruan Windows digunakan untuk mengunduh dan menginstal patch, semua pengaturan Kebijakan Grup untuk Windows Update akan dihormati. Tidak ada pengaturan Kebijakan Grup yang diperlukan untuk digunakanPatch Manager, tetapi pengaturan apa pun yang telah Anda tetapkan akan diterapkan, seperti mengarahkan node terkelola ke server Windows Server Update Services (WSUS).

> **Note**
>
> Secara default, Windows mengunduh semua tambalan dari situs Pembaruan Windows Microsoft karena Patch Manager menggunakan API Pembaruan Windows untuk mendorong unduhan dan pemasangan tambalan. Akibatnya, node yang dikelola harus dapat mencapai situs Pembaruan Microsoft Windows atau tambalan akan gagal. Atau, Anda dapat mengkonfigurasi server WSUS untuk berfungsi sebagai repositori patch dan mengkonfigurasi node terkelola Anda untuk menargetkan server WSUS menggunakan Kebijakan Grup.

Cara kerja aturan dasar patch pada sistem berbasis Linux

Aturan dalam dasar patch untuk distribusi Linux beroperasi dengan cara yang berbeda berdasarkan jenis distribusi. Tidak seperti pembaruan tambalan pada node Windows Server terkelola, aturan dievaluasi pada setiap node untuk mempertimbangkan repo yang dikonfigurasi pada instance. Patch Manager, kemampuan AWS Systems Manager, menggunakan manajer paket asli untuk mendorong pemasangan tambalan yang disetujui oleh baseline patch.

Untuk jenis sistem operasi berbasis Linux yang melaporkan tingkat keparahan patch, Patch Manager gunakan tingkat keparahan yang dilaporkan oleh penerbit perangkat lunak untuk pemberitahuan pembaruan atau tambalan individual. Patch Managertidak memperoleh tingkat keparahan dari sumber pihak ketiga, seperti Common Vulnerability Scoring System (CVSS), atau dari metrik yang dirilis oleh National Vulnerability Database (NVD).

Topik

- Cara kerja aturan dasar patch di Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 dan Amazon Linux 2023

- Cara kerja aturan dasar tambalan di CentOS dan CentOS Stream
- Cara kerja aturan dasar tambalan dan Debian ServerRaspberry Pi OS
- Cara kerja aturan dasar patch pada macOS
- Cara kerja aturan dasar patch pada Oracle Linux
- Cara kerja aturan dasar tambalan AlmaLinux,, dan RHELRocky Linux
- Cara kerja aturan dasar patch pada SUSE Linux Enterprise Server
- Cara kerja aturan dasar patch pada Ubuntu Server

Cara kerja aturan dasar patch di Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 dan Amazon Linux 2023

Di Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 dan Amazon Linux 2023, proses pemilihan tambalan adalah sebagai berikut:

1. Pada node yang dikelola, perpustakaan YUM (Amazon Linux 1 dan, Amazon Linux 2) atau pustaka DNF (Amazon Linux 2022 dan Amazon Linux 2023) mengakses `updateinfo.xml` file untuk setiap repo yang dikonfigurasi.

   > **Note**
   >
   > Jika tidak ada **`updateinfo.xml`** file yang ditemukan, apakah tambalan diinstal tergantung pada pengaturan untuk Sertakan pembaruan non-keamanan dan Persetujuan otomatis. Sebagai contoh, jika pembaruan non-keamanan diizinkan, pembaruan akan diinstal saat waktu persetujuan otomatis tiba.

2. Setiap pemberitahuan pembaruan di `updateinfo.xml` mencakup beberapa atribut yang menunjukkan properti paket dalam pemberitahuan tersebut, seperti yang dijelaskan di tabel berikut.

   Atribut pemberitahuan pembaruan

Atribut	Deskripsi
tipe	Sesuai dengan nilai atribut kunci Klasifikasi dalam tipe PatchFilterdata baseline patch. Menunjukkan jenis paket yang disertakan dalam pemberitahuan pembaruan.

Atribut	Deskripsi
	Anda dapat melihat daftar nilai yang didukung dengan menggunakan AWS CLI perintah describe-patch-properties atau operasi APIDescribePatchProperties. Anda juga dapat melihat daftar di area Aturan persetujuan pada halaman Buat dasar patch atau halaman Edit dasar patch di konsol Systems Manager.
kepelikan	Sesuai dengan nilai atribut kunci Keparahan dalam tipe PatchFilterdata baseline patch. Menunjukkan tingkat kepelikan paket yang disertakan dalam pemberitahuan pembaruan. Biasanya hanya berlaku untuk pemberitahuan pembaruan Keamanan. Anda dapat melihat daftar nilai yang didukung dengan menggunakan AWS CLI perintah describe-patch-properties atau operasi APIDescribePatchProperties. Anda juga dapat melihat daftar di area Aturan persetujuan pada halaman Buat dasar patch atau halaman Edit dasar patch di konsol Systems Manager.
update_id	Menunjukkan ID penasehat, seperti ALIAS-2017-867. ID penasihat dapat digunakan dalam RejectedPatchesatribut ApprovedPatchesor di baseline patch.
referensi	Berisi informasi tambahan tentang pemberita huan pembaruan, seperti ID CVE (format: CVE-2017-1234567). ID CVE dapat digunakan dalam RejectedPatchesatribut ApprovedPatchesor di baseline patch.

Atribut	Deskripsi
diperbarui	Sesuai dengan ApproveAfterDaysdi baseline patch. Menunjukkan tanggal dirilis (tanggal diperbarui) dari paket yang disertakan dalam pemberitahuan pembaruan. Perbandingan antara stempel waktu saat ini dan nilai atribut ini ditambah digunakan untuk menentukan apakah tambalan disetujui untuk penerapan. `ApproveAfterDays`

> **Note**
>
> Untuk informasi tentang format yang diterima untuk daftar patch yang disetujui dan patch yang ditolak, lihat Tentang format nama paket untuk daftar patch yang disetujui dan ditolak.

3. Produk dari node terkelola ditentukan olehSSM Agent. Atribut ini sesuai dengan nilai atribut kunci Produk dalam tipe PatchFilterdata baseline patch.
4. Paket dipilih untuk pembaruan sesuai dengan pedoman berikut.

Opsi keamanan	Pemilihan patch
Garis dasar patch default yang telah ditentukan sebelumnya yang disediakan oleh AWS dan baseline patch kustom di mana Sertakan pembaruan non-keamanan tidak dipilih	Untuk setiap pemberitahuan pembaruan di `updateinfo.xml` , dasar patch digunakan sebagai filter, memungkinkan hanya paket yang memenuhi syarat untuk disertakan dalam pembaruan. Jika beberapa paket berlaku setelah menerapkan definisi dasar patch, versi terbaru digunakan. Untuk Amazon Linux 1 dan Amazon Linux 2, perintah yum yang setara untuk alur kerja ini adalah:

Opsi keamanan	Pemilihan patch
	`sudo yum update-minimal --sec-severity=Critical,Important --bugfix -y` Untuk Amazon Linux 2022 dan Amazon Linux 2023, perintah dnf yang setara untuk alur kerja ini adalah: `sudo dnf upgrade-minimal --sec-severity=Critical--sec-severity=Important --bugfix -y`
Garis dasar patch kustom di mana kotak centang Sertakan pembaruan non-keamanan dipilih dengan daftar KEPARAHAN `[Critical, Important]` dan daftar KLASIFIKASI `[Security, Bugfix]`	Selain menerapkan pembaruan keamanan yang dipilih`updateinfo.xml` , Patch Manager menerapkan pembaruan nonsecurity yang memenuhi aturan pemfilteran tambalan. Untuk Amazon Linux dan Amazon Linux 2, perintah yum yang setara untuk alur kerja ini adalah: `sudo yum update-minimal --security --sec-severity=Critical,Important --bugfix -y` Untuk Amazon Linux 2022 dan Amazon Linux 2023, perintah dnf yang setara untuk alur kerja ini adalah: `sudo dnf upgrade-minimal --security --sec-severity=Critical --sec=severity=Imporant --bugfix -y`

Untuk informasi tentang nilai status kepatuhan patch, lihat Memahami nilai keadaan kepatuhan patch.

Cara kerja aturan dasar tambalan di CentOS dan CentOS Stream

CentOS dan repositori CentOS Stream default tidak menyertakan file. `updateinfo.xml` Namun, repositori khusus yang Anda buat atau gunakan mungkin menyertakan file ini. Dalam topik ini, referensi hanya `updateinfo.xml` berlaku untuk repositori khusus ini.

Pada CentOS danCentOS Stream, proses pemilihan tambalan adalah sebagai berikut:

1. Pada node terkelola, perpustakaan YUM (pada versi CentOS 6.x dan 7.x) atau pustaka DNF (pada CentOS 8.x dan CentOS Stream) `updateinfo.xml` mengakses file, jika ada di repositori khusus, untuk setiap repo yang dikonfigurasi.

   Jika tidak **`updateinfo.xml`** ditemukan, yang selalu menyertakan repo default, apakah tambalan diinstal tergantung pada pengaturan untuk Sertakan pembaruan non-keamanan dan Persetujuan otomatis. Sebagai contoh, jika pembaruan non-keamanan diizinkan, pembaruan akan diinstal saat waktu persetujuan otomatis tiba.

2. Jika `updateinfo.xml` ada, setiap pemberitahuan pembaruan dalam file menyertakan beberapa atribut yang menunjukkan properti paket dalam pemberitahuan, seperti yang dijelaskan dalam tabel berikut.

   Atribut pemberitahuan pembaruan

Atribut	Deskripsi
tipe	Sesuai dengan nilai atribut kunci Klasifikasi dalam tipe PatchFilterdata baseline patch. Menunjukkan jenis paket yang disertakan dalam pemberitahuan pembaruan. Anda dapat melihat daftar nilai yang didukung dengan menggunakan AWS CLI perintah describe-patch-properties atau operasi APIDescribePatchProperties. Anda juga dapat melihat daftar di area Aturan persetujuan pada halaman Buat dasar patch atau halaman Edit dasar patch di konsol Systems Manager.
kepelikan	Sesuai dengan nilai atribut kunci Keparahan dalam tipe PatchFilterdata baseline patch.

Atribut	Deskripsi
	Menunjukkan tingkat kepelikan paket yang disertakan dalam pemberitahuan pembaruan. Biasanya hanya berlaku untuk pemberitahuan pembaruan Keamanan. Anda dapat melihat daftar nilai yang didukung dengan menggunakan AWS CLI perintah describe-patch-properties atau operasi APIDescribePatchProperties. Anda juga dapat melihat daftar di area Aturan persetujuan pada halaman Buat dasar patch atau halaman Edit dasar patch di konsol Systems Manager.
update_id	Menunjukkan ID penasehat, seperti CVE-2019-17055. ID penasihat dapat digunakan dalam RejectedPatchesatribut ApprovedPatchesor di baseline patch.
referensi	Berisi informasi tambahan tentang pemberita huan pembaruan, seperti ID CVE (format: CVE-2019-17055) atau ID Bugzilla (format: 1463241). ID CVE dan ID Bugzilla dapat digunakan dalam RejectedPatchesatribut ApprovedPatchesor di baseline patch.
diperbarui	Sesuai dengan ApproveAfterDaysdi baseline patch. Menunjukkan tanggal dirilis (tanggal diperbarui) dari paket yang disertakan dalam pemberitahuan pembaruan. Perbandingan antara stempel waktu saat ini dan nilai atribut ini ditambah digunakan untuk menentukan apakah tambalan disetujui untuk penerapan. `ApproveAfterDays`

> **Note**
>
> Untuk informasi tentang format yang diterima untuk daftar patch yang disetujui dan patch yang ditolak, lihat Tentang format nama paket untuk daftar patch yang disetujui dan ditolak.

3. Dalam semua kasus, produk dari node yang dikelola ditentukan olehSSM Agent. Atribut ini sesuai dengan nilai atribut kunci Produk dalam tipe PatchFilterdata baseline patch.
4. Paket dipilih untuk pembaruan sesuai dengan pedoman berikut.

Opsi keamanan	Pemilihan patch
Garis dasar patch default yang telah ditentukan sebelumnya yang disediakan oleh AWS dan baseline patch kustom di mana Sertakan pembaruan non-keamanan tidak dipilih	Untuk setiap pemberitahuan pembaruan di`updateinfo.xml` , jika ada di repositori khusus, baseline patch digunakan sebagai filter, yang memungkinkan hanya paket yang memenuhi syarat untuk disertakan dalam pembaruan. Jika beberapa paket berlaku setelah menerapkan definisi dasar patch, versi terbaru digunakan. Untuk CentOS 6 dan 7 di mana `updateinfo.xml` ada, perintah yum setara untuk alur kerja ini adalah: `sudo yum update-minimal --security=Critical,Important --bugfix -y` Untuk CentOS 8 dan CentOS Stream di mana `updateinfo.xml` ada, perintah dnf setara untuk alur kerja ini adalah:

Opsi keamanan	Pemilihan patch
	`sudo dnf upgrade-minimal --sec-severity=Critical --sec-severity=Important --bugfix -y`
Garis dasar patch kustom di mana kotak centang Sertakan pembaruan non-keamanan dipilih dengan daftar KEPARAHAN `[Critical, Important]` dan daftar KLASIFIKASI `[Security, Bugfix]`	Selain menerapkan pembaruan keamanan yang dipilih `updateinfo.xml`, jika ada di repositori khusus, Patch Manager menerapkan pembaruan nonsecurity yang memenuhi aturan pemfilteran tambalan. Untuk CentOS 6 dan 7 di mana `updateinfo.xml` ada, perintah yum setara untuk alur kerja ini adalah: `sudo yum update --sec-severity=Critical,Important --bugfix -y` Untuk CentOS 8 dan CentOS Stream di mana `updateinfo.xml` ada, perintah dnf setara untuk alur kerja ini adalah: `sudo dnf upgrade --security --sec-severity=Critical --sec-severity="Important" --bugfix -y` Untuk repo default dan repo kustom tanpa `updateinfo.xml`, Anda harus memilih kotak centang Sertakan pembaruan non-keamanan untuk memperbarui paket sistem operasi (OS).

Untuk informasi tentang nilai status kepatuhan patch, lihat Memahami nilai keadaan kepatuhan patch.

Cara kerja aturan dasar tambalan dan Debian ServerRaspberry Pi OS

Pada Debian Server dan Raspberry Pi OS (sebelumnya Raspbian), layanan baseline patch menawarkan pemfilteran pada bidang Prioritas dan Bagian. Bidang ini biasanya hadir untuk semua Debian Server dan Raspberry Pi OS paket. Untuk menentukan apakah patch dipilih oleh baseline patch, Patch Manager lakukan hal berikut:

1. On Debian Server dan Raspberry Pi OS sistem, setara dengan `sudo apt-get update` dijalankan untuk menyegarkan daftar paket yang tersedia. Repo tidak dikonfigurasi dan data ditarik dari repo yang dikonfigurasi dalam daftar `sources`.
2. Jika pembaruan tersedia untuk `python3-apt` (antarmuka pustaka Python ke`libapt`), itu ditingkatkan ke versi terbaru. (Paket nonsecurity ini ditingkatkan meskipun Anda tidak memilih opsi Sertakan pembaruan nonsecurity.)

   > **Important**
   >
   > Hanya pada Debian Server 8: Karena Debian Server 8.* sistem operasi mengacu pada repositori paket usang (`jessie-backports`), Patch Manager melakukan langkah-langkah tambahan berikut untuk memastikan bahwa operasi patching berhasil:
   >
   > a. Pada node terkelola Anda, referensi ke `jessie-backports` repositori dikomentari dari daftar lokasi sumber (). `/etc/apt/sources.list.d/jessie-backports` Akibatnya, tidak ada upaya yang dilakukan untuk mengunduh patch dari lokasi tersebut.
   >
   > b. Kunci penandatanganan pembaruan keamanan Stretch diimpor. Kunci ini memberikan izin yang diperlukan untuk operasi pembaruan dan penginstalan pada distribusi Debian Server 8.*.
   >
   > c. `apt-get`Operasi dijalankan pada titik ini untuk memastikan bahwa versi terbaru diinstal sebelum proses penambalan dimulai. `python3-apt`
   >
   > d. Setelah proses instalasi selesai, referensi ke repositori `jessie-backports` dipulihkan dan kunci penandatanganan dihapus dari keyring sumber apt. Hal ini dilakukan untuk mempertahankan konfigurasi sistem seperti keadaan sebelum operasi patching.

3. Selanjutnya GlobalFilters, RejectedPatchesdaftar ApprovalRules, ApprovedPatchesdan diterapkan.

> **Note**
>
> Karena tidak mungkin menentukan tanggal rilis paket pembaruan secara andalDebian Server, opsi persetujuan otomatis tidak didukung untuk sistem operasi ini.

Namun, aturan persetujuan, juga tunduk pada apakah kontak centang Sertakan pembaruan non-keamanan dipilih saat membuat atau terakhir memperbarui dasar patch.

Jika pembaruan non-keamanan dikecualikan, diterapkan suatu aturan implisit untuk memilih hanya paket dengan pemutakhiran dalam repo keamanan. Untuk setiap paket, versi kandidat paket (yang biasanya versi terbaru) harus merupakan bagian dari repo keamanan. Dalam hal ini, untukDebian Server, versi kandidat tambalan terbatas pada tambalan yang disertakan dalam repo berikut:

Repo ini dinamakan sebagai berikut:

- Debian Server8: `debian-security jessie`
- Debian Serverdan Raspberry Pi OS 9: `debian-security stretch`
- Debian Server10: `debian-security buster`
- Debian Server11: `debian-security bullseye`
- Debian Server12: `debian-security bookworm`

Jika pembaruan non-keamanan disertakan, patch dari repositori lain juga dipertimbangkan.

> **Note**
>
> Untuk informasi tentang format yang diterima untuk daftar patch yang disetujui dan patch yang ditolak, lihat Tentang format nama paket untuk daftar patch yang disetujui dan ditolak.

Untuk melihat konten bidang Prioritas dan Bagian, jalankan perintah `aptitude` berikut ini:

> **Note**
>
> Anda mungkin perlu menginstal Aptitude terlebih dahulu pada Debian Server sistem.

```
aptitude search -F '%p %P %s %t %V#' '~U'
```

Dalam menanggapi perintah ini, semua paket yang dapat dimutakhirkan dilaporkan dalam format ini:

```
name, priority, section, archive, candidate version
```

Untuk informasi tentang nilai status kepatuhan patch, lihat Memahami nilai keadaan kepatuhan patch.

Cara kerja aturan dasar patch pada macOS

Pada macOS, proses pemilihan patch adalah sebagai berikut:

1. Pada node terkelola, Patch Manager mengakses konten `InstallHistory.plist` file yang diurai dan mengidentifikasi nama dan versi paket.

   Untuk detail tentang proses penguraian, lihat bagian macOS dalam Cara menginstal patch.
2. Produk dari node terkelola ditentukan olehSSM Agent. Atribut ini sesuai dengan nilai atribut kunci Produk dalam tipe PatchFilterdata baseline patch.
3. Paket dipilih untuk pembaruan sesuai dengan pedoman berikut.

Opsi keamanan	Pemilihan patch
Garis dasar patch default yang telah ditentuka n sebelumnya yang disediakan oleh AWS dan baseline patch kustom di mana Sertakan pembaruan non-keamanan tidak dipilih	Untuk setiap pembaruan paket yang tersedia, dasar patch digunakan sebagai filter, memungkinkan hanya paket yang memenuhi syarat untuk disertakan dalam pembaruan . Jika beberapa paket berlaku setelah menerapkan definisi dasar patch, versi terbaru digunakan.
Garis dasar patch khusus di mana Sertakan pembaruan non-keamanan dipilih	Selain menerapkan pembaruan keamanan yang diidentifikasi dengan menggunak an `InstallHistory.plist` , Patch Manager menerapkan pembaruan non-keama nan yang memenuhi aturan filter patch.

Untuk informasi tentang nilai status kepatuhan patch, lihat Memahami nilai keadaan kepatuhan patch.

Cara kerja aturan dasar patch pada Oracle Linux

Pada Oracle Linux, proses pemilihan patch adalah sebagai berikut:

1. Pada node terkelola, pustaka YUM mengakses `updateinfo.xml` file untuk setiap repo yang dikonfigurasi.

   > **Note**
   >
   > file `updateinfo.xml` mungkin tidak tersedia jika repo tidak dikelola oleh Oracle. Jika tidak **`updateinfo.xml`** ditemukan, apakah patch diinstal tergantung pada pengaturan untuk Sertakan pembaruan non-keamanan dan Persetujuan otomatis. Sebagai contoh, jika pembaruan non-keamanan diizinkan, pembaruan akan diinstal saat waktu persetujuan otomatis tiba.

2. Setiap pemberitahuan pembaruan di `updateinfo.xml` mencakup beberapa atribut yang menunjukkan properti paket dalam pemberitahuan tersebut, seperti yang dijelaskan di tabel berikut.

   Atribut pemberitahuan pembaruan

Atribut	Deskripsi
tipe	Sesuai dengan nilai atribut kunci Klasifikasi dalam tipe PatchFilterdata baseline patch. Menunjukkan jenis paket yang disertakan dalam pemberitahuan pembaruan. Anda dapat melihat daftar nilai yang didukung dengan menggunakan AWS CLI perintah describe-patch-properties atau operasi APIDescribePatchProperties. Anda juga dapat melihat daftar di area Aturan persetujuan pada halaman Buat dasar patch atau halaman Edit dasar patch di konsol Systems Manager.
kepelikan	Sesuai dengan nilai atribut kunci Keparahan dalam tipe PatchFilterdata baseline patch. Menunjukkan tingkat kepelikan paket yang

Atribut	Deskripsi
	disertakan dalam pemberitahuan pembaruan. Biasanya hanya berlaku untuk pemberitahuan pembaruan Keamanan. Anda dapat melihat daftar nilai yang didukung dengan menggunakan AWS CLI perintah describe-patch-properties atau operasi APIDescribePatchProperties. Anda juga dapat melihat daftar di area Aturan persetujuan pada halaman Buat dasar patch atau halaman Edit dasar patch di konsol Systems Manager.
update_id	Menunjukkan ID penasehat, seperti CVE-2019-17055. ID penasihat dapat digunakan dalam RejectedPatchesatribut ApprovedPatchesor di baseline patch.
referensi	Berisi informasi tambahan tentang pemberita huan pembaruan, seperti ID CVE (format: CVE-2019-17055) atau ID Bugzilla (format: 1463241). ID CVE dan ID Bugzilla dapat digunakan dalam RejectedPatchesatribut ApprovedPatchesor di baseline patch.
diperbarui	Sesuai dengan ApproveAfterDaysdi baseline patch. Menunjukkan tanggal dirilis (tanggal diperbarui) dari paket yang disertakan dalam pemberitahuan pembaruan. Perbandingan antara stempel waktu saat ini dan nilai atribut ini ditambah digunakan untuk menentukan apakah tambalan disetujui untuk penerapan. `ApproveAfterDays`

> **Note**
>
> Untuk informasi tentang format yang diterima untuk daftar patch yang disetujui dan patch yang ditolak, lihat Tentang format nama paket untuk daftar patch yang disetujui dan ditolak.

3. Produk dari node terkelola ditentukan olehSSM Agent. Atribut ini sesuai dengan nilai atribut kunci Produk dalam tipe PatchFilterdata baseline patch.
4. Paket dipilih untuk pembaruan sesuai dengan pedoman berikut.

Opsi keamanan	Pemilihan patch
Garis dasar patch default yang telah ditentukan sebelumnya yang disediakan oleh AWS dan baseline patch kustom di mana Sertakan pembaruan non-keamanan tidak dipilih	Untuk setiap pemberitahuan pembaruan di `updateinfo.xml` , dasar patch digunakan sebagai filter, memungkinkan hanya paket yang memenuhi syarat untuk disertakan dalam pembaruan. Jika beberapa paket berlaku setelah menerapkan definisi dasar patch, versi terbaru digunakan. Untuk node terkelola versi 7, perintah yum yang setara untuk alur kerja ini adalah: `sudo yum update-minimal --security=Important,Moderate --bugfix -y` Untuk node terkelola versi 8 dan 9, perintah dnf yang setara untuk alur kerja ini adalah: `sudo dnf upgrade-minimal --security --sec-severity Moderate --sec-severity Important`
Garis dasar patch kustom di mana Sertakan pembaruan non-keamanan dipilih dengan	Selain menerapkan pembaruan keamanan yang dipilih`updateinfo.xml` , Patch

Opsi keamanan	Pemilihan patch
daftar KEPARAHAN `[Critical, Important]` dan daftar KLASIFIKASI `[Security, Bugfix]`	Manager menerapkan pembaruan nonsecurity yang memenuhi aturan pemfilteran tambalan. Untuk node terkelola versi 7, perintah yum yang setara untuk alur kerja ini adalah: `sudo yum update --security --sec-severity=Critical,Important --bugfix -y` Untuk node terkelola versi 8 dan 9, perintah dnf yang setara untuk alur kerja ini adalah: `sudo dnf upgrade --security --sec-severity=Critical, --sec-severity=Important --bugfix y`

Untuk informasi tentang nilai status kepatuhan patch, lihat Memahami nilai keadaan kepatuhan patch.

Cara kerja aturan dasar tambalan AlmaLinux,, dan RHELRocky Linux

Pada AlmaLinux, Red Hat Enterprise Linux (RHEL), danRocky Linux, proses pemilihan tambalan adalah sebagai berikut:

1. Pada node terkelola, perpustakaan YUM (RHEL7) atau pustaka DNF (AlmaLinux 8 dan 9, 8 dan 9, dan RHEL Rocky Linux 8 dan 9) mengakses `updateinfo.xml` file untuk setiap repo yang dikonfigurasi.

   > **Note**
   >
   > file `updateinfo.xml` mungkin tidak tersedia jika repo tidak dikelola oleh Red Hat. Jika tidak ada `updateinfo.xml` yang ditemukan, tidak ada patch yang diterapkan.

2. Setiap pemberitahuan pembaruan di `updateinfo.xml` mencakup beberapa atribut yang menunjukkan properti paket dalam pemberitahuan tersebut, seperti yang dijelaskan di tabel berikut.

Atribut pemberitahuan pembaruan

Atribut	Deskripsi
tipe	Sesuai dengan nilai atribut kunci Klasifikasi dalam tipe PatchFilterdata baseline patch. Menunjukkan jenis paket yang disertakan dalam pemberitahuan pembaruan. Anda dapat melihat daftar nilai yang didukung dengan menggunakan AWS CLI perintah describe-patch-properties atau operasi APIDescribePatchProperties. Anda juga dapat melihat daftar di area Aturan persetujuan pada halaman Buat dasar patch atau halaman Edit dasar patch di konsol Systems Manager.
kepelikan	Sesuai dengan nilai atribut kunci Keparahan dalam tipe PatchFilterdata baseline patch. Menunjukkan tingkat kepelikan paket yang disertakan dalam pemberitahuan pembaruan. Biasanya hanya berlaku untuk pemberitahuan pembaruan Keamanan. Anda dapat melihat daftar nilai yang didukung dengan menggunakan AWS CLI perintah describe-patch-properties atau operasi APIDescribePatchProperties. Anda juga dapat melihat daftar di area Aturan persetujuan pada halaman Buat dasar patch atau halaman Edit dasar patch di konsol Systems Manager.
update_id	Menunjukkan ID penasehat, seperti RHSA-2017:0864. ID penasihat dapat digunakan dalam RejectedPatchesatribut ApprovedPatchesor di baseline patch.

Atribut	Deskripsi
referensi	Berisi informasi tambahan tentang pemberitahuan pembaruan, seperti ID CVE (format: CVE-2017-1000371) atau ID Bugzilla (format: 1463241). ID CVE dan ID Bugzilla dapat digunakan dalam RejectedPatchesatribut ApprovedPatchesor di baseline patch.
diperbarui	Sesuai dengan ApproveAfterDaysdi baseline patch. Menunjukkan tanggal dirilis (tanggal diperbarui) dari paket yang disertakan dalam pemberitahuan pembaruan. Perbandingan antara stempel waktu saat ini dan nilai atribut ini ditambah digunakan untuk menentukan apakah tambalan disetujui untuk penerapan. `ApproveAfterDays`

> **Note**
>
> Untuk informasi tentang format yang diterima untuk daftar patch yang disetujui dan patch yang ditolak, lihat Tentang format nama paket untuk daftar patch yang disetujui dan ditolak.

3. Produk dari node terkelola ditentukan olehSSM Agent. Atribut ini sesuai dengan nilai atribut kunci Produk dalam tipe PatchFilterdata baseline patch.
4. Paket dipilih untuk pembaruan sesuai dengan pedoman berikut.

Opsi keamanan	Pemilihan patch
Garis dasar patch default yang telah ditentukan sebelumnya yang disediakan oleh AWS dan baseline patch kustom di mana kotak centang Sertakan pembaruan non-keamanan tidak dipilih dalam aturan apa pun	Untuk setiap pemberitahuan pembaruan di `updateinfo.xml` , dasar patch digunakan sebagai filter, memungkinkan hanya paket yang memenuhi syarat untuk disertakan dalam pembaruan. Jika beberapa paket

Opsi keamanan	Pemilihan patch
	berlaku setelah menerapkan definisi dasar patch, versi terbaru digunakan. Untuk RHEL 7, perintah yum setara untuk alur kerja ini adalah: `sudo yum update-minimal --sec-severity=Critical,Important --bugfix -y` Untuk AlmaLinux 8 dan 9, RHEL 8 dan 9, dan Rocky Linux 8 dan 9, perintah dnf setara untuk alur kerja ini adalah: `sudo dnf upgrade-minimal --sec-severity=Critical  --sec-severity=Important --bugfix -y`

Opsi keamanan	Pemilihan patch
Garis dasar patch kustom di mana kotak centang Sertakan pembaruan non-keamanan dipilih dengan daftar KEPARAHAN `[Critical, Important]` dan daftar KLASIFIKASI `[Security, Bugfix]`	Selain menerapkan pembaruan keamanan yang dipilih`updateinfo.xml` , Patch Manager menerapkan pembaruan nonsecurity yang memenuhi aturan pemfilteran tambalan. Untuk RHEL 7, perintah yum setara untuk alur kerja ini adalah: `sudo yum update --security --sec-severity=Critical,Important --bugfix -y` Untuk AlmaLinux 8 dan 9, RHEL 8 dan 9, dan Rocky Linux 8 dan 9, perintah dnf setara untuk alur kerja ini adalah: `sudo dnf upgrade --sec-severity=Critical --sec-severity=Important --bugfix -y`

Untuk informasi tentang nilai status kepatuhan patch, lihat Memahami nilai keadaan kepatuhan patch.

Cara kerja aturan dasar patch pada SUSE Linux Enterprise Server

Pada SLES, setiap patch mencakup atribut berikut ini yang menunjukkan properti paket dalam patch tersebut:

- Kategori: Sesuai dengan nilai atribut kunci Klasifikasi dalam tipe PatchFilterdata baseline patch. Menunjukkan jenis patch yang disertakan dalam pemberitahuan pembaruan.

  Anda dapat melihat daftar nilai yang didukung dengan menggunakan AWS CLI perintah describe-patch-properties atau operasi APIDescribePatchProperties. Anda juga dapat melihat daftar di area Aturan persetujuan pada halaman Buat dasar patch atau halaman Edit dasar patch di konsol Systems Manager.
- Keparahan: Sesuai dengan nilai atribut kunci Keparahan dalam tipe PatchFilterdata baseline patch. Menunjukkan tingkat kepelikan patch.

Anda dapat melihat daftar nilai yang didukung dengan menggunakan AWS CLI perintah describe-patch-properties atau operasi APIDescribePatchProperties. Anda juga dapat melihat daftar di area Aturan persetujuan pada halaman Buat dasar patch atau halaman Edit dasar patch di konsol Systems Manager.

Produk dari node terkelola ditentukan olehSSM Agent. Atribut ini sesuai dengan nilai atribut kunci Produk dalam tipe PatchFilterdata baseline patch.

Untuk setiap patch, dasar patch digunakan sebagai filter, memungkinkan hanya paket yang memenuhi syarat untuk disertakan dalam pembaruan. Jika beberapa paket berlaku setelah menerapkan definisi dasar patch, versi terbaru digunakan.

> **Note**
>
> Untuk informasi tentang format yang diterima untuk daftar patch yang disetujui dan patch yang ditolak, lihat Tentang format nama paket untuk daftar patch yang disetujui dan ditolak.

Cara kerja aturan dasar patch pada Ubuntu Server

PadaUbuntu Server, layanan baseline patch menawarkan pemfilteran pada bidang Prioritas dan Bagian. Bidang ini biasanya hadir untuk semua Ubuntu Server paket. Untuk menentukan apakah patch dipilih oleh baseline patch, Patch Manager lakukan hal berikut:

1. Pada Ubuntu Server sistem, setara `sudo apt-get update` dijalankan untuk menyegarkan daftar paket yang tersedia. Repo tidak dikonfigurasi dan data ditarik dari repo yang dikonfigurasi dalam daftar `sources`.
2. Jika pembaruan tersedia untuk `python3-apt` (antarmuka pustaka Python `kelibapt`), itu ditingkatkan ke versi terbaru. (Paket nonsecurity ini ditingkatkan meskipun Anda tidak memilih opsi Sertakan pembaruan nonsecurity.)
3. Selanjutnya GlobalFilters, RejectedPatchesdaftar ApprovalRules, ApprovedPatchesdan diterapkan.

> **Note**
>
> Karena tidak mungkin menentukan tanggal rilis paket pembaruan secara andalUbuntu Server, opsi persetujuan otomatis tidak didukung untuk sistem operasi ini.

Namun, aturan persetujuan, juga tunduk pada apakah kontak centang Sertakan pembaruan non-keamanan dipilih saat membuat atau terakhir memperbarui dasar patch.

Jika pembaruan non-keamanan dikecualikan, diterapkan suatu aturan implisit untuk memilih hanya paket dengan pemutakhiran dalam repo keamanan. Untuk setiap paket, versi kandidat paket (yang biasanya versi terbaru) harus merupakan bagian dari repo keamanan. Dalam hal ini, untukUbuntu Server, versi kandidat tambalan terbatas pada tambalan yang disertakan dalam repo berikut:

- Ubuntu Server14.04 LTS: `trusty-security`
- Ubuntu Server16.04 LTS: `xenial-security`
- Ubuntu Server18.04 LTS: `bionic-security`
- Ubuntu Server20.04 LTS: `focal-security`
- Ubuntu Server20.10 STR: `groovy-security`
- Ubuntu Server22,04 LTS () `jammy-security`
- Ubuntu Server23.04 () `lunar-security`

Jika pembaruan non-keamanan disertakan, patch dari repositori lain juga dipertimbangkan.

> **Note**
>
> Untuk informasi tentang format yang diterima untuk daftar patch yang disetujui dan patch yang ditolak, lihat Tentang format nama paket untuk daftar patch yang disetujui dan ditolak.

Untuk melihat konten bidang Prioritas dan Bagian, jalankan perintah `aptitude` berikut ini:

> **Note**
>
> Anda mungkin perlu menginstal Aptitude terlebih dahulu pada Ubuntu Server 16 sistem.

```
aptitude search -F '%p %P %s %t %V#' '~U'
```

Dalam menanggapi perintah ini, semua paket yang dapat dimutakhirkan dilaporkan dalam format ini:

```
name, priority, section, archive, candidate version
```

Untuk informasi tentang nilai status kepatuhan patch, lihat Memahami nilai keadaan kepatuhan patch.

Perbedaan utama antara patching Linux dan Windows

Topik ini menjelaskan perbedaan penting antara Linux dan Windows patching inPatch Manager, kemampuan. AWS Systems Manager

> **Note**
>
> Untuk menambal node yang dikelola Linux, node Anda harus menjalankan SSM Agent versi 2.0.834.0 atau yang lebih baru.
> Versi terbaru dirilis setiap kali kemampuan baru ditambahkan ke Systems Manager atau pembaruan dibuat untuk kemampuan yang ada. SSM Agent Gagal menggunakan agen versi terbaru dapat mencegah node terkelola Anda menggunakan berbagai kemampuan dan fitur Systems Manager. Untuk alasan itu, kami menyarankan Anda mengotomatiskan proses menjaga agar tetap SSM Agent up to date pada mesin Anda. Untuk informasi, lihat Mengotomatiskan pembaruan ke SSM Agent. Berlangganan halaman Catatan SSM Agent Rilis GitHub untuk mendapatkan pemberitahuan tentang SSM Agent pembaruan.

Perbedaan 1: Evaluasi patch

Linux

Untuk patch Linux, Systems Manager mengevaluasi aturan dasar patch dan daftar patch disetujui dan ditolak pada setiap node terkelola. Systems Manager harus mengevaluasi patching pada setiap node karena layanan mengambil daftar patch yang diketahui dan update dari repositori yang dikonfigurasi pada node terkelola.

Windows

Patch Managermenggunakan proses yang berbeda pada node yang dikelola Windows dan node yang dikelola Linux untuk mengevaluasi tambalan mana yang harus ada. Untuk patching Windows, Systems Manager mengevaluasi aturan dasar patch dan daftar patch yang disetujui dan ditolak pada secara langsung dalam layanan. Hal ini dapat dilakukan karena patch Windows ditarik dari satu repositori (Windows Update).

Perbedaan 2: **`Not Applicable`** tambalan

Karena banyaknya paket yang tersedia untuk sistem operasi Linux, Systems Manager tidak melaporkan detail tentang patch yang berstatus Tidak Berlaku. Patch `Not Applicable` adalah, sebagai contoh, sebuah patch untuk perangkat lunak Apache ketika instans tidak memiliki Apache yang sudah diinstal. Systems Manager melaporkan jumlah `Not Applicable` tambalan dalam ringkasan, tetapi jika Anda memanggil DescribeInstancePatchesAPI untuk node terkelola, data yang dikembalikan tidak menyertakan tambalan dengan status. `Not Applicable` Perilaku ini berbeda dari Windows.

Perbedaan 3: Dukungan dokumen SSM

Dokumen `AWS-ApplyPatchBaseline` Systems Manager (dokumen SSM) tidak mendukung node yang dikelola Linux. Untuk menerapkan baseline patch ke Linux,macOS, dan node Windows Server terkelola, dokumen SSM yang disarankan adalah. `AWS-RunPatchBaseline` Untuk informasi lebih lanjut, lihat Tentang dokumen SSM untuk patching node terkelola dan Tentang dokumen SSM `AWS-RunPatchBaseline`.

Perbedaan 4: Patch aplikasi

Fokus utama Patch Manager adalah menerapkan patch ke sistem operasi. Namun, Anda juga dapat menggunakan Patch Manager untuk menerapkan tambalan ke beberapa aplikasi pada node terkelola Anda.

Linux

Pada sistem operasi Linux, Patch Manager menggunakan repositori yang dikonfigurasi untuk pembaruan, dan tidak membedakan antara sistem operasi dan patch aplikasi. Anda dapat menggunakan Patch Manager untuk menentukan repositori mana untuk mengambil pembaruan dari. Untuk informasi selengkapnya, lihat Cara menentukan repositori sumber patch alternatif (Linux).

Windows

Pada node Windows Server terkelola, Anda dapat menerapkan aturan persetujuan, serta pengecualian patch yang Disetujui dan Ditolak, untuk aplikasi yang dirilis oleh Microsoft, seperti Microsoft Word 2016 dan Microsoft Exchange Server 2016. Untuk informasi selengkapnya, lihat Bekerja dengan dasar patch kustom.

Tentang dokumen SSM untuk patching node terkelola

Topik ini menjelaskan sembilan Systems Manager SSM (dokumen SSM) yang tersedia untuk membantu Anda menjaga node terkelola Anda di-patch dengan pembaruan terkait keamanan yang terbaru.

Kami merekomendasikan hanya menggunakan lima dokumen ini dalam operasi patching Anda. Bersama-sama, lima dokumen SSM ini memberi Anda berbagai macam opsi patching menggunakan AWS Systems Manager. Empat dari dokumen-dokumen ini dirilis belakangan daripada empat dokumen SSM warisan yang mereka gantikan dan mewakili ekspansi atau konsolidasi fungsi.

Lima dokumen SSM yang direkomendasikan meliputi:

- `AWS-ConfigureWindowsUpdate`
- `AWS-InstallWindowsUpdates`
- `AWS-RunPatchBaseline`
- `AWS-RunPatchBaselineAssociation`
- `AWS-RunPatchBaselineWithHooks`

Empat dokumen SSM warisan yang masih tersedia untuk digunakan dalam beberapaWilayah AWS, tetapi mungkin tidak lagi didukung di masa future, meliputi:

- `AWS-ApplyPatchBaseline`
- `AWS-FindWindowsUpdates`
- `AWS-InstallMissingWindowsUpdates`
- `AWS-InstallSpecificWindowsUpdates`

Rujuk ke bagian berikut ini untuk informasi selengkapnya tentang menggunakan dokumen SSM ini di operasi patching Anda.

Topik

- Tentang dokumen SSM AWS-RunPatchBaselineWithHooks
- Contoh skenario untuk menggunakan InstallOverrideList parameter diAWS-RunPatchBaseline atauAWS-RunPatchBaselineAssociation
- Menggunakan BaselineOverride parameter

Dokumen SSM yang direkomendasikan untuk patching node terkelola

Lima dokumen SSM berikut ini direkomendasikan untuk digunakan dalam operasi patching node terkelola Anda.

Dokumen SSM yang direkomendasikan

- AWS-ConfigureWindowsUpdate
- AWS-InstallWindowsUpdates
- AWS-RunPatchBaseline
- AWS-RunPatchBaselineAssociation
- AWS-RunPatchBaselineWithHooks

`AWS-ConfigureWindowsUpdate`

Support konfigurasi fungsi Windows Update dasar dan menggunakannya untuk menginstal pembaruan secara otomatis (atau mematikan pembaruan otomatis). Tersedia di semua Wilayah AWS.

Dokumen SSM ini meminta Windows Update untuk mengunduh dan menginstal pembaruan tertentu dan me-reboot node terkelola sesuai kebutuhan. Gunakan dokumen ini denganState Manager, suatu kemampuanAWS Systems Manager, untuk memastikan Windows Update mempertahankan konfigurasinya. Anda juga dapat menjalankannya secara manual menggunakanRun Command, suatu kemampuanAWS Systems Manager, untuk mengubah konfigurasi Windows Update.

Parameter yang tersedia dalam support dokumen ini menentukan kategori pembaruan untuk diinstal (atau apakah akan mematikan pembaruan otomatis), serta menentukan hari dan waktu untuk menjalankan operasi patching. Dokumen SSM ini sangat berguna jika Anda tidak memerlukan kendali ketat atas Windows Update dan tidak perlu mengumpulkan informasi kepatuhan.

Menggantikan dokumen SSM warisan:

- Tidak ada

AWS-InstallWindowsUpdates

Menginstal pembaruan pada node yangWindows Server dikelola. Tersedia di semua Wilayah AWS.

Dokumen SSM ini menyediakan fungsi patching dasar dalam kasus ketika Anda ingin menginstal pembaruan tertentu (menggunakan parameter `Include Kbs`), atau ingin menginstal patch dengan klasifikasi atau kategori tertentu tetapi tidak memerlukan informasi kepatuhan patch.

Menggantikan dokumen SSM warisan:

- `AWS-FindWindowsUpdates`
- `AWS-InstallMissingWindowsUpdates`
- `AWS-InstallSpecificWindowsUpdates`

Ketiga dokumen warisan melakukan fungsi yang berbeda, tetapi Anda dapat meraih hasil yang sama dengan menggunakan pengaturan parameter yang berbeda dengan dokumen SSM yang lebih baru `AWS-InstallWindowsUpdates`. Pengaturan parameter ini dijelaskan dalam Dokumen SSM warisan untuk patching node SSM.

AWS-RunPatchBaseline

Menginstal patch pada node terkelola Anda atau memindai node untuk menentukan apakah ada patch yang memenuhi syarat yang hilang. Tersedia di semua Wilayah AWS.

`AWS-RunPatchBaseline` memungkinkan Anda untuk mengendalikan persetujuan patch menggunakan dasar patch yang ditetapkan sebagai "default" untuk sebuah jenis sistem operasi. Melaporkan informasi kepatuhan patch yang dapat Anda lihat menggunakan alat Kepatuhan Systems Manager. Alat tersebut memberi Anda wawasan tentang status kepatuhan patch dari node terkelola Anda, seperti node mana yang kehilangan patch dan patch apa yang hilang. Saat Anda menggunakan `AWS-RunPatchBaseline`, informasi kepatuhan patch dicatat menggunakan perintah API `PutInventory`. Untuk sistem operasi Linux, informasi kepatuhan disediakan untuk patch dari repositori sumber default yang dikonfigurasi pada node terkelola dan juga dari repositori sumber alternatif lain yang Anda tentukan pada sebuah dasar patch kustom. Untuk informasi selengkapnya tentang repositori sumber alternatif, lihat Cara menentukan repositori sumber patch alternatif (Linux). Untuk informasi selengkapnya tentang alat Kepatuhan Systems Manager, lihat AWS Systems Manager Kepatuhan.

Menggantikan dokumen warisan:

- `AWS-ApplyPatchBaseline`

DokumenAWS-ApplyPatchBaseline warisan, dan tidak menyediakan support untuk patching aplikasi.Windows Server AWS-RunPatchBaseline yang lebih baru menyediakan support yang sama untuk sistem Windows dan Linux. Versi 2.0.834.0 atau yangSSM Agent lebih baru diperlukan untuk menggunakanAWS-RunPatchBaseline dokumen tersebut.

Untuk informasi lebih lanjut tentang dokumen SSM AWS-RunPatchBaseline, lihat Tentang dokumen SSM AWS-RunPatchBaseline.

AWS-RunPatchBaselineAssociation

Menginstal patch pada instans Anda atau memindai instans untuk menentukan apakah ada patch yang memenuhi syarat yang hilang. Tersedia di semua Wilayah AWS komersial.

AWS-RunPatchBaselineAssociation berbeda dengan AWS-RunPatchBaseline dalam beberapa hal penting:

- AWS-RunPatchBaselineAssociationdimaksudkan untuk digunakan terutama denganState Manager asosiasi yang dibuat menggunakanQuick Setup, suatu kemampuanAWS Systems Manager. Secara khusus, ketika Anda menggunakan jenis konfigurasiQuick Setup Host Management, jika Anda memilih opsi Pindai instans untuk patch yang hilang setiap hari, maka sistem menggunakanAWS-RunPatchBaselineAssociation untuk operasi tersebut.

  Namun, dalam kebanyakan kasus, saat menyiapkan operasi patching Anda sendiri, Anda harus memilih AWS-RunPatchBaseline atau AWS-RunPatchBaselineWithHooks sebagai ganti AWS-RunPatchBaselineAssociation.

  Untuk informasi selengkapnya, lihat topik berikut:

  - AWS Systems Manager Quick Setup
  - Tentang dokumen SSM AWS-RunPatchBaselineAssociation
- AWS-RunPatchBaselineAssociation mendukung penggunaan tag untuk mengidentifikasi dasar patch yang digunakan dengan serangkaian target ketika dijalankan.
- Untuk operasi patching yang menggunakanAWS-RunPatchBaselineAssociation, data kepatuhan patch dikumpulkan berdasarkanState Manager asosiasi tertentu. data kepatuhan patch dikumpulkan saat AWS-RunPatchBaselineAssociation berjalan dicatat menggunakan perintah API PutComplianceItems dan bukan perintah PutInventory. Hal ini mencegah data kepatuhan yang tidak terkait dengan asosiasi khusus ini ditimpa.

  Untuk sistem operasi Linux, informasi kepatuhan disediakan untuk patch dari repositori sumber default yang dikonfigurasi pada sebuah instans dan juga dari repositori sumber alternatif lain yang

Anda tentukan pada sebuah dasar patch kustom. Untuk informasi selengkapnya tentang repositori sumber alternatif, lihat Cara menentukan repositori sumber patch alternatif (Linux). Untuk informasi selengkapnya tentang alat Kepatuhan Systems Manager, lihat AWS Systems Manager Kepatuhan.

Menggantikan dokumen warisan:

- Tidak ada

Untuk informasi lebih lanjut tentang dokumen SSM `AWS-RunPatchBaselineAssociation`, lihat Tentang dokumen SSM `AWS-RunPatchBaselineAssociation`.

`AWS-RunPatchBaselineWithHooks`

Menginstal patch pada node SSM Anda atau memindai node SSM untuk menentukan apakah ada patch yang memenuhi syarat yang hilang, dengan kait opsional yang dapat Anda gunakan untuk menjalankan dokumen SSM pada tiga titik selama siklus patching. Tersedia di semua Wilayah AWS komersial.

`AWS-RunPatchBaselineWithHooks` berbeda dengan `AWS-RunPatchBaseline` dalam hal operasi `Install`.

`AWS-RunPatchBaselineWithHooks`mendukung kait siklus hidup yang berjalan pada titik-titik yang ditunjuk selama patching node terkelola. Karena instalasi patch kadang-kadang memerlukan node terkelola untuk di-reboot, operasi patching dibagi menjadi dua peristiwa, dengan total tiga kait yang support fungsi kustom. Kait pertama adalah sebelum operasi `Install with NoReboot`. Kait kedua adalah setelah operasi `Install with NoReboot`. Kait ketiga tersedia setelah me-reboot node.

Menggantikan dokumen warisan:

- Tidak ada

Untuk informasi lebih lanjut tentang dokumen SSM `AWS-RunPatchBaselineWithHooks`, lihat Tentang dokumen SSM `AWS-RunPatchBaselineWithHooks`.

Dokumen SSM warisan untuk patching node SSM

Empat dokumen SSM berikut ini masih tersedia untuk digunakan dalam operasi patching Anda di beberapa Wilayah AWS. Namun, file tersebut mungkin tidak lagi didukung di future, sehingga kami

tidak merekomendasikan penggunaannya. Sebagai gantinya, gunakan dokumen yang dijelaskan dalam Dokumen SSM yang direkomendasikan untuk patching node terkelola.

Dokumen SSM Warisan

- AWS-ApplyPatchBaseline
- AWS-FindWindowsUpdates
- AWS-InstallMissingWindowsUpdates
- AWS-InstallSpecificWindowsUpdates

AWS-ApplyPatchBaseline

Hanya support nodeWindows Server terkelola, tetapi tidak menyertakan support untuk patching aplikasi yang ditemukan dalam penggantinya,`AWS-RunPatchBaseline`. Tidak tersedia di Wilayah AWS yang diluncurkan setelah Agustus 2017.

> **Note**
>
> Pengganti dokumen SSM ini,`AWS-RunPatchBaseline`, memerlukan versi 2.0.834.0 atau versi yang lebih baruSSM Agent. Anda dapat menggunakan`AWS-UpdateSSMAgent` dokumen tersebut untuk memperbarui node terkelola Anda ke versi terbaru agen tersebut.

AWS-FindWindowsUpdates

Digantikan oleh `AWS-InstallWindowsUpdates`, yang dapat melakukan semua tindakan yang sama. Tidak tersedia di Wilayah AWS yang diluncurkan setelah April 2017.

Untuk mencapai hasil yang sama yang biasanya Anda dapatkan dari dokumen SSM warisan ini, gunakan konfigurasi parameter berikut ini dengan dokumen pengganti yang direkomendasikan, `AWS-InstallWindowsUpdates`:

- `Action` = `Scan`
- `Allow Reboot` = `False`

AWS-InstallMissingWindowsUpdates

Digantikan oleh `AWS-InstallWindowsUpdates`, yang dapat melakukan semua tindakan yang sama. Tidak tersedia di setiap Wilayah AWS yang diluncurkan setelah April 2017.

Untuk mencapai hasil yang sama yang biasanya Anda dapatkan dari dokumen SSM warisan ini, gunakan konfigurasi parameter berikut ini dengan dokumen pengganti yang direkomendasikan, `AWS-InstallWindowsUpdates`:

- `Action` = `Install`
- `Allow Reboot` = `True`

`AWS-InstallSpecificWindowsUpdates`

Digantikan oleh `AWS-InstallWindowsUpdates`, yang dapat melakukan semua tindakan yang sama. Tidak tersedia di setiap Wilayah AWS yang diluncurkan setelah April 2017.

Untuk mencapai hasil yang sama yang biasanya Anda dapatkan dari dokumen SSM warisan ini, gunakan konfigurasi parameter berikut ini dengan dokumen pengganti yang direkomendasikan, `AWS-InstallWindowsUpdates`:

- `Action` = `Install`
- `Allow Reboot` = `True`
- `Include Kbs` = *`daftar artikel KB yang dipisahkan koma`*

Tentang dokumen SSM **`AWS-RunPatchBaseline`**

AWS Systems Manager mendukung`AWS-RunPatchBaseline`, dokumen Systems Manager (dokumen SSM) untukPatch Manager, kemampuan. AWS Systems Manager Dokumen SSM ini melakukan operasi penambalan pada node terkelola untuk pembaruan terkait keamanan dan jenis pembaruan lainnya. Ketika dokumen dijalankan, ia menggunakan dasar patch yang ditetapkan sebagai "default" untuk suatu jenis sistem operasi jika tidak ada grup patch yang ditentukan. Jika tidak, ia menggunakan dasar patch yang terkait dengan grup patch. Untuk informasi tentang grup patch, lihat Tentang grup patch.

Anda dapat menggunakan `AWS-RunPatchBaseline` untuk menerapkan patch untuk sistem operasi dan aplikasi. (Pada Windows Server, support aplikasi dibatasi pada pembaruan untuk aplikasi yang dirilis oleh Microsoft.)

Dokumen ini mendukung Linux,macOS, dan node Windows Server terkelola. Dokumen ini akan melakukan tindakan yang sesuai untuk setiap platform.

> **Note**
>
> Patch Managerjuga mendukung dokumen SSM lama. `AWS-ApplyPatchBaseline` Namun, dokumen ini mendukung penambalan pada node yang dikelola Windows saja. Kami mendorong Anda untuk menggunakan `AWS-RunPatchBaseline` sebagai gantinya karena mendukung patching di Linux,macOS, dan node Windows Server terkelola. Versi 2.0.834.0 atau yang lebih baru SSM Agent diperlukan untuk menggunakan dokumen. `AWS-RunPatchBaseline`

Windows Server

Pada node Windows Server terkelola, `AWS-RunPatchBaseline` dokumen mengunduh dan memanggil PowerShell modul, yang pada gilirannya mengunduh snapshot dari baseline patch yang berlaku untuk node terkelola. snapshot dasar patch ini berisi daftar patch yang disetujui yang dikumpulkan dengan melakukan kueri dasar patch pada server Windows Server Update Services (WSUS). Daftar ini diteruskan ke API Windows Update, yang mengendalikan pengunduhan dan instalasi patch yang disetujui sesuai kebutuhan.

Linux

Pada node yang dikelola Linux, `AWS-RunPatchBaseline` dokumen tersebut memanggil modul Python, yang pada gilirannya mengunduh snapshot dari baseline patch yang berlaku untuk node terkelola. Snapshot dasar tambalan ini menggunakan aturan yang ditentukan dan daftar tambalan yang disetujui dan diblokir untuk mendorong manajer paket yang sesuai untuk setiap jenis node:

- Amazon Linux 1, Amazon Linux 2, CentOSOracle Linux, dan RHEL 7 node terkelola menggunakan YUM. Untuk operasi YUM, Patch Manager membutuhkan `Python 2.6` atau versi yang didukung yang lebih baru (2.6 - 3.10).
- RHEL8 node terkelola menggunakan DNF. Untuk operasi DNF, Patch Manager memerlukan versi yang didukung dari `Python 2` atau `Python 3` (2.6 - 3.10). (Tidak ada versi yang diinstal secara default pada RHEL 8. Anda harus menginstal satu atau yang lain secara manual.)
- Debian Server,Raspberry Pi OS, dan Ubuntu Server instance menggunakan APT. Untuk operasi APT, Patch Manager memerlukan versi yang didukung `Python 3` (3.0 - 3.10).
- SUSE Linux Enterprise Servernode terkelola menggunakan Zypper. Untuk operasi Zypper, Patch Manager membutuhkan `Python 2.6` atau versi yang didukung yang lebih baru (2.6 - 3.10).

macOS

Pada node macOS terkelola, `AWS-RunPatchBaseline` dokumen memanggil modul Python, yang pada gilirannya mengunduh snapshot dari baseline patch yang berlaku untuk node terkelola. Selanjutnya, subproses Python memanggil AWS Command Line Interface (AWS CLI) pada node untuk mengambil instalasi dan memperbarui informasi untuk manajer paket tertentu dan untuk mendorong manajer paket yang sesuai untuk setiap paket pembaruan.

Setiap snapshot khusus untuk, grup patch Akun AWS, sistem operasi, dan ID snapshot. Snapshot dikirimkan melalui URL Amazon Simple Storage Service (Amazon S3) yang telah ditandatangani sebelumnya, yang kedaluwarsa 24 jam setelah snapshot dibuat. Namun, setelah URL kedaluwarsa, jika Anda ingin menerapkan konten snapshot yang sama ke node terkelola lainnya, Anda dapat membuat URL Amazon S3 yang telah ditetapkan sebelumnya hingga 3 hari setelah snapshot dibuat. Untuk melakukannya, gunakan perintah get-deployable-patch-snapshot-for-instance.

Setelah semua pembaruan yang disetujui dan berlaku telah diinstal, dengan reboot dilakukan seperlunya, informasi kepatuhan tambalan dihasilkan pada node yang dikelola dan dilaporkan kembali kePatch Manager.

> **Note**
>
> Jika `RebootOption` parameter disetel ke `NoReboot` dalam `AWS-RunPatchBaseline` dokumen, node terkelola tidak di-boot ulang setelah Patch Manager dijalankan. Untuk informasi selengkapnya, lihat Nama parameter: `RebootOption`.

Untuk informasi tentang melihat data kepatuhan patch, lihat Tentang kepatuhan patch.

Parameter **`AWS-RunPatchBaseline`**

`AWS-RunPatchBaseline` support lima parameter. parameter `Operation` diperlukan. `RebootOption`Parameter `InstallOverrideListBaselineOverride`,, dan bersifat opsional. `Snapshot-ID`secara teknis opsional, tetapi kami menyarankan Anda memberikan nilai khusus untuk itu ketika Anda menjalankan `AWS-RunPatchBaseline` di luar jendela pemeliharaan. Patch Managerdapat memberikan nilai kustom secara otomatis ketika dokumen dijalankan sebagai bagian dari operasi jendela pemeliharaan.

Parameter

- Nama parameter: Operation
- Nama parameter: AssociationId
- Nama parameter: Snapshot ID
- Nama parameter: InstallOverrideList
- Nama parameter: RebootOption
- Nama parameter: BaselineOverride

Nama parameter: **`Operation`**

Penggunaan: Wajib.

Opsi: `Scan` | `Install`.

Scan

Saat Anda memilih `Scan` opsi, `AWS-RunPatchBaseline` tentukan status kepatuhan patch dari node terkelola dan laporkan informasi ini kembali kePatch Manager. `Scan`tidak meminta pembaruan untuk diinstal atau node yang dikelola untuk di-boot ulang. Sebaliknya, operasi mengidentifikasi di mana pembaruan hilang yang disetujui dan berlaku untuk node.

Menginstal

Ketika Anda memilih `Install` opsi, `AWS-RunPatchBaseline` mencoba untuk menginstal pembaruan yang disetujui dan berlaku yang hilang dari node terkelola. Informasi kepatuhan patch yang dihasilkan sebagai bagian operasi `Install` tidak mencantumkan pembaruan yang hilang, tetapi mungkin melaporkan pembaruan yang berstatus gagal jika instalasi pembaruan tidak berhasil karena alasan apa pun. Setiap kali pembaruan diinstal pada node terkelola, node di-reboot untuk memastikan pembaruan diinstal dan aktif. (Pengecualian: Jika `RebootOption` parameter disetel ke `NoReboot` dalam `AWS-RunPatchBaseline` dokumen, node terkelola tidak di-boot ulang setelah Patch Manager dijalankan. Untuk informasi lebih lanjut, lihatNama parameter: `RebootOption`.)

> **Note**
>
> Jika patch yang ditentukan oleh aturan dasar diinstal sebelum Patch Manager memperbarui node terkelola, sistem mungkin tidak reboot seperti yang diharapkan. Hal ini dapat terjadi ketika patch diinstal secara manual oleh pengguna atau diinstal secara otomatis oleh program lain, seperti `unattended-upgrades` paket aktifUbuntu Server.

Nama parameter: **`AssociationId`**

Penggunaan: Opsional.

`AssociationId`adalah ID dari asosiasi yang ada diState Manager, kemampuan AWS Systems Manager. Ini digunakan oleh Patch Manager untuk menambahkan data kepatuhan ke asosiasi tertentu. Asosiasi ini terkait dengan operasi penambalan yang diatur dalam kebijakan tambalan di Quick Setup.

> **Note**
>
> Dengan`AWS-RunPatchBaseline`, jika `AssociationId` nilai diberikan bersama dengan penggantian dasar kebijakan tambalan, penambalan dilakukan sebagai `PatchPolicy` operasi dan `ExecutionType` nilai yang dilaporkan juga. `AWS:ComplianceItem` `PatchPolicy` Jika tidak ada `AssociationId` nilai yang diberikan, penambalan dilakukan sebagai `Command` operasi dan laporan `ExecutionType` nilai pada yang `AWS:ComplianceItem` diserahkan juga`Command`.

Jika Anda belum memiliki asosiasi yang ingin Anda gunakan, Anda dapat membuatnya dengan menjalankan create-associationperintah.

Nama parameter: **`Snapshot ID`**

Penggunaan: Opsional.

`Snapshot ID`adalah ID unik (GUID) yang digunakan oleh Patch Manager untuk memastikan bahwa satu set node terkelola yang ditambal dalam satu operasi semuanya memiliki set patch yang disetujui yang sama persis. Meskipun parameter didefinisikan sebagai opsional, rekomendasi praktik terbaik kami bergantung pada apakah Anda menjalankan `AWS-RunPatchBaseline` atau tidak di jendela pemeliharaan, seperti yang dijelaskan dalam tabel berikut.

Praktik terbaik **`AWS-RunPatchBaseline`**

Mode	Praktik terbaik	Detail
Menjalankan `AWS-RunPatchBaseline` di dalam jendela pemeliharaan	Jangan berikan ID Snapshot. Patch Managerakan memasoknya untuk Anda.	Jika Anda menggunakan jendela pemeliharaan untuk menjalankan`AWS-RunPatchBaseline`, Anda

Mode	Praktik terbaik	Detail
		seharusnya tidak memberikan ID Snapshot yang dibuat sendiri. Dalam skenario ini, Systems Manager menyediakan nilai GUID berdasarkan ID eksekusi jendela pemeliharaan. Hal ini memastikan bahwa digunakan ID yang benar untuk semua pemanggilan `AWS-RunPatchBaseline` dalam jendela pemeliharaan tersebut. Jika Anda menentukan nilai dalam skenario ini, perhatikan bahwa snapshot dari baseline patch mungkin tidak tetap di tempatnya selama lebih dari 3 hari. Setelah itu, snapshot baru akan dihasilkan bahkan jika Anda menentukan ID yang sama setelah snapshot berakhir.

Mode	Praktik terbaik	Detail
Menjalankan `AWS-RunPatchBaseline` di luar jendela pemeliharaan	Menghasilkan dan menentukan nilai GUID kustom untuk ID Snapshot.[1]	Bila Anda tidak menggunakan jendela pemeliharaan untuk menjalankan `AWS-RunPatchBaseline`, kami sarankan Anda membuat dan menentukan ID Snapshot unik untuk setiap baseline patch, terutama jika Anda menjalankan `AWS-RunPatchBaseline` dokumen pada beberapa node terkelola dalam operasi yang sama. Jika Anda tidak menentukan ID dalam skenario ini, Systems Manager akan menghasilkan ID Snapshot yang berbeda untuk setiap node terkelola tempat perintah dikirim. Hal ini dapat mengakibatkan berbagai set patch yang ditentukan di antara node yang dikelola. Misalnya, katakanlah Anda menjalankan `AWS-RunPatchBaseline` dokumen secara langsung melaluiRun Command, kemampuan AWS Systems Manager, dan menargetkan sekelompok 50 node terkelola. Menentukan ID Snapshot kustom menghasilkan pembuatan snapshot dasar tunggal yang digunakan untuk mengevaluasi dan menambal semua node,

Mode	Praktik terbaik	Detail
		memastikan bahwa mereka berakhir dalam keadaan konsisten.

[1] Anda dapat menggunakan alat yang mampu menghasilkan GUID untuk menghasilkan nilai untuk parameter ID Snapshot. Misalnya, di PowerShell, Anda dapat menggunakan `New-Guid` cmdlet untuk menghasilkan GUID dalam format. `12345699-9405-4f69-bc5e-9315aEXAMPLE`

Nama parameter: **`InstallOverrideList`**

Penggunaan: Opsional.

Dengan menggunakan `InstallOverrideList`, Anda menentukan URL https atau URL ala jalur Amazon S3 ke daftar patch yang akan diinstal. Daftar instalasi patch ini, yang Anda pertahankan dalam format YAML, menggantikan patch yang ditentukan oleh dasar patch default saat ini. Ini memberi Anda kontrol yang lebih terperinci atas tambalan mana yang diinstal pada node terkelola Anda.

Sadarilah bahwa laporan kepatuhan mencerminkan status patch berdasarkan apa yang ditentukan dalam dasar patch, bukan apa yang Anda tentukan dalam daftar patch `InstallOverrideList`. Dengan kata lain, operasi Pemindaian mengabaikan parameter `InstallOverrideList`. Hal ini untuk memastikan bahwa laporan kepatuhan secara konsisten mencerminkan keadaan patch berdasarkan kebijakan daripada apa yang disetujui untuk operasi patching tertentu.

Untuk penjelasan tentang cara Anda dapat menggunakan parameter `InstallOverrideList` untuk menerapkan berbagai jenis patch berbeda ke sebuah grup target, pada jadwal jendela pemeliharaan yang berbeda, sembari tetap menggunakan dasar patch tunggal, lihat Contoh skenario untuk menggunakan InstallOverrideList parameter di`AWS-RunPatchBaseline` atau`AWS-RunPatchBaselineAssociation`.

Format URL yang valid

> **Note**
>
> Jika file Anda disimpan dalam bucket yang tersedia secara publik, Anda dapat menentukan format URL https atau URL ala jalur Amazon S3. Jika file Anda disimpan dalam bucket privat, Anda harus menentukan URL ala jalur Amazon S3.

- format URL https:

```
https://s3.aws-api-domain/DOC-EXAMPLE-BUCKET/my-windows-override-list.yaml
```

- URL ala jalur Amazon S3:

```
s3://DOC-EXAMPLE-BUCKET/my-windows-override-list.yaml
```

Format konten YAMM yang valid

Format yang Anda gunakan untuk menentukan tambalan dalam daftar Anda bergantung pada sistem operasi node terkelola Anda. Namun, format yang umum adalah seperti berikut ini:

```
patches:
    -
        id: '{patch-d}'
        title: '{patch-title}'
        {additional-fields}:{values}
```

Meskipun Anda dapat memberikan bidang tambahan dalam file YAML Anda, mereka diabaikan selama operasi patch.

Selain itu, kami merekomendasikan untuk memverifikasi bahwa format file YAML Anda valid sebelum menambahkan atau memperbarui daftar di bucket S3 Anda. Untuk informasi lebih lanjut tentang format YAML, lihat yaml.org. Untuk pilihan alat validasi, lakukan pencarian web untuk "validator format yaml".

Linux

id

Bidang id wajib diisi. Gunakan untuk menentukan patch menggunakan nama paket dan arsitektur. Sebagai contoh: `'dhclient.x86_64'`. Anda dapat menggunakan wildcard dalam id untuk menunjukkan lebih dari satu paket. Sebagai contoh: `'dhcp*'` dan `'dhcp*1.*'`.

Judul

Bidang judul bersifat opsional, tetapi pada sistem Linux bidang tersebut memberikan kemampuan filter tambahan. Jika Anda menggunakan judul, sebaiknya berisi informasi versi paket dalam salah satu format berikut:

YUM/SUSE Linux Enterprise Server (SLES):

```
{name}.{architecture}:{epoch}:{version}-{release}
```

APT

```
{name}.{architecture}:{version}
```

Untuk judul patch Linux, Anda dapat menggunakan satu atau lebih wildcard di posisi apa pun untuk memperluas jumlah kecocokan paket. Sebagai contoh: `'*32:9.8.2-0.*.rc1.57.amzn1'`.

Sebagai contoh:

- paket apt versi 1.2.25 saat ini diinstal pada node terkelola Anda, tetapi versi 1.2.27 sekarang tersedia.
- Anda menambahkan apt.amd64 versi 1.2.27 ke daftar patch. Hal ini tergantung pada apt utils.amd64 versi 1.2.27, tapi apt-utils.amd64 versi 1.2.25 ditentukan dalam daftar.

Dalam hal ini, apt versi 1.2.27 akan diblokir dari instalasi dan dilaporkan sebagai “Gagal-.” NonCompliant

Windows Server

id

Bidang id wajib diisi. Gunakan untuk menentukan patch menggunakan ID Microsoft Knowledge Base (misalnya, KB2736693) dan ID Microsoft Security Bulletin (misalnya, MS17-023).

Bidang lain yang ingin Anda berikan dalam daftar patch untuk Windows bersifat opsional dan hanya digunakan sebagai informasi Anda sendiri. Anda dapat menggunakan bidang tambahan

seperti judul, klasifikasi, Tingkat kepelikan, atau yang lainnya untuk memberikan informasi lebih detail tentang patch yang ditentukan.

macOS

id

Bidang id wajib diisi. Nilai untuk bidang id dapat diberikan menggunakan format `{package-name}.{package-version}` atau format {nama_paket}.

Contoh daftar tambalan

- 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'
```

Nama parameter: **`RebootOption`**

Penggunaan: Opsional.

Pilihan: `RebootIfNeeded` | `NoReboot`

Default: `RebootIfNeeded`

> **Warning**
>
> Opsi default-nya adalah `RebootIfNeeded`. Pastikan untuk memilih opsi yang benar untuk kasus penggunaan Anda. Misalnya, jika node terkelola Anda harus segera reboot untuk menyelesaikan proses konfigurasi, pilih`RebootIfNeeded`. Atau, jika Anda perlu mempertahankan ketersediaan node terkelola hingga waktu reboot yang dijadwalkan, pilih`NoReboot`.

> **Important**
>
> Kami tidak menyarankan penggunaan Patch Manager untuk menambal instance cluster di Amazon EMR (sebelumnya disebut Amazon Elastic). MapReduce Secara khusus, jangan pilih `RebootIfNeeded` opsi untuk `RebootOption` parameter. (Opsi ini tersedia dalam dokumen Perintah SSM untuk ditambal`AWS-RunPatchBaseline`,`AWS-RunPatchBaselineAssociation`, dan`AWS-RunPatchBaselineWithHooks`.) Perintah yang mendasari untuk menambal menggunakan Patch Manager penggunaan `yum` dan `dnf` perintah. Oleh karena itu, operasi mengakibatkan ketidakcocokan karena bagaimana paket diinstal. Untuk informasi tentang metode yang disukai untuk memperbarui perangkat lunak di klaster EMR Amazon, lihat Menggunakan default untuk AMI Amazon EMR di Panduan Manajemen EMR Amazon.

RebootIfNeeded

Saat Anda memilih `RebootIfNeeded` opsi, node terkelola di-boot ulang dalam salah satu kasus berikut:

- Patch Managermemasang satu atau lebih tambalan.

  Patch Managertidak mengevaluasi apakah reboot diperlukan oleh tambalan. Sistem di-boot ulang bahkan jika tambalan tidak memerlukan reboot.

- Patch Managermendeteksi satu atau lebih tambalan dengan status `INSTALLED_PENDING_REBOOT` selama operasi. `Install`

Status `INSTALLED_PENDING_REBOOT` dapat berarti bahwa opsi `NoReboot` dipilih saat terakhir kali operasi `Install` dijalankan.

> **Note**
>
> Patch yang dipasang di luar tidak pernah diberi status. Patch Manager `INSTALLED_PENDING_REBOOT`

Mem-boot ulang node terkelola dalam dua kasus ini memastikan bahwa paket yang diperbarui dikeluarkan dari memori dan menjaga perilaku patching dan reboot tetap konsisten di semua sistem operasi.

NoReboot

Ketika Anda memilih `NoReboot` opsi, Patch Manager tidak me-reboot node terkelola bahkan jika itu menginstal tambalan selama `Install` operasi. Opsi ini berguna jika Anda tahu bahwa node terkelola Anda tidak memerlukan reboot setelah tambalan diterapkan, atau Anda memiliki aplikasi atau proses yang berjalan pada node yang seharusnya tidak terganggu oleh reboot operasi patching. Ini juga berguna ketika Anda ingin lebih banyak kontrol atas waktu reboot node terkelola, seperti dengan menggunakan jendela pemeliharaan.

> **Note**
>
> Jika Anda memilih opsi `NoReboot` dan sebuah patch diinstal, patch diberikan status `InstalledPendingReboot`. Node yang dikelola itu sendiri, bagaimanapun, ditandai sebagai`Non-Compliant`. Setelah reboot terjadi dan `Scan` operasi dijalankan, status node terkelola diperbarui ke`Compliant`.

File pelacakan instalasi patch: Untuk melacak instalasi patch, terutama patch yang diinstal sejak reboot sistem terakhir, Systems Manager memelihara file pada node terkelola.

> **Important**
>
> Jangan menghapus atau memodifikasi file pelacakan. Jika file ini dihapus atau rusak, laporan kepatuhan patch untuk node terkelola tidak akurat. Jika ini terjadi, reboot node dan jalankan operasi Pindai patch untuk memulihkan file.

File pelacakan ini disimpan di lokasi berikut pada node terkelola Anda:

- Sistem operasi Linux:
  - `/var/log/amazon/ssm/patch-configuration/patch-states-configuration.json`
  - `/var/log/amazon/ssm/patch-configuration/patch-inventory-from-last-operation.json`
- Sistem operasi Windows Server:
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchStatesConfiguration.json`
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchInventoryFromLastOperation.json`

Nama parameter: **`BaselineOverride`**

Penggunaan: Opsional.

Anda dapat menentukan preferensi patching pada saat runtime menggunakan parameter `BaselineOverride`. Baseline override ini disimpan sebagai objek JSON dalam bucket S3. Ini memastikan operasi patching menggunakan dasar yang disediakan yang cocok dengan sistem operasi host bukannya menerapkan aturan dari dasar patch default

Untuk informasi selengkapnya tentang cara menggunakan parameter `BaselineOverride`, lihat Menggunakan BaselineOverride parameter.

Tentang dokumen SSM **`AWS-RunPatchBaselineAssociation`**

Seperti dokumen `AWS-RunPatchBaseline`, `AWS-RunPatchBaselineAssociation` melakukan operasi patching pada instans untuk jenis pembaruan terkait keamanan dan jenis pembaruan lainnya. Anda juga dapat menggunakan dokumen `AWS-RunPatchBaselineAssociation` untuk menerapkan patch untuk sistem operasi dan aplikasi. (Pada Windows Server, support aplikasi dibatasi pada pembaruan untuk aplikasi yang dirilis oleh Microsoft.)

Dokumen ini mendukung instans Amazon Elastic Compute Cloud (Amazon EC2) untuk Linux, macOS, dan Windows Server. Ini tidak mendukung node non-EC2 di lingkungan hybrid dan multicloud. Dokumen akan melakukan tindakan yang sesuai untuk setiap platform, menjalankan modul Python di Linux macOS dan instance, dan PowerShell modul pada instance Windows.

Namun, AWS-RunPatchBaselineAssociation berbeda dari AWS-RunPatchBaseline dengan cara berikut:

- AWS-RunPatchBaselineAssociationdimaksudkan untuk digunakan terutama dengan State Manager asosiasi yang dibuat menggunakan Quick Setup, kemampuan AWS Systems Manager. Secara khusus, ketika Anda menggunakan jenis konfigurasi Manajemen Quick Setup Host, jika Anda memilih opsi Pindai instance untuk tambalan yang hilang setiap hari, sistem menggunakan AWS-RunPatchBaselineAssociation untuk operasi.

  Namun, dalam kebanyakan kasus, saat menyiapkan operasi patching Anda sendiri, Anda harus memilih AWS-RunPatchBaseline atau AWS-RunPatchBaselineWithHooks sebagai ganti AWS-RunPatchBaselineAssociation.

- Saat Anda menggunakan dokumen AWS-RunPatchBaselineAssociation, Anda dapat menentukan pasangan kunci tag dalam bidang parameter BaselineTags pada dokumen. Jika baseline patch kustom di Anda Akun AWS membagikan tag ini, kemampuan Patch Manager AWS Systems Manager, menggunakan baseline yang diberi tag saat berjalan pada instance target alih-alih baseline patch “default” yang saat ini ditentukan untuk jenis sistem operasi.

  > **Important**
  >
  > Jika Anda memilih untuk menggunakan AWS-RunPatchBaselineAssociation dalam operasi penambalan selain yang disiapkan menggunakanQuick Setup, dan Anda ingin menggunakan BaselineTags parameter opsionalnya, Anda harus memberikan beberapa izin tambahan ke profil instans untuk instans Amazon Elastic Compute Cloud (Amazon EC2). Untuk informasi selengkapnya, lihat Nama parameter: BaselineTags.

  Kedua format berikut ini valid untuk parameter BaselineTags Anda:

  ```
  Key=tag-key,Values=tag-value
  ```

  ```
  Key=tag-key,Values=tag-value1,tag-value2,tag-value3
  ```

- Saat AWS-RunPatchBaselineAssociation berjalan, data kepatuhan patch yang dikumpulkan dicatat menggunakan perintah API PutComplianceItems bukannya perintah PutInventory, yang digunakan oleh AWS-RunPatchBaseline. Perbedaan ini berarti bahwa informasi kepatuhan patch yang disimpan dan dilaporkan per asosiasi spesifik. data kepatuhan patch yang dihasilkan di luar asosiasi ini tidak ditimpa.

- Informasi kepatuhan patch yang dilaporkan setelah menjalankan `AWS-RunPatchBaselineAssociation`menunjukkan apakah suatu instans patuh atau tidak. Itu tidak termasuk detail tingkat tambalan, seperti yang ditunjukkan oleh output dari perintah AWS Command Line Interface (AWS CLI) berikut. Filter perintah pada `Association` sebagai tipe kepatuhan:

```
aws ssm list-compliance-items \
    --resource-ids "i-02573cafcfEXAMPLE" \
    --resource-types "ManagedInstance" \
    --filters "Key=ComplianceType,Values=Association,Type=EQUAL" \
    --region us-east-2
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "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"
        }
    ]
}
```

Jika nilai key pair tag telah ditentukan sebagai parameter untuk `AWS-RunPatchBaselineAssociation` dokumen, Patch Manager mencari baseline patch kustom yang cocok dengan jenis sistem operasi dan telah ditandai dengan pasangan tag-key yang sama. Pencarian ini tidak terbatas pada dasar patch default yang ditetapkan saat ini atau dasar yang

ditetapkan ke grup patch. Jika tidak ada garis dasar yang ditemukan dengan tag yang ditentukan, Patch Manager selanjutnya mencari grup tambalan, jika salah satu ditentukan dalam perintah yang berjalan. `AWS-RunPatchBaselineAssociation` Jika tidak ada grup patch yang cocok, Patch Manager kembali ke baseline patch default saat ini untuk akun sistem operasi.

Jika lebih dari satu baseline patch ditemukan dengan tag yang ditentukan dalam `AWS-RunPatchBaselineAssociation` dokumen, Patch Manager mengembalikan pesan kesalahan yang menunjukkan bahwa hanya satu baseline patch yang dapat ditandai dengan pasangan kunci-nilai agar operasi dapat dilanjutkan.

> **Note**
>
> Pada instans Linux, pengelola paket yang sesuai untuk setiap tipe instans digunakan untuk menginstal paket:
>
> - Amazon Linux 1, Amazon Linux 2, CentOSOracle Linux, dan RHEL instans menggunakan YUM. Untuk operasi YUM, Patch Manager membutuhkan `Python 2.6` atau versi yang didukung yang lebih baru (2.6 - 3.10).
> - Debian Server,Raspberry Pi OS, dan Ubuntu Server instance menggunakan APT. Untuk operasi APT, Patch Manager memerlukan versi yang didukung `Python 3` (3.0 - 3.10).
> - Instans SUSE Linux Enterprise Server menggunakan Zypper. Untuk operasi Zypper, Patch Manager membutuhkan `Python 2.6` atau versi yang didukung yang lebih baru (2.6 - 3.10).

Setelah pemindaian selesai, atau setelah semua pembaruan yang disetujui dan berlaku telah diinstal, dengan reboot dilakukan seperlunya, informasi kepatuhan patch dihasilkan pada sebuah instans dan dilaporkan kembali ke layanan Patch Manager.

> **Note**
>
> Jika `RebootOption` parameter disetel ke `NoReboot` dalam `AWS-RunPatchBaselineAssociation` dokumen, instance tidak di-boot ulang setelah Patch Manager dijalankan. Untuk informasi selengkapnya, lihat Nama parameter: `RebootOption`.

Untuk informasi tentang melihat data kepatuhan patch, lihat Tentang kepatuhan patch.

Parameter **AWS-RunPatchBaselineAssociation**

`AWS-RunPatchBaselineAssociation` mendukung empat parameter. Parameter `Operation` dan `AssociationId` diperlukan. Parameter `InstallOverrideList`, `RebootOption`, dan `BaselineTags` bersifat opsional.

Parameter

- Nama parameter: Operation
- Nama parameter: BaselineTags
- Nama parameter: AssociationId
- Nama parameter: InstallOverrideList
- Nama parameter: RebootOption

Nama parameter: **Operation**

Penggunaan: Wajib.

Opsi: `Scan` | `Install`.

Scan

Saat Anda memilih `Scan` opsi, `AWS-RunPatchBaselineAssociation` tentukan status kepatuhan patch dari instance dan laporkan informasi ini kembali kePatch Manager. `Scan`tidak meminta pembaruan untuk diinstal atau instance untuk di-boot ulang. Sebaliknya, operasi ini mengidentifikasi keberadaan pembaruan hilang yang disetujui dan dapat diterapkan ke instans tersebut.

Pasang

Saat Anda memilih opsi `Install`, `AWS-RunPatchBaselineAssociation` mencoba untuk menginstal pembaruan yang disetujui dan dapat diterapkan yang hilang dari instans tersebut. Informasi kepatuhan patch yang dihasilkan sebagai bagian operasi `Install` tidak mencantumkan pembaruan yang hilang, tetapi mungkin melaporkan pembaruan yang berstatus gagal jika instalasi pembaruan tidak berhasil karena alasan apa pun. Setiap kali pembaruan diinstal pada sebuah instans, instans tersebut di-reboot untuk memastikan pembaruan telah terinstal dan aktif. (Pengecualian: Jika `RebootOption` parameter disetel ke `NoReboot` dalam `AWS-RunPatchBaselineAssociation` dokumen, instance tidak di-boot ulang setelah Patch Manager dijalankan. Untuk informasi lebih lanjut, lihatNama parameter: `RebootOption`.)

> **Note**
>
> Jika patch yang ditentukan oleh aturan dasar diinstal sebelum Patch Manager memperbarui instance, sistem mungkin tidak reboot seperti yang diharapkan. Hal ini dapat terjadi ketika patch diinstal secara manual oleh pengguna atau diinstal secara otomatis oleh program lain, seperti `unattended-upgrades` paket aktifUbuntu Server.

Nama parameter: **`BaselineTags`**

Penggunaan: Opsional.

`BaselineTags` adalah pasangan nilai kunci tag unik yang Anda pilih dan tetapkan ke dasar patch kustom individu. Anda dapat menentukan satu atau lebih nilai untuk parameter ini. Kedua format berikut ini valid:

`Key=`*`tag-key`*`,Values=`*`tag-value`*

`Key=`*`tag-key`*`,Values=`*`tag-value1`*`,`*`tag-value2`*`,`*`tag-value3`*

`BaselineTags`Nilai ini digunakan oleh Patch Manager untuk memastikan bahwa satu set instance yang ditambal dalam satu operasi semuanya memiliki set patch yang disetujui yang sama persis. Saat operasi penambalan berjalan, Patch Manager periksa untuk melihat apakah garis dasar tambalan untuk jenis sistem operasi ditandai dengan pasangan nilai kunci yang sama yang Anda tentukan. `BaselineTags` Jika ada kecocokan, dasar patch kustom ini digunakan. Jika tidak ada kecocokan, dasar patch diidentifikasi menurut grup patch yang ditentukan untuk operasi patching tersebut. Jika tidak ada, baseline patch standar AWS terkelola untuk sistem operasi itu digunakan.

Persyaratan izin tambahan

Jika Anda menggunakan `AWS-RunPatchBaselineAssociation` dalam operasi penambalan selain yang disiapkan menggunakanQuick Setup, dan Anda ingin menggunakan `BaselineTags` parameter opsional, Anda harus menambahkan izin berikut ke profil instans untuk instans Amazon Elastic Compute Cloud (Amazon EC2).

> **Note**
>
> Quick Setupdan `AWS-RunPatchBaselineAssociation` tidak mendukung server lokal dan mesin virtual (VM).

```
{
    "Effect": "Allow",
    "Action": [
        "ssm:DescribePatchBaselines",
        "tag:GetResources"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "ssm:GetPatchBaseline",
        "ssm:DescribeEffectivePatchesForPatchBaseline"
    ],
    "Resource": "patch-baseline-arn"
}
```

Ganti *patch-baseline-arn*dengan Nama Sumber Daya Amazon (ARN) dari baseline patch yang ingin Anda berikan akses, dalam format. `arn:aws:ssm:us-east-2:123456789012:patchbaseline/pb-0c10e65780EXAMPLE`

Nama parameter: **`AssociationId`**

Penggunaan: Wajib.

`AssociationId`adalah ID dari asosiasi yang ada diState Manager, kemampuan AWS Systems Manager. Ini digunakan oleh Patch Manager untuk menambahkan data kepatuhan ke asosiasi tertentu. Asosiasi ini terkait dengan `Scan` operasi tambalan yang diaktifkan dalam konfigurasi Manajemen Host yang dibuat di Quick Setup. Dengan mengirimkan hasil patching sebagai data kepatuhan asosiasi, bukan data kepatuhan inventaris, informasi kepatuhan inventaris yang ada untuk instans Anda tidak ditimpa setelah operasi patching, atau untuk ID asosiasi lainnya. Jika Anda belum memiliki asosiasi yang ingin Anda gunakan, Anda dapat membuatnya dengan menjalankan create-associationperintah. Sebagai contoh:

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
```

Nama parameter: **`InstallOverrideList`**

Penggunaan: Opsional.

Dengan menggunakan `InstallOverrideList`, Anda menentukan URL https atau URL ala jalur Amazon Simple Storage Service (Amazon S3) ke daftar patch yang akan diinstal. Daftar instalasi patch ini, yang Anda pertahankan dalam format YAML, menggantikan patch yang ditentukan oleh dasar patch default saat ini. Hal ini memberikan Anda kendali yang lebih terperinci atas patch apa yang diinstal pada instans Anda.

Sadarilah bahwa laporan kepatuhan mencerminkan status patch berdasarkan apa yang ditentukan dalam dasar patch, bukan apa yang Anda tentukan dalam daftar patch `InstallOverrideList`. Dengan kata lain, operasi Pemindaian mengabaikan parameter `InstallOverrideList`. Hal ini untuk memastikan bahwa laporan kepatuhan secara konsisten mencerminkan keadaan patch berdasarkan kebijakan daripada apa yang disetujui untuk operasi patching tertentu.

Format URL yang valid

> **Note**
>
> Jika file Anda disimpan dalam bucket yang tersedia secara publik, Anda dapat menentukan format URL https atau URL ala jalur Amazon S3. Jika file Anda disimpan dalam bucket privat, Anda harus menentukan URL ala jalur Amazon S3.

- Contoh format URL https:

```
https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/my-windows-override-list.yaml
```

- Contoh URL gaya jalur Amazon S3:

```
s3://DOC-EXAMPLE-BUCKET/my-windows-override-list.yaml
```

Format konten YAMM yang valid

Format yang Anda gunakan untuk menentukan patch dalam daftar Anda tergantung pada sistem operasi instans Anda. Namun, format yang umum adalah seperti berikut ini:

```
patches:
    -
        id: '{patch-d}'
        title: '{patch-title}'
        {additional-fields}:{values}
```

Meskipun Anda dapat memberikan bidang tambahan dalam file YAML Anda, mereka diabaikan selama operasi patch.

Selain itu, kami merekomendasikan untuk memverifikasi bahwa format file YAML Anda valid sebelum menambahkan atau memperbarui daftar di bucket S3 Anda. Untuk informasi lebih lanjut tentang format YAML, lihat yaml.org. Untuk pilihan alat validasi, lakukan pencarian web untuk "validator format yaml".

- Microsoft Windows

  id

Bidang id wajib diisi. Gunakan untuk menentukan patch menggunakan ID Microsoft Knowledge Base (misalnya, KB2736693) dan ID Microsoft Security Bulletin (misalnya, MS17-023).

Bidang lain yang ingin Anda berikan dalam daftar patch untuk Windows bersifat opsional dan hanya digunakan sebagai informasi Anda sendiri. Anda dapat menggunakan bidang tambahan seperti judul, klasifikasi, Tingkat kepelikan, atau yang lainnya untuk memberikan informasi lebih detail tentang patch yang ditentukan.

- Linux

  id

  Bidang id wajib diisi. Gunakan untuk menentukan patch menggunakan nama paket dan arsitektur. Sebagai contoh: `'dhclient.x86_64'`. Anda dapat menggunakan wildcard dalam id untuk menunjukkan lebih dari satu paket. Sebagai contoh: `'dhcp*'` dan `'dhcp*1.*'`.

  title

  Bidang judul bersifat opsional, tetapi pada sistem Linux bidang tersebut memberikan kemampuan filter tambahan. Jika Anda menggunakan judul, sebaiknya berisi informasi versi paket dalam salah satu format berikut:

  YUM/SUSE Linux Enterprise Server (SLES):

  ```
  {name}.{architecture}:{epoch}:{version}-{release}
  ```

  APT

  ```
  {name}.{architecture}:{version}
  ```

  Untuk judul patch Linux, Anda dapat menggunakan satu atau lebih wildcard di posisi apa pun untuk memperluas jumlah kecocokan paket. Sebagai contoh: `'*32:9.8.2-0.*.rc1.57.amzn1'`.

  Sebagai contoh:

  - Paket apt versi 1.2.25 saat ini telah diinstal pada instans Anda, tetapi versi 1.2.27 sekarang telah tersedia.
  - Anda menambahkan apt.amd64 versi 1.2.27 ke daftar patch. Hal ini tergantung pada apt utils.amd64 versi 1.2.27, tapi apt-utils.amd64 versi 1.2.25 ditentukan dalam daftar.

Dalam hal ini, apt versi 1.2.27 akan diblokir dari instalasi dan dilaporkan sebagai “Gagal-.” NonCompliant

Bidang Lainnya

Bidang lain yang ingin Anda berikan dalam daftar patch untuk Linux bersifat opsional dan hanya digunakan sebagai informasi Anda sendiri. Anda dapat menggunakan bidang tambahan seperti klasifikasi, tingkat kepelikan, atau yang lainnya untuk memberikan informasi lebih detail tentang patch yang ditentukan.

Contoh daftar tambalan

- 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'
```

Nama parameter: **`RebootOption`**

Penggunaan: Opsional.

Pilihan: `RebootIfNeeded` | `NoReboot`

Default: `RebootIfNeeded`

> **Warning**
>
> Opsi default-nya adalah `RebootIfNeeded`. Pastikan untuk memilih opsi yang benar untuk kasus penggunaan Anda. Misalnya, jika instance Anda harus segera reboot untuk menyelesaikan proses konfigurasi, pilih`RebootIfNeeded`. Atau, jika Anda perlu mempertahankan ketersediaan instance hingga waktu reboot yang dijadwalkan, pilih`NoReboot`.

> **Important**
>
> Kami tidak menyarankan penggunaan Patch Manager untuk menambal instance cluster di Amazon EMR (sebelumnya disebut Amazon Elastic). MapReduce Secara khusus, jangan pilih `RebootIfNeeded` opsi untuk `RebootOption` parameter. (Opsi ini tersedia dalam dokumen Perintah SSM untuk ditambal`AWS-RunPatchBaseline`,`AWS-RunPatchBaselineAssociation`, dan`AWS-RunPatchBaselineWithHooks`.) Perintah yang mendasari untuk menambal menggunakan Patch Manager penggunaan `yum` dan `dnf` perintah. Oleh karena itu, operasi mengakibatkan ketidakcocokan karena bagaimana paket diinstal. Untuk informasi tentang metode yang disukai untuk memperbarui

perangkat lunak di klaster EMR Amazon, lihat Menggunakan default untuk AMI Amazon EMR di Panduan Manajemen EMR Amazon.

RebootIfNeeded

Saat Anda memilih `RebootIfNeeded` opsi, instance di-boot ulang dalam salah satu kasus berikut:

- Patch Managermemasang satu atau lebih tambalan.

  Patch Managertidak mengevaluasi apakah reboot diperlukan oleh tambalan. Sistem di-boot ulang bahkan jika tambalan tidak memerlukan reboot.
- Patch Managermendeteksi satu atau lebih tambalan dengan status `INSTALLED_PENDING_REBOOT` selama operasi. `Install`

  Status `INSTALLED_PENDING_REBOOT` dapat berarti bahwa opsi `NoReboot` dipilih saat terakhir kali operasi `Install` dijalankan.

  > **Note**
  >
  > Patch yang dipasang di luar tidak pernah diberi status. Patch Manager `INSTALLED_PENDING_REBOOT`

Mem-boot ulang instance dalam dua kasus ini memastikan bahwa paket yang diperbarui dikeluarkan dari memori dan menjaga perilaku patching dan reboot tetap konsisten di semua sistem operasi.

NoReboot

Ketika Anda memilih `NoReboot` opsi, Patch Manager tidak me-reboot sebuah instance bahkan jika itu menginstal patch selama `Install` operasi. Opsi ini berguna jika Anda tahu bahwa instans Anda tidak memerlukan reboot setelah patch diterapkan, atau Anda memiliki aplikasi atau proses yang berjalan pada instans yang seharusnya tidak terganggu oleh reboot operasi patching. Hal ini juga berguna ketika Anda ingin memiliki kendali lebih besar atas waktu reboot instans, seperti dengan menggunakan jendela pemeliharaan.

File pelacakan instalasi patch: Untuk melacak instalasi patch, terutama patch yang telah diinstal sejak reboot sistem terakhir kali, Systems Manager mempertahankan file pada instans terkelola.

> **Important**
>
> Jangan menghapus atau memodifikasi file pelacakan. Jika file ini dihapus atau rusak, laporan kepatuhan patch untuk instans tidak akurat. Jika ini terjadi, reboot instans dan jalankan patch operasi Pemindaian untuk memulihkan file tersebut.

file pelacakan ini disimpan di lokasi-lokasi berikut ini pada instans terkelola Anda:

- Sistem operasi Linux:
  - `/var/log/amazon/ssm/patch-configuration/patch-states-configuration.json`
  - `/var/log/amazon/ssm/patch-configuration/patch-inventory-from-last-operation.json`
- Sistem operasi Windows Server:
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchStatesConfiguration.json`
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchInventoryFromLastOperation.json`

Tentang dokumen SSM **`AWS-RunPatchBaselineWithHooks`**

AWS Systems Manager mendukung`AWS-RunPatchBaselineWithHooks`, dokumen Systems Manager (dokumen SSM) untukPatch Manager, kemampuan. AWS Systems Manager Dokumen SSM ini melakukan operasi penambalan pada node terkelola untuk pembaruan terkait keamanan dan jenis pembaruan lainnya.

`AWS-RunPatchBaselineWithHooks` berbeda dari `AWS-RunPatchBaseline` dengan cara berikut:

- Dokumen pembungkus – `AWS-RunPatchBaselineWithHooks` adalah pembungkus untuk `AWS-RunPatchBaseline` dan bergantung pada `AWS-RunPatchBaseline` untuk beberapa operasinya.
- **`Install`**Operasi — `AWS-RunPatchBaselineWithHooks` mendukung kait siklus hidup yang berjalan pada titik yang ditentukan selama patch node terkelola. Karena instalasi patch kadang-kadang memerlukan node terkelola untuk reboot, operasi patching dibagi menjadi dua peristiwa, dengan total tiga kait yang mendukung fungsionalitas kustom. Kait pertama adalah

sebelum operasi `Install with NoReboot`. Kait kedua adalah setelah operasi `Install with NoReboot`. Kait ketiga tersedia setelah reboot dari node terkelola.

- Tidak ada support daftar patch kustom – `AWS-RunPatchBaselineWithHooks` tidak support parameter `InstallOverrideList`.
- SSM Agentdukungan - `AWS-RunPatchBaselineWithHooks` mengharuskan SSM Agent 3.0.502 atau yang lebih baru diinstal pada node yang dikelola untuk menambal.

Saat dokumen dijalankan, ia menggunakan dasar patch yang ditentukan saat ini sebagai "default" untuk suatu jenis sistem operasi jika tidak ada grup patch yang ditentukan. Jika tidak, ia menggunakan dasar patch yang terkait dengan grup patch. Untuk informasi tentang grup patch, lihat Tentang grup patch.

Anda dapat menggunakan `AWS-RunPatchBaselineWithHooks` untuk menerapkan patch untuk sistem operasi dan aplikasi. (Pada Windows, support aplikasi dibatasi pada pembaruan untuk aplikasi yang dirilis oleh Microsoft.)

Dokumen ini mendukung Linux,macOS, dan node Windows Server terkelola. Dokumen ini akan melakukan tindakan yang sesuai untuk setiap platform.

Linux

Pada node yang dikelola Linux, `AWS-RunPatchBaselineWithHooks` dokumen tersebut memanggil modul Python, yang pada gilirannya mengunduh snapshot dari baseline patch yang berlaku untuk node terkelola. Snapshot dasar tambalan ini menggunakan aturan yang ditentukan dan daftar tambalan yang disetujui dan diblokir untuk mendorong manajer paket yang sesuai untuk setiap jenis node:

- Amazon Linux 1, Amazon Linux 2, CentOSOracle Linux, dan RHEL 7 node terkelola menggunakan YUM. Untuk operasi YUM, Patch Manager membutuhkan `Python 2.6` atau versi yang didukung yang lebih baru (2.6 - 3.10).
- RHEL8 node terkelola menggunakan DNF. Untuk operasi DNF, Patch Manager memerlukan versi yang didukung dari `Python 2` atau `Python 3` (2.6 - 3.10). (Tidak ada versi yang diinstal secara default pada RHEL 8. Anda harus menginstal satu atau yang lain secara manual.)
- Debian Server,Raspberry Pi OS, dan Ubuntu Server instance menggunakan APT. Untuk operasi APT, Patch Manager memerlukan versi yang didukung `Python 3` (3.0 - 3.10).

- SUSE Linux Enterprise Servernode terkelola menggunakan Zypper. Untuk operasi Zypper, Patch Manager membutuhkan `Python 2.6` atau versi yang didukung yang lebih baru (2.6 - 3.10).

macOS

Pada node macOS terkelola, `AWS-RunPatchBaselineWithHooks` dokumen memanggil modul Python, yang pada gilirannya mengunduh snapshot dari baseline patch yang berlaku untuk node terkelola. Selanjutnya, subproses Python memanggil CLI pada node untuk mengambil instalasi dan memperbarui informasi untuk manajer paket yang ditentukan dan untuk mendorong manajer paket yang sesuai untuk setiap paket pembaruan.

Windows Server

Pada node Windows Server terkelola, `AWS-RunPatchBaselineWithHooks` dokumen mengunduh dan memanggil PowerShell modul, yang pada gilirannya mengunduh snapshot dari baseline patch yang berlaku untuk node terkelola. snapshot dasar patch ini berisi daftar patch yang disetujui yang dikumpulkan dengan melakukan kueri dasar patch pada server Windows Server Update Services (WSUS). Daftar ini diteruskan ke API Windows Update, yang mengendalikan pengunduhan dan instalasi patch yang disetujui sesuai kebutuhan.

Setiap snapshot khusus untuk, grup patch Akun AWS, sistem operasi, dan ID snapshot. Snapshot dikirimkan melalui URL Amazon Simple Storage Service (Amazon S3) yang telah ditandatangani sebelumnya, yang kedaluwarsa 24 jam setelah snapshot dibuat. Namun, setelah URL kedaluwarsa, jika Anda ingin menerapkan konten snapshot yang sama ke node terkelola lainnya, Anda dapat membuat URL Amazon S3 yang telah ditetapkan sebelumnya hingga tiga hari setelah snapshot dibuat. Untuk melakukannya, gunakan perintah get-deployable-patch-snapshot-for-instance.

Setelah semua pembaruan yang disetujui dan berlaku telah diinstal, dengan reboot dilakukan seperlunya, informasi kepatuhan tambalan dihasilkan pada node yang dikelola dan dilaporkan kembali kePatch Manager.

> **Note**
>
> Jika `RebootOption` parameter disetel ke `NoReboot` dalam `AWS-RunPatchBaselineWithHooks` dokumen, node terkelola tidak di-boot ulang setelah Patch Manager dijalankan. Untuk informasi selengkapnya, lihat Nama parameter: `RebootOption`.

Untuk informasi tentang melihat data kepatuhan patch, lihat Tentang kepatuhan patch.

Langkah-langkah operasional **AWS-RunPatchBaselineWithHooks**

Saat AWS-RunPatchBaselineWithHooks berjalan, langkah-langkah berikut dilakukan:

1. Pindai - Scan Operasi yang AWS-RunPatchBaseline digunakan dijalankan pada node terkelola, dan laporan kepatuhan dibuat dan diunggah.
2. Verifikasi status patch lokal - Script dijalankan untuk menentukan langkah-langkah apa yang akan dilakukan berdasarkan operasi yang dipilih dan hasil Scan dari Langkah 1.
   a. Jika operasi yang dipilih adalah Scan, operasi ditandai selesai. Operasi berakhir.
   b. Jika operasi yang dipilih adalahInstall, Patch Manager mengevaluasi Scan hasil dari Langkah 1 untuk menentukan apa yang akan dijalankan selanjutnya:
      i. Jika tidak terdeteksi ada patch yang hilang, dan tidak perlu reboot tertunda, operasi langsung melanjutkan ke langkah terakhir (Langkah 8), yang mencakup kait yang telah Anda berikan. Setiap langkah yang ada di antaranya dilewati.
      ii. Jika tidak terdeteksi ada patch yang hilang, tapi ada reboot tertunda yang diperlukan dan opsi reboot yang dipilih adalah NoReboot, operasi langsung melanjutkan ke langkah terakhir (Langkah 8), yang mencakup kait yang telah Anda berikan. Setiap langkah yang ada di antaranya dilewati.
      iii. Jika tidak, operasi dilanjutkan ke langkah berikutnya.
3. Operasi kait pra-tambalan - Dokumen SSM yang telah Anda sediakan untuk hook siklus hidup pertama,PreInstallHookDocName, dijalankan pada node terkelola.
4. Instal dengan NoReboot - Install Operasi dengan opsi reboot untuk NoReboot menggunakan AWS-RunPatchBaseline dijalankan pada node yang dikelola, dan laporan kepatuhan dibuat dan diunggah.
5. Operasi kait pasca-instal - Dokumen SSM yang telah Anda sediakan untuk hook siklus hidup kedua,PostInstallHookDocName, dijalankan pada node terkelola.
6. Verifikasi reboot - Skrip berjalan untuk menentukan apakah reboot diperlukan untuk node terkelola dan langkah-langkah apa yang harus dijalankan:
   a. Jika opsi reboot yang dipilih adalah NoReboot, operasi langsung melanjutkan ke langkah terakhir (Langkah 8), yang mencakup kait yang telah Anda berikan. Setiap langkah yang ada di antaranya dilewati.
   b. Jika opsi reboot yang dipilih adalahRebootIfNeeded, Patch Manager periksa apakah ada reboot tertunda yang diperlukan dari inventaris yang dikumpulkan di Langkah 4. Ini berarti

bahwa operasi berlanjut ke Langkah 7 dan node terkelola di-boot ulang dalam salah satu kasus berikut:

i. Patch Managermemasang satu atau lebih tambalan. (Patch Managertidak mengevaluasi apakah reboot diperlukan oleh tambalan. Sistem di-boot ulang bahkan jika tambalan tidak memerlukan reboot.)

ii. Patch Managermendeteksi satu atau lebih tambalan dengan status `INSTALLED_PENDING_REBOOT` selama operasi Instal. `INSTALLED_PENDING_REBOOT`Status dapat berarti bahwa opsi `NoReboot` dipilih saat terakhir kali operasi Install dijalankan.

Jika tidak ada patch yang memenuhi kriteria ini ditemukan, operasi patch node terkelola selesai, dan operasi berlanjut langsung ke langkah terakhir (Langkah 8), yang mencakup hook yang telah Anda berikan. Setiap langkah yang ada di antaranya dilewati.

7. Reboot dan laporkan - Operasi instalasi dengan opsi reboot `RebootIfNeeded` berjalan pada node terkelola menggunakan`AWS-RunPatchBaseline`, dan laporan kepatuhan dibuat dan diunggah.
8. Operasi kait pasca-reboot - Dokumen SSM yang telah Anda sediakan untuk hook siklus hidup ketiga,`OnExitHookDocName`, dijalankan pada node terkelola.

Untuk operasi `Scan`, jika Langkah 1 gagal, proses menjalankan dokumen berhenti dan langkah tersebut dilaporkan gagal, meskipun langkah-langkah berikutnya dilaporkan sebagai sukses.

Untuk operasi `Install`, jika salah satu langkah `aws:runDocument` gagal selama operasi, langkah-langkah tersebut dilaporkan gagal, dan operasi langsung melanjutkan ke langkah terakhir (Langkah 8), yang mencakup kait yang telah Anda berikan. Setiap langkah yang ada di antaranya dilewati. Langkah ini dilaporkan gagal, langkah terakhir melaporkan status hasil operasi, dan semua langkah di antaranya dilaporkan sebagai sukses.

Parameter **`AWS-RunPatchBaselineWithHooks`**

`AWS-RunPatchBaselineWithHooks` support enam parameter.

parameter `Operation` diperlukan.

Parameter `RebootOption`, `PreInstallHookDocName`, `PostInstallHookDocName`, dan `OnExitHookDocName` bersifat opsional.

`Snapshot-ID` secara teknis opsional, tetapi kami merekomendasikan Anda untuk menyediakan nilai kustom untuknya ketika Anda menjalankan `AWS-RunPatchBaselineWithHooks` di luar jendela pemeliharaan. Biarkan Patch Manager memberikan nilai secara otomatis ketika dokumen dijalankan sebagai bagian dari operasi jendela pemeliharaan.

Parameter

- Nama parameter: Operation
- Nama parameter: Snapshot ID
- Nama parameter: RebootOption
- Nama parameter: PreInstallHookDocName
- Nama parameter: PostInstallHookDocName
- Nama parameter: OnExitHookDocName

Nama parameter: **`Operation`**

Penggunaan: Wajib.

Opsi: `Scan` | `Install`.

Scan

Ketika Anda memilih `Scan` opsi, sistem menggunakan `AWS-RunPatchBaseline` dokumen untuk menentukan status kepatuhan patch dari node terkelola dan melaporkan informasi ini kembali kePatch Manager. `Scan`tidak meminta pembaruan untuk diinstal atau node yang dikelola untuk di-boot ulang. Sebaliknya, operasi mengidentifikasi di mana pembaruan hilang yang disetujui dan berlaku untuk node.

Menginstal

Ketika Anda memilih `Install` opsi, `AWS-RunPatchBaselineWithHooks` mencoba untuk menginstal pembaruan yang disetujui dan berlaku yang hilang dari node terkelola. Informasi kepatuhan patch yang dihasilkan sebagai bagian operasi `Install` tidak mencantumkan pembaruan yang hilang, tetapi mungkin melaporkan pembaruan yang berstatus gagal jika instalasi pembaruan tidak berhasil karena alasan apa pun. Setiap kali pembaruan diinstal pada node terkelola, node di-reboot untuk memastikan pembaruan diinstal dan aktif. (Pengecualian: Jika `RebootOption` parameter disetel ke `NoReboot` dalam `AWS-RunPatchBaselineWithHooks` dokumen, node terkelola tidak di-boot ulang setelah Patch Manager dijalankan. Untuk informasi lebih lanjut, lihatNama parameter: `RebootOption`.)

> **Note**
>
> Jika patch yang ditentukan oleh aturan dasar diinstal sebelum Patch Manager memperbarui node terkelola, sistem mungkin tidak reboot seperti yang diharapkan. Hal ini dapat terjadi ketika patch diinstal secara manual oleh pengguna atau diinstal secara otomatis oleh program lain, seperti `unattended-upgrades` paket aktifUbuntu Server.

Nama parameter: **`Snapshot ID`**

Penggunaan: Opsional.

`Snapshot ID`adalah ID unik (GUID) yang digunakan oleh Patch Manager untuk memastikan bahwa satu set node terkelola yang ditambal dalam satu operasi semuanya memiliki set patch yang disetujui yang sama persis. Meskipun parameter didefinisikan sebagai opsional, rekomendasi praktik terbaik kami bergantung pada apakah Anda menjalankan `AWS-RunPatchBaselineWithHooks` atau tidak di jendela pemeliharaan, seperti yang dijelaskan dalam tabel berikut.

Praktik terbaik **`AWS-RunPatchBaselineWithHooks`**

Mode	Praktik terbaik	Detail
Menjalankan `AWS-RunPatchBaselineWithHooks` di dalam jendela pemeliharaan	Jangan berikan ID Snapshot. Patch Managerakan memasoknya untuk Anda.	Jika Anda menggunakan jendela pemeliharaan untuk menjalankan`AWS-RunPatchBaselineWithHooks`, Anda seharusnya tidak memberikan ID Snapshot yang dibuat sendiri. Dalam skenario ini, Systems Manager menyediakan nilai GUID berdasarkan ID eksekusi jendela pemeliharaan. Hal ini memastikan bahwa digunakan ID yang benar untuk semua pemanggilan `AWS-RunPatchBaselineWithHoo`

Mode	Praktik terbaik	Detail
		ks dalam jendela pemeliharaan tersebut. Jika Anda menentukan nilai dalam skenario ini, perhatikan bahwa snapshot dari baseline patch mungkin tidak tetap di tempatnya selama lebih dari 3 hari. Setelah itu, snapshot baru akan dihasilkan bahkan jika Anda menentukan ID yang sama setelah snapshot berakhir.

Mode	Praktik terbaik	Detail
Menjalankan `AWS-RunPatchBaselineWithHooks` di luar jendela pemeliharaan	Menghasilkan dan menentukan nilai GUID kustom untuk ID Snapshot.[1]	Bila Anda tidak menggunakan jendela pemeliharaan untuk menjalankan`AWS-RunPatchBaselineWithHooks`, kami sarankan Anda membuat dan menentukan ID Snapshot unik untuk setiap baseline patch, terutama jika Anda menjalankan `AWS-RunPatchBaselineWithHooks` dokumen pada beberapa node terkelola dalam operasi yang sama. Jika Anda tidak menentukan ID dalam skenario ini, Systems Manager akan menghasilkan ID Snapshot yang berbeda untuk setiap node terkelola tempat perintah dikirim. Hal ini dapat mengakibatkan berbagai set patch yang ditentukan di antara node. Misalnya, katakanlah Anda menjalankan `AWS-RunPatchBaselineWithHooks` dokumen secara langsung melaluiRun Command, kemampuan AWS Systems Manager, dan menargetkan sekelompok 50 node terkelola. Menentukan ID Snapshot kustom menghasilkan pembuatan snapshot

Mode	Praktik terbaik	Detail
		dasar tunggal yang digunakan untuk mengevaluasi dan menambal semua node yang dikelola, memastikan bahwa mereka berakhir dalam status konsisten.

[1] Anda dapat menggunakan alat yang mampu menghasilkan GUID untuk menghasilkan nilai untuk parameter ID Snapshot. Misalnya, di PowerShell, Anda dapat menggunakan `New-Guid` cmdlet untuk menghasilkan GUID dalam format. `12345699-9405-4f69-bc5e-9315aEXAMPLE`

Nama parameter: **`RebootOption`**

Penggunaan: Opsional.

Pilihan: `RebootIfNeeded` | `NoReboot`

Default: `RebootIfNeeded`

> **Warning**
>
> Opsi default-nya adalah `RebootIfNeeded`. Pastikan untuk memilih opsi yang benar untuk kasus penggunaan Anda. Misalnya, jika node terkelola Anda harus segera reboot untuk menyelesaikan proses konfigurasi, pilih`RebootIfNeeded`. Atau, jika Anda perlu mempertahankan ketersediaan node terkelola hingga waktu reboot yang dijadwalkan, pilih`NoReboot`.

> **Important**
>
> Kami tidak menyarankan penggunaan Patch Manager untuk menambal instance cluster di Amazon EMR (sebelumnya disebut Amazon Elastic). MapReduce Secara khusus, jangan pilih `RebootIfNeeded` opsi untuk `RebootOption` parameter. (Opsi ini tersedia dalam dokumen Perintah SSM untuk ditambal`AWS-RunPatchBaseline`,`AWS-RunPatchBaselineAssociation`, dan`AWS-RunPatchBaselineWithHooks`.)

Perintah yang mendasari untuk menambal menggunakan Patch Manager penggunaan `yum` dan `dnf` perintah. Oleh karena itu, operasi mengakibatkan ketidakcocokan karena bagaimana paket diinstal. Untuk informasi tentang metode yang disukai untuk memperbarui perangkat lunak di klaster EMR Amazon, lihat Menggunakan default untuk AMI Amazon EMR di Panduan Manajemen EMR Amazon.

RebootIfNeeded

Saat Anda memilih `RebootIfNeeded` opsi, node terkelola di-boot ulang dalam salah satu kasus berikut:

- Patch Managermemasang satu atau lebih tambalan.

  Patch Managertidak mengevaluasi apakah reboot diperlukan oleh tambalan. Sistem di-boot ulang bahkan jika tambalan tidak memerlukan reboot.

- Patch Managermendeteksi satu atau lebih tambalan dengan status `INSTALLED_PENDING_REBOOT` selama operasi. `Install`

  Status `INSTALLED_PENDING_REBOOT` dapat berarti bahwa opsi `NoReboot` dipilih saat terakhir kali operasi `Install` dijalankan.

  > **Note**
  >
  > Patch yang dipasang di luar tidak pernah diberi status. Patch Manager `INSTALLED_PENDING_REBOOT`

Mem-boot ulang node terkelola dalam dua kasus ini memastikan bahwa paket yang diperbarui dikeluarkan dari memori dan menjaga perilaku patching dan reboot tetap konsisten di semua sistem operasi.

NoReboot

Ketika Anda memilih `NoReboot` opsi, Patch Manager tidak me-reboot node terkelola bahkan jika itu menginstal tambalan selama `Install` operasi. Opsi ini berguna jika Anda tahu bahwa node terkelola Anda tidak memerlukan reboot setelah tambalan diterapkan, atau Anda memiliki aplikasi atau proses yang berjalan pada node yang seharusnya tidak terganggu oleh reboot operasi patching. Ini juga berguna ketika Anda ingin lebih banyak kontrol atas waktu reboot node terkelola, seperti dengan menggunakan jendela pemeliharaan.

> **Note**
>
> Jika Anda memilih opsi `NoReboot` dan sebuah patch diinstal, patch diberikan status `InstalledPendingReboot`. Node yang dikelola itu sendiri, bagaimanapun, ditandai sebagai`Non-Compliant`. Setelah reboot terjadi dan `Scan` operasi dijalankan, status node diperbarui ke`Compliant`.

File pelacakan instalasi patch: Untuk melacak instalasi patch, terutama patch yang diinstal sejak reboot sistem terakhir, Systems Manager memelihara file pada node terkelola.

> **Important**
>
> Jangan menghapus atau memodifikasi file pelacakan. Jika file ini dihapus atau rusak, laporan kepatuhan patch untuk node terkelola tidak akurat. Jika ini terjadi, reboot node dan jalankan operasi Pindai tambalan untuk memulihkan file.

File pelacakan ini disimpan di lokasi berikut pada node terkelola Anda:

- Sistem operasi Linux:
  - `/var/log/amazon/ssm/patch-configuration/patch-states-configuration.json`
  - `/var/log/amazon/ssm/patch-configuration/patch-inventory-from-last-operation.json`
- Sistem operasi Windows Server:
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchStatesConfiguration.json`
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchInventoryFromLastOperation.json`

Nama parameter: **`PreInstallHookDocName`**

Penggunaan: Opsional.

Default: `AWS-Noop`.

Nilai untuk disediakan untuk parameter `PreInstallHookDocName` adalah nama atau Amazon Resource Name (ARN) dari dokumen SSM pilihan Anda. Anda dapat memberikan nama dokumen AWS terkelola atau nama atau ARN dari dokumen SSM khusus yang telah Anda buat atau yang telah dibagikan dengan Anda. (Untuk dokumen SSM yang telah dibagikan dengan Anda dari yang berbeda Akun AWS, Anda harus menentukan ARN sumber daya lengkap, seperti`arn:aws:ssm:us-east-2:123456789012:document/MySharedDocument`.)

Dokumen SSM yang Anda tentukan dijalankan sebelum `Install` operasi dan melakukan tindakan apa pun yang didukungSSM Agent, seperti skrip shell untuk memeriksa pemeriksaan kesehatan aplikasi sebelum penambalan dilakukan pada node terkelola. (Untuk daftar tindakan, lihat Referensi plugin dokumen perintah). Nama dokumen SSM default adalah`AWS-Noop`, yang tidak melakukan operasi apa pun pada node terkelola.

Untuk informasi tentang membuat dokumen SSM kustom, lihat Membuat konten dokumen SSM.

Nama parameter: **`PostInstallHookDocName`**

Penggunaan: Opsional.

Default: `AWS-Noop`.

Nilai untuk disediakan untuk parameter `PostInstallHookDocName` adalah nama atau Amazon Resource Name (ARN) dari dokumen SSM pilihan Anda. Anda dapat memberikan nama dokumen AWS terkelola atau nama atau ARN dari dokumen SSM khusus yang telah Anda buat atau yang telah dibagikan dengan Anda. (Untuk dokumen SSM yang telah dibagikan dengan Anda dari yang berbeda Akun AWS, Anda harus menentukan ARN sumber daya lengkap, seperti`arn:aws:ssm:us-east-2:123456789012:document/MySharedDocument`.)

Dokumen SSM yang Anda tentukan dijalankan setelah `Install with NoReboot` operasi dan melakukan tindakan apa pun yang didukung olehSSM Agent, seperti skrip shell untuk menginstal pembaruan pihak ketiga sebelum reboot. (Untuk daftar tindakan, lihat Referensi plugin dokumen perintah). Nama dokumen SSM default adalah`AWS-Noop`, yang tidak melakukan operasi apa pun pada node terkelola.

Untuk informasi tentang membuat dokumen SSM kustom, lihat Membuat konten dokumen SSM.

Nama parameter: **`OnExitHookDocName`**

Penggunaan: Opsional.

Default: `AWS-Noop`.

Nilai untuk disediakan untuk parameter `OnExitHookDocName` adalah nama atau Amazon Resource Name (ARN) dari dokumen SSM pilihan Anda. Anda dapat memberikan nama dokumen AWS terkelola atau nama atau ARN dari dokumen SSM khusus yang telah Anda buat atau yang telah dibagikan dengan Anda. (Untuk dokumen SSM yang telah dibagikan dengan Anda dari Akun AWS berbeda, Anda harus menentukan ARN sumber daya lengkap, seperti `arn:aws:ssm:us-east-2:123456789012:document/MySharedDocument`.)

Dokumen SSM yang Anda tentukan dijalankan setelah operasi reboot node terkelola dan melakukan tindakan apa pun yang didukungSSM Agent, seperti skrip shell untuk memverifikasi kesehatan node setelah operasi patching selesai. (Untuk daftar tindakan, lihat Referensi plugin dokumen perintah). Nama dokumen SSM default adalah`AWS-Noop`, yang tidak melakukan operasi apa pun pada node terkelola.

Untuk informasi tentang membuat dokumen SSM kustom, lihat Membuat konten dokumen SSM.

Contoh skenario untuk menggunakan InstallOverrideList parameter di**AWS-RunPatchBaseline** atau**AWS-RunPatchBaselineAssociation**

Anda dapat menggunakan`InstallOverrideList` parameter ketika Anda ingin menggantikan patch yang ditentukan oleh dasar patch default saat iniPatch Manager, suatu kemampuanAWS Systems Manager. Topik ini memberikan contoh yang menunjukkan cara menggunakan parameter ini untuk mencapai hal berikut:

- Menerapkan set patch yang berbeda ke grup instans.
- Menerapkan set patch ini pada frekuensi yang berbeda.
- Menggunakan dasar patch yang sama untuk kedua operasi.

Anggaplah bahwa Anda ingin menginstal dua kategori patch yang berbeda pada node yang dikelola Amazon Linux 2 Anda. Anda ingin menginstal patch ini pada jadwal yang berbeda menggunakan jendela pemeliharaan. Anda ingin satu jendela pemeliharaan berjalan setiap minggu dan menginstal semua patch `Security`. Anda ingin jendela pemeliharaan lain untuk berjalan sebulan sekali dan menginstal semua patch yang tersedia, atau kategori patch selain `Security`.

Namun, hanya satu dasar patch yang dapat didefinisikan sebagai default untuk sistem operasi pada satu waktu. Persyaratan ini membantu menghindari situasi ketika satu dasar patch menyetujui sebuah patch sedangkan yang lain memblokirnya, yang dapat menyebabkan masalah antara versi yang bertentangan.

Dengan strategi berikut ini, Anda menggunakan parameter `InstallOverrideList` untuk menerapkan jenis patch yang berbeda ke grup target, pada jadwal yang berbeda, sambil tetap menggunakan dasar patch yang sama:

1. Dalam dasar patch default, pastikan bahwa hanya pembaruan `Security` yang ditentukan.
2. Buat jendela pemeliharaan yang menjalankan `AWS-RunPatchBaseline` atau `AWS-RunPatchBaselineAssociation` setiap minggu. Jangan tentukan daftar override.
3. Buat daftar override patch dari semua jenis yang ingin Anda terapkan setiap bulan dan simpan di bucket Amazon Simple Storage Service (Amazon S3).
4. Buat jendela pemeliharaan kedua yang berjalan sebulan sekali. Namun, untukRun Command tugas yang Anda daftarkan untuk jendela pemeliharaan ini, tentukan lokasi daftar override Anda.

Hasilnya: Hanya patch `Security`, seperti yang didefinisikan dalam dasar patch default Anda, yang diinstal setiap minggu. Semua patch yang tersedia, atau subset patch lain yang Anda tentukan, diinstal setiap bulan.

Untuk informasi lebih lanjut dan daftar contoh, lihatNama parameter: `InstallOverrideList`.

Menggunakan BaselineOverride parameter

Anda dapat menentukan preferensi patching pada saat waktu aktif menggunakan fitur baseline override dalamPatch Manager suatu kemampuanAWS Systems Manager. Lakukan ini dengan menentukan bucket Amazon Simple Storage Service (Amazon S3) yang berisi objek JSON dengan daftar dasar patch. Operasi patching menggunakan baseline yang disediakan dalam objek JSON yang cocok dengan sistem operasi host bukannya menerapkan aturan dari dasar patch default.

> **Note**
>
> Menggunakan parameter `BaselineOverride` tidak menimpa kepatuhan patch dari baseline yang disediakan dalam parameter. Hasil output dicatat dalam log Stdout dariRun Command, suatu kemampuanAWS Systems Manager. Hasil hanya mencetak paket yang ditandai sebagai `NON_COMPLIANT`. Ini berarti paket ditandai sebagai `Missing`, `Failed`, `InstalledRejected`, atau `InstalledPendingReboot`.

Menggunakan override dasar patch dengan parameter Snapshot Id atau Install Override List

Ada dua kasus ketika override dasar patch memiliki perilaku yang patut diperhatikan.

Menggunakan baseline override dan Snapshot Id pada saat yang sama

Snapshot Id memastikan bahwa semua node yang dikelola dalam perintah patching tertentu semuanya menerapkan hal yang sama. Sebagai contoh, jika Anda melakukan patching 1.000 node pada satu waktu, patch akan sama.

Saat menggunakan Snapshot Id dan override dasar patch, Snapshot Id lebih diutamakan dari override dasar patch. Aturan baseline override masih akan digunakan, tetapi hanya akan dievaluasi sekali. Pada contoh sebelumnya, patch di 1.000 node terkelola Anda akan tetap selalu sama. Jika, di tengah operasi patching, Anda mengubah file JSON di bucket S3 yang direferensikan menjadi sesuatu yang berbeda, patch yang diterapkan akan tetap sama. Hal ini karena Snapshot Id telah disediakan.

Menggunakan baseline override dan Install Override List pada saat yang sama

Anda tidak dapat menggunakan dua parameter ini pada saat yang sama. Dokumen patching gagal jika kedua parameter disediakan, dan tidak akan melakukan pemindaian atau instalasi apa pun pada node yang dikelola.

Contoh kode

Contoh kode berikut ini untuk Python menunjukkan cara menghasilkan override dasar patch.

```
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)
```

Ini menghasilkan override dasar patch seperti berikut ini.

```
[
    {
        "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": []
    }
]
```

Tentang dasar patch

Topik di bagian ini memberikan informasi tentang cara kerja dasar patchPatch Manager, suatu kemampuanAWS Systems Manager, saat Anda menjalankan suatu`Scan` atau`Install` operasi di node yang dikelola.

Topik

- Tentang dasar patch yang telah ditetapkan dan kustom
- Tentang format nama paket untuk daftar patch yang disetujui dan ditolak
- Tentang grup patch
- Mengenai aplikasi patching yang dikeluarkan oleh Microsoft pada Windows Server

Tentang dasar patch yang telah ditetapkan dan kustom

Patch Manager, kemampuan AWS Systems Manager, menyediakan garis dasar patch yang telah ditentukan untuk masing-masing sistem operasi yang didukung oleh. Patch Manager Anda dapat menggunakan baseline ini karena mereka saat ini telah dikonfigurasi (Anda tidak dapat menyesuaikannya) atau Anda dapat membuat dasar patch kustom Anda sendiri. Dasar patch kustom memungkinkan Anda untuk memiliki kendali yang lebih besar atas patch yang disetujui atau ditolak untuk lingkungan Anda. Selain itu, baseline yang telah ditetapkan akan menetapkan tingkat kepatuhan `Unspecified` ke semua patch yang diinstal menggunakan baseline tersebut. Agar nilai kepatuhan ditetapkan, Anda dapat membuat salinan dari baseline yang telah ditetapkan dan menentukan nilai kepatuhan yang ingin Anda tetapkan ke patch. Untuk informasi lebih lanjut, lihat Tentang baseline kustom dan Bekerja dengan dasar patch kustom.

> **Note**
>
> Informasi dalam topik ini berlaku tidak peduli metode atau jenis konfigurasi yang Anda gunakan untuk operasi penambalan Anda:
>
> - Kebijakan tambalan yang dikonfigurasi di Quick Setup
> - Opsi Manajemen Host yang dikonfigurasi di Quick Setup

- Jendela pemeliharaan untuk menjalankan tambalan `Scan` atau `Install` tugas
- Patch sesuai permintaan sekarang beroperasi

Topik

- Tentang baseline yang telah ditetapkan
- Tentang baseline kustom

Tentang baseline yang telah ditetapkan

Tabel berikut menjelaskan garis dasar patch yang telah ditentukan yang disediakan. Patch Manager

Untuk informasi tentang versi masing-masing sistem operasi yang Patch Manager didukung, lihatPrasyarat Patch Manager.

Nama	Sistem operasi yang didukung	Detail
`AWS-AlmaLinuxDefaultPatchBaseline`	AlmaLinux	Menyetujui semua patch sistem operasi yang diklasifikasikan sebagai "Keamanan" dan yang memiliki tingkat kepelikan "Kritis" atau "Penting". Juga menyetujui semua tambalan yang diklasifikasikan sebagai “Bugfix”. Patch disetujui secara otomatis 7 hari setelah dirilis atau diperbarui.[1]
`AWS-AmazonLinuxDefaultPatchBaseline`	Amazon Linux 1	Menyetujui semua patch sistem operasi yang diklasifikasikan sebagai "Keamanan" dan yang memiliki tingkat kepelikan "Kritis" atau "Penting". Juga secara otomatis menyetujui semua tambalan dengan klasifika

Nama	Sistem operasi yang didukung	Detail
		si “Bugfix”. Patch disetujui secara otomatis 7 hari setelah dirilis atau diperbarui.[1]
`AWS-AmazonLinux2DefaultPatchBaseline`	Amazon Linux 2	Menyetujui semua patch sistem operasi yang diklasifikasikan sebagai "Keamanan" dan yang memiliki tingkat kepelikan "Kritis" atau "Penting". Juga menyetujui semua tambalan dengan klasifikasi “Bugfix”. Patch disetujui secara otomatis 7 hari setelah rilis.[1]
`AWS-AmazonLinux2022DefaultPatchBaseline`	Amazon Linux 2022	Menyetujui semua patch sistem operasi yang diklasifikasikan sebagai "Keamanan" dan yang memiliki tingkat kepelikan "Kritis" atau "Penting". Patch disetujui secara otomatis tujuh hari setelah rilis. Juga menyetujui semua patch dengan klasifikasi "Bugfix" tujuh hari setelah rilis.

Nama	Sistem operasi yang didukung	Detail
`AWS-AmazonLinux2023DefaultPatchBaseline`	Amazon Linux 2023	Menyetujui semua patch sistem operasi yang diklasifikasikan sebagai "Keamanan" dan yang memiliki tingkat kepelikan "Kritis" atau "Penting". Patch disetujui secara otomatis tujuh hari setelah rilis. Juga menyetujui semua patch dengan klasifikasi "Bugfix" tujuh hari setelah rilis.
`AWS-CentOSDefaultPatchBaseline`	CentOS dan CentOS Stream	Menyetujui semua pembaruan 7 hari setelah tersedia, termasuk pembaruan non-keamanan.
`AWS-DebianDefaultPatchBaseline`	Debian Server	Segera menyetujui semua patch terkait keamanan sistem operasi yang memiliki prioritas "Wajib", "Penting", "Standar", "Opsional", atau "Ekstra." Tidak perlu menunggu sebelum persetujuan karena tanggal rilis yang andal tidak tersedia di repositori.
`AWS-MacOSDefaultPatchBaseline`	macOS	Menyetujui semua patch sistem operasi yang diklasifikasikan sebagai "Keamanan". Juga menyetujui semua paket dengan pembaruan saat ini.

Nama	Sistem operasi yang didukung	Detail
`AWS-OracleLinuxDefaultPatchBaseline`	Oracle Linux	Menyetujui semua patch sistem operasi yang diklasifikasikan sebagai "Keamanan" dan yang memiliki tingkat kepelikan "Penting" atau "Sedang". Juga menyetujui semua tambalan yang diklasifikasikan sebagai “Bugfix” 7 hari setelah rilis. Patch disetujui secara otomatis 7 hari setelah dirilis atau diperbarui.[1]
`AWS-DefaultRaspbianPatchBaseline`	Raspberry Pi OS	Segera menyetujui semua patch terkait keamanan sistem operasi yang memiliki prioritas "Wajib", "Penting", "Standar", "Opsional", atau "Ekstra." Tidak perlu menunggu sebelum persetujuan karena tanggal rilis yang andal tidak tersedia di repositori.
`AWS-RedHatDefaultPatchBaseline`	Red Hat Enterprise Linux (RHEL)	Menyetujui semua patch sistem operasi yang diklasifikasikan sebagai "Keamanan" dan yang memiliki tingkat kepelikan "Kritis" atau "Penting". Juga menyetujui semua tambalan yang diklasifikasikan sebagai “Bugfix”. Patch disetujui secara otomatis 7 hari setelah dirilis atau diperbarui.[1]

Nama	Sistem operasi yang didukung	Detail
`AWS-RockyLinuxDefaultPatchBaseline`	Rocky Linux	Menyetujui semua patch sistem operasi yang diklasifikasikan sebagai "Keamanan" dan yang memiliki tingkat kepelikan "Kritis" atau "Penting". Juga menyetujui semua tambalan yang diklasifikasikan sebagai “Bugfix”. Patch disetujui secara otomatis 7 hari setelah dirilis atau diperbarui.[1]
`AWS-SuseDefaultPatchBaseline`	SUSE Linux Enterprise Server (SLES)	Menyetujui semua patch sistem operasi yang diklasifikasikan sebagai "Keamanan" dan dengan kepelikan "Kritis" atau "Penting". Patch disetujui secara otomatis 7 hari setelah dirilis atau diperbarui.[1]
`AWS-UbuntuDefaultPatchBaseline`	Ubuntu Server	Segera menyetujui semua patch terkait keamanan sistem operasi yang memiliki prioritas "Wajib", "Penting", "Standar", "Opsional", atau "Ekstra." Tidak perlu menunggu sebelum persetujuan karena tanggal rilis yang andal tidak tersedia di repositori.

Nama	Sistem operasi yang didukung	Detail
`AWS-DefaultPatchBaseline`	Windows Server	Menyetujui semua tambalan sistem Windows Server operasi yang diklasifikasikan sebagai "" atau CriticalUpdates "SecurityUpdates" dan yang memiliki tingkat keparahan MSRC “Kritis” atau “Penting”. Patch disetujui secara otomatis 7 hari setelah dirilis atau diperbarui.[2]
`AWS-WindowsPredefinedPatchBaseline-OS`	Windows Server	Menyetujui semua tambalan sistem Windows Server operasi yang diklasifikasikan sebagai "" atau CriticalUpdates "SecurityUpdates" dan yang memiliki tingkat keparahan MSRC “Kritis” atau “Penting”. Patch disetujui secara otomatis 7 hari setelah dirilis atau diperbarui.[2]

Nama	Sistem operasi yang didukung	Detail
`AWS-WindowsPredefinedPatchBaseline-OS-Applications`	Windows Server	Untuk sistem Windows Server operasi, setujui semua tambalan yang diklasifikasikan sebagai "" atau "Critical UpdatesSecurityUpdates" dan yang memiliki tingkat keparahan MSRC “Kritis” atau “Penting”. Untuk aplikasi yang dirilis oleh Microsoft , menyetujui semua patch. Patch untuk OS dan aplikasi disetujui secara otomatis 7 hari setelah dirilis atau diperbarui.[2]

[1] Untuk Amazon Linux 1 dan Amazon Linux 2, penantian 7 hari sebelum patch disetujui otomatis dihitung dari `Updated Date` nilai`updateinfo.xml`, bukan nilai. `Release Date` Berbagai faktor dapat mempengaruhi `Updated Date` nilai. Sistem operasi lain menangani tanggal rilis dan pembaruan secara berbeda. Untuk informasi yang membantu Anda menghindari hasil yang tidak terduga dengan penundaan persetujuan otomatis, lihat. Bagaimana tanggal rilis paket dan tanggal pembaruan dihitung

[2] UntukWindows Server, baseline default mencakup penundaan persetujuan otomatis 7 hari. Untuk menginstal patch dalam 7 hari setelah rilis, Anda harus membuat baseline kustom.

Tentang baseline kustom

Jika Anda membuat dasar patch Anda sendiri, Anda dapat memilih patch mana yang akan disetujui secara otomatis dengan menggunakan kategori berikut.

- Sistem operasi:Windows Server, Amazon LinuxUbuntu Server, dan sebagainya.
- Nama produk (untuk sistem operasi): Misalnya, RHEL 6.5, Amazon Linux 2014.09, Windows Server 2012, Windows Server 2012 R2, dan seterusnya.
- Nama produk (untuk aplikasi yang dirilis oleh Microsoft Windows Server hanya): Misalnya, Word 2016, BizTalk Server, dan sebagainya.

- Klasifikasi: Sebagai contoh, pembaruan kritis, pembaruan keamanan, dan sebagainya.
- Kepelikan: Sebagai contoh, kritis, penting, dan sebagainya.

Untuk setiap aturan persetujuan yang Anda buat, Anda dapat memilih untuk menentukan penundaan persetujuan otomatis atau menentukan tanggal cutoff persetujuan patch.

> **Note**
>
> Karena tidak mungkin menentukan tanggal rilis paket pembaruan secara andalUbuntu Server, opsi persetujuan otomatis tidak didukung untuk sistem operasi ini.

Penundaan persetujuan otomatis adalah jumlah hari untuk menunggu setelah tambalan dirilis atau terakhir diperbarui, sebelum tambalan secara otomatis disetujui untuk ditambal. Misalnya, jika Anda membuat aturan menggunakan `CriticalUpdates` klasifikasi dan mengonfigurasinya selama 7 hari penundaan persetujuan otomatis, maka patch kritis baru yang dirilis pada 7 Juli secara otomatis disetujui pada 14 Juli.

> **Note**
>
> Jika repositori Linux tidak memberikan informasi tanggal rilis untuk paket, Systems Manager menggunakan waktu pembuatan paket sebagai penundaan persetujuan otomatis untuk Amazon Linux 1, Amazon Linux 2, RHEL dan CentOS. Jika sistem tidak dapat menemukan waktu build paket, Systems Manager memperlakukan penundaan persetujuan otomatis sebagai memiliki nilai nol.

Saat Anda menentukan tanggal batas persetujuan otomatis, Patch Manager secara otomatis menerapkan semua tambalan yang dirilis atau terakhir diperbarui pada atau sebelum tanggal tersebut. Misalnya, jika Anda menentukan 7 Juli 2023 sebagai tanggal cutoff, tidak ada tambalan yang dirilis atau terakhir diperbarui pada atau setelah 8 Juli 2023 yang diinstal secara otomatis.

> **Note**
>
> Saat membuat baseline patch kustom, Anda dapat menentukan tingkat keparahan kepatuhan untuk tambalan yang disetujui oleh baseline patch tersebut, seperti atau. `Critical` `High` Jika status tambalan dari tambalan yang disetujui dilaporkan sebagai`Missing`, maka tingkat

keparahan kepatuhan keseluruhan baseline patch yang dilaporkan adalah tingkat keparahan yang Anda tentukan.

Ingatlah hal-hal berikut ini saat Anda membuat dasar patch:

- Patch Managermenyediakan satu baseline patch yang telah ditentukan untuk setiap sistem operasi yang didukung. Dasar patch yang telah ditetapkan ini digunakan sebagai dasar patch default untuk setiap jenis sistem operasi kecuali Anda membuat dasar patch Anda sendiri dan menunjuknya sebagai default untuk jenis sistem operasi yang sesuai.

  > **Note**
  >
  > Untuk Windows Server, disediakan tiga dasar patch yang telah ditetapkan. Garis dasar patch `AWS-DefaultPatchBaseline` dan hanya `AWS-WindowsPredefinedPatchBaseline-OS` mendukung pembaruan sistem operasi pada sistem operasi Windows itu sendiri. `AWS-DefaultPatchBaseline`digunakan sebagai baseline patch default untuk node Windows Server terkelola kecuali Anda menentukan baseline patch yang berbeda. Pengaturan konfigurasi di dua dasar patch ini sama. Yang lebih baru dari keduanya, `AWS-WindowsPredefinedPatchBaseline-OS`, dibuat untuk membedakannya dari dasar patch ketiga yang telah ditetapkan untuk Windows Server. Dasar patch tersebut, `AWS-WindowsPredefinedPatchBaseline-OS-Applications`, dapat digunakan untuk menerapkan patch ke ke sistem operasi Windows Server dan aplikasi yang didukung yang dirilis oleh Microsoft.

- Untuk server lokal dan mesin virtual (VM), Patch Manager mencoba menggunakan baseline patch default kustom Anda. Jika tidak tersedia dasar patch default kustom, sistem menggunakan dasar patch yang telah ditetapkan untuk sistem operasi yang sesuai.
- Jika sebuah patch terdaftar sebagai disetujui dan ditolak dalam dasar patch yang sama, patch ditolak.
- Node terkelola hanya dapat memiliki satu baseline patch yang ditentukan untuknya.
- Format nama paket yang dapat Anda tambahkan ke daftar patch yang disetujui dan patch yang ditolak untuk dasar patch tergantung pada jenis sistem operasi yang Anda patching.

  Untuk informasi tentang format yang diterima untuk daftar patch yang disetujui dan patch yang ditolak, lihat Tentang format nama paket untuk daftar patch yang disetujui dan ditolak.

- Jika Anda menggunakan konfigurasi kebijakan tambalanQuick Setup, pembaruan yang Anda buat ke baseline patch kustom disinkronkan dengan Quick Setup satu jam sekali.

  Jika baseline patch kustom yang direferensikan dalam kebijakan tambalan dihapus, spanduk akan ditampilkan di halaman Detail Quick Setup konfigurasi untuk kebijakan tambalan Anda. Spanduk memberi tahu Anda bahwa kebijakan tambalan mereferensikan baseline tambalan yang tidak ada lagi, dan operasi penambalan berikutnya akan gagal. Dalam hal ini, kembali ke halaman Quick Setup Konfigurasi, pilih Patch Manager konfigurasi, dan pilih Tindakan, Edit konfigurasi. Nama dasar patch yang dihapus disorot, dan Anda harus memilih baseline patch baru untuk sistem operasi yang terpengaruh.

Untuk informasi tentang membuat dasar patch, lihat Bekerja dengan dasar patch kustom dan Tutorial: Menambal lingkungan server (AWS CLI).

Tentang format nama paket untuk daftar patch yang disetujui dan ditolak

Format nama paket yang dapat Anda tambahkan ke daftar patch yang disetujui dan patch yang ditolak tergantung pada jenis sistem operasi yang Anda patching.

Format nama paket untuk sistem operasi Linux

Format yang dapat Anda tentukan untuk patch yang disetujui dan ditolak di dasar patch Anda bervariasi berdasarkan jenis Linux. Secara lebih spesifik, format yang didukung bergantung pada pengelola paket yang digunakan oleh jenis sistem operasi Linux.

Topik

- Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022, Amazon Linux 2023, CentOSOracle Linux, Red Hat Enterprise Linux dan () RHEL
- Debian Server, Raspberry Pi OS (sebelumnya Raspbian), dan Ubuntu Server
- SUSE Linux Enterprise Server (SLES)

Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022, Amazon Linux 2023, CentOSOracle Linux, Red Hat Enterprise Linux dan () RHEL

Package manager: YUM, kecuali Amazon Linux 2022, Amazon Linux 2023, RHEL 8, dan CentOS 8, yang menggunakan DNF sebagai manajer paket

Patch yang disetujui: Untuk patch yang disetujui, Anda dapat menentukan salah satu dari berikut:

- ID Bugzilla, dalam format `1234567` (Sistem memproses string nomor sebagai ID Bugzilla.)
- ID CVE, dalam format `CVE-2018-1234567`
- ID Penasehat, dalam format seperti `RHSA-2017:0864` dan `ALAS-2018-123`
- Nama paket lengkap, dalam format seperti:
  - `example-pkg-0.710.10-2.7.abcd.x86_64`
  - `pkg-example-EE-20180914-2.2.amzn1.noarch`
- Nama-paket dengan satu wildcard, dalam format seperti:
  - `example-pkg-*.abcd.x86_64`
  - `example-pkg-*-20180914-2.2.amzn1.noarch`
  - `example-pkg-EE-2018*.amzn1.noarch`

Patch yang ditolak: Untuk patch yang ditolak, Anda dapat menentukan salah satu dari berikut:

- Nama paket lengkap, dalam format seperti:
  - `example-pkg-0.710.10-2.7.abcd.x86_64`
  - `pkg-example-EE-20180914-2.2.amzn1.noarch`
- Nama-paket dengan satu wildcard, dalam format seperti:
  - `example-pkg-*.abcd.x86_64`
  - `example-pkg-*-20180914-2.2.amzn1.noarch`
  - `example-pkg-EE-2018*.amzn1.noarch`

Debian Server, Raspberry Pi OS (sebelumnya Raspbian), dan Ubuntu Server

Pengelola paket: APT

Patch yang disetujui dan patch yang ditolak: Untuk patch yang disetujui dan ditolak, tentukan hal berikut:

- Nama paket, dalam format `ExamplePkg33`

> **Note**
>
> Untuk Debian Server daftar, Raspberry Pi OS daftar, dan Ubuntu Server daftar, jangan sertakan elemen seperti arsitektur atau versi. Sebagai contoh, Anda menentukan nama paket `ExamplePkg33` untuk menyertakan semua hal berikut dalam daftar patch:

- `ExamplePkg33.x86.1`
- `ExamplePkg33.x86.2`
- `ExamplePkg33.x64.1`
- `ExamplePkg33.3.2.5-364.noarch`

SUSE Linux Enterprise Server (SLES)

Pengelola paket: Zypper

Patch yang disetujui dan patch yang ditolak: Untuk daftar patch yang disetujui dan ditolak, Anda dapat menentukan salah satu hal berikut:

- Nama paket lengkap, dalam format seperti:
  - `SUSE-SLE-Example-Package-12-2018-123`
  - `example-pkg-2018.11.4-46.17.1.x86_64.rpm`
- Nama-paket dengan satu wildcard, seperti:
  - `SUSE-SLE-Example-Package-12-2018-*`
  - `example-pkg-2018.11.4-46.17.1.*.rpm`

Format nama paket untuk macOS

Pengelola paket yang didukung: softwareupdate, installer, Brew, Brew Cask

Patch yang disetujui dan patch yang ditolak: Untuk daftar patch yang disetujui dan ditolak, Anda menentukan nama paket lengkap, dalam format seperti:

- `XProtectPlistConfigData`
- `MRTConfigData`

Wildcard tidak didukung dalam daftar patch yang disetujui dan ditolak untuk macOS.

Format nama paket untuk sistem operasi Windows

Untuk sistem operasi Windows, tentukan patch menggunakan ID Microsoft Knowledge Base dan ID Microsoft Security Bulletin; misalnya:

```
KB2032276,KB2124261,MS10-048
```

Tentang grup patch

> **Important**
>
> Grup patch tidak digunakan dalam operasi patching yang didasarkan pada kebijakan patch. Untuk informasi tentang bekerja dengan kebijakan patch, lihatMenggunakan kebijakan Quick Setup tambalan.

Anda dapat menggunakan grup patch patch untuk mengasosiasikan node terkelola dengan dasar patch tertentuPatch Manager, suatu kemampuanAWS Systems Manager. Grup patch membantu memastikan bahwa Anda men-deploy patch yang sesuai, berdasarkan aturan dasar patch terkait, ke rangkaian node yang benar. Grup patch juga dapat membantu Anda menghindari men-deploy patch sebelum diuji secara memadai. Sebagai contoh, Anda dapat membuat grup patch untuk lingkungan yang berbeda (seperti pengembangan, pengujian, dan produksi) dan mendaftarkan setiap grup patch ke dasar patch yang sesuai.

Ketika Anda menjalankan`AWS-RunPatchBaseline`, Anda dapat menargetkan node terkelola menggunakan ID atau tag. SSM AgentdanPatch Manager kemudian mengevaluasi dasar patch yang akan digunakan berdasarkan nilai grup patch yang ditambahkan ke node terkelola.

Anda membuat grup patch dengan menggunakan tag Amazon Elastic Compute Cloud (Amazon EC2). Tidak seperti skenario penandaan lainnya di Systems Manager, grup patch harus didefinisikan dengan kunci tag`Patch Group` atau`PatchGroup`. Kunci ini peka terhadap huruf besar-kecil. Anda dapat menentukan nilai apa pun untuk membantu Anda mengidentifikasi dan menargetkan sumber daya dalam grup tersebut, misalnya “server web” atau “US-EAST-PROD”, tetapi kuncinya harus`Patch Group` atau`PatchGroup`.

Setelah Anda membuat grup patch dan menandai node terkelola, Anda dapat mendaftarkan grup patch dengan dasar patch. Mendaftarkan grup patch patch dengan dasar patch memastikan bahwa node dalam grup patch menggunakan aturan yang didefinisikan dalam dasar patch terkait.

Untuk informasi selengkapnya tentang cara membuat grup patch dan mengaitkan grup patch ke dasar patch, lihat Bekerja dengan kelompok patch dan Menambahkan grup patch ke dasar patch.

Untuk melihat contoh membuat dasar patch dan grup patch dengan menggunakan AWS Command Line Interface (AWS CLI), lihat Tutorial: Menambal lingkungan server (AWS CLI). Untuk informasi

selengkapnya tentang tag Amazon EC2, lihat Menandai sumber daya Amazon EC2 Anda dalam Panduan Pengguna Amazon EC2 untuk Instans Linux.

Cara kerjanya

Ketika sistem menjalankan tugas untuk menerapkan dasar patch ke node terkelola,SSM Agent verifikasi bahwa nilai grup patch didefinisikan untuk node tersebut. Jika node ditetapkan ke sebuah grup patchPatch Manager, verifikasi dasar patch yang terdaftar ke grup patch tersebut. Jika ditemukan dasar patch untuk grup tersebut,Patch Manager memberitahuSSM Agent untuk menggunakan dasar patch terkait. Jika node tidak dikonfigurasi untuk grup patch,Patch Manager secara otomatis memberitahuSSM Agent untuk menggunakan dasar patch default yang terkonfigurasi saat ini.

> **Important**
>
> Sebuah node terkelola hanya dapat berada dalam satu grup patch.
> Suatu grup patch hanya dapat didaftarkan dengan satu dasar patch untuk setiap jenis sistem operasi.
> Anda tidak dapat menerapkan`Patch Group` tag (dengan spasi) ke instans Amazon EC2 jika opsi Izinkan tag dalam metadata instans diaktifkan pada instans. Mengizinkan tag dalam metadata instance mencegah nama kunci tag berisi spasi. Jika Anda telah mengizinkan tag dalam metadata instans EC2, Anda harus menggunakan kunci tag`PatchGroup` (tanpa spasi).

diagram berikut ini menunjukkan contoh umum dari proses yang dilakukan Systems Manager saat mengirimRun Command tugas ke armada server Anda untuk di-patch menggunakanPatch Manager. Proses serupa digunakan saat jendela pemeliharaan dikonfigurasi untuk mengirim perintah untuk mem-patch menggunakan patchPatch Manager.

Dalam contoh ini, ada tiga grup instans EC2 untuk Windows Server dengan tag berikut ini diterapkan:

Grup instans EC2	Tag
Grup 1	`key=OS,value=Windows` `key=PatchGroup,value=DEV`
Grup 2	`key=OS,value=Windows`

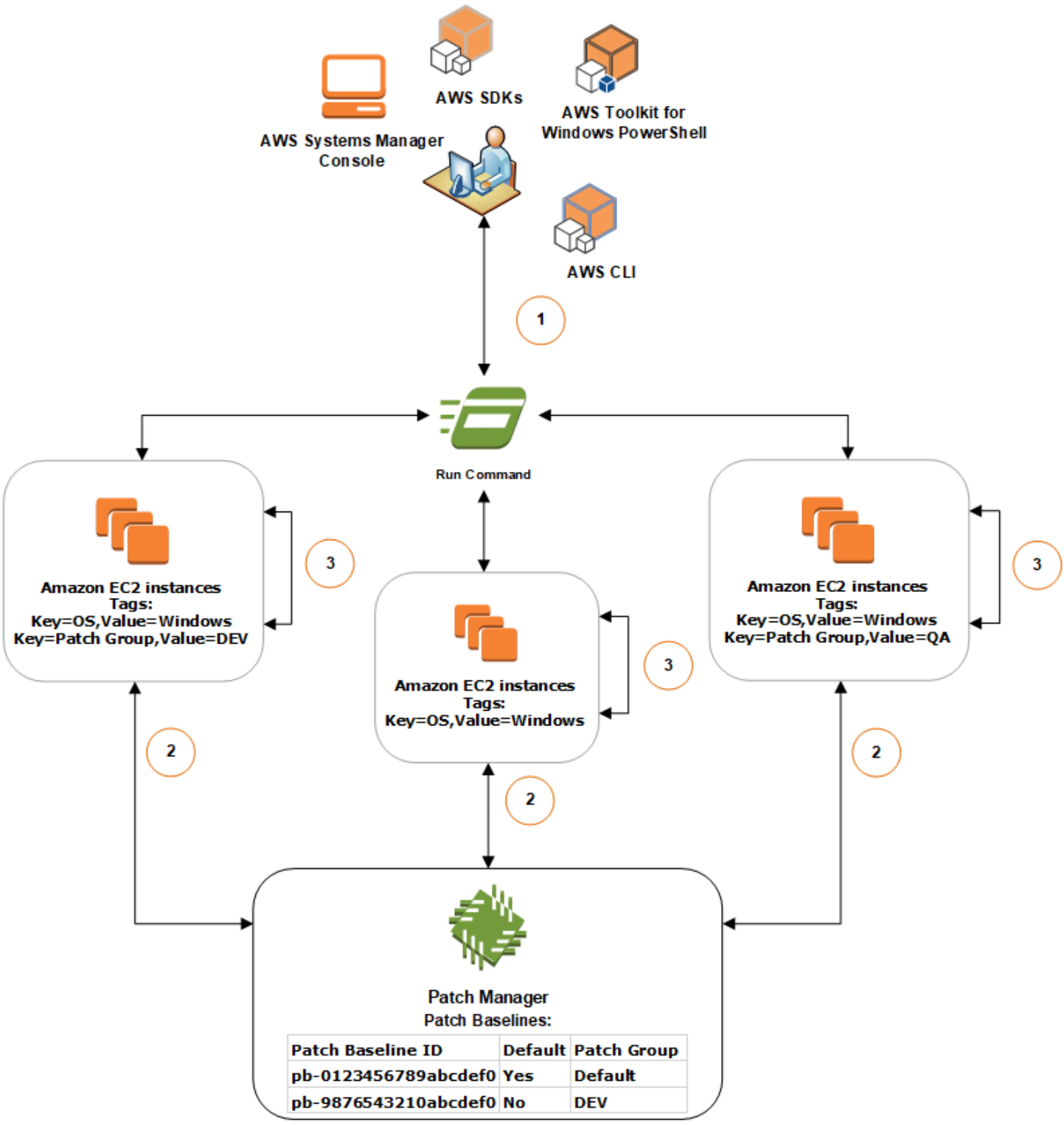
Grup instans EC2	Tag
Grup 3	`key=OS,value=Windows` `key=PatchGroup,value=QA`

Untuk contoh ini, terdapat juga dua dasar patch Windows Server ini:

ID dasar patch	Default	Grup patch terkait
`pb-0123456789abcdef0`	Ya	`Default`
`pb-9876543210abcdef0`	Tidak	`DEV`

Diagram 1: Contoh umum alur proses operasi patching

Diagram berikut menunjukkan bagaimanaPatch Manager menentukan baseline patch mana yang akan digunakan dalam operasi patching.



Proses umum untuk memindai atau menginstal patch menggunakanRun Command, suatu kemampuanAWS Systems Manager, danPatch Manager adalah sebagai berikut:

1. Kirim perintah untuk di-patch: Gunakan konsol Systems Manager, SDK,AWS Command Line Interface (AWS CLI), atauAWS Tools for Windows PowerShell untuk mengirimRun Command tugas menggunakan dokumen`AWS-RunPatchBaseline`. diagram menunjukkanRun Command tugas untuk mem-patch instans terkelola dengan menargetkan tag`key=OS,value=Windows`.
2. Penentuan dasar patch:SSM Agent memverifikasi tag grup patch yang diterapkan ke instans EC2 dan kueriPatch Manager dasar patch sesuai.
   - Nilai grup patch sesuai yang terkait dengan dasar patch:
     1. SSM Agent, yang diinstal pada instans EC2 dalam grup satu, menerima perintah yang dikeluarkan pada Langkah 1 untuk memulai operasi patching. SSM Agentmemvalidasi bahwa instans EC2 memiliki nilai tag grup patch yang`DEV` diterapkan dan kueriPatch Manager dasar patch terkait.
     2. Patch Managermemverifikasi bahwa patch baseline`pb-9876543210abcdef0` memiliki kelompok patch yang`DEV` terkait dan memberitahukanSSM Agent.
     3. SSM Agentmengambil snapshot dasar patch dariPatch Manager berdasarkan aturan persetujuan dan pengecualian yang dikonfigurasi`pb-9876543210abcdef0` dan melanjutkan ke langkah berikutnya.
   - Tidak ada tag grup patch patch yang ditambahkan ke instans:
     1. SSM Agent, yang diinstal pada instans EC2 dalam grup dua, menerima perintah yang dikeluarkan pada Langkah 1 untuk memulai operasi patching. SSM Agentmemvalidasi bahwa instans EC2 tidak memiliki`PatchGroup` tag`Patch Group` atau tag yang diterapkan dan sebagai hasilnya,SSM Agent kueriPatch Manager dasar patch Windows default.
     2. Patch Managermemverifikasi bahwa dasarWindows Server patch default adalah`pb-0123456789abcdef0` dan memberi tahuSSM Agent.
     3. SSM Agentmengambil snapshot dasar patch dariPatch Manager berdasarkan aturan persetujuan dan pengecualian yang dikonfigurasi di dasar patch default`pb-0123456789abcdef0` dan melanjutkan ke langkah berikutnya.
   - Tidak ada nilai grup patch patch sesuai yang terkait dengan dasar patch:
     1. SSM Agent, yang diinstal pada instans EC2 dalam grup tiga, menerima perintah yang dikeluarkan pada Langkah 1 untuk memulai operasi patching. SSM Agentmemvalidasi bahwa instans EC2 memiliki nilai tag grup patch yang`QA` diterapkan dan kueriPatch Manager dasar patch terkait.
     2. Patch Managertidak menemukan dasar patch yang memiliki grup patch`QA` terkait.

3. Patch ManagermemberitahukanSSM Agent untuk menggunakan dasar patch Windows defaultpb-0123456789abcdef0.
4. SSM Agentmengambil snapshot dasar patch dariPatch Manager berdasarkan aturan persetujuan dan pengecualian yang dikonfigurasi di dasar patch defaultpb-0123456789abcdef0 dan melanjutkan ke langkah berikutnya.

3. Pemindaian dan instalasi patch: Setelah menentukan dasar patch yang sesuai untuk digunakan,SSM Agent mulai memindai atau menginstal patch berdasarkan nilai operasi yang ditentukan dalam Langkah 1. Patch yang dipindai atau diinstal ditentukan oleh aturan persetujuan dan pengecualian patch yang didefinisikan dalam snapshot dasar patch yang disediakan oleh patchPatch Manager.

Info lebih lanjut

- Memahami nilai keadaan kepatuhan patch

Mengenai aplikasi patching yang dikeluarkan oleh Microsoft pada Windows Server

Gunakan informasi dalam topik ini untuk membantu Anda mempersiapkan untuk menambal aplikasi yang sedang Windows Server digunakanPatch Manager, kemampuanAWS Systems Manager.

Patching aplikasi Microsoft

Menambal dukungan untuk aplikasi pada node Windows Server terkelola terbatas pada aplikasi yang dirilis oleh Microsoft.

> **Note**
>
> Dalam beberapa kasus, Microsoft merilis patch untuk aplikasi yang tidak menentukan tanggal dan waktu yang diperbarui. Dalam kasus ini, tanggal dan waktu yang diperbarui `01/01/1970` disediakan secara default.

Dasar patch untuk patching aplikasi yang dirilis oleh Microsoft

Untuk Windows Server, disediakan tiga dasar patch yang telah ditetapkan. Garis dasar patch `AWS-DefaultPatchBaseline` dan hanya `AWS-WindowsPredefinedPatchBaseline-OS` mendukung pembaruan sistem operasi pada sistem operasi Windows itu sendiri. `AWS-DefaultPatchBaseline`digunakan sebagai baseline patch default untuk node Windows Server

terkelola kecuali Anda menentukan baseline patch yang berbeda. Pengaturan konfigurasi di dua dasar patch ini sama. Yang lebih baru dari keduanya, `AWS-WindowsPredefinedPatchBaseline-OS`, dibuat untuk membedakannya dari dasar patch ketiga yang telah ditetapkan untuk Windows Server. Dasar patch tersebut, `AWS-WindowsPredefinedPatchBaseline-OS-Applications`, dapat digunakan untuk menerapkan patch ke ke sistem operasi Windows Server dan aplikasi yang didukung yang dirilis oleh Microsoft.

Anda juga dapat membuat dasar patch kustom untuk memperbarui aplikasi yang dirilis oleh Microsoft pada mesin Windows Server.

Support untuk menambal aplikasi yang dirilis oleh Microsoft di server lokal, perangkat edge, VM, dan node non-EC2 lainnya

Untuk menambal aplikasi yang dirilis oleh Microsoft pada mesin virtual (VM) dan node terkelola non-EC2 lainnya, Anda harus mengaktifkan tingkat instance lanjutan. Biaya dikenakan untuk menggunakan tingkat instans lanjutan. Namun, tidak ada biaya tambahan untuk menambal aplikasi yang dirilis oleh Microsoft di instans Amazon Elastic Compute Cloud (Amazon EC2). Untuk informasi selengkapnya, lihat Mengonfigurasi tingkat instans.

Opsi Windows Update untuk "produk Microsoft lainnya"

Agar dapat Patch Manager menambal aplikasi yang dirilis oleh Microsoft pada node Windows Server terkelola Anda, opsi Pembaruan Windows Beri saya pembaruan untuk produk Microsoft lainnya ketika saya memperbarui Windows harus diaktifkan pada node yang dikelola.

Untuk informasi tentang mengizinkan opsi ini pada satu node terkelola, lihat Memperbarui Office dengan Microsoft Update di situs web Dukungan Microsoft.

Untuk armada node terkelola yang menjalankan Windows Server 2016 dan yang lebih baru, Anda dapat menggunakan Objek Kebijakan Grup (GPO) untuk mengaktifkan pengaturan. Di Editor Pengelolaan Kebijakan Grup, buka Konfigurasi Komputer, Templat Administratif, Komponen Windows, Windows Update, dan pilih Instal pembaruan untuk produk Microsoft lainnya. Kami juga merekomendasikan untuk mengonfigurasi GPO dengan parameter tambahan yang mencegah pembaruan otomatis yang tidak direncanakan dan reboot di luar. Patch Manager Untuk informasi selengkapnya, lihat Mengonfigurasi Pembaruan Otomatis di Lingkungan Direktori Non-Aktif di situs web dokumentasi teknis Microsoft.

Untuk armada node terkelola yang menjalankan Windows Server 2012 atau 2012 R2, Anda dapat mengaktifkan opsi dengan menggunakan skrip, seperti yang dijelaskan dalam Mengaktifkan dan

Menonaktifkan Pembaruan Microsoft di Windows 7 melalui Script di situs web Microsoft Docs Blog. Sebagai contoh, Anda dapat melakukan hal berikut:

1. Simpan script dari posting blog dalam sebuah file.
2. Unggah file ke bucket Amazon Simple Storage Service (Amazon S3) atau lokasi lain yang dapat diakses.
3. GunakanRun Command, kemampuanAWS Systems Manager, untuk menjalankan skrip pada node terkelola Anda menggunakan dokumen Systems Manager (dokumen SSM) `AWS-RunPowerShellScript` dengan perintah yang mirip dengan berikut ini.

```
Invoke-WebRequest `
    -Uri "https://s3.aws-api-domain/DOC-EXAMPLE-BUCKET/script.vbs" `
    -Outfile "C:\script.vbs" cscript c:\script.vbs
```

Persyaratan parameter minimum

Untuk menyertakan aplikasi yang dirilis oleh Microsoft di dasar patch kustom Anda, Anda harus, minimal, menentukan produk yang ingin Anda patch. Perintah AWS Command Line Interface (AWS CLI) berikut ini menunjukkan persyaratan minimal untuk mem-patch suatu produk, seperti 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}]"
```

Jika Anda menentukan keluarga produk aplikasi Microsoft, setiap produk yang Anda tentukan harus menjadi anggota yang didukung dari keluarga produk yang dipilih. Sebagai contoh, untuk mem-patch produk "Active Directory Rights Management Services Client 2.0," Anda harus menentukan keluarga produk sebagai "Active Directory" dan bukan, misalnya, "Office" atau "SQL Server." Perintah AWS CLI berikut ini menunjukkan pasangan keluarga produk dan produk yang cocok.

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**
>
> Jika Anda menerima pesan kesalahan tentang pasangan produk dan keluarga yang tidak cocok, lihat Masalah: pasangan keluarga produk/produk yang tidak cocok untuk membantu menyelesaikan masalah ini.

MenggunakanKernel Live Patchingdi node terkelola Amazon Linux 2

Kernel Live Patchinguntuk Amazon Linux 2 memungkinkan Anda untuk menerapkan kerentanan keamanan dan patch bug kritis ke kernel Linux yang berjalan tanpa reboot atau gangguan untuk menjalankan aplikasi. Ini memungkinkan Anda mendapatkan keuntungan dari peningkatan ketersediaan layanan dan aplikasi, sambil menjaga infrastruktur Anda tetap aman dan mutakhir.Kernel Live Patchingdidukung pada instans Amazon EC2,AWS IoT Greengrassperangkat inti, danmesin virtual lokalmenjalankan Amazon Linux 2.

Untuk informasi umum tentangKernel Live Patching, lihatKernel Live Patchingdi Amazon Linux 2diPanduan Pengguna Amazon EC2 untuk Instans Linux.

Setelah Anda menghidupkanKernel Live Patchingpada node terkelola Amazon Linux 2, Anda dapat menggunakanPatch Manager, kemampuanAWS Systems Manager, untuk menerapkan patch langsung kernel ke node terkelola. MenggunakanPatch Manageradalah alternatif untuk menggunakan alur kerja yum yang ada di node untuk menerapkan pembaruan.

Sebelum Anda memulai

Untuk menggunakanPatch Manageruntuk menerapkan tambalan langsung kernel ke node terkelola Amazon Linux 2 Anda, pastikan node Anda didasarkan pada arsitektur dan versi kernel yang benar. Untuk informasi, lihat Konfigurasi dan prasyarat yang didukung dalam Panduan Pengguna Amazon EC2 untuk Instans Linux.

Topik

TentangKernel Live PatchingdanPatch Manager

Memperbarui versi kernel

Anda tidak perlu me-reboot node terkelola setelah menerapkan pembaruan patch langsung kernel. Namun, AWS menyediakan patch live kernel untuk versi kernel Amazon Linux 2 hingga tiga bulan setelah rilis. Setelah periode tiga bulan, Anda harus memperbarui ke versi kernel berikutnya untuk terus menerima patch live kernel. Sebaiknya gunakan jendela pemeliharaan untuk menjadwalkan reboot node Anda setidaknya sekali setiap tiga bulan untuk meminta pembaruan versi kernel.

Menghapus instalasi patch live kernel

Patch langsung kernel tidak dapat dihapus menggunakanPatch Manager. Sebagai gantinya, Anda dapat mematikanKernel Live Patching, yang menghapus paket RPM untuk patch langsung kernel yang diterapkan. Untuk informasi selengkapnya, lihat MematikanKernel Live PatchingmemakaiRun Command.

Kepatuhan kernel

Dalam beberapa kasus, menginstal semua perbaikan CVE dari patch live untuk versi kernel saat ini dapat membawa kernel tersebut ke dalam keadaan kepatuhan yang sama dengan versi kernel yang lebih baru. Ketika itu terjadi, versi yang lebih baru dilaporkan sebagai`Installed`, dan node terkelola dilaporkan sebagai`Compliant`. Namun, tidak ada waktu instalasi yang dilaporkan untuk versi kernel yang lebih baru.

Satu patch live kernel, beberapa CVE

Jika patch live kernel menunjuk beberapa CVE, dan CVE tersebut memiliki berbagai klasifikasi dan nilai kepelikan, hanya klasifikasi dan kepelikan tertinggi di antara CVE itu yang dilaporkan untuk patch.

Sisa bagian ini menjelaskan cara menggunakanPatch Manageruntuk menerapkan patch langsung kernel ke node terkelola yang memenuhi persyaratan ini.

Cara kerjanya

AWS merilis dua jenis patch live kernel untuk Amazon Linux 2: pembaruan keamanan dan perbaikan bug. Untuk menerapkan kedua jenis patch tersebut, Anda menggunakan dokumen dasar patch yang menargetkan hanya klasifikasi dan kepelikan yang tercantum dalam tabel berikut.

Klasifikasi	Kepelikan
`Security`	`Critical`, `Important`
`Bugfix`	`All`

Anda dapat membuat dasar patch kustom yang menargetkan hanya patch ini, atau menggunakan dasar patch `AWS-AmazonLinux2DefaultPatchBaseline` yang telah ditentukan. Dengan kata lain, Anda dapat menggunakan`AWS-AmazonLinux2DefaultPatchBaseline`dengan node terkelola Amazon Linux 2 di manaKernel Live Patchingdiaktifkan, dan pembaruan langsung kernel akan diterapkan.

> **Note**
>
> The`AWS-AmazonLinux2DefaultPatchBaseline`konfigurasi menentukan masa tunggu 7 hari setelah patch dirilis atau terakhir diperbarui sebelum diinstal secara otomatis. Jika Anda

tidak ingin menunggu 7 hari agar tambalan langsung kernel disetujui secara otomatis, Anda dapat membuat dan menggunakan baseline patch khusus. Di dasar patch Anda, Anda dapat menentukan tidak ada periode tunggu persetujuan otomatis, atau menentukan waktu yang lebih pendek atau lebih lama. Untuk informasi selengkapnya, lihat Bekerja dengan dasar patch kustom.

Kami merekomendasikan strategi berikut untuk menambal node terkelola Anda dengan pembaruan langsung kernel:

1. NyalakanKernel Live Patchingpada node terkelola Amazon Linux 2 Anda.
2. GunakanRun Command, kemampuanAWS Systems Manager, untuk menjalankan`Scan`operasi pada node terkelola Anda menggunakan yang telah ditentukan`AWS-AmazonLinux2DefaultPatchBaseline`atau baseline patch khusus yang juga hanya menargetkan`Security`pembaruan dengan tingkat keparahan diklasifikasikan sebagai`Critical`dan`Important`, dan`Bugfix`tingkat keparahan`All`.
3. Gunakan Kepatuhan, kemampuanAWS Systems Manager, untuk meninjau apakah ketidakpatuhan untuk penambalan dilaporkan untuk salah satu node terkelola yang dipindai. Jika demikian, lihat detail kepatuhan node untuk menentukan apakah patch langsung kernel hilang dari node terkelola.
4. Untuk menginstal tambalan langsung kernel yang hilang, gunakanRun Commanddengan garis dasar tambalan yang sama yang Anda tentukan sebelumnya, tetapi kali ini jalankan`Install`operasi bukannya`Scan`operasi.

   Karena patch live kernel diinstal tanpa perlu reboot, Anda dapat memilih opsi reboot `NoReboot` untuk operasi ini.

   > **Note**
   >
   > Anda masih dapat me-reboot node terkelola jika diperlukan untuk jenis tambalan lain yang diinstal di dalamnya, atau jika Anda ingin memperbarui ke kernel yang lebih baru. Dalam kasus ini, pilih opsi reboot `RebootIfNeeded` sebagai gantinya.

5. Kembali ke Kepatuhan untuk memverifikasi bahwa patch live kernel telah diinstal.

MenghidupkanKernel Live PatchingmemakaiRun Command

Untuk menghidupkanKernel Live Patching, Anda juga dapat menjalankanyumperintah pada node terkelola Anda atau gunakanRun Commanddan dokumen Manajer Sistem kustom (dokumen SSM) yang Anda buat.

Untuk informasi tentang menyalakanKernel Live Patchingdengan berlariyumperintah langsung pada node terkelola, lihatAktifkanKernel Live PatchingdiPanduan Pengguna Amazon EC2 untuk Instans Linux.

> **Note**
>
> Saat Anda mengaktifkan Kernel Live Patching, jika kernel sudah berjalan di node terkelola adalahtadidaripadakernel-4.14.165-131.185.amzn2.x86_64(versi minimum yang didukung), proses menginstal versi kernel terbaru yang tersedia dan me-reboot node terkelola. Jika node sudah berjalankernel-4.14.165-131.185.amzn2.x86_64atau lebih baru, proses tidak menginstal versi yang lebih baru dan tidak me-reboot node.

Untuk menghidupkanKernel Live PatchingmemakaiRun Command(konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command.

   -atau-

   JikaAWS Systems Managerhalaman rumah terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilihRun Command.
3. Pilih Jalankan perintah.
4. Di daftar Dokumen perintah, pilih dokumen SSM kustom `AWS-ConfigureKernelLivePatching`.
5. DiParameter perintahbagian, tentukan apakah Anda ingin node terkelola reboot sebagai bagian dari operasi ini.
6. Untuk informasi tentang bekerja dengan kontrol lainnya di halaman ini, lihat Menjalankan perintah dari konsol.
7. Pilih Jalankan.

Untuk menghidupkanKernel Live Patching(AWS CLI)

- Jalankan perintah berikut di mesin lokal Anda.

  Linux & macOS

  ```
  aws ssm send-command \
      --document-name "AWS-ConfigureKernelLivePatching" \
      --parameters "EnableOrDisable=Enable" \
      --targets "Key=instanceids,Values=instance-id"
  ```

  Server Windows

  ```
  aws ssm send-command ^
      --document-name "AWS-ConfigureKernelLivePatching" ^
      --parameters "EnableOrDisable=Enable" ^
      --targets "Key=instanceids,Values=instance-id"
  ```

  Ganti*instance-id*dengan ID node terkelola Amazon Linux 2 tempat Anda ingin mengaktifkan fitur tersebut, seperti i-02573CAFCFExample. Untuk mengaktifkan fitur pada beberapa node terkelola, Anda dapat menggunakan salah satu dari format berikut.

  - `--targets "Key=instanceids,Values=instance-id1,instance-id2"`
  - `--targets "Key=tag:tag-key,Values=tag-value"`

  Untuk informasi tentang opsi lain yang dapat Anda gunakan dalam perintah, lihatsend-commanddiAWS CLIReferensi Perintah.

Menerapkan tambalan langsung kernel menggunakanRun Command

Untuk menerapkan tambalan langsung kernel, Anda dapat menjalankannyayumperintah pada node terkelola Anda atau gunakanRun Commanddan dokumen SSM`AWS-RunPatchBaseline`.

Untuk informasi tentang menerapkan patch langsung kernel dengan menjalankanyumperintah langsung pada node terkelola, lihatTerapkan tambalan langsung kerneldiPanduan Pengguna Amazon EC2 untuk Instans Linux.

Untuk menerapkan tambalan langsung kernel menggunakanRun Command(konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command.

   -atau-

   JikaAWS Systems Managerhalaman rumah terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilihRun Command.
3. Pilih Jalankan perintah.
4. Di daftar Dokumen perintah, pilih dokumen SSM `AWS-RunPatchBaseline`.
5. Di bagian Parameter perintah, lakukan salah satu hal berikut:
   - Jika Anda memeriksa apakah patch live kernel yang baru tersedia, untuk Operasi, pilih `Scan`. UntukOpsi Reboot, jika tidak ingin node terkelola Anda reboot setelah operasi ini, pilih`NoReboot`. Setelah operasi selesai, Anda dapat memeriksa patch baru dan status kepatuhan dalam Kepatuhan.
   - Jika Anda sudah memeriksa kepatuhan patch dan siap untuk menerapkan patch live kernel yang tersedia, untuk Operasi, pilih `Install`. UntukOpsi Reboot, jika Anda tidak ingin node terkelola Anda reboot setelah operasi ini, pilih`NoReboot`.
6. Untuk informasi tentang bekerja dengan kontrol lainnya di halaman ini, lihat Menjalankan perintah dari konsol.
7. Pilih Jalankan.

Untuk menerapkan tambalan langsung kernel menggunakanRun Command(AWS CLI)

1. Untuk melakukan operasi `Scan` sebelum memeriksa hasil Anda di Kepatuhan, jalankan perintah berikut dari mesin lokal Anda.

   Linux & macOS

   ```
   aws ssm send-command \
       --document-name "AWS-RunPatchBaseline" \
       --targets "Key=InstanceIds,Values=instance-id" \
       --parameters '{"Operation":["Scan"],"RebootOption":["RebootIfNeeded"]}'
   ```

Server Windows

```
aws ssm send-command ^
    --document-name "AWS-RunPatchBaseline" ^
    --targets "Key=InstanceIds,Values=instance-id" ^
    --parameters {\"Operation\":[\"Scan\"],\"RebootOption\":[\"RebootIfNeeded
\"]}
```

Untuk informasi tentang opsi lain yang dapat Anda gunakan dalam perintah, lihatsend-commanddiAWS CLIReferensi Perintah.

2. Untuk melakukan operasi `Install` setelah memeriksa hasil Anda di Kepatuhan, jalankan perintah berikut dari mesin lokal Anda.

   Linux & macOS

   ```
   aws ssm send-command \
       --document-name "AWS-RunPatchBaseline" \
       --targets "Key=InstanceIds,Values=instance-id" \
       --parameters '{"Operation":["Install"],"RebootOption":["NoReboot"]}'
   ```

   Server Windows

   ```
   aws ssm send-command ^
       --document-name "AWS-RunPatchBaseline" ^
       --targets "Key=InstanceIds,Values=instance-id" ^
       --parameters {\"Operation\":[\"Install\"],\"RebootOption\":[\"NoReboot\"]}
   ```

Dalam kedua perintah sebelumnya, ganti*instance-id*dengan ID node terkelola Amazon Linux 2 tempat Anda ingin menerapkan tambalan langsung kernel, seperti i-02573CAFCFExample. Untuk mengaktifkan fitur pada beberapa node terkelola, Anda dapat menggunakan salah satu dari format berikut.

- `--targets "Key=instanceids,Values=instance-id1,instance-id2"`
- `--targets "Key=tag:tag-key,Values=tag-value"`

Untuk informasi tentang opsi lain yang dapat Anda gunakan dalam perintah ini, lihatsend-commanddiAWS CLIReferensi Perintah.

MematikanKernel Live PatchingmemakaiRun Command

Untuk mematikanKernel Live Patching, Anda juga dapat menjalankanyumperintah pada node terkelola Anda atau gunakanRun Commanddan dokumen SSM khususAWS-ConfigureKernelLivePatching.

> **Note**
>
> Jika Anda tidak perlu lagi menggunakan Kernel Live Patching, Anda dapat menonaktifkannya kapan saja. Dalam kebanyakan kasus, Anda tidak perlu menonaktifkan fitur.

Untuk informasi tentang mematikanKernel Live Patchingdengan berlariyumperintah langsung pada node terkelola, lihatAktifkanKernel Live PatchingdiPanduan Pengguna Amazon EC2 untuk Instans Linux.

> **Note**
>
> Saat Anda mematikanKernel Live Patching, proses mencopot pemasanganKernel Live Patchingplugin dan kemudian reboot node terkelola.

Untuk mematikanKernel Live PatchingmemakaiRun Command(konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command.

   -atau-

   JikaAWS Systems Managerhalaman rumah terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilihRun Command.
3. Pilih Jalankan perintah.
4. Di daftar Dokumen perintah, pilih dokumen SSM `AWS-ConfigureKernelLivePatching`.
5. Di bagian Parameter perintah, tentukan nilai untuk parameter yang diperlukan.

6. Untuk informasi tentang bekerja dengan kontrol lainnya di halaman ini, lihat Menjalankan perintah dari konsol.
7. Pilih Jalankan.

Untuk mematikanKernel Live Patching(AWS CLI)

- Jalankan perintah yang serupa dengan yang berikut ini.

  Linux & macOS

  ```
  aws ssm send-command \
      --document-name "AWS-ConfigureKernelLivePatching" \
      --targets "Key=instanceIds,Values=instance-id" \
      --parameters "EnableOrDisable=Disable"
  ```

  Server Windows

  ```
  aws ssm send-command ^
      --document-name "AWS-ConfigureKernelLivePatching" ^
      --targets "Key=instanceIds,Values=instance-id" ^
      --parameters "EnableOrDisable=Disable"
  ```

  Ganti*instance-id*dengan ID node terkelola Amazon Linux 2 tempat Anda ingin mematikan fitur, seperti i-02573CAFCFExample. Untuk mematikan fitur pada beberapa node terkelola, Anda dapat menggunakan salah satu dari format berikut.

  - `--targets "Key=instanceids,Values=instance-id1,instance-id2"`
  - `--targets "Key=tag:tag-key,Values=tag-value"`

  Untuk informasi tentang opsi lain yang dapat Anda gunakan dalam perintah, lihatsend-commanddiAWS CLIReferensi Perintah.

Bekerja denganPatch Manager (konsol)

Untuk menggunakanPatch Manager, suatu kemampuanAWS Systems Manager, selesaikan tugas berikut. Tugas-tugas ini diterangkan dengan lebih detail dalam bagian ini.

1. Verifikasikan bahwa dasar patch yang telah ditetapkan AWS untuk setiap jenis sistem operasi yang Anda gunakan memenuhi kebutuhan Anda. Jika tidak, buat dasar patch yang mendefinisikan satu set standar patch untuk tipe node tersebut dan atur sebagai default.
2. Susun node yang dikelola ke dalam grup patch dengan menggunakan tag Amazon Elastic Compute Cloud (Amazon EC2) (opsional, tetapi direkomendasikan).
3. Lakukan salah satu dari berikut:
   - (Disarankan) Mengkonfigurasi kebijakan tambalanQuick Setup, kemampuan Systems Manager, yang memungkinkan Anda menginstal patch yang hilang pada jadwal untuk seluruh organisasi, subset unit organisasi, atau satuAkun AWS. Untuk informasi selengkapnya, lihat Patch Managerkonfigurasi penambalan organisasi.
   - Buat jendela pemeliharaan yang menggunakan dokumen Systems Manager (dokumen SSM)`AWS-RunPatchBaseline` dalam jenisRun Command tugas. Untuk informasi selengkapnya, lihat Walkthrough: Membuat jendela pemeliharaan untuk patching (konsol).
   - Jalankan secara manual`AWS-RunPatchBaseline` dalamRun Command operasi. Untuk informasi selengkapnya, lihat Menjalankan perintah dari konsol.
   - Menambal node sesuai permintaan secara manual menggunakan fitur Patch now. Untuk informasi selengkapnya, lihat Menambal node terkelola sesuai permintaan.
4. Pantau patching untuk memverifikasi kepatuhan dan menyelidiki kegagalan.

Topik

Membuat kebijakan patch

Kebijakan patch adalah konfigurasi yang Anda atur menggunakanQuick Setup, kemampuanAWS Systems Manager. Kebijakan patch memberikan kontrol yang lebih luas dan lebih terpusat atas

operasi patching Anda daripada yang tersedia dengan metode lain untuk mengkonfigurasi patching. Kebijakan patch menentukan jadwal dan garis dasar yang akan digunakan saat secara otomatis menambal node dan aplikasi Anda.

Untuk informasi lain, lihat topik berikut:

- Menggunakan kebijakan Quick Setup tambalan
- Patch Managerkonfigurasi penambalan organisasi

Melihat ringkasan Patch Dasbor

Tab Dasbor di Patch Manager memberi Anda tampilan ringkasan di konsol yang dapat Anda gunakan untuk memantau operasi penambalan Anda dalam tampilan gabungan. Patch Manageradalah kemampuanAWS Systems Manager. Pada tab Dasbor, Anda dapat melihat yang berikut:

- Cuplikan berapa banyak node terkelola yang sesuai dan tidak sesuai dengan aturan patching.
- Cuplikan usia hasil kepatuhan patch untuk node terkelola Anda.
- Hitungan terkait berapa banyak node terkelola yang tidak patuh yang ada untuk masing-masing alasan paling umum untuk ketidakpatuhan.
- Daftar tertaut dari operasi penambalan terbaru.
- Daftar terkait tugas patching berulang yang telah disiapkan.

Untuk melihat ringkasan Patch Dasbor

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Patch Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.
3. Pilih tab Dasbor.
4. Gulir ke bagian yang berisi data ringkasan yang ingin Anda lihat:

   - Manajemen instans Amazon EC2

- Ringkasan kepatuhan
- Ketidakpatuhan diperhitungkan
- Laporan kepatuhan
- Operasi berbasis kebijakan non-patch
- Tugas berulang berbasis kebijakan non-patch

Mengerjakan laporan kepatuhan patch

Gunakan informasi dalam topik berikut untuk membantu Anda menghasilkan dan bekerja dengan laporan kepatuhan patch diPatch Manager, kemampuanAWS Systems Manager.

Informasi dalam topik berikut berlaku apa pun metode atau jenis konfigurasi yang Anda gunakan untuk operasi patching Anda:

- Kebijakan tambalan yang dikonfigurasi diQuick Setup
- Opsi Manajemen Host yang dikonfigurasi diQuick Setup
- Jendela pemeliharaan untuk menjalankan patch`Scan` atau`Install` tugas
- Patch on-demand sekarang beroperasi

> **Important**
>
> Jika Anda memiliki beberapa jenis operasi untuk memindai instans Anda untuk kepatuhan patch, perhatikan bahwa setiap pemindaian menimpa data kepatuhan patch dari pemindaian sebelumnya. Akibatnya, Anda mungkin berakhir dengan hasil yang tidak terduga dalam data kepatuhan patch Anda. Untuk informasi selengkapnya, lihat Menghindari penimpaan data kepatuhan patch yang tidak disengaja.
>
> Untuk memverifikasi dasar patch mana yang digunakan untuk menghasilkan informasi kepatuhan terbaru, buka tab Pelaporan kepatuhanPatch Manager, cari baris untuk node terkelola yang ingin Anda informasikan, lalu pilih ID dasar di kolom ID Dasar yang digunakan.

Topik

- Melihat hasil kepatuhan patch
- Menghasilkan laporan kepatuhan patch .csv
- Remediasi node terkelola yang tidak sesuai denganPatch Manager

- Menghindari penimpaan data kepatuhan patch yang tidak disengaja

Melihat hasil kepatuhan patch

Gunakan prosedur ini untuk melihat informasi kepatuhan tambalan tentang node terkelola Anda.

Prosedur ini berlaku untuk operasi patch yang menggunakan dokumen `AWS-RunPatchBaseline`. Untuk informasi tentang melihat informasi kepatuhan patch untuk operasi patch yang menggunakan dokumen `AWS-RunPatchBaselineAssociation`, lihat Mengidentifikasi node terkelola yang tidak sesuai.

> **Note**
>
> Operasi pemindaian patch untuk Quick Setup dan Explorer menggunakan `AWS-RunPatchBaselineAssociation` dokumen. Quick Setupdan Explorer keduanya memiliki kemampuanAWS Systems Manager.

Identifikasi solusi patch untuk masalah CVE tertentu (Linux)

Untuk banyak sistem operasi berbasis Linux, hasil kepatuhan patch menunjukkan masalah buletin Common Vulnerabilities and Exposure (CVE) apa yang diselesaikan oleh patch tertentu. Informasi ini dapat membantu Anda menentukan seberapa mendesak Anda perlu menginstal patch yang hilang atau gagal.

Detail CVE disertakan untuk versi yang didukung dari jenis sistem operasi berikut:

- 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**
>
> Secara default, CentOS dan CentOS Stream tidak memberikan informasi CVE tentang pembaruan. Namun, Anda dapat mengizinkan support ini dengan menggunakan repositori pihak ketiga seperti repositori Extra Packages for Enterprise Linux (EPEL) yang diterbitkan oleh Fedora. Untuk informasi, lihat EPEL di Fedora Wiki.
> Saat ini, nilai ID CVE dilaporkan hanya untuk tambalan dengan status atau. `Missing` `Failed`

Anda juga dapat menambahkan ID CVE ke daftar patch yang disetujui atau ditolak dalam dasar patch Anda, sesuai dengan situasi dan jaminan tujuan patching Anda.

Untuk informasi tentang bekerja dengan daftar patch yang disetujui dan ditolak, lihat topik berikut:

- Bekerja dengan dasar patch kustom
- Tentang format nama paket untuk daftar patch yang disetujui dan ditolak
- Cara kerja aturan dasar patch pada sistem berbasis Linux
- Cara menginstal patch

> **Note**
>
> Dalam beberapa kasus, Microsoft merilis patch untuk aplikasi yang tidak menentukan tanggal dan waktu yang diperbarui. Dalam kasus ini, tanggal dan waktu yang diperbarui `01/01/1970` disediakan secara default.

Melihat hasil patching compliance

Gunakan prosedur berikut ini untuk melihat hasil kepatuhan patch di konsol AWS Systems Manager.

> **Note**
>
> Untuk informasi tentang menghasilkan laporan kepatuhan patch yang diunduh ke bucket Amazon Simple Storage Service (Amazon S3), lihat Menghasilkan laporan kepatuhan patch .csv.

Untuk melihat hasil kepatuhan patch

1. Lakukan salah satu hal berikut ini.

   Opsi 1 (disarankan) - Navigasi dariPatch Manager, kemampuanAWS Systems Manager:

   - Di panel navigasi, pilih Patch Manager.

     -atau-

     Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.
   - Pilih tab Pelaporan kepatuhan.
   - Di area detail penambalan Node, pilih ID node dari node terkelola yang ingin Anda tinjau hasil kepatuhan tambalan.
   - Di area Detail, dalam daftar Properti, pilih Patch.

   Opsi 2 — Buka Compliance, suatu kemampuan AWS Systems Manager:

   - Di panel navigasi, pilih Kepatuhan.

     -atau-

     Jika halaman beranda AWS Systems Manager terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Kepatuhan di panel navigasi.
   - Untuk Ringkasan sumber daya kepatuhan, pilih nomor di kolom untuk jenis sumber daya patch yang ingin Anda tinjau, seperti Sumber daya yang tidak patuh.
   - Di bawah ini, dalam daftar Sumber Daya, pilih ID node terkelola yang ingin Anda tinjau hasil kepatuhan tambalan.
   - Di area Detail, dalam daftar Properti, pilih Patch.

   Opsi 3 — Navigasi dariFleet Manager, kemampuanAWS Systems Manager.

   - Di panel navigasi, pilih Fleet Manager.

     -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.

- Di area Instans terkelola, pilih ID node terkelola yang ingin Anda tinjau hasil kepatuhan tambalan.
- Di area Detail, dalam daftar Properti, pilih Patch.

2. (Opsional) Di kotak pencarian (🔍), pilih dari filter yang tersedia.

   Sebagai contoh, untuk Red Hat Enterprise Linux (RHEL), pilih dari yang berikut ini:

   - Nama
   - Klasifikasi
   - Keadaan
   - Kepelikan

   Untuk Windows Server, pilih dari yang berikut ini:

   - KB
   - Klasifikasi
   - Keadaan
   - Kepelikan

3. Pilih salah satu nilai yang tersedia untuk jenis filter yang Anda pilih. Misalnya, jika Anda memilih Negara, sekarang pilih status kepatuhan seperti InstalledPendingReboot, Gagal atau Hilang.

   > **Note**
   >
   > Saat ini, nilai ID CVE dilaporkan hanya untuk tambalan dengan status atau. `Missing` `Failed`

4. Bergantung pada status kepatuhan node terkelola, Anda dapat memilih tindakan apa yang harus diambil untuk memperbaiki node yang tidak patuh.

Misalnya, Anda dapat memilih untuk segera menambal node terkelola yang tidak sesuai. Untuk informasi tentang menambal node terkelola sesuai permintaan, lihatMenambal node terkelola sesuai permintaan.

Untuk informasi tentang nilai keadaan kepatuhan patch, lihat Memahami nilai keadaan kepatuhan patch.

Menghasilkan laporan kepatuhan patch .csv

Anda dapat menggunakan konsol AWS Systems Manager untuk menghasilkan laporan kepatuhan patch yang disimpan sebagai file .csv ke bucket Amazon Simple Storage Service (Amazon S3) yang Anda pilih. Anda dapat membuat laporan sesuai permintaan tunggal atau menentukan jadwal untuk menghasilkan laporan secara otomatis.

Laporan dapat dibuat untuk satu node terkelola atau untuk semua node terkelola di yang Anda pilih Akun AWS danWilayah AWS. Untuk satu node, laporan berisi detail komprehensif, termasuk ID tambalan yang terkait dengan node yang tidak sesuai. Untuk laporan tentang semua node terkelola, hanya informasi ringkasan dan jumlah patch node yang tidak sesuai yang disediakan.

Setelah laporan dibuat, Anda dapat menggunakan alat seperti Amazon QuickSight untuk mengimpor dan menganalisis data. Amazon QuickSight adalah layanan intelijen bisnis (BI) yang dapat Anda gunakan untuk mengeksplorasi dan menafsirkan informasi dalam lingkungan visual yang interaktif. Untuk informasi selengkapnya, lihat Panduan QuickSight Pengguna Amazon.

> **Note**
>
> Saat membuat baseline patch kustom, Anda dapat menentukan tingkat keparahan kepatuhan untuk tambalan yang disetujui oleh baseline patch tersebut, seperti atau. `Critical` `High` Jika status tambalan dari setiap tambalan yang disetujui dilaporkan sebagai`Missing`, maka tingkat keparahan kepatuhan keseluruhan baseline patch yang dilaporkan adalah tingkat keparahan yang Anda tentukan.

Anda juga dapat menentukan topik Amazon Simple Notification Service (Amazon SNS) yang digunakan untuk mengirim notifikasi ketika laporan dibuat.

Peran layanan untuk membuat laporan kepatuhan patch

Saat pertama kali Anda membuat laporan, Systems Manager membuat peran Automation asumsi bernama `AWS-SystemsManager-PatchSummaryExportRole` untuk digunakan untuk proses ekspor ke S3.

> **Note**
>
> Jika Anda mengekspor data kepatuhan ke bucket S3 terenkripsi, Anda harus memperbarui kebijakan AWS KMS kunci terkait untuk memberikan izin yang diperlukan. `AWS-SystemsManager-PatchSummaryExportRole` Misalnya, tambahkan izin yang serupa dengan ini ke AWS KMS kebijakan bucket S3 Anda:
>
> ```
> {
>     "Effect": "Allow",
>     "Action": [
>         "kms:GenerateDataKey"
>     ],
>     "Resource": "role-arn"
> }
> ```
>
> Ganti *`role-arn`* dengan Amazon Resource Name (ARN) yang dibuat di akun Anda, dalam format. `arn:aws:iam::111222333444:role/service-role/AWS-SystemsManager-PatchSummaryExportRole`
> Untuk informasi selengkapnya, lihat Kebijakan kunci di AWS KMS di Panduan Developer AWS Key Management Service.

Pertama kali Anda membuat laporan pada jadwal, Systems Manager membuat peran layanan lain bernama`AWS-EventBridge-Start-SSMAutomationRole`, bersama dengan peran layanan `AWS-SystemsManager-PatchSummaryExportRole` (jika belum dibuat) untuk digunakan untuk proses ekspor. `AWS-EventBridge-Start-SSMAutomationRole`memungkinkan Amazon EventBridge untuk memulai otomatisasi menggunakan runbook AWS- ExportPatchReportTo S3.

Kami merekomendasikan agar tidak mencoba mengubah kebijakan dan peran ini. Melakukan hal itu dapat menyebabkan pembuatan laporan kepatuhan patch gagal. Untuk informasi selengkapnya, lihat Memecahkan masalah pembuatan laporan kepatuhan patch.

Topik

- Apa yang ada dalam laporan kepatuhan patch yang dibuat?
- Membuat laporan kepatuhan tambalan untuk satu node terkelola

- Membuat laporan kepatuhan tambalan untuk semua node terkelola
- Melihat riwayat pelaporan kepatuhan patch
- Melihat jadwal pelaporan kepatuhan patch
- Memecahkan masalah pembuatan laporan kepatuhan patch

Apa yang ada dalam laporan kepatuhan patch yang dibuat?

Topik ini menyediakan informasi tentang jenis konten yang disertakan dalam laporan kepatuhan patch yang dihasilkan dan diunduh ke bucket S3 tertentu.

Format laporan untuk satu node terkelola

Laporan yang dihasilkan untuk satu node terkelola memberikan ringkasan dan informasi rinci.

Unduh laporan sampel (simpul tunggal)

Informasi ringkasan untuk satu node terkelola mencakup yang berikut:

- Indeks
- ID instans
- Nama instans
- IP instans
- Nama platform
- Versi platform
- Versi SSM Agent
- Dasar patch
- Grup patch
- Status kepatuhan
- Kepelikan kepatuhan
- Jumlah patch dengan kepelikan Kritis yang tidak patuh
- Jumlah patch dengan kepelikan Tinggi yang tidak patuh
- Jumlah patch dengan kepelikan Medium yang tidak patuh
- Jumlah patch dengan kepelikan Rendah yang tidak patuh
- Jumlah patch dengan kepelikan Informasional yang tidak patuh
- Jumlah patch dengan kepelikan Tidak Ditentukan yang tidak patuh

Informasi terperinci untuk satu node terkelola mencakup yang berikut:

- Indeks
- ID instans
- Nama instans
- Nama patch
- ID KB/ID Patch
- Keadaan patch
- Waktu laporan terakhir
- Tingkat kepatuhan
- Kepelikan patch
- Klasifikasi patch
- ID CVE
- Dasar patch
- URL Log
- IP instans
- Nama platform
- Versi platform
- Versi SSM Agent

> **Note**
>
> Saat membuat baseline patch kustom, Anda dapat menentukan tingkat keparahan kepatuhan untuk tambalan yang disetujui oleh baseline patch tersebut, seperti atau. `Critical` `High` Jika status tambalan dari setiap tambalan yang disetujui dilaporkan sebagai`Missing`, maka tingkat keparahan kepatuhan keseluruhan baseline patch yang dilaporkan adalah tingkat keparahan yang Anda tentukan.

Format laporan untuk semua node terkelola

Laporan yang dibuat untuk semua node terkelola hanya menyediakan informasi ringkasan.

Unduh contoh laporan (semua node terkelola)

Informasi ringkasan untuk semua node terkelola mencakup yang berikut:

- Indeks
- ID instans
- Nama instans
- IP instans
- Nama platform
- Versi platform
- Versi SSM Agent
- Dasar patch
- Grup patch
- Status kepatuhan
- Kepelikan kepatuhan
- Jumlah patch dengan kepelikan Kritis yang tidak patuh
- Jumlah patch dengan kepelikan Tinggi yang tidak patuh
- Jumlah patch dengan kepelikan Medium yang tidak patuh
- Jumlah patch dengan kepelikan Rendah yang tidak patuh
- Jumlah patch dengan kepelikan Informasional yang tidak patuh
- Jumlah patch dengan kepelikan Tidak Ditentukan yang tidak patuh

Membuat laporan kepatuhan tambalan untuk satu node terkelola

Gunakan prosedur berikut untuk menghasilkan laporan ringkasan tambalan untuk satu node terkelola di AndaAkun AWS. Laporan untuk satu node terkelola memberikan rincian tentang setiap patch yang tidak sesuai, termasuk nama patch dan ID.

Untuk menghasilkan laporan kepatuhan tambalan untuk satu node terkelola

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Patch Manager.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.

3. Pilih tab Pelaporan kepatuhan.
4. Pilih tombol untuk baris node terkelola yang ingin Anda hasilkan laporannya, lalu pilih Lihat detail.
5. Di bagian Ringkasan tambalan, pilih Ekspor ke S3.
6. Untuk Nama laporan, masukkan nama untuk membantu Anda mengidentifikasi laporan nanti.
7. Untuk Frekuensi pelaporan, pilih salah satu hal berikut:
   - Sesuai permintaan — Buat laporan satu kali. Lewati ke Langkah 9.
   - Sesuai jadwal — Tentukan jadwal berulang untuk membuat laporan secara otomatis. Lanjutkan ke Langkah 8.
8. Untuk Jenis jadwal, tentukan ekspresi rate, seperti setiap 3 hari, atau berikan ekspresi cron untuk mengatur frekuensi laporan.

   Untuk informasi tentang ekspresi cron, lihat Referensi: Ekspresi cron dan rate untuk Systems Manager.
9. Untuk Nama bucket, pilih nama bucket S3 tempat Anda ingin menyimpan file laporan .csv.

   > ⚠ Important
   >
   > Jika Anda bekerja di sebuah Wilayah AWS yang diluncurkan setelah 20 Maret 2019, Anda harus memilih bucket S3 di Region yang sama. Region yang diluncurkan setelah tanggal tersebut dimatikan secara default. Untuk informasi selengkapnya dan daftar Wilayah ini, lihat Mengaktifkan Wilayah di. Referensi Umum Amazon Web Services
10. (Opsional) Untuk mengirim pemberitahuan saat laporan dibuat, keluarkan bagian topik SNS, lalu pilih topik Amazon SNS yang ada dari topik SNS Nama Sumber Daya Amazon (ARN).
11. Pilih Kirim.

Untuk informasi tentang melihat riwayat laporan yang dibuat, lihat Melihat riwayat pelaporan kepatuhan patch.

Untuk informasi tentang melihat detail jadwal pelaporan yang telah Anda buat, lihat Melihat jadwal pelaporan kepatuhan patch.

Membuat laporan kepatuhan tambalan untuk semua node terkelola

Gunakan prosedur berikut untuk menghasilkan laporan ringkasan tambalan untuk semua node terkelola di AndaAkun AWS. Laporan untuk semua node terkelola menunjukkan node mana yang tidak sesuai dan jumlah tambalan yang tidak sesuai. Ini tidak memberikan nama atau pengidentifikasi lain dari patch. Untuk detail tambahan ini, Anda dapat membuat laporan kepatuhan patch untuk satu node terkelola. Untuk informasi, lihat Membuat laporan kepatuhan tambalan untuk satu node terkelola sebelumnya dalam topik ini.

Untuk menghasilkan laporan kepatuhan tambalan untuk semua node terkelola

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Patch Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.
3. Pilih tab Pelaporan kepatuhan.
4. Pilih Ekspor ke S3. (Jangan pilih ID node terlebih dahulu.)
5. Untuk Nama laporan, masukkan nama untuk membantu Anda mengidentifikasi laporan nanti.
6. Untuk Frekuensi pelaporan, pilih salah satu hal berikut:
   - Sesuai permintaan — Buat laporan satu kali. Lewati ke Langkah 8.
   - Sesuai jadwal — Tentukan jadwal berulang untuk membuat laporan secara otomatis. Lanjutkan ke Langkah 7.
7. Untuk Jenis jadwal, tentukan ekspresi rate, seperti setiap 3 hari, atau berikan ekspresi cron untuk mengatur frekuensi laporan.

   Untuk informasi tentang ekspresi cron, lihat Referensi: Ekspresi cron dan rate untuk Systems Manager.
8. Untuk Nama bucket, pilih nama bucket S3 tempat Anda ingin menyimpan file laporan .csv.

> **Important**
>
> Jika Anda bekerja di sebuah Wilayah AWS yang diluncurkan setelah 20 Maret 2019, Anda harus memilih bucket S3 di Region yang sama. Region yang diluncurkan setelah tanggal tersebut dimatikan secara default. Untuk informasi selengkapnya dan daftar Wilayah ini, lihat Mengaktifkan Wilayah di. Referensi Umum Amazon Web Services

9. (Opsional) Untuk mengirim pemberitahuan saat laporan dibuat, keluarkan bagian topik SNS, lalu pilih topik Amazon SNS yang ada dari topik SNS Nama Sumber Daya Amazon (ARN).
10. Pilih Kirim.

Untuk informasi tentang melihat riwayat laporan yang dibuat, lihat Melihat riwayat pelaporan kepatuhan patch.

Untuk informasi tentang melihat detail jadwal pelaporan yang telah Anda buat, lihat Melihat jadwal pelaporan kepatuhan patch.

Melihat riwayat pelaporan kepatuhan patch

Gunakan informasi dalam topik ini untuk membantu Anda melihat detail tentang laporan kepatuhan patch yang dibuat di Akun AWS Anda.

Untuk melihat riwayat pelaporan kepatuhan patch

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Patch Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.
3. Pilih tab Pelaporan kepatuhan.
4. Pilih Lihat semua ekspor S3, lalu pilih tab Riwayat ekspor.

Melihat jadwal pelaporan kepatuhan patch

Gunakan informasi dalam topik ini untuk membantu Anda melihat detail tentang jadwal pelaporan kepatuhan patch yang dibuat di Akun AWS Anda.

Untuk melihat riwayat pelaporan kepatuhan patch

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Patch Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.
3. Pilih tab Pelaporan kepatuhan.
4. Pilih Lihat semua ekspor S3, lalu pilih tab Aturan jadwal laporan.

Memecahkan masalah pembuatan laporan kepatuhan patch

Gunakan informasi berikut untuk membantu Anda memecahkan masalah dengan menghasilkan pembuatan laporan kepatuhan tambalan diPatch Manager, kapabilitas. AWS Systems Manager

Topik

- Pesan melaporkan bahwa kebijakan AWS-SystemsManager-PatchManagerExportRolePolicy rusak
- Setelah menghapus kebijakan atau peran kepatuhan patch, laporan terjadwal tidak berhasil dibuat

Pesan melaporkan bahwa kebijakan **`AWS-SystemsManager-PatchManagerExportRolePolicy`** rusak

Masalah: Anda menerima pesan kesalahan yang serupa dengan berikut ini, menunjukkan `AWS-SystemsManager-PatchManagerExportRolePolicy` rusak:

```
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.
```

- Solusi: Gunakan Patch Manager konsol atau AWS CLI untuk menghapus peran dan kebijakan yang terpengaruh sebelum membuat laporan kepatuhan patch baru.

  Untuk menghapus kebijakan korup menggunakan konsol

  1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
  2. Lakukan salah satu hal berikut ini:

     Laporan sesuai permintaan — Jika masalah terjadi saat membuat laporan sesuai permintaan satu kali, di navigasi sebelah kiri, pilih Kebijakan, cari `AWS-SystemsManager-PatchManagerExportRolePolicy`, lalu hapus kebijakan tersebut. Selanjutnya, pilih Peran, cari `AWS-SystemsManager-PatchSummaryExportRole`, lalu hapus peran tersebut.

     Laporan terjadwal — Jika masalah terjadi saat membuat laporan sesuai jadwal, di navigasi kiri, pilih Kebijakan, cari satu per satu`AWS-SystemsManager-PatchManagerExportRolePolicy`, `AWS-EventBridge-Start-SSMAutomationRolePolicy` dan hapus setiap kebijakan. Selanjutnya, pilih Peran, cari satu per satu untuk `AWS-EventBridge-Start-SSMAutomationRole` dan `AWS-SystemsManager-PatchSummaryExportRole`, dan hapus masing-masing peran.

  Untuk menghapus kebijakan korup menggunakan AWS CLI

  Ganti *`nilai placeholder`* dengan ID akun Anda.

  - Jika masalah terjadi saat membuat laporan on-demand satu kali, jalankan perintah berikut:

    ```
    aws iam delete-policy --policy-arn arn:aws:iam::account-id:policy/AWS-SystemsManager-PatchManagerExportRolePolicy
    ```

    ```
    aws iam delete-role --role-name AWS-SystemsManager-PatchSummaryExportRole
    ```

    Jika masalah terjadi saat membuat laporan pada jadwal, jalankan perintah berikut:

    ```
    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
```

Setelah menyelesaikan salah satu prosedur, ikuti langkah-langkah untuk membuat atau menjadwalkan laporan kepatuhan patch baru.

Setelah menghapus kebijakan atau peran kepatuhan patch, laporan terjadwal tidak berhasil dibuat

Masalah: Pertama kali Anda membuat laporan, Systems Manager membuat peran layanan dan kebijakan untuk digunakan untuk proses ekspor (`AWS-SystemsManager-PatchSummaryExportRole` dan `AWS-SystemsManager-PatchManagerExportRolePolicy`). Pertama kali Anda membuat laporan terjadwal, Systems Manager membuat peran layanan lain dan kebijakan (`AWS-EventBridge-Start-SSMAutomationRole` dan `AWS-EventBridge-Start-SSMAutomationRolePolicy`). Ini memungkinkan Amazon EventBridge memulai otomatisasi menggunakan runbook AWS- ExportPatchReportTo S3.

Jika Anda menghapus salah satu kebijakan atau peran ini, hubungan antara jadwal Anda dan bucket S3 yang ditentukan dan topik Amazon SNS mungkin hilang.

- Solusi: Untuk mengatasi masalah ini, kami merekomendasikan untuk menghapus jadwal sebelumnya dan membuat jadwal baru untuk menggantikan jadwal yang mengalami masalah.

Remediasi node terkelola yang tidak sesuai denganPatch Manager

Topik di bagian ini menyediakan gambaran umum tentang cara mengidentifikasi node terkelola yang berada di luar kepatuhan patch dan cara membawa node ke dalam kepatuhan.

Topik

- Mengidentifikasi node terkelola yang tidak sesuai
- Memahami nilai keadaan kepatuhan patch
- Menambal node terkelola yang tidak sesuai

Mengidentifikasi node terkelola yang tidak sesuai

O node ut-of-compliance terkelola diidentifikasi ketika salah satu dari dua AWS Systems Manager dokumen (dokumen SSM) dijalankan. Dokumen-dokumen SSM ini mereferensikan baseline patch yang sesuai untuk setiap node terkelolaPatch Manager, kemampuan. AWS Systems Manager Mereka kemudian mengevaluasi status patch dari node terkelola dan kemudian membuat hasil kepatuhan tersedia untuk Anda.

Ada dua dokumen SSM yang digunakan untuk mengidentifikasi atau memperbarui node terkelola yang tidak sesuai: dan. `AWS-RunPatchBaseline` `AWS-RunPatchBaselineAssociation` Masing-masing digunakan oleh proses yang berbeda, dan hasil kepatuhan mereka tersedia melalui saluran yang berbeda. Tabel berikut ini menguraikan perbedaan antara dokumen-dokumen ini.

> **Note**
>
> Data kepatuhan tambalan dari Patch Manager dapat dikirim keAWS Security Hub. Security Hub memberi Anda pandangan komprehensif tentang pemberitahuan keamanan prioritas tinggi dan status kepatuhan Anda. Hub juga memantau status patching armada Anda. Untuk informasi selengkapnya, lihat Integrasi dengan Patch ManagerAWS Security Hub.

	AWS-RunPatchBaseline	**AWS-RunPatchBaselineAssociation**
Proses yang menggunakan dokumen	Patch on demand - Anda dapat memindai atau menambal node yang dikelola sesuai permintaan menggunakan opsi Patch now. Untuk informasi, lihat Menambal node terkelola sesuai permintaan. Kebijakan Quick Setup patch Systems Manager — Anda dapat membuat konfigurasi tambalanQuick Setup, kemampuanAWS	Manajemen Quick Setup Host Systems Manager — Anda dapat mengaktifkan opsi konfigurasi Manajemen Host Quick Setup untuk memindai instans terkelola untuk kepatuhan patch setiap hari. Untuk informasi, lihat Manajemen host Amazon EC2. Systems Manager Explorer— Bila Anda mengizinkanExplorer, kemampuanAWS Systems

	AWS-RunPatchBaseline	AWS-RunPatchBaseli neAssociation
	Systems Manager, yang dapat memindai atau menginsta l tambalan yang hilang pada jadwal terpisah untuk seluruh organisasi, subset unit organisasi, atau satu. Akun AWS Untuk informasi, lihat Patch Managerkonfigurasi penambalan organisasi. Jalankan perintah — Anda dapat menjalankan `AWS-RunPatchBaseline` secara manual dalam operasi diRun Command, kemampuanAWS Systems Manager. Untuk informasi, lihat Menjalankan perintah dari konsol. Jendela pemeliharaan - Anda dapat membuat jendela pemeliharaan yang menggunakan dokumen SSM `AWS-RunPatchBaseline` dalam tipe Run Command tugas. Untuk informasi, lihat Walkthrough: Membuat jendela pemeliharaan untuk patching (konsol).	Manager, itu secara teratur memindai instans terkelola Anda untuk kepatuhan patch dan hasil laporan di dasbor. Explorer

	AWS-RunPatchBaseline	**AWS-RunPatchBaselineAssociation**
Format data hasil pemindaian patch	Setelah `AWS-RunPatchBaseline` berjalan, Patch Manager mengirimkan `AWS:PatchSummary` objek ke Inventory, kemampuan AWS Systems Manager.	Setelah `AWS-RunPatchBaselineAssociation` berjalan, Patch Manager mengirimkan `AWS:ComplianceItem` objek ke Systems Manager Inventory.
Melihat laporan kepatuhan patch di konsol	Anda dapat melihat informasi kepatuhan patch untuk proses yang menggunakan `AWS-RunPatchBaseline` dalam Kepatuhan Konfigurasi Systems Manager dan Bekerja dengan node terkelola. Untuk informasi selengkapnya, lihat Melihat hasil kepatuhan patch.	Jika digunakan Quick Setup untuk memindai instans terkelola untuk kepatuhan tambalan, Anda dapat melihat laporan kepatuhan di Systems Manager State Manager, yang dapat diakses menggunakan tombol Lihat hasil diQuick Setup. Jika Anda menggunakannya Explorer untuk memindai instans terkelola untuk kepatuhan tambalan, Anda dapat melihat laporan kepatuhan di keduanya Explorer dan Systems Manager OpsCenter.

	AWS-RunPatchBaseline	**AWS-RunPatchBaselineAssociation**
Perintah AWS CLI untuk melihat hasil kepatuhan patch	Untuk proses yang digunakan `AWS-RunPatchBaseline`, Anda dapat menggunakan AWS CLI perintah berikut untuk melihat informasi ringkasan tentang tambalan pada node terkelola. • describe-instance-patch-states • describe-instance-patch-states-for-patch-group • describe-patch-group-state	Untuk proses yang menggunakan `AWS-RunPatchBaselineAssociation`, Anda dapat menggunakan perintah AWS CLI berikut ini untuk melihat informasi ringkasan tentang patch pada sebuah instans. • list-compliance-items
Operasi patching	Untuk proses yang menggunakan `AWS-RunPatchBaseline`, Anda menentukan apakah Anda ingin operasi untuk menjalankan operasi `Scan` saja, atau operasi `Scan and install`. Jika tujuan Anda adalah mengidentifikasi node terkelola yang tidak sesuai dan tidak memperbaikinya, jalankan hanya operasi. `Scan`	Quick Setupdan Explorer proses, yang menggunakan`AWS-RunPatchBaselineAssociation`, hanya menjalankan `Scan` operasi.
Info selengkapnya	Tentang dokumen SSM `AWS-RunPatchBaseline`	Tentang dokumen SSM `AWS-RunPatchBaselineAssociation`

Untuk informasi tentang berbagai keadaan kepatuhan patch yang mungkin Anda lihat dilaporkan, lihat Memahami nilai keadaan kepatuhan patch

Untuk informasi tentang remediasi node terkelola yang tidak sesuai dengan patch, lihatMenambal node terkelola yang tidak sesuai.

Memahami nilai keadaan kepatuhan patch

Informasi tentang patch untuk node terkelola mencakup laporan status, atau status, dari setiap patch individu.

> **Note**
>
> Jika Anda ingin menetapkan status kepatuhan patch tertentu ke node terkelola, Anda dapat menggunakan perintah put-compliance-items AWS Command Line Interface (AWS CLI) atau operasi PutComplianceItemsAPI. Penetapan keadaan kepatuhan tidak di-support di konsol.

Gunakan informasi dalam tabel berikut untuk membantu Anda mengidentifikasi mengapa node terkelola mungkin tidak sesuai dengan patch.

Nilai kepatuhan tambalan untukDebian Server,Raspberry Pi OS, dan Ubuntu Server

UntukDebian Server,Raspberry Pi OS, danUbuntu Server, aturan untuk klasifikasi paket ke dalam negara kepatuhan yang berbeda dijelaskan dalam tabel berikut.

> **Note**
>
> Ingatlah hal berikut saat Anda mengevaluasi nilai status Terinstal, Diinstal Lainnya, dan Hilang: Jika Anda tidak memilih kotak centang Sertakan pembaruan nonsecurity saat membuat atau memperbarui baseline patch, versi kandidat patch terbatas pada tambalan yang disertakan dalam `trusty-security` (Ubuntu Server14.04 LTS), (Ubuntu Server16.04 LTS), (Ubuntu Server18.04 LTS), (20.04 LTS), `xenial-security` (20.04 LTS), `bionic-security` (Ubuntu Server20.10 STR), `focal-security` (22.04 LTS), atau `groovy-security` Ubuntu Server `jammy-security` Ubuntu Server `debian-security`(Debian ServerdanRaspberry Pi OS). Jika Anda memilih kotak centang Sertakan pembaruan non-keamanan, patch dari repositori lain turut dipertimbangkan.

Keadaan patch	Deskripsi	Status kepatuhan
INSTALLED	Patch terdaftar di baseline patch dan diinstal pada node terkelola. Itu bisa diinstal baik secara manual oleh individu atau secara otomatis Patch Manager ketika `AWS-RunPatchBaseline` dokumen dijalankan pada node yang dikelola.	Patuh
INSTALLED_OTHER	Patch tidak disertakan dalam baseline atau tidak disetujui oleh baseline tetapi diinstal pada node terkelola. Patch mungkin telah diinstal secara manual, paket bisa menjadi ketergantungan yang diperlukan dari patch lain yang disetujui, atau patch mungkin telah disertakan dalam InstallOverrideList operasi. Jika Anda tidak menentukan `Block` sebagai tindakan Patch ditolak, patch `Installed_Other` juga mencakup patch yang terinstal tetapi ditolak.	Patuh
INSTALLED_PENDING_REBOOT	Patch Manager`Install`Operasi menerapkan tambalan ke node yang dikelola, tetapi node belum di-boot ulang sejak tambalan diterapkan. (Perhatikan	Tidak Patuh

Keadaan patch	Deskripsi	Status kepatuhan
	bahwa tambalan yang dipasang di luar tidak pernah diberi status`INSTALLED_PENDING_REBOOT` .) Patch Manager Ini biasanya berarti `NoReboot` opsi dipilih untuk `RebootOption` parameter ketika `AWS-RunPatchBaseline` dokumen terakhir dijalankan pada node terkelola. Untuk informasi selengkapnya, lihat Nama parameter: `RebootOption` .	
`INSTALLED_REJECTED`	Patch diinstal pada node terkelola tetapi ditentukan dalam daftar tambalan Ditolak. Ini biasanya berarti patch telah diinstal sebelum ditambahkan ke daftar patch yang ditolak.	Tidak Patuh
`MISSING`	Paket yang di-filter melalui baseline dan belum terinstal.	Tidak Patuh
`FAILED`	Paket yang gagal untuk diinstal selama operasi patch.	Tidak Patuh

Nilai kepatuhan patch untuk sistem operasi lain

Untuk semua sistem operasi selain Debian ServerRaspberry Pi OS,, danUbuntu Server, aturan untuk klasifikasi paket ke dalam status kepatuhan yang berbeda dijelaskan dalam tabel berikut.

Keadaan patch	Deskripsi	Nilai kepatuhan
`INSTALLED`	Patch terdaftar di baseline patch dan diinstal pada node	Patuh

Keadaan patch	Deskripsi	Nilai kepatuhan
	terkelola. Itu bisa diinstal baik secara manual oleh individu atau secara otomatis Patch Manager ketika `AWS-RunPatchBaseline` dokumen dijalankan pada node.	
`INSTALLED_OTHER` [1]	Patch tidak ada di baseline, tetapi diinstal pada node terkelola. Patch mungkin telah diinstal secara manual, atau paket dapat merupakan dependensi yang diperlukan dari patch lain yang disetujui . Jika Anda tidak menentukan `Block` sebagai tindakan Patch ditolak, patch `Installed_Other` juga mencakup patch yang terinstal tetapi ditolak.	Patuh
`INSTALLED_REJECTED`	Patch diinstal pada node terkelola tetapi ditentukan dalam daftar tambalan yang ditolak. Ini biasanya berarti patch telah diinstal sebelum ditambahkan ke daftar patch yang ditolak.	Tidak Patuh

Keadaan patch	Deskripsi	Nilai kepatuhan
INSTALLED_PENDING_REBOOT	Patch ManagerInstallOperasi menerapkan tambalan ke node terkelola (atau tambalan diterapkan ke node Windows Server terkelola di luarPatch Manager), tetapi node belum di-boot ulang sejak tambalan diterapkan. (Perhatikan bahwa tambalan yang dipasang di luar tidak pernah diberi statusINSTALLED_PENDING_REBOOT .) Patch Manager Ini biasanya berarti NoReboot opsi dipilih untuk RebootOption parameter ketika AWS-RunPatchBaseline dokumen terakhir dijalankan pada node terkelola. Untuk informasi selengkapnya, lihat Nama parameter: RebootOption .	Tidak Patuh
MISSING	Patch disetujui di baseline, tetapi tidak diinstal pada node terkelola. Jika Anda mengonfigurasi tugas dokumen AWS-RunPatchBaseline untuk memindai (bukan menginstal), sistem melaporkan status ini untuk patch yang diletakkan selama pemindaian tetapi belum diinstal.	Tidak Patuh

Keadaan patch	Deskripsi	Nilai kepatuhan
NOT_APPLICABLE [1]	Patch disetujui di baseline, tetapi layanan atau fitur yang menggunakan tambalan tidak diinstal pada node terkelola . Misalnya, patch untuk layanan server web seperti Internet Information Services (IIS) akan menunjukkan `NOT_APPLICABLE` apakah itu disetujui di baseline, tetapi layanan web tidak diinstal pada node terkelola. Sebuah patch juga dapat ditandai `NOT_APPLICABLE` jika telah digantikan oleh pembaruan berikutnya. Ini berarti bahwa pembaruan yang setelahny a diinstal dan pembaruan `NOT_APPLICABLE` tidak lagi diperlukan. **Note** Keadaan kepatuhan ini hanya dilaporkan pada sistem operasi Windows Server.	Tidak berlaku

Keadaan patch	Deskripsi	Nilai kepatuhan
`FAILED`	patch disetujui dalam baseline, tapi tidak dapat diinstal. Untuk memecahkan masalah situasi ini, tinjau output perintah untuk informasi yang mungkin membantu Anda memahami masalah.	Tidak Patuh

[1] Untuk tambalan dengan status INSTALLED_OTHER dan NOT_APPLICABLE, Patch Manager menghilangkan beberapa data dari hasil kueri berdasarkan describe-instance-patchesperintah, seperti nilai untuk dan. `Classification` `Severity` Hal ini dilakukan untuk membantu mencegah melebihi batas data untuk node individu di Inventory, kemampuan. AWS Systems Manager Untuk melihat semua detail tambalan, Anda dapat menggunakan describe-available-patchesperintah.

Menambal node terkelola yang tidak sesuai

Banyak yangAWS Systems Manager alat dan proses yang sama yang dapat Anda gunakan untuk memeriksa node terkelola untuk kepatuhan patch dapat digunakan untuk membawa kepatuhan dengan aturan patch yang saat ini berlaku untuk mereka. Untuk membawa node yang dikelola ke kepatuhan patchPatch Manager, suatu kemampuanAWS Systems Manager, harus menjalankan`Scan and install` operasi. (Jika tujuan Anda adalah untuk mengidentifikasi node terkelola yang tidak patuh dan bukan memperbaikinya, jalankan`Scan` operasi sebagai gantinya. Untuk informasi selengkapnya, lihatMengidentifikasi node terkelola yang tidak sesuai.)

Instal patch menggunakan Systems Manager

Anda dapat memilih dari beberapa alat untuk menjalankan operasi `Scan and install`:

- (Disarankan) Mengkonfigurasi kebijakan tambalanQuick Setup, kemampuan Systems Manager, yang memungkinkan Anda menginstal patch yang hilang pada jadwal untuk seluruh organisasi, subset unit organisasi, atau satuAkun AWS. Untuk informasi selengkapnya, lihat Patch Managerkonfigurasi penambalan organisasi.
- Buat jendela pemeliharaan yang menggunakan dokumen Systems Manager (dokumen SSM)`AWS-RunPatchBaseline` dalam jenisRun Command tugas. Untuk informasi, lihat Walkthrough: Membuat jendela pemeliharaan untuk patching (konsol).

- Jalankan secara manualAWS-RunPatchBaseline dalamRun Command operasi. Untuk informasi, lihat Menjalankan perintah dari konsol.

- Instal patch sesuai permintaan menggunakan opsi Patch sekarang. Untuk informasi, lihat Menambal node terkelola sesuai permintaan.

Menghindari penimpaan data kepatuhan patch yang tidak disengaja

Jika Anda memiliki beberapa jenis operasi untuk memindai instans Anda untuk kepatuhan patch, setiap pemindaian menimpa data kepatuhan patch dari pemindaian sebelumnya. Akibatnya, Anda mungkin berakhir dengan hasil yang tidak terduga dalam data kepatuhan patch Anda.

Misalnya, Anda membuat kebijakan tambalan yang memindai kepatuhan patch setiap hari pada pukul 2 pagi waktu setempat. Kebijakan tambalan itu menggunakan garis dasar tambalan yang menargetkan tambalan dengan tingkat keparahan yang ditandai sebagaiCritical,, Important dan. Moderate Garis dasar tambalan ini juga menentukan beberapa tambalan yang ditolak secara khusus.

Juga misalkan Anda sudah memiliki jendela pemeliharaan yang disiapkan untuk memindai kumpulan node terkelola yang sama setiap hari pada pukul 4 pagi waktu setempat, yang tidak Anda hapus atau nonaktifkan. Tugas jendela pemeliharaan itu menggunakan baseline tambalan yang berbeda, tugas yang hanya menargetkan tambalan dengan Critical tingkat keparahan dan tidak mengecualikan tambalan tertentu.

Ketika pemindaian kedua ini dilakukan oleh jendela pemeliharaan, data kepatuhan patch dari pemindaian pertama dihapus dan diganti dengan kepatuhan patch dari pemindaian kedua.

Oleh karena itu, kami sangat menyarankan hanya menggunakan satu metode otomatis untuk memindai dan menginstal dalam operasi penambalan Anda. Jika Anda menyiapkan kebijakan tambalan, Anda harus menghapus atau menonaktifkan metode pemindaian lain untuk kepatuhan tambalan. Untuk informasi selengkapnya, lihat topik berikut:

- Untuk menghapus tugas operasi tambalan dari jendela pemeliharaan - Memperbarui atau membatalkan pendaftaran tugas jendela pemeliharaan (konsol)
- Untuk menghapus State Manager asosiasi —Menghapus asosiasi.

Untuk menonaktifkan pemindaian kepatuhan patch harian dalam konfigurasi Manajemen Host, lakukan hal berikut di: Quick Setup

1. Di panel navigasi, pilih Quick Setup.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Quick Setupdi panel navigasi.
2. Pilih konfigurasi Manajemen Host untuk diperbarui.
3. Pilih Tindakan, Edit konfigurasi.
4. Kosongkan kotak centang instance Pindai untuk patch yang hilang setiap hari.
5. Pilih Perbarui.

> **Note**
>
> Menggunakan opsi Patch now untuk memindai node terkelola untuk kepatuhan juga menghasilkan penimpaan data kepatuhan patch.

Menambal node terkelola sesuai permintaan

Dengan menggunakan opsi Patch nowPatch Manager, kemampuanAWS Systems Manager, Anda dapat menjalankan operasi patching sesuai permintaan dari konsol Systems Manager. Ini berarti Anda tidak perlu membuat jadwal untuk memperbarui status kepatuhan node terkelola Anda atau untuk menginstal tambalan pada node yang tidak sesuai. Anda juga tidak perlu mengganti konsol Systems Manager antara Patch Manager danMaintenance Windows, kemampuanAWS Systems Manager, untuk mengatur atau memodifikasi jendela patching terjadwal.

Patch sekarang sangat berguna ketika Anda harus menerapkan pembaruan zero-day atau menginstal patch penting lainnya pada node terkelola Anda sesegera mungkin.

> **Note**
>
> Penambalan sesuai permintaan didukung untuk satu Akun AWS Wilayah AWS pasangan sekaligus. Itu tidak dapat digunakan dengan operasi penambalan yang didasarkan pada kebijakan tambalan. Sebaiknya gunakan kebijakan tambalan untuk menjaga agar semua node terkelola tetap sesuai. Untuk informasi selengkapnya tentang bekerja dengan kebijakan tambalan, lihatMenggunakan kebijakan Quick Setup tambalan.

Topik

Cara kerja 'Patch sekarang'

Untuk menjalankan Patch sekarang, Anda hanya menentukan dua pengaturan yang diperlukan:

- Apakah akan memindai patch yang hilang saja, atau untuk memindai dan menginstal tambalan pada node terkelola Anda
- Node yang mengelola untuk menjalankan operasi

Ketika operasi Patch sekarang berjalan, ini menentukan baseline patch mana yang akan digunakan dengan cara yang sama yang dipilih untuk operasi patching lainnya. Jika node terkelola dikaitkan dengan grup tambalan, baseline patch yang ditentukan untuk grup tersebut akan digunakan. Jika node terkelola tidak terkait dengan grup patch, operasi menggunakan baseline patch yang saat ini ditetapkan sebagai default untuk jenis sistem operasi dari node terkelola. Ini bisa berupa baseline yang telah ditentukan sebelumnya, atau baseline kustom yang telah Anda tetapkan sebagai default. Untuk informasi selengkapnya tentang pemilihan dasar tambalan, lihat. Tentang grup patch

Opsi yang dapat Anda tentukan untuk Patch sekarang termasuk memilih kapan, atau apakah, untuk me-reboot node terkelola setelah menambal, menentukan bucket Amazon Simple Storage Service (Amazon S3) untuk menyimpan data log untuk operasi penambalan, dan menjalankan dokumen Systems Manager (dokumen SSM) sebagai kait siklus hidup selama penambalan.

Ambang batas konkurensi dan kesalahan untuk 'Patch sekarang'

Untuk operasi Patch now, opsi konkurensi dan ambang kesalahan ditangani oleh. Patch Manager Anda tidak perlu menentukan berapa banyak node terkelola untuk ditambal sekaligus, atau berapa banyak kesalahan yang diizinkan sebelum operasi gagal. Patch Managermenerapkan pengaturan ambang konkurensi dan kesalahan yang dijelaskan dalam tabel berikut saat Anda menambal sesuai permintaan.

> **Important**
>
> Ambang batas berikut hanya berlaku untuk `Scan and install` operasi. Untuk `Scan` operasi, Patch Manager mencoba memindai hingga 1.000 node secara bersamaan, dan lanjutkan pemindaian hingga mengalami hingga 1.000 kesalahan.

Konkurensi: Instal operasi

Jumlah total node terkelola dalam operasi Patch sekarang	Jumlah node terkelola yang dipindai atau ditambal pada suatu waktu
Kurang dari 25	1
25-100	5%
101 hingga 1.000	8%
Lebih dari 1.000	10%

Ambang kesalahan: Instal operasi

Jumlah total node terkelola dalam operasi Patch sekarang	Jumlah kesalahan yang diizinkan sebelum operasi gagal
Kurang dari 25	1
25-100	5
101 hingga 1.000	10
Lebih dari 1.000	10

Menggunakan kait siklus hidup 'Patch sekarang'

Patch sekarang memberi Anda kemampuan untuk menjalankan dokumen Perintah SSM sebagai kait siklus hidup selama operasi penambalan. `Install` Anda dapat menggunakan kait ini untuk tugas-tugas seperti mematikan aplikasi sebelum menambal atau menjalankan pemeriksaan kesehatan pada aplikasi Anda setelah menambal atau setelah reboot.

Untuk informasi selengkapnya tentang penggunaan kait siklus hidup, lihat. [Tentang dokumen SSM AWS-RunPatchBaselineWithHooks]

Tabel berikut mencantumkan kait siklus hidup yang tersedia untuk masing-masing dari tiga opsi Patch now reboot, selain penggunaan sampel untuk setiap hook.

Kait siklus hidup dan penggunaan sampel

Opsi reboot	Hook: Sebelum instalasi	Hook: Setelah instalasi	Hook: Saat keluar	Hook: Setelah dijadwalkan reboot
Reboot jika diperlukan	Jalankan dokumen SSM sebelum penambalan dimulai. Contoh penggunaan: Matikan aplikasi dengan aman sebelum proses patching dimulai.	Jalankan dokumen SSM di akhir operasi patching dan sebelum reboot node terkelola. Contoh penggunaan: Jalankan operasi seperti menginstal aplikasi pihak ketiga sebelum reboot potensial.	Jalankan dokumen SSM setelah operasi patching selesai dan instance di-reboot. Contoh penggunaan: Pastikan aplikasi berjalan seperti yang diharapkan setelah menambal.	Tidak tersedia
Jangan reboot instance saya	Sama seperti di atas.	Jalankan dokumen SSM di akhir operasi patching. Contoh penggunaan: Pastikan aplikasi berjalan seperti yang diharapka	Tidak tersedia	Tidak tersedia

Opsi reboot	Hook: Sebelum instalasi	Hook: Setelah instalasi	Hook: Saat keluar	Hook: Setelah dijadwalkan reboot
		n setelah menambal.		
Jadwalkan waktu reboot	Sama seperti di atas.	Sama seperti untuk Jangan reboot instance saya.	Tidak tersedia	Jalankan dokumen SSM segera setelah reboot terjadwal selesai. Contoh penggunaan: Pastikan aplikasi berjalan seperti yang diharapkan setelah reboot.

Menjalankan 'Patch sekarang'

Gunakan prosedur berikut untuk menambal node terkelola Anda sesuai permintaan.

Untuk menjalankan 'Patch sekarang'

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Patch Manager.
3. Pada AWS Systems ManagerPatch Managerhalaman atau halaman dasar Patch, tergantung pada mana yang terbuka, pilih Patch sekarang.
4. Untuk Operasi patching, pilih salah satu hal berikut:
   - Scan: Patch Manager menemukan patch mana yang hilang dari node terkelola Anda tetapi tidak menginstalnya. Anda dapat melihat hasil di dasbor Kepatuhan atau alat lain yang Anda gunakan untuk melihat kepatuhan patch.
   - Pindai dan instal: Patch Manager menemukan tambalan mana yang hilang dari node terkelola Anda dan menginstalnya.

5. Gunakan langkah ini hanya jika Anda memilih Pindai dan instal pada langkah sebelumnya. Untuk Opsi reboot, pilih salah satu hal berikut:

   - Reboot jika diperlukan: Setelah instalasi, Patch Manager reboot node terkelola hanya jika diperlukan untuk menyelesaikan instalasi patch.
   - Jangan me-reboot instance saya: Setelah instalasi, Patch Manager tidak me-reboot node yang dikelola. Anda dapat me-reboot node secara manual ketika Anda memilih atau mengelola reboot di luar. Patch Manager
   - Jadwalkan waktu reboot: Tentukan tanggal, waktu, dan zona waktu UTC Patch Manager untuk me-reboot node terkelola Anda. Setelah Anda menjalankan operasi Patch now, reboot terjadwal terdaftar sebagai asosiasi State Manager dengan nama`AWS-PatchRebootAssociation`.

6. Untuk Instans untuk di-patch, pilih salah satu hal berikut:

   - Patch semua instance: Patch Manager menjalankan operasi yang ditentukan pada semua node terkelola Akun AWS di Anda saat iniWilayah AWS.
   - Hanya menambal instance target yang saya tentukan: Anda menentukan node terkelola mana yang akan ditargetkan pada langkah berikutnya.

7. Gunakan langkah ini hanya jika Anda memilih Patch instans target yang saya tentukan saja di langkah sebelumnya. Di bagian Pemilihan target, identifikasi node tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih node secara manual, atau menentukan grup sumber daya.

   > **Note**
   >
   > Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.
   > Jika Anda memilih untuk menargetkan resource group, perhatikan bahwa resource group yang didasarkan pada tumpukan AWS CloudFormation harus tetap ditandai dengan tag `aws:cloudformation:`*`stack-id`* default. Jika telah dihapus, Patch Manager mungkin tidak dapat menentukan node terkelola mana yang termasuk dalam grup sumber daya.

8. (Opsional) Untuk Penyimpanan log patching, jika Anda ingin membuat dan menyimpan log dari operasi patching ini, pilih bucket S3 untuk menyimpan log.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan milik pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbedaAkun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

9. (Opsional) Jika Anda ingin menjalankan dokumen SSM sebagai kait siklus hidup selama titik tertentu dari operasi patching, lakukan hal berikut:

   - Pilih Gunakan kait siklus hidup.
   - Untuk setiap kait yang tersedia, pilih dokumen SSM untuk dijalankan pada titik operasi yang ditentukan:
     - Sebelum instalasi
     - Setelah instalasi
     - Saat keluar
     - Setelah reboot terjadwal

   > **Note**
   >
   > Dokumen default, `AWS-Noop`, tidak menjalankan operasi apa pun.

10. Pilih Patch sekarang.

    Halaman Ringkasan eksekusi asosiasi terbuka. (Patch sekarang menggunakan asosiasi diState Manager, kemampuanAWS Systems Manager, untuk operasinya.) Di area ringkasan Operasi, Anda dapat memantau status pemindaian atau penambalan pada node terkelola yang Anda tentukan.

Bekerja dengan dasar patch

Suatu dasar patch mana yang disetujui untuk instalasi pada node Anda.Patch ManagerAWS Systems Manager Anda dapat menentukan patch yang disetujui atau ditolak satu per satu. Anda juga dapat membuat aturan persetujuan otomatis untuk menentukan bahwa jenis pembaruan tertentu (misalnya, pembaruan penting) harus disetujui secara otomatis. Daftar yang ditolak menggantikan aturan dan daftar persetujuan. Untuk menggunakan daftar patch yang disetujui untuk menginstal paket tertentu, pertama-tama Anda menghapus semua aturan persetujuan otomatis. Jika Anda secara eksplisit mengidentifikasi patch sebagai ditolak, patch tidak akan disetujui atau diinstal, bahkan jika cocok dengan semua kriteria dalam aturan persetujuan otomatis. Selain itu, sebuah patch hanya diinstal pada sebuah dikelola Node jika itu berlaku untuk perangkat lunak pada node tersebut, bahkan jika patch telah disetujui untuk node yang dikelola.

Topik

- Melihat dasar patch yang telah ditetapkan AWS
- Bekerja dengan dasar patch kustom
- Mengatur dasar patch yang ada sebagai default

Info lebih lanjut

- Tentang dasar patch

Melihat dasar patch yang telah ditetapkan AWS

Patch Manager, kemampuanAWS Systems Manager, termasuk baseline patch yang telah ditentukan untuk setiap sistem operasi yang didukung oleh. Patch Manager Anda dapat menggunakan dasar patch ini (Anda tidak dapat menyesuaikannya), atau Anda dapat membuat milik Anda sendiri. Prosedur berikut ini menjelaskan cara melihat dasar patch yang telah ditetapkan untuk melihat apakah memenuhi kebutuhan Anda. Untuk mempelajari selengkapnya tentang dasar patch, lihat Tentang dasar patch yang telah ditetapkan dan kustom.

Untuk melihat dasar patch yang telah ditetapkan AWS

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Patch Manager.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.

3. Dalam daftar dasar patch, pilih ID baseline dari salah satu dasar patch yang telah ditetapkan.

   -atau-

   Jika Anda mengakses Patch Manager untuk pertama kalinya saat iniWilayah AWS, pilih Mulai dengan ikhtisar, pilih tab Garis dasar Patch, lalu pilih ID dasar dari salah satu garis dasar tambalan yang telah ditentukan.

   > **Note**
   >
   > Untuk Windows Server, disediakan tiga dasar patch yang telah ditetapkan. Garis dasar patch `AWS-DefaultPatchBaseline` dan hanya `AWS-WindowsPredefinedPatchBaseline-OS` mendukung pembaruan sistem operasi pada sistem operasi Windows itu sendiri. `AWS-DefaultPatchBaseline`digunakan sebagai baseline patch default untuk node Windows Server terkelola kecuali Anda menentukan baseline patch yang berbeda. Pengaturan konfigurasi di dua dasar patch ini sama. Yang lebih baru dari keduanya, `AWS-WindowsPredefinedPatchBaseline-OS`, dibuat untuk membedakannya dari dasar patch ketiga yang telah ditetapkan untuk Windows Server. Dasar patch tersebut, `AWS-WindowsPredefinedPatchBaseline-OS-Applications`, dapat digunakan untuk menerapkan patch ke ke sistem operasi Windows Server dan aplikasi yang didukung yang dirilis oleh Microsoft.
   > Untuk informasi selengkapnya, lihat Mengatur dasar patch yang ada sebagai default.

4. Di bagian Aturan persetujuan, tinjau konfigurasi baseline patch.

5. Jika konfigurasi dapat diterima untuk node terkelola Anda, Anda dapat langsung melanjutkan ke prosedurBekerja dengan kelompok patch.

   -atau-

   Untuk membuat dasar patch default Anda sendiri, lanjutkan ke topik Bekerja dengan dasar patch kustom.

Bekerja dengan dasar patch kustom

Patch Manager, kemampuanAWS Systems Manager, termasuk baseline patch yang telah ditentukan untuk setiap sistem operasi yang didukung oleh. Patch Manager Anda dapat menggunakan dasar patch ini (Anda tidak dapat menyesuaikannya), atau Anda dapat membuat milik Anda sendiri.

Prosedur berikut ini menjelaskan cara membuat, memperbarui, dan menghapus dasar patch kustom Anda sendiri. Untuk mempelajari selengkapnya tentang dasar patch, lihat Tentang dasar patch yang telah ditetapkan dan kustom.

Topik

- Membuat dasar patch kustom (Linux)
- Membuat dasar patch kustom (macOS)
- Membuat dasar patch kustom (Windows)
- Memperbarui atau menghapus baseline patch kustom

Membuat dasar patch kustom (Linux)

Gunakan prosedur berikut untuk membuat baseline patch kustom untuk node terkelola Linux diPatch Manager, kemampuan. AWS Systems Manager

Untuk informasi tentang membuat baseline patch untuk node macOS terkelola, lihat. Membuat dasar patch kustom (macOS) Untuk informasi tentang membuat baseline patch untuk node terkelola Windows, lihat. Membuat dasar patch kustom (Windows)

Untuk membuat baseline patch kustom untuk node terkelola Linux

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Patch Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.
3. Pilih tab Patch baseline, lalu pilih Create patch baseline.

   -atau-

Jika Anda mengakses Patch Manager untuk pertama kalinya saat ini Wilayah AWS, pilih Mulai dengan ikhtisar, pilih tab Garis dasar Patch, lalu pilih Buat baseline patch.

4. Untuk Nama, masukkan nama untuk dasar patch baru Anda, misalnya, `MyRHELPatchBaseline`.
5. (Opsional) Untuk Deskripsi, masukkan deskripsi untuk dasar patch ini.
6. Untuk Sistem operasi, pilih suatu sistem operasi, misalnya, `Red Hat Enterprise Linux`.
7. Jika Anda ingin mulai menggunakan dasar patch ini sebagai default untuk sistem operasi yang dipilih segera setelah Anda membuatnya, centang kotak di sebelah Atur dasar patch ini sebagai dasar patch default untuk instans ***`Nama sistem operasi`***.

> **Note**
>
> Opsi ini hanya tersedia jika Anda pertama kali mengakses Patch Manager sebelum kebijakan tambalan dirilis pada 22 Desember 2022.
> Untuk informasi tentang mengatur dasar patch yang ada sebagai default, lihat Mengatur dasar patch yang ada sebagai default.

8. Di bagian Aturan persetujuan untuk sistem operasi, gunakan bidang tersebut untuk membuat satu atau lebih aturan persetujuan otomatis.

   - Produk: Versi sistem operasi yang berlaku untuk aturan persetujuan, seperti`RedhatEnterpriseLinux7.4`. Pilihan default adalah `All`.
   - Klasifikasi: Jenis patch yang diberlakukan aturan persetujuan, seperti `Security` atau `Enhancement`. Pilihan default adalah `All`.

> **Tip**
>
> Anda dapat mengonfigurasi baseline patch untuk mengontrol apakah upgrade versi minor untuk Linux diinstal, seperti 7.8. RHEL Upgrade versi minor dapat diinstal secara otomatis dengan Patch Manager asalkan pembaruan tersedia di repositori yang sesuai.
> Untuk sistem operasi Linux, pemutakhiran versi minor tidak diklasifikasikan secara konsisten. Mereka dapat diklasifikasikan sebagai perbaikan bug atau pembaruan keamanan, atau tidak diklasifikasikan, bahkan dalam versi kernel yang sama. Berikut ini adalah beberapa pilihan untuk mengendalikan apakah dasar patch menginstalnya.

- Opsi 1: Aturan persetujuan terluas untuk memastikan pemutakhiran versi minor diinstal ketika tersedia adalah dengan menentukan Klasifikasi sebagai `All` (*) dan pilih opsi Sertakan pembaruan non-keamanan.
- Opsi 2: Untuk memastikan patch untuk versi sistem operasi diinstal, Anda dapat menggunakan wildcard (*) untuk menentukan format kernel di bagian Pengecualian patch dari baseline. Sebagai contoh, format kernel untuk RHEL 7.* adalah `kernel-3.10.0-*.el7.x86_64`.

  Masukkan `kernel-3.10.0-*.el7.x86_64` daftar Patch yang disetujui di baseline tambalan Anda untuk memastikan semua tambalan, termasuk peningkatan versi minor, diterapkan ke node terkelola 7.* Anda. RHEL (Jika Anda tahu persis nama paket dari patch versi minor, Anda dapat memasukkan itu sebagai gantinya.)
- Opsi 3: Anda dapat memiliki kontrol paling besar atas tambalan mana yang diterapkan ke node terkelola Anda, termasuk peningkatan versi minor, dengan menggunakan InstallOverrideListparameter dalam dokumen. `AWS-RunPatchBaseline` Untuk informasi selengkapnya, lihat Tentang dokumen SSM `AWS-RunPatchBaseline`.

- Kepelikan: Nilai kepelikan patch yang diberlakukan aturan, seperti `Critical`. Pilihan default adalah `All`.
- Persetujuan otomatis: Metode untuk memilih patch untuk persetujuan otomatis.

> **Note**
>
> Karena tidak mungkin menentukan tanggal rilis paket pembaruan secara andalUbuntu Server, opsi persetujuan otomatis tidak didukung untuk sistem operasi ini.

- Menyetujui patch setelah beberapa hari tertentu: Jumlah hari Patch Manager untuk menunggu setelah patch dirilis atau terakhir diperbarui sebelum patch secara otomatis disetujui. Anda dapat memasukkan bilangan bulat apa saja dari nol (0) sampai 360. Untuk sebagian besar skenario, kami merekomendasikan untuk menunggu tidak lebih dari 100 hari.
- Menyetujui patch yang dirilis hingga tanggal tertentu: Tanggal rilis patch yang Patch Manager secara otomatis menerapkan semua patch yang dirilis atau diperbarui pada atau sebelum tanggal tersebut. Misalnya, jika Anda menentukan 7 Juli 2023, tidak ada tambalan

yang dirilis atau terakhir diperbarui pada atau setelah 8 Juli 2023, yang diinstal secara otomatis.

- (Opsional) Pelaporan kepatuhan: Tingkat keparahan yang ingin Anda tetapkan ke tambalan yang disetujui oleh baseline, seperti atau. `Critical` `High`

> **Note**
>
> Jika Anda menentukan tingkat pelaporan kepatuhan dan status patch dari setiap patch yang disetujui dilaporkan sebagai`Missing`, maka tingkat keparahan kepatuhan yang dilaporkan secara keseluruhan baseline patch adalah tingkat keparahan yang Anda tentukan.

- Sertakan pembaruan non-keamanan: Pilih kotak centang untuk menginstal patch sistem operasi Linux non-keamanan yang tersedia di repositori sumber, selain patch terkait keamanan.

> **Note**
>
> UntukSUSE Linux Enterprise Server, (SLES) tidak perlu memilih kotak centang karena tambalan untuk masalah keamanan dan nonkeamanan diinstal secara default pada node SLES terkelola. Untuk informasi selengkapnya, lihat konten SLES dalam Cara pemilihan patch keamanan.

Untuk informasi selengkapnya tentang bekerja dengan aturan persetujuan di dasar patch kustom, lihat Tentang baseline kustom.

9. Jika Anda ingin secara eksplisit menyetujui patch lain selain yang memenuhi aturan persetujuan Anda, lakukan hal berikut di bagian Pengecualian patch:

- Untuk Patch yang disetujui, masukkan daftar patch yang dipisahkan koma yang ingin Anda setujui.

> **Note**
>
> Untuk informasi tentang format yang diterima untuk daftar patch yang disetujui dan patch yang ditolak, lihat Tentang format nama paket untuk daftar patch yang disetujui dan ditolak.

- (Opsional) Untuk Tingkat kepatuhan patch yang disetujui, tetapkan tingkat kepatuhan pada patch dalam daftar.
- Jika ada patch yang disetujui yang Anda tentukan tidak terkait dengan keamanan, pilih kotak centang Sertakan pembaruan non-keamanan untuk tambalan ini yang akan diinstal pada sistem operasi Linux Anda juga.

10. Jika Anda ingin secara eksplisit menolak patch lain selain yang memenuhi aturan persetujuan Anda, lakukan hal berikut di bagian Pengecualian patch:

    - Untuk Patch yang ditolak, masukkan daftar patch yang dipisahkan koma yang ingin Anda tolak.

      > **Note**
      >
      > Untuk informasi tentang format yang diterima untuk daftar patch yang disetujui dan patch yang ditolak, lihat Tentang format nama paket untuk daftar patch yang disetujui dan ditolak.

    - Untuk tindakan tambalan Ditolak, pilih tindakan yang akan diambil pada tambalan yang disertakan dalam daftar tambalan Ditolak. Patch Manager
      - Diizinkan sebagai dependensi: Sebuah paket di daftar Patch yang ditolak diinstal hanya jika merupakan dependensi dari paket lain. Ini dianggap sesuai dengan baseline patch dan statusnya dilaporkan sebagai. InstalledOther Ini adalah tindakan default jika tidak ada pilihan yang ditentukan.
      - Blokir: Paket dalam daftar tambalan Ditolak, dan paket yang menyertakannya sebagai dependensi, tidak diinstal dalam Patch Manager keadaan apa pun. Jika sebuah paket diinstal sebelum ditambahkan ke daftar tambalan Ditolak, atau diinstal di luar Patch Manager sesudahnya, paket tersebut dianggap tidak sesuai dengan garis dasar tambalan dan statusnya dilaporkan sebagai. InstalledRejected

11. (Opsional) Jika Anda ingin menentukan repositori patch alternatif untuk versi sistem operasi yang berbeda, seperti AmazonLinux2016.03 dan AmazonLinux2017.09, lakukan hal berikut untuk setiap produk di bagian Sumber Patch:

    - Di Nama, masukkan nama untuk membantu Anda mengidentifikasi konfigurasi sumber.
    - Di Produk, pilih versi sistem operasi yang digunakan untuk repositori sumber patch, seperti `RedhatEnterpriseLinux7.4`.

- Di Konfigurasi, masukkan nilai konfigurasi repositori yum yang akan digunakan dalam format berikut:

```
[main]
name=MyCustomRepository
baseurl=https://my-custom-repository
enabled=1
```

> **Tip**
>
> Untuk informasi tentang opsi lain yang tersedia untuk konfigurasi repositori yum Anda, lihat dnf.conf(5).

Pilih Tambah sumber lain untuk menentukan repositori sumber untuk setiap versi sistem operasi tambahan, hingga maksimum 20.

Untuk informasi selengkapnya tentang repositori patch sumber alternatif, lihat Cara menentukan repositori sumber patch alternatif (Linux).

12. (Opsional) Untuk Kelola tag, terapkan satu atau lebih pasangan nama/nilai kunci tag ke dasar patch.

    Tag adalah metadata opsional yang Anda tetapkan ke sumber daya. Tag memungkinkan Anda untuk mengkategorikan sumber daya dengan berbagai cara, seperti berdasarkan tujuan, pemilik, atau lingkungan. Sebagai contoh, Anda mungkin ingin menandai dasar patch untuk mengidentifikasi tingkat kepelikan patch yang ditentukannya, keluarga sistem operasi tempat patch diterapkan, dan jenis lingkungan. Dalam kasus ini, Anda dapat menentukan tag yang serupa dengan pasangan nama/nilai kunci berikut:

    - `Key=PatchSeverity,Value=Critical`
    - `Key=OS,Value=RHEL`
    - `Key=Environment,Value=Production`

13. Pilih Buat dasar patch.

Membuat dasar patch kustom (macOS)

Gunakan prosedur berikut untuk membuat baseline patch kustom untuk node macOS terkelola diPatch Manager, kemampuan. AWS Systems Manager

Untuk informasi tentang membuat baseline patch untuk node Windows Server terkelola, lihat. Membuat dasar patch kustom (Windows) Untuk informasi tentang membuat baseline patch untuk node terkelola Linux, lihat. Membuat dasar patch kustom (Linux)

> **Note**
>
> macOStidak didukung sama sekali Wilayah AWS. Untuk informasi selengkapnya tentang dukungan Amazon EC2macOS, lihat instans Amazon EC2 Mac di Panduan Pengguna Amazon EC2 untuk Instans Linux.

Untuk membuat baseline patch khusus untuk macOS node terkelola

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Patch Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.
3. Pilih tab Patch baseline, lalu pilih Create patch baseline.

   -atau-

   Jika Anda mengakses Patch Manager untuk pertama kalinya saat ini Wilayah AWS, pilih Mulai dengan ikhtisar, pilih tab Garis dasar Patch, lalu pilih Buat baseline patch.
4. Untuk Nama, masukkan nama untuk dasar patch baru Anda, misalnya, `MymacOSPatchBaseline`.
5. (Opsional) Untuk Deskripsi, masukkan deskripsi untuk dasar patch ini.
6. Untuk Sistem operasi, pilih macOS.

7. Jika Anda ingin mulai menggunakan dasar patch ini sebagai default untuk macOS segera setelah Anda membuatnya, centang kotak di sebelah Atur dasar patch ini sebagai dasar patch default untuk instans macOS.

   > **Note**
   >
   > Opsi ini hanya tersedia jika Anda pertama kali mengakses Patch Manager sebelum kebijakan tambalan dirilis pada 22 Desember 2022.
   > Untuk informasi tentang mengatur dasar patch yang ada sebagai default, lihat Mengatur dasar patch yang ada sebagai default.

8. Di bagian Aturan persetujuan untuk sistem operasi, gunakan bidang tersebut untuk membuat satu atau lebih aturan persetujuan otomatis.

   - Produk: Versi sistem operasi yang berlaku untuk aturan persetujuan, seperti `Mojave10.14.1` atau`Catalina10.15.1`. Pilihan default adalah `All`.

     > **Note**
     >
     > Sistem manajemen paket perangkat lunak sumber terbuka Homebrew telah menghentikan dukungan untuk macOS 10.14.x (Mojave) dan 10.15.x (Catalina). Akibatnya, operasi penambalan pada versi ini saat ini tidak didukung.

   - Klasifikasi: Pengelola paket atau beberapa pengelola paket yang Anda ingin terapkan paket selama proses patching. Anda memilih dari yang berikut ini:
     - softwareupdate
     - installer
     - brew
     - brew cask

     Pilihan default adalah `All`.

   - (Opsional) Pelaporan kepatuhan: Tingkat keparahan yang ingin Anda tetapkan ke tambalan yang disetujui oleh baseline, seperti atau. `Critical` `High`

> **Note**
>
> Jika Anda menentukan tingkat pelaporan kepatuhan dan status patch dari setiap patch yang disetujui dilaporkan sebagai`Missing`, maka tingkat keparahan kepatuhan yang dilaporkan secara keseluruhan baseline patch adalah tingkat keparahan yang Anda tentukan.

- Sertakan pembaruan non-keamanan: Pilih kotak centang untuk menginstal patch sistem operasi non-keamanan yang tersedia di repositori sumber, selain patch terkait keamanan.

Untuk informasi selengkapnya tentang bekerja dengan aturan persetujuan di dasar patch kustom, lihat Tentang baseline kustom.

9. Jika Anda ingin secara eksplisit menyetujui patch lain selain yang memenuhi aturan persetujuan Anda, lakukan hal berikut di bagian Pengecualian patch:

   - Untuk Patch yang disetujui, masukkan daftar patch yang dipisahkan koma yang ingin Anda setujui.

   > **Note**
   >
   > Untuk informasi tentang format yang diterima untuk daftar patch yang disetujui dan patch yang ditolak, lihat Tentang format nama paket untuk daftar patch yang disetujui dan ditolak.

   - (Opsional) Untuk Tingkat kepatuhan patch yang disetujui, tetapkan tingkat kepatuhan pada patch dalam daftar.
   - Jika patch disetujui yang Anda tentukan tidak terkait dengan keamanan, pilih kotak centang Sertakan pembaruan non-keamanan untuk tambalan ini yang akan diinstal pada sistem macOS operasi Anda juga.

10. Jika Anda ingin secara eksplisit menolak patch lain selain yang memenuhi aturan persetujuan Anda, lakukan hal berikut di bagian Pengecualian patch:

    - Untuk Patch yang ditolak, masukkan daftar patch yang dipisahkan koma yang ingin Anda tolak.

> **Note**
>
> Untuk informasi tentang format yang diterima untuk daftar patch yang disetujui dan patch yang ditolak, lihat Tentang format nama paket untuk daftar patch yang disetujui dan ditolak.

- Untuk tindakan tambalan Ditolak, pilih tindakan yang akan diambil pada tambalan yang disertakan dalam daftar tambalan Ditolak. Patch Manager
  - Diizinkan sebagai dependensi: Sebuah paket di daftar Patch yang ditolak diinstal hanya jika merupakan dependensi dari paket lain. Ini dianggap sesuai dengan baseline patch dan statusnya dilaporkan sebagai. InstalledOther Ini adalah tindakan default jika tidak ada pilihan yang ditentukan.
  - Blokir: Paket dalam daftar tambalan Ditolak, dan paket yang menyertakannya sebagai dependensi, tidak diinstal dalam Patch Manager keadaan apa pun. Jika sebuah paket diinstal sebelum ditambahkan ke daftar tambalan Ditolak, atau diinstal di luar Patch Manager sesudahnya, paket tersebut dianggap tidak sesuai dengan garis dasar tambalan dan statusnya dilaporkan sebagai. InstalledRejected

11. (Opsional) Untuk Kelola tag, terapkan satu atau lebih pasangan nama/nilai kunci tag ke dasar patch.

    Tag adalah metadata opsional yang Anda tetapkan ke sumber daya. Tag memungkinkan Anda untuk mengkategorikan sumber daya dengan berbagai cara, seperti berdasarkan tujuan, pemilik, atau lingkungan. Sebagai contoh, Anda mungkin ingin menandai dasar patch untuk mengidentifikasi tingkat kepelikan patch yang ditentukannya, pengelola paket tempat patch diterapkan, dan jenis lingkungan. Dalam kasus ini, Anda dapat menentukan tag yang serupa dengan pasangan nama/nilai kunci berikut:

    - `Key=PatchSeverity,Value=Critical`
    - `Key=PackageManager,Value=softwareupdate`
    - `Key=Environment,Value=Production`

12. Pilih Buat dasar patch.

Membuat dasar patch kustom (Windows)

Gunakan prosedur berikut untuk membuat baseline patch kustom untuk node terkelola Windows diPatch Manager, kemampuan. AWS Systems Manager

Untuk informasi tentang membuat baseline patch untuk node terkelola Linux, lihat. Membuat dasar patch kustom (Linux) Untuk informasi tentang membuat baseline patch untuk node macOS terkelola, lihat. Membuat dasar patch kustom (macOS)

Untuk contoh membuat dasar patch yang terbatas untuk menginstal Windows Service Pack saja, lihat Tutorial: Buat baseline patch untuk menginstal Paket Layanan Windows (konsol).

Untuk membuat dasar patch kustom (Windows)

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Patch Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.
3. Pilih tab Patch baseline, lalu pilih Create patch baseline.

   -atau-

   Jika Anda mengakses Patch Manager untuk pertama kalinya saat ini Wilayah AWS, pilih Mulai dengan ikhtisar, pilih tab Garis dasar Patch, lalu pilih Buat baseline patch.
4. Untuk Nama, masukkan nama untuk dasar patch baru Anda, misalnya, `MyWindowsPatchBaseline`.
5. (Opsional) Untuk Deskripsi, masukkan deskripsi untuk dasar patch ini.
6. Untuk Sistem operasi, pilih `Windows`.
7. Jika Anda ingin mulai menggunakan dasar patch ini sebagai default untuk Windows segera setelah Anda membuatnya, pilih Atur dasar patch ini sebagai dasar patch default untuk instans Windows Server.

> **Note**
>
> Opsi ini hanya tersedia jika Anda pertama kali mengakses Patch Manager sebelum kebijakan tambalan dirilis pada 22 Desember 2022.
> Untuk informasi tentang mengatur dasar patch yang ada sebagai default, lihat Mengatur dasar patch yang ada sebagai default.

8. Di bagian Aturan persetujuan untuk sistem operasi, gunakan bidang tersebut untuk membuat satu atau lebih aturan persetujuan otomatis.

   - Produk: Versi sistem operasi yang berlaku untuk aturan persetujuan, seperti`WindowsServer2012`. Pilihan default adalah `All`.
   - Klasifikasi: Jenis patch yang diberlakukan aturan persetujuan, seperti `CriticalUpdates`, `Drivers`, dan `Tools`. Pilihan default adalah `All`.

     > **Tip**
     >
     > Anda dapat menyertakan instalasi Windows Service Pack dalam aturan persetujuan Anda dengan menyertakan `ServicePacks` atau dengan memilih `All` di daftar Klasifikasi Anda. Sebagai contoh, lihat Tutorial: Buat baseline patch untuk menginstal Paket Layanan Windows (konsol).

   - Kepelikan: Nilai kepelikan patch yang diberlakukan aturan, seperti `Critical`. Pilihan default adalah `All`.
   - Persetujuan otomatis: Metode untuk memilih patch untuk persetujuan otomatis.
     - Menyetujui patch setelah beberapa hari tertentu: Jumlah hari Patch Manager untuk menunggu setelah patch dirilis atau diperbarui sebelum patch secara otomatis disetujui. Anda dapat memasukkan bilangan bulat apa saja dari nol (0) sampai 360. Untuk sebagian besar skenario, kami merekomendasikan untuk menunggu tidak lebih dari 100 hari.
     - Menyetujui patch yang dirilis hingga tanggal tertentu: Tanggal rilis patch yang Patch Manager secara otomatis menerapkan semua patch yang dirilis atau diperbarui pada atau sebelum tanggal tersebut. Misalnya, jika Anda menentukan 7 Juli 2023, tidak ada tambalan yang dirilis atau terakhir diperbarui pada atau setelah 8 Juli 2023, yang diinstal secara otomatis.
   - (Opsional) Laporan kepatuhan: Tingkat kepelikan yang ingin Anda tetapkan untuk patch yang disetujui oleh baseline, seperti `High`.

     > **Note**
     >
     > Jika Anda menentukan tingkat pelaporan kepatuhan dan status patch dari setiap patch yang disetujui dilaporkan sebagai`Missing`, maka tingkat keparahan kepatuhan yang dilaporkan secara keseluruhan baseline patch adalah tingkat keparahan yang Anda tentukan.

9. (Opsional) Di bagian Aturan persetujuan untuk sistem operasi, gunakan bidang tersebut untuk membuat satu atau lebih aturan persetujuan otomatis.

> **Note**
>
> Alih-alih menentukan aturan persetujuan, Anda dapat menentukan daftar patch yang disetujui dan ditolak sebagai pengecualian patch. Lihat langkah 10 dan 11.

- Keluarga produk: Keluarga produk Microsoft umum yang Anda ingin tentukan aturan, seperti `Office` atau `Exchange Server`.
- Produk: Versi aplikasi yang berlaku untuk aturan persetujuan, seperti `Office 2016` atau`Active Directory Rights Management Services Client 2.0 2016`. Pilihan default adalah `All`.
- Klasifikasi: Jenis patch yang diberlakukan aturan persetujuan, seperti `CriticalUpdates`. Pilihan default adalah `All`.
- Kepelikan: Nilai kepelikan patch yang diberlakukan aturan, seperti `Critical`. Pilihan default adalah `All`.
- Persetujuan otomatis: Metode untuk memilih patch untuk persetujuan otomatis.
  - Menyetujui patch setelah beberapa hari tertentu: Jumlah hari Patch Manager untuk menunggu setelah patch dirilis atau diperbarui sebelum patch secara otomatis disetujui. Anda dapat memasukkan bilangan bulat apa saja dari nol (0) sampai 360. Untuk sebagian besar skenario, kami merekomendasikan untuk menunggu tidak lebih dari 100 hari.
  - Menyetujui patch yang dirilis hingga tanggal tertentu: Tanggal rilis patch yang Patch Manager secara otomatis menerapkan semua patch yang dirilis atau diperbarui pada atau sebelum tanggal tersebut. Misalnya, jika Anda menentukan 7 Juli 2023, tidak ada tambalan yang dirilis atau terakhir diperbarui pada atau setelah 8 Juli 2023, yang diinstal secara otomatis.
- (Opsional) Pelaporan kepatuhan: Tingkat keparahan yang ingin Anda tetapkan ke tambalan yang disetujui oleh baseline, seperti atau. `Critical` `High`

> **Note**
>
> Jika Anda menentukan tingkat pelaporan kepatuhan dan status patch dari setiap patch yang disetujui dilaporkan sebagai`Missing`, maka tingkat keparahan kepatuhan yang

dilaporkan secara keseluruhan baseline patch adalah tingkat keparahan yang Anda tentukan.

10. (Opsional) Jika Anda ingin secara eksplisit menyetujui patch apa saja dan bukan membiarkan patch dipilih sesuai dengan aturan persetujuan, lakukan hal berikut di bagian Pengecualian patch:

    - Untuk Patch yang disetujui, masukkan daftar patch yang dipisahkan koma yang ingin Anda setujui.

      > **Note**
      >
      > Untuk informasi tentang format yang diterima untuk daftar patch yang disetujui dan patch yang ditolak, lihat Tentang format nama paket untuk daftar patch yang disetujui dan ditolak.

    - (Opsional) Untuk Tingkat kepatuhan patch yang disetujui, tetapkan tingkat kepatuhan pada patch dalam daftar.

11. Jika Anda ingin secara eksplisit menolak patch lain selain yang memenuhi aturan persetujuan Anda, lakukan hal berikut di bagian Pengecualian patch:

    - Untuk Patch yang ditolak, masukkan daftar patch yang dipisahkan koma yang ingin Anda tolak.

      > **Note**
      >
      > Untuk informasi tentang format yang diterima untuk daftar patch yang disetujui dan patch yang ditolak, lihat Tentang format nama paket untuk daftar patch yang disetujui dan ditolak.

    - Untuk tindakan tambalan Ditolak, pilih tindakan yang akan diambil pada tambalan yang disertakan dalam daftar tambalan Ditolak. Patch Manager
      - Diizinkan sebagai dependensi: Sebuah paket di daftar Patch yang ditolak diinstal hanya jika merupakan dependensi dari paket lain. Ini dianggap sesuai dengan baseline patch dan statusnya dilaporkan sebagai. InstalledOther Ini adalah tindakan default jika tidak ada pilihan yang ditentukan.
      - Blokir: Paket dalam daftar tambalan Ditolak, dan paket yang menyertakannya sebagai dependensi, tidak diinstal dalam Patch Manager keadaan apa pun. Jika sebuah paket

diinstal sebelum ditambahkan ke daftar tambalan Ditolak, atau diinstal di luar Patch Manager sesudahnya, paket tersebut dianggap tidak sesuai dengan garis dasar tambalan dan statusnya dilaporkan sebagai. InstalledRejected

12. (Opsional) Untuk Kelola tag, terapkan satu atau lebih pasangan nama/nilai kunci tag ke dasar patch.

    Tag adalah metadata opsional yang Anda tetapkan ke sumber daya. Tag memungkinkan Anda untuk mengkategorikan sumber daya dengan berbagai cara, seperti berdasarkan tujuan, pemilik, atau lingkungan. Sebagai contoh, Anda mungkin ingin menandai dasar patch untuk mengidentifikasi tingkat kepelikan patch yang ditentukannya, keluarga sistem operasi tempat patch diterapkan, dan jenis lingkungan. Dalam kasus ini, Anda dapat menentukan tag yang serupa dengan pasangan nama/nilai kunci berikut:

    - `Key=PatchSeverity,Value=Critical`
    - `Key=OS,Value=RHEL`
    - `Key=Environment,Value=Production`

13. Pilih Buat dasar patch.

Memperbarui atau menghapus baseline patch kustom

Anda dapat memperbarui atau menghapus baseline patch kustom yang telah Anda buatPatch Manager, kemampuan. AWS Systems Manager Ketika Anda memperbarui dasar patch, Anda dapat mengubah nama atau deskripsi, aturan persetujuan, dan pengecualian untuk patch yang disetujui dan ditolak. Anda juga dapat memperbarui tag yang diterapkan ke dasar patch. Anda tidak dapat mengubah jenis sistem operasi yang telah dibuatkan sebuah dasar patch, dan Anda tidak dapat membuat perubahan pada dasar patch yang telah ditetapkan yang disediakan oleh AWS.

Memperbarui atau menghapus baseline patch

Ikuti langkah-langkah berikut ini untuk memperbarui atau menghapus dasar patch.

> **Important**
>
> Berhati-hatilah saat menghapus baseline patch khusus yang mungkin digunakan oleh konfigurasi kebijakan tambalan di. Quick Setup
> Jika Anda menggunakan konfigurasi kebijakan tambalanQuick Setup, pembaruan yang Anda buat ke baseline patch kustom disinkronkan dengan Quick Setup satu jam sekali.

Jika baseline patch kustom yang direferensikan dalam kebijakan tambalan dihapus, spanduk akan ditampilkan di halaman Detail Quick Setup konfigurasi untuk kebijakan tambalan Anda. Spanduk memberi tahu Anda bahwa kebijakan tambalan mereferensikan baseline tambalan yang tidak ada lagi, dan operasi penambalan berikutnya akan gagal. Dalam hal ini, kembali ke halaman Quick Setup Konfigurasi, pilih Patch Manager konfigurasi, dan pilih Tindakan, Edit konfigurasi. Nama dasar patch yang dihapus disorot, dan Anda harus memilih baseline patch baru untuk sistem operasi yang terpengaruh.

Untuk memperbarui atau menghapus baseline patch

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Patch Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.
3. Pilih dasar patch yang ingin Anda perbarui atau hapus, lalu lakukan salah satu hal berikut:

   - Untuk menghapus dasar patch dari Akun AWS Anda, pilih Hapus. Sistem meminta Anda untuk mengonfirmasi tindakan Anda.
   - Untuk membuat perubahan pada nama atau deskripsi, aturan persetujuan, atau pengecualian patch untuk dasar patch tersebut, pilih Edit. Pada halaman Edit dasar patch, ubah nilai dan pilihan yang Anda inginkan, lalu pilih Simpan perubahan.
   - Untuk menambah, mengubah, atau menghapus tag yang diterapkan ke dasar patch, pilih tab Tag, dan kemudian pilih Edit tag. Pada halaman Edit tag dasar patch, buat pembaruan ke tag dasar patch, lalu pilih Simpan perubahan.

   Untuk informasi tentang pilihan konfigurasi yang dapat Anda buat, lihat Bekerja dengan dasar patch kustom.

Mengatur dasar patch yang ada sebagai default

> **Important**
>
> Setiap pilihan dasar tambalan default yang Anda buat di sini tidak berlaku untuk operasi penambalan yang didasarkan pada kebijakan tambalan. Kebijakan patch menggunakan spesifikasi dasar patch mereka sendiri. Untuk informasi selengkapnya tentang kebijakan tambalan, lihatMenggunakan kebijakan Quick Setup tambalan.

Saat Anda membuat baseline patch kustom diPatch Manager, kemampuanAWS Systems Manager, Anda dapat mengatur baseline sebagai default untuk jenis sistem operasi terkait segera setelah Anda membuatnya. Untuk informasi, lihat Bekerja dengan dasar patch kustom.

Anda juga dapat mengatur dasar patch yang ada sebagai default untuk sebuah jenis sistem operasi.

> **Note**
>
> Langkah-langkah yang Anda ikuti bergantung pada apakah Anda pertama kali mengakses Patch Manager sebelum atau setelah rilis kebijakan tambalan pada 22 Desember 2022. Jika Anda menggunakan Patch Manager sebelum tanggal itu, Anda dapat menggunakan prosedur konsol. Jika tidak, gunakan AWS CLI prosedurnya. Menu Tindakan yang direferensikan dalam prosedur konsol tidak ditampilkan di Wilayah yang Patch Manager tidak digunakan sebelum rilis kebijakan tambalan.

Untuk mengatur dasar patch yang ada sebagai default

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Patch Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.
3. Pilih tab Dasar patch.

4. Dalam daftar dasar patch, pilih tombol dasar patch yang saat ini tidak diatur sebagai default untuk suatu jenis sistem operasi.

   Kolom Baseline default menunjukkan baseline apa yang saat ini diatur sebagai default.

5. Di menu Tindakan, pilih Atur dasar patch default.

   > **Important**
   >
   > Menu Tindakan tidak tersedia jika Anda tidak bekerja dengan Patch Manager saat ini Akun AWS dan Wilayah sebelum 22 Desember 2022. Lihat Catatan sebelumnya dalam topik ini untuk informasi selengkapnya.

6. Di kotak dialog konfirmasi, pilih Atur default.

Untuk menyetel baseline patch sebagai default () AWS CLI

1. Jalankan describe-patch-baselinesperintah untuk melihat daftar baseline patch yang tersedia dan ID serta Amazon Resource Names (ARN).

   ```
   aws ssm describe-patch-baselines
   ```

2. Jalankan register-default-patch-baselineperintah untuk menetapkan garis dasar sebagai default untuk sistem operasi yang terkait dengannya. Ganti *baseline-id-or-ARN* dengan ID dari baseline patch kustom atau baseline yang telah ditentukan untuk digunakan.

   Linux & macOS

   ```
   aws ssm register-default-patch-baseline \
       --baseline-id baseline-id-or-ARN
   ```

   Berikut ini adalah contoh pengaturan baseline kustom sebagai default.

   ```
   aws ssm register-default-patch-baseline \
       --baseline-id pb-abc123cf9bEXAMPLE
   ```

   Berikut ini adalah contoh pengaturan garis dasar yang telah ditentukan yang dikelola oleh AWS sebagai default.

```
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
```

Berikut ini adalah contoh pengaturan baseline kustom sebagai default.

```
aws ssm register-default-patch-baseline ^
    --baseline-id pb-abc123cf9bEXAMPLE
```

Berikut ini adalah contoh pengaturan garis dasar yang telah ditentukan yang dikelola oleh AWS sebagai default.

```
aws ssm register-default-patch-baseline ^
    --baseline-id arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-071da192df1226b63
```

Melihat metrik yang tersedia

Dengan kemampuan Patch ManagerAWS Systems Manager, Anda dapat melihat semua tambalan yang tersedia untuk sistem operasi tertentu dan, secara opsional, versi sistem operasi tertentu.

> **Tip**
>
> Untuk menghasilkan daftar tambalan yang tersedia dan menyimpannya ke file, Anda dapat menggunakan describe-available-patchesperintah dan menentukan output pilihan Anda.

Untuk melihat patch yang tersedia

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Patch Manager.

-atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.

3. Pilih tab Patch.

   -atau-

   Jika Anda mengakses Patch Manager untuk pertama kalinya saat iniWilayah AWS, pilih Mulai dengan ikhtisar, lalu pilih tab Patch.

   > **Note**
   >
   > UntukWindows Server, tab Patch menampilkan pembaruan yang tersedia dari Windows Server Update Service (WSUS).

4. Untuk Sistem operasi, pilih sistem operasi yang ingin Anda lihat patch yang tersedia, seperti `Windows` atau `Amazon Linux`.
5. (Opsional) Untuk Produk, pilih versi OS, seperti `WindowsServer2019` atau `AmazonLinux2018.03`.
6. (Opsional) Untuk menambah atau menghapus kolom informasi untuk hasil Anda, pilih tombol konfigurasikan (⚙) di kanan atas daftar Patch. (Secara default, tab Patch menampilkan kolom untuk hanya beberapa metadata patch yang tersedia.)

   Untuk informasi tentang jenis metadata yang dapat Anda tambahkan ke tampilan Anda, lihat Patch dalam Referensi API AWS Systems Manager.

Bekerja dengan kelompok patch

Jika Andatidakmenggunakan kebijakan tambalan dalam operasi Anda, Anda dapat mengatur upaya penambalan dengan menambahkan node terkelola ke grup patch dengan menggunakan tag.

> **Important**
>
> Grup patch tidak digunakan dalam operasi patching yang didasarkan padakebijakan tambalan. Untuk informasi selengkapnya tentang bekerja dengan kebijakan tambalan, lihatMenggunakan kebijakan Quick Setup tambalan.

Untuk menggunakan tag dalam operasi patching, Anda harus menerapkan kunci tag`Patch GroupatauPatchGroup`ke node terkelola Anda. Anda juga harus menentukan nama yang ingin Anda berikan pada grup tambalan sebagai nilai tag. Anda dapat menentukan nilai tag apa pun, tetapi kunci tag harus`Patch GroupatauPatchGroup`.

`PatchGroup`(tanpa spasi) diperlukan jika Anda memilikitag yang diizinkan dalam metadata instans EC2.

Setelah mengelompokkan node terkelola menggunakan tag, Anda menambahkan nilai grup tambalan ke garis dasar tambalan. Dengan mendaftarkan grup patch dengan dasar patch, Anda memastikan bahwa patch yang benar diinstal selama operasi patching. Untuk informasi selengkapnya tentang grup patch, lihat Tentang grup patch.

Selesaikan tugas dalam topik ini untuk mempersiapkan node terkelola Anda untuk ditambal menggunakan tag dengan node dan patch baseline Anda. Tugas 1 hanya diperlukan jika Anda menambal instans Amazon EC2. Tugas 2 hanya diperlukan jika Anda menambal instans non-EC2 dihibrida dan multicloudlingkungan. Tugas 3 diperlukan untuk semua node yang dikelola.

> **Tip**
>
> Anda juga dapat menambahkan tag ke node yang dikelola menggunakanAWS CLIkomando`add-tags-to-resource`atau operasi API Manajer Sistem`AddTagsToResource`.

Tugas

- Tugas 1: Tambahkan instans EC2 ke grup patch menggunakan tag
- Tugas 2: Menambahkan node terkelola ke grup tambalan menggunakan tag
- Tugas 3: Tambahkan grup patch ke dasar patch

Tugas 1: Tambahkan instans EC2 ke grup patch menggunakan tag

Anda dapat menambahkan tag ke instans EC2 menggunakan konsol Systems Manager atau konsol Amazon EC2. Tugas ini hanya diperlukan jika Anda menambal instans Amazon EC2.

> **Important**
>
> Anda tidak dapat menerapkan`Patch Group`tag (dengan spasi) ke instans Amazon EC2 jikaIzinkan tag dalam metadata instancepilihan diaktifkan pada contoh. Mengizinkan tag dalam metadata instance mencegah nama kunci tag berisi spasi. Jika Anda memilikitag yang diizinkan dalam metadata instans EC2, Anda harus menggunakan kunci tag`PatchGroup`(tanpa spasi).

Opsi 1: Untuk menambahkan instans EC2 ke grup tambalan (konsol Manajer Sistem)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   JikaAWS Systems Managerhalaman rumah terbuka pertama, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilihFleet Managerdi panel navigasi.
3. DalamNode yang dikelolalist, pilih ID instans EC2 terkelola yang ingin Anda konfigurasikan untuk ditambal. ID Node untuk instans EC2 dimulai dengan`i-`.

   > **Note**
   >
   > Saat menggunakan konsol Amazon EC2 danAWS CLI, itu mungkin untuk menerapkan`Key = Patch Group`atau`Key = PatchGroup`tag ke instance yang belum dikonfigurasi untuk digunakan dengan Systems Manager.
   > Jika node terkelola yang Anda harapkan untuk melihat tidak terdaftar, lihatMemecahkan masalah ketersediaan node terkelolauntuk tips pemecahan masalah.
4. PilihTagtab, lalu pilihMengedit.
5. Di kolom kiri, masukkan**`Patch Group`**atau**`PatchGroup`**. Jika Anda memilikitag yang diizinkan dalam metadata instans EC2, Anda harus menggunakan`PatchGroup`(tanpa spasi).

6. Di kolom kanan, masukkan nilai tag untuk berfungsi sebagai nama untuk grup tambalan.
7. Pilih Save (Simpan).
8. Ulangi prosedur ini untuk menambahkan instans EC2 lainnya ke grup patch yang sama.

Opsi 2: Untuk menambahkan instans EC2 ke grup patch (konsol Amazon EC2)

1. Buka konsol Amazon EC2, lalu pilih Instans di panel navigasi.
2. Di daftar instans terkelola, pilih instans yang ingin Anda konfigurasikan untuk patching.
3. DalamAksimenu, pilihPengaturan instans,Mengelola tag.
4. Pilih Add new tag (Tambahkan tanda baru).
5. UntukKunci, masuklah**`Patch Group`**atau**`PatchGroup`**. Jika Anda memilikitag yang diizinkan dalam metadata instans EC2, Anda harus menggunakan`PatchGroup`(tanpa spasi).
6. UntukNilai, masukkan nilai untuk berfungsi sebagai nama untuk grup patch.
7. Pilih Simpan.
8. Ulangi prosedur ini untuk menambahkan instans lainnya ke grup patch yang sama.

Tugas 2: Menambahkan node terkelola ke grup tambalan menggunakan tag

Ikuti langkah-langkah dalam topik ini untuk menambahkan tagAWS IoT Greengrassperangkat inti dan node terkelola non-EC2 yang diaktifkan hibrida (mi-*). Tugas ini hanya diperlukan jika Anda menambal instans non-EC2 di lingkungan hybrid dan multicloud.

> **Note**
>
> Anda tidak dapat menambahkan tag untuk node yang dikelola non-EC2 menggunakan konsol Amazon EC2.

Untuk menambahkan node terkelola non-EC2 ke grup tambalan (konsol Manajer Sistem)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

JikaAWS Systems Managerhalaman rumah terbuka pertama, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilihFleet Managerdi panel navigasi.

3. DalamNode yang dikelolalist, pilih nama node terkelola yang ingin Anda konfigurasikan untuk ditambal.

> **Note**
>
> Jika node terkelola yang Anda harapkan untuk melihat tidak terdaftar, lihatMemecahkan masalah ketersediaan node terkelolauntuk tips pemecahan masalah.

4. PilihTagtab, lalu pilihMengedit.
5. Di kolom kiri, masukkan**`Patch Group`**atau**`PatchGroup`**. Jika Anda memilikitag yang diizinkan dalam metadata instans EC2, Anda harus menggunakan`PatchGroup`(tanpa spasi).
6. Di kolom kanan, masukkan nilai tag untuk berfungsi sebagai nama untuk grup tambalan.
7. Pilih Save (Simpan).
8. Ulangi prosedur ini untuk menambahkan node terkelola lainnya ke grup patch yang sama.

Tugas 3: Tambahkan grup patch ke dasar patch

Untuk mengaitkan baseline patch tertentu dengan node terkelola, Anda harus menambahkan nilai grup patch ke garis dasar patch. Dengan mendaftarkan grup patch dengan dasar patch, Anda memastikan bahwa patch yang benar diinstal selama operasi patching. Tugas ini diperlukan apakah Anda menambal instans EC2, node yang dikelola non-EC2, atau keduanya.

Untuk informasi selengkapnya tentang grup patch, lihat Tentang grup patch.

> **Note**
>
> Langkah-langkah yang Anda ikuti bergantung pada apakah Anda pertama kali mengaksesPatch Managersebelum atau sesudahkebijakan tambalanrilis pada 22 Desember 2022.

Untuk menambahkan grup tambalan ke garis dasar patch (konsol Manajer Sistem)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.

2. Di panel navigasi, pilih Patch Manager.
3. Jika Anda mengaksesPatch Manageruntuk pertama kalinya di saat iniWilayah AWSdanPatch Managerhalaman awal terbuka, pilihMulailah dengan ikhtisar.
4. PilihPatch garis dasartab, dan kemudian diPatch garis dasardaftar, pilih nama dasar patch yang ingin Anda konfigurasikan untuk grup tambalan Anda.

   Jika Anda tidak pertama kali mengaksesPatch Managersampai setelah rilis kebijakan patch, Anda harus memilih dasar kustom yang telah Anda buat.
5. JikaID dasarrincian halaman termasukAksimenu, lakukan hal berikut:

   - Pilih Tindakan, kemudian Modifikasi grup patch.
   - Masukkan tagnilaiyang Anda tambahkan ke node terkelolaTugas 2: Menambahkan node terkelola ke grup tambalan menggunakan tag, lalu pilihMenambahkan.

   JikaID dasarHalaman rincian tidaktidaktermasukAksimenu, grup tambalan tidak dapat dikonfigurasi di konsol. Sebagai gantinya, Anda dapat melakukan salah satu dari berikut ini:

   - (Disarankan) Menyiapkan kebijakan tambalan diQuick Setup, kemampuanAWS Systems Manager, untuk memetakan patch baseline ke satu atau lebih instans EC2.

     Untuk informasi lebih lanjut, lihatMenggunakanQuick Setupkebijakan tambalandanMengotomatiskan patching seluruh organisasi menggunakanQuick Setupkebijakan tambalan.
   - Gunakanregister-patch-baseline-for-patch-groupperintah diAWS Command Line Interface(AWS CLI) untuk mengkonfigurasi grup patch.

Bekerja denganPatch Managerpengaturan

Topik

- Integrasi dengan Patch ManagerAWS Security Hub

Integrasi dengan Patch ManagerAWS Security Hub

AWS Security Hub menyediakan pandangan komprehensif kepada Anda tentang status keamanan Anda di AWS. Security Hub mengumpulkan data keamanan dari seluruh Akun AWSLayanan AWS, dan mendukung produk mitra pihak ketiga. Dengan Security Hub, Anda dapat memeriksa lingkungan

Anda terkait standar industri dan praktik terbaik untuk keamanan. Security Hub membantu Anda menganalisis tren keamanan Anda dan mengidentifikasi masalah keamanan prioritas tertinggi.

Dengan menggunakan integrasi antaraPatch Manager, kemampuan, dan Security HubAWS Systems Manager, Anda dapat mengirim temuan tentang node yang tidak sesuai dari Patch Manager ke Security Hub. Temuan adalah catatan yang dapat diamati dari pemeriksaan keamanan atau deteksi terkait keamanan. Security Hub kemudian dapat memasukkan temuan terkait tambalan tersebut dalam analisisnya tentang postur keamanan Anda.

> **Note**
>
> Informasi dalam topik berikut berlaku tidak peduli metode atau jenis konfigurasi yang Anda gunakan untuk operasi patching Anda:
>
> - Kebijakan tambalan yang dikonfigurasi di Quick Setup
> - Opsi Manajemen Host yang dikonfigurasi di Quick Setup
> - Jendela pemeliharaan untuk menjalankan tambalan `Scan` atau `Install` tugas
> - Patch sesuai permintaan sekarang beroperasi

Daftar Isi

Cara Patch Manager mengirim temuan ke Security Hub

Di Security Hub, masalah keamanan dilacak sebagai temuan. Beberapa temuan berasal dari masalah yang terdeteksi oleh pihak lain Layanan AWS atau oleh mitra pihak ketiga. Security Hub juga memiliki seperangkat aturan yang digunakan untuk mendeteksi masalah keamanan dan menghasilkan temuan.

Patch Manageradalah salah satu kemampuan Systems Manager yang mengirimkan temuan ke Security Hub. Setelah Anda melakukan operasi patching dengan menjalankan dokumen SSM (`AWS-RunPatchBaseline`, `AWS-RunPatchBaselineAssociation`, atau `AWS-RunPatchBaselineWithHooks`), informasi patching dikirim ke Inventaris atau Kepatuhan, kemampuan AWS Systems Manager, atau keduanya. Setelah Inventaris, Kepatuhan, atau keduanya menerima data, Patch Manager menerima pemberitahuan. Kemudian, Patch Manager mengevaluasi data untuk akurasi, pemformatan, dan kepatuhan. Jika semua kondisi terpenuhi, Patch Manager teruskan data ke Security Hub.

Security Hub menyediakan alat untuk mengelola temuan dari seluruh sumber tersebut. Anda dapat melihat dan mem-filter daftar temuan dan melihat detail suatu temuan. Untuk informasi lebih lanjut, lihat Melihat temuan dalam Panduan Pengguna AWS Security Hub. Anda juga dapat melacak status penyelidikan temuan. Untuk informasi lebih lanjut, lihat Mengambil tindakan pada temuan dalam Panduan Pengguna AWS Security Hub.

Semua temuan di Security Hub menggunakan format JSON standar yang disebut AWS Security Finding Format (ASFF). ASFF mencakup detail tentang sumber masalah, sumber daya yang terdampak, dan status temuan saat ini. Untuk informasi lebih lanjut, lihat AWS Security Finding Format (ASFF) di Panduan Pengguna AWS Security Hub.

Jenis temuan yang dikirim Patch Manager

Patch Managermengirimkan temuan ke Security Hub menggunakan AWSSecurity Finding Format (ASFF). Dalam ASFF, bidang `Types` menyediakan jenis temuan. Temuan dari Patch Manager memiliki nilai sebagai berikut untuk`Types`:

- Pemeriksaan Perangkat Lunak dan Konfigurasi/Pengelolaan Patch

Patch Managermengirimkan satu temuan per node terkelola yang tidak sesuai. Temuan ini dilaporkan dengan tipe sumber daya `AwsEc2Instance`sehingga temuan dapat dikorelasikan dengan integrasi Security Hub lainnya yang melaporkan jenis `AwsEc2Instance` sumber daya. Patch Managerhanya meneruskan temuan ke Security Hub jika operasi menemukan node terkelola tidak patuh. Temuan ini mencakup hasil Ringkasan Patch. Untuk informasi selengkapnya tentang definisi kepatuhan, lihat Memahami nilai keadaan kepatuhan patch. Untuk informasi selengkapnya`PatchSummary`, lihat PatchSummarydi Referensi AWS Security Hub API.

Latensi untuk mengirim temuan

Saat Patch Manager membuat temuan baru, biasanya dikirim ke Security Hub dalam beberapa detik hingga 2 jam. Kecepatan ini bergantung pada lalu lintas dalam Wilayah AWS yang sedang diproses pada saat itu.

Mencoba kembali saat Security Hub tidak tersedia

Jika ada pemadaman layanan, fungsi AWS Lambda dijalankan untuk menempatkan pesan kembali ke antrean utama setelah layanan berjalan lagi. Setelah pesan berada di antrean utama, coba lagi berjalan secara otomatis.

Jika Security Hub tidak Patch Manager tersedia, coba lagi mengirimkan temuan sampai diterima.

Memperbarui temuan yang ada di Security Hub

Prosedur ini menjelaskan cara melihat temuan di Security Hub tentang node terkelola di armada Anda yang tidak sesuai dengan patch.

Untuk meninjau temuan Security Hub untuk kepatuhan patch

1. Masuk ke AWS Management Console dan buka AWS Security Hub konsol di https://console.aws.amazon.com/securityhub/.
2. Di panel navigasi, pilih Temuan.
3. Pilih kotak Tambahkan filter ().
4. Di menu, di bawah Filter, pilih Nama produk.
5. Di kotak dialog yang terbuka, pilih ada di bidang pertama dan kemudian masukkan **`Systems Manager Patch Manager`** di bidang kedua.
6. Pilih Apply (Terapkan).
7. Tambahkan filter tambahan yang Anda inginkan untuk membantu mempersempit hasil Anda.
8. Dalam daftar hasil, pilih judul temuan yang Anda inginkan informasi lebih lanjut.

   Panel terbuka di sisi kanan layar dengan detail lebih lanjut tentang sumber daya, masalah yang ditemukan, dan perbaikan yang disarankan.

> **Important**
>
> Pada saat ini, Security Hub melaporkan jenis sumber daya dari semua node terkelola sebagai`EC2 Instance`. Ini termasuk server lokal dan mesin virtual (VM) yang telah Anda daftarkan untuk digunakan dengan Systems Manager.

Klasifikasi keparahan

Daftar temuan untuk **`Systems Manager Patch Manager`** mencakup laporan tingkat keparahan temuan. Tingkat keparahan meliputi yang berikut, dari terendah ke tertinggi:

- INFORMASI - Tidak ada masalah yang ditemukan.
- RENDAH — Masalah ini tidak memerlukan remediasi.
- MEDIUM — Masalah ini harus ditangani tetapi tidak mendesak.
- TINGGI — Masalah ini harus ditangani sebagai prioritas.
- KRITIS — Masalah ini harus segera diperbaiki untuk menghindari eskalasi.

Tingkat keparahan ditentukan oleh paket noncompliant yang paling parah pada sebuah instance. Karena Anda dapat memiliki beberapa garis dasar tambalan dengan beberapa tingkat keparahan, tingkat keparahan tertinggi dilaporkan dari semua paket yang tidak sesuai. Misalnya, Anda memiliki dua paket yang tidak sesuai di mana tingkat keparahan paket A adalah “Kritis” dan tingkat keparahan paket B adalah “Rendah”. “Kritis” akan dilaporkan sebagai tingkat keparahan.

Perhatikan bahwa bidang keparahan berkorelasi langsung dengan bidang. Patch Manager `Compliance` Ini adalah bidang yang Anda tetapkan tetapkan ke tambalan individual yang cocok dengan aturan. Karena `Compliance` bidang ini ditetapkan ke tambalan individual, itu tidak tercermin pada tingkat Ringkasan Patch.

Konten terkait

- Temuan dalam Panduan AWS Security Hub Pengguna
- Kepatuhan patch Multi-Akun Patch Manager dan Security Hub di Blog AWSManajemen & Tata Kelola

Temuan standar dari Patch Manager

Patch Managermengirimkan temuan ke Security Hub menggunakan AWSSecurity Finding Format (ASFF).

Berikut adalah contoh temuan Patch Manager.


```
{
  "SchemaVersion": "2018-10-08",
  "Id": "arn:aws:patchmanager:us-east-2:111122223333:instance/i-02573cafcfEXAMPLE/
document/AWS-RunPatchBaseline/run-command/d710f5bd-04e3-47b4-82f6-df4e0EXAMPLE",
  "ProductArn": "arn:aws:securityhub:us-east-1::product/aws/ssm-patch-manager",
  "GeneratorId": "d710f5bd-04e3-47b4-82f6-df4e0EXAMPLE",
  "AwsAccountId": "111122223333",
  "Types": [
    "Software & Configuration Checks/Patch Management/Compliance"
  ],
  "CreatedAt": "2021-11-11T22:05:25Z",
  "UpdatedAt": "2021-11-11T22:05:25Z",
  "Severity": {
    "Label": "INFORMATIONAL",
    "Normalized": 0
  },
  "Title": "Systems Manager Patch Summary - Managed Instance Non-Compliant",
  "Description": "This AWS control checks whether each instance that is managed by AWS
 Systems Manager is in compliance with the rules of the patch baseline that applies to
 that instance when a compliance Scan runs.",
  "Remediation": {
    "Recommendation": {
      "Text": "For information about bringing instances into patch compliance, see
 'Remediating out-of-compliance instances (Patch Manager)'.",
      "Url": "https://docs.aws.amazon.com/systems-manager/latest/userguide/patch-
compliance-remediation.html"
    }
  },
  "SourceUrl": "https://us-east-2.console.aws.amazon.com/systems-manager/managed-
instances/i-02573cafcfEXAMPLE/patch?region=us-east-2",
  "ProductFields": {
    "aws/securityhub/FindingId": "arn:aws:securityhub:us-east-2::product/aws/ssm-
patch-manager/arn:aws:patchmanager:us-east-2:111122223333:instance/i-02573cafcfEXAMPLE/
document/AWS-RunPatchBaseline/run-command/d710f5bd-04e3-47b4-82f6-df4e0EXAMPLE",
    "aws/securityhub/ProductName": "Systems Manager Patch Manager",
    "aws/securityhub/CompanyName": "AWS"
  },
```

```
  "Resources": [
    {
      "Type": "AwsEc2Instance",
      "Id": "i-02573cafcfEXAMPLE",
      "Partition": "aws",
      "Region": "us-east-2"
    }
  ],
  "WorkflowState": "NEW",
  "Workflow": {
    "Status": "NEW"
  },
  "RecordState": "ACTIVE",
  "PatchSummary": {
    "Id": "pb-0c10e65780EXAMPLE",
    "InstalledCount": 45,
    "MissingCount": 2,
    "FailedCount": 0,
    "InstalledOtherCount": 396,
    "InstalledRejectedCount": 0,
    "InstalledPendingReboot": 0,
    "OperationStartTime": "2021-11-11T22:05:06Z",
    "OperationEndTime": "2021-11-11T22:05:25Z",
    "RebootOption": "NoReboot",
    "Operation": "SCAN"
  }
}
```


Mengaktifkan dan mengonfigurasikan integrasi

Untuk menggunakan Patch Manager integrasi dengan Security Hub, Anda harus mengaktifkan Security Hub. Untuk informasi tentang cara mengaktifkan Security Hub, lihat Menyiapkan Security Hub di Panduan Pengguna AWS Security Hub.

Prosedur berikut menjelaskan cara mengintegrasikan Patch Manager dan Security Hub ketika Security Hub sudah aktif tetapi Patch Manager integrasi dimatikan. Anda hanya perlu menyelesaikan prosedur ini jika integrasi dinonaktifkan secara manual.

Patch ManagerUntuk menambah integrasi Security Hub

1. Di panel navigasi, pilih Patch Manager.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.

2. Pilih tab Pengaturan.

   -atau-

   Jika Anda mengakses Patch Manager untuk pertama kalinya saat iniWilayah AWS, pilih Mulai dengan ikhtisar, lalu pilih tab Pengaturan.

3. Di bawah bagian Ekspor ke Security Hub, di sebelah kanan Temuan kepatuhan patch tidak diekspor ke Security Hub, pilih Aktifkan.

Cara menghentikan pengiriman temuan

Untuk berhenti mengirim temuan ke Security Hub, Anda dapat menggunakan konsol Security Hub atau API.

Untuk informasi selengkapnya, lihat topik berikut di Panduan Pengguna AWS Security Hub:

- Menonaktifkan dan mengaktifkan aliran temuan dari integrasi (konsol)
- Menonaktifkan alur temuan dari integrasi (Security Hub API,) AWS CLI

Bekerja denganPatch Manager(AWS CLI)

Bagian ini mencakup contoh-contohAWS Command Line Interface(AWS CLI) perintah yang dapat Anda gunakan untuk melakukan tugas konfigurasiPatch Manager, sebuah kemampuanAWS Systems Manager.

Untuk ilustrasi menggunakan AWS CLI untuk patching lingkungan server dengan menggunakan dasar patch kustom, lihat Tutorial: Menambal lingkungan server (AWS CLI).

Untuk informasi selengkapnya tentang menggunakan AWS CLI untuk tugas AWS Systems Manager, lihat bagian AWS Systems Manager dari Referensi Perintah AWS CLI.

Topik

- Perintah AWS CLI untuk dasar patch
- Perintah AWS CLI untuk grup patch

- Perintah AWS CLI untuk melihat ringkasan dan detail patch
- AWS CLIperintah untuk memindai dan menambal node terkelola

Perintah AWS CLI untuk dasar patch

Sampel perintah untuk dasar patch

- Membuat dasar patch
- Buat dasar patch dengan repositori kustom untuk versi OS yang berbeda
- Perbarui dasar patch
- Ubah nama dasar patch
- Hapus dasar patch
- Cantumkan semua dasar patch
- Cantumkan semua dasar patch yang disediakan AWS
- Cantumkan dasar patch saya
- Tampilkan dasar patch
- Dapatkan dasar patch default
- Atur dasar patch kustom sebagai default
- Atur ulang dasar patch AWS sebagai default
- Tandai dasar patch
- Cantumkan tag untuk dasar patch
- Hapus tag dari dasar patch

Membuat dasar patch

Perintah berikut membuat baseline patch yang menyetujui semua pembaruan keamanan penting dan pentingWindows Server2012 R2 5 hari setelah mereka dirilis. Patch juga telah ditentukan untuk daftar patch yang Disetujui dan Ditolak. Selain itu, dasar patch telah ditandai untuk menunjukkan bahwa itu untuk lingkungan produksi.

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}]"
```

Server Windows

```
aws ssm create-patch-baseline ^
    --name "Windows-Server-2012R2" ^
    --tags "Key=Environment,Value=Production" ^
    --description "Windows Server 2012 R2, Important and Critical security updates"
 ^
    --approved-patches "KB2032276,MS10-048" ^
    --rejected-patches "KB2124261" ^
    --rejected-patches-action "ALLOW_AS_DEPENDENCY" ^
    --approval-rules
 "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=MSRC_SEVERITY,Values=[Important,Critical
{Key=CLASSIFICATION,Values=SecurityUpdates},
{Key=PRODUCT,Values=WindowsServer2012R2}]},ApproveAfterDays=5}]"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Buat dasar patch dengan repositori kustom untuk versi OS yang berbeda

Berlaku untuk node yang dikelola Linux saja. Perintah berikut ini menunjukkan cara menentukan repositori patch yang akan digunakan untuk versi sistem operasi Amazon Linux tertentu. Sampel ini menggunakan repositori sumber yang diizinkan secara default di Amazon Linux 2017.09, tetapi dapat disesuaikan dengan repositori sumber berbeda yang telah Anda konfigurasi untuk node terkelola.

> **Note**
>
> Untuk memperlihatkan perintah yang lebih kompleks ini secara lebih baik, kami menggunakan opsi `--cli-input-json` dengan opsi tambahan yang disimpan di file JSON eksternal.

1. Buat file JSON dengan nama seperti `my-patch-repository.json` dan tambahkan konten berikut ke file tersebut.

```
{
    "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. Di direktori tempat Anda menyimpan file, jalankan perintah berikut.

```
aws ssm create-patch-baseline --cli-input-json file://my-patch-repository.json
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "BaselineId": "pb-0c10e65780EXAMPLE"
}
```

Perbarui dasar patch

Perintah berikut ini menambahkan dua patch sebagai yang ditolak dan satu patch yang disetujui ke dasar patch yang ada.

> **Note**
>
> Untuk informasi tentang format yang diterima untuk daftar patch yang disetujui dan patch yang ditolak, lihat Tentang format nama paket untuk daftar patch yang disetujui dan ditolak.

Linux & macOS

```
aws ssm update-patch-baseline \
    --baseline-id pb-0c10e65780EXAMPLE \
    --rejected-patches "KB2032276" "MS10-048" \
    --approved-patches "KB2124261"
```

Server Windows

```
aws ssm update-patch-baseline ^
    --baseline-id pb-0c10e65780EXAMPLE ^
    --rejected-patches "KB2032276" "MS10-048" ^
    --approved-patches "KB2124261"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "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"
}
```

Ubah nama dasar patch

Linux & macOS

```
aws ssm update-patch-baseline \
    --baseline-id pb-0c10e65780EXAMPLE \
    --name "Windows-Server-2012-R2-Important-and-Critical-Security-Updates"
```

Server Windows

```
aws ssm update-patch-baseline ^
    --baseline-id pb-0c10e65780EXAMPLE ^
    --name "Windows-Server-2012-R2-Important-and-Critical-Security-Updates"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "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"
}
```

Hapus dasar patch

```
aws ssm delete-patch-baseline --baseline-id "pb-0c10e65780EXAMPLE"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Cantumkan semua dasar patch

```
aws ssm describe-patch-baselines
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "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"
      }
   ]
}
```

Berikut ini adalah perintah lain yang mencantumkan semua dasar patch dalam sebuah Wilayah AWS.

Linux & macOS

```
aws ssm describe-patch-baselines \
    --region us-east-2 \
    --filters "Key=OWNER,Values=[All]"
```

Server Windows

```
aws ssm describe-patch-baselines ^
    --region us-east-2 ^
    --filters "Key=OWNER,Values=[All]"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "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"
      }
   ]
}
```

Cantumkan semua dasar patch yang disediakan AWS

Linux & macOS

```
aws ssm describe-patch-baselines \
    --region us-east-2 \
    --filters "Key=OWNER,Values=[AWS]"
```

Server Windows

```
aws ssm describe-patch-baselines ^
    --region us-east-2 ^
    --filters "Key=OWNER,Values=[AWS]"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "BaselineIdentities":[
      {
         "BaselineName":"AWS-DefaultPatchBaseline",
         "DefaultBaseline":true,
         "BaselineDescription":"Default Patch Baseline Provided by AWS.",
         "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/
pb-0c10e65780EXAMPLE"
      }
   ]
}
```

Cantumkan dasar patch saya

Linux & macOS

```
aws ssm describe-patch-baselines \
    --region us-east-2 \
    --filters "Key=OWNER,Values=[Self]"
```

Server Windows

```
aws ssm describe-patch-baselines ^
    --region us-east-2 ^
    --filters "Key=OWNER,Values=[Self]"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "BaselineIdentities":[
      {
         "BaselineName":"Windows-Server-2012R2",
```

```
         "DefaultBaseline":false,
         "BaselineDescription":"Windows Server 2012 R2, Important and Critical security
 updates",
         "BaselineId":"pb-0c10e65780EXAMPLE"
      }
   ]
}
```

Tampilkan dasar patch

```
aws ssm get-patch-baseline --baseline-id pb-0c10e65780EXAMPLE
```

> **Note**
>
> Untuk dasar patch kustom, Anda dapat menentukan ID dasar patch atau Amazon Resource Name (ARN) lengkap. Untuk dasar patch yang disediakan AWS, Anda harus menentukan ARN lengkap. Sebagai contoh, `arn:aws:ssm:us-east-2:075727635805:patchbaseline/pb-0c10e65780EXAMPLE`.

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "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"
}
```

Dapatkan dasar patch default

```
aws ssm get-default-patch-baseline --region us-east-2
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/pb-0c10e65780EXAMPLE"
}
```

Atur dasar patch kustom sebagai default

Linux & macOS

```
aws ssm register-default-patch-baseline \
    --region us-east-2 \
    --baseline-id "pb-0c10e65780EXAMPLE"
```

Server Windows

```
aws ssm register-default-patch-baseline ^
    --region us-east-2 ^
    --baseline-id "pb-0c10e65780EXAMPLE"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Atur ulang dasar patch AWS sebagai default

Linux & macOS

```
aws ssm register-default-patch-baseline \
    --region us-east-2 \
    --baseline-id "arn:aws:ssm:us-east-2:123456789012:patchbaseline/
pb-0c10e65780EXAMPLE"
```

Server Windows

```
aws ssm register-default-patch-baseline ^
    --region us-east-2 ^
    --baseline-id "arn:aws:ssm:us-east-2:123456789012:patchbaseline/
pb-0c10e65780EXAMPLE"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
```

```
    "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Tandai dasar patch

Linux & macOS

```
aws ssm add-tags-to-resource \
    --resource-type "PatchBaseline" \
    --resource-id "pb-0c10e65780EXAMPLE" \
    --tags "Key=Project,Value=Testing"
```

Server Windows

```
aws ssm add-tags-to-resource ^
    --resource-type "PatchBaseline" ^
    --resource-id "pb-0c10e65780EXAMPLE" ^
    --tags "Key=Project,Value=Testing"
```

Cantumkan tag untuk dasar patch

Linux & macOS

```
aws ssm list-tags-for-resource \
    --resource-type "PatchBaseline" \
    --resource-id "pb-0c10e65780EXAMPLE"
```

Server Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "PatchBaseline" ^
    --resource-id "pb-0c10e65780EXAMPLE"
```

Hapus tag dari dasar patch

Linux & macOS

```
aws ssm remove-tags-from-resource \
    --resource-type "PatchBaseline" \
    --resource-id "pb-0c10e65780EXAMPLE" \
```

```
    --tag-keys "Project"
```

Server Windows

```
aws ssm remove-tags-from-resource ^
    --resource-type "PatchBaseline" ^
    --resource-id "pb-0c10e65780EXAMPLE" ^
    --tag-keys "Project"
```

Perintah AWS CLI untuk grup patch

Contoh perintah untuk grup patch

- Buat grup patch
- Daftarkan grup patch "server web" dengan dasar patch
- Daftarkan grup patch "Backend" dengan dasar patch yang disediakan AWS
- Tampilkan pendaftaran grup patch
- Batalkan pendaftaran grup patch dari dasar patch

Buat grup patch

Untuk membantu Anda mengatur upaya penambalan, kami sarankan Anda menambahkan node terkelola ke grup tambalan dengan menggunakan tag. Grup tambalan memerlukan penggunaan kunci tag`Patch Group`atau`PatchGroup`. Jika Anda memilikitag yang diizinkan dalam metadata instans EC2, Anda harus menggunakan`PatchGroup`(tanpa ruang). Anda dapat menentukan nilai tag apa pun, tetapi kunci tag harus`Patch Group`atau`PatchGroup`. Untuk informasi selengkapnya tentang grup patch, lihat Tentang grup patch.

Setelah mengelompokkan node terkelola menggunakan tag, Anda menambahkan nilai grup patch ke baseline patch. Dengan mendaftarkan grup patch dengan dasar patch, Anda memastikan bahwa patch yang benar diinstal selama operasi patching.

Tugas 1: Tambahkan instans EC2 ke grup patch menggunakan tag

> **Note**
>
> Saat menggunakan konsol Amazon Elastic Compute Cloud (Amazon EC2) danAWS CLI, itu mungkin untuk diterapkan`Key = Patch Group`atau`Key = PatchGroup`tag ke instance

yang belum dikonfigurasi untuk digunakan dengan Manajer Sistem. Jika instans EC2 yang Anda harapkan untuk dilihatPatch Managertidak terdaftar setelah menerapkan`Patch Group`atau`Key = PatchGroup`tag, lihatMemecahkan masalah ketersediaan node terkelolauntuk tips pemecahan masalah.

Jalankan perintah berikut ini untuk menambahkan tag `PatchGroup` ke sebuah instans EC2.

```
aws ec2 create-tags --resources "i-1234567890abcdef0" --tags
 "Key=PatchGroup,Value=GroupValue"
```

Tugas 2: Tambahkan node terkelola ke grup tambalan menggunakan tag

Jalankan perintah berikut untuk menambahkan`PatchGroup`tag ke node terkelola.

Linux & macOS

```
aws ssm add-tags-to-resource \
    --resource-type "ManagedInstance" \
    --resource-id "mi-0123456789abcdefg" \
    --tags "Key=PatchGroup,Value=GroupValue"
```

Server Windows

```
aws ssm add-tags-to-resource ^
    --resource-type "ManagedInstance" ^
    --resource-id "mi-0123456789abcdefg" ^
    --tags "Key=PatchGroup,Value=GroupValue"
```

Tugas 3: Tambahkan grup patch ke dasar patch

Jalankan perintah berikut untuk mengaitkan nilai tag `PatchGroup` ke dasar patch yang ditentukan.

Linux & macOS

```
aws ssm register-patch-baseline-for-patch-group \
    --baseline-id "pb-0c10e65780EXAMPLE" \
    --patch-group "Development"
```

Server Windows

```
aws ssm register-patch-baseline-for-patch-group ^
    --baseline-id "pb-0c10e65780EXAMPLE" ^
    --patch-group "Development"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
  "PatchGroup": "Development",
  "BaselineId": "pb-0c10e65780EXAMPLE"
}
```

Daftarkan grup patch "server web" dengan dasar patch

Linux & macOS

```
aws ssm register-patch-baseline-for-patch-group \
    --baseline-id "pb-0c10e65780EXAMPLE" \
    --patch-group "Web Servers"
```

Server Windows

```
aws ssm register-patch-baseline-for-patch-group ^
    --baseline-id "pb-0c10e65780EXAMPLE" ^
    --patch-group "Web Servers"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "PatchGroup":"Web Servers",
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Daftarkan grup patch "Backend" dengan dasar patch yang disediakan 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"
```

Server Windows

```
aws ssm register-patch-baseline-for-patch-group ^
    --region us-east-2 ^
    --baseline-id "arn:aws:ssm:us-east-2:111122223333:patchbaseline/
pb-0c10e65780EXAMPLE" ^
    --patch-group "Backend"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "PatchGroup":"Backend",
   "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/pb-0c10e65780EXAMPLE"
}
```

Tampilkan pendaftaran grup patch

```
aws ssm describe-patch-groups --region us-east-2
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "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"
         }
      }
   ]
}
```

Batalkan pendaftaran grup patch dari dasar patch

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"
```

Server Windows

```
aws ssm deregister-patch-baseline-for-patch-group ^
    --region us-east-2 ^
    --patch-group "Production" ^
    --baseline-id "arn:aws:ssm:us-east-2:111122223333:patchbaseline/
pb-0c10e65780EXAMPLE"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "PatchGroup":"Production",
   "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/pb-0c10e65780EXAMPLE"
}
```

Perintah AWS CLI untuk melihat ringkasan dan detail patch

Contoh perintah untuk melihat ringkasan dan detail patch

- Dapatkan semua patch yang didefinisikan oleh dasar patch

- Dapatkan semua tambalan untuk AmazonLinux2018.03 yang memiliki KlasifikasiSECURITYdan tingkat keparahanCritical
- Dapatkan semua patch untuk Windows Server 2012 yang memiliki kepelikan MSRC Critical
- Dapatkan semua patch yang tersedia
- Dapatkan status ringkasan tambalan per node yang dikelola
- Dapatkan detail kepatuhan tambalan untuk node terkelola
- Melihat hasil kepatuhan patching (AWS CLI)

Dapatkan semua patch yang didefinisikan oleh dasar patch

> **Note**
>
> Perintah ini hanya didukung untuk dasar patch Windows Server.

Linux & macOS

```
aws ssm describe-effective-patches-for-patch-baseline \
    --region us-east-2 \
    --baseline-id "pb-0c10e65780EXAMPLE"
```

Server Windows

```
aws ssm describe-effective-patches-for-patch-baseline ^
    --region us-east-2 ^
    --baseline-id "pb-0c10e65780EXAMPLE"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "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---
```

Dapatkan semua tambalan untuk AmazonLinux2018.03 yang memiliki Klasifikasi**SECURITY**dan tingkat keparahan**Critical**

Linux & macOS

```
aws ssm describe-available-patches \
    --region us-east-2 \
    --filters Key=PRODUCT,Values=AmazonLinux2018.03 Key=SEVERITY,Values=Critical
```

Server Windows

```
aws ssm describe-available-patches ^
    --region us-east-2 ^
    --filters Key=PRODUCT,Values=AmazonLinux2018.03 Key=SEVERITY,Values=Critical
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "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---
```

Dapatkan semua patch untuk Windows Server 2012 yang memiliki kepelikan MSRC **`Critical`**

Linux & macOS

```
aws ssm describe-available-patches \
    --region us-east-2 \
    --filters Key=PRODUCT,Values=WindowsServer2012 Key=MSRC_SEVERITY,Values=Critical
```

Server Windows

```
aws ssm describe-available-patches ^
    --region us-east-2 ^
    --filters Key=PRODUCT,Values=WindowsServer2012 Key=MSRC_SEVERITY,Values=Critical
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "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---
```

Dapatkan semua patch yang tersedia

```
aws ssm describe-available-patches --region us-east-2
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "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---
```

Dapatkan status ringkasan tambalan per node yang dikelola

Ringkasan node per terkelola memberi Anda jumlah tambalan dalam status berikut per node:”NotApplicable“, “Hilang”, “Gagal”,”InstalledOther“dan” Dipasang”.

Linux & macOS

```
aws ssm describe-instance-patch-states \
    --instance-ids i-08ee91c0b17045407 i-09a618aec652973a9
```

Server Windows

```
aws ssm describe-instance-patch-states ^
    --instance-ids i-08ee91c0b17045407 i-09a618aec652973a9
```

Sistem mengembalikan informasi seperti berikut ini.


```
{
   "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---
```

Dapatkan detail kepatuhan tambalan untuk node terkelola

```
aws ssm describe-instance-patches --instance-id i-08ee91c0b17045407
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "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---
```

Melihat hasil kepatuhan patching (AWS CLI)

Untuk melihat hasil kepatuhan tambalan untuk satu node terkelola

Jalankan perintah berikut diAWS Command Line Interface(AWS CLI) untuk melihat hasil kepatuhan tambalan untuk satu node terkelola.

```
aws ssm describe-instance-patch-states --instance-id instance-id
```

Ganti*instance-id*dengan ID dari node terkelola yang ingin Anda lihat hasilnya, dalam formati-02573cafcfEXAMPLEataumi-0282f7c436EXAMPLE.

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "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"
        }
    ]
}
```

Untuk melihat ringkasan jumlah tambalan untuk semua instans EC2 di Wilayah

`describe-instance-patch-states` mendukung mengambil hasil untuk satu instans terkelola saja dalam satu waktu. Namun, menggunakan script kustom dengan perintah `describe-instance-patch-states`, Anda dapat membuat laporan yang lebih terperinci.

Sebagai contoh, jika alat filter jq diinstal pada mesin lokal Anda, Anda dapat menjalankan perintah berikut ini untuk mengidentifikasi instans EC2 apa yang ada di Wilayah AWS tertentu yang berstatus `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 mewakili pengidentifikasi untuk sebuah Wilayah AWS yang didukung oleh AWS Systems Manager, seperti `us-east-2` untuk Region US East (Ohio). Untuk daftar yang didukung*daerah*nilai, lihatWilayahkolom diTitik akhir layanan Manajer SistemdiReferensi Umum Amazon Web Services.

Misalnya:

```
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}'
```

Sistem mengembalikan informasi seperti berikut ini.

```
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
```

Selain `InstalledPendingRebootCount`, daftar jenis jumlah yang dapat Anda cari termasuk yang berikut:

- `CriticalNonCompliantCount`
- `SecurityNonCompliantCount`
- `OtherNonCompliantCount`

- `UnreportedNotApplicableCount`
- `InstalledPendingRebootCount`
- `FailedCount`
- `NotApplicableCount`
- `InstalledRejectedCount`
- `InstalledOtherCount`
- `MissingCount`
- `InstalledCount`

AWS CLIperintah untuk memindai dan menambal node terkelola

Setelah menjalankan perintah berikut ini untuk memindai kepatuhan patch atau menginstal patch, Anda dapat menggunakan perintah di bagian Perintah AWS CLI untuk melihat ringkasan dan detail patch untuk melihat informasi tentang status dan kepatuhan patch.

Contoh perintah

- Pindai node terkelola untuk kepatuhan tambalan (AWS CLI)
- Instal tambalan pada node yang dikelola (AWS CLI)

Pindai node terkelola untuk kepatuhan tambalan (AWS CLI)

Untuk memindai node terkelola tertentu untuk kepatuhan patch

Jalankan perintah berikut.

Linux & macOS

```
aws ssm send-command \
    --document-name 'AWS-RunPatchBaseline' \
    --targets Key=InstanceIds,Values='i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE' \
    --parameters 'Operation=Scan' \
    --timeout-seconds 600
```

Server Windows

```
aws ssm send-command ^
```

```
    --document-name "AWS-RunPatchBaseline" ^
    --targets Key=InstanceIds,Values="i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE" ^
    --parameters "Operation=Scan" ^
    --timeout-seconds 600
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "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---

    }
}
```

Untuk memindai node terkelola untuk kepatuhan patch dengan tag grup patch

Jalankan perintah berikut.

Linux & macOS

```
aws ssm send-command \
    --document-name 'AWS-RunPatchBaseline' \
    --targets Key='tag:PatchGroup',Values='Web servers' \
    --parameters 'Operation=Scan' \
    --timeout-seconds 600
```

Server Windows

```
aws ssm send-command ^
    --document-name "AWS-RunPatchBaseline" ^
    --targets Key="tag:PatchGroup",Values="Web servers" ^
    --parameters "Operation=Scan" ^
    --timeout-seconds 600
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "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---

    }
}
```

Instal tambalan pada node yang dikelola (AWS CLI)

Untuk menginstal patch pada node terkelola tertentu

Jalankan perintah berikut.

> **Note**
>
> Node terkelola target akan reboot sesuai kebutuhan untuk menyelesaikan instalasi patch. Untuk informasi selengkapnya, lihat Tentang dokumen SSM `AWS-RunPatchBaseline`.

Linux & macOS

```
aws ssm send-command \
    --document-name 'AWS-RunPatchBaseline' \
    --targets Key=InstanceIds,Values='i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE' \
    --parameters 'Operation=Install' \
    --timeout-seconds 600
```

Server Windows

```
aws ssm send-command ^
    --document-name "AWS-RunPatchBaseline" ^
    --targets Key=InstanceIds,Values="i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE" ^
    --parameters "Operation=Install" ^
    --timeout-seconds 600
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "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---

    }
}
```

Untuk menginstal patch pada node terkelola dalam grup patch tertentu

Jalankan perintah berikut.

Linux & macOS

```
aws ssm send-command \
    --document-name 'AWS-RunPatchBaseline' \
    --targets Key='tag:PatchGroup',Values='Web servers' \
    -parameters 'Operation=Install' \
    --timeout-seconds 600
```

Server Windows

```
aws ssm send-command ^
```

```
    --document-name "AWS-RunPatchBaseline" ^
    --targets Key="tag:PatchGroup",Values="Web servers" ^
    --parameters "Operation=Install" ^
    --timeout-seconds 600
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
    "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---

    }
}
```

Tutorial AWS Systems ManagerPatch Manager

Panduan di bagian ini memperlihatkan cara menggunakanPatch Manager, suatu kemampuanAWS Systems Manager, untuk beberapa skenario patch.

Topik

- Tutorial: Buat baseline patch untuk menginstal Paket Layanan Windows (konsol)
- Tutorial: Perbarui dependensi aplikasi, mem-patch sebuah node terkelola, dan melakukan pemeriksaan kesehatan khusus aplikasi
- Tutorial: Menambal lingkungan server (AWS CLI)

Tutorial: Buat baseline patch untuk menginstal Paket Layanan Windows (konsol)

Ketika Anda membuat dasar patch kustom, Anda dapat menentukan bahwa semua, beberapa, atau hanya satu jenis patch yang didukung telah diinstal.

Di dasar patch untuk Windows, Anda dapat memilih `ServicePacks` sebagai satu-satunya opsi Klasifikasi untuk membatasi pembaruan patching hanya untuk Service Packs. Paket Layanan dapat diinstal secara otomatis olehPatch Manager, kemampuanAWS Systems Manager, asalkan pembaruan tersedia di Windows Update atau Windows Server Update Services (WSUS).

Anda dapat mengkonfigurasi dasar patch untuk mengendalikan apakah Service Packs untuk semua versi Windows diinstal, atau hanya untuk versi tertentu, seperti Windows 7 atau Windows Server 2016.

Gunakan prosedur berikut untuk membuat baseline patch kustom untuk digunakan secara eksklusif untuk menginstal semua Paket Layanan pada node terkelola Windows Anda.

Untuk membuat baseline patch untuk menginstal Windows Service Packs (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Patch Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.
3. Pilih tab Patch baseline, lalu pilih Create patch baseline.
4. Untuk Nama, masukkan nama untuk dasar patch baru Anda, misalnya, `MyWindowsServicePackPatchBaseline`.
5. (Opsional) Untuk Deskripsi, masukkan deskripsi untuk dasar patch ini.

6. Untuk Sistem operasi, pilih `Windows`.
7. Jika Anda ingin mulai menggunakan dasar patch ini sebagai default untuk Windows segera setelah Anda membuatnya, pilih Atur dasar patch ini sebagai dasar patch default untuk instans Windows Server.

> **Note**
>
> Opsi ini hanya tersedia jika Anda pertama kali mengakses Patch Manager sebelum kebijakan tambalan dirilis pada 22 Desember 2022.
> Untuk informasi tentang mengatur dasar patch yang ada sebagai default, lihat Mengatur dasar patch yang ada sebagai default.

8. Di bagian Aturan persetujuan untuk sistem operasi, gunakan bidang tersebut untuk membuat satu atau lebih aturan persetujuan otomatis.

   - Produk: Versi sistem operasi yang berlaku untuk aturan persetujuan, seperti`WindowsServer2012`. Anda dapat memilih satu, lebih dari satu, atau semua versi Windows yang didukung. Pilihan default adalah `All`.
   - Klasifikasi: Pilih `ServicePacks`.
   - Kepelikan: Nilai kepelikan patch yang diberlakukan aturan. Untuk memastikan bahwa semua Service Packs disertakan oleh aturan, pilih `All`.
   - Persetujuan otomatis: Metode untuk memilih patch untuk persetujuan otomatis.
     - Menyetujui patch setelah beberapa hari tertentu: Jumlah hari Patch Manager untuk menunggu setelah patch dirilis atau diperbarui sebelum patch secara otomatis disetujui. Anda dapat memasukkan bilangan bulat apa saja dari nol (0) sampai 360. Untuk sebagian besar skenario, kami merekomendasikan untuk menunggu tidak lebih dari 100 hari.
     - Menyetujui patch yang dirilis hingga tanggal tertentu: Tanggal rilis patch yang Patch Manager secara otomatis menerapkan semua patch yang dirilis atau diperbarui pada atau sebelum tanggal tersebut. Misalnya, jika Anda menentukan 7 Juli 2023, tidak ada tambalan yang dirilis atau terakhir diperbarui pada atau setelah 8 Juli 2023, yang diinstal secara otomatis.
   - (Opsional) Laporan kepatuhan: Tingkat kepelikan yang ingin Anda tetapkan untuk Service Packs yang disetujui oleh baseline, seperti `High`.

> **Note**
>
> Jika Anda menentukan tingkat pelaporan kepatuhan dan status patch dari Paket Layanan yang disetujui dilaporkan sebagai`Missing`, maka tingkat keparahan kepatuhan yang dilaporkan secara keseluruhan baseline patch adalah tingkat keparahan yang Anda tentukan.

9. (Opsional) Untuk Kelola tag, terapkan satu atau lebih pasangan nama/nilai kunci tag ke dasar patch.

   Tag adalah metadata opsional yang Anda tetapkan ke sumber daya. Tag memungkinkan Anda untuk mengkategorikan sumber daya dengan berbagai cara, seperti berdasarkan tujuan, pemilik, atau lingkungan. Untuk dasar patch ini yang didedikasikan untuk memperbarui Service Packs, Anda dapat menentukan pasangan kunci-nilai seperti berikut:

   - `Key=OS,Value=Windows`
   - `Key=Classification,Value=ServicePacks`
10. Pilih Buat dasar patch.

Tutorial: Perbarui dependensi aplikasi, mem-patch sebuah node terkelola, dan melakukan pemeriksaan kesehatan khusus aplikasi

Dalam banyak kasus, node terkelola harus di-reboot setelah di-patch dengan pembaruan perangkat lunak terbaru. Namun, me-reboot node dalam produksi tanpa adanya pengamanan dapat menyebabkan beberapa masalah, seperti memanggil alarm, merekam data metrik yang salah, dan mengganggu sinkronisasi data.

Tutorial ini memperlihatkan cara menghindari masalah seperti ini dengan menggunakanAWS Systems Manager dokumen (dokumen SSM)`AWS-RunPatchBaselineWithHooks` untuk melakukan operasi patching multi-langkah yang kompleks yang mencapai hal berikut:

1. Mencegah koneksi baru ke aplikasi
2. Menginstal pembaruan sistem operasi
3. Memperbarui dependensi paket aplikasi
4. Memulai ulang sistem
5. Melakukan pemeriksaan kesehatan khusus aplikasi

Untuk contoh ini, kami telah menyiapkan infrastruktur kami dengan cara ini:

- Mesin virtual yang ditargetkan terdaftar sebagai node terkelola dengan Systems Manager.
- `Iptables` digunakan sebagai firewall lokal.
- Aplikasi yang di-host pada node terkelola berjalan pada port 443.
- Aplikasi yang di-host pada node terkelola adalah`nodeJS` aplikasi.
- Aplikasi yang di-host pada node terkelola oleh pengelola proses pm2.
- Aplikasi ini sudah memiliki titik akhir pemeriksaan kesehatan yang ditentukan.
- Titik akhir pemeriksaan kesehatan aplikasi tidak memerlukan autentikasi pengguna akhir. Titik akhir memungkinkan dilakukannya pemeriksaan kesehatan yang memenuhi persyaratan organisasi dalam menetapkan ketersediaan. (Di lingkungan Anda, mungkin cukup untuk memastikan bahwa`nodeJS` aplikasi berjalan dan dapat mendengarkan permintaan. Dalam kasus lain, Anda mungkin juga ingin memverifikasi bahwa koneksi ke lapisan caching atau lapisan database telah dibuat.)

Contoh-contoh dalam tutorial ini adalah hanya untuk tujuan demonstrasi dan tidak dimaksudkan untuk diimplementasikan apa adanya ke dalam lingkungan produksi. Selain itu, perlu diingat bahwa fitur kait siklus hidup dariPatch Manager, suatu kemampuan Systems Manager, dengan`AWS-RunPatchBaselineWithHooks` dokumen dapat mendukung banyak skenario lainnya. Berikut adalah beberapa contoh tanda.

- Menghentikan agen pelaporan metrik sebelum patching dan memulai ulang setelah node terkelola di-reboot.
- Melepaskan node terkelola dari klaster CRM atau PCS sebelum patching dan melampirkannya kembali setelah node di-reboot.
- Perbarui perangkat lunak pihak ketiga (misalnya, aplikasi Java, Tomcat, Adobe, dan sebagainya) padaWindows Server mesin setelah pembaruan sistem operasi (OS) diterapkan, tetapi sebelum node terkelola di-reboot.

Untuk memperbarui dependensi aplikasi, mem-patch sebuah node terkelola, dan melakukan pemeriksaan kesehatan khusus aplikasi

1. Buat dokumen SSM untuk script pra-instalasi Anda dengan konten berikut ini dan berikan nama `NodeJSAppPrePatch`. Ganti *`your_application`* dengan nama aplikasi Anda.

Script ini segera memblokir permintaan masuk baru dan menyediakan lima detik untuk permintaan yang sudah aktif agar diselesaikan sebelum memulai operasi patching. Untuk opsi `sleep`, tentukan jumlah detik lebih besar daripada yang biasanya diperlukan untuk permintaan masuk diselesaikan.

```
# 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
```

Untuk informasi tentang membuat dokumen SSM, lihat Membuat konten dokumen SSM.

2. Buat dokumen SSM lain dengan konten berikut ini untuk script pasca instalasi Anda untuk memperbarui dependensi aplikasi Anda dan berikan nama `NodeJSAppPostPatch`. Ganti */your/application/path* dengan jalur ke aplikasi Anda.

```
cd /your/application/path
npm update
# you can use npm-check-updates if you want to upgrade major versions
```

3. Buat dokumen SSM lain dengan konten berikut ini untuk script `onExit` Anda untuk menyalakan kembali aplikasi Anda dan melakukan pemeriksaan kesehatan. Namakan dokumen SSM ini `NodeJSApp0nExitPatch`. Ganti *your_application* dengan nama aplikasi Anda.

```
# 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. Buat asosiasi diState Manager, suatu kemampuanAWS Systems Manager, untuk meluncurkan operasi dengan melakukan langkah-langkah berikut:

   1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
   2. Di panel navigasi, pilih State Manager, lalu pilih Buat asosiasi.
   3. Untuk Nama, berikan nama untuk membantu mengidentifikasi tujuan asosiasi.
   4. Di daftar Dokumen, pilih `AWS-RunPatchBaselineWithHooks`.
   5. Untuk Operasi, pilih Instal.
   6. (Opsional) Untuk Snapshot Id, berikan GUID yang Anda buat untuk membantu mempercepat operasi dan memastikan konsistensi. Nilai GUID dapat sesederhana `00000000-0000-0000-0000-111122223333`.
   7. Untuk Pre Install Hook Doc Name, masukkan `NodeJSAppPrePatch`.
   8. Untuk Post Install Hook Doc Name, masukkan `NodeJSAppPostPatch`.
   9. Untuk On ExitHook Doc Name, masukkan`NodeJSApp0nExitPatch`.

5. Untuk Target, identifikasi node terkelola Anda dengan menentukan tag, memilih node secara manual, memilih resource group, atau memilih semua node terkelola.
6. Untuk Tentukan jadwal, tentukan seberapa sering untuk menjalankan asosiasi. Untuk patching node terkelola, sekali per minggu adalah irama yang umum.
7. Di bagian Pengendalian rate, pilih opsi untuk mengontrol bagaimana asosiasi berjalan pada beberapa node terkelola. Pastikan bahwa hanya sebagian dari node terkelola yang diperbarui pada satu waktu. Jika tidak, semua atau sebagian besar armada Anda dapat dijadikan offline sekaligus. Untuk informasi lebih lanjut tentang menggunakan kontrol rate, lihat Tentang target dan kontrol tingkat dalam State Manager asosiasi.
8. (Opsional) Untuk Pilihan output, untuk menyimpan output perintah ke file, pilih kotak Aktifkan output penulisan ke S3. Masukkan nama bucket dan prefiks (folder) di dalam kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah dari profil instans yang ditetapkan ke node terkelola, bukan data pengguna IAM yang melaksanakan tugas ini. Untuk informasi lebih lanjut, lihat Mengkonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berada dalam yang berbedaAkun AWS,

verifikasi bahwa profil instans atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

9. Pilih Buat Asosiasi.

Tutorial: Menambal lingkungan server (AWS CLI)

Prosedur berikut ini menjelaskan cara melakukan patching untuk lingkungan server dengan menggunakan dasar patch kustom, grup patch, dan jendela pemeliharaan.

Sebelum Anda memulai

- Instal atau perbaruiSSM Agentpada node terkelola Anda. Untuk menambal node yang dikelola Linux, node Anda harus berjalanSSM Agentversi 2.0.834.0 atau yang lebih baru. Untuk informasi selengkapnya, lihat Memperbarui SSM Agent penggunaan Run Command.
- Konfigurasikan peran dan izin untukMaintenance WindowsSebuah kemampuan dariAWS Systems Manager. Untuk informasi selengkapnya, lihat Menyiapkan Maintenance Windows.
- Instal dan konfigurasikanAWS Command Line Interface(AWS CLI), jika Anda belum melakukannya.

  Untuk informasi, lihatMenginstal atau memperbarui versi terbaruAWS CLI.

Untuk mengkonfigurasiPatch Managerdan menambal node yang dikelola (baris perintah)

1. Jalankan perintah berikut ini untuk membuat dasar patch untuk Windows yang bernama `Production-Baseline`. Garis dasar tambalan ini menyetujui tambalan untuk lingkungan produksi 7 hari setelah dirilis atau terakhir diperbarui. Artinya, kami menandai dasar patch untuk menunjukkan bahwa itu untuk lingkungan produksi.

   > **Note**
   >
   > The`OperatingSystem`parameter dan`PatchFilters`bervariasi tergantung pada sistem operasi node terkelola target yang berlaku untuk patch baseline. Untuk informasi selengkapnya, lihat OperatingSystem dan 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"
```

Server Windows

```
aws ssm create-patch-baseline ^
    --name "Production-Baseline" ^
    --operating-system "WINDOWS" ^
    --tags "Key=Environment,Value=Production" ^
    --approval-rules
 "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=MSRC_SEVERITY,Values=[Critical,Impor
{Key=CLASSIFICATION,Values=[SecurityUpdates,Updates,ServicePacks,UpdateRollups,CriticalU
 ^
    --description "Baseline containing all updates approved for production
 systems"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

2. Jalankan perintah berikut ini untuk mendaftarkan dasar patch "Production-Baseline" untuk dua grup patch. Grup tersebut bernama "Database Servers" dan "Front-End Servers".

   Linux & macOS

   ```
   aws ssm register-patch-baseline-for-patch-group \
       --baseline-id pb-0c10e65780EXAMPLE \
       --patch-group "Database Servers"
   ```

   Server Windows

   ```
   aws ssm register-patch-baseline-for-patch-group ^
   ```

```
    --baseline-id pb-0c10e65780EXAMPLE ^
    --patch-group "Database Servers"
```

Sistem mengembalikan informasi seperti berikut.

```
{
   "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"
```

Server Windows

```
aws ssm register-patch-baseline-for-patch-group ^
    --baseline-id pb-0c10e65780EXAMPLE ^
    --patch-group "Front-End Servers"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "PatchGroup":"Front-End Servers",
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

3. Jalankan perintah berikut ini untuk membuat dua jendela pemeliharaan untuk server produksi. Jendela pertama berjalan setiap hari Selasa pukul 10 malam. Jendela kedua berjalan setiap hari Sabtu pukul 10 malam. Selain itu, jendela pemeliharaan telah ditandai untuk menunjukkan bahwa itu untuk lingkungan produksi.

   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
```

Server Windows

```
aws ssm create-maintenance-window ^
    --name "Production-Tuesdays" ^
    --tags "Key=Environment,Value=Production" ^
    --schedule "cron(0 0 22 ? * TUE *)" ^
    --duration 1 ^
    --cutoff 0 ^
    --no-allow-unassociated-targets
```

Sistem mengembalikan informasi seperti berikut.

```
{
   "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
```

Server Windows

```
aws ssm create-maintenance-window ^
    --name "Production-Saturdays" ^
    --tags "Key=Environment,Value=Production" ^
    --schedule "cron(0 0 22 ? * SAT *)" ^
    --duration 2 ^
    --cutoff 0 ^
    --no-allow-unassociated-targets
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "WindowId":"mw-9a8b7c6d5eEXAMPLE"
}
```

4. Jalankan perintah berikut ini untuk mendaftarkan grup patch server `Database` dan `Front-End` dengan jendela pemeliharaan masing-masing.

   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"
   ```

   Server Windows

   ```
   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"
   ```

   Sistem mengembalikan informasi seperti berikut.

   ```
   {
      "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"
   ```

Server Windows

```
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"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "WindowTargetId":"faa01c41-1d57-496c-ba77-ff9caEXAMPLE"
}
```

5. Jalankan perintah berikut ini untuk mendaftarkan sebuah tugas patch yang menginstal pembaruan yang hilang pada server `Database` dan `Front-End` selama jendela pemeliharaan masing-masing.

   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}}"
```

   Server Windows

```
aws ssm register-task-with-maintenance-window ^
    --window-id mw-0c50858d01EXAMPLE ^
    --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
 ^
    --task-arn "AWS-RunPatchBaseline" ^
    --service-role-arn "arn:aws:iam::123456789012:role/MW-Role" ^
```

```
    --task-type "RUN_COMMAND" ^
    --max-concurrency 2 ^
    --max-errors 1 ^
    --priority 1 ^
    --task-invocation-parameters "RunCommand={Parameters={Operation=Install}}"
```

Sistem mengembalikan informasi seperti berikut.

```
{
   "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}}"
```

Server Windows

```
aws ssm register-task-with-maintenance-window ^
    --window-id mw-9a8b7c6d5eEXAMPLE ^
    --targets "Key=WindowTargetIds,Values=faa01c41-1d57-496c-ba77-ff9caEXAMPLE"
 ^
    --task-arn "AWS-RunPatchBaseline" ^
    --service-role-arn "arn:aws:iam::123456789012:role/MW-Role" ^
    --task-type "RUN_COMMAND" ^
    --max-concurrency 2 ^
    --max-errors 1 ^
    --priority 1 ^
    --task-invocation-parameters "RunCommand={Parameters={Operation=Install}}"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "WindowTaskId":"8a5c4629-31b0-4edd-8aea-33698EXAMPLE"
}
```

6. Jalankan perintah berikut ini untuk mendapatkan ringkasan kepatuhan patch tingkat tinggi untuk grup patch. Ringkasan kepatuhan patch tingkat tinggi mencakup jumlah node terkelola dengan tambalan di masing-masing status patch.

> **Note**
>
> Diharapkan melihat nol untuk jumlah node terkelola dalam ringkasan hingga tugas tambalan berjalan selama jendela pemeliharaan pertama.

Linux & macOS

```
aws ssm describe-patch-group-state \
    --patch-group "Database Servers"
```

Server Windows

```
aws ssm describe-patch-group-state ^
    --patch-group "Database Servers"
```

Sistem mengembalikan informasi seperti berikut ini.

```
{
   "Instances": number,
   "InstancesWithFailedPatches": number,
   "InstancesWithInstalledOtherPatches": number,
   "InstancesWithInstalledPatches": number,
   "InstancesWithInstalledPendingRebootPatches": number,
   "InstancesWithInstalledRejectedPatches": number,
   "InstancesWithMissingPatches": number,
   "InstancesWithNotApplicablePatches": number,
   "InstancesWithUnreportedNotApplicablePatches": number
```

```
}
```

7. Jalankan perintah berikut untuk mendapatkan status ringkasan tambalan per node yang dikelola untuk grup tambalan. Ringkasan node per terkelola mencakup sejumlah tambalan di masing-masing status patch per node terkelola untuk grup patch.

   Linux & macOS

   ```
   aws ssm describe-instance-patch-states-for-patch-group \
       --patch-group "Database Servers"
   ```

   Server Windows

   ```
   aws ssm describe-instance-patch-states-for-patch-group ^
       --patch-group "Database Servers"
   ```

   Sistem mengembalikan informasi seperti berikut ini.

   ```
   {
      "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
         }
   ```

```
    ]
}
```

Untuk contoh lainnyaAWS CLIperintah yang dapat Anda gunakan untukPatch Managertugas konfigurasi, lihatBekerja denganPatch Manager(AWS CLI).

Pemecahan Masalah Patch Manager

Gunakan informasi berikut untuk membantu Anda memecahkan masalahPatch ManagerSebuah kemampuan dariAWS Systems Manager.

Topik

- Masalah: “Memohon-PatchBaselineOperation : Akses Ditolak” kesalahan atau kesalahan “Tidak dapat mengunduh file dari S3" untukbaseline_overrides.json
- Masalah: Penambalan gagal tanpa penyebab atau pesan kesalahan yang jelas
- Masalah: Hasil kepatuhan tambalan yang tidak terduga
- Kesalahan saat menjalankan AWS-RunPatchBaseline pada Linux
- Kesalahan saat menjalankan AWS-RunPatchBaseline pada Windows Server
- Menghubungi AWS Support

Masalah: “Memohon-PatchBaselineOperation : Akses Ditolak” kesalahan atau kesalahan “Tidak dapat mengunduh file dari S3" untuk**`baseline_overrides.json`**

Masalah: Saat operasi penambalan yang ditentukan oleh kebijakan tambalan Anda berjalan, Anda menerima kesalahan yang mirip dengan contoh berikut.

Example error on Server Windows

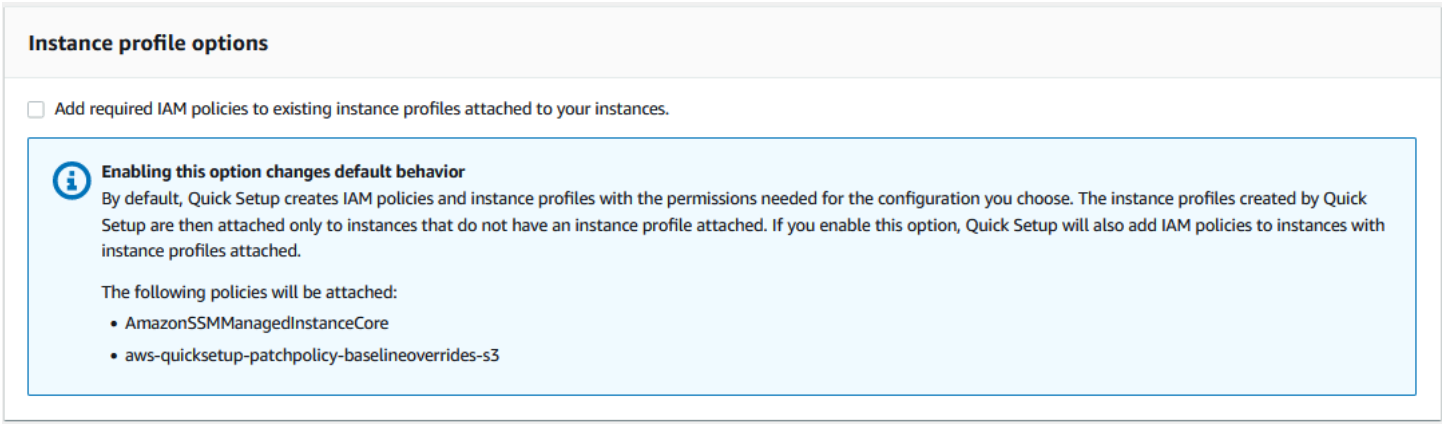
```
----------ERROR-------
Invoke-PatchBaselineOperation : Access Denied
At C:\ProgramData\Amazon\SSM\InstanceData\i-02573cafcfEXAMPLE\document\orchestr
ation\792dd5bd-2ad3-4f1e-931d-abEXAMPLE\PatchWindows\_script.ps1:219 char:13
+ $response = Invoke-PatchBaselineOperation -Operation Install -Snapsho ...
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
+ 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.
```

Menyebabkan: Anda membuat kebijakan tambalan diQuick Setup, dan beberapa node terkelola Anda sudah memiliki profil instance yang terpasang (untuk instans EC2) atau peran layanan yang terpasang (untuk mesin non-EC2). Namun, Anda tidak memilihTambahkan kebijakan IAM yang diperlukan ke profil instans yang ada yang dilampirkan ke instans Andakotak centang, seperti yang ditunjukkan pada gambar berikut.



Saat Anda membuat kebijakan tambalan, bucket Amazon S3 juga dibuat untuk menyimpan konfigurasi kebijakan`baseline_overrides.json`berkas. Jika Anda tidak memilihTambahkan kebijakan IAM yang diperlukan ke profil instans yang ada yang dilampirkan ke instans Andakotak centang saat membuat kebijakan, kebijakan IAM, dan tag sumber daya yang diperlukan untuk mengakses`baseline_overrides.json`di bucket S3 tidak secara otomatis ditambahkan ke profil instans IAM dan peran layanan Anda yang ada.

Solusi 1: Hapus konfigurasi kebijakan tambalan yang ada, lalu buat pengganti, pastikan untuk memilihTambahkan kebijakan IAM yang diperlukan ke profil instans yang ada yang dilampirkan ke instans Andakotak centang. Pilihan ini menerapkan kebijakan IAM yang dibuat oleh iniQuick Setupkonfigurasi ke node yang sudah memiliki profil instance atau peran layanan terlampir.

(Secara default,Quick Setupmenambahkan kebijakan yang diperlukan ke instance dan node yang melakukannyatidaksudah memiliki profil instance atau peran layanan.) Untuk informasi lebih lanjut, lihatOtomatiskan penambalan di seluruh organisasi menggunakanQuick Setupkebijakan tambalan.

Solusi 2: Tambahkan izin dan tag yang diperlukan secara manual ke setiap profil instans IAM dan peran layanan IAM yang Anda gunakanQuick Setup. Untuk petunjuk, lihat Izin untuk bucket S3 kebijakan patch.

Masalah: Penambalan gagal tanpa penyebab atau pesan kesalahan yang jelas

Masalah: Operasi patching gagal tanpa mengembalikan pesan kesalahan.

Kemungkinan penyebabnya: Jika lebih dari satu doa`AWS-RunPatchBaseline`terjadi pada suatu waktu, mereka dapat bertentangan satu sama lain, menyebabkan tugas patching gagal. Ini mungkin tidak ditunjukkan dalam log penambalan.

Untuk memeriksa apakah operasi patching bersamaan mungkin telah saling mengganggu, tinjau riwayat perintah diRun CommandSebuah kemampuan dariAWS Systems Manager. Untuk node terkelola dengan kegagalan tambalan, periksa untuk melihat apakah beberapa operasi mencoba menambal mesin dalam waktu 2 menit satu sama lain. Skenario ini terkadang dapat menyebabkan kegagalan.

Anda juga dapat menggunakanAWS Command Line Interface(AWS CLI) untuk memeriksa upaya patching bersamaan dengan menggunakan perintah berikut. Ganti nilai untuk*node-id*dengan ID untuk node terkelola Anda.

```
aws ssm list-commands \
    --filter "key=DocumentName,value=AWS-RunPatchBaseline" \
    --query 'Commands[*].
{CommandId:CommandId,RequestedDateTime:RequestedDateTime,Status:Status}' \
    --instance-id node-id \
    --output table
```

Solusi: Jika Anda menentukan bahwa patching gagal karena operasi patching yang bersaing pada node terkelola yang sama, sesuaikan konfigurasi patching Anda untuk menghindari hal ini terjadi lagi. Misalnya, jika dua jendela pemeliharaan menentukan waktu penambalan yang tumpang tindih, hapus atau revisi salah satunya. Jika jendela pemeliharaan menentukan satu operasi penambalan, tetapi kebijakan tambalan menentukan yang berbeda untuk waktu yang sama, pertimbangkan untuk menghapus tugas dari jendela pemeliharaan.

Jika Anda menentukan bahwa operasi patching yang bertentangan bukanlah penyebab kegagalan dalam skenario ini, kami sarankan untuk menghubungiAWS Support.

Masalah: Hasil kepatuhan tambalan yang tidak terduga

Masalah: Saat meninjau detail kepatuhan tambalan yang dihasilkan setelah`Scan`operasi, hasilnya mencakup informasi yang tidak mencerminkan aturan yang diatur di baseline patch Anda. Misalnya, pengecualian yang Anda tambahkan keTambalan yang ditolakdaftar di baseline patch terdaftar sebagai`Missing`. Atau tambalan diklasifikasikan sebagai`Important`terdaftar sebagai hilang meskipun baseline patch Anda menentukan`Critical`tambalan saja.

Menyebabkan:Patch Managersaat ini mendukung beberapa metode menjalankan`Scan`operasi:

- Kebijakan tambalan yang dikonfigurasi diQuick Setup
- Opsi Manajemen Host yang dikonfigurasi diQuick Setup
- Jendela pemeliharaan untuk menjalankan tambalan`Scan`atau`Install`tugas
- Sesuai permintaanPatch sekarangoperasi

Ketika`Scan`operasi berjalan, itu menimpa rincian kepatuhan dari pemindaian terbaru. Jika Anda memiliki lebih dari satu metode yang disiapkan untuk menjalankan`Scan`operasi, dan mereka menggunakan baseline patch yang berbeda dengan aturan yang berbeda, mereka akan menghasilkan hasil kepatuhan patch yang berbeda.

Solusi: Untuk menghindari hasil kepatuhan patch yang tidak terduga, kami sarankan hanya menggunakan satu metode pada satu waktu untuk menjalankanPatch Manager `Scan`operasi. Untuk informasi selengkapnya, lihat Menghindari penimpaan data kepatuhan patch yang tidak disengaja.

Kesalahan saat menjalankan **`AWS-RunPatchBaseline`** pada Linux

Topik

- Masalah: Kesalahan 'Tidak ada file atau direktori tersebut”
- Masalah: kesalahan 'proses lain telah mengakuisisi yum lock'
- Masalah: kesalahan 'Izin ditolak / gagal menjalankan perintah'
- Masalah: kesalahan 'Tidak dapat mengunduh muatan'
- Masalah: kesalahan 'kombinasi pengelola paket dan versi python yang tidak didukung’
- Masalah:Patch Managertidak menerapkan aturan yang ditentukan untuk mengecualikan paket tertentu

- Masalah: Penambalan gagal danPatch Managermelaporkan bahwa ekstensi Indikasi Nama Server ke TLS tidak tersedia
- Masalah:Patch Managerlaporan 'Tidak ada lagi cermin untuk dicoba'
- Masalah: Penambalan gagal dengan 'Kode kesalahan yang dikembalikan dari curl adalah 23'
- Masalah: Penambalan gagal dengan pesan 'Kesalahan membongkar paket rpm... '
- Masalah: Penambalan gagal dengan pesan 'Kesalahan ditemui saat mengunduh paket'
- Masalah: Penambalan gagal dengan pesan bahwa 'Tanda tangan berikut tidak dapat diverifikasi karena kunci publik tidak tersedia'
- Masalah: Penambalan gagal dengan 'NoMoreMirrorsRepoError'pesan
- Masalah: Penambalan gagal dengan pesan 'Tidak dapat mengunduh payload'
- Masalah: Penambalan gagal dengan pesan 'kesalahan instal: dpkg: kesalahan: frontend dpkg dikunci oleh proses lain'
- Masalah: MenambalUbuntu Servergagal dengan kesalahan 'dpkg terganggu'
- Masalah: Utilitas manajer paket tidak dapat menyelesaikan ketergantungan paket

Masalah: Kesalahan 'Tidak ada file atau direktori tersebut”

Masalah: Ketika Anda menjalankan `AWS-RunPatchBaseline`, patching gagal dengan salah satu kesalahan berikut.

```
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
```

Penyebab 1: Dua perintah untuk dijalankan`AWS-RunPatchBaseline`berjalan pada saat yang sama pada node terkelola yang sama. Hal ini menciptakan kondisi balapan yang menyebabkan `file patch-baseline-operations*` sementara tidak dibuat atau diakses dengan benar.

Penyebab 2: Ruang penyimpanan yang tersisa tidak cukup dalam direktori `/var`.

Solusi 1: Pastikan tidak ada jendela pemeliharaan yang memiliki dua atau lebihRun Commandtugas yang berjalan`AWS-RunPatchBaseline`dengan tingkat Prioritas yang sama dan berjalan pada ID

target yang sama. Jika ini masalahnya, susun ulang prioritasnya.Run Commandadalah kemampuan dariAWS Systems Manager.

Solusi 2: Pastikan hanya satu jendela pemeliharaan pada satu waktu yang berjalanRun Commandtugas yang menggunakan`AWS-RunPatchBaseline`pada target yang sama dan pada jadwal yang sama. Jika demikian, ubah jadwalnya.

Solusi 3Pastikan bahwa hanya satuState ManagerAsosiasi sedang berjalan`AWS-RunPatchBaseline`pada jadwal yang sama dan menargetkan node terkelola yang sama.State Manageradalah kemampuan dariAWS Systems Manager.

Solusi 4: Bebaskan ruang penyimpanan yang cukup dalam direktori `/var` untuk paket pembaruan.

Masalah: kesalahan 'proses lain telah mengakuisisi yum lock'

Masalah: Ketika Anda menjalankan `AWS-RunPatchBaseline`, patching gagal dengan kesalahan berikut.

```
12/20/2019 21:41:48 root [INFO]: another process has acquired yum lock, waiting 2 s and
 retry.
```

Menyebabkan:`AWS-RunPatchBaseline`dokumen telah mulai berjalan pada node terkelola di mana ia sudah berjalan di operasi lain dan telah memperoleh manajer paketyumproses.

Solusi: Pastikan tidakState Managerasosiasi, tugas jendela pemeliharaan, atau konfigurasi lain yang berjalan`AWS-RunPatchBaseline`pada jadwal menargetkan node terkelola yang sama sekitar waktu yang sama.

Masalah: kesalahan 'Izin ditolak / gagal menjalankan perintah'

Masalah: Ketika Anda menjalankan `AWS-RunPatchBaseline`, patching gagal dengan kesalahan berikut.

```
sh:
/var/lib/amazon/ssm/instanceid/document/orchestration/commandid/PatchLinux/_script.sh:
 Permission denied
failed to run commands: exit status 126
```

Penyebab: `/var/lib/amazon/` mungkin dipasang dengan izin `noexec`. Ini menjadi masalah karena`SSM Agent`mengunduh skrip payload ke`/var/lib/amazon/ssm`dan menjalankannya dari lokasi itu.

Solusi: Pastikan Anda telah mengonfigurasi partisi eksklusif`/var/log/amazon`dan`/var/lib/amazon`, dan bahwa mereka dipasang dengan`exec`izin.

Masalah: kesalahan 'Tidak dapat mengunduh muatan'

Masalah: Ketika Anda menjalankan `AWS-RunPatchBaseline`, patching gagal dengan kesalahan berikut.

```
Unable to download payload: https://s3.DOC-EXAMPLE-BUCKET.region.amazonaws.com/
aws-ssm-region/patchbaselineoperations/linux/payloads/patch-baseline-operations-
X.XX.tar.gz.failed to run commands: exit status 156
```

Menyebabkan: Node terkelola tidak memiliki izin yang diperlukan untuk mengakses bucket Amazon Simple Storage Service (Amazon S3) yang ditentukan.

Solusi: Perbarui konfigurasi jaringan Anda sehingga titik akhir S3 dapat dijangkau. Untuk detail selengkapnya, lihat informasi tentang akses yang diperlukan ke bucket S3Patch Managerdi[SSM Agentkomunikasi dengan bucket S3 AWS terkelola](#).

Masalah: kesalahan 'kombinasi pengelola paket dan versi python yang tidak didukung’

Masalah: Ketika Anda menjalankan `AWS-RunPatchBaseline`, patching gagal dengan kesalahan berikut.

```
An unsupported package manager and python version combination was found. Apt requires
 Python3 to be installed.
failed to run commands: exit status 1
```

Menyebabkan: Versi python3 yang didukung tidak diinstal padaDebian Server,Raspberry Pi OS, atauUbuntu Servercontoh.

Solusi: Instal versi python3 yang didukung (3.0 - 3.10) di server, yang diperlukan untukDebian Server,Raspberry Pi OS, danUbuntu Servernode terkelola.

Masalah:Patch Managertidak menerapkan aturan yang ditentukan untuk mengecualikan paket tertentu

Masalah: Anda telah mencoba untuk mengecualikan paket tertentu dengan menentukannya di`/etc/yum.conf`file, dalam format`exclude=`*`package-name`*, tetapi mereka tidak dikecualikan selamaPatch Manager `Install`operasi.

Menyebabkan:Patch Managertidak memasukkan pengecualian yang ditentukan dalam`/etc/yum.conf`berkas.

Solusi: Untuk mengecualikan paket tertentu, buat dasar patch kustom dan buat aturan untuk mengecualikan paket yang tidak ingin diinstal.

Masalah: Penambalan gagal danPatch Managermelaporkan bahwa ekstensi Indikasi Nama Server ke TLS tidak tersedia

Masalah: Operasi patching mengeluarkan pesan berikut ini.

```
/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
```

Penyebab: Pesan ini tidak menunjukkan kesalahan. Sebaliknya, ini adalah peringatan bahwa versi lama Python yang didistribusikan dengan sistem operasi tidak mendukung Server Name Indication (SNI) TLS. Script muatan patch Systems Manager mengeluarkan peringatan ini saat membuat koneksi ke API AWS yang mendukung SNI.

Solusi: Untuk memecahkan masalah kegagalan patch ketika pesan ini dilaporkan, tinjau konten file `stdout` dan `stderr`. Jika Anda belum mengonfigurasi baseline patch untuk menyimpan file-file ini di bucket S3 atau di Amazon CloudWatch Log, Anda dapat menemukan file di lokasi berikut pada node terkelola Linux Anda.

`/var/lib/amazon/ssm/`*`instance-id`*`/document/orchestration/`*`Run-Command-execution-id`*`/awsrunShellScript/PatchLinux`

Masalah:Patch Managerlaporan 'Tidak ada lagi cermin untuk dicoba'

Masalah: Operasi patching mengeluarkan pesan berikut ini.

```
[Errno 256] No more mirrors to try.
```

Menyebabkan: Repositori yang dikonfigurasi pada node terkelola tidak berfungsi dengan benar. Kemungkinan penyebab untuk hal ini meliputi:

- Cache `yum` rusak.
- URL repositori tidak dapat dijangkau karena masalah terkait jaringan.

Solusi:Patch Managermenggunakan manajer paket default node terkelola untuk melakukan operasi patching. Periksa kembali bahwa repositori dikonfigurasi dan beroperasi dengan benar.

Masalah: Penambalan gagal dengan 'Kode kesalahan yang dikembalikan dari curl adalah 23'

Masalah: Operasi patching yang menggunakan`AWS-RunPatchBaseline`gagal dengan kesalahan yang mirip dengan berikut ini:

```
05/01/2023 17:04:30 root [ERROR]: Error code returned from curl is 23
```

Menyebabkan: Alat curl yang digunakan pada sistem Anda tidak memiliki izin yang diperlukan untuk menulis ke sistem file. Ini dapat terjadi ketika jika alat curl default manajer paket digantikan oleh versi yang berbeda, seperti yang diinstal dengan snap.

Solusi: Jika versi curl yang disediakan oleh manajer paket dihapus saat versi lain diinstal, instal ulang.

Jika Anda perlu menyimpan beberapa versi curl diinstal, pastikan bahwa versi yang terkait dengan manajer paket ada di direktori pertama yang terdaftar di`PATH`variabel. Anda dapat memeriksa ini dengan menjalankan perintah`echo $PATH`untuk melihat urutan direktori saat ini yang diperiksa untuk file yang dapat dieksekusi pada sistem Anda.

Masalah: Penambalan gagal dengan pesan 'Kesalahan membongkar paket rpm... '

Masalah: Operasi penambalan gagal dengan kesalahan yang mirip dengan berikut ini:

```
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
```

Penyebab 1: Ketika paket tertentu hadir di beberapa installer paket, seperti keduanya`apt`dan`yum`atau`dnf`, konflik dapat terjadi saat menggunakan manajer paket default.

Contoh umum terjadi dengan`urllib3`paket, yang ditemukan di`pip`,`yum`, dan`dnf`.

Penyebab 2:`python-urllib3`paket rusak. Ini dapat terjadi jika file paket diinstal atau diperbarui oleh`pip`setelah`rpm`adalah paket sebelumnya diinstal oleh`yum`atau`dnf`.

Solusi: Hapus`python-urllib3`paket dari pip dengan menjalankan perintah`sudo pip uninstall urllib3`, menyimpan paket hanya di manajer paket default (`yum`atau`dnf`).

Masalah: Penambalan gagal dengan pesan 'Kesalahan ditemui saat mengunduh paket'

Masalah: Selama menambal, Anda menerima kesalahan yang mirip dengan yang berikut ini:

```
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',
```

Menyebabkan: Kesalahan ini dapat terjadi ketika memori tidak cukup tersedia pada node terkelola.

Solusi: Konfigurasikan memori swap, atau tingkatkan instance ke jenis yang berbeda untuk meningkatkan dukungan memori. Kemudian mulailah operasi penambalan baru.

Masalah: Penambalan gagal dengan pesan bahwa 'Tanda tangan berikut tidak dapat diverifikasi karena kunci publik tidak tersedia'

Masalah: Penambalan gagalUbuntu Serverdengan kesalahan yang mirip dengan berikut ini:

```
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
```

MenyebabkanKunci GNU Privacy Guard (GPG) telah kedaluwarsa atau hilang.

Solusi: Segarkan kembali tombol GPG, atau tambahkan kunci lagi.

Misalnya, menggunakan kesalahan yang ditunjukkan sebelumnya, kita melihat bahwa`467B942D3A79BD29`kunci hilang dan harus ditambahkan. Untuk melakukannya, jalankan salah satu dari perintah berikut:

```
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
```

Atau, untuk menyegarkan semua tombol, jalankan perintah berikut:

```
sudo apt-key adv --keyserver hkps://keyserver.ubuntu.com --refresh-keys
```

Jika kesalahan berulang setelah ini, kami sarankan untuk melaporkan masalah ke organisasi yang memelihara repositori. Sampai perbaikan tersedia, Anda dapat mengedit`/etc/apt/sources.list`file untuk menghilangkan repositori selama proses patching.

Untuk melakukannya, buka`sources.list`file untuk mengedit, menemukan baris untuk repositori, dan menyisipkan#karakter di awal baris untuk mengomentarinya. Kemudian simpan dan tutup file.

Masalah: Penambalan gagal dengan 'NoMoreMirrorsRepoError'pesan

Masalah: Anda menerima kesalahan yang mirip dengan berikut ini:

```
NoMoreMirrorsRepoError: failure: repodata/repomd.xml from pgdg94: [Errno 256] No more
 mirrors to try.
```

Menyebabkan: Ada kesalahan dalam repositori sumber.

Solusi: Kami merekomendasikan untuk melaporkan masalah ini ke organisasi yang memelihara repositori. Sampai kesalahan diperbaiki, Anda dapat menonaktifkan repositori di tingkat sistem operasi. Untuk melakukannya, jalankan perintah berikut, ganti nilai untuk*repo-nama*dengan nama repositori Anda:

```
yum-config-manager --disable repo-name
```

Berikut adalah contohnya.

```
yum-config-manager --disable pgdg94
```

Setelah Anda menjalankan perintah ini, jalankan operasi patching lain.

Masalah: Penambalan gagal dengan pesan 'Tidak dapat mengunduh payload'

Masalah: Anda menerima kesalahan yang mirip dengan berikut ini:

```
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
```

Menyebabkan: Konfigurasi node terkelola berisi kesalahan atau tidak lengkap.

Solusi: Pastikan node terkelola dikonfigurasi dengan yang berikut:

- Aturan TCP 443 keluar dalam grup keamanan.
- Aturan keluar TCP 443 di NACL.
- Aturan Ingress TCP 1024-65535 di NACL.
- NAT/IGW dalam tabel rute untuk menyediakan konektivitas ke titik akhir S3. Jika instans tidak memiliki akses internet, berikan konektivitas dengan titik akhir S3. Untuk melakukan itu, tambahkan titik akhir gateway S3 di VPC dan integrasikan dengan tabel rute node terkelola.

Masalah: Penambalan gagal dengan pesan 'kesalahan instal: dpkg: kesalahan: frontend dpkg dikunci oleh proses lain'

Masalah: Patching gagal dengan kesalahan yang mirip dengan berikut ini:

```
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
```

Menyebabkan: Manajer paket sudah menjalankan proses lain pada node terkelola di tingkat sistem operasi. Jika proses lain itu membutuhkan waktu lama untuk diselesaikan,Patch Manageroperasi penambalan dapat habis waktu dan gagal.

Solusi: Setelah proses lain yang menggunakan manajer paket selesai, jalankan operasi penambalan baru.

Masalah: MenambalUbuntu Servergagal dengan kesalahan 'dpkg terganggu'

Masalah: PadaUbuntu Server, penambalan gagal dengan kesalahan yang mirip dengan yang berikut ini:

```
E: dpkg was interrupted, you must manually run
'dpkg --configure -a' to correct the problem.
```

Menyebabkan: Satu atau lebih paket salah dikonfigurasi.

Solusi: Lakukan langkah-langkah berikut:

1. Periksa untuk melihat paket mana yang terpengaruh, dan apa masalahnya dengan setiap paket dengan menjalankan perintah berikut, satu per satu:

```
sudo apt-get check
```

```
sudo dpkg -C
```

```
dpkg-query -W -f='${db:Status-Abbrev} ${binary:Package}\n' | grep -E ^.[^nci]
```

2. Perbaiki paket dengan masalah dengan menjalankan perintah berikut:

```
sudo dpkg --configure -a
```

3. Jika perintah sebelumnya tidak sepenuhnya menyelesaikan masalah, jalankan perintah berikut:

```
sudo apt --fix-broken install
```

Masalah: Utilitas manajer paket tidak dapat menyelesaikan ketergantungan paket

Masalah: Manajer paket asli pada node terkelola tidak dapat menyelesaikan ketergantungan paket dan tambalan gagal. Contoh pesan kesalahan berikut menunjukkan jenis kegagalan pada sistem operasi yang menggunakan`yum`sebagai manajer paket.

```
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']
```

Menyebabkan: Pada sistem operasi Linux,Patch Managermenggunakan manajer paket asli pada mesin untuk menjalankan operasi penambalan. sepertiyum,`dnf`,`apt`, dan`zypper`. Aplikasi secara otomatis mendeteksi, menginstal, memperbarui, atau menghapus paket dependen sesuai kebutuhan. Namun, beberapa kondisi dapat mengakibatkan manajer paket tidak dapat menyelesaikan operasi ketergantungan, seperti:

- Beberapa repositori yang saling bertentangan dikonfigurasi pada sistem operasi.

- URL repositori jarak jauh tidak dapat diakses karena masalah terkait jaringan.
- Paket untuk arsitektur yang salah ditemukan di repositori.

Solusi: Patching mungkin gagal karena masalah ketergantungan karena berbagai alasan. Oleh karena itu, kami menyarankan Anda untuk menghubungiAWS Supportuntuk membantu pemecahan masalah.

Kesalahan saat menjalankan **`AWS-RunPatchBaseline`** pada Windows Server

Topik

Masalah: pasangan keluarga produk/produk yang tidak cocok

Masalah: Ketika Anda membuat dasar patch di konsol Systems Manager, Anda menentukan keluarga produk dan produk. Sebagai contoh, Anda dapat memilih:

- Keluarga produk: `Office`

  Produk: `Office 2016`

Penyebab: Jika Anda mencoba untuk membuat dasar patch dengan pasangan keluarga produk/produk yang tidak cocok, sebuah pesan kesalahan akan ditampilkan. Berikut ini adalah beberapa alasan mengapa hal ini terjadi:

- Anda memilih pasangan keluarga produk dan produk yang valid tetapi kemudian menghapus pilihan keluarga produk.
- Anda memilih produk dari sub-daftar Pilihan usang atau tidak cocok bukan dari sub-daftar Pilihan yang tersedia dan cocok.

  Item dalam sub-daftar produk Pilihan usang atau tidak cocok mungkin telah salah dimasukkan melalui SDK atau perintah AWS Command Line Interface (AWS CLI) `create-patch-baseline`.

Ini bisa berarti adanya kesalahan pengetikan atau produk ditugaskan ke keluarga produk yang salah. Sebuah produk juga disertakan dalam sub-daftar Pilihan usang atau tidak cocok jika ditentukan untuk dasar patch sebelumnya tetapi tidak memiliki patch yang tersedia dari Microsoft.

Solusi: Untuk menghindari masalah ini di konsol, selalu pilih opsi dari sub-daftar Pilihan yang tersedia saat ini.

Anda juga dapat melihat produk yang memiliki patch yang tersedia dengan menggunakan perintah `describe-patch-properties` di AWS CLI atau perintah API `DescribePatchProperties`.

Masalah: Output **`AWS-RunPatchBaseline`** mengembalikan sebuah **`HRESULT`** (Windows Server)

Masalah: Anda menerima kesalahan seperti berikut ini.

```
----------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
```

Penyebab: Output ini menunjukkan bahwa API native Windows Update tidak dapat menjalankan operasi patching.

Solusi: Periksa`HResult`kode dalam topik microsoft.com berikut untuk mengidentifikasi langkah-langkah pemecahan masalah untuk menyelesaikan kesalahan:

- Kode kesalahan Pembaruan Windows berdasarkan komponen
- Windows Update kesalahan umum dan mitigasi

Masalah: node terkelola tidak memiliki akses ke Katalog Pembaruan Windows atau WSUS

Masalah: Anda menerima kesalahan seperti berikut ini.

```
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----
```

Penyebab: Kesalahan ini dapat berhubungan dengan komponen Windows Update, atau tidak adanya konektivitas ke Katalog Windows Update atau Windows Server Update Services (WSUS).

Solusi: Konfirmasikan bahwa node yang dikelola memiliki konektivitas keKatalog Pembaruan Microsoftmelalui gateway internet, gateway NAT, atau instance NAT. Jika Anda menggunakan WSUS, konfirmasikan bahwa node terkelola memiliki konektivitas ke server WSUS di lingkungan Anda. Jika konektivitas tersedia untuk tujuan yang dimaksudkan, periksa dokumentasi Microsoft untuk potensi penyebab `HResult 0x80072EE2`. Ini mungkin menunjukkan masalah tingkat sistem operasi.

Masalah: PatchBaselineOperations PowerShell modul tidak dapat diunduh

Masalah: Anda menerima pesan kesalahan seperti berikut ini.

```
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
```

Solusi: Periksa konektivitas node terkelola dan izin ke Amazon Simple Storage Service (Amazon S3). Node terkelolaAWS Identity and Access Management(IAM) peran harus menggunakan izin

minimum yang dikutip dalamSSM Agentkomunikasi dengan bucket S3 AWS terkelola. Node harus berkomunikasi dengan titik akhir Amazon S3 melalui titik akhir gateway Amazon S3, gateway NAT, atau gateway internet. Untuk informasi selengkapnya tentang persyaratan VPC Endpoint untukAWS Systems Manager SSM Agent(SSM Agent), lihatLangkah 2: Buat titik akhir VPC.

Masalah: patch yang hilang

Masalah: `AWS-RunPatchbaseline` berhasil diselesaikan, tetapi ada beberapa patch yang hilang.

Berikut ini adalah beberapa penyebab umum dan solusinya.

Penyebab 1: Baseline tidak efektif.

Solusi 1: Untuk memeriksa apakah ini penyebabnya, gunakan prosedur berikut.

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command.

   -atau-

   JikaAWS Systems Managerhalaman rumah terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilihRun Command.
3. Pilih tab Riwayat perintah lalu pilih perintah yang baseline-nya ingin Anda periksa.
4. Pilih node terkelola yang memiliki tambalan yang hilang.
5. Pilih Langkah 1 - Output dan temukan nilai `BaselineId`.
6. Periksa konfigurasi dasar patch yang ditetapkan, yaitu, sistem operasi, nama produk, klasifikasi, dan kepelikan untuk dasar patch.
7. Buka Katalog Microsoft Update.
8. Cari ID artikel Microsoft Knowledge Base (KB) (misalnya, KB3216916).
9. Verifikasi bahwa nilai di bawahProdukcocok dengan node terkelola Anda dan pilih yang sesuaiJudul. Jendela Detail Pembaruan baru akan terbuka.
10. Di tab Gambaran Umum, klasifikasi dan Kepelikan MSRC harus cocok dengan konfigurasi dasar patch yang Anda temukan sebelumnya.

Penyebab 2: patch diganti.

Solusi 2: Untuk memeriksa apakah ini benar, gunakan prosedur berikut.

1. Buka Katalog Microsoft Update.
2. Cari ID artikel Microsoft Knowledge Base (KB) (misalnya, KB3216916).
3. Verifikasi bahwa nilai di bawahProdukcocok dengan node terkelola Anda dan pilih yang sesuaiJudul. Jendela Detail Pembaruan baru akan terbuka.
4. Buka tab Detail paket. Cari entri di bawah tajuk Pembaruan ini telah digantikan oleh pembaruan berikut:.

Penyebab 3: patch yang sama mungkin memiliki nomor KB yang berbeda karena pembaruan online WSUS dan Windows ditangani Release Channels berbeda oleh Microsoft.

Solusi 3: Periksa kelayakan patch. Jika paket tidak tersedia di bawah WSUS, instal OS Build 14393.3115. Jika paket tersedia untuk semua build sistem operasi, instal OS Build 18362.1256 dan 18363.1256.

Menghubungi AWS Support

Jika Anda tidak dapat menemukan solusi pemecahan masalah di bagian iniatau dalam masalah Manajer Sistem diAWSRe: posting, dan Anda memilikiPengembang, Bisnis, atau PerusahaanAWS Supportrencana, Anda dapat membuat kasus dukungan teknis diAWS Support.

Sebelum Anda menghubungi AWS Support, kumpulkan item berikut:

- Log agen SSM
- Run CommandID perintah, ID jendela pemeliharaan, atau ID eksekusi Otomasi
- UntukWindows Servernode terkelola, juga kumpulkan yang berikut ini:
  - `%PROGRAMDATA%\Amazon\PatchBaselineOperations\Logs` seperti yang dijelaskan pada tab Windows pada Cara menginstal patch
  - Log pembaruan Windows: Untuk Windows Server 2012 R2 dan yang sebelumnya, gunakan `%windir%/WindowsUpdate.log`. UntukWindows Server2016 dan yang lebih baru, pertama jalankan PowerShell komando`Get-WindowsUpdateLog`sebelum menggunakan`%windir%/WindowsUpdate.log`
- Untuk node yang dikelola Linux, kumpulkan juga yang berikut ini:
  - Isi direktori`/var/lib/amazon/ssm/`*`instance-id`*`/document/orchestration/`*`Run-Command-execution-id`*`/awsrunShellScript/PatchLinux`

AWS Systems Manager Distributor

Distributor, kemampuan AWS Systems Manager, membantu Anda mengemas dan mempublikasikan perangkat lunak ke node yang AWS Systems Manager dikelola. Anda dapat mengemas dan mempublikasikan perangkat lunak Anda sendiri atau menggunakannya Distributor untuk menemukan dan menerbitkan paket perangkat lunak agen yang AWS disediakan, seperti AmazonCloudWatchAgent, atau paket pihak ketiga seperti Trend Micro.Menerbitkan paket mengiklankan versi tertentu dari dokumen paket ke node terkelola yang Anda identifikasi menggunakan ID node, Akun AWS ID, tag, atau file Wilayah AWS. Untuk memulaiDistributor, buka konsol Systems Manager. Di panel navigasi, pilih Distributor.

Setelah Anda membuat paketDistributor, Anda dapat menginstal paket dengan salah satu cara berikut:

- Satu kali dengan menggunakan AWS Systems Manager Run Command
- Pada jadwal dengan menggunakan AWS Systems Manager State Manager

> **Important**
>
> Paket yang didistribusikan oleh penjual pihak ketiga tidak dikelola oleh AWS dan dipublikasikan oleh vendor paket. Kami mendorong Anda untuk melakukan uji tuntas tambahan untuk memastikan kepatuhan terhadap kontrol keamanan internal Anda. Keamanan adalah tanggung jawab bersama antara Anda AWS dan Anda. Hal ini digambarkan sebagai model tanggung jawab bersama. Untuk mempelajari selengkapnya, lihat model tanggung jawab bersama.

Bagaimana bisa Distributor menguntungkan organisasi saya?

Distributormenawarkan manfaat ini:

- Satu paket, banyak platform

  Saat Anda membuat paketDistributor, sistem membuat AWS Systems Manager dokumen (dokumen SSM). Anda dapat melampirkan file.zip ke dokumen ini. Ketika Anda menjalankanDistributor, sistem memproses instruksi dalam dokumen SSM dan menginstal paket perangkat lunak dalam file.zip pada target yang ditentukan. Distributormendukung beberapa sistem

operasi, termasuk Windows,Ubuntu Server,Debian Server, danRed Hat Enterprise Linux. Untuk informasi selengkapnya tentang platform yang didukung, lihat Platform dan arsitektur paket yang didukung.

- Kontrol akses paket di seluruh grup instance terkelola

  Anda dapat menggunakan Run Command atau State Manager mengontrol node terkelola mana yang mendapatkan paket dan versi paket mana. Run Commanddan State Manager merupakan kemampuan dari AWS Systems Manager. Node terkelola dapat dikelompokkan berdasarkan contoh atau ID perangkat, Akun AWS angka, tag, atau Wilayah AWS. Anda dapat menggunakan State Manager asosiasi untuk mengirimkan versi paket yang berbeda ke grup instance yang berbeda.

- Banyak paket AWS agen disertakan dan siap digunakan

  Distributortermasuk banyak paket AWS agen yang siap untuk Anda terapkan ke node terkelola. Cari paket di halaman Distributor `Packages` daftar yang diterbitkan oleh`Amazon`. Contohnya termasuk `AmazonCloudWatchAgent` dan `AWSPVDriver`.

- Mengotomatiskan penerapan

  Untuk menjaga lingkungan Anda tetap terkini, gunakan State Manager untuk menjadwalkan paket untuk penerapan otomatis pada node terkelola target saat mesin tersebut pertama kali diluncurkan.

Siapa yang harus menggunakanDistributor?

- Setiap AWS pelanggan yang ingin membuat baru atau menyebarkan paket perangkat lunak yang ada, termasuk paket yang AWS diterbitkan, ke beberapa node yang dikelola Systems Manager sekaligus.
- Pengembang perangkat lunak yang membuat paket perangkat lunak.
- Administrator yang bertanggung jawab untuk menjaga node terkelola Systems Manager saat ini dengan sebagian besar paket up-to-date perangkat lunak.

Apa saja fitur-fiturnyaDistributor?

- Penyebaran paket ke instance Windows dan Linux

  DenganDistributor, Anda dapat menerapkan paket perangkat lunak ke AWS IoT Greengrass instans Elastic Compute Cloud (Amazon EC2) dan perangkat inti Amazon Elastic Compute Cloud

(Amazon EC2) untuk Linux dan perangkat inti. Windows Server Untuk daftar jenis sistem operasi instand yang didukung, lihat the section called “Platform dan arsitektur paket yang didukung”.

> **Note**
>
> Distributortidak didukung pada sistem macOS operasi.

- Menyebarkan paket satu kali, atau pada jadwal otomatis

  Anda dapat memilih untuk men-deploy paket satu kali, pada jadwal reguler, atau kapan pun versi paket default diubah ke versi yang berbeda.

- Instal ulang paket sepenuhnya, atau lakukan pembaruan di tempat

  Untuk menginstal versi paket baru, Anda dapat sepenuhnya menghapus versi saat ini dan menginstal yang baru di tempatnya, atau hanya memperbarui versi saat ini dengan komponen baru dan diperbarui, menurut skrip pembaruan yang Anda berikan. Aplikasi paket Anda tidak tersedia selama instalasi ulang, tetapi dapat tetap tersedia selama pembaruan di tempat. Pembaruan di tempat sangat berguna untuk aplikasi pemantauan keamanan atau skenario lain di mana Anda perlu menghindari downtime aplikasi.

- Konsol, CLI PowerShell, dan akses SDK ke kemampuan Distributor

  Anda dapat bekerja Distributor dengan menggunakan konsol Systems Manager, AWS Command Line Interface (AWS CLI) AWS Tools for PowerShell, atau AWS SDK pilihan Anda.

- Kontrol akses IAM

  Dengan menggunakan kebijakan AWS Identity and Access Management (IAM), Anda dapat mengontrol anggota organisasi mana yang dapat membuat, memperbarui, menyebarkan, atau menghapus paket atau versi paket. Misalnya, Anda mungkin ingin memberikan izin administrator untuk men-deploy paket, tetapi tidak untuk mengubah paket atau membuat versi paket baru.

- Dukungan kemampuan logging dan audit

  Anda dapat mengaudit dan mencatat tindakan Distributor pengguna di dalam Anda Akun AWS melalui integrasi dengan yang lain Layanan AWS. Untuk informasi selengkapnya, lihat Audit dan loggingDistributoraktivitas.

Apa itu paket?

Paket adalah kumpulan perangkat lunak atau aset yang dapat diinstal yang mencakup hal-hal berikut.

- Sebuah file .zip perangkat lunak per platform sistem operasi target. Setiap file .zip harus mencakup hal-hal berikut.
  - Sebuah install dan uninstall naskah. Windows Servernode terkelola berbasis memerlukan PowerShell skrip (skrip bernama `install.ps1` dan`uninstall.ps1`). Node terkelola berbasis Linux memerlukan skrip shell (skrip bernama dan). `install.sh` `uninstall.sh` AWS Systems Manager SSM Agentmembaca dan melaksanakan instruksi dalam install dan uninstall skrip.
  - File yang dapat dieksekusi. SSM Agentharus menemukan executable ini untuk menginstal paket pada node terkelola target.
- File manifes berformat JSON yang menjelaskan isi paket. Manifes tidak termasuk dalam file .zip, tetapi disimpan dalam bucket Amazon Simple Storage Service (Amazon S3) yang sama sebagai file .zip yang membentuk paket. Manifes mengidentifikasi versi paket dan memetakan file.zip dalam paket untuk menargetkan atribut node terkelola, seperti versi atau arsitektur sistem operasi. Untuk informasi tentang cara membuat manifes, lihat Langkah 2: Buat manifes paket JSON.

Saat Anda memilih pembuatan paket sederhana di Distributor konsol, Distributor buat skrip instalasi dan penghapusan instalasi, hash file, dan manifes paket JSON untuk Anda, berdasarkan nama file yang dapat dieksekusi perangkat lunak dan platform serta arsitektur target.

Platform dan arsitektur paket yang didukung

Anda dapat menggunakan Distributor untuk mempublikasikan paket ke platform node terkelola Systems Manager berikut. Nilai versi harus sesuai dengan versi rilis yang tepat Amazon Machine Image (AMI) sistem operasi yang Anda targetkan. Untuk informasi selengkapnya tentang cara menentukan versi ini, lihat langkah 4 Langkah 2: Buat manifes paket JSON.

> **Note**
>
> Systems Manager tidak mendukung semua sistem operasi berikut untuk perangkat AWS IoT Greengrass inti. Untuk informasi selengkapnya, lihat Menyiapkan perangkat AWS IoT Greengrass inti di Panduan AWS IoT Greengrass Version 2 Pengembang.

Platform	Nilai kode dalam file manifest	Arsitektur
Windows Server	`windows`	`x86_64` atau `386`
Debian Server	`debian`	`x86_64` atau `386`
Ubuntu Server	`ubuntu`	`x86_64` atau `386` `arm64`(Ubuntu Server16 dan yang lebih baru, jenis instans A1)
Red Hat Enterprise Linux (RHEL)	`redhat`	`x86_64` atau `386` `arm64` (RHEL 7.6 dan yang lebih baru, tipe instans A1)
Centos	`centos`	`x86_64` atau `386`
Amazon Linux 1, Amazon Linux 2, dan Amazon Linux 2023	`amazon`	`x86_64` atau `386` `arm64`(Amazon Linux 2 dan AL2023, tipe instans A1)
SUSE Linux Enterprise Server (SLES)	`suse`	`x86_64` atau `386`
openSUSE	`opensuse`	`x86_64` atau `386`
openSUSE Leap	`opensuseleap`	`x86_64` atau `386`
Oracle Linux	`oracle`	`x86_64`

Topik

- Menyiapkan Distributor
- Bekerja dengan Distributor
- Audit dan loggingDistributoraktivitas
- Pemecahan Masalah AWS Systems ManagerDistributor

Menyiapkan Distributor

Sebelum Anda menggunakanDistributor, kemampuanAWS Systems Manager, untuk membuat, mengelola, dan men-deploy paket perangkat lunak, ikuti langkah-langkah berikut.

Topik

Langkah 1: LengkapDistributorprasyarat

Sebelum Anda menggunakanDistributor, kemampuanAWS Systems Manager, pastikan lingkungan Anda memenuhi persyaratan berikut.

Prasyarat Distributor

Persyaratan	Deskripsi
SSM Agent	AWS Systems Manager SSM Agentversi 2.3.274.0 atau yang lebih baru harus diinstal di node yang dikelola tempat Anda ingin menyebarkan atau yang Anda ingin menghapus paket. Untuk menginstal atau memperbaruiSSM Agent, LihatBekerja dengan SSM Agent.
AWS CLI	(Opsional) Untuk menggunakan AWS Command Line Interface (AWS CLI) bukannya konsol Systems Manager untuk membuat dan mengelola paket, instal rilis terbaru AWS CLI di komputer lokal Anda. Untuk informasi tentang cara menginstal atau meningkatkan CLI, lihat Menginstal AWS

Persyaratan	Deskripsi
	Command Line Interface dalam Panduan Pengguna AWS Command Line Interface.
AWS Tools for PowerShell	(Opsional) Untuk menggunakan Alat untuk PowerShell bukannya konsol Systems Manager untuk membuat dan mengelola paket, instal rilis terbaru Tools for PowerShell di komputer lokal Anda. Untuk informasi selengkapnya tentang cara menginstal atau meningkatkan Tools for PowerShell, LihatMenyiapkanAWS Tools for Windows PowerShellatauAWS Tools for PowerShell Coredi dalamAWS Tools for Windows PowerShellPanduan Pengguna.

> **Note**
>
> Systems Manager tidak mendukung pendistribusian paket keOracle Linuxnode yang dikelola menggunakanDistributor.

Langkah 2: Verifikasi atau buat profil instans IAM denganDistributor izin

Secara default, AWS Systems Manager tidak memiliki izin untuk melakukan tindakan pada instans Anda. Anda harus memberikan akses dengan menggunakan profil instans AWS Identity and Access Management (IAM). Profil instans adalah kontainer yang menyampaikan informasi IAM role ke instans Amazon Elastic Compute Cloud (Amazon EC2) saat peluncuran. Persyaratan ini berlaku pada izin untuk semua kemampuan Systems Manager, bukan hanyaDistributor, yang merupakan kemampuanAWS Systems Manager.

> **Note**
>
> Ketika Anda mengkonfigurasi perangkat edge Anda untuk menjalankan perangkat lunakAWS IoT Greengrass Core danSSM Agent, Anda menentukan peran layanan IAM yang

memungkinkan Systems Manager untuk melakukan tindakan di atasnya. Anda tidak perlu mengonfigurasi perangkat tepi terkelola dengan profil instans.

Jika Anda sudah menggunakan kemampuan Systems Manager lainnyaState Manager, sepertiRun Command dan, profil instans dengan izin yang diperlukan untuk instans Anda.Distributor Cara termudah untuk memastikan Anda memiliki izin untuk melakukanDistributor tugas adalah melampirkan ManagedInstanceCoreAmazonssmsmsmagedssmsmsmsmsmsmsmsmsmsmsmsmsmsmsmsmsmsms Untuk informasi lebih lanjut, lihat Mengkonfigurasi izin instans untuk Systems Manager.

Langkah 3: Mengontrol akses pengguna ke paket

Dengan menggunakan kebijakan (IAM) AWS Identity and Access Management, Anda dapat mengontrol siapa yang dapat membuat, men-deploy, dan mengelola paket. Anda juga mengontrol operasi State Manager API Run Command dan mana yang dapat mereka lakukan di node terkelola. SepertiDistributor, keduanya Run Command danState Manager, adalah kemampuanAWS Systems Manager.

Format ARN

Paket yang ditetapkan pengguna terkait dengan dokumen Amazon Resource Names (ARN) dan memiliki format berikut.

```
arn:aws:ssm:region:account-id:document/document-name
```

Berikut adalah contohnya.

```
arn:aws:ssm:us-west-1:123456789012:document/ExampleDocumentName
```

Anda dapat menggunakan sepasang kebijakan IAM default yang AWS disediakan, satu untuk pengguna akhir dan satu untuk administrator, untuk memberikan izin bagi Distributor kegiatan. Atau Anda dapat membuat kebijakan IAM kustom yang sesuai dengan persyaratan izin Anda.

Untuk informasi selengkapnya tentang penggunaan variabel dalam kebijakan IAM, lihat Elemen Kebijakan IAM: Variabel.

Untuk informasi tentang cara membuat kebijakan dan melampirkannya ke pengguna atau grup, lihat Membuat Kebijakan IAM dan Menambahkan dan Menghapus Kebijakan IAM di Panduan Pengguna IAM.

Langkah 4: Buat atau pilih bucket Amazon S3

Saat Anda membuat paket dengan menggunakanSederhanaalur kerja diAWS Systems Managerkonsol, Anda memilih bucket Amazon Simple Storage Service (Amazon S3) yang adaDistributorupload perangkat lunak Anda.Distributoradalah suatu kemampuanAWS Systems Manager. Pada alur kerja Lanjutan, Anda harus mengunggah file .zip perangkat lunak atau aset Anda ke bucket Amazon S3 sebelum Anda mulai. Apakah Anda membuat paket dengan menggunakan alur kerja Sederhana atau Lanjutan di konsol, atau dengan menggunakan API, Anda harus memiliki bucket Amazon S3 sebelum Anda mulai membuat paket Anda. Sebagai bagian dari proses pembuatan paket,Distributormenyalin perangkat lunak dan aset yang dapat diinstal dari bucket ini ke toko Systems Manager internal. Karena aset disalin ke toko internal, Anda dapat menghapus atau menggunakan kembali bucket Amazon S3 Anda ketika pembuatan paket selesai.

Untuk informasi selengkapnya tentang cara membuat bucket, lihat Membuat bucket di Panduan Memulai Amazon Simple Storage Service. Untuk informasi selengkapnya tentang cara menjalankan perintah AWS CLI untuk membuat bucket, lihat mb di Referensi Perintah AWS CLI.

Bekerja dengan Distributor

Anda dapat menggunakanAWS Systems Managerkonsol,AWSalat baris perintah baris perintah (AWS CLIdanAWS Tools for PowerShell), danAWSSDK untuk menambah, mengelola, atau menyebarkan paket diDistributor.Distributoradalah kemampuanAWS Systems Manager. Sebelum Anda menambahkan paket keDistributor:

- Buat dan zip aset yang dapat diinstal.
- (Opsional) Buat file manifest JSON untuk paket. Ini tidak diperlukan untuk menggunakanSederhanaproses pembuatan paket diDistributorkonsol. Pembuatan paket sederhana menghasilkan file manifes JSON untuk Anda.

  Anda dapat menggunakan konsol AWS Systems Manager atau teks atau editor JSON untuk membuat file manifes.
- Siapkan bucket Amazon Simple Storage Service (Amazon S3) untuk menyimpan aset atau perangkat lunak yang dapat diinstal Anda. Jika Anda menggunakan proses pembuatan paket Lanjutan, unggah aset Anda ke bucket Amazon S3 sebelum Anda mulai.

> **Note**
>
> Anda dapat menghapus atau menggunakan kembali bucket ini setelah Anda selesai membuat paket Anda karenaDistributormemindahkan isi paket ke bucket Systems Manager internal sebagai bagian dari proses pembuatan paket.

Paket AWS yang dipublikasikan sudah dikemas dan siap untuk deployment. Untuk menggunakanAWS-published package ke node terkelola, lihatMenginstal atau memperbarui paket.

Anda dapat berbagiDistributorpaket antaraAkun AWS. Saat menggunakan paket yang dibagikan dari akun lain diAWS CLIperintah menggunakan paket Amazon Resource Name (ARN) alih-alih nama paket.

Topik

- Lihat paket
- Buat paket
- Mengedit izin paket (konsol)
- Mengedit tanda paket (konsol)
- Menambahkan versi paketDistributor
- Menginstal atau memperbarui paket
- Menghapus instalasi paket.
- Menghapus paket

Lihat paket

Untuk melihat paket yang tersedia untuk instalasi, Anda dapat menggunakanAWS Systems Managerkonsol atau pilihan AndaAWSalat baris perintah.Distributoradalah kemampuanAWS Systems Manager. Untuk mengaksesDistributor, membukaAWS Systems Managerkonsol dan pilihDistributorDi panel navigasi kiri. Anda akan melihat semua paket yang tersedia untuk Anda.

Bagian berikut menjelaskan bagaimana Anda dapat melihatDistributorpaket yang menggunakan alat baris perintah pilihan Anda.

Lihat paket (baris perintah)

Bagian ini berisi informasi tentang bagaimana Anda dapat menggunakan alat baris perintah pilihan Anda untuk melihatDistributorpaket menggunakan perintah yang disediakan.

Linux & macOS

Untuk melihat paket yang menggunakan AWS CLI di Linux

- Untuk melihat semua paket, tidak termasuk paket bersama, jalankan perintah berikut.

```
aws ssm list-documents \
    --filters Key=DocumentType,Values=Package
```

- Untuk melihat semua paket yang dimiliki Amazon, jalankan perintah berikut.

```
aws ssm list-documents \
    --filters Key=DocumentType,Values=Package Key=Owner,Values=Amazon
```

- Untuk melihat semua paket yang dimiliki pihak ketiga, jalankan perintah berikut.

```
aws ssm list-documents \
    --filters Key=DocumentType,Values=Package Key=Owner,Values=ThirdParty
```

Windows

Untuk melihat paket yang menggunakan AWS CLI di Windows

- Untuk melihat semua paket, tidak termasuk paket bersama, jalankan perintah berikut.

```
aws ssm list-documents ^
    --filters Key=DocumentType,Values=Package
```

- Untuk melihat semua paket yang dimiliki Amazon, jalankan perintah berikut.

```
aws ssm list-documents ^
    --filters Key=DocumentType,Values=Package Key=Owner,Values=Amazon
```

- Untuk melihat semua paket yang dimiliki pihak ketiga, jalankan perintah berikut.

```
aws ssm list-documents ^
```

```
    --filters Key=DocumentType,Values=Package Key=Owner,Values=ThirdParty
```

PowerShell

Untuk melihat paket yang menggunakan Tools for PowerShell

- Untuk melihat semua paket, tidak termasuk paket bersama, jalankan perintah berikut.

```
$filter = New-Object Amazon.SimpleSystemsManagement.Model.DocumentKeyValuesFilter
$filter.Key = "DocumentType"
$filter.Values = "Package"

Get-SSMDocumentList `
    -Filters @($filter)
```

- Untuk melihat semua paket yang dimiliki Amazon, jalankan perintah berikut.

```
$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)
```

- Untuk melihat semua paket yang dimiliki pihak ketiga, jalankan perintah berikut.

```
$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)
```

Buat paket

Untuk membuat paket, siapkan perangkat lunak atau aset yang dapat diinstal Anda, satu file per platform sistem operasi. Setidaknya satu file diperlukan untuk membuat paket.

Platform yang berbeda terkadang menggunakan file yang sama, namun semua file yang Anda lampirkan ke paket harus tercantum dalam bagian `Files` manifes. Jika Anda membuat paket dengan menggunakan alur kerja sederhana di konsol, manifes akan dibuat untuk Anda. Jumlah maksimum file yang dapat Anda lampirkan ke dokumen tunggal adalah 20. Ukuran maksimum setiap file adalah 1 GB. Untuk informasi selengkapnya tentang platform yang didukung, lihat Platform dan arsitektur paket yang didukung.

Saat membuat paket, sistem membuat dokumen SSM baru. Dokumen ini memungkinkan Anda untuk men-deploy paket ke node terkelola.

Untuk tujuan demonstrasi saja, sebuah contoh paket, ExamplePackage.zip, tersedia untuk Anda unduh dari situs web kami. Contoh paket menyertakan manifes JSON yang telah selesai dan tiga file. zip yang berisi penginstal untuk v7.0.0. PowerShell Skrip instalasi dan penghapusan instalasi tidak berisi perintah yang valid. Meskipun Anda harus men-zip setiap perangkat lunak yang dapat diinstal dan skrip ke file .zip untuk membuat paket di alur kerja Lanjutan, Anda tidak men-zip aset yang dapat diinstal di alur kerja Sederhana.

Topik

- Membuat sebuah paket (sederhana)
- Membuat paket (lanjutan)

Membuat sebuah paket (sederhana)

Bagian ini menjelaskan cara membuat paket Distributor dengan memilih alur kerja pembuatan paket Sederhana di Distributor konsol. Distributoradalah kemampuanAWS Systems Manager. Untuk membuat paket, siapkan aset yang dapat diinstal Anda, satu file per platform sistem operasi. Setidaknya satu file diperlukan untuk membuat paket. Proses pembuatan paket Sederhana menghasilkan skrip instalasi dan penghapusan instalasi, hash file, dan manifes yang diformat JSON untuk Anda. Alur kerja Sederhana menangani proses unggah dan pen-zip-an file yang dapat

diinstal, serta membuat paket baru dan Dokumen SSM terkait. Untuk informasi selengkapnya tentang platform yang didukung, lihat Platform dan arsitektur paket yang didukung.

Bila Anda menggunakan metode Sederhana untuk membuat paket, Distributor membuat `install` dan `uninstall` skrip untuk Anda. Namun, ketika Anda membuat paket untuk pembaruan di tempat, Anda harus menyediakan konten skrip `update` Anda sendiri pada tab Memperbarui skrip. Ketika Anda menambahkan perintah input untuk `update` skrip, Distributor sertakan skrip ini dalam paket .zip yang dibuat untuk Anda, bersama dengan `uninstall` skrip `install` dan skrip.

> **Note**
>
> Gunakan opsi pembaruan `In-place` untuk menambahkan file baru atau yang diperbarui ke instalasi paket yang ada tanpa mengambil aplikasi terkait secara offline.

Untuk membuat paket (sederhana)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Distributor.
3. Pada halaman Distributor utama, pilih Buat paket, lalu pilih Sederhana.
4. Pada halaman Buat paket, masukkan nama untuk paket Anda. Nama hanya dapat berisi huruf, angka, titik, tanda hubung, dan garis bawah. Nama harus cukup generik untuk diterapkan ke semua versi lampiran paket, tetapi cukup spesifik untuk mengidentifikasi tujuan paket.
5. (Opsional) Untuk Nama versi, masukkan nama versi. Nama versi dapat maksimal 512 karakter, dan tidak dapat berisi karakter khusus.
6. Untuk Lokasi, pilih bucket dengan menggunakan nama bucket dan awalan atau dengan menggunakan URL bucket.
7. Untuk Unggah perangkat lunak, pilih Tambahkan perangkat lunak, lalu pergi ke file perangkat lunak yang dapat diinstal dengan ekstensi `.rpm`, `.msi`, atau `.deb`. Jika nama file berisi spasi, pengunggahan gagal. Anda dapat mengunggah lebih dari satu file perangkat lunak dalam satu tindakan.
8. Untuk Platform target, verifikasi bahwa platform sistem operasi target yang ditampilkan untuk setiap file yang dapat diinstal benar. Jika sistem operasi yang ditampilkan tidak benar, pilih sistem operasi yang benar dari daftar dropdown.

   Untuk alur kerja pembuatan paket Sederhana, karena Anda mengunggah setiap file yang dapat diinstal hanya sekali, langkah tambahan diperlukan untuk memerintahkan Distributor

untuk menargetkan satu file di beberapa sistem operasi. Misalnya, jika Anda mengunggah file perangkat lunak yang dapat diinstal bernama `Logtool_v1.1.1.rpm`, Anda harus mengubah beberapa default di alur kerja Sederhana untuk menargetkan perangkat lunak yang sama di sistem operasi Amazon Linux dan Ubuntu. Saat menargetkan beberapa platform, lakukan salah satu hal berikut ini.

- Gunakan alur kerja Lanjutan sebagai gantinya, zip setiap file yang dapat diinstal ke dalam file .zip sebelum Anda memulai, dan secara manual tulis manifest sehingga satu file yang dapat diinstal dapat ditargetkan pada beberapa versi atau platform sistem operasi. Untuk informasi selengkapnya, lihat Membuat paket (lanjutan).
- Edit file manifest secara manual di alur kerja Sederhana sehingga file .zip Anda ditargetkan pada beberapa platform atau versi sistem operasi. Untuk informasi selengkapnya tentang cara melakukan ini, lihat akhir langkah 4 di Langkah 2: Buat manifes paket JSON.

9. Untuk Versi platform, verifikasi bahwa versi platform sistem operasi yang ditampilkan adalah **_any**, versi rilis utama diikuti dengan wildcard (7.*), atau versi rilis sistem operasi yang tepat yang Anda ingin perangkat lunak Anda diterapkan padanya. Untuk informasi selengkapnya tentang menentukan versi platform sistem operasi, lihat langkah 4 di Langkah 2: Buat manifes paket JSON.
10. Untuk Arsitektur, pilih arsitektur prosesor yang benar untuk setiap file yang dapat diinstal dari daftar dropdown. Untuk informasi selengkapnya tentang arsitektur prosesor yang didukung, lihat Platform dan arsitektur paket yang didukung.
11. (Opsional) Perluas Skrip, dan tinjau skrip yang Distributor dihasilkan untuk perangkat lunak yang dapat diinstal Anda.
12. (Opsional) Guna menyediakan skrip pembaruan untuk digunakan dengan pembaruan di tempat, perluas Skrip, pilih tab Memperbarui skrip, dan masukkan perintah skrip pembaruan Anda.

    Systems Manager tidak membuat skrip pembaruan atas nama Anda.
13. Untuk menambahkan lebih banyak file perangkat lunak yang dapat diinstal, pilih Tambahkan perangkat lunak. Jika tidak, lanjutkan ke langkah berikutnya.
14. (Opsional) Perluas Manifes, dan tinjau manifes paket JSON yang Distributor dihasilkan untuk perangkat lunak yang dapat diinstal Anda. Jika Anda mengubah informasi tentang perangkat lunak Anda sejak Anda memulai prosedur ini, seperti versi platform atau platform target, pilih Buat manifes untuk menampilkan manifes paket yang diperbarui.

Anda dapat mengedit manifes secara manual jika Anda ingin menargetkan perangkat lunak yang dapat diinstal di lebih dari satu sistem operasi, seperti yang dijelaskan pada langkah 8. Untuk informasi selengkapnya tentang manifest, lihat Langkah 2: Buat manifes paket JSON.

15. Pilih Buat paket.

Tunggu Distributor hingga selesai mengunggah perangkat lunak Anda dan membuat paket Anda. Distributormenampilkan status pengunggahan untuk setiap file yang dapat diinstal. Tergantung pada jumlah dan ukuran paket yang Anda tambahkan, ini dapat memakan waktu beberapa menit. Distributorsecara otomatis mengalihkan Anda ke halaman Detail Package untuk paket baru, tetapi Anda dapat memilih untuk membuka halaman ini sendiri setelah perangkat lunak telah diunggah. Halaman Detail Package tidak menampilkan semua informasi tentang paket Anda hingga Distributor selesai proses pembuatan paket. Untuk menghentikan proses pengunggahan dan pembuatan paket, pilih Batalkan.

Jika tidak Distributor dapat mengunggah file perangkat lunak, maka akan menampilkan pesan Pengunggahan Gagal. Untuk mencoba lagi pengunggahan, pilih Coba lagi pengunggahan. Untuk informasi selengkapnya tentang cara memecahkan masalah kegagalan pembuatan paket, lihat Pemecahan Masalah AWS Systems ManagerDistributor.

Membuat paket (lanjutan)

Pada bagian ini, pelajari tentang bagaimana pengguna tingkat lanjut dapat membuat paket Distributor setelah mengunggah aset yang dapat diinstal yang di-zip dengan skrip instalasi dan penghapusan instalasi, dan file manifes JSON, ke bucket Amazon S3.

Untuk membuat paket, siapkan file .zip aset yang dapat diinstal Anda, satu file .zip per platform sistem operasi. Setidaknya satu file .zip diperlukan untuk membuat sebuah paket. Selanjutnya, buatlah manifes JSON. Manifes menyertakan penunjuk ke file kode paket Anda. Ketika Anda sudah menambahkan file kode yang diperlukan ke folder atau direktori, dan manifes diisi dengan nilai-nilai yang benar, unggah paket Anda ke bucket S3.

Contoh paket, ExamplePackage.zip, tersedia untuk Anda unduh dari situs web kami. Contoh paket termasuk manifes JSON yang sudah selesai dan tiga file .zip.

Topik

- Langkah 1: Membuat file ZIP
- Langkah 2: Buat manifes paket JSON

- Langkah 3: Unggah paket dan manifes ke bucket Amazon S3
- Langkah 4: Tambahkan paket ke Distributor

Langkah 1: Membuat file ZIP

Dasar paket Anda setidaknya satu file .zip perangkat lunak atau aset yang dapat diinstal. Sebuah paket mencakup satu file .zip per sistem operasi yang ingin Anda dukung, kecuali satu file .zip dapat diinstal pada beberapa sistem operasi. Misalnya, instans Red Hat Enterprise Linux dan Amazon Linux biasanya dapat menjalankan file RPM yang dapat dieksekusi yang sama, sehingga Anda hanya perlu melampirkan satu file .zip ke paket Anda untuk mendukung kedua sistem operasi.

File yang diperlukan

Item berikut diperlukan dalam setiap file .zip:

- Sebuah install dan uninstall naskah. Windows Serverberbasis node dikelola memerlukan PowerShell script (script bernama `install.ps1` dan`uninstall.ps1`). Node terkelola berbasis Linux memerlukan skrip shell (skrip bernama `install.sh` dan). `uninstall.sh` SSM Agentmenjalankan instruksi dalam install dan uninstall script.

  Misalnya, skrip instalasi Anda mungkin menjalankan penginstal (seperti .rpm atau .msi), mereka dapat menyalin file, atau mungkin mengatur konfigurasi.
- File yang dapat dieksekusi, paket penginstal (.rpm, .deb, .msi, dll.), skrip lain, atau file konfigurasi.

File opsional

Item berikut adalah opsional dalam masing-masing file .zip:

- Skrip update. Menyediakan skrip pembaruan memungkinkan Anda untuk menggunakan opsi `In-place update` untuk menginstal sebuah paket. Bila Anda ingin menambahkan file baru atau yang diperbarui ke instalasi paket yang ada, `In-place update` opsi tidak mengambil aplikasi paket secara offline ketika pembaruan dilakukan. Windows Serverberbasis node dikelola memerlukan PowerShell script (script bernama`update.ps1`). Node terkelola berbasis Linux memerlukan skrip shell (skrip bernama). `update.sh` SSM Agentmenjalankan instruksi dalam update script.

Untuk informasi selengkapnya tentang menginstal atau memperbarui paket, lihat Menginstal atau memperbarui paket.

Untuk contoh file .zip, termasuk contoh install dan uninstall skrip, unduh contoh paket, ExamplePackage.zip.

Langkah 2: Buat manifes paket JSON

Setelah Anda mempersiapkan dan men-zip file yang dapat diinstal Anda, buatlah manifes JSON. Berikut ini adalah sebuah templat. Bagian-bagian templat manifes ini dijelaskan dalam prosedur di bagian ini. Anda dapat menggunakan editor JSON untuk membuat manifes ini dalam file terpisah. Atau, Anda dapat menulis manifes di konsol AWS Systems Manager saat Anda membuat sebuah paket.

```
{
  "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"
      }
    }
  }
}
```

Untuk membuat manifes paket JSON

1. Tambahkan versi skema ke manifes Anda. Dalam rilis ini, versi skema selalu `2.0`.

   ```
   { "schemaVersion": "2.0",
   ```

2. Tambahkan versi paket yang ditetapkan pengguna ke manifes Anda. Ini juga merupakan nilai Nama versi yang Anda tentukan ketika Anda menambahkan paket Anda keDistributor. Ini menjadi bagian dari AWS Systems Manager dokumen yang Distributor dibuat ketika Anda menambahkan paket Anda. Anda juga memberikan nilai ini sebagai input dalam dokumen `AWS-ConfigureAWSPackage` untuk menginstal versi paket selain yang terbaru. Nilai `version` dapat berisi huruf, angka, garis bawah, tanda hubung, dan titik, dengan panjang maksimum 128 karakter. Kami sarankan Anda menggunakan versi paket yang dapat dibaca manusia agar lebih mudah bagi Anda dan administrator lain dalam menentukan versi paket yang tepat ketika Anda men-deploy. Berikut adalah contoh.

   ```
   "version": "1.0.1",
   ```

3. (Opsional) Tambahkan nama penerbit. Berikut adalah contoh.

   ```
   "publisher": "MyOrganization",
   ```

4. Tambahkan paket. Bagian `"packages"` menjelaskan platform, versi rilis, dan arsitektur yang didukung oleh file .zip dalam paket Anda. Untuk informasi selengkapnya, lihat Platform dan arsitektur paket yang didukung.

   `Versi-platform` dapat menjadi nilai wildcard, `_any`. Gunakan ini untuk menunjukkan bahwa file .zip mendukung setiap rilis platform. Anda juga dapat menentukan versi rilis utama diikuti dengan wildcard sehingga semua versi minor didukung, misalnya 7.*. Jika Anda memilih untuk menentukan nilai *`versi-platform`* untuk versi sistem operasi tertentu, pastikan itu cocok

dengan versi rilis yang tepat dari AMI sistem operasi yang Anda targetkan. Berikut ini adalah sumber daya yang disarankan untuk mendapatkan nilai sistem operasi yang benar.

- Pada node terkelola Windows Server berbasis, versi rilis tersedia sebagai data Windows Management Instrumentation (WMI). Anda dapat menjalankan perintah berikut dari command prompt untuk mendapatkan informasi versi, kemudian mengurai hasilnya. `version` Perintah ini tidak menampilkan versi untuk Nano Windows Server; nilai versi untuk Nano Windows Server adalah `nano`.

  ```
  wmic OS get /format:list
  ```

- Pada node terkelola berbasis Linux, dapatkan versi dengan pertama-tama memindai rilis sistem operasi (perintah berikut). Cari nilai `VERSION_ID`.

  ```
  cat /etc/os-release
  ```

  Jika itu tidak mengembalikan hasil yang Anda butuhkan, jalankan perintah berikut untuk mendapatkan informasi rilis LSB dari file `/etc/lsb-release`, dan cari nilai `DISTRIB_RELEASE`.

  ```
  lsb_release -a
  ```

  Jika metode ini gagal, Anda biasanya dapat menemukan rilis berdasarkan distribusi. Misalnya, padaDebian Server, Anda dapat memindai `/etc/debian_version` file, atau padaRed Hat Enterprise Linux, `/etc/redhat-release` file.

  ```
  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"
        }
      }
    }
  }
```

Berikut adalah contohnya. Pada contoh ini, platform sistem operasi adalah `amazon`, versi rilis yang didukung adalah `2016.09`, arsitekturnya adalah `x86_64`, dan file .zip yang mendukung platform ini adalah `test.zip`.

```
{
    "amazon": {
        "2016.09": {
            "x86_64": {
                "file": "test.zip"
            }
        }
    }
},
```

Anda dapat menambahkan nilai wildcard `_any` untuk menunjukkan bahwa paket mendukung semua versi elemen induk. Misalnya, untuk menunjukkan bahwa paket didukung pada setiap versi rilis Amazon Linux, pernyataan paket Anda harus mirip dengan berikut ini. Anda dapat menggunakan wildcard `_any` di tingkat arsitektur atau versi untuk mendukung semua versi platform, atau semua arsitektur dalam versi, atau semua versi dan semua arsitektur platform.

```
{
    "amazon": {
        "_any": {
            "x86_64": {
                "file": "test.zip"
            }
        }
```

```
    }
},
```

Contoh berikut menambahkan `_any` untuk menunjukkan bahwa paket pertama, `data1.zip`, didukung untuk semua arsitektur Amazon Linux 2016.09. Paket kedua,`data2.zip`, didukung untuk semua rilis Amazon Linux, tetapi hanya untuk node terkelola dengan `x86_64` arsitektur. Baik versi `2016.09` dan `_any` adalah entri di bawah `amazon`. Ada satu platform (Amazon Linux), tetapi versi yang didukung, arsitektur, dan file .zip terkait berbeda.

```
{
    "amazon": {
        "2016.09": {
            "_any": {
                "file": "data1.zip"
            }
        },
        "_any": {
            "x86_64": {
                "file": "data2.zip"
            }
        }
    }
}
```

Anda dapat merujuk ke file .zip lebih dari sekali dalam bagian `"packages"` manifes, jika file.zip mendukung lebih dari satu platform. Misalnya, jika Anda memiliki file .zip mendukung Red Hat Enterprise Linux versi 7.x dan Amazon Linux, Anda memiliki dua entri di bagian `"packages"` yang mengarah ke file .zip yang sama, seperti yang ditunjukkan dalam contoh berikut.

```
{
    "amazon": {
        "2018.03": {
            "x86_64": {
                "file": "test.zip"
            }
        }
    },
    "redhat": {
        "7.*": {
            "x86_64": {
                "file": "test.zip"
```

```
            }
        }
    }
},
```

5. Tambahkan daftar file .zip yang merupakan bagian dari paket ini dari langkah 4. Setiap entri file memerlukan nama file dan checksum nilai hash `sha256`. Nilai checksum dalam manifes harus sesuai dengan nilai hash `sha256` dalam aset yang di-zip untuk mencegah instalasi paket gagal.

   Untuk mendapatkan checksum yang tepat dari file yang dapat diinstal Anda, Anda dapat menjalankan perintah berikut. Di Linux, jalankan `shasum -a 256 file-name.zip` atau `openssl dgst -sha256 file-name.zip`. Pada Windows, jalankan `Get-FileHash -Path path-to-.zip-file` cmdlet di. PowerShell

   Bagian `"files"` manifes termasuk satu referensi ke masing-masing file .zip dalam paket Anda.

```
"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. Setelah Anda menambahkan informasi paket Anda, simpan dan tutup file manifes.

Berikut ini adalah contoh manifes yang telah selesai. Pada contoh ini, Anda memiliki file .zip, `NewPackage_LINUX.zip`, yang mendukung lebih dari satu platform, tetapi dirujuk dalam bagian `"files"` hanya sekali.

```
{
    "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"
            }
        }
    }
}
```

Contoh paket

Contoh paket, ExamplePackage.zip, tersedia untuk Anda unduh dari situs web kami. Contoh paket termasuk manifes JSON yang telah selesai dan tiga file .zip.

Langkah 3: Unggah paket dan manifes ke bucket Amazon S3

Siapkan paket Anda dengan menyalin atau memindahkan semua file .zip ke dalam folder atau direktori. Sebuah paket yang valid memerlukan manifes yang Anda buat di Langkah 2: Buat manifes paket JSON dan semua file .zip yang diidentifikasi dalam daftar file manifes.

Untuk mengunggah paket dan manifes ke Amazon S3

1. Salin atau pindahkan semua file arsip .zip yang Anda tentukan dalam manifes ke folder atau direktori. Jangan men-zip folder atau direktori tempat Anda memindahkan file arsip .zip dan file manifes Anda.
2. Buat bucket atau pilih bucket yang ada. Untuk informasi selengkapnya, lihat Membuat Bucket di Panduan Memulai Amazon Simple Storage Service. Untuk informasi selengkapnya tentang cara menjalankan perintah AWS CLI untuk membuat bucket, lihat mb di Referensi Perintah AWS CLI.
3. Unggah folder atau direktori ke bucket. Untuk instruksi, kunjungi Menambah Objek ke Bucket dalam Panduan Memulai Amazon Simple Storage Service. Jika Anda berencana untuk menempelkan manifes JSON Anda ke dalam konsol AWS Systems Manager, jangan mengunggah manifes. Untuk informasi selengkapnya tentang cara menjalankan perintah AWS CLI untuk mengunggah file ke bucket, lihat mv di Referensi Perintah AWS CLI.
4. Di halaman utama bucket, pilih folder atau direktori yang Anda unggah. Jika Anda mengunggah file Anda ke subfolder dalam bucket, pastikan untuk mencatat subfolder (juga dikenal sebagai prefiks). Anda perlu prefiks untuk menambahkan paket Anda keDistributor.

Langkah 4: Tambahkan paket ke Distributor

Anda dapat menggunakan AWS Systems Manager konsol, alat baris AWS perintah (AWS CLIdanAWS Tools for PowerShell), atau AWS SDK untuk menambahkan paket baru. Distributor Saat menambahkan sebuah paket, Anda menambahkan Dokumen SSM baru. Dokumen ini memungkinkan Anda untuk men-deploy paket ke node terkelola.

Topik

- Menambahkan paket (konsol)
- Menambahkan paket (AWS CLI)

Menambahkan paket (konsol)

Anda dapat menggunakan konsol AWS Systems Manager untuk membuat sebuah paket. Siapkan nama bucket tempat Anda mengunggah paket Anda di Langkah 3: Unggah paket dan manifes ke bucket Amazon S3.

Untuk menambahkan paket ke Distributor (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.

2. Di panel navigasi, pilih Distributor.
3. Pada halaman Distributor utama, pilih Buat paket, lalu pilih Lanjutan.
4. Pada halaman Buat paket, masukkan nama untuk paket Anda. Nama hanya dapat berisi huruf, angka, titik, tanda hubung, dan garis bawah. Nama harus cukup generik untuk diterapkan ke semua versi lampiran paket, tetapi cukup spesifik untuk mengidentifikasi tujuan paket.
5. Untuk Nama versi, masukkan nilai yang tepat dari entri `version` dalam file manifes Anda.
6. Untuk Nama bucket S3, pilih nama bucket tempat Anda mengunggah file .zip dan manifes dalam the section called “Langkah 3: Unggah paket dan manifes ke bucket Amazon S3”.
7. Untuk prefiks kunci S3, masukkan subfolder bucket tempat file .zip dan manifes disimpan.
8. Untuk Manifes, pilih Ekstrak dari paket untuk menggunakan manifes yang telah Anda unggah ke bucket Amazon S3 dengan file .zip Anda.

   (Opsional) Jika Anda tidak mengunggah manifest JSON Anda ke bucket S3 tempat Anda menyimpan file .zip Anda, pilih Manifes baru. Anda dapat menulis atau menempelkan seluruh manifes di kolom editor JSON. Untuk informasi selengkapnya tentang cara membuat manifes JSON, lihatLangkah 2: Buat manifes paket JSON.
9. Setelah Anda selesai dengan manifes, pilih Buat paket.
10. Tunggu Distributor untuk membuat paket Anda dari file .zip dan manifes Anda. Bergantung pada jumlah dan ukuran paket yang Anda tambahkan, ini bisa memakan waktu beberapa menit. Distributorsecara otomatis mengalihkan Anda ke halaman Detail Package untuk paket baru, tetapi Anda dapat memilih untuk membuka halaman ini sendiri setelah perangkat lunak telah diunggah. Halaman Detail Package tidak menampilkan semua informasi tentang paket Anda hingga Distributor selesai proses pembuatan paket. Untuk menghentikan proses pengunggahan dan pembuatan paket, pilih Batalkan.

Menambahkan paket (AWS CLI)

Anda dapat menggunakan AWS CLI untuk membuat sebuah paket. Siapkan URL dari bucket tempat Anda mengunggah paket Anda di Langkah 3: Unggah paket dan manifes ke bucket Amazon S3.

Untuk menambahkan paket ke Amazon S3 (AWS CLI)

1. Untuk menggunakan AWS CLI untuk membuat paket, jalankan perintah berikut, ganti *`nama-paket`* dengan nama paket Anda dan *`path-to-manifest-file`*dengan file manifest JSON Anda. *`DOC-EXAMPLE-BUCKET`* adalah URL bucket Amazon S3 tempat seluruh paket disimpan.

Ketika Anda menjalankan create-document perintah diDistributor, Anda menentukan `Package` nilai untuk`--document-type`.

Jika Anda tidak menambahkan file manifes Anda ke bucket Amazon S3, nilai parameter `--content` adalah jalur file ke file manifes JSON.

```
aws ssm create-document \
    --name "package-name" \
    --content file://path-to-manifest-file \
    --attachments Key="SourceUrl",Values="DOC-EXAMPLE-BUCKET" \
    --version-name version-value-from-manifest \
    --document-type Package
```

Berikut adalah contoh.

```
aws ssm create-document \
    --name "ExamplePackage" \
    --content file://path-to-manifest-file \
    --attachments Key="SourceUrl",Values="https://s3.amazonaws.com/DOC-EXAMPLE-
BUCKET/ExamplePackage" \
    --version-name 1.0.1 \
    --document-type Package
```

2. Verifikasi bahwa paket Anda telah ditambahkan dan menampilkan manifes paket dengan menjalankan perintah berikut, dengan mengganti *nama-paket* dengan nama paket Anda. Untuk mendapatkan versi tertentu dari dokumen (tidak sama dengan versi paket), Anda dapat menambahkan parameter `--document-version`.

```
aws ssm get-document \
    --name "package-name"
```

Untuk informasi tentang pilihan lain yang dapat Anda gunakan dengan perintah create-document, lihat create-document di bagian AWS Systems Manager Referensi Perintah AWS CLI. Untuk informasi tentang pilihan lain yang dapat Anda gunakan dengan perintah get-document, lihat get-document.

Mengedit izin paket (konsol)

Setelah Anda menambahkan paket keDistributor, kemampuanAWS Systems Manager, Anda dapat mengedit izin paket di konsol Systems Manager. Anda dapat menambahkan Akun AWS ke izin paket. Paket dapat dibagikan dengan akun lain di Wilayah AWS yang sama saja. Berbagi lintas wilayah tidak didukung. Secara default, paket diatur ke Privat, yang berarti hanya mereka yang memiliki akses ke Akun AWS pembuat paket yang dapat melihat informasi paket dan memperbarui atau menghapus paket. Jika izin Privat dapat diterima, Anda dapat melewati prosedur ini.

> **Note**
>
> Anda dapat memperbarui izin paket yang dibagikan dengan 20 akun atau lebih sedikit.

Untuk mengedit izin paket (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Distributor.
3. Pada halaman Paket, pilih paket yang ingin Anda edit izinnya.
4. Pada tab Detail paket, pilih Mengedit izin untuk mengubah izin.
5. Untuk Mengedit izin, pilih Dibagikan dengan akun tertentu.
6. Di bawah Dibagikan dengan akun tertentu tambahkan, tambahkan nomor Akun AWS, satu per satu. Setelah selesai, pilih Simpan.

Mengedit tanda paket (konsol)

Setelah Anda menambahkan paket keDistributor, kemampuanAWS Systems Manager, Anda dapat mengedit tanda paket di konsol Systems Manager. Tanda ini diterapkan ke paket, dan tidak terhubung ke tanda pada node terkelola tempat Anda ingin men-deploy paketnya. Tanda adalah pasangan kunci dan nilai yang sensitif huruf yang dapat membantu Anda mengelompokkan dan memfilter paket berdasarkan kriteria yang relevan dengan organisasi Anda. Jika Anda tidak ingin menambahkan tanda, Anda siap untuk menginstal paket atau menambahkan versi baru.

Untuk mengedit tanda paket (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Distributor.

3. Pada halaman Paket, pilih paket yang ingin Anda edit tandanya.
4. Di tab Detail paket, pilih Tanda, pilih Edit.
5. Untuk Tambahkan tanda, masukkan kunci tanda, atau pasangan kunci tanda dan nilai, lalu pilih Tambahkan. Ulangi jika Anda ingin menambahkan lebih banyak tanda. Untuk menghapus tanda, pilih X di tanda pada bagian bawah jendela.
6. Setelah selesai menambahkan tanda ke paket Anda, pilih Simpan.

Menambahkan versi paketDistributor

Untuk menambahkan versi paket,Buat paket, dan kemudian gunakanDistributoruntuk menambahkan versi paket dengan menambahkan entri keAWS Systems Manager(SSM) dokumen yang sudah ada untuk versi yang lebih lama.Distributoradalah kemampuanAWS Systems Manager. Untuk menghemat waktu, perbarui manifes untuk versi paket yang lebih lama, ubah nilai entri `version` dalam manifes (sebagai contoh, dari `Test_1.0` ke `Test_2.0`) dan simpan sebagai manifes untuk versi baru. SederhanaTambahkan versialur kerjaDistributorkonsol memperbarui file manifes untuk Anda.

Versi paket baru dapat:

- Ganti setidaknya satu file yang dapat diinstal yang terlampir ke versi saat ini.
- Tambahkan file baru yang dapat diinstal untuk mendukung platform tambahan.
- Hapus file untuk menghentikan dukungan untuk platform tertentu.

Versi yang lebih baru dapat menggunakan bucket Amazon Simple Storage Service (Amazon S3) yang sama, tetapi harus memiliki URL dengan nama file yang berbeda ditampilkan di bagian akhir. Anda dapat menggunakan konsol Systems Manager atau AWS Command Line Interface (AWS CLI) untuk menambahkan versi baru. Mengunggah file yang dapat diinstal dengan nama yang tepat sama seperti file yang dapat diinstal yang ada di bucket Amazon S3 akan menimpa file yang ada. File yang dapat diinstal tidak akan disalin dari versi lama ke versi baru; Anda harus mengunggah file yang dapat diinstal dari versi lama agar mereka menjadi bagian dari versi baru. SetelahDistributorselesai membuat versi paket baru Anda, Anda dapat menghapus atau menggunakan ulang bucket Amazon S3, karenaDistributormenyalin perangkat lunak Anda ke bucket Systems Manager internal sebagai bagian dari proses pembuatan versi.

> **Note**
>
> Setiap paket disimpan hingga maksimal 25 versi. Anda dapat menghapus versi yang tidak diperlukan lagi.

Topik

- Menambahkan versi paket (konsol)
- Menambahkan versi paket (AWS CLI)

Menambahkan versi paket (konsol)

Sebelum Anda melakukan langkah-langkah ini, ikuti petunjuk di Buat paket untuk membuat paket baru untuk versi tersebut. Kemudian, gunakan konsol Systems ManagerDistributor.

Menambahkan versi paket (sederhana)

Untuk menambahkan versi paket dengan menggunakan alur kerja Sederhana, siapkan file yang dapat diinstal yang telah diperbarui atau tambahkan file yang dapat diinstal untuk mendukung lebih banyak platform dan arsitektur. Kemudian, gunakanDistributoruntuk mengunggah file baru yang dapat diinstal dan menambahkan versi paket. Yang disederhanakanTambahkan versialur kerjaDistributorconsole memperbarui file manifest dan dokumen SSM terkait untuk Anda.

Untuk menambahkan versi paket (sederhana)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Distributor.
3. PadaDistributorhalaman utama, pilih paket yang ingin Anda tambahkan versi lain.
4. Pada halaman Tambahkan versi, pilih Sederhana.
5. Untuk Nama versi, masukkan nama versi. Nama versi untuk versi baru harus berbeda dengan nama versi lama. Nama versi dapat maksimal 512 karakter, dan tidak dapat berisi karakter khusus.
6. Untuk Nama bucket S3, pilih bucket S3 yang ada dari daftar. Ini bisa bucket yang sama yang Anda gunakan untuk menyimpan file yang dapat diinstal untuk versi yang lebih lama, tetapi nama file yang dapat diinstal harus berbeda untuk menghindari menimpa file yang dapat diinstal yang ada di dalam bucket.

7. Untuk prefiks kunci S3, masukkan subfolder bucket tempat aset yang dapat diinstal Anda disimpan.
8. Untuk Unggah perangkat lunak, arahkan ke file perangkat lunak yang dapat diinstal yang ingin Anda lampirkan ke versi baru. File yang dapat diinstal dari versi yang ada tidak secara otomatis disalin ke versi baru; Anda harus mengunggah file yang dapat diinstal dari versi paket yang lebih lama jika Anda ingin salah satu file yang dapat diinstal yang sama menjadi bagian dari versi baru. Anda dapat mengunggah lebih dari satu file perangkat lunak dalam satu tindakan.
9. Untuk Platform target, verifikasikan bahwa platform sistem operasi target yang ditampilkan untuk setiap file yang dapat diinstal benar. Jika sistem operasi yang ditampilkan tidak benar, pilih sistem operasi yang benar dari daftar dropdown.

   Di alur kerja versioning Sederhana, karena Anda mengunggah setiap file yang dapat diinstal hanya sekali, langkah tambahan diperlukan untuk menargetkan satu file di beberapa sistem operasi. Misalnya, jika Anda mengunggah file perangkat lunak yang dapat diinstal bernama`Logtool_v1.1.1.rpm`, Anda harus mengubah beberapa default di`Sederhana`alur kerja`Distributor`untuk menargetkan perangkat lunak yang sama di sistem operasi Amazon Linux dan Ubuntu. Anda dapat melakukan salah satu hal berikut untuk mengatasi keterbatasan ini.

   - Gunakan alur kerja versioning Lanjutan sebagai gantinya, zip setiap file yang dapat diinstal ke dalam file .zip sebelum Anda memulai, dan secara manual menulis manifes sehingga satu file yang dapat diinstal dapat ditargetkan pada beberapa platform atau versi sistem operasi . Untuk informasi selengkapnya, lihat Menambahkan versi paket (lanjutan).
   - Edit file manifes secara manual di alur kerja Sederhana sehingga file .zip Anda ditargetkan pada beberapa versi atau platform sistem operasi. Untuk informasi selengkapnya tentang cara melakukan ini, lihat akhir langkah 4 di Langkah 2: Buat manifes paket JSON.
10. Untuk Versi platform, verifikasi bahwa versi platform sistem operasi yang ditampilkan adalah **_any**, versi rilis utama diikuti oleh wildcard (7.*), atau versi rilis sistem operasi yang tepat yang ingin diterapkan perangkat lunak Anda. Untuk informasi selengkapnya tentang bagaimana cara menentukan versi platform, lihat langkah 4 di Langkah 2: Buat manifes paket JSON.
11. Untuk Arsitektur, pilih arsitektur prosesor yang benar untuk setiap file yang dapat diinstal dari daftar drop-down. Untuk informasi selengkapnya tentang arsitektur yang didukung, lihat Platform dan arsitektur paket yang didukung.
12. (Opsional) Perluas`Skrip`, dan tinjau skrip instalasi dan penghapusan instalasi yang`Distributor`menghasilkan untuk perangkat lunak yang dapat diinstal Anda.
13. Untuk menambahkan lebih banyak file perangkat lunak yang dapat diinstal ke versi baru, pilih Tambahkan perangkat lunak. Jika tidak, lanjutkan ke langkah berikutnya.

14. (Opsional) PerluasManifest, dan meninjau manifes paketDistributormenghasilkan untuk perangkat lunak yang dapat diinstal Anda. Jika Anda mengubah informasi tentang perangkat lunak yang dapat diinstal sejak Anda memulai prosedur ini, seperti versi platform atau platform target, pilih Buat manifes untuk menampilkan manifes paket yang diperbarui.

    Anda dapat mengedit manifest secara manual jika Anda ingin menargetkan perangkat lunak yang dapat diinstal di lebih dari satu sistem operasi, seperti yang dijelaskan pada langkah 9. Untuk informasi selengkapnya tentang mengedit manifes, lihat Langkah 2: Buat manifes paket JSON.

15. Setelah selesai menambahkan perangkat lunak dan meninjau platform target, versi, dan data arsitektur, pilih Tambahkan versi.

16. MenungguDistributoruntuk menyelesaikan mengunggah perangkat lunak Anda dan membuat versi paket baru.Distributormenampilkan status pengunggahan untuk setiap file yang dapat diinstal. Bergantung pada jumlah dan ukuran paket yang Anda tambahkan, ini bisa memakan waktu beberapa menit.Distributorsecara otomatis mengarahkan Anda keDetail Packageuntuk paket, tetapi Anda dapat memilih untuk membuka halaman ini sendiri setelah perangkat lunak telah diunggah. ParameterDetail PackageHalaman tidak menampilkan semua informasi tentang paket AndaDistributorselesai membuat versi paket baru. Untuk menghentikan pengunggahan dan pembuatan versi paket, pilih Hentikan pengunggahan.

17. JikaDistributorTidak dapat mengunggah file perangkat lunak, yang dapat diinstalPengunggahanpesan. Untuk mencoba lagi pengunggahan, pilih Coba unggah lagi. Untuk informasi selengkapnya tentang cara memecahkan masalah kegagalan pembuatan versi paket, lihat Pemecahan Masalah AWS Systems ManagerDistributor.

18. SaatDistributorselesai membuat versi paket baru, pada paketRincianhalaman, padaVersitab, lihat versi baru dalam daftar versi paket yang tersedia. Tetapkan versi default paket dengan memilih versi, dan kemudian memilih Tetapkan versi default.

    Jika Anda tidak menetapkan versi default, versi paket terbaru adalah versi default.

Menambahkan versi paket (lanjutan)

Untuk menambahkan versi paket,Buat paket, dan kemudian gunakanDistributoruntuk menambahkan versi paket dengan menambahkan entri ke dokumen SSM yang ada untuk versi yang lebih lawas. Untuk menghemat waktu, perbarui manifes untuk versi paket yang lebih lama, ubah nilai entri `version` dalam manifes (sebagai contoh, dari `Test_1.0` ke `Test_2.0`) dan simpan itu sebagai

manifes untuk versi baru. Anda harus memiliki manifest yang diperbarui untuk menambahkan versi paket baru dengan menggunakan alur kerja Lanjutan.

Untuk menambahkan versi paket (lanjutan)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Distributor.
3. PadaDistributorhalaman utama, pilih paket yang ingin Anda tambahkan versi lain, dan kemudian pilihTambahkan versi.
4. Untuk Nama versi, masukkan nilai yang tepat yang ada di entri `version` file manifes Anda.
5. Untuk Nama bucket S3, pilih bucket S3 yang ada dari daftar. Ini bisa bucket yang sama yang Anda gunakan untuk menyimpan file yang dapat diinstal untuk versi yang lebih lama, tetapi nama file yang dapat diinstal harus berbeda untuk menghindari menimpa file yang dapat diinstal yang ada di dalam bucket.
6. Untuk Prefiks kunci S3, masukkan subfolder bucket tempat aset yang dapat diinstal Anda disimpan.
7. Untuk Manifes, pilih Ekstrak dari paket untuk menggunakan manifes yang Anda unggah ke bucket S3 dengan file .zip Anda.

   (Opsional) Jika Anda tidak mengunggah manifest JSON yang telah direvisi ke bucket Amazon S3 tempat Anda menyimpan file .zip, pilih Manifes baru. Anda dapat menulis atau menempelkan seluruh manifes di kolom editor JSON. Untuk informasi selengkapnya tentang cara membuat manifes JSON, lihat Langkah 2: Buat manifes paket JSON.
8. Setelah Anda selesai dengan manifes, pilih Tambah versi paket.
9. Pada halaman Detail paket, pada tab Versi, lihat versi baru dalam daftar versi paket yang tersedia. Tetapkan versi default paket dengan memilih versi, dan kemudian memilih Tetapkan versi default.

   Jika Anda tidak menetapkan versi default, versi paket terbaru adalah versi default.

Menambahkan versi paket (AWS CLI)

Anda dapat menggunakanAWS CLIuntuk menambahkan versi paketDistributor. Sebelum menjalankan perintah ini, Anda harus membuat versi paket baru dan mengunggahnya ke S3, seperti yang dijelaskan pada awal topik ini.

Untuk menambahkan versi paket (AWS CLI)

1. Jalankan perintah berikut untuk mengedit dokumen AWS Systems Manager dengan entri untuk versi paket baru. Ganti *nama-dokumen* dengan nama dokumen Anda. Ganti *DOC-EXAMPLE-BUCKET* dengan URL manifes JSON yang Anda salin di Langkah 3: Unggah paket dan manifes ke bucket Amazon S3. *S3-bucket-URL-of-package* adalah URL bucket Amazon S3 tempay seluruh paket disimpan. Ganti*version-name-from-updated-manifes*dengan nilai`version`dalam manifes Tetapkan parameter `--document-version` ke `$LATEST` untuk membuat dokumen yang terkait dengan versi paket ini menjadi versi terbaru dokumen.

```
aws ssm update-document \
    --name "document-name" \
    --content "S3-bucket-URL-to-manifest-file" \
    --attachments Key="SourceUrl",Values="DOC-EXAMPLE-BUCKET" \
    --version-name version-name-from-updated-manifest \
    --document-version $LATEST
```

Berikut adalah contoh.

```
aws ssm update-document \
    --name ExamplePackage \
    --content "https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/ExamplePackage/
manifest.json" \
    --attachments Key="SourceUrl",Values="https://s3.amazonaws.com/DOC-EXAMPLE-
BUCKET/ExamplePackage" \
    --version-name 1.1.1 \
    --document-version $LATEST
```

2. Jalankan perintah berikut untuk memverifikasi bahwa paket Anda telah diperbarui dan menampilkan manifes paket. Ganti *package-name* dengan nama paket Anda, dan secara opsional, *document-version* dengan nomor versi dokumen (tidak sama dengan versi paket) yang Anda perbarui. Jika versi paket ini terkait dengan versi terbaru dokumen, Anda dapat menentukan `$LATEST` untuk nilai parameter `--document-version` opsional.

```
aws ssm get-document \
    --name "package-name" \
    --document-version "document-version"
```

Untuk informasi tentang pilihan lain yang dapat Anda gunakan dengan perintah update-document, lihat update-document di bagian AWS Systems Manager Referensi Perintah AWS CLI.

Menginstal atau memperbarui paket

Anda dapat menerapkan paket ke node AWS Systems Manager terkelola Anda dengan menggunakanDistributor, kemampuan. AWS Systems Manager Untuk men-deploy paket, gunakan AWS Management Console atau AWS Command Line Interface (AWS CLI). Anda dapat men-deploy satu versi dari satu paket per perintah. Anda dapat menginstal paket baru atau memperbarui instalasi yang ada di tempat. Anda dapat memilih untuk menggunakan versi tertentu atau memilih untuk selalu men-deploy versi terbaru paket untuk deployment. Kami merekomendasikan menggunakanState Manager, kemampuanAWS Systems Manager, untuk menginstal paket. Menggunakan State Manager membantu memastikan bahwa node terkelola Anda selalu menjalankan sebagian besar up-to-date versi paket Anda.

Preferensi	Tindakan AWS Systems Manager	Info selengkapnya
Instal atau perbarui paket segera.	Run Command	• Menginstal atau memperbarui paket satu kali (konsol) • Menginstal paket satu kali (AWS CLI) • Memperbarui paket satu kali (AWS CLI)
Menginstal atau memperbarui paket menurut jadwal, sehingga instalasi selalu menyertakan versi default.	State Manager	• Penjadwalan instalasi atau pembaruan paket (konsol) • Menjadwalkan instalasi paket (AWS CLI) • Menjadwalkan pembaruan paket (AWS CLI)
Instal paket secara otomatis pada node terkelola baru yang memiliki tag atau kumpulan tag tertentu. Misalnya,	State Manager	Salah satu cara untuk melakukannya adalah dengan menerapkan tag ke node terkelola baru, lalu tentukan tag sebagai target dalam State

Preferensi	Tindakan AWS Systems Manager	Info selengkapnya
menginstal CloudWatch agen Amazon pada instance baru.		Manager asosiasi Anda. State Managersecara otomatis menginstal paket dalam asosiasi pada node terkelola yang memiliki tag yang cocok. Lihat Tentang target dan kontrol tingkat dalam State Manager asosiasi.

Topik

- Menginstal atau memperbarui paket satu kali (konsol)
- Penjadwalan instalasi atau pembaruan paket (konsol)
- Menginstal paket satu kali (AWS CLI)
- Memperbarui paket satu kali (AWS CLI)
- Menjadwalkan instalasi paket (AWS CLI)
- Menjadwalkan pembaruan paket (AWS CLI)

Menginstal atau memperbarui paket satu kali (konsol)

Anda dapat menggunakan konsol AWS Systems Manager untuk menginstal atau memperbarui paket satu kali. Ketika Anda mengkonfigurasi instalasi satu kaliAWS Systems Manager Run Command, Distributor menggunakan, kemampuanAWS Systems Manager, untuk melakukan instalasi.

Untuk menginstal atau memperbarui paket satu kali (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Distributor.
3. Di Distributor halaman beranda, pilih paket yang ingin Anda instal.
4. Pilih Instal satu kali.

   Perintah ini terbuka Run Command dengan dokumen perintah `AWS-ConfigureAWSPackage` dan Distributor paket Anda sudah dipilih.

5. Untuk Versi dokumen, pilih versi dokumen `AWS-ConfigureAWSPackage` yang ingin Anda jalankan.
6. Untuk Tindakan, pilih Instal.
7. Untuk Jenis Instalasi, pilih salah satu dari berikut ini:

   - Menghapus instalasi dan instalasi ulang: Paket benar-benar dihapus instalasinya, dan kemudian diinstal ulang. Aplikasi ini tidak tersedia sampai instalasi ulang selesai.
   - Pembaruan di tempat: Hanya file baru atau diubah ditambahkan ke instalasi yang ada sesuai dengan petunjuk yang Anda berikan dalam skrip `update`. Aplikasi tetap tersedia selama proses pembaruan. Opsi ini tidak didukung untuk paket AWS yang dipublikasikan kecuali paket `AWSEC2Launch-Agent`.
8. Untuk Nama, verifikasi bahwa nama paket yang Anda pilih telah dimasukkan.
9. (Opsional) Untuk Versi, masukkan nilai nama versi paket. Jika Anda membiarkan bidang ini kosong, Run Command instal versi default yang Anda pilih. Distributor
10. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat secara manual, atau dengan menentukan grup sumber daya.

    > **Note**
    >
    > Jika Anda tidak melihat node terkelola dalam daftar, lihatMemecahkan masalah ketersediaan node terkelola.

11. Untuk Parameter lainnya:

    - Untuk Komentar, ketik informasi tentang perintah ini.
    - Untuk Waktu habis (detik), tentukan jumlah detik untuk menunggu sistem sebelum gagal menjalankan perintah keseluruhan.
12. Untuk Kontrol Tingkat:

    - Untuk Concurrency, tentukan angka atau persentase target untuk menjalankan perintah secara bersamaan.

      > **Note**
      >
      > Jika Anda memilih target dengan menentukan tag atau grup sumber daya dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah

target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

- Untuk ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada target lain setelah gagal pada angka atau persentase node terkelola. Misalnya, jika Anda menentukan tiga kesalahan, maka Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

13. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan izin pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbedaAkun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

14. Di bagian Notifikasi SNS, jika Anda ingin notifikasi dikirim tentang status eksekusi perintah, pilih kotak centang Aktifkan notifikasi SNS.

    Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Run Command Amazon SNS, lihat. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

15. Saat Anda siap menginstal paket, pilih Jalankan.
16. Area Status perintah melaporkan kemajuan eksekusi. Jika perintah masih berlangsung, pilih ikon segarkan di pojok kiri atas konsol sampai kolom Status keseluruhan atau Status terperinci menampilkan Sukses atau Gagal.
17. Di area Target dan output, pilih tombol di sebelah nama node terkelola, lalu pilih Lihat output.

    Halaman output perintah menampilkan hasil eksekusi perintah Anda.

18. (Opsional) Jika Anda memilih menulis output perintah ke bucket Amazon S3, pilih Amazon S3 untuk melihat data log output.

Penjadwalan instalasi atau pembaruan paket (konsol)

Anda dapat menggunakan konsol AWS Systems Manager untuk menjadwalkan instalasi atau pembaruan paket. Saat Anda menjadwalkan instalasi atau pembaruan paket, Distributor gunakan AWS Systems Manager State Manager untuk menginstal atau memperbarui.

Untuk menjadwalkan instalasi paket (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Distributor.
3. Di Distributor halaman beranda, pilih paket yang ingin Anda instal atau perbarui.
4. Untuk Paket pilih Instal menurut jadwal.

   Perintah ini terbuka State Manager untuk asosiasi baru yang dibuat untuk Anda.
5. Untuk Nama masukkan sebuah nama (misalnya, **`Deploy-test-agent-package`**). Ini memang opsional, tetapi direkomendasikan. Spasi tidak diperbolehkan dalam nama.
6. Pada daftar Dokumen, nama dokumen `AWS-ConfigureAWSPackage` telah dipilih.
7. Untuk Tindakan, verifikasi bahwa Instal telah dipilih.
8. Untuk Jenis instalasi, lakukan hal berikut:

   - Menghapus instalasi dan instalasi ulang: Paket benar-benar dihapus instalasinya, dan kemudian diinstal ulang. Aplikasi ini tidak tersedia sampai instalasi ulang selesai.
   - Pembaruan di tempat: Hanya file baru atau diubah ditambahkan ke instalasi yang ada sesuai dengan petunjuk yang Anda berikan dalam skrip `update`. Aplikasi tetap tersedia selama proses pembaruan.
9. Untuk Nama, verifikasi bahwa nama paket Anda dimasukkan.
10. Untuk Versi, jika Anda ingin menginstal versi paket selain versi terbaru yang dipublikasikan, masukkan pengenal versi.
11. Untuk Target pilih Memilih semua instans terkelola di akun ini, Menentukan tanda, atau Memilih Instans Secara Manual. Jika Anda menargetkan sumber daya dengan menggunakan tanda, masukkan kunci tanda dan nilai tanda di kolom yang disediakan.

> **Note**
>
> Anda dapat memilih perangkat AWS IoT Greengrass inti terkelola dengan memilih memilih semua instans terkelola di akun ini atau Memilih Instans Secara Manual.

12. Untuk Tentukan jadwal pilih Menurut Jadwal untuk menjalankan asosiasi menurut jadwal reguler, atau Tanpa Jadwaluntuk menjalankan asosiasi sekali. Untuk informasi selengkapnya tentang opsi ini, lihat Bekerja dengan asosiasi di Systems Manager. Gunakan kontrol untuk membuat `cron` atau jadwal tingkat untuk asosiasi.
13. Pilih Buat Asosiasi.
14. Pada halaman Asosiasi, pilih tombol di sebelah asosiasi yang Anda buat, lalu pilih Terapkan asosiasi sekarang.

    State Managermembuat dan segera menjalankan asosiasi pada target yang ditentukan. Untuk informasi selengkapnya tentang hasil asosiasi yang berjalan, lihat Bekerja dengan asosiasi di Systems Manager dalam panduan ini.

Untuk informasi selengkapnya tentang bekerja dengan opsi di Opsi lanjutan, Pengendalian rate, dan Opsi output, lihat Bekerja dengan asosiasi di Systems Manager.

Menginstal paket satu kali (AWS CLI)

Anda dapat menjalankan send-command AWS CLI untuk menginstal Distributor paket satu kali. Jika paket sudah terinstal, aplikasi akan diambil secara offline saat paket dihapus instalasinya dan versi baru diinstal di tempatnya.

Untuk menginstal paket satu kali (AWS CLI)

- Jalankan perintah berikut di 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**
>
> Perilaku default untuk `installationType` adalah `Uninstall and reinstall`. Anda dapat menghilangkan `"installationType":["Uninstall and reinstall"]` dari perintah ini ketika Anda menginstal paket lengkap.

Berikut adalah contoh.

```
aws ssm send-command \
    --document-name "AWS-ConfigureAWSPackage" \
    --instance-ids "i-00000000000000" \
    --parameters '{"action":["Install"],"installationType":["Uninstall and
 reinstall"],"name":["ExamplePackage"]}'
```

Untuk informasi tentang pilihan lain yang dapat Anda gunakan dengan perintah send-command, lihat send-command di bagian AWS Systems Manager Referensi Perintah AWS CLI.

Memperbarui paket satu kali (AWS CLI)

Anda dapat menjalankan send-command AWS CLI untuk memperbarui Distributor paket tanpa membuat aplikasi terkait offline. Hanya file baru atau yang diperbarui dalam paket yang diganti.

Untuk memperbarui paket satu kali (AWS CLI)

- Jalankan perintah berikut di 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**
>
> Ketika Anda menambahkan file baru atau yang diubah, Anda harus menyertakan `"installationType":["In-place update"]` dalam perintah.

Berikut adalah contoh.

```
aws ssm send-command \
    --document-name "AWS-ConfigureAWSPackage" \
    --instance-ids "i-02573cafcfEXAMPLE" \
```

```
    --parameters '{"action":["Install"],"installationType":["In-place
 update"],"name":["ExamplePackage"]}'
```

Untuk informasi tentang pilihan lain yang dapat Anda gunakan dengan perintah send-command, lihat send-command di bagian AWS Systems Manager Referensi Perintah AWS CLI.

Menjadwalkan instalasi paket (AWS CLI)

Anda dapat menjalankan create-association AWS CLI untuk menginstal Distributor paket sesuai jadwal. Nilai `--name`, nama dokumen, selalu `AWS-ConfigureAWSPackage`. Perintah berikut menggunakan kunci `InstanceIds` untuk menentukan target node terkelola. Jika paket sudah terinstal, aplikasi akan diambil secara offline saat paket dihapus instalasinya dan versi baru diinstal di tempatnya.

```
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**
>
> Perilaku default untuk `installationType` adalah `Uninstall and reinstall`. Anda dapat menghilangkan `"installationType":["Uninstall and reinstall"]` dari perintah ini ketika Anda menginstal paket lengkap.

Berikut adalah contoh.

```
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\"]}]
```

Untuk informasi tentang pilihan lain yang dapat Anda gunakan dengan perintah create-association, lihat create-association di bagian AWS Systems Manager Referensi Perintah AWS CLI.

Menjadwalkan pembaruan paket (AWS CLI)

Anda dapat menjalankan create-association AWS CLI untuk memperbarui Distributor paket sesuai jadwal tanpa membuat aplikasi terkait offline. Hanya file baru atau yang diperbarui dalam paket yang diganti. Nilai `--name`, nama dokumen, selalu `AWS-ConfigureAWSPackage`. Perintah berikut menggunakan kunci `InstanceIds` untuk menentukan instans target.

```
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**
>
> Ketika Anda menambahkan file baru atau yang diubah, Anda harus menyertakan `"installationType":["In-place update"]` dalam perintah.

Berikut adalah contoh.

```
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\"]}]
```

Untuk informasi tentang pilihan lain yang dapat Anda gunakan dengan perintah create-association, lihat create-association di bagian AWS Systems Manager Referensi Perintah AWS CLI.

Menghapus instalasi paket.

Anda dapat menggunakan AWS Management Console atau AWS Command Line Interface (AWS CLI) untuk menghapus Distributor paket dari node AWS Systems Manager terkelola Anda dengan

menggunakanRun Command. DistributorRun Commanddan kemampuanAWS Systems Manager. Dalam rilis ini, Anda dapat menghapus instalasi satu versi dari satu paket per perintah. Anda dapat menghapus instalasi versi tertentu atau versi default.

Topik

- Menghapus instalasi paket (konsol)
- Menghapus instalasi paket (AWS CLI)

Menghapus instalasi paket (konsol)

Anda dapat menggunakan Run Command konsol Systems Manager untuk menghapus instalasi paket satu kali. Distributormenggunakan AWS Systems Manager Run Command untuk menghapus paket.

Untuk menghapus instalasi paket (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command.
3. Di halaman Run Command beranda, pilih Jalankan perintah.
4. Pilih dokumen perintah `AWS-ConfigureAWSPackage`.
5. Dari Tindakan, pilih Menghapus instalasi
6. Untuk Nama, masukkan nama paket yang ingin Anda hapus instalasinya.
7. Untuk Target, pilih cara Anda menargetkan node Anda. Anda dapat menentukan kunci tag dan nilai yang dibagikan oleh target. Anda juga dapat menentukan target dengan memilih atribut, seperti ID, platform, dan SSM Agent versi.
8. Anda dapat menggunakan opsi lanjutan untuk menambahkan komentar tentang operasi, ubah nilai Konkurensi dan Ambang kesalahan dalam Pengendalian rate, tentukan opsi output atau konfigurasikan notifikasi Amazon Simple Notification Service (Amazon SNS). Untuk informasi selengkapnya, lihat Menjalankan Perintah dari Konsol dalam panduan ini.
9. Saat Anda siap menghapus instalasi paket, pilih Run Command, lalu pilih Lihat hasil.
10. Dalam daftar perintah, pilih opsi perintah `AWS-ConfigureAWSPackage` yang Anda jalankan. Jika perintah masih berlangsung, pilih ikon segarkan di sudut kanan atas konsol.
11. Saat kolom Status menampilkan Sukses atau Gagal, pilih tab Output.
12. Pilih Tampilkan output. Halaman output perintah menampilkkan hasil eksekusi perintah Anda.

Menghapus instalasi paket (AWS CLI)

Anda dapat menggunakan AWS CLI untuk menghapus Distributor paket dari node yang dikelola dengan menggunakanRun Command.

Untuk menghapus instalasi paket (AWS CLI)

- Jalankan perintah berikut di 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)"]}'
```

  Berikut adalah contoh.

```
aws ssm send-command \
    --document-name "AWS-ConfigureAWSPackage" \
    --instance-ids "i-02573cafcfEXAMPLE" \
    --parameters '{"action":["Uninstall"],"name":["Test-ConfigureAWSPackage"]}'
```

Untuk informasi tentang pilihan lain yang dapat Anda gunakan dengan perintah send-command, lihat send-command di bagian AWS Systems Manager Referensi Perintah AWS CLI.

Menghapus paket

Bagian ini menjelaskan cara menghapus paket. Anda tidak dapat menghapus versi paket, hanya seluruh paket.

Menghapus paket (konsol)

Anda dapat menggunakanAWS Systems Managerkonsol untuk menghapus paket atau versi paketDistributor, suatu kemampuanAWS Systems Manager. Menghapus paket akan menghapus semua versi paketDistributor.

Untuk menghapus paket (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Distributor.

3. PadaDistributorhalaman utama, pilih paket yang ingin Anda hapus.
4. Pada halaman rincian paket, pilih Hapus paket.
5. Saat diminta mengonfirmasi penghapusan, pilih Hapus paket.

Menghapus versi paket (konsol)

Anda dapat menggunakan konsol Systems Manager untuk menghapus versi paketDistributor.

Untuk menghapus versi paket (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Distributor.
3. PadaDistributorhalaman utama, pilih paket yang ingin Anda hapus versinya.
4. Pada halaman versi paket, pilih versi yang akan dihapus dan pilih Hapus versi.
5. Saat diminta mengonfirmasi penghapusan, pilih Hapus versi paket.

Menghapus paket (baris perintah)

Anda dapat menggunakan alat baris perintah pilihan Anda untuk menghapus paketDistributor.

Linux & macOS

Untuk menghapus paket (AWS CLI)

1. Jalankan perintah berikut untuk membuat daftar dokumen paket tertentu. Pada hasil perintah tersebut, cari paket yang ingin Anda hapus.

```
aws ssm list-documents \
    --filters Key=Name,Values=package-name
```

2. Jalankan perintah berikut untuk menghapus paket. Ganti *Nama paket* dengan nama paket.

```
aws ssm delete-document \
    --name "package-name"
```

3. Jalankan perintah list-documents lagi untuk memverifikasi bahwa paket telah dihapus. Paket yang Anda hapus tidak boleh disertakan dalam daftar.

```
aws ssm list-documents \
```

```
    --filters Key=Name,Values=package-name
```

Windows

Untuk menghapus paket (AWS CLI)

1. Jalankan perintah berikut untuk membuat daftar dokumen paket tertentu. Pada hasil perintah tersebut, cari paket yang ingin Anda hapus.

```
aws ssm list-documents ^
    --filters Key=Name,Values=package-name
```

2. Jalankan perintah berikut untuk menghapus paket. Ganti *Nama paket* dengan nama paket.

```
aws ssm delete-document ^
    --name "package-name"
```

3. Jalankan perintah list-documents lagi untuk memverifikasi bahwa paket telah dihapus. Paket yang Anda hapus tidak boleh disertakan dalam daftar.

```
aws ssm list-documents ^
    --filters Key=Name,Values=package-name
```

PowerShell

Untuk menghapus paket (Tools for PowerShell)

1. Jalankan perintah berikut untuk membuat daftar dokumen paket tertentu. Pada hasil perintah tersebut, cari paket yang ingin Anda hapus.

```
$filter = New-Object
 Amazon.SimpleSystemsManagement.Model.DocumentKeyValuesFilter
$filter.Key = "Name"
$filter.Values = "package-name"

Get-SSMDocumentList `
    -Filters @($filter)
```

2. Jalankan perintah berikut untuk menghapus paket. Ganti *Nama paket* dengan nama paket.

```
Remove-SSMDocument `
    -Name "package-name"
```

3. Jalankan perintah Get-SSMDocumentList lagi untuk memverifikasi bahwa paket telah dihapus. Paket yang Anda hapus tidak boleh disertakan dalam daftar.

```
$filter = New-Object
 Amazon.SimpleSystemsManagement.Model.DocumentKeyValuesFilter
$filter.Key = "Name"
$filter.Values = "package-name"

Get-SSMDocumentList `
    -Filters @($filter)
```

Menghapus versi paket (baris perintah)

Anda dapat menggunakan alat baris perintah pilihan Anda untuk menghapus versi paketDistributor.

Linux & macOS

Untuk menghapus versi paket (AWS CLI)

1. Jalankan perintah berikut untuk menampilkan versi paket Anda. Pada hasil perintah tersebut, cari versi paket yang ingin Anda hapus.

```
aws ssm list-document-versions \
    --name "package-name"
```

2. Jalankan perintah berikut untuk menghapus versi paket. Ganti *Nama paket* dengan nama paket dan *versi* dengan nomor versi.

```
aws ssm delete-document \
    --name "package-name" \
    --document-version version
```

3. Jalankan perintah list-document-versions untuk memverifikasi bahwa versi paket telah dihapus. Versi paket yang Anda hapus seharusnya tidak ditemukan.

```
aws ssm list-document-versions \
```

```
    --name "package-name"
```

Windows

Untuk menghapus versi paket (AWS CLI)

1. Jalankan perintah berikut untuk menampilkan versi paket Anda. Pada hasil perintah tersebut, cari versi paket yang ingin Anda hapus.

```
aws ssm list-document-versions ^
    --name "package-name"
```

2. Jalankan perintah berikut untuk menghapus versi paket. Ganti *Nama paket* dengan nama paket dan *versi* dengan nomor versi.

```
aws ssm delete-document ^
    --name "package-name" ^
    --document-version version
```

3. Jalankan perintah list-document-versions untuk memverifikasi bahwa versi paket telah dihapus. Versi paket yang Anda hapus seharusnya tidak ditemukan.

```
aws ssm list-document-versions ^
    --name "package-name"
```

PowerShell

Untuk menghapus versi paket (Tools for PowerShell)

1. Jalankan perintah berikut untuk menampilkan versi paket Anda. Pada hasil perintah tersebut, cari versi paket yang ingin Anda hapus.

```
Get-SSMDocumentVersionList `
    -Name "package-name"
```

2. Jalankan perintah berikut untuk menghapus versi paket. Ganti *Nama paket* dengan nama paket dan *versi* dengan nomor versi.

```
Remove-SSMDocument `
```

```
    -Name "package-name" `
    -DocumentVersion version
```

3. Jalankan perintah Get-SSMDocumentVersionList untuk memverifikasi bahwa versi paket telah dihapus. Versi paket yang Anda hapus seharusnya tidak ditemukan.

```
Get-SSMDocumentVersionList `
    -Name "package-name"
```

Untuk informasi tentang pilihan lain yang dapat Anda gunakan dengan perintah list-documents, lihat list-documents di bagian AWS Systems Manager Referensi PerintahAWS CLI. Untuk informasi tentang pilihan lain yang dapat Anda gunakan dengan perintah delete-document, lihat delete-document.

Audit dan loggingDistributoraktivitas

Anda dapat menggunakanAWS CloudTrailuntuk mengaudit kegiatan yang berkaitan denganDistributor, suatu kemampuanAWS Systems Manager. Untuk informasi selengkapnya tentang opsi pengauditan dan pencatatan untuk Systems Manager, lihat Pemantauan AWS Systems Manager.

AuditDistributoraktivitas menggunakan CloudTrail

CloudTrail menangkap panggilan API yang dibuat diAWS Systems Managerkonsol,AWS Command Line Interface(AWS CLI), dan SDK Systems Manager. Informasi dapat dilihat di CloudTrail konsol atau disimpan dalam bucket Amazon Simple Storage Service (Amazon S3). Satu bucket digunakan untuk semua CloudTrail log untuk akun Anda.

Log dariRun CommanddanState Managertindakan menunjukkan aktivitas pembuatan dokumen, instalasi paket, dan penghapusan instalasi paket.Run CommanddanState Manageradalah kemampuanAWS Systems Manager. Untuk informasi lebih lanjut tentang melihat dan menggunakan CloudTrail log aktivitas Systems Manager, lihatPencatatan panggilan AWS Systems Manager API dengan AWS CloudTrail.

Pemecahan Masalah AWS Systems ManagerDistributor

Informasi berikut dapat membantu Anda memecahkan masalah yang mungkin terjadi ketika Anda menggunakanDistributor, kemampuan. AWS Systems Manager

Topik

Paket yang salah dengan nama yang sama telah diinstal

Masalah: Anda telah menginstal paket, tetapi Distributor menginstal paket yang berbeda sebagai gantinya.

Penyebab: Selama instalasi, Systems Manager menemukan paket AWS dipublikasikan sebagai hasil sebelum paket eksternal yang ditetapkan pengguna. Jika nama paket yang ditetapkan pengguna Anda sama dengan nama paket yang dipublikasikan AWS, paket AWS yang diinstal bukan paket Anda.

Solusi: Untuk menghindari masalah ini, namai paket Anda dengan nama yang berbeda dengan nama paket AWS yang dipublikasikan.

Kesalahan: Gagal mengambil manifes: Tidak dapat menemukan versi terbaru paket

Masalah: Anda menerima pesan kesalahan seperti berikut ini.

```
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
```

Penyebab: Anda menggunakan versi SSM Agent with Distributor yang lebih awal dari versi 2.3.274.0.

Solusi: Perbarui versi SSM Agent ke versi 2.3.274.0 atau yang lebih baru. Untuk informasi selengkapnya, lihat Memperbarui SSM Agent penggunaan Run Command atau Walkthrough: Perbarui secara otomatis (SSM AgentCLI).

Kesalahan: Gagal mengambil manifes: Pengecualian validasi

Masalah: Anda menerima pesan kesalahan seperti berikut ini.

```
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
```

Penyebab: Anda menggunakan versi SSM Agent with Distributor yang lebih awal dari versi 2.3.274.0.

Solusi: Perbarui versi SSM Agent ke versi 2.3.274.0 atau yang lebih baru. Untuk informasi selengkapnya, lihat Memperbarui SSM Agent penggunaan Run Command atau Walkthrough: Perbarui secara otomatis (SSM AgentCLI).

Paket tidak didukung (paket tidak memiliki tindakan instalasi)

Masalah: Anda menerima pesan kesalahan seperti berikut ini.

```
Package is not supported (package is missing install action)
```

Penyebab: Struktur direktori paket tidak benar.

Solusi: Jangan zip direktori induk yang berisi perangkat lunak dan skrip yang diperlukan. Sebaliknya, buat file `.zip` semua isi yang diperlukan secara langsung di jalur absolut. Untuk memverifikasi file `.zip` dibuat dengan benar, unzip direktori platform target dan tinjau struktur direktori. Sebagai contoh, jalur absolut skrip instal harus */ExamplePackage_targetPlatform*`/install.sh`.

Kesalahan: Gagal mengunduh manifes: Dokumen dengan nama tidak ada

Masalah: Anda menerima pesan kesalahan seperti berikut ini.

```
Failed to download manifest - failed to retrieve package document description:
 InvalidDocument: Document with name filename does not exist.
```

Penyebab: tidak Distributor dapat menemukan paket dengan nama paket saat membagikan Distributor paket dari akun lain.

Solusi: Saat berbagi paket dari akun lain, gunakan Amazon Resource Name (ARN) lengkap untuk paket dan bukan hanya namanya.

Unggahan gagal.

Masalah: Anda menerima pesan kesalahan seperti berikut ini.

```
Upload failed. At least one of your files was not successfully uploaded to your S3
 bucket.
```

Penyebab: Nama paket perangkat lunak Anda termasuk spasi. Misalnya, `Hello World.msi` akan gagal mengunggah.

AWS Systems Manager Sumber Daya Bersama

Systems Manager menggunakan sumber daya bersama berikut untuk mengelola dan mengonfigurasi sumber daya AWS Anda.

Topik

- AWS Systems ManagerDokumen

AWS Systems ManagerDokumen

Dokumen SSM AWS Systems Manager (dokumen SSM) menentukan tindakan yang dilakukan Systems Manager pada instans terkelola Anda. Systems Manager mencakup lebih dari 100 dokumen pra-konfigurasi yang dapat Anda gunakan dengan menentukan parameter di runtime. Anda dapat menemukan dokumen yang telah dikonfigurasi sebelumnya di konsol Systems Manager Documents dengan memilih tab Dimiliki oleh Amazon, atau dengan menentukan Amazon untuk `Owner` filter saat memanggil operasi `ListDocuments` API. Dokumen menggunakan JavaScript Object Notation (JSON) atau YAMM, dan dokumen tersebut menyertakan langkah-langkah dan parameter yang Anda tentukan. Untuk memulai dengan dokumen SSM, buka konsol Systems Manager. Di panel navigasi, pilih Dokumen.

Bagaimana kemampuan Dokumen dapat bermanfaat bagi organisasi saya?

Dokumen, kemampuanAWS Systems Manager, menawarkan manfaat ini:

- Kategori dokumen

  Untuk membantu Anda menemukan dokumen yang Anda butuhkan, pilih kategori tergantung pada jenis dokumen yang Anda cari. Untuk memperluas pencarian Anda, Anda dapat memilih beberapa kategori dari jenis dokumen yang sama. Memilih kategori dari jenis dokumen yang berbeda tidak didukung. Kategori hanya didukung untuk dokumen yang dimiliki oleh Amazon.

- Versi dokumen

  Anda dapat membuat dan menyimpan berbagai versi dokumen. Anda kemudian dapat menentukan versi default untuk setiap dokumen. Versi default dokumen dapat diperbarui ke versi yang lebih baru atau dikembalikan ke versi lama dokumen. Ketika Anda mengubah konten dokumen, Systems Manager secara otomatis menambahkan versi dokumen. Anda dapat mengambil atau

menggunakan versi dokumen apa pun dengan menentukan versi dokumen di konsol, perintah AWS Command Line Interface (AWS CLI), atau panggilan API.

- Sesuaikan dokumen untuk kebutuhan Anda

  Jika Anda ingin menyesuaikan langkah dan tindakan dalam dokumen, Anda dapat membuat milik Anda sendiri. Sistem menyimpan dokumen dengan Anda Akun AWS di tempat Wilayah AWS Anda membuatnya. Untuk informasi lebih lanjut tentang cara membuat dokumen SSM, lihat Membuat konten dokumen SSM.

- Menandai dokumen

  Anda dapat menandai dokumen untuk membantu mengidentifikasi satu atau beberapa dokumen dengan cepat berdasarkan tag yang telah Anda tetapkan. Misalnya, Anda dapat menandai dokumen untuk lingkungan, departemen, pengguna, grup, atau periode tertentu. Anda juga dapat membatasi akses ke dokumen dengan membuat kebijakan AWS Identity and Access Management (IAM) yang menentukan tag yang dapat diakses oleh pengguna atau grup. Untuk informasi selengkapnya, lihat Menandai dokumen Systems Manager.

- Bagikan dokumen

  Anda dapat membuat dokumen publik atau membagikannya dengan Akun AWS yang sama Wilayah AWS. Berbagi dokumen antar akun dapat berguna jika, misalnya, Anda ingin semua instans Amazon Elastic Compute Cloud (Amazon EC2) yang Anda berikan kepada pelanggan atau karyawan memiliki konfigurasi yang sama. Selain menjaga aplikasi atau tambalan pada instans tetap mutakhir, Anda mungkin ingin membatasi instance pelanggan dari aktivitas tertentu. Atau Anda mungkin ingin memastikan bahwa instans yang digunakan oleh akun karyawan di seluruh organisasi Anda diberikan akses ke sumber daya internal tertentu. Untuk informasi selengkapnya, lihat Membagikan dokumen SSM.

Siapa yang harus menggunakan dokumen?

- Setiap AWS pelanggan yang ingin menggunakan kemampuan Systems Manager untuk meningkatkan efisiensi operasional mereka dalam skala besar, mengurangi kesalahan yang terkait dengan intervensi manual, dan mengurangi waktu untuk menyelesaikan masalah umum.
- Pakar infrastruktur yang ingin mengotomatiskan tugas penerapan dan konfigurasi.
- Administrator yang ingin menyelesaikan masalah umum dengan andal, meningkatkan efisiensi pemecahan masalah, dan mengurangi operasi berulang.
- Pengguna yang ingin mengotomatiskan tugas yang biasanya mereka lakukan secara manual.

Apa jenis dokumen SSM?

Tabel berikut menjelaskan berbagai jenis dokumen SSM dan penggunaannya.

Tipe	Menggunakan dengan	Detail
ApplicationConfiguration ApplicationConfigurationSchema	AWS AppConfig	AWS AppConfig, kemampuan AWS Systems Manager, memungkinkan Anda untuk membuat, mengelola, dan dengan cepat menyebarkan konfigurasi aplikasi. Anda dapat menyimpan data konfigurasi dalam dokumen SSM dengan membuat dokumen yang menggunakan jenis `ApplicationConfiguration` dokumen. Untuk informasi selengkapnya, lihat Konfigurasi bentuk bebas di AWS AppConfigPanduan Pengguna. Jika Anda membuat konfigurasi dalam dokumen SSM, maka Anda harus menentukan Skema JSON yang sesuai. Skema menggunakan jenis `ApplicationConfigurationSchema` dokumen dan, seperti seperangkat aturan, mendefinisikan properti yang diizinkan untuk setiap pengaturan konfigurasi aplikasi. Untuk informasi selengkapnya, lihat Tentang

Tipe	Menggunakan dengan	Detail
		validator di AWS AppConfig Panduan Pengguna.
Runbook otomatisasi	Otomasi State Manager Maintenance Windows	Gunakan runbook Otomasi saat melakukan tugas pemeliharaan dan penerapan umum seperti membuat atau memperbarui Amazon Machine Image ()AMI. State Managermenggunakan runbook Otomasi untuk menerapkan konfigurasi. Tindakan ini dapat dijalankan pada satu atau beberapa target kapan saja selama siklus hidup sebuah instance. Maintenance Windowsmenggunakan runbook Otomasi untuk melakukan tugas pemeliharaan dan penerapan umum berdasarkan jadwal yang ditentukan. Semua runbook otomatisasi yang didukung untuk sistem operasi berbasis Linux juga didukung oleh instans EC2 untuk macOS.

Tipe	Menggunakan dengan	Detail
Dokumen Perubahan Kalender	Change Calendar	Change Calendar, kemampuanAWS Systems Manager, menggunakan jenis `ChangeCalendar` dokumen. Change CalendarDokumen menyimpan entri kalender dan acara terkait yang dapat memungkinkan atau mencegah tindakan Otomasi mengubah lingkungan Anda. DiChange Calendar, dokumen menyimpan data iCalendar 2.0 dalam format teks biasa. Change Calendartidak didukung pada instans EC2 untuk. macOS

Tipe	Menggunakan dengan	Detail
templat AWS CloudFormation	AWS CloudFormation	AWS CloudFormationtemplate menjelaskan sumber daya yang ingin Anda sediakan di CloudFormation tumpukan Anda. Dengan menyimpan CloudFormation template sebagai dokumen Systems Manager, Anda bisa mendapatkan keuntungan dari fitur dokumen Systems Manager. Ini termasuk membuat dan membandingkan beberapa versi template Anda, dan berbagi template Anda dengan akun lain di Wilayah AWS. Anda dapat membuat dan mengedit CloudFormation template dan tumpukan dengan menggunakanApplication Manager, kemampuan Systems Manager. Untuk informasi selengkapnya, lihat Bekerja denganAWS CloudFormationtemplate dan tumpukan diApplication Manager.

Tipe	Menggunakan dengan	Detail
Dokumen perintah	Run Command State Manager Maintenance Windows	Run Command, kemampuan AWS Systems Manager, menggunakan dokumen Command untuk menjalankan perintah. State Manager, kemampuanAWS Systems Manager, menggunakan dokumen perintah untuk menerapkan konfigurasi. Tindakan ini dapat dijalankan pada satu atau beberapa target kapan saja selama siklus hidup sebuah instance. Maintenance Windows, kemampuanAWS Systems Manager, menggunakan dokumen Command untuk menerapkan konfigurasi berdasarkan jadwal yang ditentukan. Sebagian besar dokumen Perintah didukung oleh semua Linux dan sistem operasi Windows Server yang didukung oleh Systems Manager. Dokumen-dokumen Perintah berikut didukung pada instans EC2 untuk macOS: • `AWS-ConfigureAWSPackage` • `AWS-RunPatchBaseline`

Tipe	Menggunakan dengan	Detail
		• `AWS-RunPatchBaselineAssociation` • `AWS-RunShellScript`
AWS Configtemplat paket kesesuaian	AWS Config	AWS ConfigTemplat paket kesesuaian adalah dokumen berformat YAMM yang digunakan untuk membuat paket kesesuaian yang berisi daftar aturan terkelola atau kustom serta tindakan remediasi. AWS Config Untuk informasi selengkapnya, lihat Paket Kesesuaian.
Dokumen paket	Distributor	DalamDistributor, kemampuan AWS Systems Manager, paket diwakili oleh dokumen SSM. Dokumen paket termasuk lampiran file arsip ZIP yang berisi perangkat lunak atau aset untuk diinstal pada instans terkelola. Membuat paket dalam Distributor membuat dokumen paket. Distributortidak didukung di Oracle Linux dan instance macOS terkelola.

Tipe	Menggunakan dengan	Detail
Dokumen kebijakan	State Manager	Inventaris, kemampuan AWS Systems Manager, menggunakan dokumen `AWS-GatherSoftware Inventory` Kebijakan dengan State Manager asosiasi untuk mengumpulkan data inventaris dari instans terkelola. Ketika membuat Anda dokumen SSM sendiri, Runbook otomatisasi dan dokumen perintah adalah metode yang lebih disukai untuk menegakkan kebijakan pada instans terkelola. Inventaris Systems Manager dan Dokumen kebijakan `AWS-GatherSoftware Inventory` yang didukung oleh semua sistem operasi yang didukung oleh Systems Manager.

Tipe	Menggunakan dengan	Detail
Template analisis pasca insiden	Analisis pasca-insiden Manajer Insiden	Incident Manager menggunakan template analisis pasca insiden untuk membuat analisis berdasarkan manajemen operasi praktik AWS terbaik. Gunakan templat untuk membuat analisis yang dapat digunakan tim Anda untuk mengidentifikasi peningkatan respons insiden Anda.

Tipe	Menggunakan dengan	Detail
Dokumen sesi	Session Manager	Session ManagerKe mampuanAWS Systems Manager, menggunak an dokumen Sesi untuk menentukan jenis sesi yang akan dimulai, seperti sesi penerusan port, sesi untuk menjalankan perintah interakti f, atau sesi untuk membuat terowongan SSH. Dokumen sesi didukung di semua Linux dan sistem operasi Windows Server yang didukung oleh Systems Manager. Dokumen-dokumen Perintah berikut didukung pada instans EC2 untuk macOS: • `AWS-PasswordReset` • `AWS-StartInteracti veCommand` • `AWS-StartPortForwa rdingSession` • `AWS-StartPortForwa rdingSessionToSock et` • `AWS-StartSSHSession`

Kuota dokumen SSM

Untuk informasi tentang kuota dokumen SSM, lihat kuota layanan Systems Manager di bagian. Referensi Umum Amazon Web Services

Topik

Komponen dokumen

Bagian ini mencakup informasi tentang komponen yang membentuk dokumen SSM.

Konten

Skema, fitur, dan contoh

Dokumen (SSM) AWS Systems Manager menggunakan versi skema berikut.

- Jenis dokumen `Command` dapat menggunakan skema versi 1.2, 2.0, dan 2.2. Jika Anda menggunakan skema dokumen 1.2, kami sarankan Anda membuat dokumen yang menggunakan skema versi 2.2.
- Jenis dokumen `Policy` harus menggunakan skema versi 2.0 atau yang lebih baru.
- Jenis dokumen `Automation` harus menggunakan skema versi 0.3.
- Anda dapat membuat dokumen di JSON atau YAML.

Dengan menggunakan versi skema terbaru untuk dokumen `Command` dan `Policy`, Anda dapat memanfaatkan fitur berikut.

Fitur dokumen skema versi 2.2

Fitur	Detail
Mengedit dokumen	Dokumen sekarang dapat diperbarui. Dengan versi 1.2, setiap update dokumen yang diperlukan yang Anda simpan dengan nama yang berbeda.

Fitur	Detail
Versioning otomatis	Setiap pembaruan ke dokumen menciptakan versi baru. Ini bukan versi skema, tetapi versi dokumen.
Versi default	Jika Anda memiliki beberapa versi dokumen, Anda dapat menentukan yang mana versi dokumen default.
Pengurutan	Plugin atau Langkah dalam dokumen yang dijalankan sesuai urutan yang Anda tentukan.
Dukungan lintas platform	Dukungan lintas platform mengizinkan Anda untuk menentukan sistem operasi yang berbeda untuk plugin yang berbeda dalam dokumen SSM yang sama. Dukungan lintas platform menggunakan parameter `precondition` dalam langkahnya.

> **Note**
>
> Anda harus AWS Systems Manager SSM Agent terus memperbarui instans dengan versi terbaru untuk menggunakan fitur Systems Manager dan fitur dokumen SSM yang baru. Untuk informasi selengkapnya, lihat Memperbarui SSM Agent penggunaan Run Command.

Tabel berikut mencantumkan perbedaan antara versi utama skema.

Versi 1.2	Versi 2.2 (versi terbaru)	Detail
runtimeConfig	mainSteps	Di versi 2.2, `mainSteps` menggantikan bagian `runtimeConfig`. Bagian `mainSteps` memungkinkan Systems Manager untuk

Versi 1.2	Versi 2.2 (versi terbaru)	Detail
		menjalankan langkah-langkah secara berurutan.
properti	masukan	Di versi 2.2, bagian `inputs` menggantikan bagian `properties` . Bagian `inputs` menerima langkah-l angkah parameter.
perintah	runCommand	Di versi 2.2,bagian `inputs` mengambil parameter `runCommand` bukan parameter `commands`.
id	tindakan	Versi 2.2, `Action` mengganti kan `ID`. Ini hanya perubahan nama.
tidak berlaku	nama	Versi 2.2, `name` adalah nama yang ditetapkan oleh pengguna untuk langkah.

Menggunakan parameter prasyarat

Dengan skema versi 2.2 atau yang lebih baru, Anda dapat menggunakan parameter `precondition` untuk menentukan target sistem operasi untuk setiap plugin atau untuk memvalidasi parameter input yang telah Anda tetapkan dalam dokumen SSM Anda. Parameter `precondition` mendukung referensi parameter input dokumen SSM Anda, dan `platformType` menggunakan nilai dari `Linux`, `MacOS`, dan `Windows`. Hanya `StringEquals` operator yang didukung.

Untuk dokumen yang menggunakan skema versi 2.2 atau yang terbaru, jika `precondition` tidak ditentukan, setiap plugin yang dijalankan atau dilewati berdasarkan kompatibilitas plugin dengan sistem operasi. Kompatibilitas plugin dengan sistem operasi dievaluasi sebelum `precondition`. Untuk dokumen yang menggunakan skema 2.0 atau sebelumnya, plugin yang tidak kompatibel akan membuang kesalahan.

Sebagai contoh, dalam dokumen skema versi 2.2, jika `precondition` tidak dispesifikasikan dan plugin `aws:runShellScript` yang terdaftar, maka langkah yang berjalan pada instans Linux, tetapi melewati sistem instans Windows Server karena `aws:runShellScript` tidak kompatibel dengan instans Windows Server. Namun, untuk dokumen skema versi 2.0, jika Anda menentukan plugin `aws:runShellScript`, dan kemudian menjalankan dokumen pada instans Windows Server, eksekusi akan gagal. Anda dapat melihat contoh parameter prasyarat dalam dokumen SSM nanti di bagian ini.

Skema versi 2.2

Elemen tingkat atas

Contoh berikut menunjukkan elemen-elemen tingkat atas dari dokumen SSM menggunakan skema versi 2.2.

YAML

```
---
schemaVersion: "2.2"
description: A description of the document.
parameters:
  parameter 1:
    property 1: "value"
    property 2: "value"
  parameter 2:
    property 1: "value"
    property 2: "value"
mainSteps:
  - action: Plugin name
    name: A name for the step.
    inputs:
      input 1: "value"
      input 2: "value"
      input 3: "{{ parameter 1 }}"
```

JSON

```
{
   "schemaVersion": "2.2",
   "description": "A description of the document.",
   "parameters": {
       "parameter 1": {
```

```
            "property 1": "value",
            "property 2": "value"
        },
        "parameter 2":{
            "property 1": "value",
            "property 2": "value"
        }
    },
   "mainSteps": [
      {
         "action": "Plugin name",
         "name": "A name for the step.",
         "inputs": {
            "input 1": "value",
            "input 2": "value",
            "input 3": "{{ parameter 1 }}"
         }
      }
   ]
}
```

Contoh skema versi 2.2

Contoh berikut menggunakan `aws:runPowerShellScript` plugin untuk menjalankan PowerShell perintah pada instance target.

YAML

```
---
schemaVersion: "2.2"
description: "Example document"
parameters:
  Message:
    type: "String"
    description: "Example parameter"
    default: "Hello World"
mainSteps:
  - action: "aws:runPowerShellScript"
    name: "example"
    inputs:
      timeoutSeconds: '60'
      runCommand:
```

```
      - "Write-Output {{Message}}"
```

JSON

```
{
   "schemaVersion": "2.2",
   "description": "Example document",
   "parameters": {
      "Message": {
         "type": "String",
         "description": "Example parameter",
         "default": "Hello World"
      }
   },
   "mainSteps": [
      {
         "action": "aws:runPowerShellScript",
         "name": "example",
         "inputs": {
            "timeoutSeconds": "60",
            "runCommand": [
               "Write-Output {{Message}}"
            ]
         }
      }
   ]
}
```

Skema versi 2.2 contoh parameter prasyarat

Skema versi 2.2 menyediakan dukungan lintas-platform. Ini berarti bahwa dalam satu dokumen SSM Anda dapat menentukan sistem operasi yang berbeda untuk plugin yang berbeda. Dukungan lintas platform dalam setiap langkah menggunakan parameter `precondition`, seperti yang ditunjukkan dalam contoh berikut. Anda juga dapat menggunakan parameter `precondition` untuk memvalidasi parameter input yang telah ditetapkan dalam dokumen SSM Anda. Anda dapat melihat ini di kedua contoh berikut.

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"
            ]
         }
      }
   ]
}
```

Contoh skema versi 2.2 State Manager

Anda dapat menggunakan dokumen SSM berikut denganState Manager, kemampuan Systems Manager, untuk mengunduh dan menginstal perangkat lunak antivirus ClamAV. State Managermemberlakukan konfigurasi tertentu, yang berarti bahwa setiap kali State Manager asosiasi dijalankan, sistem memeriksa untuk melihat apakah perangkat lunak ClamAV diinstal. Jika tidak, State Manager jalankan kembali dokumen ini.

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"
            ]
         }
      }
   ]
}
```

Contoh inventaris skema versi 2.2

Anda dapat menggunakan dokumen SSM berikut State Manager untuk mengumpulkan metadata inventaris tentang instans Anda.

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 }}"
            }
        }
    ]
}
```

Contoh skema versi 2.2 **`AWS-ConfigureAWSPackage`**

Contoh berikut menunjukkan dokumen `AWS-ConfigureAWSPackage`. Bagian `mainSteps` mencakup plugin `aws:configurePackage` di langkah `action`.

> **Note**
>
> Pada sistem operasi Linux, hanya paket `AmazonCloudWatchAgent` dan `AWSSupport-EC2Rescue` yang didukung.

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 }}"
         }
      }
   ]
}
```

Skema versi 1.2

Contoh berikut menunjukkan unsur-unsur tingkat atas dokumen skema versi 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"
            }
         ]
      }
   }
}
```

Contoh skema versi 1.2 **aws:runShellScript**

Contoh berikut menunjukkan Dokumen SSM `AWS-RunShellScript`. Bagian runtimeConfig mencakup plugin `aws:runShellScript`.

```
{
    "schemaVersion":"1.2",
    "description":"Run a shell script or specify the commands to run.",
    "parameters":{
        "commands":{
            "type":"StringList",
            "description":"(Required) Specify a shell script or a command to run.",
            "minItems":1,
            "displayType":"textarea"
```

```
        },
        "workingDirectory":{
            "type":"String",
            "default":"",
            "description":"(Optional) The path to the working directory on your
 instance.",
            "maxChars":4096
        },
        "executionTimeout":{
            "type":"String",
            "default":"3600",
            "description":"(Optional) The time in seconds for a command to complete
 before it is considered to have failed. Default is 3600 (1 hour). Maximum is 172800
 (48 hours).",
            "allowedPattern":"([1-9][0-9]{0,3})|(1[0-9]{1,4})|(2[0-7][0-9]{1,3})|
(28[0-7][0-9]{1,2})|(28800)"
        }
    },
    "runtimeConfig":{
        "aws:runShellScript":{
            "properties":[
                {
                    "id":"0.aws:runShellScript",
                    "runCommand":"{{ commands }}",
                    "workingDirectory":"{{ workingDirectory }}",
                    "timeoutSeconds":"{{ executionTimeout }}"
                }
            ]
        }
    }
}
```

Skema versi 0.3

Elemen tingkat atas

Contoh berikut menunjukkan elemen-elemen tingkat atas dari skema versi 0.3 runbook otomatisasi dalam format JSON.

```
{
    "description": "document-description",
    "schemaVersion": "0.3",
    "assumeRole": "{{assumeRole}}",
    "parameters": {
```

```
        "parameter1": {
            "type": "String",
            "description": "parameter-1-description",
            "default": ""
        },
        "parameter2": {
            "type": "String",
            "description": "parameter-2-description",
            "default": ""
        }
    },
    "variables": {
        "variable1": {
            "type": "StringMap",
            "description": "variable-1-description",
            "default": {}
        },
        "variable2": {
            "type": "String",
            "description": "variable-2-description",
            "default": "default-value"
        }
    },
    "mainSteps": [
        {
            "name": "myStepName",
            "action": "action-name",
            "maxAttempts": 1,
            "inputs": {
                "Handler": "python-only-handler-name",
                "Runtime": "runtime-name",
                "Attachment": "script-or-zip-name"
            },
            "outputs": {
                "Name": "output-name",
                "Selector": "selector.value",
                "Type": "data-type"
            }
        }
    ],
    "files": {
        "script-or-zip-name": {
            "checksums": {
                "sha256": "checksum"
```

```
            },
            "size": 1234
        }
    }
}
```

Contoh runbook otomatisasi YAML

Contoh berikut menunjukkan isi dari sebuah runbook otomatisasi, dalam format YAML. Contoh kerja ini dari skema dokumen versi 0.3 juga menunjukkan penggunaan Potongan harga untuk memformat deskripsi dokumen.

```
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
```

Elemen dan parameter data

Topik ini menjelaskan elemen data yang digunakan dalam dokumen SSM. Versi skema yang digunakan untuk membuat dokumen mendefinisikan sintaks dan elemen data yang diterima dokumen. Kami menyarankan Anda menggunakan skema versi 2.2 atau yang lebih baru untuk dokumen Command. Runbook otomatisasi menggunakan skema versi 0.3. Selain itu, runbook Otomatisasi mendukung penggunaan penurunan harga, bahasa markup, yang memungkinkan Anda menambahkan deskripsi gaya wiki ke dokumen dan langkah-langkah individual dalam dokumen. Untuk informasi selengkapnya tentang penggunaan Markdown, lihat Menggunakan Penurunan Harga di Konsol di Panduan AWS Management Console Memulai.

Bagian berikut menjelaskan elemen data yang dapat Anda sertakan dalam dokumen SSM.

Elemen data tingkat atas

schemaVersion

Versi skema untuk digunakan.

Jenis: Versi

Wajib: Ya

deskripsi

Informasi yang Anda berikan untuk menjelaskan tujuan dokumen. Anda juga dapat menggunakan bidang ini untuk menentukan apakah parameter memerlukan nilai untuk menjalankan dokumen, atau jika memberikan nilai untuk parameter adalah opsional. Parameter yang diperlukan dan opsional dapat dilihat pada contoh di seluruh topik ini.

Tipe: String

Wajib: Tidak

parameter

Struktur yang menentukankan parameter dokumen menerima.

Untuk parameter yang sering Anda gunakan, kami sarankan Anda menyimpan parameter tersebutParameter Store, kemampuan AWS Systems Manager. Kemudian, Anda dapat menentukan parameter dalam dokumen Anda yang mereferensikan Parameter Store parameter sebagai nilai defaultnya. Untuk referensi Parameter Store parameter, gunakan sintaks berikut.

```
{{ssm:parameter-name}}
```

Anda dapat menggunakan parameter yang mereferensikan Parameter Store parameter dengan cara yang sama seperti parameter dokumen lainnya. Dalam contoh berikut, nilai default untuk `commands` parameter adalah Parameter Store parameter`myShellCommands`. Dengan menentukan `commands` parameter sebagai `runCommand` string, dokumen menjalankan perintah yang disimpan dalam `myShellCommands` parameter.

YAML

```
---
schemaVersion: '2.2'
description: runShellScript with command strings stored as Parameter Store
 parameter
parameters:
  commands:
    type: StringList
    description: "(Required) The commands to run on the instance."
    default: ["{{ ssm:myShellCommands }}"]
mainSteps:
- action: aws:runShellScript
  name: runShellScriptDefaultParams
  inputs:
    runCommand:
    - "{{ commands }}"
```

JSON

```
{
    "schemaVersion": "2.2",
    "description": "runShellScript with command strings stored as Parameter Store
 parameter",
    "parameters": {
      "commands": {
        "type": "StringList",
        "description": "(Required) The commands to run on the instance.",
        "default": ["{{ ssm:myShellCommands }}"]
      }
    },
    "mainSteps": [
      {
        "action": "aws:runShellScript",
```

```
        "name": "runShellScriptDefaultParams",
        "inputs": {
            "runCommand": [
              "{{ commands }}"
          ]
        }
      }
    ]
  }
```

> **Note**
>
> Anda dapat referensi `String` dan `StringList` Parameter Store parameter di `parameters` bagian dokumen Anda. Anda tidak dapat mereferensikan `SecureString` Parameter Store parameter.

Untuk informasi selengkapnya tentang Parameter Store, lihat AWS Systems Manager Parameter Store.

Jenis: Struktur

`parameters` menerima bidang dan nilai-nilai berikut:

- `type`: Nilai yang (Diperlukan) diizinkan mencakup hal berikut: `String`, `StringList`, `Integer` `Boolean`, `MapList`, dan `StringMap`. Untuk melihat contoh dari setiap jenis, lihat Contoh parameter dokumen SSM `type` di bagian berikutnya.

  > **Note**
  >
  > Dokumen tipe perintah hanya mendukung tipe `String` dan `StringList` parameter.

- `description`: (Opsional) Deskripsi parameter.
- `default`: (Opsional) Nilai default parameter atau referensi ke parameter diParameter Store.
- `allowedValues`: (Opsional) Array nilai diperbolehkan untuk parameter. Menentukan nilai yang diperbolehkan untuk parameter memvalidasi input pengguna. Jika pengguna input nilai tidak diperbolehkan, eksekusi gagal untuk memulai.

  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`: (Opsional) Sebuah ekspresi reguler yang memvalidasi apakah input pengguna cocok dengan pola yang ditetapkan untuk parameter. Jika input pengguna tidak cocok dengan pola yang diperbolehkan, eksekusi gagal untuk memulai.

> **Note**
>
> Systems Manager melakukan dua validasi untuk. `allowedPattern` Validasi pertama dilakukan menggunakan pustaka regex Java di tingkat API saat Anda menggunakan dokumen. Validasi kedua dilakukan SSM Agent dengan menggunakan pustaka regexp GO sebelum memproses dokumen.

YAML

```
InstanceId:
  type: String
  description: "(Required) The instance ID to target."
  allowedPattern: "^i-[a-z0-9]{8,17}$"
  default: ''
```

JSON

```
"InstanceId": {
  "type": "String",
  "description": "(Required) The instance ID to target.",
  "allowedPattern": "^i-[a-z0-9]{8,17}$",
  "default": ""
}
```

- `displayType`: (Opsional) Digunakan untuk menampilkan baik a `textfield` atau a `textarea` di AWS Management Console. `textfield`adalah kotak teks satu baris. `textarea`adalah area teks multi-baris.
- `minItems`: (Opsional) Jumlah minimum item yang diperbolehkan.
- `maxItems`: (Opsional) Jumlah maksimum item yang diperbolehkan.
- `minChars`: (Opsional) Jumlah minimum karakter parameter yang diperbolehkan.
- `maxChars`: (Opsional) Jumlah maksimum karakter parameter yang diperbolehkan.

Diperlukan: Tidak

variabel

(Skema versi 0.3 saja) Nilai yang dapat Anda referensikan atau perbarui di seluruh langkah di runbook Otomasi. Variabel mirip dengan parameter, tetapi berbeda dalam cara yang sangat penting. Nilai parameter statis dalam konteks runbook, tetapi nilai variabel dapat diubah dalam konteks runbook. Saat memperbarui nilai variabel, tipe data harus cocok dengan tipe data yang ditentukan. Untuk informasi tentang memperbarui nilai variabel dalam otomatisasi, lihat [aws:updateVariable— Memperbarui nilai untuk variabel runbook]

Jenis: Boolean | Integer | | String MapList | | StringList StringMap

Diperlukan: Tidak

YAML

```
variables:
    payload:
        type: StringMap
        default: "{}"
```

JSON

```
{
    "variables": [
        "payload": {
            "type": "StringMap",
            "default": "{}"
        }
    ]
}
```

runtimeConfig

(Hanya skema versi 1.2) Konfigurasi untuk instans seperti yang diterapkan oleh satu atau beberapa plugin Systems Manager. Plugin tidak dijamin untuk dapat berjalan secara berurutan.

Jenis: Kamus <String, > PluginConfiguration

Diperlukan: Tidak

mainSteps

(Hanya skema versi 0.3, 2.0, dan 2.2) Sebuah objek yang dapat mencakup beberapa langkah (plugin). Plugin ditentukan dalam langkah-langkah. Langkah-langkah yang dijalankan secara berurutan seperti yang tercantum dalam dokumen.

Jenis: Kamus <String, > PluginConfiguration

Diperlukan: Ya

keluaran

(Hanya skema versi 0.3) Data yang dihasilkan oleh eksekusi dokumen ini yang dapat digunakan dalam proses lainnya. Misalnya, jika dokumen Anda membuat yang baruAMI, Anda dapat menentukan "CreateImage. ImageId"sebagai nilai output, dan kemudian menggunakan output ini untuk membuat instance baru dalam eksekusi otomatisasi berikutnya. Untuk informasi selengkapnya tentang opsi, lihat Menggunakan output tindakan sebagai input.

Jenis: Kamus <String, > OutputConfiguration

Diperlukan: Tidak

file

(Hanya skema versi 0.3) File skrip (dan checksum mereka) dilampirkan pada dokumen dan dijalankan selama eksekusi otomatisasi. Hanya berlaku untuk dokumen yang mencakup tindakan `aws:executeScript` dan lampiran yang telah ditentukan dalam satu atau beberapa langkah.

Untuk dukungan runtime skrip, Runbook Otomasi mendukung skrip untuk Python 3.7, Python 3.8, Core 6.0, dan 7.0. PowerShell PowerShell Untuk informasi selengkapnya tentang termasuk skrip di runbook otomatisasi, lihat Menggunakan skrip di runbook dan Menggunakan Document Builder untuk membuat runbook.

Saat membuat runbook Otomasi dengan lampiran, Anda juga harus menentukan file lampiran menggunakan `--attachments` opsi (untuk AWS CLI) atau `Attachments` (untuk API dan SDK). Anda dapat menentukan lokasi file untuk kedua file lokal dan file yang disimpan di bucket Amazon Simple Storage Service (Amazon S3). Untuk informasi selengkapnya, lihat Lampiran di Referensi AWS Systems Manager API.

YAML

```
---
files:
  launch.py:
    checksums:
      sha256: 18871b1311b295c43d0f...
[truncated]...772da97b67e99d84d342ef4aEXAMPLE
```

JSON

```
"files": {
    "launch.py": {
        "checksums": {
            "sha256": "18871b1311b295c43d0f...
[truncated]...772da97b67e99d84d342ef4aEXAMPLE"
        }
    }
}
```

Jenis: Kamus <String, > FilesConfiguration

Diperlukan: Tidak

Contoh parameter dokumen SSM **`type`**

Jenis parameter dalam dokumen SSM adalah statis. Ini berarti jenis parameter tidak dapat diubah setelah ditentukan. Ketika menggunakan parameter dengan plugin dokumen SSM, jenis parameter tidak dapat diubah secara dinamis dalam input plugin. Misalnya, Anda tidak dapat mereferensikan parameter `Integer` dalam input `runCommand` dari plugin `aws:runShellScript` karena input ini menerima string atau daftar string. Untuk menggunakan parameter pada input plugin, jenis parameter harus sepadan dengan jenis yang diterima. Misalnya, Anda harus menentukan jenis parameter `Boolean` untuk input `allowDowngrade` dari plugin `aws:updateSsmAgent`. Jika jenis parameter Anda tidak cocok dengan jenis input untuk plugin, dokumen SSM gagal untuk memvalidasi dan sistem tidak dapat membuat dokumen. Ini juga berlaku saat menggunakan parameter hilir dalam input untuk plugin lain atau AWS Systems Manager tindakan Otomasi. Misalnya, Anda tidak dapat mereferensikan `StringList` parameter dalam `documentParameters` input `aws:runDocument` plugin. `documentParameters`Input menerima peta string meskipun tipe parameter dokumen SSM hilir adalah parameter dan cocok dengan `StringList` parameter yang Anda referensikan.

Saat menggunakan parameter dengan Tindakan otomatisasi , jenis parameter tidak divalidasi saat Anda membuat dokumen SSM dalam banyak kasus. Hanya ketika Anda menggunakan tindakan `aws:runCommand` jenis parameter saat Anda membuat dokumen SSM divalidasi. Dalam semua kasus lain, validasi parameter terjadi selama eksekusi otomatisasi ketika input tindakan diverifikasi sebelum menjalankan tindakan. Misalnya, dokumen SSM dibuat jika parameter input Anda adalah `String` dan referensi Anda sebagai nilai untuk input `MaxInstanceCount` dari tindakan `aws:runInstances`. Namun, ketika menjalankan dokumen, otomatisasi gagal sementara memvalidasi Tindakan `aws:runInstances` karena input `MaxInstanceCount` memerlukan `Integer`.

Berikut ini adalah contoh dari setiap parameter `type`.

Tali

Urutan karakter Unicode nol atau lebih dalam tanda kutip. Misalnya, "i-1234567890abcdef0". Gunakan garis miring terbalik untuk keluar.

YAML

```
---
InstanceId:
  type: String
  description: "(Optional) The target EC2 instance ID."
```

JSON

```
"InstanceId":{
  "type":"String",
  "description":"(Optional) The target EC2 instance ID."
}
```

StringList

Daftar item String dipisahkan dengan koma. Sebagai contoh, ["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"
}
```

Boolean

Hanya menerima `true` atau `false`. Tidak menerima "benar" atau 0.

YAML

```
---
canRun:
  type: Boolean
  description: ''
  default: true
```

JSON

```
"canRun": {
  "type": "Boolean",
  "description": "",
  "default": true
}
```

Bulat

Nomor integral. Tidak menerima angka desimal, misalnya 3.14159, atau angka yang dalam tanda kutip, misalnya "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

Sebuah pemetaan kunci untuk nilai-nilai. Kunci dan nilai harus berupa string. Misalnya, {"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

Daftar StringMap objek.

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
}
```

Melihat konten dokumen SSM Command

Untuk melihat pratinjau parameter yang diperlukan dan opsional untuk dokumen Command AWS Systems Manager (SSM), selain tindakan yang dijalankan dokumen, Anda dapat melihat konten dokumen di konsol Systems Manager.

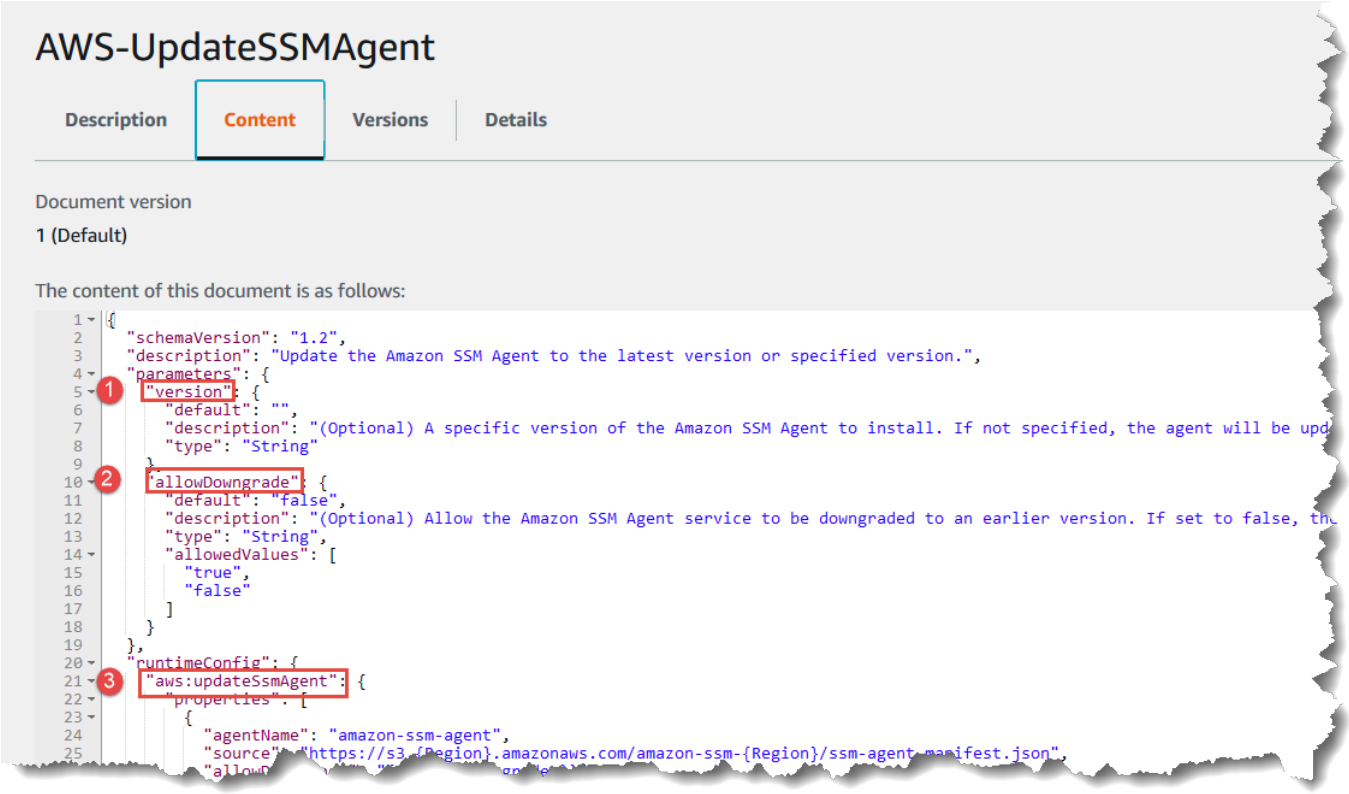
Untuk melihat konten dokumen SSM Command

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.
3. Di kotak pencarian, pilih Jenis dokumen, dan kemudian pilih Perintah.
4. Pilih nama dokumen, dan kemudian pilih tab Konten.
5. Di bidang konten, tinjau parameter dan langkah-langkah tindakan yang tersedia untuk dokumen.

   Misalnya, gambar berikut menunjukkan bahwa (1) `version` dan (2) `allowDowngrade` adalah parameter opsional untuk dokumen `AWS-UpdateSSMAgent`, dan bahwa tindakan pertama yang dijalankan oleh dokumen adalah (3) `aws:updateSsmAgent`.



Referensi plugin dokumen perintah

Referensi ini menjelaskan plugin yang dapat Anda tentukan dalam dokumen tipe Command AWS Systems Manager (SSM). Plugin ini tidak dapat digunakan dalam runbook otomatisasi SSM, yang menggunakan tindakan otomatisasi. Untuk informasi tentang tindakan AWS Systems Manager Otomasi, lihatReferensi tindakan Otomatisasi Systems Manager.

Systems Manager menentukan tindakan untuk menjalankan instans terkelola dengan membaca isi dokumen SSM. Setiap dokumen termasuk bagian eksekusi kode. Tergantung pada versi skema dokumen Anda, bagian eksekusi kode ini dapat mencakup satu atau beberapa plugin atau langkah-langkah. Untuk tujuan topik bantuan ini, plugin dan langkah-langkah disebut sebagai plugin. Bagian ini mencakup informasi tentang setiap plugin Systems Manager. Untuk informasi lebih lanjut tentang dokumen, termasuk informasi tentang membuat dokumen dan perbedaan antara versi skema, lihat AWS Systems ManagerDokumen.

> **Note**
>
> Beberapa plugin yang dijelaskan di sini hanya dijalankan di instans Windows Server atau instans Linux. Dependensi platform dicatat dalam setiap plugin.

Plugin dokumen berikut didukung oleh instans Amazon Elastic Compute Cloud (Amazon EC2) untuk macOS:

- `aws:refreshAssociation`
- `aws:runShellScript`
- `aws:runPowerShellScript`
- `aws:softwareInventory`
- `aws:updateSsmAgent`

Daftar Isi

Input bersama

Dengan SSM Agent versi 3.0.502 dan yang lebih baru saja, semua plugin dapat menggunakan input berikut:

finallyStep

Langkah terakhir untuk menjalankan dokumen yang Anda inginkan. Jika input ini ditentukan dalam sebuah langkah, `exit` lebih diutamakan dari nilai yang ditentukan dalam `onFailure` atau input `onSuccess`. Dalam rangka untuk menjalankan langkah input ini seperti yang diharapkan, langkah terakhir yang harus ditentukan dalam `mainSteps` dari dokumen Anda.

Tipe: Boolean

Nilai yang valid: `true` | `false`

Wajib: Tidak

onFailure

Jika Anda menentukan input ini untuk plugin dengan nilai `exit` dan langkah gagal, status langkah mencerminkan kegagalan dan dokumen tidak menjalankan langkah-langkah yang tersisa kecuali `finallyStep` yang telah ditentukan. Jika Anda menentukan input ini untuk plugin dengan nilai `successAndExit` dan langkah gagal, status langkah menunjukkan sukses dan dokumen tidak menjalankan langkah-langkah yang tersisa kecuali `finallyStep` yang telah ditentukan.

Tipe: String

Nilai yang valid: `exit` | `successAndExit`

Wajib: Tidak

onSuccess

Jika Anda menentukan input ini untuk plugin dan langkah yang dijalankan berhasil, dokumen tidak menjalankan langkah-langkah yang tersisa kecuali `finallyStep` yang telah ditentukan.

Tipe: String

Nilai valid: `exit`

Wajib: Tidak

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

Menginstal, memperbaiki, atau menghapus aplikasi pada instans EC2. Plugin ini hanya berjalan pada Sistem operasi Windows Server.

Sintaks

Skema 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 }}"
      }
    }
  ]
}
```

Skema 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 }}"
            }
         ]
      }
   }
}
```

Properti

tindakan

Tindakan yang harus diambil.

Jenis: Enum

Nilai valid: `Install` | `Repair` | `Uninstall`

Wajib: Ya

parameter

Parameter untuk penginstal.

Tipe: String

Wajib: Tidak

sumber

URL dari file `.msi` untuk aplikasi.

Tipe: String

Wajib: Ya

sourceHash

The SHA256 hash dari file `.msi`.

Tipe: String

Wajib: Tidak

`aws:cloudWatch`

Ekspor data dari Windows Server ke Amazon CloudWatch atau Amazon CloudWatch Logs dan pantau data menggunakan CloudWatch metrik. Plugin ini hanya berjalan pada Sistem operasi Windows Server. Untuk informasi selengkapnya tentang mengonfigurasi CloudWatch integrasi dengan Amazon Elastic Compute Cloud (Amazon EC2), lihat Mengumpulkan metrik dan log dari instans Amazon EC2 dan server lokal dengan agen. CloudWatch

> **Important**
>
> CloudWatch Agen terpadu telah diganti SSM Agent sebagai alat untuk mengirim data log ke Amazon CloudWatch Logs. Plugin SSM Agent AWS:CloudWatch tidak didukung. Sebaiknya gunakan hanya CloudWatch agen terpadu untuk proses pengumpulan log Anda. Untuk informasi selengkapnya, lihat topik berikut:
>
> - Mengirim log simpul ke CloudWatch Log terpadu (CloudWatch agen)
> - Migrasikan koleksi log node Windows Server ke agen CloudWatch
> - Mengumpulkan metrik dan log dari instans Amazon EC2 dan server lokal dengan CloudWatch agen di Panduan Pengguna Amazon. CloudWatch

Anda dapat mengekspor dan memantau jenis data berikut:

ApplicationEventLog

Mengirim data log peristiwa aplikasi ke CloudWatch Log.

CustomLogs

Mengirim file log berbasis teks apa pun ke Amazon CloudWatch Logs. CloudWatch Plugin membuat sidik jari untuk file log. Sistem kemudian menghubungkan data offset dengan setiap sidik jari. Plugin mengunggah file ketika ada perubahan, rekaman offset, dan menghubungkan offset dengan sidik jari. Metode ini digunakan untuk menghindari situasi di mana pengguna menyalakan plugin, mengaitkan layanan dengan direktori yang berisi sebagian besar file, dan sistem mengunggah semua file.

> **Warning**
>
> Sadarilah bahwa jika aplikasi Anda memotong atau mencoba untuk membersihkan log selama polling, setiap log yang ditentukan untuk `LogDirectoryPath` dapat kehilangan entri. Jika, misalnya, Anda ingin membatasi ukuran file log, membuat file log baru ketika batas telah tercapai, dan kemudian melanjutkan menulis data ke file baru.

ETW

Mengirim data Event Tracing untuk Windows (ETW) ke CloudWatch Log.

IIS

Mengirim data log IIS ke CloudWatch Log.

PerformanceCounter

Mengirim penghitung kinerja Windows ke CloudWatch. Anda dapat memilih kategori yang berbeda untuk diunggah CloudWatch sebagai metrik. Untuk setiap penghitung kinerja yang ingin Anda unggah, buat PerformanceCounterbagian dengan ID unik (misalnya, "PerformanceCounter2"," PerformanceCounter 3", dan seterusnya) dan konfigurasikan propertinya.

> **Note**
>
> Jika AWS Systems Manager SSM Agent atau CloudWatch plugin dihentikan, data penghitung kinerja tidak masuk CloudWatch. Perilaku ini berbeda dari log kustom atau log Peristiwa Windows. Log kustom dan log Peristiwa Windows mempertahankan data penghitung kinerja dan mengunggahnya ke CloudWatch setelah SSM Agent atau CloudWatch plugin tersedia.

SecurityEventLog

Mengirim data log peristiwa keamanan ke CloudWatch Log.

SystemEventLog

Mengirim data log peristiwa sistem ke CloudWatch Log.

Anda dapat menentukan tujuan untuk data berikut:

CloudWatch

Tujuan pengiriman data metrik penghitung kinerja Anda. Anda dapat menambahkan lebih banyak bagian dengan ID unik (misalnya, "CloudWatch2", CloudWatch 3", dan seterusnya), dan menentukan Wilayah yang berbeda untuk setiap ID baru untuk mengirim data yang sama ke lokasi yang berbeda.

CloudWatchLogs

Tujuan dimana data log Anda dikirim. Anda dapat menambahkan lebih banyak bagian dengan ID unik (misalnya, "CloudWatchLogs2", CloudWatchLogs 3", dan seterusnya), dan menentukan Wilayah yang berbeda untuk setiap ID baru untuk mengirim data yang sama ke lokasi yang berbeda.

Sintaks

```
"runtimeConfig":{
        "aws:cloudWatch":{
            "settings":{
                "startType":"{{ status }}"
            },
            "properties":"{{ properties }}"
        }
    }
```

Pengaturan dan properti

AccessKey

Kunci akses ID Anda. Properti ini diperlukan kecuali Anda telah meluncurkan instans Anda menggunakan IAM role. Properti ini tidak dapat digunakan dengan SSM.

Tipe: String

Wajib: Tidak

CategoryName

Kategori penghitungan kinerja dari Pemantauan kinerja.

Tipe: String

Diperlukan: Ya

CounterName

Nama penghitung kinerja dari Pemantauan kinerja.

Tipe: String

Diperlukan: Ya

CultureName

Lokal tempat stempel waktu dicatat. Jika CultureNamekosong, defaultnya ke lokal yang sama yang digunakan oleh instance Anda. Windows Server

Jenis: String

Nilai yang benar: Untuk daftar nilai yang mendukung, lihat National Language Support (NLS) di situs web Microsoft. Div, div-MV, hu, dan Hu-hu nilai tidak mendukung.

Diperlukan: Tidak

DimensionName

Dimensi untuk CloudWatch metrik Amazon Anda. Jika Anda menentukan `DimensionName`, Anda harus menentukan `DimensionValue`. Parameter ini memberikan tampilan lain saat mendaftar metrik. Anda dapat menggunakan dimensi yang sama untuk beberapa metrik sehingga Anda dapat melihat semua metrik milik dimensi tertentu.

Tipe: String

Wajib: Tidak

DimensionValue

Nilai dimensi untuk CloudWatch metrik Amazon Anda.

Tipe: String

Wajib: Tidak

Encoding

File pengodean yang akan digunakan (misalnya, UTF-8). Gunakan nama pengodean, bukan nama tampilan.

Jenis: String

Nilai yang valid: Untuk daftar nilai yang didukung, lihat Kelas Pengkodean di Perpustakaan Microsoft Learn.

Diperlukan: Ya

Filter

Prefiks nama log. Biarkan parameter ini kosong untuk memantau semua file.

Jenis: String

Nilai yang valid: Untuk daftar nilai yang didukung, lihat FileSystemWatcherFilter Properti di Pustaka MSDN.

Diperlukan: Tidak

Alur

Setiap tipe data yang akan diunggah, bersama dengan tujuan untuk data (CloudWatch atau CloudWatch Log). Misalnya, untuk mengirim penghitung kinerja yang ditentukan di bawah `"Id": "PerformanceCounter"` ke CloudWatch tujuan yang ditentukan di bawah`"Id": "CloudWatch"`, masukkan "PerformanceCounter,CloudWatch". Demikian pula, untuk mengirim log kustom, log ETW, dan log sistem ke tujuan Log yang CloudWatch ditentukan di bawah`"Id": "ETW"`, masukkan “(ETW), CloudWatchLogs”. Selain itu, Anda dapat mengirim penghitung kinerja yang sama atau berkas log ke lebih dari satu tujuan. Misalnya, untuk mengirim log aplikasi ke dua tujuan berbeda yang Anda tentukan di bawah `"Id": "CloudWatchLogs"` dan`"Id": "CloudWatchLogs2"`, masukkan "ApplicationEventLog, (CloudWatchLogs, CloudWatchLogs 2)”.

Jenis: String

Nilai yang valid (sumber): `ApplicationEventLog` | `CustomLogs` | `ETW` | `PerformanceCounter` | `SystemEventLog` | `SecurityEventLog`

Nilai yang valid (tujuan): `CloudWatch` | `CloudWatchLogs` | `CloudWatch`*n* | `CloudWatchLogs`*n*

Diperlukan: Ya

FullName

Nama lengkap pada komponen.

Tipe: String

Wajib: Ya

Id

Mengidentifikasi sumber data atau tujuan. Pengidentifikasi ini harus unik dalam file konfigurasi.

Tipe: String

Diperlukan: Ya

InstanceName

Nama penghitung kinerja instans. Jangan gunakan tanda bintang (*) untuk menunjukkan semua instans karena setiap komponen penghitung kinerja hanya mendukung satu metrik. Anda bisa, namun menggunakan _Total.

Tipe: String

Wajib: Ya

Tingkat

Jenis pesan untuk dikirim ke Amazon CloudWatch.

Jenis: String

Nilai valid:

- 1 - Hanya pesan kesalahan yang diunggah.
- 2 - Hanya pesan peringatan yang diunggah.
- 4 - Hanya pesan informasi yang diunggah.

Anda dapat menambahkankan nilai-nilainya bersamaan untuk menyertakan lebih dari satu jenis pesan. Misalnya, 3 berarti bahwa pesan kesalahan (1) dan pesan peringatan (2) disertakan.

Nilai dari 7 berarti bahwa pesan kesalahan (1), pesan peringatan (2), dan pesan informasi (4) disertakan.

Wajib: Ya

> **Note**
>
> Log keamanan Windows harus diatur pada tingkat ke 7.

LineCount

Jumlah baris di header untuk mengidentifikasi berkas log. Sebagai contoh, berkas log IIS memiliki header yang hampir identik. Anda bisa memasukkan 3, yang akan membaca tiga baris pertama header berkas log untuk mengidentifikasinya. Dalam file log IIS, baris ketiga adalah tanggal dan stempel waktu, yang berbeda antara file log.

Jenis: Integer

Wajib: Tidak

LogDirectoryPath

Untuk CustomLogs, jalur tempat log disimpan di instans EC2 Anda. *Untuk log IIS, folder tempat log IIS disimpan untuk situs individual (misalnya, **C:\\ inetpub\\ logs** W3SVC n). LogFiles* Untuk log IIS, hanya format log W3C yang mendukung. IIS, NCSA, dan format kustom yang tidak mendukung.

Tipe: String

Diperlukan: Ya

LogGroup

Nama untuk grup log Anda. Nama ini ditampilkan di layar Grup Log di CloudWatch konsol.

Tipe: String

Diperlukan: Ya

LogName

Nama file log.

1. Untuk menemukan nama log, di Event Viewer, di panel navigasi, pilih Log Aplikasi dan Layanan.
2. Dalam daftar log, klik kanan log yang ingin Anda upload (misalnya, Microsoft>Windows>Backup>Operasional), dan kemudian pilih Buat Tampilan Khusus.
3. Di Buat Tampilan Khusus kotak dialog, pilih XML tab. LogNameAda di tag <Select Path=> (misalnya,). `Microsoft-Windows-Backup` Salin teks ini ke dalam LogNameparameter.

Jenis: String

Nilai valid: `Application` | `Security` | `System` | `Microsoft-Windows-WinINet/Analytic`

Diperlukan: Ya

LogStream

Pengaliran log tujuan. Jika Anda menggunakan {instance_id}, yaitu default-nya, instans ID pada instans ini digunakan sebagai nama stream log.

Tipe: String

Nilai valid: `{instance_id}` | `{hostname}` | `{ip_address}` *`<log_stream_name>`*

Jika Anda memasukkan nama aliran log yang belum ada, CloudWatch Log secara otomatis membuatnya untuk Anda. Anda dapat menggunakan string literal atau variabel yang telah ditetapkan ({instance_id}, {hostname}, {ip_address}, atau kombinasi dari ketiga untuk menentukan nama pengaliran log.

Nama aliran log yang ditentukan dalam parameter ini ditampilkan di Grup Log > Aliran untuk ***< YourLogStream >*** layar di CloudWatch konsol.

Diperlukan: Ya

MetricName

CloudWatch Metrik yang Anda inginkan data kinerja disertakan di bawah.

> **Note**
>
> Jangan gunakan karakter khusus dalam nama. Jika Anda melakukannya, metrik dan alarm terkait mungkin tidak berfungsi.

Tipe: String

Diperlukan: Ya

NameSpace

Namespace metrik di mana Anda ingin penulisan data penghitungan kinerja.

Tipe: String

Diperlukan: Ya

PollInterval

Berapa detik berlalu sebelum penghitung kinerja baru dan data log diunggah.

Jenis: Integer

Nilai valid: Atur ini dalam 5 detik atau lebih. Disarankan lima belas detik (00:00:15).

Wajib: Ya

Wilayah

Wilayah AWS Tempat Anda ingin mengirim data log. Meskipun Anda dapat mengirim penghitung kinerja ke Wilayah yang berbeda dari tempat Anda mengirim data log Anda, kami sarankan Anda menetapkan parameter ini untuk Wilayah yang sama di mana instans Anda berjalan.

Jenis: String

Nilai yang valid: ID Wilayah yang Wilayah AWS didukung oleh Systems Manager dan CloudWatch Log, seperti`us-east-2`,`eu-west-1`, dan`ap-southeast-1`. Untuk daftar yang Wilayah AWS didukung oleh setiap layanan, lihat Titik Akhir Layanan Amazon CloudWatch Logs dan titik akhir layanan Systems Manager di. Referensi Umum Amazon Web Services

Diperlukan: Ya

SecretKey

Kunci akses rahasia Anda. Properti ini diperlukan kecuali Anda telah meluncurkan instans Anda menggunakan IAM role.

Tipe: String

Wajib: Tidak

startType

Nyalakan atau matikan CloudWatch pada instance.

Tipe: String

Nilai yang valid: `Enabled` | `Disabled`

Diperlukan: Ya

TimestampFormat

Format waktu yang ingin Anda gunakan. Untuk daftar nilai yang mendukung, lihat Custom Date and Time Format Strings di Perpustakaan MSDN.

Tipe: String

Diperlukan: Ya

TimeZoneKind

Memberikan informasi zona waktu ketika tidak ada informasi zona waktu yang disertakan dalam stempel waktu log Anda. Jika parameter ini dibiarkan kosong dan jika stempel waktu Anda tidak menyertakan informasi zona waktu, CloudWatch Log default ke zona waktu lokal. Parameter ini diabaikan jika stempel waktu Anda sudah berisi informasi zona waktu.

Tipe: String

Nilai yang valid: `Local` | `UTC`

Wajib: Tidak

Unit

Unit pengukuran yang tepat untuk metrik.

Tipe: String

Nilai valid: Detik | Mikrodetik | Milidetik | Byte | Kilobit | Megabit | Gigabit | Terabit | Bit | Kilobit | Megabit | Gigabit | Terabit | Persen | Hitung | Byte/Detik | Kilobit/Detik | Megabit/Detik | Gigabit/Detik | Terabit/Detik | Bit/Detik | Kilobit/Detik | Megabit/Detik | Gigabit/Detik | Terabits/Kedua | Hitung/Detik | Tidak ada

Wajib: Ya

`aws:configureDocker`

(Skema versi 2.0 atau yang lebih baru) Mengkonfigurasi instans untuk bekerja dengan kontainer dan Docker. Plugin ini didukung oleh Linux dan Sistem operasi Windows Server.

Sintaks

Skema 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 }}"
      }
    }
  ]
}
```

Masukan

tindakan

Jenis tindakan yang harus dilakukan.

Jenis: Enum

Nilai valid: `Install` | `Uninstall`

Diperlukan: Ya

`aws:configurePackage`

(Skema versi 2.0 atau yang lebih baru) Instal atau hapus instalan paket. AWS Systems Manager Distributor Anda dapat menginstal versi terbaru, versi default, atau versi paket yang Anda tentukan. Paket yang AWS disediakan juga didukung. Plugin ini menjalankan Windows Server dan sistem operasi Linux, namun tidak semua paket yang tersedia didukung pada sistem operasi Linux.

AWS Paket yang tersedia untuk Windows Server meliputi:`AWSPVDriver`,,`AWSNVMe`,`AwsEnaNetworkDriver`,`AwsVssComponents`,`AmazonCloudWatchAge` dan `AWSSupport-EC2Rescue.`

AWS Paket yang tersedia untuk sistem operasi Linux meliputi:`AmazonCloudWatchAgent`,`CodeDeployAgent`, dan`AWSSupport-EC2Rescue`.

Sintaks

Skema 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\"}"
            }
         }
      ]
}
```

Masukan

name

Nama AWS paket untuk menginstal atau menghapus instalasi. Paket-paket yang tersedia meliputi: `AWSPVDriver`, `AwsEnaNetworkDriver`, `AwsVssComponents`, dan `AmazonCloudWatchAgent`.

Tipe: String

Wajib: Ya

tindakan

Menginstal atau menghapus paket.

Jenis: Enum

Nilai valid: `Install` | `Uninstall`

Wajib: Ya

installationType

Jenis instalasi untuk menjalankan. Jika Anda menentukan `Uninstall and reinstall`, paket benar-benar dihapus, dan kemudian diinstal ulang. Aplikasi ini tidak tersedia sampai penginstalan ulang selesai. Jika Anda menentukan `In-place update`, hanya file baru atau diubah yang ditambahkan ke instalasi yang ada sesuai dengan petunjuk yang Anda berikan dalam skrip pembaruan. Aplikasi tetap tersedia selama proses pembaruan. `In-place update`Opsi ini tidak didukung untuk paket AWS-published. `Uninstall and reinstall`adalah nilai default.

Jenis: Enum

Nilai valid: `Uninstall and reinstall` | `In-place update`

Wajib: Tidak

additionalArguments

String JSON dari parameter tambahan yang akan disediakan untuk menginstal, menghapus instalasi, atau memperbarui skrip Anda. Setiap parameter harus diawali dengan `SSM_`. Anda dapat mereferensikan Parameter Store parameter dalam argumen tambahan Anda dengan menggunakan konvensi`{{ssm:`*`parameter-name`*`}}`. Untuk menggunakan parameter tambahan dalam skrip install, uninstall, atau update, referensi parameter Anda harus sebagai variabel lingkungan menggunakan sintaks yang sesuai untuk sistem operasi. Misalnya, di PowerShell, Anda mereferensikan `SSM_arg` argumen sebagai`$Env:SSM_arg`. Tidak ada batas jumlah argumen yang Anda tentukan, tetapi input argumen tambahan memiliki batas karakter 4096. Batas ini mencakup semua kunci dan nilai yang Anda tentukan.

Jenis: StringMap

Diperlukan: Tidak

versi

Versi tertentu dari paket untuk menginstal atau menghapus instalasi. Jika memasang, sistem memasang versi terbaru yang diterbitkan, secara default. Jika menghapus instalasi, sistem menghapus instalasi versi yang dipasang pada masa ini, secara default. Jika tidak ada versi yang diinstal ditemukan, versi terbaru yang dipublikasi telah diunduh, dan tindakan uninstall dijalankan.

Tipe: String

Wajib: Tidak

`aws:domainJoin`

Bergabung dengan instans EC2 untuk domain. Plugin ini berjalan di Linux dan Sistem operasi Windows Server. *Plugin ini mengubah nama host untuk instance Linux ke format EC2AMAZ- XXXXXXX.* Untuk informasi selengkapnya tentang bergabung dengan instans EC2, lihat Menggabungkan Instans EC2 ke Direktori AWS Microsoft AD Terkelola Anda di Panduan Administrasi AWS Directory Service .

Sintaks

Skema 2.2

YAML

```
---
schemaVersion: '2.2'
description: aws:domainJoin
parameters:
  directoryId:
    description: "(Required) The ID of the directory."
    type: String
  directoryName:
    description: "(Required) The name of the domain."
    type: String
  directoryOU:
    description: "(Optional) The organizational unit to assign the computer object
 to."
    type: String
  dnsIpAddresses:
    description: "(Required) The IP addresses of the DNS servers for your
 directory."
```

```
    type: StringList
mainSteps:
- action: aws:domainJoin
  name: domainJoin
  inputs:
    directoryId: "{{ directoryId }}"
    directoryName: "{{ directoryName }}"
    directoryOU: "{{ directoryOU }}"
    dnsIpAddresses: "{{ dnsIpAddresses }}"
```

JSON

```
{
  "schemaVersion": "2.2",
  "description": "aws:domainJoin",
  "parameters": {
    "directoryId": {
      "description": "(Required) The ID of the directory.",
      "type": "String"
    },
    "directoryName": {
      "description": "(Required) The name of the domain.",
      "type": "String"
    },
    "directoryOU": {
        "description": "(Optional) The organizational unit to assign the computer
 object to.",
       "type": "String"
     },
    "dnsIpAddresses": {
      "description": "(Required) The IP addresses of the DNS servers for your
 directory.",
      "type": "StringList"
    },
  },
  "mainSteps": [
    {
      "action": "aws:domainJoin",
      "name": "domainJoin",
      "inputs": {
        "directoryId": "{{ directoryId }}",
        "directoryName": "{{ directoryName }}",
        "directoryOU":"{{ directoryOU }}",
```

```
        "dnsIpAddresses":"{{ dnsIpAddresses }}"
      }
    }
  ]
}
```

Skema 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 }}"
         }
      }
   }
}
```

Properti

directoryId

ID direktori.

Tipe: String

Wajib: Ya

Contoh: "directoryId": "d-1234567890"

directoryName

Nama domain.

Tipe: String

Wajib: Ya

Contoh: "directoryName": "example.com"

directoryOU

Unit organisasi (OU).

Tipe: String

Wajib: Tidak

Contoh: "directoryOU": "OU=test,DC=example,DC=com"

dnsIpAddresses

Alamat IP dari server DNS.

Jenis: StringList

Diperlukan: Ya

Contoh: "dnsIpAddresses“: [" 198.51.100.1", "198.51.100.2"]

Contoh-contoh

Sebagai contoh, lihat Menggabungkan Instans Amazon EC2 ke AWS Managed Microsoft AD dalam Panduan AWS Directory Service Administrasi.

`aws:downloadContent`

(Skema versi 2.0 atau yang lebih baru) Unduh dokumen dan skrip SSM dari lokasi terpencil. GitHub Enterpriserepositori tidak didukung. Plugin ini didukung oleh Linux dan Sistem operasi Windows Server.

Sintaks

Skema 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 }}"
      }
    }
  ]
}
```

Masukan

sourceType

Sumber unduhan. Systems Manager mendukung jenis sumber berikut untuk men-download skrip dan dokumen SSM: `GitHub`, `Git`, `HTTP`, `S3`, dan `SSMDocument`.

Tipe: String

Wajib: Ya

sourceInfo

Informasi yang diperlukan untuk mengambil konten dari sumber yang diperlukan.

Jenis: StringMap

Diperlukan: Ya

Untuk SourceType tentukan yang **`GitHub,`** berikut:

- owner: Pemilik repositori.
- repositori: Nama repositori.
- jalur: Jalur ke file atau direktori yang ingin Anda unduh.
- getOptions: Pilihan tambahan untuk mengambil konten dari cabang selain cabang utama atau dengan komit repositori tertentu. getOptions dapat dihilangkan jika Anda menggunakan komit terbaru di cabang utama. Jika repositori Anda dibuat setelah 1 Oktober 2020 cabang default mungkin diberi nama main (utama) dan bukan master (utama). Dalam hal ini, Anda perlu menentukan nilai untuk parameter GetOptions.

  Parameter ini menggunakan format berikut:

  - *`cabang: refs/kepala/branch_name`*

Nilai default-nya `master`.

Untuk menentukan cabang non-default gunakan format berikut:

cabang: refs/kepala/branch_name

- commitID: *commitID*

  Default-nya adalah `head`.

  Untuk menggunakan versi dokumen SSM Anda di komit selain yang terbaru, tentukan ID komit penuh. Sebagai contoh:

```
"getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",
```

- TokenInfo: Parameter Systems Manager (SecureStringparameter) tempat Anda menyimpan informasi token akses GitHub Anda, dalam format. {{ssm-secure:*secure-string-token-name*}}

> **Note**
>
> `tokenInfo`Bidang ini adalah satu-satunya bidang plugin dokumen SSM yang mendukung SecureString parameter. SecureString parameter tidak didukung untuk bidang lain, atau untuk plugin dokumen SSM lainnya.

```
{
    "owner":"TestUser",
    "repository":"GitHubTest",
    "path":"scripts/python/test-script",
    "getOptions":"branch:master",
    "tokenInfo":"{{ssm-secure:secure-string-token}}"
}
```

Untuk SourceType, Anda **`Git`** harus menentukan yang berikut ini:

- repositori

  URL repositori Git ke file atau direktori yang ingin Anda unduh.

  Tipe: String

Selain itu, Anda dapat menentukan parameter opsional berikut:

- GetOptions

  Pilihan tambahan untuk mengambil konten dari cabang selain cabang utaman atau dari komit tertentu dalam repositori. GetOptions dapat dihilangkan jika Anda menggunakan komit terbaru di cabang utama.

  Tipe: String

  Parameter ini menggunakan format berikut:

  - *cabang: refs/kepala/branch_name*

    Default-nya adalah `master`.

    `"branch"` diperlukan hanya jika dokumen SSM Anda disimpan di cabang selain `master`. Sebagai contoh:

    ```
    "getOptions": "branch:refs/head/main"
    ```

  - commitID: *commitID*

    Default-nya adalah `head`.

    Untuk menggunakan versi dokumen SSM Anda di komit selain yang terbaru, tentukan ID komit penuh. Sebagai contoh:

    ```
    "getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",
    ```

- privateSSHKey

  Kunci SSH untuk digunakan saat menghubungkan ke `repository` yang Anda tentukan. Anda dapat menggunakan format berikut untuk referensi parameter `SecureString` pada nilai kunci SSH Anda: `{{ssm-secure:`*your-secure-string-parameter*`}}`.

  Jenis: String

- skipHostKeyMemeriksa

  Menentukan nilai StrictHostKeyChecking opsi saat menghubungkan ke yang `repository` Anda tentukan. Nilai default-nya adalah `false`.

Jenis: Boolean

- nama pengguna

  Nama pengguna yang digunakan saat menghubungkan ke `repository` yang Anda tentukan menggunakan HTTP. Anda dapat menggunakan format berikut untuk referensi parameter `SecureString` untuk nilai nama pengguna Anda: `{{ssm-secure:`*`your-secure-string-parameter`*`}}`.

  Tipe: String

- Kata sandi

  Kata sandi digunakan saat menghubungkan ke `repository` yang Anda tentukan menggunakan HTTP. Anda dapat menggunakan format berikut untuk referensi parameter `SecureString` untuk nilai kata sandi Anda: `{{ssm-secure:`*`your-secure-string-parameter`*`}}`.

  Tipe: String

Untuk SourceType, Anda **`HTTP`** harus menentukan yang berikut ini:

- url

  URL ke file atau direktori yang ingin Anda unduh.

  Tipe: String

Selain itu, Anda dapat menentukan parameter opsional berikut:

- allowInsecureDownload

  Tentukan apakah unduhan dapat dilakukan melalui sambungan yang tidak dienkripsi dengan Secure Socket Layer (SSL) atau Transport Layer (TLS). Nilai default-nya adalah `false`. Kami tidak menyarankan untuk melakukan unduhan tanpa enkripsi. Jika Anda memilih untuk melakukannya, Anda menanggung semua risiko yang berkaitan. Keamanan adalah tanggung jawab bersama antara Anda AWS dan Anda. Hal ini digambarkan sebagai model tanggung jawab bersama. Untuk mempelajari informasi lebih lanjut, lihat Model tanggung jawab bersama.

  Jenis: Boolean

- authMethod

Tentukan apakah nama pengguna dan kata sandi yang digunakan untuk otentikasi saat menghubungkan ke `url` yang Anda tentukan. Jika Anda menentukan `Basic` atau `Digest`, Anda harus memberikan nilai untuk parameter `username` dan `password`. Untuk menggunakan `Digest` metode ini, SSM Agent versi 3.0.1181.0 atau yang lebih baru harus diinstal pada instance Anda. Metode `Digest` mendukung enkripsi MD5 dan SHA256.

Tipe: String

Nilai valid: `None` | `Basic` | `Digest`

- nama pengguna

  Nama pengguna yang digunakan saat menghubungkan ke `url` yang anda tentukan menggunakan Autentikasi `Basic`. Anda dapat menggunakan format berikut untuk referensi parameter `SecureString` untuk nilai nama pengguna Anda: {{ssm-secure:*your-secure-string-parameter*}}.

  Tipe: String

- Kata sandi

  Kata sandi untuk digunakan saat menghubungkan ke `url` yang anda tentukan menggunakan Autentikasi `Basic`. Anda dapat menggunakan format berikut untuk referensi parameter `SecureString` untuk nilai kata sandi Anda: {{ssm-secure:*your-secure-string-parameter*}}.

  Tipe: String

Untuk SourceType, **`S3`** tentukan yang berikut ini:

- jalur: URL ke file atau direktori yang ingin Anda unduh dari Amazon S3.

```
{
    "path": "https://s3.amazonaws.com/doc-example-bucket/powershell/
helloPowershell.ps1"
}
```

Untuk SourceType, ***`SSMDocument`*** tentukan salah satu dari berikut ini:

- Nama: Nama dan versi dokumen dalam format berikut: `name:version`. Versi adalah opsional.

```
{
```

```
    "name": "Example-RunPowerShellScript:3"
}
```

- Nama: ARN untuk dokumen dalam format berikut: `arn:aws:ssm:region:account_id:document/document_name`

```
{
   "name":"arn:aws:ssm:us-east-2:3344556677:document/MySharedDoc"
}
```

destinationPath

Jalur lokal opsional instans ada ditempat Anda ingin mengunduh file. Jika Anda tidak menentukan jalurnya, konten diunduh ke jalur relatif pada ID perintah Anda.

Tipe: String

Wajib: Tidak

aws:psModule

Instal PowerShell modul pada instans Amazon EC2. Plugin ini hanya berjalan pada Sistem operasi Windows Server.

Sintaks

Skema 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 }}"
      }
    }
  ]
}
```

Skema 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 }}"
            }
         ]
      }
   }
}
```

Properti

runCommand

PowerShell Perintah untuk menjalankan setelah modul diinstal.

Jenis: StringList

Diperlukan: Tidak

sumber

URL atau jalur lokal pada instans untuk aplikasi file `.zip`.

Tipe: String

Wajib: Ya

sourceHash

The SHA256 hash dari file `.zip`.

Tipe: String

Wajib: Tidak

timeoutSeconds

Waktu dalam detik untuk perintah yang harus diselesaikan sebelum dianggap telah gagal.

Tipe: String

Wajib: Tidak

workingDirectory

Jalur direktori kerja pada instans Anda.

Tipe: String

Wajib: Tidak

`aws:refreshAssociation`

(Skema versi 2.0 atau yang terbaru) Segarkan (berlaku paksa) sebuah asosiasi sesuai permintaan. Tindakan ini akan mengubah status sistem berdasarkan pada apa yang ditentukan dalam asosiasi yang dipilih atau semua asosiasi yang terikat pada target. Plugin ini berjalan pada Sistem operasi Linux dan Microsoft Windows Server.

Sintaks

Skema 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 }}"
        ]
      }
    }
  ]
}
```

Masukan

associationIds

Daftar ID asosiasi. Jika kosong, semua asosiasi yang berhubungan dengan target yang ditentukan telah diterapkan.

Jenis: StringList

Diperlukan: Tidak

`aws:runDockerAction`

(Skema versi 2.0 atau yang terbaru) Menjalankan tindakan Docker pada kontainer. Plugin ini berjalan pada Sistem operasi Linux dan Microsoft Windows Server.

Sintaks

Skema 2.2

YAML

```
---
mainSteps:
- action: aws:runDockerAction
  name: RunDockerAction
  inputs:
    action: "{{ action }}"
    container: "{{ container }}"
    image: "{{ image }}"
    memory: "{{ memory }}"
    cpuShares: "{{ cpuShares }}"
    volume: "{{ volume }}"
    cmd: "{{ cmd }}"
    env: "{{ env }}"
    user: "{{ user }}"
    publish: "{{ publish }}"
```

JSON

```
{
   "mainSteps":[
      {
         "action":"aws:runDockerAction",
         "name":"RunDockerAction",
         "inputs":{
            "action":"{{ action }}",
            "container":"{{ container }}",
            "image":"{{ image }}",
            "memory":"{{ memory }}",
            "cpuShares":"{{ cpuShares }}",
            "volume":"{{ volume }}",
            "cmd":"{{ cmd }}",
            "env":"{{ env }}",
            "user":"{{ user }}",
            "publish":"{{ publish }}"
         }
      }
   ]
```

```
}
```

Masukan

tindakan

Jenis tindakan yang harus dilakukan.

Tipe: String

Wajib: Ya

kontainer

ID kontainer Docker.

Tipe: String

Wajib: Tidak

image

Nama gambar Docker.

Tipe: String

Wajib: Tidak

cmd

Perintah kontainer.

Tipe: String

Wajib: Tidak

memori

Batas memori kontainer.

Tipe: String

Wajib: Tidak

cpuShares

Pembagian CPU kontainer (berat relatif).

Tipe: String

Wajib: Tidak

volume

Pemasangan volume kontainer.

Jenis: StringList

Diperlukan: Tidak

env

Variabel lingkungan kontainer.

Tipe: String

Wajib: Tidak

pengguna

Nama pengguna kontainer.

Tipe: String

Wajib: Tidak

menerbitkan

Port kontainer diterbitkan.

Tipe: String

Wajib: Tidak

`aws:runDocument`

(Skema versi 2.0 atau yang terbaru) Menjalankan dokumen SSM yang disimpan di Systems Manager atau pada berbagi lokal. Anda dapat menggunakan plugin ini dengan plugin `aws:downloadContent` untuk mengunduh dokumen SSM dari lokasi jarak jauh untuk berbagi lokal,

dan kemudian jalankannya. Plugin ini didukung oleh Linux dan Sistem operasi Windows Server. Plugin ini tidak mendukung menjalankan `AWS-UpdateSSMAgent` dokumen atau dokumen apa pun yang menggunakan `aws:updateSsmAgent` plugin.

Sintaks

Skema 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 }}"
      }
    }
  ]
}
```

Masukan

documentType

Jenis dokumen untuk dijalankan. Anda dapat menjalankan dokumen lokal (`LocalPath`) atau dokumen yang disimpan dalam Systems Manager (`SSMDocument`).

Tipe: String

Wajib: Ya

documentPath

Jalur ke dokumen. Jika `documentType` adalah `LocalPath`, kemudian tentukan jalur ke dokumen pada berbagi lokal. Jika `documentType` adalah `SSMDocument`, kemudian tentukan nama dokumen.

Tipe: String

Wajib: Tidak

documentParameters

Parameter untuk dokumen.

Jenis: StringMap

Diperlukan: Tidak

`aws:runPowerShellScript`

Jalankan PowerShell skrip atau tentukan jalur ke skrip yang akan dijalankan. Plugin ini berjalan pada Microsoft Windows Server dan sistem operasi Linux.

Sintaks

Skema 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 }}"
        ]
      }
    }
  ]
}
```

Skema 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 }}"
            }
         ]
      }
   }
}
```

Properti

runCommand

Tentukan perintah untuk menjalankan atau jalur skrip yang ada pada instans.

Jenis: StringList

Diperlukan: Ya

timeoutSeconds

Waktu dalam detik untuk perintah yang harus diselesaikan sebelum dianggap telah gagal. Ketika batas waktu tercapai, Systems Manager berhenti mengeksekusi perintah.

Tipe: String

Wajib: Tidak

workingDirectory

Jalur direktori kerja pada instans Anda.

Tipe: String

Wajib: Tidak

`aws:runShellScript`

Jalankan skrip shell Linux atau tentukan jalur untuk menjalankan skrip. Plugin ini hanya berjalan pada sistem operasi Linux.

Sintaks

Skema 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 }}"
        ]
      }
    }
  ]
}
```

Skema 1.2

YAML

```
---
```

```
runtimeConfig:
  aws:runShellScript:
    properties:
    - runCommand: "{{ commands }}"
      workingDirectory: "{{ workingDirectory }}"
      timeoutSeconds: "{{ executionTimeout }}"
```

JSON

```
{
   "runtimeConfig":{
      "aws:runShellScript":{
         "properties":[
            {
               "runCommand":"{{ commands }}",
               "workingDirectory":"{{ workingDirectory }}",
               "timeoutSeconds":"{{ executionTimeout }}"
            }
         ]
      }
   }
}
```

Properti

runCommand

Tentukan perintah untuk menjalankan atau jalur skrip yang ada pada instans.

Jenis: StringList

Diperlukan: Ya

timeoutSeconds

Waktu dalam detik untuk perintah yang harus diselesaikan sebelum dianggap telah gagal. Ketika batas waktu tercapai, Systems Manager berhenti mengeksekusi perintah.

Tipe: String

Wajib: Tidak

workingDirectory

Jalur direktori kerja pada instans Anda.

Tipe: String

Wajib: Tidak

`aws:softwareInventory`

(Skema versi 2.0 atau yang terbaru) Kumpulkan metadata tentang aplikasi, file, dan konfigurasi pada instans terkelola Anda. Plugin ini berjalan pada sistem operasi Linux dan Microsoft Windows Server. Saat Anda mengonfigurasi koleksi inventaris, Anda mulai dengan membuat AWS Systems Manager State Manager asosiasi. Systems Manager mengumpulkan data inventaris saat asosiasi dijalankan. Jika Anda tidak membuat asosiasi terlebih dahulu, dan mencoba untuk meminta plugin `aws:softwareInventory` sistem akan menampilkan kesalahan berikut:

```
The aws:softwareInventory plugin can only be invoked via ssm-associate.
```

Instans hanya dapat mengonfigurasi satu asosiasi inventaris dalam satu waktu. Jika Anda mengonfigurasi instans dengan dua atau beberapa asosiasi, inventaris tidak berjalan dan tidak ada data inventaris yang dikumpulkan. Untuk informasi lebih lanjut tentang pengumpulan inventaris, lihat AWS Systems ManagerInventaris.

Sintaks

Skema 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 }}"
         }
      }
   ]
}
```

Masukan

aplikasi

(Opsional) Kumpulkan metadata untuk aplikasi yang diinstal.

Tipe: String

Wajib: Tidak

awsComponents

(Opsional) Kumpulkan metadata untuk AWS komponen seperti. amazon-ssm-agent

Tipe: String

Wajib: Tidak

file

(Opsional, memerlukan SSM Agent versi 2.2.64.0 atau yang lebih baru) Kumpulkan metadata untuk file, termasuk nama file, file waktu dibuat, file waktu terakhir diubah dan diakses, dan ukuran file, untuk beberapa nama. Untuk informasi lebih lanjut tentang pengumpulan file inventaris, lihat Menggunakan file dan inventaris registri Windows.

Tipe: String

Wajib: Tidak

NetworkConfig

(Opsional) Kumpulkan metadata untuk konfigurasi jaringan.

Tipe: String

Wajib: Tidak

windowsUpdates

(Opsional) Kumpulkan metadata untuk semua pembaruan Windows.

Tipe: String

Wajib: Tidak

instanceDetailedInformation

(Opsional) Kumpulkan lebih banyak informasi instans yang disediakan oleh plugin inventaris default (`aws:instanceInformation`), termasuk model CPU, kecepatan, dan jumlah inti, untuk beberapa nama.

Tipe: String

Wajib: Tidak

layanan

(Opsional, hanya OS Windows, memerlukan SSM Agent versi 2.2.64.0 atau yang lebih baru) Kumpulkan metadata untuk konfigurasi layanan.

Tipe: String

Wajib: Tidak

windowsRegistry

(Opsional, hanya OS Windows, memerlukan SSM Agent versi 2.2.64.0 atau yang lebih baru) Kumpulkan kunci dan nilai Windows Registry. Anda dapat memilih jalur kunci dan mengumpulkan semua kunci dan nilai secara berulang. Anda juga dapat mengumpulkan kunci registri tertentu dan nilainya untuk jalur tertentu. Inventaris mengumpulkan jalur, nama, jenis, dan nilai kunci. Untuk informasi lebih lanjut tentang pengumpulan inventaris Registri Windows, lihat Menggunakan file dan inventaris registri Windows.

Tipe: String

Wajib: Tidak

windowsRoles

(Opsional, hanya OS Windows, memerlukan SSM Agent versi 2.2.64.0 atau yang lebih baru) Kumpulkan metadata untuk konfigurasi peran Microsoft Windows.

Tipe: String

Wajib: Tidak

customInventory

(Opsional) Kumpulkan data inventaris kustom. Untuk informasi lebih lanjut tentang inventaris kustom, lihat Menggunakan inventaris kustom

Tipe: String

Wajib: Tidak

`aws:updateAgent`

Memperbarui layanan EC2Config ke versi terbaru atau tetapkan versi lama. Plugin ini hanya berjalan di Sistem operasi Microsoft Windows Server. Untuk informasi selengkapnya tentang layanan EC2config, lihat Mengonfigurasi Instans Windows menggunakan layanan EC2config.

Sintaks

Skema 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"
      }
    }
  ]
}
```

Skema 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 }}"
         }
      }
   }
}
```

Properti

agentName

EC2Config. Ini adalah nama agen yang menjalankan layanan EC2Config.

Tipe: String

Wajib: Ya

allowDowngrade

Memungkinkan layanan EC2Config untuk diturunkan ke versi sebelumnya. Jika diatur ke false, hanya layanan (default) yang dapat ditingkatkan ke versi yang terbaru. Jika diatur ke true, tentukan versi sebelumnya.

Tipe: Boolean

Wajib: Tidak

sumber

Lokasi di mana Systems Manager menyalin versi EC2Config untuk menginstal. Anda tidak dapat mengubah lokasi ini.

Tipe: String

Wajib: Ya

targetVersion

Versi spesifik dari layanan EC2Config untuk menginstal. Jika tidak ditentukan, layanan akan diperbarui ke versi terbaru.

Tipe: String

Wajib: Tidak

`aws:updateSsmAgent`

Perbarui SSM Agent ke versi terbaru atau tentukan versi yang lebih lama. Plugin ini berjalan di Linux dan sistem operasi Windows Server. Untuk informasi selengkapnya, lihat Bekerja dengan SSM Agent.

Sintaks

Skema 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 }}"
      }
    }
  ]
}
```

Skema 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 }}"
            }
         ]
      }
   }
}
```

Properti

agentName

amazon-ssm-agent. Ini adalah nama agen Systems Manager yang memproses permintaan dan menjalankan perintah pada instans.

Tipe: String

Wajib: Ya

allowDowngrade

Izinkan SSM Agent untuk diturunkan ke versi sebelumnya. Jika diatur ke false, hanya agen (default) yang dapat ditingkatkan ke versi yang terbaru. Jika diatur ke true, tentukan versi sebelumnya.

Tipe: Boolean

Wajib: Ya

sumber

Lokasi di mana Systems Manager menyalin SSM Agent versi yang akan diinstal. Anda tidak dapat mengubah lokasi ini.

Tipe: String

Wajib: Ya

targetVersion

Versi khusus SSM Agent untuk menginstal. Jika tidak ditentukan, agen akan diperbarui ke versi terbaru.

Tipe: String

Wajib: Tidak

Membuat konten dokumen SSM

Jika dokumen AWS Systems Manager publik tidak melakukan semua tindakan yang ingin Anda lakukan pada AWS sumber daya Anda, Anda dapat membuat dokumen SSM Anda sendiri. Anda juga dapat melakukan klon dokumen SSM menggunakan konsol. Pengandaan dokumen dengan menyalin konten dari dokumen yang ada ke dokumen baru yang dapat Anda modifikasi. Saat membuat atau mengkloning dokumen, konten dokumen tidak boleh melebihi 64KB. Kuota ini juga mencakup konten yang ditentukan untuk parameter input saat runtime. Saat Anda membuat dokumen `Command` atau `Policy`, sebaiknya gunakan skema versi 2.2 atau yang terbaru sehingga Anda dapat memanfaatkan fitur terbaru, seperti pengeditan dokumen, versioning otomatis, pengurutan, dan banyak lagi.

Menulis konten dokumen SSM

Untuk membuat konten dokumen SSM Anda sendiri, penting untuk memahami skema yang berbeda, fitur, plugin, dan sintaks yang tersedia untuk dokumen SSM. Kami mensarankan untuk membiasakan diri dengan sumber daya berikut.

- Menulis AWS Systems Manager dokumen Anda sendiri
- Elemen dan parameter data
- Skema, fitur, dan contoh
- Referensi plugin dokumen perintah
- Referensi tindakan Otomatisasi Systems Manager

- Variabel sistem Otomatisasi
- Contoh runbook tambahan
- Bekerja dengan runbook Otomatisasi Systems Manager Menggunakan AWS Toolkit for Visual Studio Code
- Menggunakan Document Builder untuk membuat runbook
- Menggunakan skrip di runbook

AWS Dokumen SSM yang telah ditentukan sebelumnya mungkin melakukan beberapa tindakan yang Anda butuhkan. Anda dapat memanggil dokumen-dokumen ini dengan menggunakan plugin `aws:runDocument`, `aws:runCommand`, atau `aws:executeAutomation` dalam dokumen SSM kustom Anda, tergantung pada jenis dokumen. Anda juga dapat menyalin sebagian dari dokumen tersebut ke dalam dokumen SSM kustom, dan mengedit konten tersebut untuk memenuhi kebutuhan Anda.

Tip

Saat membuat konten dokumen SSM, Anda dapat mengubah konten dan memperbarui dokumen SSM Anda beberapa kali saat pengujian. Perintah berikut dapat memperbarui dokumen SSM dengan konten terbaru Anda, dan memperbarui versi default dokumen ke versi terbaru dari dokumen.

Note

Perintah Linux dan Windows menggunakan alat baris perintah `jq` untuk menyaring data respons 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
```

Mengkloning dokumen SSM

Anda dapat mengkloning AWS Systems Manager dokumen menggunakan konsol Systems Manager Documents untuk membuat dokumen SSM. Mengkloning dokumen SSM menyalin konten dari dokumen yang ada ke dokumen baru yang dapat Anda modifikasi. Anda tidak dapat mengkloning dokumen yang lebih besar dari 64KB.

Untuk mengkloning dokumen SSM

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.

2. Di panel navigasi, pilih Dokumen.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.
3. Di kotak pencarian, masukkan nama dokumen yang ingin Anda kloning.
4. Pilih nama dokumen yang ingin Anda kloning, lalu pilih Dokumen pengkloning dalam Actions dropdown.
5. Memodifikasi dokumen yang Anda inginkan, dan kemudian pilih Buat dokumen untuk menyimpan dokumen.

Setelah menulis konten dokumen SSM Anda, Anda dapat menggunakan konten Anda untuk membuat dokumen SSM menggunakan salah satu metode berikut.

Membuat dokumen SSM

- Membuat dokumen gabungan

Membuat dokumen gabungan

Dokumen komposit AWS Systems Manager (SSM) adalah dokumen kustom yang melakukan serangkaian tindakan dengan menjalankan satu atau lebih dokumen SSM sekunder. Dokumen komposit mempromosikan Infrastruktur sebagai code dengan memungkinkan Anda untuk membuat satu set standar dokumen SSM untuk tugas-tugas umum seperti boot-strapping perangkat lunak atau penggabungan domain instans. Anda kemudian dapat membagikan dokumen-dokumen ini Akun AWS secara bersamaan Wilayah AWS untuk mengurangi pemeliharaan dokumen SSM dan memastikan konsistensi.

Sebagai contoh, Anda dapat membuat dokumen komposit untuk melakukan tindakan berikut:

1. Menginstal semua patch dalam daftar yang izinkan.
2. Menginstal perangkat lunak antivirus.
3. Mengunduh skrip dari GitHub dan menjalankannya.

Dalam contoh ini, dokumen SSM kustom Anda mencakup plugin berikut untuk melakukan tindakan ini:

1. `aws:runDocument`Plugin untuk menjalankan `AWS-RunPatchBaseline` dokumen, yang menginstal semua memungkinkan patch terdaftar.
2. Plugin `aws:runDocument` untuk menjalankan dokumen `AWS-InstallApplication`, yang menginstal perangkat lunak antivirus.
3. `aws:downloadContent`Plugin untuk mengunduh skrip dari GitHub dan menjalankannya.

Dokumen komposit dan sekunder dapat disimpan di Systems Manager, GitHub (repositori publik dan pribadi), atau Amazon S3. Dokumen komposit dan dokumen sekunder dapat dibuat di JSON atau YAML.

> **Note**
>
> Dokumen komposit hanya dapat menjalankan hingga kedalaman maksimum tiga dokumen. Ini berarti bahwa dokumen komposit dapat memanggil dokumen turunan; dan dokumen turunan dapat memanggil satu dokumen terakhir.

Untuk membuat dokumen komposit, tambahkan plugin `aws:runDocument` dalam dokumen SSM kustom dan tentukan input yang diperlukan. Berikut ini adalah contoh dokumen komposit yang melakukan tindakan berikut:

1. Menjalankan `aws:downloadContent` plugin untuk mengunduh dokumen SSM dari repositori GitHub publik ke direktori lokal yang disebut bootstrap. Dokumen SSM disebut StateManagerBootstrap .yml/dokumen YAMM.
2. Menjalankan `aws:runDocument` plugin untuk menjalankan StateManagerBootstrap dokumen.yl. Tidak ada parameter yang ditentukan.
3. Menjalankan plugin `aws:runDocument` untuk menjalankan dokumen SSM `AWS-ConfigureDocker pre-defined`. Parameter yang ditentukan untuk menginstal Docker pada instans.

```
{
  "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\"}"
      }
    }
  ]
}
```

Info lebih lanjut

- Untuk informasi tentang me-reboot server dan instance saat menggunakan Run Command untuk memanggil skrip, lihat. Menangani reboot saat menjalankan perintah
- Untuk informasi selengkapnya tentang plugin yang dapat Anda tambahkan ke dokumen SSM kustom, lihat Referensi plugin dokumen perintah.

- Jika Anda hanya ingin menjalankan dokumen dari lokasi jarak jauh (tanpa membuat dokumen komposit), lihat Menjalankan dokumen dari lokasi terpencil.

Bekerja dengan dokumen

Bagian ini mencakup informasi tentang cara menggunakan dan bekerja dengan dokumen SSM.

Daftar Isi

Menggunakan dokumen SSM dalam Asosiasi State Manager

Jika Anda membuat dokumen SSM untukState Manager, kemampuanAWS Systems Manager, Anda harus mengaitkan dokumen dengan instans terkelola setelah Anda menambahkan dokumen ke sistem. Untuk informasi selengkapnya, lihat Bekerja dengan asosiasi di Systems Manager.

Ingatlah detail berikut saat menggunakan dokumen SSM dalam State Manager asosiasi.

- Anda dapat menetapkan beberapa dokumen ke target dengan membuat State Manager asosiasi berbeda yang menggunakan dokumen berbeda.
- Jika Anda membuat dokumen dengan plugin yang bertentangan (misalnya, domain bergabung dan menghapus dari domain), plugin terakhir yang dijalankan akan menjadi status akhir. State Managertidak memvalidasi urutan logis atau rasionalitas perintah atau plugin dalam dokumen Anda.
- Ketika memproses dokumen, asosiasi instans diterapkan pertama kali, dan asosiasi grup yang ditandai diterapkan berikutnya. Jika instans adalah bagian dari beberapa kelompok yang ditandai,

maka dokumen yang merupakan bagian dari kelompok yang ditandai tidak akan dijalankan dalam urutan tertentu. Jika instans langsung ditargetkan melalui beberapa dokumen dengan ID instans nya, tidak ada urutan tertentu pada eksekusi.

- Jika Anda mengubah versi default dokumen Kebijakan SSM untukState Manager, asosiasi apa pun yang menggunakan dokumen akan mulai menggunakan versi default baru saat Systems Manager menerapkan asosiasi ke instance berikutnya.
- Jika Anda membuat asosiasi menggunakan dokumen SSM yang dibagikan dengan Anda, dan kemudian pemilik berhenti membagikan dokumen dengan Anda, asosiasi Anda tidak lagi memiliki akses ke dokumen tersebut. Namun, jika pemilik membagikan dokumen SSM yang sama dengan Anda lagi nanti, asosiasi Anda secara otomatis memetakan ulang dokumen tersebut.

Membandingkan versi dokumen SSM

Anda dapat membandingkan perbedaan dalam konten antar beberapa versi dokumen (SSM) AWS Systems Manager di konsol dokumen Systems Manager. Ketika membandingkan versi dokumen SSM, perbedaan antara konten dari versi yang disorot.

Untuk membandingkan konten dokumen SSM (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

   Jika halaman beranda AWS Systems Manager terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.
3. Dalam daftar dokumen, pilih dokumen yang kontennya ingin Anda bandingkan.
4. Pada tab Konten, pilih Bandingkan versi, dan pilih versi dokumen yang ingin Anda bandingkan kontennya.

Membuat dokumen SSM (konsol)

Setelah Anda membuat konten untuk dokumen SSM kustom Anda, seperti yang dijelaskan dalam Menulis konten dokumen SSM, Anda dapat menggunakan konsol Systems Manager untuk membuat dokumen SSM menggunakan konten Anda.

Untuk membuat dokumen SSM (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

   Jika halaman beranda AWS Systems Manager terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.
3. Pilih Buat perintah atau sesi.
4. Masukkan nama deskriptif untuk dokumen
5. (Opsional) Untuk Jenis target, tentukan jenis sumber daya dokumen yang dapat dijalankan.
6. Di daftar Jenis dokumen, pilih jenis dokumen yang ingin Anda buat.
7. Hapus tanda kurung di bidang Konten, dan kemudian paste dokumen konten yang Anda buat sebelumnya.
8. (Opsional) Dalam bagian Tag dokumen, terapkan satu pasangan nilai kunci tag atau lebih ke dokumen.

   Tag adalah metadata opsional yang Anda tetapkan ke sumber daya. Tag memungkinkan Anda untuk mengkategorikan sumber daya dengan berbagai cara, seperti berdasarkan tujuan, pemilik, atau lingkungan. Misalnya, Anda mungkin ingin menandai dokumen untuk mengidentifikasi jenis tugas yang dijalankannya, jenis sistem operasi target, dan lingkungan tempat ia berjalan. Dalam kasus ini, Anda bisa menentukan pasangan nama/nilai kunci berikut:

   - `Key=TaskType,Value=MyConfigurationUpdate`
   - `Key=OS,Value=AMAZON_LINUX_2`
   - `Key=Environment,Value=Production`

   Untuk informasi selengkapnya tentang penandaan sumber daya Systems Manager, lihat Penandaan sumber daya Systems Manager.
9. Pilih Buat dokumen untuk menyimpan dokumen.

Membuat dokumen SSM (baris perintah)

Setelah Anda membuat konten untuk kustom dokumen (SSM) AWS Systems Manager, seperti yang diterangkan dalam Menulis konten dokumen SSM, Anda dapat menggunakan AWS Command Line Interface (AWS CLI) atau AWS Tools for PowerShell untuk membuat dokumen SSM menggunakan konten Anda. Hal ini ditunjukkan dalam perintah berikut.

Sebelum Anda memulai

Instal dan konfigurasikan AWS CLI atau AWS Tools for PowerShell, jika Anda belum melakukannya. Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI dan Menginstal AWS Tools for PowerShell.

Jalankan perintah berikut. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

Linux & macOS

```
aws ssm create-document \
--content file://path/to/file/documentContent.json \
--name "document-name" \
--document-type "Command" \
--tags "Key=tag-key,Value=tag-value"
```

Windows

```
aws ssm create-document ^
--content file://C:\path\to\file\documentContent.json ^
--name "document-name" ^
--document-type "Command" ^
--tags "Key=tag-key,Value=tag-value"
```

PowerShell

```
$json = Get-Content -Path "C:\path\to\file\documentContent.json" | Out-String
New-SSMDocument `
-Content $json `
-Name "document-name" `
-DocumentType "Command" `
-Tags "Key=tag-key,Value=tag-value"
```

Jika berhasil, sistem menampilkan respon seperti berikut ini.

```
{
"DocumentDescription":{
  "CreatedDate":1.585061751738E9,
  "DefaultVersion":"1",
  "Description":"MyCustomDocument",
  "DocumentFormat":"JSON",
  "DocumentType":"Command",
  "DocumentVersion":"1",
  "Hash":"0d3d879b3ca072e03c12638d0255ebd004d2c65bd318f8354fcde820dEXAMPLE",
  "HashType":"Sha256",
  "LatestVersion":"1",
  "Name":"Example",
  "Owner":"111122223333",
  "Parameters":[
     --truncated--
  ],
  "PlatformTypes":[
     "Windows",
     "Linux"
  ],
  "SchemaVersion":"0.3",
  "Status":"Creating",
  "Tags": [
        {
            "Key": "Purpose",
            "Value": "Test"
        }
    ]
}
}
```

Membuat dokumen SSM (API)

Setelah membuat konten untuk dokumen kustom AWS Systems Manager (SSM), seperti yang dijelaskan dalamMenulis konten dokumen SSM, Anda dapat menggunakan SDK pilihan Anda untuk memanggil operasi AWS Systems Manager CreateDocumentAPI guna membuat dokumen SSM menggunakan konten Anda. String JSON atau YAML untuk permintaan parameter `Content` umumnya dibaca dari sebuah file. Contoh berikut adalah fungsi membuat dokumen SSM menggunakan SDK untuk Python, Go, dan Java.

Python

```
import boto3

ssm = boto3.client('ssm')
filepath = '/path/to/file/documentContent.yaml'

def createDocumentApiExample():
with open(filepath) as openFile:
    documentContent = openFile.read()
    createDocRequest = ssm.create_document(
        Content = documentContent,
        Name = 'createDocumentApiExample',
        DocumentType = 'Automation',
        DocumentFormat = 'YAML'
    )
    print(createDocRequest)

createDocumentApiExample()
```

Go

```
package main

import (
"github.com/aws/aws-sdk-go/aws"
"github.com/aws/aws-sdk-go/aws/session"
"github.com/aws/aws-sdk-go/service/ssm"

"fmt"
"io/ioutil"
"log"
)


func main() {
openFile, err := ioutil.ReadFile("/path/to/file/documentContent.yaml")
if err != nil {
    log.Fatal(err)
}
documentContent := string(openFile)
```

```
sesh := session.Must(session.NewSessionWithOptions(session.Options{
    SharedConfigState: session.SharedConfigEnable}))

ssmClient := ssm.New(sesh)
createDocRequest, err := ssmClient.CreateDocument(&ssm.CreateDocumentInput{
    Content: &documentContent,
    Name:  aws.String("createDocumentApiExample"),
    DocumentType: aws.String("Automation"),
    DocumentFormat: aws.String("YAML"),
})
result := *createDocRequest
fmt.Println(result)
}
```

Java

```


import java.io.IOException;
import java.nio.charset.Charset;
import java.nio.charset.StandardCharsets;
import java.nio.file.Files;
import java.nio.file.Paths;

import com.amazonaws.AmazonClientException;
import com.amazonaws.AmazonServiceException;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.simplesystemsmanagement.AWSSimpleSystemsManagement;
import
 com.amazonaws.services.simplesystemsmanagement.AWSSimpleSystemsManagementClientBuilder;
import com.amazonaws.services.simplesystemsmanagement.model.*;

public class createDocumentApiExample {
public static void main(String[] args) {
try {
  createDocumentMethod(getDocumentContent());
}
catch (IOException e) {
  e.printStackTrace();
}
}
public static String getDocumentContent() throws IOException {
```

```
  String filepath = new String("/path/to/file/documentContent.yaml");
    byte[] encoded = Files.readAllBytes(Paths.get(filepath));
    String documentContent = new String(encoded, StandardCharsets.UTF_8);
    return documentContent;
}

public static void createDocumentMethod (final String documentContent) {
  AWSSimpleSystemsManagement ssm =
 AWSSimpleSystemsManagementClientBuilder.defaultClient();
  final CreateDocumentRequest createDocRequest = new CreateDocumentRequest()
    .withContent(documentContent)
    .withName("createDocumentApiExample")
    .withDocumentType("Automation")
    .withDocumentFormat("YAML");
  final CreateDocumentResult result = ssm.createDocument(createDocRequest);
}
}
```

Untuk informasi selengkapnya tentang membuat konten dokumen kustom, lihat Elemen dan parameter data.

Menghapus dokumen SSM kustom

Jika Anda tidak lagi ingin menggunakan dokumen SSM khusus, Anda dapat menghapusnya dengan menggunakan AWS Command Line Interface (AWS CLI) atau AWS Systems Manager konsol.

Untuk menghapus dokumen SSM () AWS CLI

1. Sebelum Anda menghapus dokumen, kami sarankan Anda memisahkan semua instance yang terkait dengan dokumen.

   Jalankan perintah berikut untuk memisahkan instance dari dokumen.

   ```
   aws ssm delete-association --instance-id "123456789012" --name "documentName"
   ```

   Tidak ada output jika perintah berhasil.

2. Jalankan perintah berikut. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

Linux

```
aws ssm delete-document \
    --name "document-name" \
    --document-version "document-version" \
    --version-name "version-name"
```

Windows

```
aws ssm delete-document ^
    --name "document-name" ^
    --document-version "document-version" ^
    --version-name "version-name"
```

PowerShell

```
Delete-SSMDocument `
    -Name "document-name" `
    -DocumentVersion 'document-version' `
    -VersionName 'version-name'
```

Tidak ada output jika perintah berhasil.

> **Important**
>
> Jika `document-version` atau tidak `version-name` disediakan, semua versi dokumen dihapus.

Untuk menghapus dokumen SSM (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.

3. Pilih dokumen yang ingin Anda hapus.
4. Pilih Hapus. Saat diminta untuk menghapus dokumen, pilih Hapus.

Menjalankan dokumen dari lokasi terpencil

Anda dapat menjalankan dokumen AWS Systems Manager (SSM) dari lokasi terpencil dengan menggunakan dokumen SSM yang `AWS-RunDocument` telah ditentukan sebelumnya. Dokumen ini mendukung menjalankan dokumen SSM yang disimpan di lokasi berikut:

- GitHubRepositori publik dan pribadi (tidak GitHub Enterprise didukung)
- Buket Amazon S3
- Systems Manager

Meskipun Anda juga dapat menjalankan dokumen jarak jauh dengan menggunakan State Manager atau Otomasi, kemampuan AWS Systems Manager, prosedur berikut hanya menjelaskan cara menjalankan dokumen SSM jarak jauh dengan menggunakan AWS Systems Manager Run Command di konsol Systems Manager.

> **Note**
>
> `AWS-RunDocument` hanya dapat digunakan untuk menjalankan jenis perintah dokumen SSM, bukan jenis lain seperti otomatisasi runbook. `AWS-RunDocument` Menggunakan `aws:downloadContent` plugin. Untuk informasi lebih lanjut tentang plugin `aws:downloadContent` , lihat `aws:downloadContent`.

Sebelum Anda memulai

Sebelum Anda menjalankan dokumen jarak jauh, Anda harus menyelesaikan tugas berikut.

- Membuat perintah dokumen SSM dan menyimpannya di lokasi jarak jauh. Untuk informasi selengkapnya, lihat Membuat konten dokumen SSM

- Jika Anda berencana untuk menjalankan dokumen jarak jauh yang disimpan dalam GitHub repositori pribadi, maka Anda harus membuat `SecureString` parameter Systems Manager untuk token akses GitHub keamanan Anda. Anda tidak dapat mengakses dokumen jarak jauh di GitHub repositori pribadi dengan meneruskan token Anda secara manual melalui SSH. Token akses harus diteruskan sebagai parameter `SecureString` Systems Manager. Untuk informasi lebih lanjut tentang pembuatan parameter `SecureString`, lihat Menandai parameter Systems Manager.

Jalankan dokumen jarak jauh (konsol)

Untuk menjalankan dokumen jarak jauh

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Run Command.
3. Pilih Jalankan perintah.
4. Di daftar Dokumen, pilih **`AWS-RunDocument`**.
5. Masuk Parameter perintah, untuk Jenis sumber, pilih satu opsi.

   - Jika Anda memilih GitHub, tentukan informasi Info Sumber dalam format berikut:

   ```
   {
       "owner": "owner_name",
       "repository": "repository_name",
       "path": "path_to_document",
       "getOptions":"branch:branch_name",
       "tokenInfo": "{{ssm-secure:secure-string-token}}"
   }
   ```

   Sebagai contoh:

   ```
   {
       "owner":"TestUser",
       "repository":"GitHubTestExamples",
       "path":"scripts/python/test-script",
   ```

```
    "getOptions":"branch:exampleBranch",
    "tokenInfo":"{{ssm-secure:my-secure-string-token}}"
}
```

> **Note**
>
> `getOptions` adalah pilihan tambahan untuk mengambil konten dari selain cabang utama, atau dari komit tertentu dalam repositori. `getOptions` dapat dihilangkan jika Anda menggunakan perbaikan terbaru di cabang utama. Parameter `branch` diperlukan hanya jika dokumen SSM Anda disimpan di cabang selain `master`. Untuk menggunakan versi dokumen SSM Anda dalam Komitmen di repositori anda, gunakan `commitID` dengan `getOptions` daripada `branch`. Sebagai contoh:
>
> ```
> "getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",
> ```

- Jika Anda memilih S3, tentukan Info sumber dalam format berikut:

  ```
  {"path":"URL_to_document_in_S3"}
  ```

  Sebagai contoh:

  ```
  {"path":"https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/scripts/ruby/mySSMdoc.json"}
  ```

- Jika Anda memilih Dokumen SSM, tentukan Info sumber dalam format berikut:

  ```
  {"name": "document_name"}
  ```

  Sebagai contoh:

  ```
  {"name": "mySSMdoc"}
  ```

6. Di Parameter dokumen, masukkan parameter untuk dokumen SSM jarak jauh. Sebagai contoh, jika Anda menjalankan dokumen `AWS-RunPowerShell`, Anda dapat menentukan:

   ```
   {"commands": ["date", "echo \"Hello World\""]}
   ```

   Jika Anda menjalankan dokumen `AWS-ConfigureAWSPack`, Anda dapat menentukan:

```
{
   "action":"Install",
   "name":"AWSPVDriver"
}
```

7. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

> **Tip**
>
> Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

8. Untuk Parameter lainnya:

   - Untuk Komentar, ketik informasi tentang perintah ini.
   - Untuk Waktu habis (detik), tentukan jumlah detik untuk menunggu sistem sebelum gagal menjalankan perintah keseluruhan.

9. Untuk Pengendalian rate:

   - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

> **Note**
>
> Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

   - Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

10. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

    > **Note**
    >
    > Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan milik pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbeda Akun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

11. Di bagian Notifikasi SNS, jika Anda ingin notifikasi dikirim tentang status eksekusi perintah, pilih kotak centang Aktifkan notifikasi SNS.

    Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Run Command Amazon SNS, lihat. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

12. Pilih Jalankan.

> **Note**
>
> Untuk informasi tentang me-reboot server dan instance saat menggunakan Run Command untuk memanggil skrip, lihat. Menangani reboot saat menjalankan perintah

Membagikan dokumen SSM

Anda dapat berbagi dokumen AWS Systems Manager (SSM) secara pribadi atau publik dengan akun yang sama. Wilayah AWS Untuk membagikan dokumen secara pribadi, Anda mengubah izin dokumen dan mengizinkan individu tertentu untuk mengaksesnya sesuai dengan ID Akun AWS . Untuk membagikan dokumen SSM ke publik, Anda dapat mengubah izin dokumen dan menentukan `All`. Dokumen tidak dapat dibagikan secara publik dan pribadi secara bersamaan.

> **Warning**
>
> Gunakan dokumen SSM bersama hanya dari sumber tepercaya. Ketika menggunakan dokumen bersama, hati-hati meninjau konten dokumen sebelum menggunakannya sehingga Anda dapat memahami bagaimana hal itu akan mengubah konfigurasi instans Anda. Untuk informasi selengkapnya tentang praktik terbaik dokumen bersama, lihat Praktik terbaik untuk dokumen SSM bersama.

Batasan

Ketika Anda mulai bekerja dengan dokumen SSM, perhatikan batasan berikut.

- Hanya pemilik yang dapat berbagi dokumen.
- Anda harus berhenti berbagikan dokumen sebelum dapat menghapusnya. Untuk informasi selengkapnya, lihat Memodifikasi izin untuk dokumen SSM bersama.
- Anda dapat berbagi dokumen dengan maksimum 1000 Akun AWS. Anda dapat meminta peningkatan pada batas ini di Pusat AWS Support. Untuk Jenis batas, pilih EC2 Systems Manager dan jelaskan alasan Anda atas permintaan tersebut.
- Anda dapat berbagi secara publik maksimal lima dokumen SSM. Anda dapat meminta peningkatan pada batas ini di Pusat AWS Support. Untuk Jenis batas, pilih EC2 Systems Manager dan jelaskan alasan Anda atas permintaan tersebut.
- Dokumen dapat dibagikan dengan akun lain Wilayah AWS hanya dalam hal yang sama. Tidak mendukung berbagi Lintas Wilayah.

Untuk informasi selengkapnya tentang kuota layanan Systems Manager, lihat Service Quotas AWS Systems Manager.

Daftar Isi

Praktik terbaik untuk dokumen SSM bersama

Tinjau pedoman berikut sebelum Anda berbagi atau menggunakan dokumen bersama.

Hapus informasi sensitif

Tinjau dokumen AWS Systems Manager (SSM) Anda dengan hati-hati dan hapus informasi sensitif apa pun. Misalnya, verifikasi bahwa dokumen tidak menyertakan AWS kredensil Anda. Jika Anda berbagi dokumen dengan individu tertentu, pengguna tersebut dapat melihat informasi dalam dokumen. Jika Anda berbagi dokumen secara publik, siapa pun dapat melihat informasi dalam dokumen.

Memblokir berbagi dokumen untuk publik

Kecuali kasus penggunaan Anda memerlukan berbagi publik untuk diaktifkan, sebaiknya aktifkan pengaturan blokir berbagi publik untuk dokumen Systems Manager Anda di bagian Preferensi pada konsol dokumen Systems Manager.

Batasi Run Command tindakan menggunakan kebijakan kepercayaan IAM

Buat kebijakan restriktif AWS Identity and Access Management (IAM) untuk pengguna yang akan memiliki akses ke dokumen. Kebijakan IAM menentukan dokumen SSM mana yang dapat dilihat pengguna di konsol Amazon Elastic Compute Cloud (Amazon EC2) atau dengan menelepon menggunakan () atau. `ListDocuments` AWS Command Line Interface AWS CLI AWS Tools for Windows PowerShell Kebijakan ini juga membatasi tindakan yang dapat dilakukan pengguna dengan dokumen SSM. Anda dapat membuat kebijakan yang lebih ketat sehingga pengguna hanya dapat menggunakan dokumen tertentu. Untuk informasi selengkapnya, lihat Contoh kebijakan yang dikelola pelanggan.

Mengunakan dengan hati-hati saat menggunakan dokumen SSM bersama

Tinjau isi setiap dokumen yang dibagikan dengan Anda, terutama dokumen publik, untuk memahami perintah yang akan dijalankan di instans Anda. Dokumen dapat dengan sengaja atau tidak sengaja memiliki dampak negatif setelah dijalankan. Jika dokumen referensi jaringan eksternal, meninjau sumber eksternal sebelum Anda menggunakan dokumen.

Kirim perintah menggunakan hash dokumen

Ketika Anda berbagi dokumen, sistem menciptakan hash Sha-256 dan menetapkannya ke dokumen. Sistem ini juga menyimpan snapshot dari konten dokumen. Ketika Anda mengirim perintah menggunakan dokumen bersama, Anda dapat menentukan hash dalam perintah Anda untuk memastikan bahwa kondisi berikut ini benar:

- Anda menjalankan perintah dari dokumen Systems Manager yang benar
- Konten dokumen tidak berubah sejak dibagikan dengan Anda.

Jika hash tidak cocok dengan dokumen tertentu atau jika isi dari dokumen bersama telah berubah, perintah mengembalikan pengecualian `InvalidDocument`. Hash tidak dapat memverifikasi konten dokumen dari lokasi eksternal.

Memblokir berbagi dokumen SSM untuk publik

Kecuali kasus penggunaan Anda mengharuskan berbagi publik diaktifkan, sebaiknya aktifkan pengaturan blokir berbagi publik untuk dokumen AWS Systems Manager (SSM) Anda. Mengaktifkan pengaturan ini untuk mencegah akses yang tidak diinginkan ke dokumen SSM Anda. Pengaturan blok berbagi publik adalah pengaturan tingkat akun yang dapat berbeda untuk masing-masing Wilayah AWS. Selesaikan tugas berikut untuk memblokir berbagi dokumen SSM Anda untuk publik.

Memblokir berbagi publik (konsol)

Untuk memblokir berbagi dokumen SSM secara publik

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.
3. Pilih Preferensi, lalu pilih Edit di bagian Memblokir berbagi publik.
4. Pilih opsi kotak centang Memblokir berbagi publik, dan kemudian pilih Simpan.

Blokir berbagi publik (baris perintah)

Buka AWS Command Line Interface (AWS CLI) atau AWS Tools for Windows PowerShell di komputer lokal Anda dan jalankan perintah berikut untuk memblokir berbagi publik dokumen SSM Anda.

Linux & macOS

```
aws ssm update-service-setting  \
```

```
    --setting-id /ssm/documents/console/public-sharing-permission \
    --setting-value Disable \
    --region 'The Wilayah 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 Wilayah AWS you want to block public sharing in"
```

PowerShell

```
Update-SSMServiceSetting `
    -SettingId /ssm/documents/console/public-sharing-permission `
    -SettingValue Disable `
    -Region The Wilayah AWS you want to block public sharing in
```

Konfirmasikan nilai pengaturan yang diperbarui menggunakan perintah berikut.

Linux & macOS

```
aws ssm get-service-setting   \
    --setting-id /ssm/documents/console/public-sharing-permission \
    --region The Wilayah AWS you blocked public sharing in
```

Windows

```
aws ssm get-service-setting  ^
    --setting-id /ssm/documents/console/public-sharing-permission ^
    --region "The Wilayah AWS you blocked public sharing in"
```

PowerShell

```
Get-SSMServiceSetting `
    -SettingId /ssm/documents/console/public-sharing-permission `
    -Region The Wilayah AWS you blocked public sharing in
```

Membatasi akses untuk memblokir berbagi publik dengan IAM

Anda dapat membuat kebijakan AWS Identity and Access Management (IAM) yang membatasi pengguna untuk memodifikasi setelan blokir berbagi publik. Hal ini mencegah pengguna mengizinkan akses yang tidak diinginkan ke dokumen SSM Anda.

Berikut ini adalah contoh kebijakan IAM yang mencegah pengguna memperbarui pengaturan blokir berbagi publik. Untuk menggunakan contoh ini, Anda harus mengganti contoh ID akun Amazon Web Services dengan ID akun Anda sendiri.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "ssm:UpdateServiceSetting",
            "Resource": "arn:aws:ssm:*:987654321098:servicesetting/ssm/documents/
console/public-sharing-permission"
        }
    ]
}
```

Berbagi dokumen SSM

Anda dapat berbagi dokumen AWS Systems Manager (SSM) dengan menggunakan konsol Systems Manager. Saat berbagi dokumen dari konsol, hanya versi default dokumen yang dapat dibagikan. Anda juga dapat membagikan dokumen SSM secara terprogram dengan memanggil operasi `ModifyDocumentPermission` API menggunakan AWS Command Line Interface (AWS CLI) AWS Tools for Windows PowerShell, atau SDK. AWS Sebelum Anda berbagi dokumen, dapatkan ID Akun AWS orang yang ingin Anda bagikan. Anda akan menentukan ID akun ini ketika Anda berbagi dokumen.

Bagikan dokumen (konsol)

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.

3. Dalam daftar dokumen, pilih dokumen yang ingin Anda bagikan, lalu pilih Tampilkan detail. Pada tab Izin, verifikasi bahwa Anda adalah pemilik dokumen. Hanya pemilik dokumen yang dapat berbagi dokumen.
4. Pilih Edit.
5. Untuk berbagi perintah secara publik, pilih publik lalu pilih Simpan. Untuk berbagi perintah secara pribadi, pilih Pribadi, masukkan ID Akun AWS , pilih Tambah izin, lalu pilih Simpan.

Berbagi dokumen (baris perintah)

Prosedur berikut mengharuskan Anda menentukan Wilayah AWS untuk sesi baris perintah Anda.

1. Buka AWS CLI atau AWS Tools for Windows PowerShell di komputer lokal Anda dan jalankan perintah berikut untuk menentukan kredensil Anda.

   Dalam perintah berikut, ganti *wilayah* dengan informasi Anda sendiri. Untuk daftar nilai *wilayah* yang didukung, lihat kolom Region di titik akhir layanan Systems Manager di Referensi Umum 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. Gunakan perintah berikut untuk mencantumkan semua dokumen SSM yang tersedia untuk Anda. Daftar ini mencakup dokumen yang Anda buat dan dokumen yang dibagikan dengan Anda.

   Linux & macOS

   ```
   aws ssm list-documents
   ```

   Windows

   ```
   aws ssm list-documents
   ```

   PowerShell

   ```
   Get-SSMDocumentList
   ```

3. Gunakan perintah berikut untuk mendapatkan dokumen tertentu.

   Linux & macOS

   ```
   aws ssm get-document \
       --name document name
   ```

   Windows

   ```
   aws ssm get-document ^
       --name document name
   ```

   PowerShell

   ```
   Get-SSMDocument `
       -Name document name
   ```

4. Gunakan perintah berikut untuk mendapatkan deskripsi dokumen.

Linux & macOS

```
aws ssm describe-document \
    --name document name
```

Windows

```
aws ssm describe-document ^
    --name document name
```

PowerShell

```
Get-SSMDocumentDescription `
    -Name document name
```

5. Gunakan perintah berikut untuk menampilkan izin untuk dokumen.

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. Gunakan perintah berikut untuk mengubah izin untuk dokumen dan berbagi. Anda harus menjadi pemilik dokumen untuk mengedit izin. Secara opsional, Anda dapat menentukan versi dokumen yang ingin Anda bagikan menggunakan `--shared-document-version` parameter. Jika Anda tidak menentukan versi, sistem membagikan `Default` versi dokumen. Perintah contoh ini

secara pribadi membagikan dokumen dengan individu tertentu, berdasarkan Akun AWS ID orang tersebut.

Linux & macOS

```
aws ssm modify-document-permission \
    --name document name \
    --permission-type Share \
    --account-ids-to-add Akun AWS ID
```

Windows

```
aws ssm modify-document-permission ^
    --name document name ^
    --permission-type Share ^
    --account-ids-to-add Akun AWS ID
```

PowerShell

```
Edit-SSMDocumentPermission `
    -Name document name `
    -PermissionType Share `
    -AccountIdsToAdd Akun AWS ID
```

7. Gunakan perintah berikut untuk berbagi dokumen secara publik.

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')
```

Memodifikasi izin untuk dokumen SSM bersama

Jika Anda berbagi perintah, pengguna dapat melihat dan menggunakan perintah itu sampai Anda menghapus akses ke dokumen AWS Systems Manager (SSM) atau menghapus dokumen SSM. Namun, Anda tidak dapat menghapus dokumen selama dokumen dibagikan. Anda harus berhenti membagikan terlebih dahulu dan kemudian menghapusnya.

Berhenti membagikan dokumen (konsol)

Berhenti membagikan dokumen

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.
3. Dalam daftar dokumen, pilih dokumen yang ingin Anda hentikan berbagi, lalu pilih Detail. Di bagian Izin, verifikasi bahwa Anda adalah pemilik dokumen. Hanya pemilik dokumen yang dapat berhenti berbagi dokumen.
4. Pilih Edit.
5. Pilih X untuk menghapus Akun AWS ID yang seharusnya tidak lagi memiliki akses ke perintah, lalu pilih Simpan.

Berhenti berbagi dokumen (baris perintah)

Buka AWS CLI atau AWS Tools for Windows PowerShell di komputer lokal Anda dan jalankan perintah berikut untuk berhenti berbagi perintah.

Linux & macOS

```
aws ssm modify-document-permission \
    --name document name \
    --permission-type Share \
    --account-ids-to-remove 'Akun AWS ID'
```

Windows

```
aws ssm modify-document-permission ^
    --name document name ^
    --permission-type Share ^
    --account-ids-to-remove "Akun AWS ID"
```

PowerShell

```
Edit-SSMDocumentPermission `
    -Name document name `
    -PermissionType Share `
    –AccountIdsToRemove Akun AWS ID
```

Menggunakan dokumen SSM bersama

Saat Anda membagikan dokumen AWS Systems Manager (SSM), sistem menghasilkan Nama Sumber Daya Amazon (ARN) dan menetapkannya ke perintah. Jika Anda memilih dan menjalankan dokumen bersama dari konsol Systems Manager, Anda tidak akan melihat ARN. Namun, jika Anda ingin menjalankan dokumen SSM bersama menggunakan metode selain konsol Systems Manager, Anda harus menentukan ARN lengkap dokumen untuk parameter permintaan`DocumentName`. Anda akan ditampilkan ARN penuh untuk dokumen SSM ketika Anda menjalankan perintah untuk daftar dokumen.

> **Note**
>
> Anda tidak perlu menentukan ARN untuk dokumen AWS publik (dokumen yang dimulai dengan`AWS-*`) atau dokumen yang Anda miliki.

Menggunakan dokumen SSM bersama (baris perintah)

Untuk mencantumkan semua dokumen SSM publik

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)
```

Untuk membuat daftar dokumen SSM pribadi yang telah dibagikan dengan Anda

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)
```

Untuk mencantumkan semua dokumen SSM yang tersedia untuk Anda

Linux & macOS

```
aws ssm list-documents
```

Windows

```
aws ssm list-documents
```

PowerShell

```
Get-SSMDocumentList
```

Untuk mendapatkan informasi tentang dokumen SSM yang telah dibagikan dengan Anda

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
```

Untuk menjalankan dokumen SSM bersama

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
```

Mencari dokumen SSM

Anda dapat mencari dokumen (SSM) AWS Systems Manager simpan untuk dokumen SSM dengan menggunakan pencarian teks bebas atau pencarian berbasis filter. Anda juga dapat dokumen favorit untuk membantu Anda menemukan dokumen SSM yang sering digunakan. Bagian berikut menjelaskan cara menggunakan fitur-fitur ini.

Menggunakan pencarian teks bebas

Kotak pencarian di halaman Dokumen Systems Manager mendukung pencarian teks bebas. Pencarian teks bebas membandingkan istilah pencarian atau istilah yang Anda masukkan terhadap nama dokumen di setiap dokumen SSM. Jika Anda memasukkan istilah pencarian tunggal, misalnya **`ansible`**, kemudian Systems Manager mengembalikan semua dokumen SSM di mana istilah ini ditemukan. Jika Anda memasukkan beberapa istilah pencarian, kemudian pencarian Systems Manager dengan menggunakan pernyataan `OR`. Misalnya, jika Anda menentukan **`ansible`** dan **`linux`**, kemudian pencarian mengembalikan semua dokumen dengan baik kata kunci dalam nama mereka.

Jika Anda memasukkan istilah pencarian teks bebas dan memilih opsi pencarian, seperti Jenis platform, lalu cari menggunakan pernyataan `AND` dan mengembalikan semua dokumen dengan kata kunci dalam nama mereka dan jenis platform yang ditentukan.

> **Note**
>
> Perhatikan rincian berikut tentang pencarian teks bebas.
>
> - Pencarian teks bebas adalah bukan kasus sensitif.

- Istilah pencarian membutuhkan minimal tiga karakter dan maksimal 20 karakter.
- Pencarian teks bebas menerima hingga lima istilah penelusuran.
- Jika Anda memasukkan spasi di antara istilah pencarian, sistem mencakup ruang saat mencari.
- Anda dapat menggabungkan pencarian teks bebas dengan opsi pencarian lain seperti Jenis dokumen atau Jenis platform.
- Prefiks Nama Dokumen memfilter dan mencari teks bebas tidak dapat digunakan bersama-sama. Mereka bersifat eksklusif.

Untuk mencari dokumen SSM

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

   Jika halaman beranda AWS Systems Manager terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.
3. Masukkan istilah pencarian Anda di kotak pencarian, dan tekan Enter.

Melakukan pencarian dokumen teks bebas dengan menggunakan AWS CLI

Untuk melakukan pencarian dokumen teks bebas dengan menggunakan CLI

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI.
2. Untuk melakukan pencarian dokumen teks bebas dengan satu istilah, jalankan perintah berikut. Dalam perintah ini, ganti *search_term* dengan informasi Anda sendiri.

   ```
   aws ssm list-documents --filters Key="SearchKeyword",Values="search_term"
   ```

   Inilah contohnya.

```
aws ssm list-documents --filters Key="SearchKeyword",Values="aws-asg" --region us-
east-2
```

Untuk mencari menggunakan beberapa istilah yang membuat pernyataan AND, jalankan perintah berikut. Dalam perintah ini, ganti *search_term_1* dan *search_term_2* dengan informasi Anda sendiri.

```
aws ssm list-documents --filters
 Key="SearchKeyword",Values="search_term_1","search_term_2","search_term_3" --
region us-east-2
```

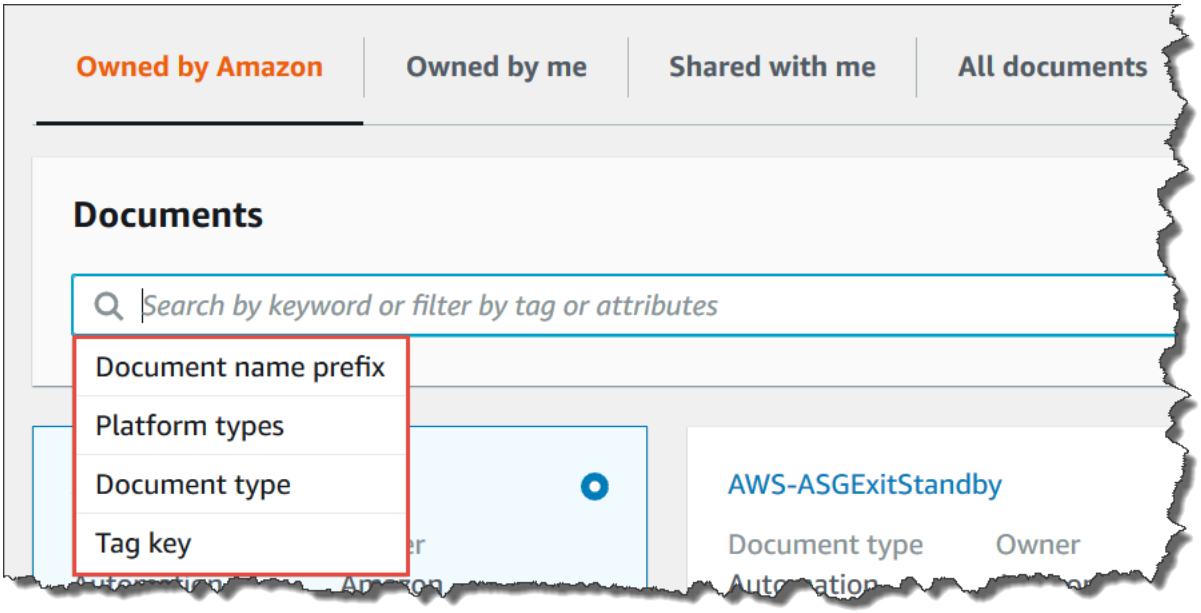
Inilah contohnya.

```
aws ssm list-documents --filters Key="SearchKeyword",Values="aws-asg","aws-
ec2","restart" --region us-east-2
```

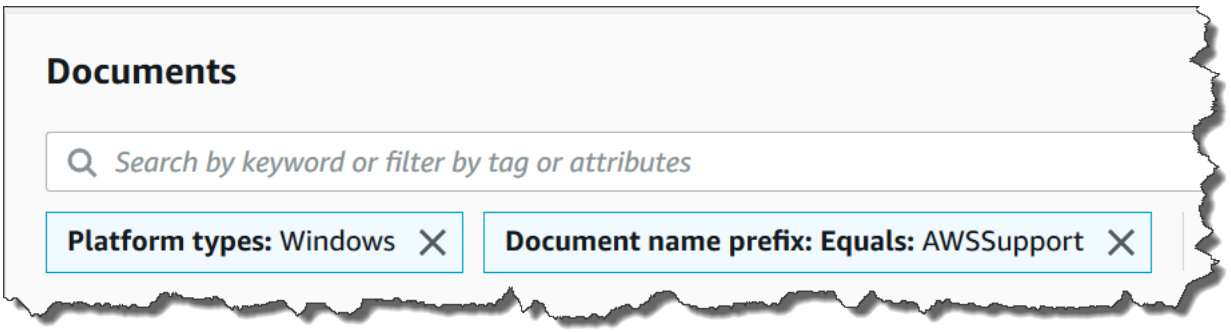
Menggunakan filter

Dokumen Systems Manager secara otomatis menampilkan filter berikut ketika Anda memilih kotak pencarian.

- Prefiks nama dokumen
- Jenis platform
- Jenis dokumen
- Tombol tag



Anda dapat mencari dokumen SSM dengan menggunakan filter tunggal. Jika Anda ingin mengembalikan kumpulan dokumen SSM yang lebih spesifik, Anda dapat menerapkan beberapa filter. Berikut adalah contoh pencarian yang menggunakan filter Jenis platform dan Prefiks nama dokumen.



Jika Anda menerapkan beberapa filter, Systems Manager membuat pernyataan pencarian yang berbeda berdasarkan filter yang Anda pilih:

- Jika Anda menerapkan filter sama beberapa kali, misalnya Prefiks nama dokumen, kemudian Systems Manager menggunakan pencarian dengan pernyataan `OR`. Sebagai contoh, jika Anda menentukan satu filter Prefiks nama dokumen=**`AWS`** dan filter kedua Prefiks nama dokumen = **`Lambda`**, kemudian pencarian mengembalikan semua dokumen dengan prefiks "`AWS`" dan semua dokumen dengan prefiks "`Lambda`".
- Jika Anda menerapkan filter berbeda, misalnya Prefiks nama dokumen dan Jenis platform, kemudian Systems Manager menggunakan pencarian dengan pernyataan `AND`. Sebagai contoh,

jika Anda menentukan Prefiks nama dokumen = filter **`AWS`** dan Jenis platform = filter **`Linux`**, kemudian pencarian mengembalikan semua dokumen prefiks "`AWS`" yang khusus untuk platform Linux.

> **Note**
>
> Pencarian yang menggunakan filter peka terhadap huruf besar-kecil.

Menambahkan dokumen ke favorit Anda

Untuk membantu Anda menemukan dokumen SSM yang sering digunakan, tambahkan dokumen ke favorit Anda. Anda dapat memfavoritkan hingga 20 dokumen per jenis dokumen, per Akun AWS danWilayah AWS. Anda dapat memilih, memodifikasi, dan melihat favorit Anda dari dokumenAWS Management Console. Prosedur berikut menjelaskan cara memilih, memodifikasi, dan melihat favorit Anda.

Untuk memfavoritkan dokumen SSM

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.
3. Pilih ikon bintang di sebelah nama dokumen yang ingin Anda favoritkan.

Untuk menghapus dokumen SSM dari favorit Anda

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.

3. Hapus pilihan ikon bintang di sebelah nama dokumen yang ingin Anda hapus dari favorit Anda.

Untuk melihat favorit Anda dari dokumen AWS Management Console

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.
3. Pilih tab Favorit.

Keamanan di AWS Systems Manager

Keamanan cloud di Amazon Web Services merupakan prioritas tertinggi. Sebagai pelanggan AWS, Anda mendapatkan manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan organisasi yang paling sensitif terhadap keamanan.

Keamanan menjadi tanggung jawab bersama antara AWS dan Anda. Model tanggung jawab bersama menjelaskan hal ini sebagai keamanan dari cloud dan keamanan dalam cloud:

- Keamanan cloud —AWS bertanggung jawab untuk melindungi infrastruktur yang berjalanLayanan AWS diAWS Cloud. AWSjuga menyediakan layanan yang dapat Anda gunakan dengan aman. Auditor pihak ketiga melakukan pengujian dan verifikasi secara berkala terhadap efektivitas keamanan kami sebagai bagian dari Program Kepatuhan AWS. Untuk mempelajari tentang program kepatuhan yang berlaku diAWS Systems Manager, lihat Layanan AWSdalam Cakupan menurut Program Kepatuhan yang .
- Keamanan dalam cloud — Tanggung jawab Anda ditentukan olehLayanan AWS yang Anda gunakan. Anda juga bertanggung jawab atas faktor lain termasuk sensitivitas data Anda, persyaratan perusahaan Anda, serta hukum dan peraturan yang berlaku.

Dokumentasi ini membantu Anda memahami cara menerapkan model tanggung jawab bersama saat menggunakan AWS Systems Manager. Topik berikut menunjukkan cara mengonfigurasi Systems Manager untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga mempelajari cara menggunakanLayanan AWS yang membantu Anda memantau dan mengamankanSystems Manager sumber daya Anda.

Topik

- Praktik terbaik keamanan untuk Systems Manager

Perlindungan data di AWS Systems Manager

Perlindungan data mengacu pada melindungi data saat transit (saat bepergian ke dan dariSystems Manager) dan saat istirahat (saat disimpan di pusat AWS data).

Model tanggung jawab AWS bersama model berlaku untuk perlindungan data diAWS Systems Manager. Seperti yang dijelaskan dalam model AWS ini, bertanggung jawab untuk melindungi infrastruktur global yang menjalankan semua AWS Cloud. Anda bertanggung jawab untuk mempertahankan kendali atas konten yang di-host pada infrastruktur ini. Anda juga bertanggung jawab atas tugas-tugas konfigurasi dan manajemen keamanan untuk Layanan AWS yang Anda gunakan. Lihat informasi yang lebih lengkap tentang privasi data dalam Pertanyaan Umum Privasi Data. Lihat informasi tentang perlindungan data di Eropa di pos blog Model Tanggung Jawab Bersama dan GDPR AWS di Blog Keamanan AWS .

Untuk tujuan perlindungan data, kami menyarankan Anda melindungi Akun AWS kredensyal dan mengatur pengguna individu dengan AWS IAM Identity Center atau AWS Identity and Access Management (IAM). Dengan cara itu, setiap pengguna hanya diberi izin yang diperlukan untuk memenuhi tanggung jawab tugasnya. Kami juga menyarankan supaya Anda mengamankan data dengan cara-cara berikut:

- Gunakan autentikasi multi-faktor (MFA) pada setiap akun.
- Gunakan SSL/TLS untuk berkomunikasi dengan sumber daya. AWS Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail.
- Gunakan solusi AWS enkripsi, bersama dengan semua kontrol keamanan default di dalamnya Layanan AWS.
- Gunakan layanan keamanan terkelola lanjut seperti Amazon Macie, yang membantu menemukan dan mengamankan data sensitif yang disimpan di Amazon S3.
- Jika Anda memerlukan modul kriptografi tervalidasi FIPS 140-2 saat mengakses AWS melalui antarmuka baris perintah atau API, gunakan titik akhir FIPS. Lihat informasi yang lebih lengkap tentang titik akhir FIPS yang tersedia di Standar Pemrosesan Informasi Federal (FIPS) 140-2.

Kami sangat merekomendasikan agar Anda tidak pernah memasukkan informasi identifikasi yang sensitif, seperti nomor rekening pelanggan Anda, ke dalam tanda atau bidang isian bebas seperti

bidang Nama. Ini termasuk saat Anda bekerja dengan Systems Manager atau lainnya Layanan AWS menggunakan konsol, API AWS CLI, atau AWS SDK. Data apa pun yang Anda masukkan ke dalam tanda atau bidang isian bebas yang digunakan untuk nama dapat digunakan untuk log penagihan atau log diagnostik. Saat Anda memberikan URL ke server eksternal, kami sangat menganjurkan supaya Anda tidak menyertakan informasi kredensial di dalam URL untuk memvalidasi permintaan Anda ke server itu.

Enkripsi data

Enkripsi saat istirahat

Parameter Parameter Store

Jenis parameter yang dapat Anda buatParameter Store, kemampuan AWS Systems Manager, termasuk, `String`, `StringList`, dan`SecureString`.

Untuk mengenkripsi nilai `SecureString` parameter, Parameter Store gunakan AWS KMS key in AWS Key Management Service (AWS KMS). AWS KMS menggunakan kunci yang dikelola pelanggan atau Kunci yang dikelola AWS untuk mengenkripsi nilai parameter dalam database AWS terkelola.

> **Important**
>
> Jangan simpan data sensitif dalam parameter `String` atau `StringList`. Untuk semua data sensitif yang harus tetap dienkripsi, gunakan hanya tipe parameter `SecureString`. Untuk informasi selengkapnya, lihat Apa itu parameter? dan Membatasi akses ke parameter Systems Manager menggunakan kebijakan IAM.

Konten dalam ember S3

Sebagai bagian dari Systems Manager operasi, Anda dapat memilih untuk mengunggah atau menyimpan data dalam satu atau beberapa bucket Amazon Simple Storage Service (Amazon S3).

Untuk informasi tentang enkripsi bucket S3, lihat Melindungi data menggunakan enkripsi dan perlindungan Data di Amazon S3 di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

Berikut ini adalah jenis data yang dapat Anda unggah atau simpan di bucket S3 sebagai bagian dari aktivitas AndaSystems Manager:

- Output dari perintah diRun Command, kemampuan AWS Systems Manager
- Paket dalamDistributor, kemampuan AWS Systems Manager
- Menambal log operasiPatch Manager, kemampuan AWS Systems Manager
- Patch Managerdaftar penggantian tambalan
- Skrip atau Ansible Playbook untuk dijalankan dalam alur kerja runbook di Otomasi, kemampuan AWS Systems Manager
- Chef InSpecprofil untuk digunakan dengan pemindaian dalam Kepatuhan, kemampuan AWS Systems Manager
- AWS CloudTrail log
- Riwayat sesi log inSession Manager, kemampuan AWS Systems Manager
- Laporan dariExplorer, kemampuan AWS Systems Manager
- OpsData dariOpsCenter, kemampuan AWS Systems Manager
- AWS CloudFormation template untuk digunakan dengan alur kerja Otomasi
- Data kepatuhan dari pemindaian sinkronisasi data sumber daya
- Output permintaan untuk membuat atau mengedit asosiasi diState Manager, kemampuan AWS Systems Manager, pada node terkelola
- Dokumen Systems Manager khusus (SSM dokumen) yang dapat Anda jalankan menggunakan AWS Dokumen SSM terkelola `AWS-RunDocument`

CloudWatch Grup log log

Sebagai bagian dari Systems Manager operasi, Anda dapat memilih untuk mengalirkan data ke satu atau beberapa grup CloudWatch log Amazon Logs.

Untuk informasi tentang enkripsi grup CloudWatch log log, lihat Mengenkripsi data CloudWatch log di Log menggunakan AWS Key Management Service Panduan Pengguna CloudWatch Log Amazon.

Berikut ini adalah jenis data yang mungkin telah dialirkan ke grup CloudWatch log Log sebagai bagian dari Systems Manager aktivitas Anda:

- Output dari Run Command perintah
- Output dari skrip dijalankan menggunakan tindakan `aws:executeScript` dalam runbook otomatisasi

- Session Managerlog riwayat sesi
- Log dari SSM Agent node terkelola

Enkripsi dalam bergerak

Kami menyarankan Anda menggunakan protokol enkripsi seperti Transport Layer Security (TLS) untuk mengenkripsi data sensitif dalam perjalanan antara klien dan node Anda.

Systems Managermenyediakan dukungan berikut untuk enkripsi data Anda dalam perjalanan.

Koneksi ke Systems Manager titik akhir API

Systems ManagerTitik akhir API hanya mendukung koneksi aman melalui HTTPS. Saat Anda mengelola Systems Manager sumber daya dengan AWS Management Console, AWS SDK, atau Systems Manager API, semua komunikasi dienkripsi dengan Transport Layer Security (TLS). Untuk daftar lengkap titik akhir API, lihat Layanan AWS titik akhir di. Referensi Umum Amazon Web Services

Instans terkelola

AWS menyediakan konektivitas aman dan pribadi antara instans Amazon Elastic Compute Cloud (Amazon EC2). Selain itu, kami secara otomatis mengenkripsi lalu lintas dalam transit antar instans yang didukung di virtual private cloud (VPC) atau VPC sejawat, dengan menggunakan algoritma AEAD dengan enkripsi 256-bit. Fitur enkripsi ini menggunakan kemampuan offload dari perangkat keras yang mendasari, dan tidak ada dampak pada kinerja jaringan. Instans yang didukung adalah: C5n, G4, I3en, M5dn, M5n, P3dn, R5dn, dan R5n.

Session Managersesi

Secara default, Session Manager menggunakan TLS 1.2 untuk mengenkripsi data sesi yang dikirimkan antara mesin lokal pengguna di akun Anda dan instans EC2 Anda. Anda juga dapat memilih untuk mengenkripsi lebih lanjut data dalam perjalanan menggunakan AWS KMS key yang telah dibuat di AWS KMS. AWS KMS enkripsi tersedia untuk`Standard_Stream`,`InteractiveCommands`, dan jenis `NonInteractiveCommands` sesi.

Run Commandakses

Secara default, akses jarak jauh ke node Anda menggunakan Run Command dienkripsi menggunakan TLS 1.2, dan permintaan untuk membuat koneksi ditandatangani menggunakan SigV4.

Privasi lalu lintas jaringan internet

Anda dapat menggunakan Amazon Virtual Private Cloud (Amazon VPC) untuk membuat batasan antara sumber daya di node terkelola dan mengontrol lalu lintas di antara mereka, jaringan lokal, dan internet. Untuk detailnya, lihat Membuat titik akhir VPC.

Untuk informasi selengkapnya tentang keamanan Amazon Virtual Private Cloud, lihat Privasi lalu lintas Internetwork di Amazon VPC di Panduan Pengguna Amazon VPC.

Identity and access management untuk AWS Systems Manager

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu administrator mengontrol akses ke sumber daya AWS secara aman. Administrator IAM mengontrol siapa yang dapat terautentikasi (masuk) dan berwenang (memiliki izin) untuk menggunakan sumber daya Systems Manager. IAM adalah layanan Layanan AWS yang dapat Anda gunakan tanpa dikenakan biaya tambahan.

Topik

Audiens

Cara menggunakan AWS Identity and Access Management (IAM) berbeda, tergantung pada pekerjaan yang Anda lakukan di Systems Manager.

Pengguna layanan – Jika Anda menggunakan layanan Systems Manager untuk melakukan tugas Anda, administrator Anda akan memberikan kredensial dan izin yang Anda butuhkan. Saat Anda menggunakan lebih banyak fitur Systems Manager untuk melakukan pekerjaan, Anda mungkin memerlukan izin tambahan. Memahami cara mengelola akses dapat membantu Anda meminta izin

yang tepat dari administrator Anda. Jika Anda tidak dapat mengakses fitur di Systems Manager, lihat Pemecahan masalah identitas dan akses AWS Systems Manager.

Administrator layanan – Jika Anda bertanggung jawab atas sumber daya Systems Manager di perusahaan Anda, Anda mungkin memiliki akses penuh ke Systems Manager. Tugas Anda adalah menentukan Systems Manager fitur dan sumber daya mana yang harus diakses pengguna layanan Anda. Kemudian, Anda harus mengirimkan permintaan kepada administrator IAM Anda untuk mengubah izin pengguna layanan Anda. Tinjau informasi di halaman ini untuk memahami konsep dasar IAM. Untuk mempelajari lebih lanjut tentang cara perusahaan Anda dapat menggunakan IAM dengan Systems Manager, lihat Cara kerja AWS Systems Manager dengan IAM.

Administrator IAM – Jika Anda adalah administrator IAM, Anda mungkin ingin belajar dengan lebih detail tentang cara Anda menulis kebijakan untuk mengelola akses ke Systems Manager. Untuk melihat contoh kebijakan berbasis identitas Systems Manager yang dapat Anda gunakan di IAM, lihat Contoh kebijakan berbasis identitas AWS Systems Manager.

Autentikasi menggunakan identitas

Autentikasi adalah cara Anda untuk masuk ke AWS menggunakan kredensial identitas Anda. Anda harus terautentikasi (masuk ke AWS) sebagai Pengguna root akun AWS, sebagai pengguna IAM, atau dengan mengambil peran IAM.

Anda dapat masuk ke AWS sebagai identitas terfederasi dengan menggunakan kredensial yang disediakan melalui sumber identitas. Pengguna AWS IAM Identity Center Pengguna (Pusat Identitas IAM), autentikasi Single Sign-On perusahaan Anda, dan kredensial Google atau Facebook Anda merupakan contoh identitas terfederasi. Saat Anda masuk sebagai identitas gabungan, administrator Anda sebelumnya menyiapkan federasi identitas menggunakan peran IAM. Ketika Anda mengakses AWS dengan menggunakan federasi, Anda secara tidak langsung mengambil suatu peran.

Bergantung pada jenis pengguna Anda, Anda dapat masuk ke AWS Management Console atau portal akses AWS. Untuk informasi selengkapnya tentang cara masuk ke AWS, lihat Cara masuk ke Akun AWS dalam Panduan Pengguna AWS Sign-In.

Jika Anda mengakses AWS secara terprogram, AWS memberikan Kit Pengembangan Perangkat Lunak (SDK) dan antarmuka baris perintah (CLI) untuk menandatangani permintaan Anda secara kriptografis dengan menggunakan kredensial Anda. Jika Anda tidak menggunakan peralatan AWS, Anda harus menandatangani permintaan sendiri. Untuk informasi selengkapnya tentang cara menggunakan metode yang disarankan untuk menandatangani permintaan sendiri, lihat Menandatangani permintaan API AWS dalam Panduan Pengguna IAM.

Apa pun metode autentikasi yang digunakan, Anda mungkin diminta untuk menyediakan informasi keamanan tambahan. Sebagai contoh, AWS menyarankan Anda menggunakan autentikasi multi-faktor (MFA) untuk meningkatkan keamanan akun Anda. Untuk mempelajari lebih lanjut, lihat Autentikasi multi-faktor dalam Panduan Pengguna AWS IAM Identity Center dan Menggunakan autentikasi multi-faktor (MFA) di AWS dalam Panduan Pengguna IAM.

Pengguna root Akun AWS

Ketika membuat Akun AWS, Anda memulai dengan satu identitas masuk yang memiliki akses penuh ke semua Layanan AWS dan sumber daya di akun tersebut. Identitas ini disebut pengguna root Akun AWS dan diakses dengan cara masuk menggunakan alamat email dan kata sandi yang Anda gunakan untuk membuat akun. Kami sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari Anda. Lindungi kredensial pengguna root Anda dan gunakan kredensial tersebut untuk melakukan tugas yang hanya dapat dilakukan pengguna root. Untuk daftar tugas lengkap yang mengharuskan Anda masuk sebagai pengguna root, lihat Tugas yang memerlukan kredensial pengguna root dalam Panduan Pengguna IAM.

Pengguna dan grup IAM

Pengguna IAM adalah identitas dalam Akun AWS Anda yang memiliki izin khusus untuk satu orang atau aplikasi. Jika memungkinkan, sebaiknya andalkan kredensial temporer, dan bukan membuat pengguna IAM yang memiliki kredensial jangka panjang seperti kata sandi dan kunci akses. Namun, jika Anda memiliki kasus penggunaan khusus yang memerlukan kredensial jangka panjang dengan pengguna IAM, sebaiknya rotasikan kunci akses. Untuk informasi selengkapnya, lihat Merotasi kunci akses secara teratur untuk kasus penggunaan yang memerlukan kredensial jangka panjang dalam Panduan Pengguna IAM.

Grup IAM adalah identitas yang menentukan kumpulan pengguna IAM. Anda tidak dapat masuk sebagai grup. Anda dapat menggunakan grup untuk menentukan izin untuk beberapa pengguna sekaligus. Grup membuat izin lebih mudah dikelola untuk sekelompok besar pengguna. Misalnya, Anda dapat memiliki grup yang bernama IAMAdmins dan memberikan izin kepada grup tersebut untuk mengelola sumber daya IAM.

Pengguna berbeda dari peran. Pengguna secara unik terkait dengan satu orang atau aplikasi, tetapi peran tersebut dimaksudkan untuk dapat diambil oleh siapa pun yang membutuhkannya. Pengguna memiliki kredensial jangka panjang permanen, tetapi peran memberikan kredensial sementara. Untuk mempelajari selengkapnya, silakan lihat Kapan harus membuat pengguna IAM (bukan peran) dalam Panduan Pengguna IAM.

Peran IAM

Peran IAM merupakan identitas dalam Akun AWS Anda yang memiliki izin khusus. Peran ini mirip dengan pengguna IAM, tetapi tidak terkait dengan orang tertentu. Anda dapat mengambil peran IAM untuk sementara dalam AWS Management Console dengan berganti peran. Anda dapat mengambil peran dengan cara memanggil operasi API AWS CLI atau AWS atau menggunakan URL kustom. Untuk informasi selengkapnya tentang metode untuk menggunakan peran, lihat Menggunakan peran IAM dalam Panduan Pengguna IAM.

Peran IAM dengan kredensial sementara berguna dalam situasi berikut:

- Akses pengguna gabungan – Untuk menetapkan izin ke sebuah identitas gabungan, Anda dapat membuat peran dan menentukan izin untuk peran tersebut. Saat identitas terfederasi diautentikasi, identitas tersebut dikaitkan dengan peran dan diberikan izin yang ditentukan oleh peran. Untuk informasi tentang peran untuk federasi, lihat Membuat peran untuk Penyedia Identitas pihak ketiga dalam Panduan Pengguna IAM. Jika Anda menggunakan Pusat Identitas IAM, Anda mengonfigurasi sekumpulan izin. Untuk mengontrol apa yang dapat diakses identitas Anda setelah identitas tersebut diautentikasi, Pusat Identitas IAM mengaitkan izin yang ditetapkan ke peran dalam IAM. Untuk informasi tentang rangkaian izin, lihat Rangkaian izin dalam Panduan Pengguna AWS IAM Identity Center.
- Izin pengguna IAM sementara – Pengguna atau peran IAM dapat mengambil peran IAM guna mendapatkan berbagai izin secara sementara untuk tugas tertentu.
- Akses lintas akun – Anda dapat menggunakan peran IAM untuk mengizinkan seseorang (pengguna utama tepercaya) dengan akun berbeda untuk mengakses sumber daya yang ada di akun Anda. Peran adalah cara utama untuk memberikan akses lintas akun. Namun, pada beberapa Layanan AWS, Anda dapat menyertakan kebijakan secara langsung ke sumber daya (bukan menggunakan peran sebagai proksi). Untuk mempelajari perbedaan antara kebijakan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat Bagaimana peran IAM berbeda dari kebijakan berbasis sumber daya dalam Panduan Pengguna IAM.
- Akses lintas layanan – Sebagian Layanan AWS menggunakan fitur di Layanan AWS lainnya. Contoh, ketika Anda melakukan panggilan dalam layanan, umumnya layanan tersebut menjalankan aplikasi di Amazon EC2 atau menyimpan objek di Amazon S3. Suatu layanan mungkin melakukan hal tersebut menggunakan izin pengguna utama panggilan, menggunakan peran layanan, atau peran terkait layanan.
  - Sesi akses maju (FAS) – Ketika Anda menggunakan pengguna IAM atau peran IAM untuk melakukan tindakan di AWS, Anda akan dianggap sebagai seorang pengguna utama. Saat menggunakan beberapa layanan, Anda mungkin melakukan tindakan yang kemudian dilanjutkan

oleh tindakan lain pada layanan yang berbeda. FAS menggunakan izin dari pengguna utama untuk memanggil Layanan AWS, yang dikombinasikan dengan Layanan AWS yang diminta untuk membuat permintaan ke layanan hilir. Permintaan FAS hanya diajukan saat layanan menerima permintaan yang memerlukan interaksi dengan Layanan AWS lain atau sumber daya lain untuk diselesaikan. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan ketika mengajukan permintaan FAS, lihat Meneruskan sesi akses.

- Peran IAM – Peran layanan adalah peran IAM yang diambil layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, memodifikasi, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat Membuat peran untuk mendelegasikan izin ke Layanan AWS dalam Panduan pengguna IAM.
- Peran terkait layanan – Peran terkait layanan adalah tipe peran layanan yang terkait dengan Layanan AWS. Layanan tersebut dapat mengambil peran untuk melakukan sebuah tindakan atas nama Anda. Peran terkait layanan akan muncul di Akun AWS Anda dan dimiliki oleh layanan tersebut. Administrator IAM dapat melihat, tetapi tidak dapat mengedit izin untuk peran terkait layanan.

- Aplikasi yang berjalan di Amazon EC2 – Anda dapat menggunakan peran IAM untuk mengelola kredensial sementara untuk aplikasi yang berjalan di instans EC2 dan mengajukan permintaan API AWS CLI atau AWS. Cara ini lebih dianjurkan daripada menyimpan kunci akses dalam instans EC2. Untuk menetapkan peran AWS ke instans EC2 dan menyediakannya bagi semua aplikasinya, Anda dapat membuat profil instans yang dilampirkan ke instans tersebut. Profil instans berisi peran dan memungkinkan program yang berjalan di instans EC2 mendapatkan kredensial sementara. Untuk informasi selengkapnya, lihat Menggunakan peran IAM untuk memberikan izin ke aplikasi yang berjalan di instans Amazon EC2 dalam Panduan Pengguna IAM.

Untuk mempelajari apakah kita harus menggunakan peran IAM atau pengguna IAM, lihat Kapan harus membuat peran IAM (bukan pengguna) dalam Panduan Pengguna IAM.

Mengelola akses menggunakan kebijakan

Anda mengendalikan akses di AWS dengan membuat kebijakan dan melampirkannya ke identitas atau sumber daya AWS. Kebijakan adalah objek di AWS yang, ketika terkait dengan identitas atau sumber daya, akan menentukan izinnya. AWS mengevaluasi kebijakan-kebijakan tersebut ketika seorang pengguna utama (pengguna, pengguna root, atau sesi peran) mengajukan permintaan. Izin dalam kebijakan menentukan apakah permintaan diizinkan atau ditolak. Sebagian besar kebijakan disimpan di AWS sebagai dokumen JSON. Untuk informasi selengkapnya tentang struktur dan

isi dokumen kebijakan JSON, silakan lihat Gambaran Umum kebijakan JSON dalam Panduan Pengguna IAM.

Administrator dapat menggunakan kebijakan JSON AWS untuk menentukan secara spesifik siapa yang memiliki akses terhadap apa. Artinya, pengguna utama manakah yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Secara default, pengguna dan peran tidak memiliki izin. Untuk memberikan izin kepada pengguna untuk melakukan tindakan pada sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM. Administrator kemudian dapat menambahkan kebijakan IAM ke peran, dan pengguna dapat menjalankan peran.

Kebijakan IAM mendefinisikan izin untuk suatu tindakan terlepas dari metode yang Anda gunakan untuk operasi. Sebagai contoh, anggap saja Anda memiliki kebijakan yang mengizinkan tindakan `iam:GetRole`. Pengguna dengan kebijakan tersebut dapat memperoleh informasi peran dari AWS Management Console, AWS CLI, atau API AWS.

Kebijakan berbasis identitas

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang dapat Anda lampirkan ke sebuah identitas, seperti pengguna IAM, grup pengguna IAM, atau peran IAM. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan pengguna dan peran, di sumber daya mana, dan dengan ketentuan apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Membuat kebijakan IAM dalam Panduan Pengguna IAM.

Kebijakan berbasis identitas dapat dikategorikan lebih lanjut sebagai kebijakan inline atau kebijakan terkelola. Kebijakan inline disematkan langsung ke satu pengguna, grup, atau peran. Kebijakan terkelola adalah kebijakan mandiri yang dapat Anda lampirkan ke beberapa pengguna, grup, dan peran di Akun AWS Anda. Kebijakan terkelola meliputi kebijakan yang dikelola AWS dan kebijakan yang dikelola pelanggan. Untuk mempelajari cara memilih antara kebijakan terkelola atau kebijakan inline, lihat Memilih antara kebijakan terkelola dan kebijakan inline dalam Panduan Pengguna IAM.

Untuk informasi tentang kebijakan AWS terkelolaSystems Manager, lihatKebijakan terkelola AWS Systems Manager.

Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya,

administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya yang dilampiri kebijakan tersebut, kebijakan ini menentukan jenis tindakan yang dapat dilakukan oleh pengguna utama tertentu di sumber daya tersebut dan apa ketentuannya. Anda harus menentukan pengguna utama dalam kebijakan berbasis sumber daya. Pengguna utama dapat mencakup akun, pengguna, peran, pengguna gabungan, atau Layanan AWS.

Kebijakan berbasis sumber daya merupakan kebijakan inline yang terletak di layanan tersebut. Anda tidak dapat menggunakan kebijakan yang dikelola AWS dari IAM dalam kebijakan berbasis sumber daya.

Daftar kontrol akses (ACL)

Daftar kontrol akses (ACL) mengendalikan pengguna utama mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACL serupa dengan kebijakan berbasis sumber daya, meskipun tidak menggunakan format dokumen kebijakan JSON.

Amazon S3, AWS WAF, dan Amazon VPC adalah contoh layanan yang mendukung ACL. Untuk mempelajari ACL selengkapnya, silakan lihat Gambaran umum daftar kontrol akses (ACL) di Panduan Developer Layanan Penyimpanan Ringkas Amazon.

Tipe kebijakan lain

AWS mendukung jenis kebijakan tambahan yang kurang umum. Tipe-tipe kebijakan ini dapat mengatur izin maksimum yang diberikan kepada Anda berdasarkan tipe kebijakan yang lebih umum.

- Batasan izin – Batasan izin adalah fitur lanjutan di mana Anda menetapkan izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas kepada entitas IAM (pengguna atau peran IAM). Anda dapat menetapkan batasan izin untuk suatu entitas. Izin yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas milik entitas dan batasan izinnya. Kebijakan berbasis sumber daya yang menentukan pengguna atau peran dalam bidang `Principal` tidak dibatasi oleh batasan izin. Penolakan secara eksplisit terhadap salah satu kebijakan ini akan mengesampingkan izin tersebut. Untuk informasi selengkapnya tentang batasan izin, lihat Batasan izin untuk entitas IAM dalam Panduan Pengguna IAM.
- Kebijakan kontrol layanan (SCP) – SCP adalah kebijakan JSON yang menentukan izin maksimum untuk sebuah organisasi atau unit organisasi (OU) di AWS Organizations. AWS Organizations adalah layanan untuk mengelompokkan dan mengelola beberapa akun Akun AWS yang dimiliki bisnis Anda secara terpusat. Jika Anda mengaktifkan semua fitur di organisasi, Anda dapat menerapkan kebijakan kontrol layanan (SCP) ke salah satu atau semua akun Anda. SCP membatasi izin untuk entitas dalam akun anggota, termasuk setiap Pengguna root akun AWS.

Untuk informasi selengkapnya tentang Organisasi dan SCP, lihat Cara kerja SCP dalam Panduan Pengguna AWS Organizations.

- Kebijakan sesi – Kebijakan sesi adalah kebijakan lanjutan yang Anda teruskan sebagai parameter saat Anda membuat sesi sementara secara terprogram untuk peran atau pengguna gabungan. Izin sesi yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas pengguna atau peran dan kebijakan sesi. Izin juga bisa datang dari kebijakan berbasis sumber daya. Penolakan eksplisit di salah satu kebijakan ini akan membatalkan izin tersebut. Untuk informasi selengkapnya, lihat Kebijakan sesi dalam Panduan Pengguna IAM.

Berbagai jenis kebijakan

Jika beberapa jenis kebijakan diberlakukan untuk satu permintaan, izin yang dihasilkan lebih rumit untuk dipahami. Untuk mempelajari cara AWS menentukan apakah akan mengizinkan permintaan ketika ada beberapa jenis kebijakan, lihat Logika evaluasi kebijakan dalam Panduan Pengguna IAM.

Cara kerja AWS Systems Manager dengan IAM

Sebelum Anda menggunakan AWS Identity and Access Management (IAM) untuk mengelola akses AWS Systems Manager, Anda harus memahami fitur IAM apa yang tersedia untuk digunakan. Systems Manager Untuk mendapatkan tampilan tingkat tinggi tentang bagaimana Systems Manager dan Layanan AWS pekerjaan lainnya dengan IAM, lihat Layanan AWS bahwa bekerja dengan IAM di Panduan Pengguna IAM.

Topik

- Kebijakan berbasis identitas Systems Manager
- Systems Manager Kebijakan berbasis sumber daya
- Otorisasi berdasarkan tag Systems Manager
- Peran IAM Systems Manager

Kebijakan berbasis identitas Systems Manager

Dengan kebijakan berbasis identitas IAM, Anda dapat menentukan tindakan dan sumber daya yang diizinkan atau ditolak, dan juga ketentuan di mana tindakan tersebut diperbolehkan atau ditolak. Systems Manager mendukung tindakan, sumber daya, dan kunci kondisi tertentu. Untuk mempelajari semua elemen yang Anda gunakan dalam kebijakan JSON, lihat Referensi elemen kebijakan IAM JSON dalam Panduan Pengguna IAM.

Tindakan

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, pengguna utama mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Elemen `Action` dari kebijakan JSON menjelaskan tindakan yang dapat Anda gunakan untuk mengizinkan atau menolak akses dalam sebuah kebijakan. Tindakan kebijakan biasanya memiliki nama yang sama dengan operasi AWS API terkait. Ada beberapa pengecualian, misalnya tindakan hanya izin yang tidak memiliki operasi API yang cocok. Ada juga beberapa operasi yang memerlukan beberapa tindakan dalam suatu kebijakan. Tindakan tambahan ini disebut tindakan dependen.

Sertakan tindakan dalam kebijakan untuk memberikan izin untuk melakukan operasi terkait.

Tindakan kebijakan di Systems Manager menggunakan prefiks berikut sebelum tindakan: `ssm:`. Misalnya, untuk memberikan izin kepada seseorang untuk membuat Systems Manager parameter (parameter SSM) dengan operasi Systems Manager `PutParameter` API, Anda menyertakan `ssm:PutParameter` tindakan tersebut dalam kebijakan mereka. Pernyataan kebijakan harus menyertakan elemen `Action` atau `NotAction`. Systems Manager menentukan set tindakan sendiri yang menjelaskan tugas yang dapat Anda lakukan dengan layanan ini.

Untuk menetapkan beberapa tindakan dalam satu pernyataan, pisahkan tindakan-tindakan tersebut menggunakan koma seperti berikut:

```
"Action": [
      "ssm:action1",
      "ssm:action2"
```

> **Note**
>
> Kemampuan berikut AWS Systems Manager menggunakan awalan yang berbeda sebelum tindakan.
>
> - AWS AppConfig menggunakan awalan `appconfig:` sebelum tindakan.
> - Manajer Insiden menggunakan awalan `ssm-incidents:` atau `ssm-contacts:` sebelum tindakan.
> - Systems Manager GUI Connect menggunakan awalan `ssm-guiconnect` sebelum tindakan.

Anda dapat menentukan beberapa tindakan menggunakan wildcard (*). Sebagai contoh, untuk menentukan semua tindakan yang dimulai dengan kata `Describe`, sertakan tindakan berikut:

```
"Action": "ssm:Describe*"
```

Untuk melihat daftar tindakan Systems Manager, lihat Tindakan yang Ditentukan oleh AWS Systems Manager dalam Referensi Otorisasi Layanan.

Sumber daya

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, pengguna utama mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Elemen kebijakan JSON `Resource` menentukan objek atau beberapa objek yang menjadi target penerapan tindakan. Pernyataan harus menyertakan elemen `Resource` atau `NotResource`. Praktik terbaiknya, tentukan sumber daya menggunakan Amazon Resource Name (ARN). Anda dapat melakukan ini untuk tindakan yang mendukung jenis sumber daya tertentu, yang dikenal sebagai izin tingkat sumber daya.

Untuk tindakan yang tidak mendukung izin di tingkat sumber daya, misalnya operasi pencantuman, gunakan wildcard (*) untuk mengindikasikan bahwa pernyataan tersebut berlaku untuk semua sumber daya.

```
"Resource": "*"
```

Misalnya, sumber daya jendela Systems Manager pemeliharaan memiliki format ARN berikut.

```
arn:aws:ssm:region:account-id:maintenancewindow/window-id
```

Untuk menentukan jendela pemeliharaan MW-0C50858D01Example dalam pernyataan Anda di Wilayah AS Timur (Ohio), Anda akan menggunakan ARN yang mirip dengan yang berikut ini.

```
"Resource": "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-0c50858d01EXAMPLE"
```

Untuk menentukan semua windowa pemeliharaan milik akun tertentu, gunakan wildcard (*).

```
"Resource": "arn:aws:ssm:region:123456789012:maintenancewindow/*"
```

Untuk operasi `Parameter Store` API, Anda dapat menyediakan atau membatasi akses ke semua parameter dalam satu tingkat hierarki dengan menggunakan nama hierarkis dan kebijakan AWS Identity and Access Management (IAM) sebagai berikut.

```
"Resource": "arn:aws:ssm:region:123456789012:parameter/Dev/ERP/Oracle/*"
```

Beberapa tindakan Systems Manager, seperti tindakan untuk membuat sumber daya, tidak dapat dilakukan pada sumber daya tertentu. Dalam kasus tersebut, Anda harus menggunakan wildcard (*).

```
"Resource": "*"
```

Beberapa operasi Systems Manager API menerima banyak sumber daya. Untuk menentukan beberapa tindakan dalam satu pernyataan, pisahkan ARN dengan koma seperti berikut.

```
"Resource": [
      "resource1",
      "resource2"
```

> **Note**
>
> Sebagian besar Layanan AWS memperlakukan titik dua (:) atau garis miring (/) sebagai karakter yang sama di ARN. Namun, Systems Manager membutuhkan kecocokan yang tepat dalam pola dan aturan sumber daya. Saat membuat pola peristiwa, pastikan untuk menggunakan karakter ARN yang benar sehingga cocok dengan ARN sumber daya.

Tabel di bawah ini menjelaskan format ARN untuk jenis sumber daya yang didukung oleh. Systems Manager

> **Note**
>
> Perhatikan pengecualian berikut untuk format ARN.
>
> - Kemampuan berikut AWS Systems Manager menggunakan awalan yang berbeda sebelum tindakan.
>   - AWS AppConfig menggunakan awalan `appconfig:` sebelum tindakan.

- Manajer Insiden menggunakan awalan `ssm-incidents:` atau `ssm-contacts:` sebelum tindakan.
- Systems Manager GUI Connect menggunakan awalan `ssm-guiconnect` sebelum tindakan.

- Dokumen dan sumber daya definisi otomatisasi yang dimiliki oleh Amazon, serta parameter publik yang disediakan oleh Amazon dan sumber pihak ketiga, tidak menyertakan ID akun dalam format ARN mereka. Sebagai contoh:
  - Dokumen `AWS-RunPatchBaseline` SSM:

    ```
    arn:aws:ssm:us-east-2:::document/AWS-RunPatchBaseline
    ```

  - Runbook `AWS-ConfigureMaintenanceWindows` otomatisasi:

    ```
    arn:aws:ssm:us-east-2:::automation-definition/AWS-
    ConfigureMaintenanceWindows
    ```

  - Parameter publik`/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
    ```

  Untuk informasi selengkapnya tentang ketiga jenis sumber daya ini, lihat topik berikut:

  - Bekerja dengan dokumen
  - Menjalankan Otomatisasi
  - Menggunakan dengan parameter publik

Jenis sumber daya	Format ARN
Aplikasi (AWS AppConfig)	*arn:aws:appconfig: wilayah: account-id:application/application-id*
Asosiasi	arn:aws:ssm:*region*:*account-id* :association/*association-id*
Eksekusi otomatisasi	*arn:aws:ssm: wilayah: account-id: automation-execution/ automation-execution-id*

Jenis sumber daya	Format ARN
Definisi otomatisasi (dengan versi subsumber daya)	*arn:aws:ssm: wilayah: account-id:automation-definition/: version-id automation-definition-id*[1]
Profil konfigurasi (AWS AppConfig)	*arn:aws:appconfig: wilayah: account-id:application/application-id /configurationprofile/ configurationprofile-id*
Kontak (Manajer Insiden)	*arn:aws:ssm-contacts: region: account-id:contact /contact-alias*
Strategi penyebaran ()AWS AppConfig	*arn:aws:appconfig: wilayah: account-id:deploymentstrategy/ deploymentstrategy-id*
Dokumen	arn:aws:ssm:*region*:*account-id* :document/*document-name*
Lingkungan (AWS AppConfig)	*arn:aws:appconfig: wilayah: account-id:application/application-id/environment/environment-id*
Insiden	*arn:aws:ssm-insidents: wilayah: account-id:insident-record//insident-id response-plan-name*
Jendela pemeliharaan	arn:aws:ssm:*region*:*account-id* :maintenancewindow/*window-id*
Node terkelola	*arn:aws:ssm: wilayah: account-id:managed-instance / managed-node-id*
Inventaris simpul terkelola	*arn:aws:ssm: wilayah: account-id:/managed-instance-inventory managed-node-id*
OpsItem	*arn:aws:ssm: wilayah: account-id:opsitem/ -id OpsItem*

Jenis sumber daya	Format ARN
Parameter	Parameter satu tingkat: • arn:aws:ssm:*region*:*account-id* :parameter/*parameter-name*/ Parameter bernama dengan konstruksi hierarkis: • *arn:aws:ssm: wilayah: account-id: parameter//level-2 /level-3/ level-4/level-5 parameter-name-root* [2]
Garis dasar patch	*arn:aws:ssm: wilayah: account-id: patchbaseline/ patch-baseline-id*
Rencana respons	*arn:aws:ssm-insidents: wilayah: account-id:response-plan/ response-plan-name*
Sesi	arn:aws:ssm:*region*:*account-id* :session/*session-id* [3]
Semua sumber daya Systems Manager	arn:aws:ssm:*
Semua Systems Manager sumber daya yang dimiliki oleh yang ditentukan Akun AWS dalam yang ditentukan Wilayah AWS	arn:aws:ssm:*region*:*account-id* :*

[1] Untuk definisi otomatisasi, Systems Manager mendukung sumber daya tingkat kedua, ID versi. Pada tahun AWS, sumber daya tingkat kedua ini dikenal sebagai subsumber daya. Menentukan versi subsumber daya untuk sumber definisi otomatisasi memungkinkan Anda untuk menyediakan akses ke versi

tertentu dari definisi otomatisasi. Misalnya, Anda mungkin ingin memastikan bahwa hanya versi terbaru dari definisi otomatisasi yang digunakan dalam manajemen node Anda.

2

Untuk mengatur dan mengelola parameter, Anda dapat membuat nama untuk parameter dengan konstruksi hierarkis. Dengan konstruksi hirarkis, nama parameter dapat mencakup jalur yang Anda tentukan dengan menggunakan garis miring ke depan. Anda dapat menyebutkan sumber daya parameter maksimum lima belas tingkat. Kami menyarankan agar Anda membuat hierarki yang mencerminkan struktur hierarkis yang ada di lingkungan Anda. Untuk informasi selengkapnya, lihat Menandai parameter Systems Manager.

Dalam sebagian besar kasus

3

, ID sesi dibangun menggunakan ID dari pengguna akun yang memulai sesi, ditambah akhiran alfanumerik. Sebagai contoh:

```
arn:aws:us-east-2:111122223333:session/JohnDoe-1a2b3c4sEXAMPLE
```

Namun, jika ID pengguna tidak tersedia, ARN dibangun dengan cara ini sebagai gantinya:

```
arn:aws:us-east-2:111122223333:session/session-1a2b3c4sEXAMPLE
```

Untuk informasi selengkapnya tentang format ARN, lihat Amazon Resource Name (ARN) di Referensi Umum Amazon Web Services.

Untuk daftar jenis Systems Manager sumber daya dan ARNnya, lihat Sumber Daya yang Ditentukan oleh AWS Systems Manager dalam Referensi Otorisasi Layanan. Untuk mempelajari tindakan mana yang dapat menentukan ARN setiap sumber daya, lihat Tindakan yang Ditentukan oleh AWS Systems Manager.

Kunci kondisi untuk Systems Manager

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, pengguna utama mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Elemen `Condition` (atau blok `Condition`) memungkinkan Anda menentukan kondisi di mana suatu pernyataan akan diterapkan. Elemen `Condition` bersifat opsional. Anda dapat membuat

ekspresi kondisional yang menggunakan operator kondisi, misalnya sama dengan atau kurang dari, untuk mencocokkan kondisi dalam kebijakan dengan nilai-nilai yang diminta.

Jika Anda menentukan beberapa elemen `Condition` dalam satu pernyataan, atau beberapa kunci dalam satu elemen `Condition`, AWS akan mengevaluasinya dengan menggunakan operasi `AND` logis. Jika Anda menentukan beberapa nilai untuk satu kunci kondisi, AWS mengevaluasi kondisi menggunakan `OR` operasi logis. Semua kondisi harus dipenuhi sebelum izin pernyataan diberikan.

Anda juga dapat menggunakan variabel placeholder saat menentukan kondisi. Misalnya, Anda dapat memberikan izin kepada pengguna IAM untuk mengakses sumber daya hanya jika izin tersebut mempunyai tag yang sesuai dengan nama pengguna IAM mereka. Untuk informasi selengkapnya, lihat Elemen kebijakan IAM: variabel dan tag di Panduan Pengguna IAM.

AWS mendukung kunci kondisi global dan kunci kondisi khusus layanan. Untuk melihat semua kunci kondisi AWS global, lihat kunci konteks kondisi AWS global di Panduan Pengguna IAM.

Untuk melihat daftar kunci syarat Systems Manager, lihat Kunci Syarat untuk AWS Systems Manager dalam Referensi Otorisasi Layanan. Untuk mempelajari tindakan dan sumber daya mana yang dapat Anda gunakan kunci syarat, lihat Tindakan yang Ditentukan oleh AWS Systems Manager.

Untuk informasi tentang penggunaan kunci kondisi `ssm:resourceTag/*`, lihat topik berikut:

- Membatasi akses ke perintah tingkat root melaluiSSM Agent
- MembatasiRun Command akses akses berdasarkan tag
- Batasi akses sesi berdasarkan tag instance

Untuk informasi lebih lanjut tentang menggunakan kunci kondisi `ssm:Recursive` dan `ssm:Overwrite`, lihat Bekerja dengan hierarki parameter.

Contoh-contoh

Untuk melihat contoh kebijakan berbasis identitas Systems Manager, lihat Contoh kebijakan berbasis identitas AWS Systems Manager.

Systems Manager Kebijakan berbasis sumber daya

Lainnya Layanan AWS, seperti Amazon Simple Storage Service (Amazon S3), mendukung kebijakan izin berbasis sumber daya. Misalnya, Anda dapat melampirkan kebijakan izin ke bucket S3 untuk mengelola izin akses ke bucket tersebut.

Systems Manager tidak mendukung kebijakan berbasis sumber daya.

Otorisasi berdasarkan tag Systems Manager

Anda dapat melampirkan tanda ke sumber daya Systems Manager atau meneruskan tanda dalam sebuah permintaan ke Systems Manager. Untuk mengontrol akses berdasarkan tag, Anda memberikan informasi tag di elemen kondisi kebijakan menggunakan kunci kondisi `ssm:resourceTag/key-name`, `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`, atau `aws:TagKeys`. Anda dapat menambahkan tag ke jenis sumber daya berikut saat Anda membuat atau memperbaruinya:

- Dokumen
- Node terkelola
- Jendela pemeliharaan
- Parameter
- Garis dasar patch
- OpsItem

Untuk informasi tentang menandai Systems Manager sumber daya, lihatPenandaan sumber daya Systems Manager.

Untuk melihat contoh kebijakan berbasis identitas untuk membatasi akses ke sumber daya berdasarkan tanda pada sumber daya tersebut, lihat MelihatSystems Manager dokumen berdasarkan tag.

Peran IAM Systems Manager

Peran IAM adalah entitas di dalam Anda Akun AWS yang memiliki izin khusus.

Menggunakan kredensial sementara dengan Systems Manager

Anda dapat menggunakan kredensial sementara untuk masuk dengan gabungan, menjalankan IAM role, atau menjalankan peran lintas akun. Anda memperoleh kredensyal keamanan sementara dengan memanggil AWS Security Token Service (AWS STS) operasi API seperti AssumeRoleatau. GetFederationToken

Systems Manager mendukung penggunaan kredensial sementara.

Peran terkait layanan

Peran terkait layanan memungkinkan Layanan AWS untuk mengakses sumber daya di layanan lain untuk menyelesaikan tindakan atas nama Anda. Peran terkait layanan tercantum dalam akun IAM Anda dan dimiliki oleh layanan tersebut. Administrator dapat melihat tetapi tidak dapat mengedit izin untuk peran yang terkait dengan layanan.

Systems Manager mendukung peran terkait layanan Untuk informasi selengkapnya tentang cara membuat atau mengelola peran terkait layanan Systems Manager, lihat Menggunakan peran terkait layanan untuk Systems Manager.

Peran layanan

Fitur ini memungkinkan layanan untuk menerima peran layanan atas nama Anda. Peran ini memungkinkan layanan mengakses sumber daya di layanan lain untuk menyelesaikan tindakan atas nama Anda. Peran layanan ditampilkan dalam akun IAM Anda dan dimiliki oleh akun tersebut. Ini berarti bahwa administrator dapat mengubah izin untuk peran ini. Namun, melakukannya mungkin merusak fungsi layanan.

Systems Manager mendukung peran layanan

Memilih IAM role dalam Systems Manager

Systems ManagerAgar dapat berinteraksi dengan node terkelola Anda, Anda harus memilih peran yang Systems Manager memungkinkan mengakses node atas nama Anda. Jika sebelumnya Anda telah membuat peran layanan atau peran terkait layanan, Systems Manager berikan daftar peran yang dapat dipilih. Penting untuk memilih peran yang memungkinkan akses untuk memulai dan menghentikan node yang dikelola.

Untuk mengakses instans EC2, Anda harus mengonfigurasi izin instans. Untuk selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager.

Untuk mengakses node non-EC2 dalam hybrid dan multicloud, peran yang Anda Akun AWS butuhkan adalah peran layanan IAM. Untuk informasi, lihat Membuat peran layanan IAM untuk lingkungan hibrid.

Alur kerja otomatisasi dapat dimulai di bawah konteks peran layanan (atau peran asumsi). Hal ini mengizinkan layanan untuk kemudian melakukan tindakan atas nama Anda. Jika Anda tidak menentukan peran asumsi, otomatisasi menggunakan konteks pengguna yang dipanggil eksekusi.

Namun, situasi tertentu mengharuskan Anda menentukan peran layanan untuk otomatisasi. Untuk informasi selengkapnya, lihat Mengonfigurasi akses peran layanan (peran asumsi) untuk otomatisasi.

Kebijakan terkelola AWS Systems Manager

AWS mengatasi banyak kasus penggunaan umum dengan menyediakan kebijakan IAM mandiri yang dibuat dan dikelola oleh. AWSKebijakan terkelola AWS ini memberikan izin yang diperlukan untuk kasus penggunaan umum sehingga Anda tidak perlu menyelidiki izin mana yang diperlukan. (Anda juga dapat membuat kebijakan IAM kustom Anda sendiri untuk mengizinkan izin Systems Manager tindakan dan sumber daya.)

Untuk informasi selengkapnya tentang kebijakan terkelola untuk Systems Manager, lihat AWS kebijakan terkelola untuk AWS Systems Manager

Untuk informasi umum tentang kebijakan terkelola, lihat kebijakan AWS terkelola di Panduan Pengguna IAM.

Contoh kebijakan berbasis identitas AWS Systems Manager

Secara default, entitasAWS Identity and Access Management (pengguna dan peran) tidak memiliki izin untuk membuat atau memodifikasiAWS Systems Manager sumber daya. Mereka juga tidak dapat melakukan tugas menggunakan konsol Systems Manager, AWS Command Line Interface (AWS CLI), atau AWS API. Administrator harus membuat kebijakan IAM yang memberikan izin kepada pengguna dan peran untuk melakukan operasi API tertentu pada sumber daya tertentu yang diperlukan. Administrator kemudian harus melampirkan kebijakan tersebut ke pengguna atau grup yang memerlukan izin tersebut.

Berikut ini adalah contoh kebijakan izin yang memungkinkan pengguna menghapus dokumen dengan nama yang dimulai**`MyDocument-`** di US East (Ohio) (us-east-2)Wilayah AWS.

```
{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "ssm:DeleteDocument"
      ],
      "Resource" : [
        "arn:aws:ssm:us-east-2:111122223333:document/MyDocument-*"
      ]
```

```
    }
  ]
}
```

Untuk mempelajari cara membuat kebijakan berbasis identitas IAM menggunakan contoh dokumen Kebijakan JSON ini, lihat Membuat kebijakan IAM dalam Panduan Pengguna IAM.

Topik

- Praktik terbaik kebijakan
- Menggunakan konsol Systems Manager
- Izinkan para pengguna untuk melihat izin mereka sendiri
- Pencegahan wakil bingung lintas layanan
- Contoh kebijakan yang dikelola pelanggan
- MelihatSystems Manager dokumen berdasarkan tag

Praktik terbaik kebijakan

Kebijakan berbasis identitas menentukan apakah seseorang dapat membuat, mengakses, atau menghapusSystems Manager sumber daya di akun Anda. Tindakan ini membuat Akun AWS Anda terkena biaya. Ketika Anda membuat atau mengedit kebijakan berbasis identitas, ikuti panduan dan rekomendasi ini:

- Memulai kebijakanAWS terkelola dan beralih ke izin paling sedikit hak istimewa — Untuk mulai memberikan izin kepada pengguna dan beban kerja Anda, gunakan kebijakanAWS terkelola yang memberikan izin untuk banyak kasus penggunaan umum. Mereka tersedia di AndaAkun AWS. Kami menyarankan Anda mengurangi izin lebih lanjut dengan menentukan kebijakan yang dikelolaAWS pelanggan yang spesifik untuk kasus penggunaan Anda. Untuk informasi selengkapnya, lihat kebijakanAWSAWS terkelola atau kebijakan terkelola untuk fungsi pekerjaan di Panduan Pengguna IAM.
- Terapkan izin paling rendah — Saat Anda mengatur izin dengan kebijakan IAM, hanya berikan izin yang diperlukan untuk melaksanakan tugas. Anda melakukan ini dengan menentukan tindakan yang dapat diambil pada sumber daya tertentu dalam kondisi tertentu, juga dikenal sebagai izin paling tidak memiliki hak istimewa. Untuk informasi selengkapnya tentang penggunaan IAM untuk menerapkan izin, lihat Kebijakan dan izin di IAM dalam Panduan Pengguna IAM.
- Gunakan ketentuan dalam kebijakan IAM untuk membatasi akses lebih lanjut — Anda dapat menambahkan kondisi pada kebijakan Anda untuk membatasi akses ke tindakan dan sumber

daya. Misalnya, Anda dapat menulis kondisi kebijakan untuk menentukan bahwa semua permintaan harus dikirim menggunakan SSL. Anda juga dapat menggunakan kondisi untuk memberikan akses ke tindakan layanan jika digunakan melalui spesifikLayanan AWS, sepertiAWS CloudFormation. Untuk informasi lebih lanjut, lihat Elemen Kebijakan IAM JSON: Syarat dalam Panduan Pengguna IAM.

- Gunakan IAM Access Analyzer untuk memvalidasi kebijakan IAM Anda untuk memastikan izin yang aman dan fungsional - IAM Access Analyzer memvalidasi kebijakan baru dan yang sudah ada sehingga kebijakan mematuhi bahasa kebijakan IAM (JSON) dan praktik terbaik IAM. IAM Access Analyzer menyediakan lebih dari 100 pemeriksaan kebijakan dan rekomendasi yang dapat ditindaklanjuti untuk membantu Anda membuat kebijakan yang aman dan fungsional. Untuk informasi selengkapnya, lihat validasi kebijakan IAM Access Analyzer di Panduan Pengguna IAM.
- Memerlukan otentikasi multi-faktor (MFA) — Jika Anda memiliki skenario yang mengharuskan pengguna IAM atau pengguna root di AndaAkun AWS, aktifkan MFA untuk keamanan tambahan. Untuk mewajibkan MFA saat operasi API dipanggil, tambahkan kondisi MFA ke kebijakan Anda. Untuk informasi selengkapnya, lihat Mengonfigurasi akses API yang dilindungi MFA di Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang praktik terbaik dalam IAM, lihat praktik terbaik keamanan di IAM dalam Panduan Pengguna IAM.

Menggunakan konsol Systems Manager

Untuk mengakses konsol Systems Manager, Anda harus memiliki rangkaian izin minimum. Izin ini harus memperbolehkan Anda untuk membuat daftar dan melihat detail tentangSystems Manager sumber daya dan sumber daya lainnya di AndaAkun AWS.

Untuk menggunakan sepenuhnyaSystems Manager diSystems Manager konsol, Anda harus memiliki izin dari layanan berikut:

- AWS Systems Manager
- Amazon Elastic Compute Cloud (Amazon EC2)
- AWS Identity and Access Management (IAM)

Anda dapat memberikan izin yang diperlukan dengan pernyataan kebijakan berikut.

```
{
    "Version": "2012-10-17",
```

```
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:*",
                "ec2:describeInstances",
                "iam:ListRoles"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "ssm.amazonaws.com"
                }
            }
        }
    ]
}
```

Jika Anda membuat kebijakan berbasis identitas yang lebih ketat dari izin minimum yang diperlukan, konsol tidak akan berfungsi sebagaimana dimaksudkan untuk entitas IAM (pengguna atau peran) dengan kebijakan tersebut.

Anda tidak perlu mengizinkan konsol minimum untuk pengguna yang melakukan panggilan hanya ke AWS CLI atau API AWS. Alih-alih, izinkan akses hanya ke tindakan yang sesuai dengan operasi API yang Anda coba lakukan.

Izinkan para pengguna untuk melihat izin mereka sendiri

Contoh ini menunjukkan cara Anda dapat membuat kebijakan yang mengizinkan para pengguna IAM untuk melihat kebijakan inline dan terkelola yang dilampirkan ke identitas pengguna mereka. Kebijakan ini mencakup izin untuk menyelesaikan tindakan ini pada konsol tersebut atau secara terprogram menggunakan AWS CLI atau AWS API.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
```

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

Pencegahan wakil bingung lintas layanan

Masalah wakil yang bingung adalah masalah keamanan di mana entitas yang tidak memiliki izin untuk melakukan tindakan dapat memaksa entitas yang lebih istimewa untuk melakukan tindakan. DalamAWS, peniruan lintas layanan dapat mengakibatkan masalah wakil bingung. Peniruan lintas layanan dapat terjadi ketika satu layanan (layanan panggilan) memanggil layanan lain (layanan yang disebut). Layanan panggilan dapat dimanipulasi untuk menggunakan izinnya untuk bertindak atas sumber daya pelanggan lain dengan cara yang seharusnya tidak memiliki izin untuk mengakses. Untuk mencegah hal ini,AWS sediakan alat yang membantu Anda melindungi data Anda untuk semua layanan dengan prinsipal layanan yang telah diberikan akses ke sumber daya di akun Anda.

Sebaiknya gunakan kunci konteks kondisi `aws:SourceAccount`global `aws:SourceArn`dan dalam kebijakan sumber daya untuk membatasi izin yangAWS Systems Manager memberikan layanan

lain ke sumber daya. Jikaaws:SourceArn nilai tidak berisi ID akun, seperti Amazon Resource Name (ARN) untuk bucket S3, Anda harus menggunakan kedua kunci konteks kondisi global untuk membatasi izin. Jika Anda menggunakan kunci konteks kondisi global danaws:SourceArn nilainya berisi ID akun,aws:SourceAccount nilai dan akun dalamaws:SourceArn nilai tersebut harus menggunakan ID akun yang sama saat digunakan dalam pernyataan kebijakan yang sama. Gunakanaws:SourceArn jika Anda hanya ingin satu sumber daya dikaitkan dengan akses lintas-layanan. Gunakanaws:SourceAccount jika Anda ingin mengizinkan sumber daya apa pun di akun tersebut dikaitkan dengan penggunaan lintas layanan.

Bagian berikut memberikan contoh kebijakan untukAWS Systems Manager kemampuan.

Contoh kebijakan aktivasi hibrida

Untuk peran layanan yang digunakan dalam aktivasi hybrid, nilaiaws:SourceArn harus ARN dariAkun AWS. Pastikan untuk menentukanWilayah AWS di ARN tempat Anda membuat aktivasi hybrid Anda. Jika Anda tidak mengetahui ARN lengkap sumber daya atau jika Anda menentukan beberapa sumber daya, gunakan kunci kondisi konteksaws:SourceArn global dengan wildcard (*) untuk bagian ARN yang tidak diketahui. Sebagai contoh, `arn:aws:ssm:*:region:123456789012:*`.

Contoh berikut menunjukkan penggunaan kunci konteks kondisiaws:SourceAccount global untuk Otomasi untuk mencegah masalah deputi yang membingungkan di Wilayah US East (Ohio) Region (Ohio) Region (us-east-2).aws:SourceArn

```
{
   "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:*"
            }
```

```
            }
        }
    ]
}
```

Contoh kebijakan sinkronisasi data

InventarisExplorer, dan Kepatuhan Systems Manager memungkinkan Anda membuat sinkronisasi data sumber daya untuk memusatkan penyimpanan data operasi (OpsData) di bucket Amazon Simple Storage Service pusat. Jika Anda ingin mengenkripsi sinkronisasi data sumber daya dengan menggunakanAWS Key Management Service (AWS KMS), maka Anda harus membuat kunci baru yang mencakup kebijakan berikut, atau Anda harus memperbarui kunci yang ada dan menambahkan kebijakan ini ke sana. Kunci`aws:SourceArn` dan`aws:SourceAccount` kondisi dalam kebijakan ini adalah masalah deputi yang membingungkan. Berikut ini adalah contoh kebijakan.

```
{
    "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**
>
> ARN dalam contoh kebijakan memungkinkan sistem untuk mengenkripsiOpsData dari semua sumber kecualiAWS Security Hub. Jika Anda perlu mengenkripsi data Security Hub, misalnya jika Anda menggunakanExplorer untuk mengumpulkan data Security Hub, maka Anda harus melampirkan kebijakan tambahan yang menentukan ARN berikut:
> `"aws:SourceArn": "arn:aws:ssm:*:account-id:role/aws-service-role/opsdatasync.ssm.amazonaws.com/AWSServiceRoleForSystemsManagerOpsDataSync"`

Contoh kebijakan yang dikelola pelanggan

Anda dapat membuat kebijakan mandiri yang Anda kelola di Akun AWS Anda. Kami menyebut ini sebagai Kebijakan terkelola pelanggan. Anda dapat melampirkan kebijakan ke beberapa entitas prinsipal di akun Akun AWS Anda. Saat Anda melampirkan kebijakan pada entitas prinsipal, Anda memberikan entitas sebuah izin yang ditentukan dalam kebijakan. Untuk informasi selengkapnya, lihat Contoh kebijakan terkelola pelanggan di Panduan Pengguna IAM.

Contoh kebijakan pengguna berikut memberikan izin untuk berbagai tindakan Systems Manager. Gunakan mereka untuk membatasiSystems Manager akses bagi entitas IAM (pengguna dan peran). Kebijakan ini berfungsi saat melakukan tindakan diSystems Manager API,AWS SDK, atauAWS CLI. Untuk pengguna yang menggunakan konsol, Anda perlu memberikan izin tambahan khusus untuk konsol. Untuk informasi selengkapnya, lihat Menggunakan konsol Systems Manager.

> **Note**
>
> Semua contoh menggunakan wilayah US West (Oregon) (us-west-2) dan berisi ID akun fiktif . ID akun tidak boleh ditentukan dalam Amazon Resource Name (ARN) untuk dokumen publik AWS (dokumen yang dimulai dengan `AWS-*`).

Contoh

- Contoh 1: Mengizinkan pengguna untuk melakukanSystems Manager operasi dalam satu Region
- Contoh 2: Memungkinkan pengguna untuk mendaftarkan dokumen untuk satu Wilayah

Contoh 1: Mengizinkan pengguna untuk melakukanSystems Manager operasi dalam satu Region

Contoh berikut memberikan izin untuk melakukanSystems Manager operasi hanya di Region US East (Ohio) (us-east-2).

```
{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "ssm:*"
      ],
      "Resource" : [
        "arn:aws:ssm:us-east-2:aws-account-ID:*"
      ]
    }
  ]
}
```

Contoh 2: Memungkinkan pengguna untuk mendaftarkan dokumen untuk satu Wilayah

Contoh berikut memberikan izin untuk mencantumkan semua nama dokumen yang dimulai dengan**`Update`** Region US East (Ohio) Region US East (Ohio).

```
{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "ssm:ListDocuments"
      ],
      "Resource" : [
        "arn:aws:ssm:us-east-2:aws-account-ID:document/Update*"
      ]
    }
  ]
}
```

Contoh 3: Mengizinkan pengguna untuk menggunakan dokumen SSM untuk menjalankan perintah pada node tertentu

Contoh kebijakan IAM mengizinkan pengguna untuk melakukan hal berikut di Region US East (Ohio) (us-east-2):

- DaftarSystems Manager dokumen (dokumen SSM) dan versi dokumen
- Lihat detail tentang dokumen.
- Kirim perintah menggunakan dokumen yang telah ditentukan dalam kebijakan. Nama dokumen ditentukan oleh entri berikut.

```
arn:aws:ssm:us-east-2:aws-account-ID:document/Systems-Manager-document-name
```

- Kirim perintah ke tiga node. Node ditentukan oleh entri berikut di`Resource` bagian kedua.

```
"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"
```

- Melihat rincian tentang perintah setelah dikirim.
- Memulai dan menghentikan alur kerja di otomatisasi, suatu kemampuan AWS Systems Manager.
- Dapatkan informasi tentang alur kerja Otomatisasi.

Jika Anda ingin memberikan izin pengguna untuk menggunakan dokumen ini untuk mengirim perintah pada setiap node pengguna yang memiliki akses, Anda dapat menentukan entri yang mirip dengan berikut di`Resource` bagian dan menghapus entri simpul lainnya. Contoh berikut menggunakan Region US East (Ohio) (us-east-2).

```
"arn:aws:ec2:us-east-2:*:instance/*"
```

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "ssm:ListDocuments",
                "ssm:ListDocumentVersions",
                "ssm:DescribeDocument",
                "ssm:GetDocument",
```

```
                "ssm:DescribeInstanceInformation",
                "ssm:DescribeDocumentParameters",
                "ssm:DescribeInstanceProperties"
            ],
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Action": "ssm:SendCommand",
            "Effect": "Allow",
            "Resource": [
                "arn:aws:ec2:us-east-2:aws-account-ID:instance/i-02573cafcfEXAMPLE",
                "arn:aws:ec2:us-east-2:aws-account-ID:instance/i-0471e04240EXAMPLE",
                "arn:aws:ec2:us-east-2:aws-account-ID:instance/i-07782c72faEXAMPLE",

                "arn:aws:ssm:us-east-2:aws-account-ID:document/Systems-Manager-
document-name"
            ]
        },
        {
            "Action": [
                "ssm:CancelCommand",
                "ssm:ListCommands",
                "ssm:ListCommandInvocations"
            ],
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Action": "ec2:DescribeInstanceStatus",
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Action": "ssm:StartAutomationExecution",
            "Effect": "Allow",
            "Resource": [
                "arn:aws:ssm:us-east-2:aws-account-ID:automation-definition/*"
            ]
        },
        {
            "Action": "ssm:DescribeAutomationExecutions",
            "Effect": "Allow",
            "Resource": [
```

```
                "*"
            ]
        },
        {
            "Action": [
                "ssm:StopAutomationExecution",
                "ssm:GetAutomationExecution"
            ],
            "Effect": "Allow",
            "Resource": [
                "*"
            ]
        }
    ]
}
```

MelihatSystems Manager dokumen berdasarkan tag

Anda dapat menggunakan syarat dalam kebijakan berbasis identitas Anda untuk mengontrol akses ke sumber daya Systems Manager berdasarkan tanda. Contoh ini menunjukkan cara bagaimana Anda dapat membuat kebijakan yang memperbolehkan melihat dokumen SSM. Namun, izin diberikan hanya jika tag dokumen `Owner` memiliki nilai nama pengguna dari pengguna tersebut. Kebijakan ini juga memberi izin yang diperlukan untuk menyelesaikan tindakan ini pada konsol tersebut.

```
{
    "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}"}
            }
        }
    ]
```

```
}
```

Anda dapat melampirkan kebijakan ini ke pengguna di akun Anda. Jika pengguna bernama`richard-roe` mencoba untuk melihatSystems Manager dokumen, dokumen harus ditandai`0wner=richard-roe` atau`owner=richard-roe`. Jika tidak, mereka ditolak aksesnya. Kunci tag kondisi `0wner` cocok dengan `0wner` dan `owner` karena nama kunci kondisi tidak terpengaruh huruf besar/kecil. Untuk informasi selengkapnya, lihat Elemen kebijakan JSON IAM: Syarat dalam Panduan Pengguna IAM.

AWS kebijakan terkelola untuk AWS Systems Manager

Kebijakan AWS terkelola adalah kebijakan mandiri yang dibuat dan dikelola oleh AWS. AWS Kebijakan terkelola dirancang untuk memberikan izin bagi banyak kasus penggunaan umum sehingga Anda dapat mulai menetapkan izin kepada pengguna, grup, dan peran.

Perlu diingat bahwa kebijakan AWS terkelola mungkin tidak memberikan izin hak istimewa paling sedikit untuk kasus penggunaan spesifik Anda karena tersedia untuk digunakan semua pelanggan. AWS Kami menyarankan Anda untuk mengurangi izin lebih lanjut dengan menentukan kebijakan yang dikelola pelanggan yang khusus untuk kasus penggunaan Anda.

Anda tidak dapat mengubah izin yang ditentukan dalam kebijakan AWS terkelola. Jika AWS memperbarui izin yang ditentukan dalam kebijakan AWS terkelola, pembaruan akan memengaruhi semua identitas utama (pengguna, grup, dan peran) yang dilampirkan kebijakan tersebut. AWS kemungkinan besar akan memperbarui kebijakan AWS terkelola saat baru Layanan AWS diluncurkan atau operasi API baru tersedia untuk layanan yang ada.

Untuk informasi selengkapnya, lihat Kebijakan yang dikelola AWS dalam Panduan Pengguna IAM.

AWS kebijakan terkelola: AmazonSSM ServiceRolePolicy

Anda tidak dapat melampirkan `AmazonSSMServiceRolePolicy` ke entitas AWS Identity and Access Management (IAM) Anda. Kebijakan ini dilampirkan pada peran terkait layanan yang memungkinkan AWS Systems Manager untuk melakukan tindakan atas nama Anda. Untuk informasi selengkapnya, lihat Menggunakan peran untuk mengumpulkan inventaris dan melihat OpsData.

`AmazonSSMServiceRolePolicy`memungkinkan Systems Manager untuk menyelesaikan tindakan berikut pada semua sumber daya terkait (`"Resource": "*"`), kecuali jika ditunjukkan:

- ssm:CancelCommand
- ssm:GetCommandInvocation
- ssm:ListCommandInvocations
- ssm:ListCommands
- ssm:SendCommand
- ssm:GetAutomationExecution
- ssm:GetParameters
- ssm:StartAutomationExecution
- ssm:StopAutomationExecution
- ssm:ListTagsForResource
- ssm:GetCalendarState
- ssm:UpdateServiceSetting [1]
- ssm:GetServiceSetting [1]
- ec2:DescribeInstanceAttribute
- ec2:DescribeInstanceStatus
- ec2:DescribeInstances
- lambda:InvokeFunction [2]
- states:DescribeExecution [3]
- states:StartExecution [3]
- resource-groups:ListGroups
- resource-groups:ListGroupResources
- resource-groups:GetGroupQuery
- tag:GetResources
- config:SelectResourceConfig
- config:DescribeComplianceByConfigRule
- config:DescribeComplianceByResource
- config:DescribeRemediationConfigurations
- config:DescribeConfigurationRecorders
- cloudwatch:DescribeAlarms
- compute-optimizer:GetEC2InstanceRecommendations

- compute-optimizer:GetEnrollmentStatus
- support:DescribeTrustedAdvisorChecks
- support:DescribeTrustedAdvisorCheckSummaries
- support:DescribeTrustedAdvisorCheckResult
- support:DescribeCases
- iam:PassRole [4]
- cloudformation:DescribeStacks
- cloudformation:ListStackResources
- cloudformation:ListStackInstances [5]
- cloudformation:DescribeStackSetOperation [5]
- cloudformation:DeleteStackSet [5]
- cloudformation:DeleteStackInstances [6]
- events:PutRule [7]
- events:PutTargets [7]
- events:RemoveTargets [8]
- events:DeleteRule [8]
- events:DescribeRule
- securityhub:DescribeHub

[1] Tindakan ssm:UpdateServiceSetting dan ssm:GetServiceSetting yang diperbolehkan izin untuk sumber daya berikut ini saja.

```
arn:aws:ssm:*:*:servicesetting/ssm/opsitem/*
arn:aws:ssm:*:*:servicesetting/ssm/opsdata/*
```

[2] Tindakan lambda:InvokeFunction yang diperbolehkan izin untuk sumber daya berikut ini saja.

```
arn:aws:lambda:*:*:function:SSM*
arn:aws:lambda:*:*:function:*:SSM*
```

[3] Tindakan states: yang diperbolehkan izin pada sumber daya berikut saja.

```
arn:aws:states:*:*:stateMachine:SSM*
```

```
arn:aws:states:*:*:execution:SSM*
```

[4] `iam:PassRole` Tindakan ini diizinkan izin dengan kondisi berikut hanya untuk Systems Manager layanan.

```
"Condition": {
   "StringEquals": {
      "iam:PassedToService": [
         "ssm.amazonaws.com"
      ]
   }
}
```

[5] Tindakan `cloudformation:ListStackInstances`, `cloudformation:DescribeStackSetOperation`, dan `cloudformation:DeleteStackSet` yang diperbolehkan izin pada sumber daya berikut saja.

```
arn:aws:cloudformation:*:*:stackset/AWS-QuickSetup-SSM*:*
```

[6] Tindakan `cloudformation:DeleteStackInstances` yang diperbolehkan izin pada sumber daya berikut ini saja.

```
arn:aws:cloudformation:*:*:stackset/AWS-QuickSetup-SSM*:*
arn:aws:cloudformation:*:*:stackset-target/AWS-QuickSetup-SSM*:*
arn:aws:cloudformation:*:*:type/resource/*
```

[7] `events:PutTargets` Tindakan `events:PutRule` dan diizinkan izin dengan kondisi berikut untuk Systems Manager layanan saja.

```
"Condition": {
    "StringEquals": {
        "events:ManagedBy": "ssm.amazonaws.com"
    }
}
```

[8] Tindakan `events:RemoveTargets` dan `events:DeleteRule` yang diperbolehkan izin pada sumber daya berikut saja.

```
arn:aws:events:*:*:rule/SSMExplorerManagedRule
```

Untuk melihat detail selengkapnya tentang kebijakan, termasuk versi terbaru dari dokumen kebijakan JSON, lihat AmazonSSM ServiceRolePolicy di Panduan Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: AmazonSSM ReadOnlyAccess

Anda dapat melampirkan kebijakan `AmazonSSMReadOnlyAccess` ke identitas IAM Anda. Kebijakan ini memberikan akses hanya-baca ke operasi AWS Systems Manager API termasuk`Describe*`,, `Get*` dan. `List*`

Untuk melihat detail selengkapnya tentang kebijakan, termasuk versi terbaru dari dokumen kebijakan JSON, lihat AmazonSSM ReadOnlyAccess di Panduan Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: AWSSystemsManagerOpsDataSyncServiceRolePolicy

Anda tidak dapat melampirkan `AWSSystemsManagerOpsDataSyncServiceRolePolicy` ke entitas IAM Anda. Kebijakan ini dilampirkan ke peran terkait layanan yang mengizinkan Systems Manager untuk melakukan tindakan atas nama Anda. Untuk informasi selengkapnya, lihat Menggunakan peran untuk membuat OpsData danOpsItemsuntukExplorer.

`AWSSystemsManagerOpsDataSyncServiceRolePolicy`memungkinkan peran `AWSServiceRoleForSystemsManagerOpsDataSync` terkait layanan untuk membuat dan memperbarui OpsItems dan OpsData dari AWS Security Hub temuan.

Kebijakan ini memungkinkan Systems Manager untuk menyelesaikan tindakan berikut pada semua sumber daya terkait (`"Resource": "*"`), kecuali jika ditunjukkan:

- `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] `ssm:UpdateOpsItem` Tindakan `ssm:GetOpsItem` dan diizinkan izin dengan kondisi berikut untuk Systems Manager layanan saja.

```
"Condition": {
    "StringEquals": {
        "aws:ResourceTag/ExplorerSecurityHubOpsItem": "true"
    }
}
```

[2] Tindakan `ssm:AddTagsToResource` yang diperbolehkan izin untuk sumber daya berikut ini saja.

```
arn:aws:ssm:*:*:opsitem/*
```

[3] Tindakan `ssm:UpdateServiceSetting` dan `ssm:GetServiceSetting` yang diperbolehkan izin untuk sumber daya berikut ini saja.

```
arn:aws:ssm:*:*:servicesetting/ssm/opsitem/*
arn:aws:ssm:*:*:servicesetting/ssm/opsdata/*
```

[4] `securityhub:BatchUpdateFindings` Izin ditolak oleh kondisi berikut hanya untuk Systems Manager layanan.

```
{
   "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
     }
    }
```

Untuk melihat detail selengkapnya tentang kebijakan, termasuk versi terbaru dari dokumen kebijakan JSON, lihat AWSSystemsManagerOpsDataSyncServiceRolePolicydi Panduan Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: AmazonsSMManageDEC2 InstanceDefaultPolicy

Anda hanya boleh melampirkan `AmazonSSMManagedEC2InstanceDefaultPolicy` ke peran IAM untuk instans Amazon EC2 yang Anda ingin memiliki izin untuk menggunakan fungsionalitas. Systems Manager Anda tidak boleh melampirkan peran ini ke entitas IAM lainnya, seperti pengguna

IAM dan grup IAM, atau ke peran IAM yang melayani tujuan lain. Untuk informasi selengkapnya, lihat Menggunakan pengaturan Konfigurasi Manajemen Host Default.

Kebijakan ini memberikan izin yang memungkinkan instans Amazon EC2 untuk mengambil Dokumen, menjalankan perintah Run Command menggunakan, membuat sesi Session Manager menggunakan, mengumpulkan inventaris instans, dan memindai patch dan kepatuhan patch menggunakan. SSM Agent Patch Manager

Systems Manager menggunakan token otorisasi yang dipersonalisasi untuk setiap instance untuk memastikan bahwa SSM Agent menjalankan operasi API pada instance yang benar. Systems Manager memvalidasi token otorisasi yang dipersonalisasi terhadap Amazon Resource Name (ARN) instance, yang disediakan dalam operasi API.

Kebijakan izin `AmazonSSMManagedEC2InstanceDefaultPolicy` peran memungkinkan Systems Manager untuk menyelesaikan tindakan berikut pada semua sumber daya terkait:

- `ssm:DescribeAssociation`
- `ssm:GetDeployablePatchSnapshotForInstance`
- `ssm:GetDocument`
- `ssm:DescribeDocument`
- `ssm:GetManifest`
- `ssm:ListAssociations`
- `ssm:ListInstanceAssociations`
- `ssm:PutInventory`
- `ssm:PutComplianceItems`
- `ssm:PutConfigurePackageResult`
- `ssm:UpdateAssociationStatus`
- `ssm:UpdateInstanceAssociationStatus`
- `ssm:UpdateInstanceInformation`
- `ssmmessages:CreateControlChannel`
- `ssmmessages:CreateDataChannel`
- `ssmmessages:OpenControlChannel`
- `ssmmessages:OpenDataChannel`
- `ec2messages:AcknowledgeMessage`

- `ec2messages:DeleteMessage`
- `ec2messages:FailMessage`
- `ec2messages:GetEndpoint`
- `ec2messages:GetMessages`
- `ec2messages:SendReply`

Untuk melihat detail selengkapnya tentang kebijakan, termasuk versi terbaru dari dokumen kebijakan JSON, lihat AmazonSSMManagedEC2InstanceDefaultPolicy di Panduan Referensi Kebijakan Terkelola.AWS

Systems Managerpembaruan kebijakan AWS terkelola

Di tabel berikut, lihat detail tentang pembaruan kebijakan AWS terkelola Systems Manager sejak layanan ini mulai melacak perubahan ini pada 12 Maret 2021. Untuk informasi tentang kebijakan terkelola lainnya untuk layanan Systems Manager, lihat Kebijakan terkelola tambahan untuk Systems Manager nanti dalam topik ini. Untuk peringatan otomatis tentang perubahan pada halaman ini, berlangganan RSS feed pada halaman Systems Manager Riwayat dokumen.

Perubahan	Deskripsi	Tanggal
`AWSSystemsManagerOpsDataSyncServiceRolePolicy` — Perbarui ke kebijakan yang ada.	OpsCentermemperbarui kebijakan untuk meningkatkan keamanan kode layanan dalam peran terkait layanan untuk mengelola OpsData operasi Explorer terkait.	28 Juni 2023
`AmazonSSMManagedEC2InstanceDefaultPolicy` — Kebijakan baru.	Systems Managerme nambahkan kebijakan baru untuk mengizinkan Systems Manager fungsionalitas pada instans Amazon EC2 tanpa menggunakan profil instans IAM.	18 Agustus 2022

Perubahan	Deskripsi	Tanggal
AmazonSSM ServiceRolePolicy - Perbarui ke kebijakan yang ada.	Systems Managerme nambahkan izin baru Explorer untuk memungkinkan membuat aturan terkelola saat Anda mengaktifkan Security Hub dari Explorer atauOpsCenter. Izin baru ditambahkan untuk memeriksa konfigurasi dan pengoptimal komputasi memenuhi persyaratan yang diperlukan sebelum mengizinkan. OpsData	27 April 2021
AWSSystemsManagerOpsDataSyncServiceRolePolicy — Kebijakan baru.	Systems Managerme nambahkan kebijakan baru untuk membuat dan memperbarui OpsItems dan OpsData dari temuan Security Hub di Explorer danOpsCenter.	27 April 2021
AmazonSSMServiceRolePolicy — Perbarui ke kebijakan yang ada.	Systems Managerme nambahkan izin baru untuk memungkinkan melihat agregat OpsData dan OpsItems detail dari beberapa akun dan Wilayah AWS masuk. Explorer	24 Maret 2021
Systems Manager mulai melacak perubahan	Systems Managermulai melacak perubahan untuk kebijakan yang AWS dikelola.	12 Maret 2021

Kebijakan terkelola tambahan untuk Systems Manager

Selain kebijakan terkelola yang dijelaskan sebelumnya dalam topik ini, kebijakan berikut juga didukung oleh Systems Manager.

- `AmazonSSMAutomationApproverAccess`— kebijakan AWS terkelola yang memungkinkan akses untuk melihat eksekusi otomatisasi dan mengirim keputusan persetujuan ke otomatisasi yang menunggu persetujuan.
- `AmazonSSMAutomationRole`— kebijakan AWS terkelola yang memberikan izin bagi layanan Systems Manager Otomasi untuk menjalankan aktivitas yang ditentukan dalam runbook Otomasi. Menetapkan kebijakan ini untuk administrator dan pengguna daya terpercaya.
- `AmazonSSMDirectoryServiceAccess`— kebijakan AWS terkelola yang memungkinkan SSM Agent untuk mengakses AWS Directory Service atas nama pengguna untuk permintaan bergabung dengan domain oleh node terkelola.
- `AmazonSSMFullAccess`— kebijakan AWS terkelola yang memberikan akses penuh ke Systems Manager API dan dokumen.
- `AmazonSSMMaintenanceWindowRole`— kebijakan AWS terkelola yang menyediakan jendela pemeliharaan dengan izin ke Systems Manager API.
- `AmazonSSMManagedInstanceCore`— kebijakan AWS terkelola yang memungkinkan node menggunakan fungsionalitas inti Systems Manager layanan.
- `AmazonSSMPatchAssociation`— kebijakan AWS terkelola yang menyediakan akses ke instance turunan untuk operasi asosiasi tambalan.
- `AmazonSSMReadOnlyAccess`— kebijakan AWS terkelola yang memberikan akses ke operasi API Systems Manager hanya-baca, seperti dan. `Get*` `List*`
- `AWSSSMOpsInsightsServiceRolePolicy`— kebijakan AWS terkelola yang memberikan izin untuk membuat dan memperbarui wawasan operasional OpsItemsdiSystems Manager. Digunakan untuk memberikan izin melalui peran terkait layanan. `AWSServiceRoleForAmazonSSM_OpsInsights`
- `AWSSystemsManagerAccountDiscoveryServicePolicy`— kebijakan AWS terkelola yang memberikan izin kepada Systems Manager untuk menemukan Akun AWS informasi.
- `AWSSystemsManagerChangeManagementServicePolicy`Kebijakan AWS terkelola yang menyediakan akses ke AWS sumber daya yang dikelola atau digunakan oleh kerangka manajemen Systems Manager perubahan dan digunakan oleh peran terkait layanan. `AWSServiceRoleForSystemsManagerChangeManagement`

- `AmazonEC2RoleforSSM`Kebijakan ini tidak lagi didukung dan tidak boleh digunakan. Sebagai gantinya, gunakan `AmazonSSMManagedInstanceCore` kebijakan untuk mengizinkan fungsionalitas inti Systems Manager layanan pada instans EC2. Untuk selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager.

Pemecahan masalah identitas dan akses AWS Systems Manager

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temukan saat bekerja dengan AWS Systems Manager dan AWS Identity and Access Management (IAM).

Topik

- Saya tidak diotorisasi untuk melakukan tindakan di Systems Manager
- Saya tidak diotorisasi untuk melakukan iamam.PassRole
- Saya ingin mengizinkan orang di luar Akun AWS saya untuk mengakses sumber daya Systems Manager saya

Saya tidak diotorisasi untuk melakukan tindakan di Systems Manager

Jika AWS Management Console memberi tahu Anda bahwa Anda tidak memiliki izin untuk melakukan tindakan, Anda harus menghubungi administrator untuk mendapatkan bantuan. Administrator Anda adalah orang yang memberikan Anda kredenisasi masuk.

Contoh kesalahan terjadi ketika`mateojackson` pengguna mencoba menggunakan konsol untuk melihat detail tentang dokumen tetapi tidak memiliki`ssm:GetDocument` izin.

```
User: arn:aws:ssm::123456789012:user/mateojackson isn't authorized to perform:
 ssm:GetDocument on resource: MyExampleDocument
```

Dalam hal ini, Mateo meminta administratornya untuk memperbarui kebijakannya untuk memungkinkannya mengakses sumber daya `MyExampleDocument` dengan menggunakan tindakan `ssm:GetDocument`.

Saya tidak diotorisasi untuk melakukan iamam.PassRole

Jika Anda menerima kesalahan bahwa Anda tidak terotorisasi untuk melakukan tindakan tersebut, maka Anda menerima kesalahan bahwa Anda tidak terotorisasi untuk melakukan`iam:PassRole`

tindakan tersebut maka Anda menerima kesalahan bahwa Anda tidak terotorisasi untuk melakukan tindakan tersebut, maka Anda menerima kesalahan bahwa Anda tidak terotorisasi untuk melakukan tindakan tersebutSystems Manager.

BeberapaLayanan AWS memungkinkan Anda memberikan peran yang sudah ada ke layanan tersebut alih-alih membuat peran layanan atau peran tertaut-layanan. Untuk melakukannya, Anda harus memiliki izin untuk meneruskan peran ke layanan.

Contoh kesalahan berikut terjadi ketika pengguna IAM bernama `marymajor` mencoba menggunakan konsol tersebut untuk melakukan tindakan di Systems Manager. Namun, tindakan tersebut mengharuskan layanan untuk memiliki izin yang diberikan oleh peran layanan. Mary tidak memiliki izin untuk meneruskan peran tersebut ke layanan.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

Dalam kasus ini, kebijakan Mary harus diperbarui untuk memungkinkannya melakukan`iam:PassRole` tindakan.

Jika Anda memerlukan bantuan, hubungiAWS administrator Anda. Administrator Anda adalah orang yang memberikan Anda kredenisasi masuk.

Saya ingin mengizinkan orang di luar Akun AWS saya untuk mengakses sumber daya Systems Manager saya

Anda dapat membuat peran yang dapat digunakan para pengguna di akun lain atau orang-orang di luar organisasi Anda untuk mengakses sumber daya Anda. Anda dapat menentukan siapa yang dipercaya untuk mengambil peran tersebut. Untuk layanan yang mendukung kebijakan berbasis sumber daya atau daftar kontrol akses (ACL), Anda dapat menggunakan kebijakan tersebut untuk memberi akses pada orang ke sumber daya Anda.

Untuk mempelajari selengkapnya, lihat hal berikut:

- Untuk mempelajari apakah Systems Manager mendukung fitur-fitur ini, lihat Cara kerja AWS Systems Manager dengan IAM.
- Untuk mempelajari cara memberikan akses ke sumber daya di seluruh akun Akun AWS yang Anda miliki, lihat Memberikan akses ke pengguna IAM di akun Akun AWS lain yang Anda miliki dalam Panduan Pengguna IAM.

- Untuk mempelajari cara memberikan akses ke sumber daya Anda ke Akun AWS pihak ketiga, lihat Menyediakan akses ke akun Akun AWS yang dimiliki oleh pihak ketiga dalam Panduan Pengguna IAM.
- Untuk mempelajari cara memberikan akses melalui federasi identitas, lihat Menyediakan akses ke pengguna terautentikasi eksternal (gabungan identitas) dalam Panduan Pengguna IAM .
- Untuk mempelajari perbedaan antara penggunaan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat Perbedaan IAM role dan kebijakan berbasis sumber daya di Panduan Pengguna IAM.

Menggunakan peran terkait layanan untuk Systems Manager

AWS Systems Managermenggunakan AWS Identity and Access Management peran terkait layanan (IAM). Peran yang terkait dengan layanan adalah tipe IAM role unik yang terkait langsung ke layanan. Peran terkait layanan telah ditentukan sebelumnya oleh Systems Manager dan menyertakan semua izin yang diperlukan layanan untuk memanggil orang lain Layanan AWS atas nama Anda.

> **Note**
>
> Peran peran layanan berbeda dari peran terkait layanan. Peran layanan adalah jenis peran AWS Identity and Access Management (IAM) yang memberikan izin kepada layanan Layanan AWS sehingga layanan dapat mengakses sumber daya. AWS Hanya beberapa skenario Systems Manager yang memerlukan peran layanan. Saat Anda membuat peran layanan untuk Systems Manager, Anda memilih izin yang akan diberikan agar dapat mengakses atau berinteraksi dengan AWS sumber daya lain.

Peran terkait layanan memudahkan penyiapan Systems Manager karena Anda tidak perlu menambahkan izin yang diperlukan secara manual. Systems Manager Peran ini menentukan izin peran terkait layanannya, dan kecuali ditentukan lain, hanya Systems Manager dapat mengambil perannya. Izin yang ditentukan mencakup kebijakan kepercayaan dan kebijakan izin, dan kebijakan izin tersebut tidak dapat dilampirkan ke entitas IAM lainnya.

Anda dapat menghapus peran terkait layanan hanya setelah menghapus sumber daya yang terkait terlebih dahulu. Ini melindungi sumber daya Systems Manager karena Anda tidak dapat secara tidak sengaja menghapus izin untuk mengakses sumber daya.

> **Note**
>
> Untuk node non-EC2 di lingkungan hybrid dan multicloud, Anda memerlukan peran IAM tambahan yang memungkinkan mesin tersebut berkomunikasi dengan layanan. Systems Manager Ini adalah peran layanan IAM untukSystems Manager. Peran ini memberikan AWS Security Token Service (AWS STS) AssumeRolekepercayaan pada Systems Manager layanan. Tindakan `AssumeRole` mengembalikan kredensial keamanan sementara (yang terdiri dari token keamanan, access key ID, dan secret access key). Anda menggunakan kredensi sementara ini untuk mengakses AWS sumber daya yang biasanya tidak dapat Anda akses. Untuk informasi selengkapnya, lihat Membuat peran layanan IAM untuk lingkungan hybrid dan AssumeRoledi Referensi AWS Security Token Service API.

Untuk informasi tentang layanan lain yang mendukung peran terkait layanan, lihat layanan Layanan AWS yang berfungsi dengan IAM dan cari layanan yang memiliki Ya di kolom Peran terkait layanan. Pilih Ya dengan sebuah tautan untuk melihat dokumentasi peran terkait layanan untuk layanan tersebut.

Topik

- Menggunakan peran untuk mengumpulkan inventaris dan melihat OpsData
- Menggunakan peran untuk mengumpulkan Akun AWS informasi untuk OpsCenter dan Explorer
- Menggunakan peran untuk membuat OpsData danOpsItemsuntukExplorer
- Menggunakan peran untuk menciptakan wawasan operasional OpsItems di Manajer SistemOpsCenter

Menggunakan peran untuk mengumpulkan inventaris dan melihat OpsData

Systems Managermenggunakan peran terkait layanan bernama**`AWSServiceRoleForAmazonSSM`**—AWS Systems Managermenggunakan peran layanan IAM ini untuk mengelolaAWSsumber daya atas nama Anda.

Izin peran terkait layanan untuk inventaris, OpsData, dan OpsItems

Peran terkait layanan `AWSServiceRoleForAmazonSSM` hanya mempercayai layanan `ssm.amazonaws.com` untuk menjalankan peran.

Anda dapat menggunakan peran terkait layanan Manajer SistemAWSServiceRoleForAmazonSSMuntuk yang berikut ini:

- Kemampuan Inventaris Manajer Sistem menggunakan peran terkait layananAWSServiceRoleForAmazonSSMuntuk mengumpulkan metadata inventaris dari tag dan grup sumber daya.
- TheExplorerkapabilitas menggunakan peran terkait layananAWSServiceRoleForAmazonSSMuntuk mengaktifkan tampilan OpsData danOpsItemsdari beberapa akun. Peran terkait layanan ini juga memungkinkanExploreruntuk membuat aturan terkelola saat Anda mengaktifkan Security Hub sebagai sumber data dariExploreratauOpsCenter.

> **Important**
>
> Sebelumnya, konsol Manajer Sistem memberi Anda kemampuan untuk memilihAWSperan terkait layanan IAM yang dikelolaAWSServiceRoleForAmazonSSMuntuk digunakan sebagai peran pemeliharaan untuk tugas-tugas Anda. Menggunakan peran ini dan kebijakan terkaitnya,AmazonSSMServiceRolePolicy, untuk tugas jendela pemeliharaan tidak lagi direkomendasikan. Jika Anda menggunakan peran ini untuk tugas jendela pemeliharaan sekarang, kami mendorong Anda untuk berhenti menggunakannya. Sebagai gantinya, buat peran IAM Anda sendiri yang memungkinkan komunikasi antara Manajer Sistem dan lainnyaLayanan AWSsaat tugas jendela pemeliharaan Anda berjalan.
> Untuk informasi selengkapnya, lihat Menyiapkan Maintenance Windows.

Kebijakan terkelola yang digunakan untuk memberikan izin untuk peran AWSServiceRoleForAmazonSSM adalah AmazonSSMServiceRolePolicy. Untuk detail tentang izin yang diberikannya, lihat AWS kebijakan terkelola: AmazonSSM ServiceRolePolicy.

Menciptakan**AWSServiceRoleForAmazonSSM**peran terkait layanan untukSystems Manager

Anda juga dapat menggunakan konsol IAM untuk membuat peran terkait layanan dengan menggunakankasus EC2. Menggunakan perintah untuk IAM di AWS Command Line Interface (AWS CLI) atau menggunakan API IAM, membuat peran terkait layanan dengan nama layanan ssm.amazonaws.com. Untuk informasi selengkapnya, lihat Membuat peran tertaut layanan dalam Panduan Pengguna IAM.

Jika Anda menghapus peran tertaut layanan ini, dan ingin membuatnya lagi, Anda dapat mengulangi proses yang sama untuk membuat kembali peran tersebut di akun Anda.

Mengedit**AWSServiceRoleForAmazonSSM**peran terkait layanan untukSystems Manager

Systems Managertidak memungkinkan Anda untuk mengedit`AWSServiceRoleForAmazonSSM`peran terkait layanan. Setelah membuat peran terkait layanan, Anda tidak dapat mengubah nama peran karena berbagai entitas mungkin mereferensikan peran tersebut. Namun, Anda dapat menyunting penjelasan peran menggunakan IAM. Untuk informasi selengkapnya, lihat Mengedit peran yang terkait dengan layanan dalam Panduan Pengguna IAM.

Menghapus**AWSServiceRoleForAmazonSSM**peran terkait layanan untukSystems Manager

Jika Anda tidak perlu lagi menggunakan fitur atau layanan yang memerlukan peran terkait layanan, kami menyarankan Anda menghapus peran tersebut. Dengan begitu Anda tidak memiliki entitas yang tidak digunakan yang tidak dipantau atau dipelihara secara aktif. Anda dapat menggunakan konsol IAM, AWS CLI, atau IAM API untuk menghapus peran terkait layanan secara manual. Untuk melakukan ini, Anda harus membersihkan sumber daya peran terkait layanan sebelum menghapusnya secara manual.

Karena`AWSServiceRoleForAmazonSSM`peran terkait layanan dapat digunakan oleh beberapa kemampuan, memastikan bahwa tidak ada yang menggunakan peran sebelum mencoba menghapusnya.

- Persediaan:Jika Anda menghapus peran terkait layanan yang digunakan oleh kemampuan Inventaris, maka data Inventaris untuk tag dan grup sumber daya tidak akan disinkronkan lagi. Anda harus membersihkan sumber daya peran terkait layanan sebelum menghapusnya secara manual.
- Explorer:Jika Anda menghapus peran terkait layanan yang digunakan olehExplorerkemampuan, lalu lintas akun dan lintas wilayah OpsData danOpsItemstidak lagi dapat dilihat.

> **Note**
>
> JikaSystems Managerlayanan menggunakan peran saat Anda mencoba menghapus tag atau grup sumber daya, maka penghapusan mungkin gagal. Jika hal itu terjadi, tunggu beberapa menit dan coba lagi.

Untuk menghapus sumber daya Systems Manager yang digunakan oleh **`AWSServiceRoleForAmazonSSM`**

1. Untuk menghapus tag, lihatMenambahkan dan menghapus tag pada sumber daya individu.
2. Untuk menghapus grup sumber daya, lihatHapus grup dariAWS Resource Groups.

Untuk menghapus secara manual**`AWSServiceRoleForAmazonSSM`**peran terkait layanan menggunakan IAM

Anda dapat menggunakan konsol IAM, AWS CLI, atau IAM API untuk menghapus peran terkait layanan `AWSServiceRoleForAmazonSSM`. Untuk informasi selengkapnya, lihat Menghapus peran tertaut layanan dalam Panduan Pengguna IAM.

Wilayah yang Didukung untukSystems Manager **`AWSServiceRoleForAmazonSSM`**peran terkait layanan

Systems Managermendukung menggunakan`AWSServiceRoleForAmazonSSM`peran terkait layanan di semuaWilayah AWSdi mana layanan tersedia. Untuk informasi lebih lanjut, lihat AWS Systems Manager kuota dan titik akhir.

Menggunakan peran untuk mengumpulkan Akun AWS informasi untuk OpsCenter dan Explorer

Systems Managermenggunakan peran terkait layanan bernama. **`AWSServiceRoleForAmazonSSM_AccountDiscovery`** AWS Systems Manager menggunakan peran layanan IAM ini untuk memanggil orang lain Layanan AWS untuk menemukan Akun AWS informasi.

Izin peran terkait layanan untuk penemuan akun Systems Manager

Peran tertaut layanan `AWSServiceRoleForAmazonSSM_AccountDiscovery` memercayai layanan berikut untuk mengambil peran tersebut:

- `accountdiscovery.ssm.amazonaws.com`

Kebijakan izin peran mengizinkan Systems Manager untuk menyelesaikan tindakan berikut pada sumber daya yang ditentukan:

- `organizations:DescribeAccount`
- `organizations:DescribeOrganizationalUnit`
- `organizations:DescribeOrganization`
- `organizations:ListAccounts`
- `organizations:ListAWSServiceAccessForOrganization`
- `organizations:ListChildren`
- `organizations:ListParents`
- `organizations:ListDelegatedServicesForAccount`
- `organizations:ListDelegatedAdministrators`
- `organizations:ListRoots`

Anda harus mengonfigurasi izin untuk mengizinkan entitas IAM (seperti pengguna, grup, atau peran) untuk membuat, mengedit, atau menghapus peran terkait layanan. Untuk informasi selengkapnya, lihat Izin peran tertaut layanan dalam Panduan Pengguna IAM.

Membuat peran **`AWSServiceRoleForAmazonSSM_AccountDiscovery`** terkait layanan untuk Systems Manager

Anda harus membuat peran terkait layanan jika ingin menggunakan Explorer danOpsCenter, kemampuan Systems Manager, di beberapa. Akun AWS UntukOpsCenter, Anda harus membuat peran terkait layanan secara manual. Untuk informasi selengkapnya, lihat (Opsional) Menyiapkan OpsCenter untuk mengelola secara terpusat OpsItems di seluruh akun.

UntukExplorer, jika Anda membuat sinkronisasi data sumber daya dengan menggunakan Systems Manager di AWS Management Console, Anda dapat membuat peran terkait layanan dengan memilih

tombol Buat peran. Jika Anda ingin membuat sinkronisasi data sumber daya secara terprogram, maka Anda harus membuat peran sebelum Anda membuat sinkronisasi data sumber daya. Anda dapat membuat peran dengan menggunakan operasi CreateServiceLinkedRoleAPI.

Mengedit peran **AWSServiceRoleForAmazonSSM_AccountDiscovery** terkait layanan untuk Systems Manager

Systems Managertidak memungkinkan Anda untuk mengedit peran `AWSServiceRoleForAmazonSSM_AccountDiscovery` terkait layanan. Setelah membuat peran terkait layanan, Anda tidak dapat mengubah nama peran karena berbagai entitas mungkin mereferensikan peran tersebut. Namun, Anda dapat menyunting penjelasan peran menggunakan IAM. Untuk informasi selengkapnya, lihat Mengedit peran tertaut layanan dalam Panduan Pengguna IAM.

Menghapus peran terkait **AWSServiceRoleForAmazonSSM_AccountDiscovery** layanan untuk Systems Manager

Jika Anda tidak lagi memerlukan penggunaan fitur atau layanan yang memerlukan peran terkait layanan, kami menyarankan Anda untuk menghapus peran tersebut. Dengan begitu Anda tidak memiliki entitas yang tidak digunakan yang tidak dipantau atau dipelihara secara aktif. Namun, Anda harus membersihkan peran terkait layanan sebelum Anda dapat menghapusnya.

Membersihkan peran **AWSServiceRoleForAmazonSSM_AccountDiscovery** terkait layanan

Sebelum Anda dapat menggunakan IAM untuk menghapus peran `AWSServiceRoleForAmazonSSM_AccountDiscovery` terkait layanan, Anda harus terlebih dahulu menghapus semua sinkronisasi data Explorer sumber daya. Untuk informasi selengkapnya, lihat Menghapus sinkronisasi data sumber daya untuk Systems Manager Explorer.

> **Note**
>
> Jika layanan Systems Manager menggunakan peran tersebut ketika Anda mencoba menghapus sumber daya, penghapusan mungkin gagal. Jika hal itu terjadi, tunggu beberapa menit dan coba mengoperasikannya lagi.

Hapus peran terkait **`AWSServiceRoleForAmazonSSM_AccountDiscovery`** layanan secara manual

Gunakan konsol IAM, the AWS CLI, atau AWS API untuk menghapus peran `AWSServiceRoleForAmazonSSM_AccountDiscovery` terkait layanan. Untuk informasi selengkapnya, silakan lihat Menghapus peran terkait layanan di Panduan Pengguna IAM.

Wilayah yang Didukung untuk peran Systems Manager**`AWSServiceRoleForAmazonSSM_AccountDiscovery`** terkait layanan

Systems Manager mendukung penggunaan peran terkait layanan di semua Wilayah tempat layanan tersedia. Untuk informasi lebih lanjut, lihat AWS Systems Manager kuota dan titik akhir.

Pembaruan untuk peran AWSServiceRoleForAmazonSSM_AccountDiscovery terkait layanan

Lihat detail tentang pembaruan pada peran AWSServiceRoleForAmazonSSM_AccountDiscovery terkait layanan sejak layanan ini mulai melacak perubahan ini. Untuk peringatan otomatis tentang perubahan pada halaman ini, berlangganan RSS feed pada halaman Systems Manager Riwayat dokumen.

Perubahan	Deskripsi	Tanggal
Izin baru ditambahkan	Peran terkait layanan ini sekarang termasuk `organizations:DescribeOrganizationalUnit` dan `organizations:ListRoots` izin. Izin ini memungkinkan akun AWS Organizations manajemen atau akun administrator yang didelegasikan Systems Manager untuk bekerja dengan OpsItems seluruh akun. Untuk informasi selengkapnya, lihat (Opsional) Menyiapkan OpsCenter untuk	Oktober 17, 2022

Perubahan	Deskripsi	Tanggal
	mengelola secara terpusat OpsItems di seluruh akun.	

Menggunakan peran untuk membuat OpsData danOpsItemsuntukExplorer

Systems Managermenggunakan peran terkait layanan bernama**AWSServiceRoleForSystemsManagerOpsDataSync**—AWS Systems Managermenggunakan peran layanan IAM ini untukExploreruntuk membuat OpsData danOpsItems.

Izin peran terkait layanan untukSystems Manager OpsData sinkronisasi

`AWSServiceRoleForSystemsManagerOpsDataSync` peran terkait layanan memercayakan layanan berikut untuk menjalankan peran tersebut:

- `opsdatasync.ssm.amazonaws.com`

Kebijakan izin peran mengizinkan Systems Manager untuk menyelesaikan tindakan berikut pada sumber daya yang ditentukan:

- Systems Manager Explorer mensyaratkan bahwa peran terkait layanan memberikan izin untuk memperbarui temuan keamanan saatOpsItemdiperbarui, membuat dan memperbaruiOpsItem, dan matikan sumber data Hub Keamanan saat aturan terkelola SSM dihapus oleh pelanggan.

Kebijakan terkelola yang digunakan untuk memberikan izin untuk peran `AWSServiceRoleForSystemsManagerOpsDataSync` adalah `AWSSystemsManagerOpsDataSyncServiceRolePolicy`. Untuk detail tentang izin yang diberikannya, lihat AWS kebijakan terkelola: AWSSystemsManagerOpsDataSyncServiceRolePolicy.

Anda harus mengonfigurasikan izin untuk mengizinkan entitas IAM (seperti pengguna, grup, atau peran) untuk membuat, menyunting, atau menghapus peran terhubung dengan layanan. Untuk informasi selengkapnya, lihat Izin peran tertaut layanan dalam Panduan Pengguna IAM.

Menciptakan**AWSServiceRoleForSystemsManagerOpsDataSync**peran terkait layanan untukSystems Manager

Anda tidak perlu membuat peran terkait layanan secara manual. Saat Anda mengaktifkanExplorerdiAWS Management Console,Systems Managermenciptakan peran terkait layanan untuk Anda.

> **Important**
>
> Peran terkait layanan ini dapat ditampilkan di akun Anda jika Anda menyelesaikan tindakan di layanan lain yang menggunakan fitur yang didukung oleh peran ini. Juga, jika Anda menggunakanSystems Managerlayanan sebelum 1 Januari 2017, ketika mulai mendukung peran terkait layanan, kemudianSystems ManagermenciptakanAWSServiceRoleForSystemsManagerOpsDataSyncperan dalam akun Anda. Untuk mempelajari lebih lanjut, lihatPeran baru muncul di akun IAM saya.

Jika Anda menghapus peran tertaut layanan ini, dan ingin membuatnya lagi, Anda dapat mengulangi proses yang sama untuk membuat kembali peran tersebut di akun Anda. Saat Anda mengaktifkanExplorerdiAWS Management Console,Systems Managermenciptakan peran terkait layanan untuk Anda lagi.

Anda juga dapat menggunakan konsol IAM untuk membuat peran terkait layanan denganAWSperan layanan yang memungkinkanExploreruntuk membuat OpsData danOpsItemskasus penggunaan. Di AWS CLI atau API AWS, buat peran yang terhubung dengan layanan dengan nama layanan `opsdatasync.ssm.amazonaws.com`. Untuk informasi lebih lanjut, lihat Membuat peran terkait layanan dalam Panduan Pengguna IAM. Jika Anda menghapus peran tertaut layanan ini, Anda dapat mengulang proses yang sama untuk membuat peran tersebut lagi.

Mengedit**AWSServiceRoleForSystemsManagerOpsDataSync**peran terkait layanan untukSystems Manager

Systems Managertidak memungkinkan Anda untuk mengeditAWSServiceRoleForSystemsManagerOpsDataSyncperan terkait layanan. Setelah membuat peran terkait layanan, Anda tidak dapat mengubah nama peran karena berbagai entitas mungkin mereferensikan peran tersebut. Namun, Anda dapat menyunting penjelasan peran menggunakan IAM. Untuk informasi selengkapnya, lihat Mengedit peran yang terkait dengan layanan dalam Panduan Pengguna IAM.

Menghapus**AWSServiceRoleForSystemsManagerOpsDataSync**peran terkait layanan untukSystems Manager

Jika Anda tidak lagi memerlukan penggunaan fitur atau layanan yang memerlukan peran terkait layanan, kami menyarankan Anda untuk menghapus peran tersebut. Dengan begitu Anda tidak memiliki entitas yang tidak digunakan yang tidak dipantau atau dipelihara secara aktif. Tetapi, Anda harus membersihkan sumber daya peran terkait layanan sebelum menghapusnya secara manual.

> **Note**
>
> Jika layanan Systems Manager menggunakan peran tersebut ketika Anda mencoba menghapus sumber daya, penghapusan mungkin gagal. Jika hal itu terjadi, tunggu beberapa menit dan coba mengoperasikannya lagi.

Prosedur untuk menghapusSystems Managersumber daya yang digunakan oleh`AWSServiceRoleForSystemsManagerOpsDataSync`peran tergantung pada apakah Anda telah mengonfigurasi`Explorer`atau`OpsCenter`untuk berintegrasi dengan Security Hub.

Untuk menghapusSystems Managersumber daya yang digunakan oleh**`AWSServiceRoleForSystemsManagerOpsDataSync`**peran

- Untuk berhenti`Explorer`dari menciptakan baru`OpsItems`untuk temuan Security Hub, lihatCara berhenti menerima temuan.
- Untuk berhenti`OpsCenter`dari menciptakan baru`OpsItems`untuk temuan Security Hub, lihat

Untuk menghapus secara manual**`AWSServiceRoleForSystemsManagerOpsDataSync`**peran terkait layanan menggunakan IAM

Gunakan konsol IAM, AWS CLI, atau AWS API untuk menghapus peran terkait layanan `AWSServiceRoleForSystemsManagerOpsDataSync`. Untuk informasi selengkapnya, lihat Menghapus peran tertaut layanan dalam Panduan Pengguna IAM.

Wilayah yang Didukung untukSystems Manager **`AWSServiceRoleForSystemsManagerOpsDataSync`**peran terkait layanan

Systems Manager memberikan dukungan dengan peran yang terhubung dengan layanan di semua Wilayah tempat layanan tersedia. Untuk informasi lebih lanjut, lihat AWS Systems Manager kuota dan titik akhir.

Systems Managertidak mendukung penggunaan peran terkait layanan di setiap Wilayah tempat layanan tersedia. Anda dapat menggunakan`AWSServiceRoleForSystemsManagerOpsDataSync`berperan dalam daerah-daerah berikut.

Wilayah AWS nama	Identitas wilayah	Dukungan diSystems Manager
US East (Northern Virginia)	us-east-1	Ya
US East (Ohio)	us-east-2	Ya
US West (N. California)	us-west-1	Ya
US West (Oregon)	us-west-2	Ya
Asia Pacific (Mumbai)	ap-south-1	Ya
Asia Pacific (Osaka)	ap-northeast-3	Ya
Asia Pacific (Seoul)	ap-northeast-2	Ya
Asia Pacific (Singapore)	ap-southeast-1	Ya
Asia Pacific (Sydney)	ap-southeast-2	Ya
Asia Pacific (Tokyo)	ap-northeast-1	Ya
Canada (Central)	ca-sentral-1	Ya
Eropa (Frankfurt)	eu-central-1	Ya
Eropa (Irlandia)	eu-west-1	Ya

Wilayah AWS nama	Identitas wilayah	Dukungan diSystems Manager
Eropa (London)	eu-west-2	Ya
Europe (Paris)	eu-west-3	Ya
Eropa (Stockholm)	eu-north-1	Ya
South America (São Paulo)	sa-east-1	Ya
AWS GovCloud (US)	us-gov-west-1	Tidak

Menggunakan peran untuk menciptakan wawasan operasional OpsItems di Manajer SistemOpsCenter

Systems Managermenggunakan peran terkait layanan bernama**`AWSServiceRoleForAmazonSSM_OpsInsights`**.AWS Systems Managermenggunakan peran layanan IAM ini untuk membuat dan memperbarui wawasan operasional OpsItems di Manajer SistemOpsCenter.

`AWSServiceRoleForAmazonSSM_OpsInsights`izin peran terkait layanan untukSystems Managerwawasan operasional OpsItems

`AWSServiceRoleForAmazonSSM_OpsInsights` peran terkait layanan memercayakan layanan berikut untuk menjalankan peran tersebut:

- `opsinsights.ssm.amazonaws.com`

Kebijakan izin peran mengizinkan Systems Manager untuk menyelesaikan tindakan berikut pada sumber daya yang ditentukan:

```
{
 "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"
    }
   }
  }
 ]
}
```

Anda harus mengonfigurasi izin untuk mengizinkan entitas IAM (seperti pengguna, grup, atau peran) untuk membuat, mengedit, atau menghapus peran terkait layanan. Untuk informasi selengkapnya, lihat Izin peran tertaut layanan dalam Panduan Pengguna IAM.

Menciptakan**AWSServiceRoleForAmazonSSM_OpsInsights**peran terkait layanan untukSystems Manager

Anda harus membuat peran terkait layanan. Jika Anda mengaktifkan wawasan operasional dengan menggunakanSystems ManagerdiAWS Management Console, Anda dapat membuat peran terkait layanan dengan memilihAktifkantombol.

Mengedit**AWSServiceRoleForAmazonSSM_OpsInsights**peran terkait layanan untukSystems Manager

Systems Manager tidak mengizinkan Anda untuk mengedit peran tertaut layanan `AWSServiceRoleForAmazonSSM_OpsInsights`. Setelah Anda membuat peran terkait layanan, Anda tidak dapat mengubah nama peran karena berbagai entitas mungkin mereferensikan peran tersebut. Namun, Anda dapat mengedit penjelasan peran menggunakan IAM. Untuk informasi selengkapnya, lihat Mengedit peran yang terkait dengan layanan dalam Panduan Pengguna IAM.

Menghapus**AWSServiceRoleForAmazonSSM_OpsInsights**peran terkait layanan untukSystems Manager

Jika Anda tidak perlu lagi menggunakan fitur atau layanan yang memerlukan peran terkait layanan, kami merekomendasikan Anda menghapus peran tersebut. Dengan begitu, Anda tidak memiliki entitas yang tidak digunakan dan tidak dipantau atau dipelihara secara aktif. Namun, Anda harus membersihkan peran tertaut layanan terlebih dahulu sebelum dapat menghapusnya secara manual.

Membersihkan**AWSServiceRoleForAmazonSSM_OpsInsights**peran terkait layanan

Sebelum Anda dapat menggunakan IAM untuk menghapus`AWSServiceRoleForAmazonSSM_OpsInsights`peran terkait layanan, Anda harus terlebih dahulu menonaktifkan wawasan operasional di Manajer SistemOpsCenter. Untuk informasi selengkapnya, lihat Menganalisis wawasan operasional untuk mengurangi OpsItems.

Hapus secara manual**AWSServiceRoleForAmazonSSM_OpsInsights**peran terkait layanan

Gunakan konsol IAM, AWS CLI, atau AWS API untuk menghapus peran terkait layanan `AWSServiceRoleForAmazonSSM_OpsInsights` Untuk informasi selengkapnya, lihat Menghapus peran tertaut layanan dalam Panduan Pengguna IAM.

Wilayah yang Didukung untukSystems Manager **AWSServiceRoleForAmazonSSM_OpsInsights**peran terkait layanan

Systems Managertidak mendukung penggunaan peran terkait layanan di setiap Wilayah tempat layanan tersedia. Anda dapat menggunakan AWSServiceRoleForAmazonSSM_OpsInsights berperan dalam daerah-daerah berikut.

Nama wilayah	Identitas wilayah	Dukungan diSystems Manager
US East (Northern Virginia)	us-east-1	Ya
US East (Ohio)	us-east-2	Ya
US West (N. California)	us-west-1	Ya
US West (Oregon)	us-west-2	Ya
Asia Pacific (Mumbai)	ap-south-1	Ya

Nama wilayah	Identitas wilayah	Dukungan diSystems Manager
Asia Pacific (Tokyo)	ap-northeast-1	Ya
Asia Pacific (Seoul)	ap-northeast-2	Ya
Asia Pacific (Singapore)	ap-southeast-1	Ya
Asia Pacific (Sydney)	ap-southeast-2	Ya
Asia Pasifik (Hong Kong)	ap-east-1	Ya
Canada (Central)	ca-sentral-1	Ya
Eropa (Frankfurt)	eu-central-1	Ya
Eropa (Irlandia)	eu-west-1	Ya
Eropa (London)	eu-west-2	Ya
Europe (Paris)	eu-west-3	Ya
Eropa (Stockholm)	eu-north-1	Ya
Eropa (Milan)	eu-south-1	Ya
South America (São Paulo)	sa-east-1	Ya
Timur Tengah (Bahrain)	me-south-1	Ya
Afrika (Cape Town)	af-south-1	Ya
AWS GovCloud (US)	us-gov-west-1	Ya
AWS GovCloud (US)	us-gov-east-1	Ya

Pencatatan dan pemantauan di AWS Systems Manager

Pemantauan adalah bagian penting dari pemeliharaan keandalan, ketersediaan, dan performa AWS Systems Manager serta solusi AWS Anda. Anda harus mengumpulkan data pemantauan dari

semua bagianAWSsolusi sehingga Anda dapat lebih melakukan debug kegagalan multi-titik jika terjadi.AWSmenyediakan beberapa alat untuk memantauSystems Managerdan sumber daya lainnya dan merespons potensi insiden.

Log AWS CloudTrail

CloudTrail menyediakan catatan tindakan yang diambil oleh pengguna, peran, atauLayanan AWSdiSystems Manager. Menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuatSystems Manager, alamat IP tempat permintaan dibuat, siapa yang membuat permintaan, kapan permintaan dibuat, dan detail tambahan. Untuk informasi selengkapnya, lihat Pencatatan panggilan AWS Systems Manager API dengan AWS CloudTrail.

Amazon CloudWatch alarm

Menggunakan Amazon CloudWatch alarm, Anda dapat melihat satu metrik selama periode waktu yang Anda tentukan untuk instans Amazon Elastic Compute Cloud (Amazon EC2) dan sumber daya lainnya. Jika metrik melebihi ambang batas tertentu, notifikasi dikirim ke topik Amazon Simple Notification Service (Amazon SNS) atauAWS Auto Scalingkebijakan. CloudWatch alarm tidak memicu tindakan hanya karena alarm tersebut berada dalam keadaan tertentu. Sebaliknya, kondisi harus diubah dan dipertahankan untuk beberapa periode tertentu. Untuk informasi selengkapnya, lihatMenggunakan Amazon CloudWatch alarmdi dalamAmazon CloudWatch Panduan Pengguna.

Amazon CloudWatch dasbor

CloudWatch dasbor adalah halaman beranda yang dapat disesuaikan di CloudWatch konsol yang dapat Anda gunakan untuk memantau sumber daya Anda dalam satu tampilan, bahkan sumber daya yang tersebar di berbagaiWilayah AWS. Anda dapat menggunakan CloudWatch dasbor untuk membuat tampilan metrik dan alarm yang disesuaikan untukAWSsumber daya Untuk informasi selengkapnya, lihat CloudWatchDasbor Amazon dasbor Amazon dasbor Amazon dasbor Amazon.

Amazon EventBridge

Menggunakan Amazon EventBridge, Anda dapat mengkonfigurasi aturan untuk mengingatkan Anda tentang perubahanSystems Managersumber daya, dan untuk mengarahkan EventBridge untuk mengambil tindakan berdasarkan isi acara tersebut. EventBridge memberikan dukungan untuk sejumlah peristiwa yang dipancarkan oleh berbagaiSystems Managerkemampuan. Untuk informasi selengkapnya, lihat Pemantauan peristiwa Systems Manager dengan Amazon EventBridge.

Amazon CloudWatch Log danSSM Agentlog

SSM Agentmenulis informasi tentang eksekusi, tindakan terjadwal, kesalahan, dan status kondisi untuk file log pada setiap node. Anda dapat melihat file log dengan menghubungkan secara manual ke node. Kami menyarankan untuk secara otomatis mengirimkan data log agen ke grup log di CloudWatch Log untuk analisis. Untuk informasi selengkapnya, lihat Mengirim log simpul ke CloudWatch Log terpadu (CloudWatch agen) dan Melihat SSM Agent log.

AWS Systems Manager Kepatuhan

Anda dapat menggunakan Kepatuhan, kemampuanAWS Systems Manager, untuk memindai armada node terkelola untuk kepatuhan patch dan inkonsistensi konfigurasi. Anda dapat mengumpulkan dan menggabungkan data dari beberapa Akun AWS dan Wilayah AWS, lalu menelusuri ke sumber daya tertentu yang tidak sesuai. Secara default, Kepatuhan menampilkan data kepatuhan tentang patching diPatch Manager, suatu kemampuanAWS Systems Manager, dan asosiasi dalamState Manager, suatu kemampuanAWS Systems Manager. Untuk informasi selengkapnya, lihat AWS Systems Manager Kepatuhan.

AWS Systems Manager Explorer

Explorer, suatu kemampuanAWS Systems Manager, adalah dasbor operasi yang dapat disesuaikan yang melaporkan informasi tentangAWSsumber dayaExplorermenampilkan tampilan gabungan data operasi (OpsData) untukAkun AWSdan di seberangWilayah AWS. MasukExplorer, OpsData menyertakan metadata tentang instans EC2 Anda, detail kepatuhan patch, dan item pekerjaan operasional (OpsItems).Explorermenyediakan konteks tentang bagaimanaOpsItemsdidistribusikan ke seluruh unit bisnis atau aplikasi Anda, bagaimana trennya dari waktu ke waktu, dan bagaimana perbedaannya menurut kategori. Anda dapat mengelompokkan dan memfilter informasiExploreruntuk fokus pada item yang relevan dengan Anda dan yang memerlukan tindakan. Untuk informasi selengkapnya, lihat AWS Systems Manager Explorer.

AWS Systems Manager OpsCenter

OpsCenter, suatu kemampuanAWS Systems Manager, menyediakan lokasi pusat di mana teknisi operasi dan profesional IT dapat melihat, menyelidiki, dan menyelesaikan item pekerjaan operasional (OpsItems) terkaitAWSsumber dayaOpsCenteragregat dan standarisasiOpsItemslintas layanan sambil memberikan data investigasi kontekstual tentang masing-masingOpsItem, terkaitOpsItems, dan sumber daya terkait.OpsCenterjuga menyediakan runbook di Otomatisasi, suatu kemampuanAWS Systems Manager, yang dapat Anda gunakan untuk menyelesaikan masalah dengan cepat.OpsCenterterintegrasi dengan Amazon

EventBridge. Ini berarti Anda dapat membuat EventBridge aturan yang secara otomatis membuatOpsItemsuntuk setiapLayanan AWSyang menerbitkan acara ke EventBridge. Untuk informasi selengkapnya, lihat AWS Systems Manager OpsCenter.

Amazon Simple Notification Service

Anda dapat mengonfigurasi Amazon Simple Notification Service (Amazon SNS) untuk mengirim notifikasi tentang status perintah yang Anda kirim menggunakanRun CommandatauMaintenance Windows, kemampuanAWS Systems Manager. Amazon SNS mengoordinasikan dan mengelola pemberitahuan pengiriman ke pelanggan atau titik akhir yang berlangganan. Anda dapat menerima notifikasi setiap kali perintah berubah ke tahapan baru atau keadaan tertentu, sepertiFailedatauTimed Out. Dalam kasus di mana Anda mengirim perintah ke beberapa node, Anda dapat menerima notifikasi untuk setiap salinan perintah yang dikirimkan ke node tertentu. Untuk informasi selengkapnya, lihat Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS.

AWS Trusted Advisor dan AWS Health Dashboard

Trusted Advisor mengacu pada praktik terbaik yang dipelajari dari melayani ratusan ribu pelanggan AWS. Trusted Advisor memeriksa lingkungan AWS Anda lalu membuat rekomendasi ketika ada peluang untuk menghemat uang, meningkatkan ketersediaan dan performa sistem, atau membantu menutup kesenjangan keamanan. Semua pelanggan AWS memiliki akses ke lima pemeriksaan Trusted Advisor. Pelanggan dengan rencana Business atau Enterprise AWS Support dapat melihat semua pemeriksaan Trusted Advisor. Untuk informasi selengkapnya, lihatAWS Trusted Advisordi dalamAWS SupportPanduan PenggunadanAWS HealthPanduan Pengguna.

Info selengkapnya

- Pemantauan AWS Systems Manager

Validasi kepatuhan untuk AWS Systems Manager

Topik ini membahas pematuhan AWS Systems Manager dengan program jaminan pihak ketiga. Untuk informasi tentang data kepatuhan untuk informasi tentang data kepatuhan untuk informasi yang terkelolaAWS Systems Manager Kepatuhan,

Auditor pihak ketiga menilai keamanan dan kepatuhan pada Systems Manager sebagai bagian dari beberapa program kepatuhan AWS. Program ini mencakup SOC, PCI, FedRAMP, HIPAA, dan lainnya.

Untuk daftarLayanan AWS dalam cakupan program kepatuhan tertentu, lihat AWSLayanan dalam Cakupan berdasarkan Program KepatuhanLayanan AWS . Untuk informasi umum, lihat Program Kepatuhan AWS.

Anda bisa mengunduh laporan audit pihak ketiga menggunakan AWS Artifact. Untuk informasi selengkapnya, lihat Mengunduh laporan diAWS Artifact .

Tanggung jawab kepatuhan Anda saat menggunakan Systems Manager ditentukan oleh sensitivitas data Anda, tujuan kepatuhan perusahaan Anda, serta undang-undang dan peraturan yang berlaku. AWS menyediakan sumber daya berikut untuk membantu dengan kepatuhan:

- Panduan Quick Start Keamanan dan Kepatuhan – Panduan penerapan ini membahas pertimbangan arsitektur dan menyediakan langkah untuk deployment lingkungan dasar yang berfokus pada keamanan dan kepatuhan di AWS.
- Arsitek untuk Whitepaper Keamanan dan Kepatuhan HIPAA — Laporan resmi ini menjelaskan bagaimana perusahaan bisa menggunakanAWS HIPAA.
- Sumber Daya Kepatuhan AWS – Kumpulan buku kerja dan panduan ini mungkin berlaku untuk industri dan lokasi Anda.
- Mengevaluasi Sumber Daya dengan Aturan di Panduan Developer AWS Config – Layanan AWS Config menilai seberapa baik konfigurasi sumber daya Anda dalam mematuhi praktik-praktik internal, pedoman industri, dan regulasi internal.
- AWS Security Hub – Layanan AWS ini menyediakan pandangan yang komprehensif tentang status keamanan Anda dalam AWS yang membantu Anda memeriksa kepatuhan Anda terhadap standar industri dan praktik terbaik untuk keamanan.

Ketahanan di AWS Systems Manager

Infrastruktur global AWS dibangun di sekitar Wilayah AWS dan Availability Zone. Wilayah AWS menyediakan beberapa Availability Zone yang terpisah secara fisik dan terisolasi yang terhubung dengan jaringan latensi rendah, throughput tinggi, dan jaringan yang sangat berlebihan. Dengan Availability Zone, Anda dapat merancang serta mengoperasikan aplikasi dan basis data yang secara otomatis mengalami fail over antar zona tanpa gangguan. Availability Zone memiliki ketersediaan yang lebih baik, toleran terhadap kegagalan, dan dapat diukur skalanya jika dibandingkan dengan satu atau beberapa infrastruktur pusat data tradisional.

Untuk informasi selengkapnya tentang Wilayah AWS dan Availability Zone, lihat Infrastruktur Global AWS.

Keamanan infrastruktur dalam AWS Systems Manager

Sebagai layanan terkelola,AWS Systems Managerdilindungi olehAWSkeamanan jaringan global. Untuk informasi tentangAWSlayanan keamanan dan bagaimanaAWSmelindungi infrastruktur, lihatAWSKeamanan Cloud. Untuk mendesainAWSlingkungan menggunakan praktik terbaik untuk keamanan infrastruktur, lihatPerlindungan InfrastrukturdiPilar KeamananAWSKerangka Kerja yang Diarsiteksikan dengan Baik.

Anda menggunakan panggilan API AWS yang dipublikasikan untuk mengakses Systems Manager melalui jaringan. Klien harus mendukung hal berikut:

- Transport Layer Security (TLS). Kami membutuhkan TLS 1.2 dan merekomendasikan TLS 1.3.
- Suite cipher dengan kerahasiaan maju sempurna (PFS) seperti DHE (Ephemeral Diffie-Hellman) atau ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Sebagian besar sistem modern seperti Java 7 dan sistem yang lebih baru mendukung mode ini.

Selain itu, permintaan harus ditandatangani menggunakan access key ID dan secret access key yang terkait dengan principal IAM. Atau Anda bisa menggunakan AWS Security Token Service (AWS STS) untuk membuat kredensial keamanan sementara guna menandatangani permintaan.

Analisis konfigurasi dan kerentanan dalam AWS Systems Manager

AWSmenangani tugas keamanan dasar seperti konfigurasi firewall dan pemulihan bencana. Prosedur ini telah ditinjau dan disertifikasi oleh pihak ketiga yang sesuai. Untuk detail selengkapnya, lihat sumber daya berikut:

- Validasi kepatuhan untuk AWS Systems Manager
- Model Tanggung Jawab Bersama
- Praktik Terbaik untuk Keamanan, Identitas, & Kepatuhan

Praktik terbaik keamanan untuk Systems Manager

AWS Systems Manager menyediakan sejumlah fitur keamanan untuk dipertimbangkan ketika Anda mengembangkan dan menerapkan kebijakan keamanan Anda sendiri. Praktik terbaik berikut adalah pedoman umum dan tidak mewakili solusi keamanan yang lengkap. Karena praktik terbaik

ini mungkin tidak sesuai atau cukup untuk lingkungan Anda, anggap praktik terbaik tersebut sebagai pertimbangan yang membantu dan bukan sebagai rekomendasi.

Topik

Systems Managerpraktik terbaik keamanan preventif

Praktik terbaik berikut ini Systems Manager dapat membantu mencegah insiden keamanan.

Terapkan akses hak istimewa yang paling rendah

Saat memberikan izin, Anda memutuskan siapa yang mendapatkan izin apa untuk sumber daya mana. Systems Manager Anda mengizinkan tindakan tertentu yang ingin Anda lakukan di sumber daya tersebut. Oleh karena itu, Anda harus memberikan hanya izin yang diperlukan untuk melaksanakan tugas. Menerapkan akses hak istimewa yang terkecil adalah hal mendasar dalam mengurangi risiko keamanan dan dampak yang dapat diakibatkan oleh kesalahan atau niat jahat.

Alat bantu berikut tersedia untuk menerapkan akses hak istimewa terkecil:

- Kebijakan IAM dan batasan Izin untuk entitas IAM
- Kebijakan kontrol layanan

Gunakan SecureString parameter untuk mengenkripsi dan melindungi data rahasia

DalamParameter Store, kemampuanAWS Systems Manager, `SecureString` parameter adalah setiap data sensitif yang perlu disimpan dan direferensikan dengan cara yang aman. Jika Anda memiliki data yang tidak ingin pengguna ubah atau referensi dalam teks biasa, seperti kata sandi atau kunci lisensi, buat parameter tersebut `SecureString` menggunakan tipe data. Parameter Storemenggunakan AWS KMS key in AWS Key Management Service (AWS KMS) untuk mengenkripsi nilai parameter. AWS KMSmenggunakan kunci yang dikelola pelanggan atau Kunci yang dikelola AWS saat mengenkripsi nilai parameter. Untuk keamanan maksimal, kami menyarankan untuk menggunakan kunci KMS Anda sendiri. Jika Anda menggunakanKunci yang dikelola AWS, setiap pengguna dengan izin untuk menjalankan GetParameterdan GetParameterstindakan di akun Anda dapat melihat atau mengambil konten dari semua `SecureString` parameter. Jika Anda menggunakan kunci yang dikelola pelanggan untuk mengenkripsi keamanan nilai-nilai `SecureString`, Anda dapat menggunakan kebijakan IAM dan kebijakan kunci untuk mengelola izin untuk mengenkripsi dan mendekripsi parameter. Lebih

sulit untuk membuat kebijakan kontrol akses untuk operasi ini ketika Anda menggunakan kunci yang dikelola pelanggan. Misalnya, jika Anda menggunakan `SecureString` parameter Kunci yang dikelola AWS untuk mengenkripsi dan tidak ingin pengguna bekerja dengan `SecureString` parameter, kebijakan IAM mereka harus secara eksplisit menolak akses ke kunci default.

Untuk informasi selengkapnya, lihat Membatasi akses ke parameter Systems Manager menggunakan kebijakan IAM dan Cara AWS Systems ManagerParameter Store Penggunaan AWS KMS di Panduan AWS Key Management Service Pengembang.

Tentukan allowedValues dan allowedPattern untuk parameter dokumen

Anda dapat memvalidasi input pengguna untuk parameter dalam dokumen Systems Manager (dokumen SSM) dengan mendefinisikan `allowedValues` dan `allowedPattern`. Untuk `allowedValues`, Anda mendefinisikan sebuah array nilai yang diizinkan untuk parameter. Jika pengguna input nilai tidak diperbolehkan, eksekusi gagal untuk memulai. Untuk `allowedPattern`, Anda menentukan ekspresi reguler yang memvalidasi apakah input pengguna sesuai dengan pola yang ditetapkan untuk parameter. Jika input pengguna tidak cocok dengan pola yang diperbolehkan, eksekusi gagal untuk memulai.

Untuk informasi selengkapnya tentang `allowedValues` dan `allowedPattern`, lihat Elemen dan parameter data.

Memblokir berbagi dokumen untuk publik

Kecuali kasus penggunaan Anda mengharuskan berbagi publik diizinkan, sebaiknya aktifkan pengaturan blokir berbagi publik untuk dokumen SSM Anda di bagian Preferensi konsol kumpulan dokumen Systems Manager.

Gunakan Amazon Virtual Private Cloud (Amazon VPC) dan VPC endpoint

Anda dapat menggunakan Amazon VPC untuk memulai sumber daya AWS ke dalam jaringan virtual yang telah Anda tentukan. Jaringan virtual ini sangat mirip dengan jaringan tradisional yang akan Anda operasikan di pusat data Anda sendiri, dengan manfaatnya yaitu menggunakan infrastruktur AWS yang dapat diskalakan.

Dengan menerapkan titik akhir VPC, Anda dapat menghubungkan VPC Anda secara pribadi ke layanan endpoint VPC yang didukung Layanan AWS dan AWS PrivateLink didukung tanpa memerlukan gateway internet, perangkat NAT, koneksi VPN, atau koneksi. AWS Direct Connect Instans di VPC Anda tidak memerlukan alamat IP publik untuk berkomunikasi dengan sumber daya dalam layanan. Lalu lintas antara VPC Anda dan layanan lainnya tidak meninggalkan jaringan Amazon.

Untuk informasi selengkapnya tentang keamanan Amazon VPC, lihat Membuat titik akhir VPC dan privasi lalu lintas Internetwork di Amazon VPC di Panduan Pengguna Amazon VPC.

Batasi Session Manager pengguna untuk sesi menggunakan perintah interaktif dan dokumen sesi SSM tertentu

Session Manager, kemampuanAWS Systems Manager, menyediakan beberapa metode untuk memulai sesi ke node terkelola Anda. Untuk koneksi yang paling aman, Anda dapat meminta pengguna untuk terhubung menggunakan perintah interaktif metode untuk membatasi interaksi pengguna untuk perintah tertentu atau urutan perintah. Ini membantu Anda mengelola tindakan interaktif yang dapat dilakukan pengguna. Untuk informasi selengkapnya, lihat Memulai sesi (perintah interaktif dan noninteraktif).

Untuk keamanan tambahan, Anda dapat membatasi Session Manager akses ke instans Amazon EC2 tertentu dan dokumen sesi tertentuSession Manager. Anda memberikan atau mencabut Session Manager akses dengan cara ini dengan menggunakan kebijakan AWS Identity and Access Management (IAM). Untuk informasi selengkapnya, lihat Langkah 3: Kontrol akses sesi ke node yang dikelola.

Berikan izin node sementara untuk alur kerja Otomasi

Selama alur kerja di Otomasi, kemampuanAWS Systems Manager, node Anda mungkin memerlukan izin yang diperlukan untuk eksekusi itu saja tetapi tidak untuk operasi lainSystems Manager. Misalnya, alur kerja Otomasi mungkin memerlukan node untuk memanggil operasi API tertentu atau mengakses AWS sumber daya secara khusus selama alur kerja. Jika panggilan atau sumber daya ini adalah panggilan yang ingin Anda batasi aksesnya, Anda dapat memberikan izin tambahan sementara untuk node Anda dalam runbook Otomasi itu sendiri alih-alih menambahkan izin ke profil instans IAM Anda. Pada akhir alur kerja otomatisasi, izin sementara akan dihapus. Untuk informasi selengkapnya, lihat Memberikan izin instans sementara dengan Otomatisasi AWS Systems Manager pada Blog Pengelolaan dan Tata Kelola AWS.

Tetap AWS dan Systems Manager alat up to date

AWSsecara teratur merilis versi terbaru dari alat dan plugin yang dapat Anda gunakan dalam Systems Manager operasi AndaAWS. Menjaga agar sumber daya ini tetap mutakhir memastikan bahwa pengguna dan node di akun Anda memiliki akses ke fungsionalitas dan fitur keamanan terbaru di alat ini.

- SSM Agent— AWS Systems Manager Agent (SSM Agent) adalah perangkat lunak Amazon yang dapat diinstal dan dikonfigurasi pada instans Amazon Elastic Compute Cloud (Amazon EC2), server lokal, atau mesin virtual (VM). SSM Agentmemungkinkan Systems Manager untuk

memperbarui, mengelola, dan mengkonfigurasi sumber daya ini. Kami merekomendasikan memeriksa versi baru, atau mengotomatisasi pembaruan untuk agen, setidaknya setiap dua minggu. Untuk informasi, lihat Mengotomatiskan pembaruan ke SSM Agent. Kami juga menyarankan untuk memverifikasi tanda tangan SSM Agent sebagai bagian dari proses pembaruan Anda. Untuk informasi, lihat Memverifikasi tanda tangan SSM Agent.

- AWS CLI— The AWS Command Line Interface (AWS CLI) adalah alat open source yang memungkinkan Anda berinteraksi dengan Layanan AWS menggunakan perintah di shell baris perintah Anda. Untuk memperbarui AWS CLI, Anda menjalankan perintah yang sama yang digunakan untuk menginstal AWS CLI. Kami merekomendasikan membuat tugas terjadwal pada mesin lokal Anda untuk menjalankan perintah yang sesuai untuk sistem operasi Anda setidaknya sekali setiap dua minggu. Untuk informasi tentang perintah instalasi, lihat Menginstal AWS CLI versi 2 di Panduan AWS Command Line Interface Pengguna.
- AWS Tools for Windows PowerShell— Alat untuk Windows PowerShell adalah seperangkat PowerShell modul yang dibangun di atas fungsionalitas yang diekspos oleh AWS SDK for .NET. Ini AWS Tools for Windows PowerShell memungkinkan Anda untuk skrip operasi pada AWS sumber daya Anda dari baris PowerShell perintah. Secara berkala, saat versi terbaru dari Alat untuk Windows PowerShell dirilis, Anda harus memperbarui versi yang Anda jalankan secara lokal. Untuk selengkapnya, lihat AWS Tools for Windows PowerShellMemperbarui Windows atau Memperbarui AWS Tools for Windows PowerShell di Linux atau macOS di Panduan Pengguna Simulator Kebijakan IAM.
- Session ManagerPlugin — Jika pengguna di organisasi Anda dengan izin untuk menggunakan Session Manager ingin terhubung ke node menggunakanAWS CLI, mereka harus terlebih dahulu menginstal Session Manager plugin pada mesin lokal mereka. Untuk memperbarui plugin, Anda menjalankan perintah yang sama seperti yang digunakan untuk menginstal plugin. Kami merekomendasikan membuat tugas terjadwal pada mesin lokal Anda untuk menjalankan perintah yang sesuai untuk sistem operasi Anda setidaknya sekali setiap dua minggu. Untuk informasi, lihat Instal Session Manager plugin untuk AWS CLI.
- CloudWatch agen — Anda dapat mengonfigurasi dan menggunakan CloudWatch agen untuk mengumpulkan metrik dan log dari instans EC2, instans lokal, dan mesin virtual (VM). Log ini dapat dikirim ke Amazon CloudWatch Logs untuk pemantauan dan analisis. Kami merekomendasikan memeriksa versi baru, atau mengotomatisasi pembaruan untuk agen, setidaknya setiap dua minggu. Untuk pembaruan yang paling sederhana, gunakan Pengaturan Cepat AWS Systems Manager. Untuk informasi, lihat AWS Systems Manager Quick Setup.

Systems Managerpemantauan dan audit praktik terbaik

Praktik terbaik berikut ini Systems Manager dapat membantu mendeteksi potensi kelemahan dan insiden keamanan.

Identifikasi dan audit semua Systems Manager sumber daya Anda

Identifikasi aset IT Anda adalah aspek penting dari tata kelola dan keamanan. Anda perlu mengidentifikasi semua sumber Systems Manager daya Anda untuk menilai postur keamanan mereka dan mengambil tindakan pada area kelemahan potensial.

Gunakan Editor Tag untuk mengidentifikasi sumber daya yang sensitif terhadap keamanan atau audit, kemudian gunakan tag tersebut saat Anda perlu mencari sumber daya ini. Untuk informasi selengkapnya, lihat Menemukan sumber daya untuk ditandai di Panduan AWS Resource Groups Pengguna.

Buat grup sumber daya untuk Systems Manager sumber daya Anda. Untuk informasi selengkapnya, lihat Apa itu grup sumber daya?

Menerapkan pemantauan menggunakan alat CloudWatch pemantauan Amazon

Pemantauan adalah bagian penting dari menjaga keandalan, keamanan, ketersediaan, dan kinerja Systems Manager dan AWS solusi Anda. Amazon CloudWatch menyediakan beberapa alat dan layanan untuk membantu Anda memantau Systems Manager dan lainnyaLayanan AWS. Untuk informasi selengkapnya, lihat Mengirim log simpul ke CloudWatch Log terpadu (CloudWatch agen) dan Pemantauan peristiwa Systems Manager dengan Amazon EventBridge.

Gunakan CloudTrail

AWS CloudTrailmenyediakan catatan tindakan yang diambil oleh pengguna, peran, atau Layanan AWS dalamSystems Manager. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuatSystems Manager, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan. Untuk informasi selengkapnya, lihat Pencatatan panggilan AWS Systems Manager API dengan AWS CloudTrail.

Mengaktifkan AWS Config

AWS Configmemungkinkan Anda untuk menilai, mengaudit, dan mengevaluasi konfigurasi AWS sumber daya Anda. AWS Configmemantau konfigurasi sumber daya, memungkinkan Anda mengevaluasi konfigurasi yang direkam terhadap konfigurasi aman yang diperlukan. Dengan AWS Config, Anda dapat meninjau perubahan dalam konfigurasi dan hubungan antara sumber

daya AWS, menyelidiki riwayat konfigurasi sumber daya yang detail, dan menentukan kepatuhan secara keseluruhan terhadap konfigurasi yang ditentukan dalam pedoman internal Anda. Ini dapat membantu Anda menyederhanakan audit kepatuhan, analisis keamanan, manajemen perubahan, dan pemecahan masalah operasional. Untuk informasi lebih lanjut, lihat Menyiapkan AWS Config dengan Konsol tersebut pada Panduan Developer AWS Config. Saat menentukan jenis sumber daya yang akan direkam, pastikan Anda menyertakan Systems Manager sumber daya.

Memantau laporan keamanan AWS

Anda harus memeriksa laporan keamanan yang di-posting di Trusted Advisor untuk Akun AWS Anda secara berkala. Anda dapat melakukan ini secara terprogram menggunakan. describe-trusted-advisor-checks

Lebih lanjut, secara aktif memantau alamat surel utama yang terdaftar ke setiap Akun AWS. AWS akan mengkontak Anda, menggunakan alamat surel ini, tentang masalah keamanan yang muncul yang mungkin memengaruhi Anda.

Masalah operasional AWS dengan dampak luas di-posting pada Service Health Dashboard AWS. Masalah operasional juga di-posting ke akun individu melalui Personal Health Dashboard. Untuk informasi selengkapnya, lihat Dokumentasi AWS Health.

Info lebih lanjut

- Praktik Terbaik untuk Keamanan, Identitas, & Kepatuhan
- Memulai: Ikuti keamanan konfigurasi praktik terbaik Sumber Daya AWS Anda (Blog Keamanan AWS)
- Praktik terbaik keamanan di IAM
- Praktik terbaik keamanan di AWS CloudTrail
- Praktik Terbaik Keamanan untuk Amazon S3
- Praktik terbaik keamanan untuk AWS Key Management Service

Pemantauan AWS Systems Manager

Pemantauan adalah bagian penting dari menjaga keandalan, ketersediaan, dan kinerja AWS Systems Manager dan AWS solusi Anda. Anda harus mengumpulkan data pemantauan dari semua bagian AWS solusi Anda sehingga Anda dapat men-debug kegagalan multipoint jika terjadi. Tetapi, sebelum Anda mulai memantau Systems Manager, buat rencana pemantauan yang mencakup jawaban atas pertanyaan berikut:

- Apa sasaran pemantauan Anda?
- Sumber daya apa yang akan Anda pantau?
- Seberapa sering Anda akan memantau sumber daya ini?
- Alat pemantauan apa yang akan Anda gunakan?
- Siapa yang melakukan tugas pemantauan?
- Siapa yang harus diberi tahu saat terjadi kesalahan?

Setelah Anda menentukan tujuan pemantauan Anda dan membuat rencana pemantauan, langkah berikutnya adalah menetapkan dasar untuk performa Systems Manager normal di lingkungan Anda. Anda harus mengukur performa Systems Manager pada berbagai waktu dan di bawah berbagai kondisi beban. Saat memantau Systems Manager, Anda harus menyimpan riwayat data pemantauan yang telah Anda kumpulkan. Anda dapat membandingkan performa Systems Manager saat ini dengan data historis ini untuk membantu Anda mengidentifikasi pola performa normal dan anomali performa, serta membuat metode untuk menanganinya.

Misalnya, Anda dapat memantau keberhasilan atau kegagalan operasi seperti alur kerja Otomatisasi, aplikasi garis dasar patch, pemeliharaan acara jendela, dan kepatuhan konfigurasi. Otomasi adalah kemampuan AWS Systems Manager.

Anda juga dapat memantau pemanfaatan CPU, disk I/O, dan pemanfaatan jaringan node terkelola Anda. Ketika kinerja berada di luar baseline yang ditetapkan, Anda mungkin perlu mengkonfigurasi ulang atau mengoptimalkan node untuk mengurangi pemanfaatan CPU, meningkatkan I/O disk, atau mengurangi lalu lintas jaringan. Untuk informasi selengkapnya tentang pemantauan instans EC2, lihat Memantau Amazon EC2 di Panduan Pengguna Amazon EC2 untuk Instans Linux.

Topik

- MengirimSSM Agent log ke CloudWatch Log
- Memantau peristiwa permintaan perubahan
- Pemantauan Otomatisasi Anda
- PemantauanRun CommandMetrik menggunakan Amazon CloudWatch
- Pencatatan panggilan AWS Systems Manager API dengan AWS CloudTrail
- Pencatatan output tindakan Otomatisasi dengan CloudWatch Logs
- MengonfigurasiCloudWatch Log Amazon untukRun Command
- Pemantauan peristiwa Systems Manager dengan Amazon EventBridge
- Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

Alat-alat pemantauan

Konten dalam Bab ini menyediakan informasi untuk menggunakan alat yang tersedia untuk memantau Systems Manager Anda dan AWS sumber daya lainnya. Untuk daftar alat yang lebih lengkap, lihat Pencatatan dan pemantauan di AWS Systems Manager.

Mengirim log simpul ke CloudWatch Log terpadu (CloudWatch agen)

Anda dapat mengonfigurasi dan menggunakan CloudWatch agen Amazon untuk mengumpulkan metrik dan log dari node alih-alih menggunakan AWS Systems Manager Agent (SSM Agent) untuk tugas ini. CloudWatch Agen memungkinkan Anda mengumpulkan lebih banyak metrik pada instans EC2 daripada yang tersedia. SSM Agent Selain itu, Anda dapat mengumpulkan metrik dari server lokal menggunakan agen. CloudWatch

Anda juga dapat menyimpan pengaturan konfigurasi agen di Systems Manager Parameter Store untuk digunakan dengan CloudWatch agen. Parameter Storeadalah kemampuanAWS Systems Manager.

> **Note**
>
> AWS Systems Managermendukung migrasi dari SSM Agent ke CloudWatch agen terpadu untuk mengumpulkan log dan metrik hanya pada versi Windows 64-bit. Untuk informasi tentang cara menyiapkan CloudWatch agen terpadu di sistem operasi lain, dan untuk

informasi lengkap tentang penggunaan CloudWatch agen, lihat Mengumpulkan metrik dan log dari instans Amazon EC2 dan server lokal dengan CloudWatch agen di Panduan Pengguna Amazon. CloudWatch

Anda dapat menggunakan CloudWatch agen pada sistem operasi lain yang didukung, tetapi Anda tidak akan dapat menggunakan Systems Manager untuk melakukan migrasi alat.

SSM Agentmenulis informasi tentang eksekusi, tindakan terjadwal, kesalahan, dan status kesehatan untuk mencatat file di setiap node. Menghubungkan secara manual ke node untuk melihat file log dan memecahkan masalah dengan SSM Agent memakan waktu. Untuk pemantauan node yang lebih efisien, Anda dapat mengonfigurasi SSM Agent dirinya sendiri atau CloudWatch agen untuk mengirim data log ini ke Amazon CloudWatch Logs.

Important

CloudWatch Agen terpadu telah diganti SSM Agent sebagai alat untuk mengirim data log ke Amazon CloudWatch Logs. Plugin SSM Agent AWS:CloudWatch tidak didukung. Sebaiknya gunakan hanya CloudWatch agen terpadu untuk proses pengumpulan log Anda. Untuk informasi selengkapnya, lihat topik berikut:

- Mengirim log simpul ke CloudWatch Log terpadu (CloudWatch agen)
- Migrasikan koleksi log node Windows Server ke agen CloudWatch
- Mengumpulkan metrik dan log dari instans Amazon EC2 dan server lokal dengan CloudWatch agen di Panduan Pengguna Amazon. CloudWatch

Menggunakan CloudWatch Log, Anda dapat memantau data log secara real time, mencari dan memfilter data log dengan membuat satu atau beberapa filter metrik, dan mengarsipkan dan mengambil data historis saat Anda membutuhkannya. Untuk informasi selengkapnya tentang CloudWatch Log, lihat Panduan Pengguna CloudWatch Log Amazon.

Mengkonfigurasi agen untuk mengirim data log ke Amazon CloudWatch Logs memberikan manfaat berikut:

- Penyimpanan file log terpusat untuk semua file SSM Agent log.
- Akses lebih cepat ke file untuk menyelidiki kesalahan.
- Retensi berkas log tidak terbatas (dapat dikonfigurasi).

- Log dapat dipertahankan dan diakses terlepas dari status node.
- Akses ke CloudWatch fitur lain seperti metrik dan alarm.

Untuk informasi tentang Session Manager aktivitas pemantauan, lihat Mengaudit aktivitas sesi dan Mengaktifkan dan menonaktifkan pencatatan aktivitas sesi.

Migrasikan koleksi log node Windows Server ke agen CloudWatch

Jika Anda menggunakan SSM Agent pada Windows Server node yang didukung untuk mengirim file SSM Agent log ke Amazon CloudWatch Logs, Anda dapat menggunakan Systems Manager untuk bermigrasi dari SSM Agent ke CloudWatch agen sebagai alat pengumpulan log, dan memigrasikan pengaturan konfigurasi Anda.

CloudWatch Agen tidak didukung pada versi 32-bit. Windows Server

Untuk instans EC2 64-bitWindows Server, Anda dapat melakukan migrasi ke CloudWatch agen secara otomatis atau manual. Untuk server on-premise dan mesin virtual, proses harus dilakukan secara manual.

> **Note**
>
> Selama proses migrasi, data yang dikirim ke CloudWatch mungkin terputus atau digandakan. Metrik dan data log Anda akan direkam kembali secara akurat CloudWatch setelah migrasi selesai.

Kami merekomendasikan pengujian migrasi pada sejumlah node terbatas sebelum memigrasikan seluruh armada ke CloudWatch agen. Setelah migrasi, jika Anda lebih suka koleksi log denganSSM Agent, Anda dapat kembali menggunakannya sebagai gantinya.

> **Important**
>
> Dalam kasus berikut, Anda tidak akan dapat bermigrasi ke CloudWatch agen menggunakan langkah-langkah yang dijelaskan dalam topik ini:
>
> - Konfigurasi yang ada untuk SSM Agent menentukan beberapa Wilayah.
> - Konfigurasi yang ada untuk SSM Agent menentukan beberapa set kredensial akses/kunci rahasia.

Dalam kasus ini, perlu untuk mematikan pengumpulan log SSM Agent dan menginstal CloudWatch agen tanpa proses migrasi. Untuk informasi selengkapnya, lihat topik berikut di Panduan CloudWatch Pengguna Amazon:

- Instalasi CloudWatch agen
- Menginstal CloudWatch agen di server lokal

Sebelum Anda memulai

Sebelum memulai migrasi ke CloudWatch agen pengumpulan log, pastikan bahwa node tempat Anda akan melakukan migrasi memenuhi persyaratan berikut:

- OS adalah Server Windows versi 64-bit.
- SSM Agent2.2.93.0 atau yang lebih baru diinstal pada node.
- SSM Agentdikonfigurasi untuk pemantauan pada node.

Topik

Bermigrasi secara otomatis ke agen CloudWatch

Windows ServerHanya untuk instans EC2, Anda dapat menggunakan AWS Systems Manager konsol atau AWS Command Line Interface (AWS CLI) untuk secara otomatis bermigrasi ke CloudWatch agen sebagai alat pengumpulan log Anda.

Note

AWS Systems Managermendukung migrasi dari SSM Agent ke CloudWatch agen terpadu untuk mengumpulkan log dan metrik hanya pada versi Windows 64-bit. Untuk informasi tentang cara menyiapkan CloudWatch agen terpadu di sistem operasi lain, dan untuk informasi lengkap tentang penggunaan CloudWatch agen, lihat Mengumpulkan metrik dan log dari instans Amazon EC2 dan server lokal dengan CloudWatch agen di Panduan Pengguna Amazon. CloudWatch

Anda dapat menggunakan CloudWatch agen pada sistem operasi lain yang didukung, tetapi Anda tidak akan dapat menggunakan Systems Manager untuk melakukan migrasi alat.

Setelah migrasi berhasil, periksa hasil Anda CloudWatch untuk memastikan Anda menerima metrik, log, atau log peristiwa Windows yang Anda harapkan. Jika Anda puas dengan hasilnya, Anda dapat secara opsional Menyimpan pengaturan konfigurasi CloudWatch agen di Parameter Store. Jika migrasi tidak berhasil atau hasilnya tidak seperti yang diharapkan, Anda dapat mencoba Bergulir kembali ke koleksi log dengan SSM Agent.

Note

Jika Anda ingin memigrasikan file konfigurasi sumber yang menyertakan entri `{hostname}`, maka ketahuilah bahwa entri `{hostname}` dapat mengubah nilai bidang setelah migrasi selesai. Misalnya, katakan bahwa `"LogStream": "{hostname}"` entri berikut memetakan ke server bernama MyLogServer001.

```
{
"Id": "CloudWatchIISLogs",
"FullName":
 "AWS.EC2.Windows.CloudWatch.CloudWatchLogsOutput,AWS.EC2.Windows.CloudWatch",
"Parameters": {
"AccessKey": "",
"SecretKey": "",
"Region": "us-east-1",
"LogGroup": "Production-Windows-IIS",
"LogStream": "{hostname}"
    }
}
```

Setelah migrasi, entri ini memetakan ke domain, seperti ip-11-1-1-11.production. ExampleCompany.com. Untuk mempertahankan nilai hostname lokal, tentukan `{local_hostname}` dan bukan `{hostname}`.

Untuk secara otomatis bermigrasi ke CloudWatch agen (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command, lalu pilih Jalankan perintah.

3. Di daftar Dokumen perintah, pilih `AmazonCloudWatch-MigrateCloudWatchAgent`.
4. Untuk Status, pilih Enabled (Diaktifkan).
5. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

> **Tip**
>
> Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

6. Untuk Pengendalian rate:

   - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

   > **Note**
   >
   > Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

   - Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.
7. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan milik pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang

ditentukan berbedaAkun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

8. Di bagian Notifikasi SNS, jika Anda ingin notifikasi dikirim tentang status eksekusi perintah, pilih kotak centang Aktifkan notifikasi SNS.

   Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Run Command Amazon SNS, lihat. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

9. Pilih Jalankan.

Untuk secara otomatis bermigrasi ke CloudWatch agen () AWS CLI

- Jalankan perintah berikut.

```
aws ssm send-command --document-name AmazonCloudWatch-MigrateCloudWatchAgent --
targets Key=instanceids,Values=ID1,ID2,ID3
```

ID1, *ID2, dan ID3* mewakili ID node yang ingin Anda perbarui, seperti I-02573CAFCFExample.

Migrasi secara manual ke agen CloudWatch

Untuk Windows Server node lokal atau instans EC2Windows Server, ikuti langkah-langkah berikut untuk memigrasikan koleksi log secara manual ke agen Amazon. CloudWatch

> **Note**
>
> Jika Anda ingin memigrasikan file konfigurasi sumber yang menyertakan entri `{hostname}`, maka ketahuilah bahwa entri `{hostname}` dapat mengubah nilai bidang setelah migrasi selesai. Misalnya, katakan bahwa `"LogStream": "{hostname}"` entri berikut memetakan ke server bernama MyLogServer001.
>
> ```
> {
> "Id": "CloudWatchIISLogs",
> "FullName":
>  "AWS.EC2.Windows.CloudWatch.CloudWatchLogsOutput,AWS.EC2.Windows.CloudWatch",
> "Parameters": {
> ```

```
"AccessKey": "",
"SecretKey": "",
"Region": "us-east-1",
"LogGroup": "Production-Windows-IIS",
"LogStream": "{hostname}"
    }
}
```

Setelah migrasi, entri ini memetakan ke domain, seperti ip-11-1-1-11.production.ExampleCompany.com. Untuk mempertahankan nilai hostname lokal, tentukan `{local_hostname}` dan bukan `{hostname}`.

Satu: Untuk menginstal CloudWatch agen (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command, lalu pilih Jalankan perintah.
3. Di daftar Dokumen perintah, pilih `AWS-ConfigureAWSPackage`.
4. Untuk Tindakan, pilih`Install`.
5. Untuk Nama, masukkan **`AmazonCloudWatchAgent`**.
6. Untuk Versi, masukkan **`latest`** jika belum disediakan secara default.
7. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

   > **Tip**
   >
   > Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

8. Untuk Pengendalian rate:

   - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

> **Note**
>
> Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

- Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

9. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan milik pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbedaAkun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

10. Di bagian Notifikasi SNS, jika Anda ingin notifikasi dikirim tentang status eksekusi perintah, pilih kotak centang Aktifkan notifikasi SNS.

    Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Run Command Amazon SNS, lihat. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

11. Pilih Jalankan.

Dua: Untuk memperbarui format JSON data konfigurasi

- Untuk memperbarui pemformatan JSON dari pengaturan konfigurasi yang ada untuk CloudWatch agen, gunakan Run Command, kemampuanAWS Systems Manager, atau masuk ke node secara langsung dengan koneksi RDP untuk menjalankan PowerShell perintah Windows berikut pada node, satu per satu.

```
cd ${Env:ProgramFiles}\\Amazon\\AmazonCloudWatchAgent
```

```
.\\amazon-cloudwatch-agent-config-wizard.exe --isNonInteractiveWindowsMigration
```

{Env:ProgramFiles} mewakili lokasi di mana direktori Amazon yang berisi CloudWatch agen dapat ditemukan, biasanya`C:\Program Files`.

Tiga: Untuk mengkonfigurasi dan memulai CloudWatch agen (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command, lalu pilih Jalankan perintah.
3. Di daftar Dokumen perintah, pilih `AWS-RunPowerShellScript`.
4. Untuk Perintah, masukkan dua perintah berikut.

```
cd ${Env:ProgramFiles}\Amazon\AmazonCloudWatchAgent
```

```
.\amazon-cloudwatch-agent-ctl.ps1 -a fetch-config -m ec2 -c file:config.json -s
```

{Env:ProgramFiles} mewakili lokasi di mana direktori Amazon yang berisi CloudWatch agen dapat ditemukan, biasanya`C:\Program Files`.

5. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

> **Tip**
>
> Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

6. Untuk Pengendalian rate:

   - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

     > **Note**
     >
     > Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

   - Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

7. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

   > **Note**
   >
   > Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan milik pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbedaAkun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

8. Di bagian Notifikasi SNS, jika Anda ingin notifikasi dikirim tentang status eksekusi perintah, pilih kotak centang Aktifkan notifikasi SNS.

   Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Run Command Amazon SNS, lihat. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

9. Pilih Jalankan.

Empat: Untuk mematikan koleksi log di SSM Agent (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command, lalu pilih Jalankan perintah.
3. Di daftar Dokumen perintah, pilih `AWS-ConfigureCloudWatch`.
4. Untuk Status, pilih Dinonaktifkan.
5. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

   > **Tip**
   >
   > Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

6. Untuk Status, pilih`Disabled`.
7. Untuk Pengendalian rate:

   - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

     > **Note**
     >
     > Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

   - Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

8. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan milik pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbedaAkun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

9. Di bagian Notifikasi SNS, jika Anda ingin notifikasi dikirim tentang status eksekusi perintah, pilih kotak centang Aktifkan notifikasi SNS.

   Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Run Command Amazon SNS, lihat. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

10. Pilih Jalankan.

    Setelah menyelesaikan langkah-langkah ini, periksa log Anda CloudWatch untuk memverifikasi bahwa Anda menerima metrik, log, atau log peristiwa Windows yang Anda harapkan. Jika hasilnya memuaskan, Anda dapat secara opsional Menyimpan pengaturan konfigurasi CloudWatch agen di Parameter Store. Jika migrasi tidak berhasil atau hasilnya tidak seperti yang diharapkan, Anda dapat Bergulir kembali ke koleksi log dengan SSM Agent.

Menyimpan pengaturan konfigurasi CloudWatch agen di Parameter Store

Anda dapat menyimpan konten file konfigurasi CloudWatch agen diParameter Store. Dengan mempertahankan data konfigurasi ini dalam suatu parameter, beberapa node dapat memperoleh pengaturan konfigurasinya darinya, dan Anda menghindari keharusan membuat atau memperbarui file konfigurasi secara manual di node Anda. Misalnya, Anda dapat menggunakan Run Command untuk menulis isi parameter ke file konfigurasi pada beberapa node, atau menggunakanState Manager, kemampuanAWS Systems Manager, untuk membantu menghindari penyimpangan konfigurasi dalam pengaturan konfigurasi CloudWatch agen di seluruh armada node.

Saat menjalankan wizard konfigurasi CloudWatch agen, Anda dapat memilih untuk membiarkan wizard menyimpan pengaturan konfigurasi Anda sebagai parameter baruParameter Store. Untuk

informasi tentang menjalankan wizard konfigurasi CloudWatch agen, lihat Membuat file konfigurasi CloudWatch agen dengan wizard di Panduan CloudWatch Pengguna Amazon.

Jika Anda menjalankan wizard tetapi tidak memilih opsi untuk menyimpan pengaturan sebagai parameter, atau Anda membuat file konfigurasi CloudWatch agen secara manual, Anda dapat mengambil data untuk disimpan sebagai parameter pada node Anda di file berikut.

```
${Env:ProgramFiles}\Amazon\AmazonCloudWatchAgent\config.json
```

{Env:ProgramFiles} mewakili lokasi di mana direktori Amazon yang berisi CloudWatch agen dapat ditemukan, biasanyaC:\Program Files.

Sebaiknya simpan cadangan JSON di file ini di lokasi selain node itu sendiri.

Untuk informasi tentang pembuatan parameter, lihat Menandai parameter Systems Manager.

Untuk informasi selengkapnya tentang CloudWatch agen, lihat Mengumpulkan metrik dan log dari instans Amazon EC2 dan server lokal dengan CloudWatch agen di Panduan Pengguna Amazon. CloudWatch

Bergulir kembali ke koleksi log dengan SSM Agent

Jika Anda ingin kembali menggunakan SSM Agent untuk pengumpulan log, ikuti langkah-langkah ini.

Satu: Untuk mengambil data konfigurasi dari SSM Agent

1. Pada node tempat Anda ingin kembali mengumpulkan log denganSSM Agent, cari konten file SSM Agent konfigurasi. File JSON ini biasanya ditemukan di lokasi berikut:

   ```
   ${Env:ProgramFiles}\\Amazon\\SSM\\Plugins\\awsCloudWatch\
   \AWS.EC2.Windows.CloudWatch.json
   ```

   {Env:ProgramFiles} mewakili lokasi di mana Amazon direktori dapat ditemukan, biasanyaC:\Program Files.
2. Salin data ini ke dalam file teks untuk digunakan di langkah selanjutnya.

   Sebaiknya simpan cadangan JSON di lokasi selain node itu sendiri.

Dua: Untuk menghapus instalan CloudWatch agen (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.

2. Di panel navigasi, pilih Run Command, lalu pilih Jalankan perintah.
3. Di daftar Dokumen perintah, pilih `AWS-ConfigureAWSPackage`.
4. Untuk Tindakan, pilih Uninstall.
5. Untuk Nama, masukkan **`AmazonCloudWatchAgent`**.
6. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

   > **Tip**
   >
   > Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

7. Untuk Pengendalian rate:

   - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

     > **Note**
     >
     > Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

   - Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

8. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

   > **Note**
   >
   > Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang

ditetapkan ke instance, bukan milik pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbedaAkun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

9. Di bagian Notifikasi SNS, jika Anda ingin notifikasi dikirim tentang status eksekusi perintah, pilih kotak centang Aktifkan notifikasi SNS.

   Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Run Command Amazon SNS, lihat. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

10. Pilih Jalankan.

Tiga: Untuk mengaktifkan kembali koleksi log di SSM Agent (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command, lalu pilih Jalankan perintah.
3. Di daftar Dokumen perintah, pilih `AWS-ConfigureCloudWatch`.
4. Untuk Status, pilih`Enabled`.
5. Untuk Properti, tempel konten data konfigurasi lama yang Anda simpan ke file teks.
6. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

   > **Tip**
   >
   > Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

7. Untuk Pengendalian rate:

   - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

> **Note**
>
> Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

- Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

8. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan milik pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbedaAkun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

9. Di bagian Notifikasi SNS, jika Anda ingin notifikasi dikirim tentang status eksekusi perintah, pilih kotak centang Aktifkan notifikasi SNS.

   Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Run Command Amazon SNS, lihat. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

10. Pilih Jalankan.

MengirimSSM Agent log ke CloudWatch Log

AWS Systems ManagerAgen (SSM Agent) adalah peranti lunak Amazon yang berjalan pada instans EC2, peranti edge, server on-premise, dan mesin virtual (VM) yang dikonfigurasi untuk Systems Manager. SSM Agentmemproses permintaan permintaan permintaan permintaan permintaan dari layanan Systems Manager di cloud dan mengonfigurasi mesin Anda seperti yang ditentukan di permintaan. Untuk informasi selengkapnya tentang SSM Agent, lihat Bekerja dengan SSM Agent.

Selain itu, dengan menggunakan langkah berikut, Anda dapatSSM Agent mengonfigurasi data log data log data CloudWatch log.

Sebelum Anda memulai

Membuat grup log log log log log CloudWatch log di Logs. Untuk informasi selengkapnya, lihat Memulai CloudWatch Log di Panduan Pengguna Amazon CloudWatch Logs.

Untuk mengkonfigurasiSSM Agent untuk mengirim log ke CloudWatch

1. Masuklah log log log log log log log log log masuk ke node dan temukan file berikut:

   Linux

   Pada kebanyakan jenis node Linux:`/etc/amazon/ssm/seelog.xml.template`.

   PadaUbuntu Server 20.10 STR & 20.04 LTS:`/snap/amazon-ssm-agent/current/seelog.xml.template`

   macOS

   `/opt/aws/ssm/seelog.xml.template`

   Windows

   `%ProgramFiles%\Amazon\SSM\seelog.xml.template`
2. Mengubah nama file dari `seelog.xml.template` ke `seelog.xml`

   > **Note**
   >
   > PadaUbuntu Server 20.10 STR & 20.04 LTS, dan 16.04 LTS, file tersebut`seelog.xml` harus dibuat di direktori`/etc/amazon/ssm/`. Anda dapat membuat direktori dan file ini dengan menjalankan perintah berikut.

```
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. Buka file `seelog.xml` di editor teks, dan temukan bagian berikut:

   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. Edit file, dan tambahkan elemen nama kustom setelah tanda </filter> penutupan. Di contoh berikut, nama kustom telah ditentukan sebagai `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. Simpan perubahan Anda, dan kemudian mulai ulangSSM Agent atau node.
6. Buka CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/.
7. Di panel navigasi, pilih Grup log, lalu pilih nama grup log Anda.

> **Tip**
>
> Aliran log log log log log logSSM Agent log log log log log log log log log log log log log log log log log log log log log log log log log log log

Memantau peristiwa permintaan perubahan

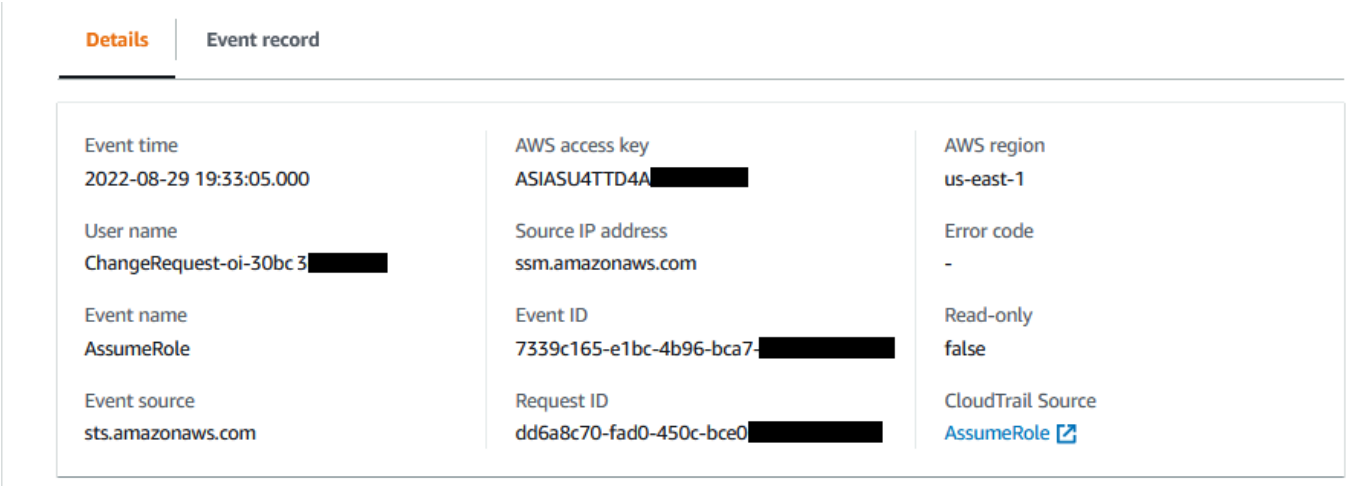
Setelah mengaktifkan integrasi denganAWS CloudTrail Lake dan membuat penyimpanan data peristiwa, Anda dapat melihat detail yang dapat diaudit tentang permintaan perubahan yang dijalankan di akun atau organisasi Anda. Ini termasuk rincian seperti berikut ini:

- Identitas pengguna yang memulai permintaan perubahan
- DiWilayah AWS mana perubahan itu dibuat.
- Alamat sumber sumber sumber sumber untuk permintaan sumber sumber sumber sumber sumber sumber sumber sumber sumber sumber
- KunciAWS akses yang digunakan untuk permintaan
- Tindakan API yang dijalankan untuk permintaan perubahan
- Parameter permintaan yang disertakan untuk tindakan tersebut
- Sumber daya diperbarui selama proses

Berikut ini adalah contoh detail peristiwa yang dapat Anda lihat untuk permintaan perubahan setelah membuat penyimpanan data acara diAWS CloudTrail Danau.

Details

Gambar berikut menunjukkan informasi tingkat tinggi tentang permintaan perubahan yang tersedia di tab Detail. Rincian ini mencakup informasi seperti waktu operasi permintaan perubahan dimulai, ID pengguna yang memulai permintaan perubahan, dampakWilayah AWS, dan ID peristiwa serta ID permintaan yang terkait dengan permintaan.



Event record

Gambar berikut menunjukkan struktur konten JSON yang disediakan olehCloudTrail Lake untuk peristiwa permintaan perubahan. Data ini disediakan di Rekaman peristiwa tab dalam permintaan perubahan.

Important

Jika Anda menggunakanChange Manager untuk organisasi, Anda dapat menyelesaikan prosedur berikut saat masuk ke akun manajemen atau akun administrator yang didelegasikanChange Manager.

Namun, untuk menggunakan akun administrator yang didelegasikan untuk menyelesaikan langkah-langkah ini, akun administrator yang didelegasikan yang sama harus ditentukan untuk keduanyaCloudTrail danChange Manager.

Saat masuk ke akun manajemenChange Manager, Anda dapat menambahkan atau mengubah akun administrator yang didelegasikanCloudTrail di halamanCloudTrail Pengaturan. Ini harus dilakukan sebelum akun administrator yang didelegasikan dapat membuat penyimpanan data peristiwa untuk digunakan oleh seluruh organisasi.

Untuk mengaktifkan pelacakan acaraCloudTrail Danau dariChange Manager

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Change Manager.

-atau-

JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Change Manager.

3. Pilih tab Permintaan.
4. Pilih permintaan perubahan yang ada, lalu pilih tab Peristiwa terkait.
5. Pilih AktifkanCloudTrail Danau.
6. Ikuti langkah-langkah di Membuat penyimpanan data peristiwa untukCloudTrail acara di PanduanAWS CloudTrail Pengguna.

   Untuk memastikan bahwa data peristiwa untuk permintaan perubahan Anda disimpan, buat pilihan berikut saat Anda menyelesaikan prosedur:

   - Untuk Jenis acara, biarkan AWSacara dan CloudTrailacara default dipilih.
   - Jika Anda menggunakanChange Manager organisasi, pilih Aktifkan untuk semua akun di organisasi saya.
   - Untuk acara Manajemen, jangan kosongkan kotak centang Tulis.

   Opsi lain yang Anda pilih saat membuat penyimpanan data peristiwa tidak memengaruhi penyimpanan data peristiwa untuk permintaan perubahan Anda.

Pemantauan Otomatisasi Anda

Metrik adalah konsep dasar di Amazon CloudWatch. Metrik mewakili serangkaian titik data yang diurutkan waktu yang diurutkan waktu CloudWatch. Pikirkan metrik sebagai variabel untuk memantau dan titik data sebagai representasi nilai-nilai variabel tersebut dari waktu ke waktu.

Otomatisasi adalah kemampuan AWS Systems Manager. Systems Manager menerbitkan metrik tentang penggunaan Otomasi ke CloudWatch. Ini memungkinkan Anda untuk mengatur alarm berdasarkan metrik tersebut.

Melihat metrik Otomasi di CloudWatch konsol

1. Buka CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/.
2. Di panel navigasi, pilih Metrik.

3. Pilih SSM.
4. Pada tab Metrik, pilih Penggunaan, lalu pilih BerdasarkanAWS Sumber Daya.
5. Di kotak pencarian di dekat daftar metrik, masukkan SSM.

Untuk melihat metrik Otomatisasi menggunakanAWS CLI

Buka wantian perintah, dan gunakan perintah berikut.

```
aws cloudwatch list-metrics \
    --namespace "AWS/Usage"
```

Metrik Otomatisasi

Systems Manager mengirimkan metrik Otomatisasi berikut ke CloudWatch.

Metrik	Deskripsi
`ConcurrentAutomationUsage`	Jumlah otomatisasi berjalan pada saat yang sama di saat iniAkun AWS danWilayah AWS.
`QueuedAutomationUsage`	Jumlah otomatisasi saat ini antri yang belum dimulai dan memiliki status`Pending`.

Untuk informasi tentang bekerja dengan CloudWatch metrik, lihat topik-topik berikut di Panduan CloudWatch Pengguna Amazon:

- Metrik
- Menggunakan CloudWatch metrik Amazon
- Menggunakan CloudWatch alarm Amazon

PemantauanRun CommandMetrik menggunakan Amazon CloudWatch

Metrikadalah konsep dasar di Amazon CloudWatch. Sebuah metrik merupakan serangkaian titik data yang diurutkan waktu yang dipublikasikan CloudWatch. Pikirkan metrik sebagai variabel untuk memantau, dan titik data sebagai representasi nilai-nilai variabel tersebut dari waktu ke waktu.

AWS Systems Managermenerbitkan metrik tentang statusRun Commandperintah untuk CloudWatch, yang memungkinkan Anda untuk mengatur alarm berdasarkan metrik tersebut.Run Commandadalah kemampuanAWS Systems Manager. Statistik ini dicatat dalam jangka waktu yang lama sehingga Anda dapat mengakses informasi historis dan mendapatkan perspektif yang lebih baik tentang tingkat keberhasilan perintah yang dijalankan di Akun AWS Anda.

Nilai status terminal untuk perintah yang dapat Anda lacak metriknya mencakup `Success`, `Failed`, dan `Delivery Timed Out`. Misalnya, untuk dokumen Perintah SSM yang diatur untuk berjalan setiap jam, Anda dapat mengonfigurasi alarm untuk memberitahu Anda saat status `Success` tidak dilaporkan pada jam-jam tersebut. Untuk informasi lebih lanjut tentang nilai status perintah, lihat Memahami status perintah.

Untuk melihat metrik di CloudWatch konsol

1. Buka CloudWatch konsol dihttps://console.aws.amazon.com/cloudwatch/.
2. Di panel navigasi, pilih Metrik.
3. DiAlarm olehAWSlayanandaerah, untukLayanan, pilihSSMRun Command.

Untuk melihat metrik menggunakan AWS CLI

Buka wantian perintah, dan gunakan perintah berikut.

```
aws cloudwatch list-metrics --namespace "AWS/SSM-RunCommand"
```

Untuk mencantumkan semua metrik yang tersedia, gunakan perintah berikut.

```
aws cloudwatch list-metrics
```

Systems ManagerRun CommandMetrik dan dimensi

Systems Manager mengirimkanRun Commandmetrik perintah ke CloudWatch Satu kali setiap menit.

Systems Manager mengirimkan metrik perintah berikut ke CloudWatch.

> **Note**
>
> Metrik ini menggunakan `Count` sebagai unit, sehingga `Sum` dan `SampleCount` merupakan statistik yang paling berguna.

Metrik	Deskripsi
`CommandsDeliveryTimedOut`	Jumlah perintah yang memiliki status terminal `Delivery Timed Out`.
`CommandsFailed`	Jumlah perintah yang memiliki status terminal `Failed`.
`CommandsSucceeded`	Jumlah perintah yang memiliki status terminal `Success`.

Untuk informasi lebih lanjut tentang bekerja dengan CloudWatch Metrik, lihat topik berikut diAmazon CloudWatch Panduan Pengguna:

- Metrik
- Menggunakan Amazon CloudWatch metrik
- Menggunakan Amazon CloudWatch alarm

Pencatatan panggilan AWS Systems Manager API dengan AWS CloudTrail

AWS Systems Manager terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau Layanan AWS. CloudTrail menangkap panggilan API untuk Systems Manager sebagai peristiwa. Panggilan yang diambil termasuk panggilan dari konsol Systems Manager dan panggilan kode ke operasi Systems Manager API. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat untuk Systems Manager, alamat IP dari mana permintaan itu dibuat, kapan dibuat, dan detail tambahan.

Setiap acara atau entri log berisi informasi yang membantu Anda menentukan siapa yang membuat permintaan.

- Pengguna root akun AWS
- Kredensi keamanan sementara dari peran AWS Identity and Access Management (IAM) atau pengguna federasi.
- Kredensi keamanan jangka panjang dari pengguna IAM.

- Permintaan dibuat atas nama pengguna IAM Identity Center.
- Lain Layanan AWS.

Untuk informasi lain, lihat Elemen userIdentity CloudTrail.

CloudTrail aktif di Anda Akun AWS ketika Anda membuat akun dan Anda secara otomatis memiliki akses ke riwayat CloudTrail Acara. Riwayat CloudTrail Acara menyediakan catatan yang dapat dilihat, dapat dicari, dapat diunduh, dan tidak dapat diubah dari 90 hari terakhir dari peristiwa manajemen yang direkam dalam file. Wilayah AWS Untuk informasi selengkapnya, lihat Bekerja dengan riwayat CloudTrail Acara di Panduan AWS CloudTrail Pengguna. Tidak ada CloudTrail biaya untuk melihat riwayat Acara.

Untuk catatan acara yang sedang berlangsung dalam 90 hari Akun AWS terakhir Anda, buat jejak atau penyimpanan data acara CloudTrailDanau.

CloudTrail jalan setapak

Jejak memungkinkan CloudTrail untuk mengirimkan file log ke bucket Amazon S3. Semua jalur yang dibuat menggunakan AWS Management Console Multi-region. Anda dapat membuat jalur Single-region atau Multi-region dengan menggunakan. AWS CLI Membuat jejak Multi-wilayah disarankan karena Anda menangkap aktivitas Wilayah AWS di semua akun Anda. Jika Anda membuat jejak wilayah Tunggal, Anda hanya dapat melihat peristiwa yang dicatat di jejak. Wilayah AWS Untuk informasi selengkapnya tentang jejak, lihat Membuat jejak untuk Anda Akun AWS dan Membuat jejak untuk organisasi di Panduan AWS CloudTrail Pengguna.

Anda dapat mengirimkan satu salinan acara manajemen yang sedang berlangsung ke bucket Amazon S3 Anda tanpa biaya CloudTrail dengan membuat jejak, namun, ada biaya penyimpanan Amazon S3. Untuk informasi selengkapnya tentang CloudTrail harga, lihat AWS CloudTrail Harga. Untuk informasi tentang harga Amazon S3, lihat Harga Amazon S3.

CloudTrail Menyimpan data acara danau

CloudTrail Lake memungkinkan Anda menjalankan kueri berbasis SQL pada acara Anda. CloudTrail Lake mengonversi peristiwa yang ada dalam format JSON berbasis baris ke format Apache ORC. ORC adalah format penyimpanan kolumnar yang dioptimalkan untuk pengambilan data dengan cepat. Peristiwa digabungkan ke dalam penyimpanan data peristiwa, yang merupakan kumpulan peristiwa yang tidak dapat diubah berdasarkan kriteria yang Anda pilih dengan menerapkan pemilih acara tingkat lanjut. Penyeleksi yang Anda terapkan ke penyimpanan data acara mengontrol peristiwa mana yang bertahan dan tersedia untuk Anda

kueri. Untuk informasi lebih lanjut tentang CloudTrail Danau, lihat Bekerja dengan AWS CloudTrail Danau di Panduan AWS CloudTrail Pengguna.

CloudTrail Penyimpanan data acara danau dan kueri menimbulkan biaya. Saat Anda membuat penyimpanan data acara, Anda memilih opsi harga yang ingin Anda gunakan untuk penyimpanan data acara. Opsi penetapan harga menentukan biaya untuk menelan dan menyimpan peristiwa, dan periode retensi default dan maksimum untuk penyimpanan data acara. Untuk informasi selengkapnya tentang CloudTrail harga, lihat AWS CloudTrail Harga.

Peristiwa data Systems Manager di CloudTrail

Peristiwa data memberikan informasi tentang operasi sumber daya yang dilakukan pada atau di sumber daya (misalnya, membuat atau membuka saluran kontrol). Ini juga dikenal sebagai operasi bidang data. Peristiwa data seringkali merupakan aktivitas volume tinggi. Secara default, CloudTrail tidak mencatat peristiwa data. Riwayat CloudTrail peristiwa tidak merekam peristiwa data.

Biaya tambahan berlaku untuk peristiwa data. Untuk informasi selengkapnya tentang CloudTrail harga, lihat AWS CloudTrail Harga.

Anda dapat mencatat peristiwa data untuk jenis sumber daya Systems Manager menggunakan CloudTrail konsol AWS CLI, atau operasi CloudTrail API. Untuk informasi selengkapnya tentang cara mencatat peristiwa data, lihat Mencatat peristiwa data dengan AWS Management Console dan Logging peristiwa data dengan AWS Command Line Interface di Panduan AWS CloudTrail Pengguna.

Tabel berikut mencantumkan jenis sumber daya Systems Manager yang dapat Anda log peristiwa data. Kolom tipe peristiwa data (konsol) menunjukkan nilai yang akan dipilih dari daftar tipe peristiwa Data di CloudTrail konsol. Kolom nilai resources.type menunjukkan **`resources.type`** nilai, yang akan Anda tentukan saat mengonfigurasi penyeleksi acara lanjutan menggunakan API atau. AWS CLI CloudTrail CloudTrailKolom API Data yang dicatat ke menampilkan panggilan API yang dicatat CloudTrail untuk jenis sumber daya.

Jenis peristiwa data (konsol)	nilai resources.type	API data masuk CloudTrail
Systems Manager	`AWS::SSMMessages::ControlChannel`	• `CreateControlChannel` • `OpenControlChannel`

Jenis peristiwa data (konsol)	nilai resources.type	API data masuk CloudTrail
		Untuk informasi selengkapnya tentang operasi ini, lihat Tindakan yang ditentukan oleh Amazon Message Gateway Service di Referensi Otorisasi Layanan.
Node terkelola Systems Manager	`AWS::SSM::ManagedNode`	• `RequestManagedInstanceRoleToken` — Peristiwa ini dihasilkan ketika Agen Systems Manager (Agen SSM) yang berjalan pada node yang dikelola oleh Systems Manager meminta kredensil dari layanan kredensi Systems Manager.

Anda dapat mengonfigurasi pemilih acara lanjutan untuk memfilter pada `eventName`,`readOnly`, dan `resources.ARN` bidang untuk mencatat hanya peristiwa yang penting bagi Anda. Untuk informasi selengkapnya tentang bidang ini, lihat AdvancedFieldSelectordi Referensi AWS CloudTrail API.

Acara manajemen Systems Manager di CloudTrail

Acara manajemen memberikan informasi tentang operasi manajemen yang dilakukan pada sumber daya di Anda Akun AWS. Ini juga dikenal sebagai operasi pesawat kontrol. Secara default, CloudTrail mencatat peristiwa manajemen.

Systems Manager mencatat semua operasi bidang kontrol ke CloudTrail sebagai peristiwa manajemen. Operasi API Systems Manager didokumentasikan dalam Referensi AWS Systems Manager API. Misalnya, panggilan ke`CreateMaintenanceWindows`, `PutInventorySendCommand`, dan `StartSession` tindakan menghasilkan entri dalam file CloudTrail log. Untuk contoh pengaturan CloudTrail untuk memantau panggilan API Systems Manager, lihatMemantau aktivitas sesi sesi sesi menggunakan Amazon EventBridge (konsol) .

Contoh acara Systems Manager

Peristiwa mewakili permintaan tunggal dari sumber manapun dan mencakup informasi tentang operasi API yang diminta, tanggal dan waktu operasi, parameter permintaan, dan sebagainya. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, sehingga peristiwa tidak muncul dalam urutan tertentu.

Contoh:

- Contoh acara manajemen
- Contoh peristiwa data

Contoh acara manajemen

Contoh 1: **`DeleteDocument`**

Contoh berikut menunjukkan CloudTrail peristiwa yang menunjukkan `DeleteDocument` operasi pada dokumen yang disebutkan `example-Document` di Wilayah Timur AS (Ohio) (us-timur-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"
}
```


Contoh 2: **`StartConnection`**

Contoh berikut menunjukkan CloudTrail peristiwa untuk pengguna yang memulai koneksi RDP menggunakan Fleet Manager di Wilayah AS Timur (Ohio) (us-timur-2). Tindakan API yang mendasarinya adalah`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"
}
```


Contoh peristiwa data

Contoh 1: **`CreateControlChannel`**

Contoh berikut menunjukkan CloudTrail peristiwa yang menunjukkan `CreateControlChannel` operasi.

```
{
  "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"
}
```

Contoh 2: **`RequestManagedInstanceRoleToken`**

Contoh berikut menunjukkan CloudTrail peristiwa yang menunjukkan `RequestManagedInstanceRoleToken` operasi.

```
{
    "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"
}
```


Untuk informasi tentang konten CloudTrail rekaman, lihat konten CloudTrail rekaman di Panduan AWS CloudTrail Pengguna.

Pencatatan output tindakan Otomatisasi dengan CloudWatch Logs

Otomatisasi, sebuah kemampuanAWS Systems Manager, berintegrasi dengan Amazon CloudWatch Logs. Anda dapat mengirim output dari tindakan `aws:executeScript` di runbook Anda ke grup log yang Anda tentukan. Systems Manager tidak membuat grup log atau aliran log apa pun untuk dokumen yang tidak menggunakan tindakan `aws:executeScript`. Jika dokumen tidak digunakan`aws:executeScript`, output yang CloudWatch dikirimkan ke tindakan tersebut. Anda dapat menggunakan output`aws:executeScript` tindakan yang disimpan di grup CloudWatch log Anda untuk tujuan debugging dan pemecahan masalah. Jika Anda memilih grup log yang dienkripsi, output tindakan `aws:executeScript` juga dienkripsi. Pencatatan output dari tindakan `aws:executeScript` adalah pengaturan tingkat akun.

Untuk mengirim output tindakan ke CloudWatch Log untuk runbook milik Amazon, pengguna atau peran yang menjalankan otomatisasi harus memiliki izin untuk operasi berikut:

- `logs:CreateLogGroup`
- `logs:CreateLogStream`
- `logs:DescribeLogGroups`
- `logs:DescribeLogStreams`
- `logs:PutLogEvents`

Untuk runbook yang Anda miliki, izin yang sama harus ditambahkan ke peran layanan IAM (atau AssumeRole) yang Anda gunakan untuk menjalankan runbook.

Untuk mengirim output tindakan ke CloudWatch Logs (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Pada panel navigasi, pilih Otomatisasi.
3. Pilih tab Preferensi, dan kemudian pilih Edit.
4. Pilih kotak centang di samping Mengirim output ke CloudWatch Logs.
5. (Disarankan) Pilih kotak centang di samping Mengenkripsi data log. Dengan pengaktifan pilihan ini, data log dienkripsi menggunakan kunci enkripsi sisi server yang ditentukan untuk grup log. Jika Anda tidak ingin mengenkripsi data log yang dikirimkan ke CloudWatch Logs, kosongkan kotak centang. Kosongkan kotak centang jika enkripsi tidak diizinkan pada grup log.
6. Untuk GrupCloudWatch log, untuk menentukan grup CloudWatch log yang ada di yang merupakan tujuan dari output tindakanAkun AWS yang ingin Anda kirimkan, pilih salah satu hal berikut:
   - Mengirimkan output ke grup log default – Jika grup log default tidak ada (`/aws/ssm/automation/executeScript`), Otomatisasi membuatnya untuk Anda.
   - Memilih dari daftar grup log – Pilih grup log yang telah dibuat di akun Anda untuk menyimpan output tindakan.
   - Masukkan nama grup log - Masukkan nama grup log di kotak teks yang telah dibuat di akun Anda untuk menyimpan output tindakan.
7. Pilih Save (Simpan).

Untuk mengirim output tindakan ke CloudWatch Logs (baris perintah)

1. Buka alat baris perintah pilihan Anda dan jalankan perintah berikut untuk memperbarui tujuan output tindakan.

   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"
   ```

   Tidak ada output jika perintah berhasil.

2. Jalankan perintah berikut untuk menentukan grup log yang merupakan tujuan dari output tindakan yang ingin Anda kirimkan.

   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"
```

Tidak ada output jika perintah berhasil.

3. Jalankan perintah berikut untuk melihat pengaturan layanan saat ini untuk preferensi pencatatan tindakan Otomatisasi di Akun AWS dan Wilayah AWS saat ini.

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"
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
{
    "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"
    }
}
```

MengonfigurasiCloudWatch Log Amazon untukRun Command

Ketika Anda mengirimkan perintah dengan menggunakanRun Command, sebuah kemampuanAWS Systems Manager, Anda dapat menentukan ke mana Anda ingin mengirimkan perintah output. Secara default, Systems Manager menampilkan 48.000 karakter pertama dari output perintah saja. Jika Anda ingin melihat detail lengkap dari output perintah, Anda dapat menentukan bucket Amazon Simple Storage Service (Amazon S3). Atau Anda dapat menentukan AmazonCloudWatch Logs. Jika Anda menentukanCloudWatch Logs,Run Command secara berkala mengirimkan semua output perintah dan log kesalahan keCloudWatch Logs. Anda dapat memantau log output mendekati waktu nyata, mencari frasa, nilai, atau pola tertentu, dan membuat alarm berdasarkan pencarian.

Jika Anda mengonfigurasi node terkelola untuk menggunakan Kebijakan terkelolaAWS Identity and Access Management (IAM)`AmazonSSMManagedInstanceCore` dan`CloudWatchAgentServerPolicy`, maka node Anda tidak memerlukan konfigurasi tambahan untuk mengirimkan output keCloudWatch Logs. Pilih pilihan ini jika pengiriman perintah dari konsol, atau tambahkan bagian `cloud-watch-output-config` dan parameter `CloudWatchOutputEnabled` jika menggunakan AWS Command Line Interface (AWS CLI), AWS Tools for Windows PowerShell, atau operasi API. Bagian `cloud-watch-output-config` dan parameter `CloudWatchOutputEnabled` dijelaskan secara lebih detail nanti di topik ini.

Untuk informasi tentang penambahan kebijakan ke profil instans untuk instans EC2, lihat Mengonfigurasi izin instans untuk Systems Manager. Untuk informasi tentang penambahan kebijakan ke peran layanan untuk server on-premise dan mesin virtual yang ingin Anda gunakan sebagai node terkelola, lihat Membuat peran layanan IAM untuk lingkungan hibrida.

Jika Anda menggunakan kebijakan kustom pada node, perbarui kebijakan pada setiap node untuk mengizinkan Systems Manager mengirimkan output danCloudWatch log ke Logs. Tambahkan objek kebijakan berikut ke kebijakan kustom Anda. Untuk informasi lebih lanjut tentang pembaruan kebijakan IAM, lihat Pengeditan kebijakan IAM di Panduan Pengguna 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/*"
},
```

PenentuanCloudWatch Logs saat Anda mengirimkan perintah Logs

Untuk menentukanCloudWatch Logs sebagai output saat Anda mengirimkan perintah dari, pilih Output di bagian Pilihan CloudWatchoutput saat Anda mengirimkan perintah dariAWS Management Console, pilih Output di bagian Pilihan output. Secara opsional, Anda dapat menentukan nama grupCloudWatch Logs yang merupakan tujuan dari output perintah yang ingin Anda kirimkan. Jika Anda tidak menentukan nama grup, Systems Manager secara otomatis membuat grup log untuk Anda. Grup log menggunakan format penamaan berikut: `/aws/ssm/`*`SystemsManagerDocumentName`*

Jika Anda menjalankan perintah dengan menggunakan AWS CLI, tentukan bagian `cloud-watch-output-config` di perintah Anda. Bagian ini memungkinkan Anda untuk menentukan parameter `CloudWatchOutputEnabled`, dan secara opsional, parameter `CloudWatchLogGroupName`. Inilah contohnya.

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}'
```

Melihat output perintah diCloudWatch Logs Logs

Segera setelah perintah mulai berjalan, Systems Manager mengirimkan output keCloudWatch Logs mendekati waktu nyata. Output diCloudWatch Logs menggunakan format berikut:

CommandID/*InstanceID*/*PluginID*/stdout

CommandID/*InstanceID*/*PluginID*/stderr

Output dari eksekusi diunggah setiap 30 detik atau saat buffer melebihi 200 KB, yang mana yang terjadi terlebih dahulu.

> **Note**
>
> Aliran log hanya dibuat saat data output tersedia. Misalnya, jika tidak ada data kesalahan untuk eksekusi, aliran stderr tidak dibuat.

Berikut adalah contoh dari output perintah seperti yang ditampilkan diCloudWatch Logs.

```
Group - /aws/ssm/AWS-RunShellScript
Streams –
1234-567-8910/i-abcd-efg-hijk/AWS-RunPowerShellScript/stdout
24/1234-567-8910/i-abcd-efg-hijk/AWS-RunPowerShellScript/stderr
```

Pemantauan peristiwa Systems Manager dengan Amazon EventBridge

Amazon EventBridge adalah layanan bus peristiwa nirkabel yang memungkinkan Anda untuk menghubungkan aplikasi Anda dengan data dari berbagai sumber. EventBridge mengirimkan pengaliran data waktu nyata dari aplikasi Anda sendiri, aplikasi software-as-a-service (SaaS),Layanan AWS dan merutekan data tersebut ke target sepertiAWS Lambda. Anda dapat mengatur aturan perutean untuk menentukan di mana akan mengirim data Anda untuk membangun arsitektur aplikasi yang bereaksi secara waktu nyata ke semua sumber data Anda. EventBridge memungkinkan Anda untuk membangun arsitektur yang digerakkan peristiwa, yang digabungkan longgar dan terdistribusi.

EventBridge sebelumnya disebut Amazon CloudWatch Events. EventBridge menyertakan fitur baru yang memungkinkan Anda untuk menerima peristiwa dari mitra SaaS dan aplikasi Anda sendiri. Pengguna CloudWatch Peristiwa yang ada dapat mengakses bus, aturan, dan peristiwa default yang ada di EventBridge konsol baru dan di konsol CloudWatch Peristiwa. EventBridge menggunakan API CloudWatch Peristiwa yang sama, sehingga semua penggunaan API CloudWatch Peristiwa yang ada tetap sama.

EventBridge dapat menambahkan peristiwa dari puluhanLayanan AWS aturan Anda, dan target dari lebih dari 20Layanan AWS.

EventBridge menyediakan dukungan untukAWS Systems Manager peristiwa dan target Systems Manager.

Jenis peristiwa Systems Manager yang didukung

Di antara banyak jenis peristiwa Systems Manager yang EventBridge dapat dideteksi adalah:

- Jendela pemeliharaan yang dimatikan.
- Penyelesaian alur kerja Otomatisasi yang berhasil. Otomatisasi adalah kemampuan AWS Systems Manager.
- Node terkelola yang berada di luar kepatuhan patch.
- Nilai parameter yang diperbarui.

EventBridge mendukung peristiwa dariAWS Systems Manager kemampuan berikut:

- Otomatisasi (Peristiwa dipancarkan atas dasar upaya terbaik.)

- Change Calendar(Peristiwa dipancarkan atas dasar upaya terbaik.)
- Kepatuhan
- Inventaris (Peristiwa dipancarkan atas dasar upaya terbaik.)
- Maintenance Windows(Peristiwa dipancarkan atas dasar upaya terbaik.)
- Parameter Store(Peristiwa dipancarkan atas dasar upaya terbaik.)
- Run Command(Peristiwa dipancarkan atas dasar upaya terbaik.)
- State Manager(Peristiwa dipancarkan atas dasar upaya terbaik.)

Untuk detail lengkap tentang jenis peristiwa Systems Manager yang didukung, lihatReferensi: Pola dan jenis EventBridge acara Amazon untuk Systems Manager danContoh EventBridge acara Amazon untuk Systems Manager.

Jenis target Systems Manager yang didukung

EventBridge mendukung tiga kemampuan Systems Manager berikut sebagai target aturan peristiwa:

- Menjalankan alur kerja Otomatisasi
- Menjalankan dokumenRun Command Command (Peristiwa dipancarkan atas dasar upaya terbaik.)
- MenciptakanOpsCenterOpsItem

Untuk cara yang disarankan agar Anda dapat menggunakan target ini, lihat Skenario: Target Systems Manager di Amazon EventBridge aturan.

Untuk informasi lebih lanjut tentang cara memulai EventBridge dan menyiapkan aturan, lihat Memulai Amazon EventBridge di Panduan EventBridge Pengguna Amazon. Untuk informasi selengkapnya tentang bekerja sama EventBridge, lihat Panduan EventBridge Pengguna Amazon.

Topik

- EventBridge Pengonfigurasan peristiwa Systems Manager
- Contoh EventBridge acara Amazon untuk Systems Manager
- Skenario: Target Systems Manager di Amazon EventBridge aturan

EventBridge Pengonfigurasan peristiwa Systems Manager

Anda dapat menggunakan Amazon EventBridge untuk melakukan peristiwa target saatAWS Systems Manager status yang didukung berubah, tahapan berubah, atau kondisi lainnya terjadi. Anda dapat membuat aturan yang berjalan setiap kali terjadi transisi tahapan atau status, atau saat ada transisi ke satu atau beberapa tahapan yang penting.

Prosedur berikut menyediakan langkah umum untuk membuat EventBridge aturan yang terlibat saat peristiwa tertentu dipancarkan oleh Systems Manager. Untuk daftar prosedur di panduan pengguna yang membahas skenario tertentu ini, lihat Info lebih lanjut di bagian akhir topik ini.

> **Note**
>
> Ketika layanan di Akun AWS Anda memancarkan peristiwa, layanan tersebut akan selalu menuju ke bus peristiwa default akun Anda. Untuk menulis aturan yang menanggapi peristiwa dariLayanan AWS akun Anda, kaitkan dengan bus peristiwa default. Anda dapat membuat aturan pada bus peristiwa kustom yang mencari peristiwa dariLayanan AWS, tetapi aturan ini hanya terlibat saat Anda menerima peristiwa tersebut dari akun lainnya melalui pengiriman peristiwa lintas akun. Untuk informasi selengkapnya, lihat Mengirim dan menerima EventBridge peristiwa AmazonAkun AWS di antara Panduan EventBridge Pengguna Amazon.

EventBridge Untuk mengonfigurasi peristiwa Systems Manager

1. Buka EventBridge konsol Amazon di https://console.aws.amazon.com/events/.
2. Di panel navigasi, pilih Aturan.
3. Pilih Buat aturan.
4. Masukkan nama dan deskripsi untuk aturan.

   Aturan tidak boleh memiliki nama yang sama dengan aturan lain di Wilayah AWS yang sama dan di bus peristiwa yang sama.
5. Untuk bus peristiwa, pilih bus peristiwa yang ingin Anda kaitkan dengan aturan ini. Jika Anda ingin aturan ini menanggapi peristiwa yang berasal dari akun AndaAkun AWS, pilih default. Saat kejadianLayanan AWS di akun Anda menghasilkan kejadian, peristiwa tersebut akan selalu masuk ke bus kejadian default akun Anda.
6. Untuk jenis Aturan, pilih Aturan dengan pola peristiwa.

7. Pilih Selanjutnya.
8. Untuk Sumber acara, pilih AWSacara atau acara EventBridge mitra.
9. Di bagian Pola acara, pilih Bentuk pola acara.
10. Untuk sumber Event, pilih AWSlayanan.
11. Untuk AWSlayanan, pilih Systems Manager.
12. Untuk Jenis peristiwa, lakukan salah satu hal berikut:

    - Pilih Semua Peristiwa.

      Jika Anda memilih Semua peristiwa, semua peristiwa yang dipancarkan oleh Systems Manager ini akan sesuai dengan aturan. Ketahuilah bahwa pilihan ini dapat mengakibatkan banyak tindakan target peristiwa.
    - Pilih jenis peristiwa Systems Manager untuk digunakan bagi aturan ini. EventBridge mendukung peristiwa dariAWS Systems Manager kemampuan berikut:
      - Otomatisasi
      - Change Calendar
      - Kepatuhan
      - Inventaris
      - Maintenance Windows
      - Parameter Store
      - Run Command
      - State Manager

    > **Note**
    >
    > Untuk tindakan Systems Manager yang tidak didukung oleh EventBridge, Anda dapat memilih panggilanAWS API melalui CloudTrail untuk membuat aturan peristiwa yang didasarkan pada panggilan API, yang dicatat oleh CloudTrail. Sebagai contoh, lihat Memantau aktivitas sesi sesi sesi menggunakan Amazon EventBridge (konsol) .

13. (Opsional) Untuk membuat aturan lebih spesifik, tambahkan nilai filter. Misalnya, jika Anda memilih State Managerdan ingin membatasi aturan ke status instans terkelola tunggal yang ditargetkan oleh Asosiasi, untuk jenis Spesifik, pilih Perubahan Status Asosiasi Instans Manajer Negara Bagian EC2.

Untuk detail selengkapnya tentang jenis detail yang didukung, lihatReferensi: Pola dan jenis EventBridge acara Amazon untuk Systems Manager.

Beberapa jenis detail memiliki opsi lain yang didukung seperti status. Pilihan yang tersedia tergantung dari kemampuan yang Anda pilih.

14. Pilih Selanjutnya.
15. Untuk jenis Target, pilih AWSlayanan.
16. Untuk Pilih target, pilih target seperti topik atauAWS Lambda fungsi Amazon SNS. Target terpicu saat peristiwa diterima yang sesuai dengan pola peristiwa yang ditentukan dalam aturan.
17. Untuk banyak jenis target, EventBridge membutuhkan izin untuk mengirim peristiwa ke target. Dalam kasus ini, EventBridge dapat membuatAWS Identity and Access Management (IAM) role yang diperlukan bagi aturan Anda untuk menjalankan:
    - Untuk membuat IAM role secara otomatis, pilih Buat peran baru untuk sumber daya khusus ini.
    - Untuk menggunakan IAM role yang Anda buat sebelumnya, pilih Gunakan peran yang ada.
18. (Opsional) Pilih Tambahkan target lainnya untuk menambahkan target lain untuk aturan ini.
19. Pilih Selanjutnya.
20. (Opsional) Masukkan satu atau lebih tanda untuk aturan. Untuk informasi selengkapnya, lihat EventBridge tag Amazon di Panduan EventBridge Pengguna Amazon.
21. Pilih Selanjutnya.
22. Tinjau detail aturan dan pilih Buat aturan.

Info lebih lanjut

- Membuat EventBridge acara yang menggunakan runbook (konsol)
- Melewati data ke Otomasi menggunakan transformator input
- Memperbaiki masalah kepatuhan menggunakan EventBridge
- Melihat tindakan penghapusan inventaris di EventBridge
- Konfigurasikan EventBridge aturan untuk dibuat OpsItems
- Mengkonfigurasi EventBridge aturan untuk parameter dan kebijakan parameter

Contoh EventBridge acara Amazon untuk Systems Manager

Berikut ini adalah contoh, dalam format JSON, dari EventBridge acara yang didukung untukAWS Systems Manager.

Jenis peristiwa Systems Manager

- Peristiwa Otomatisasi AWS Systems Manager
- AWS Systems ManagerChange CalendarAcara
- AWS Systems ManagerChange ManagerAcara
- Peristiwa Kepatuhan AWS Systems Manager
- AWS Systems ManagerMaintenance WindowsAcara
- AWS Systems ManagerParameter StoreAcara
- AWS Systems ManagerOpsCenterAcara
- AWS Systems ManagerRun CommandAcara
- AWS Systems ManagerState ManagerAcara

Peristiwa Otomatisasi AWS Systems Manager

Pemberitahuan Perubahan Status Langkah Otomasi

```
{
  "version": "0",
  "id": "eeca120b-a321-433e-9635-dab369006a6b",
  "detail-type": "EC2 Automation Step Status-change Notification",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2016-11-29T19:43:35Z",
  "region": "us-east-1",
  "resources": ["arn:aws:ssm:us-east-2:123456789012:automation-
execution/333ba70b-2333-48db-b17e-a5e69c6f4d1c",
    "arn:aws:ssm:us-east-2:123456789012:automation-definition/runcommand1:1"],
  "detail": {
    "ExecutionId": "333ba70b-2333-48db-b17e-a5e69c6f4d1c",
    "Definition": "runcommand1",
    "DefinitionVersion": 1.0,
    "Status": "Success",
    "EndTime": "Nov 29, 2016 7:43:25 PM",
    "StartTime": "Nov 29, 2016 7:43:23 PM",
```

```
    "Time": 2630.0,
    "StepName": "runFixedCmds",
    "Action": "aws:runCommand"
  }
}
```

Pemberitahuan Perubahan Status Eksekusi Otomasi

```
{
  "version": "0",
  "id": "d290ece9-1088-4383-9df6-cd5b4ac42b99",
  "detail-type": "EC2 Automation Execution Status-change Notification",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2016-11-29T19:43:35Z",
  "region": "us-east-2",
  "resources": ["arn:aws:ssm:us-east-2:123456789012:automation-
execution/333ba70b-2333-48db-b17e-a5e69c6f4d1c",
    "arn:aws:ssm:us-east-2:123456789012:automation-definition/runcommand1:1"],
  "detail": {
    "ExecutionId": "333ba70b-2333-48db-b17e-a5e69c6f4d1c",
    "Definition": "runcommand1",
    "DefinitionVersion": 1.0,
    "Status": "Success",
    "StartTime": "Nov 29, 2016 7:43:20 PM",
    "EndTime": "Nov 29, 2016 7:43:26 PM",
    "Time": 5753.0,
    "ExecutedBy": "arn:aws:iam::123456789012:user/userName"
  }
}
```

AWS Systems ManagerChange CalendarAcara

Berikut ini adalah contoh acara untuk AWS Systems ManagerChange Calendar.

> **Note**
>
> Perubahan status untuk kalender yang dibagikan dari yang lain saat Akun AWS ini tidak didukung.

Kalender BUKA

```
{
    "version": "0",
    "id": "47a3f03a-f30d-1011-ac9a-du3bdEXAMPLE",
    "detail-type": "Calendar State Change",
    "source": "aws.ssm",
    "account": "123456789012",
    "time": "2020-09-19T18:00:07Z",
    "region": "us-east-2",
    "resources": [
        "arn:aws:ssm:us-east-2:123456789012:document/MyCalendar"
    ],
    "detail": {
        "state": "OPEN",
        "atTime": "2020-09-19T18:00:07Z",
        "nextTransitionTime": "2020-10-11T18:00:07Z"
    }
}
```

Kalender DITUTUP

```
{
    "version": "0",
    "id": "f30df03a-1011-ac9a-47a3-f761eEXAMPLE",
    "detail-type": "Calendar State Change",
    "source": "aws.ssm",
    "account": "123456789012",
    "time": "2020-09-17T21:40:02Z",
    "region": "us-east-2",
    "resources": [
        "arn:aws:ssm:us-east-2:123456789012:document/MyCalendar"
    ],
    "detail": {
        "state": "CLOSED",
        "atTime": "2020-08-17T21:40:00Z",
        "nextTransitionTime": "2020-09-19T18:00:07Z"
    }
}
```

AWS Systems ManagerChange ManagerAcara

Ubah pemberitahuan pembaruan status permintaan - contoh 1

```
{
  "version": "0",
  "id": "feab80c1-a8ff-c721-b8b1-96ce70939696",
  "detail-type": "Change Request Status Update",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2023-10-24T10:51:52Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ssm:us-west-2:123456789012:opsitem/oi-12345abcdef",
    "arn:aws:ssm:us-west-2:123456789012:document/MyRunbook1"
  ],
  "detail": {
    "change-request-id": "d0585556-80f6-4522-8dad-dada6d45b67d",
    "change-request-title": "A change request title",
    "ops-item-id": "oi-12345abcdef",
    "ops-item-created-by": "arn:aws:iam::123456789012:user/JohnDoe",
    "ops-item-created-time": "2023-10-24T10:50:33.180334Z",
    "ops-item-modified-by": "arn:aws:iam::123456789012:user/JohnDoe",
    "ops-item-modified-time": "2023-10-24T10:50:33.180340Z",
    "ops-item-status": "InProgress",
    "change-template-document-name": "MyChangeTemplate",
    "runbook-document-arn": "arn:aws:ssm:us-west-2:123456789012:document/MyRunbook1",
    "runbook-document-version": "1",
    "auto-approve": true,
    "approvers": [
      "arn:aws:iam::123456789012:user/JaneDoe"
    ]
  }
}
```

Ubah pemberitahuan pembaruan status permintaan - contoh 2

```
{
  "version": "0",
  "id": "25ce6b03-2e4e-1a2b-2a8f-6c9de8d278d2",
  "detail-type": "Change Request Status Update",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2023-10-24T10:51:52Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ssm:us-west-2:123456789012:opsitem/oi-abcdef12345",
```

```
    "arn:aws:ssm:us-west-2:123456789012:document/MyRunbook1"
  ],
  "detail": {
    "change-request-id": "d0585556-80f6-4522-8dad-dada6d45b67d",
    "change-request-title": "A change request title",
    "ops-item-id": "oi-abcdef12345",
    "ops-item-created-by": "arn:aws:iam::123456789012:user/JohnDoe",
    "ops-item-created-time": "2023-10-24T10:50:33.180334Z",
    "ops-item-modified-by": "arn:aws:iam::123456789012:user/JohnDoe",
    "ops-item-modified-time": "2023-10-24T10:50:33.997163Z",
    "ops-item-status": "Rejected",
    "change-template-document-name": "MyChangeTemplate",
    "runbook-document-arn": "arn:aws:ssm:us-west-2:123456789012:document/MyRunbook1",
    "runbook-document-version": "1",
    "auto-approve": true,
    "approvers": [
      "arn:aws:iam::123456789012:user/JaneDoe"
    ]
  }
}
```

Peristiwa Kepatuhan AWS Systems Manager

Berikut ini adalah contoh peristiwa untuk AWS Systems Manager.

Sesuai Asosiasi

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Configuration Compliance State Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-07-17T19:03:26Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:managed-instance/i-01234567890abcdef"
  ],
  "detail": {
    "last-runtime": "2017-01-01T10:10:10Z",
    "compliance-status": "compliant",
    "resource-type": "managed-instance",
    "resource-id": "i-01234567890abcdef",
```

```
    "compliance-type": "Association"
  }
}
```

Asosiasi Tidak Patuh

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Configuration Compliance State Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-07-17T19:02:31Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:managed-instance/i-01234567890abcdef"
  ],
  "detail": {
    "last-runtime": "2017-01-01T10:10:10Z",
    "compliance-status": "non_compliant",
    "resource-type": "managed-instance",
    "resource-id": "i-01234567890abcdef",
    "compliance-type": "Association"
  }
}
```

Patch Compliant

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Configuration Compliance State Change",
  "source": "aws.123456789012",
  "account": "123456789012",
  "time": "2017-07-17T19:03:26Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:managed-instance/i-01234567890abcdef"
  ],
  "detail": {
    "resource-type": "managed-instance",
    "resource-id": "i-01234567890abcdef",
    "compliance-status": "compliant",
```

```
    "compliance-type": "Patch",
    "patch-baseline-id": "PB789",
    "severity": "critical"
  }
}
```

Patch Tidak Sesuai

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Configuration Compliance State Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-07-17T19:02:31Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:managed-instance/i-01234567890abcdef"
  ],
  "detail": {
    "resource-type": "managed-instance",
    "resource-id": "i-01234567890abcdef",
    "compliance-status": "non_compliant",
    "compliance-type": "Patch",
    "patch-baseline-id": "PB789",
    "severity": "critical"
  }
}
```

AWS Systems ManagerMaintenance WindowsAcara

Berikut ini adalah contoh peristiwa untuk Systems ManagerMaintenance Windows.

Daftarkan Target

Nilai status valid lainnya adalah `DEREGISTERED`.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window Target Registration Notification",
   "source":"aws.ssm",
```

```
   "account":"123456789012",
   "time":"2016-11-16T00:58:37Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-0ed7251d3fcf6e0c2",
      "arn:aws:ssm:us-east-2:123456789012:windowtarget/
e7265f13-3cc5-4f2f-97a9-7d3ca86c32a6"
   ],
   "detail":{
      "window-target-id":"e7265f13-3cc5-4f2f-97a9-7d3ca86c32a6",
      "window-id":"mw-0ed7251d3fcf6e0c2",
      "status":"REGISTERED"
   }
}
```

Jenis Eksekusi Jendela

Nilai status valid lainnya adalah `PENDING`, `IN_PROGRESS`, `SUCCESS`, `FAILED`, `TIMED_OUT`, dan `SKIPPED_OVERLAPPING`.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window Execution State-change Notification",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2016-11-16T01:00:57Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-123456789012345678"
   ],
   "detail":{
      "start-time":"2016-11-16T01:00:56.427Z",
      "end-time":"2016-11-16T01:00:57.070Z",
      "window-id":"mw-0ed7251d3fcf6e0c2",
      "window-execution-id":"b60fb56e-776c-4e5c-84ee-123456789012",
      "status":"TIMED_OUT"
   }
}
```

Jenis Eksekusi Tugas

Nilai status valid lainnya adalah `IN_PROGRESS`, `SUCCESS`, `FAILED`, dan `TIMED_OUT`.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window Task Execution State-change Notification",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2016-11-16T01:00:56Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-123456789012345678"
   ],
   "detail":{
      "start-time":"2016-11-16T01:00:56.759Z",
      "task-execution-id":"6417e808-7f35-4d1a-843f-123456789012",
      "end-time":"2016-11-16T01:00:56.847Z",
      "window-id":"mw-0ed7251d3fcf6e0c2",
      "window-execution-id":"b60fb56e-776c-4e5c-84ee-123456789012",
      "status":"TIMED_OUT"
   }
}
```

Target Tugas Diproses

Nilai status valid lainnya adalah `IN_PROGRESS`, `SUCCESS`, `FAILED`, dan `TIMED_OUT`.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window Task Target Invocation State-change Notification",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2016-11-16T01:00:57Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-123456789012345678"
   ],
   "detail":{
      "start-time":"2016-11-16T01:00:56.427Z",
      "end-time":"2016-11-16T01:00:57.070Z",
      "window-id":"mw-0ed7251d3fcf6e0c2",
      "window-execution-id":"b60fb56e-776c-4e5c-84ee-123456789012",
      "task-execution-id":"6417e808-7f35-4d1a-843f-123456789012",
      "window-target-id":"e7265f13-3cc5-4f2f-97a9-123456789012",
```

```
      "status":"TIMED_OUT",
      "owner-information":"Owner"
   }
}
```

Perubahan Status Jendela

Nilai status valid adalah `ENABLED` dan `DISABLED`.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window State-change Notification",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2016-11-16T00:58:37Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-123456789012345678"
   ],
   "detail":{
      "window-id":"mw-123456789012",
      "status":"DISABLED"
   }
}
```

AWS Systems ManagerParameter StoreAcara

Berikut ini adalah contoh peristiwa untuk Systems ManagerParameter Store.

Buat Parameter

```
{
  "version": "0",
  "id": "6a7e4feb-b491-4cf7-a9f1-bf3703497718",
  "detail-type": "Parameter Store Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-05-22T16:43:48Z",
  "region": "us-east-2",
  "resources": [
```

```
    "arn:aws:ssm:us-east-2:123456789012:parameter/MyExampleParameter"
  ],
  "detail": {
    "operation": "Create",
    "name": "MyExampleParameter",
    "type": "String",
    "description": "Sample Parameter"
  }
}
```

Perbarui Parameter

```
{
  "version": "0",
  "id": "9547ef2d-3b7e-4057-b6cb-5fdf09ee7c8f",
  "detail-type": "Parameter Store Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-05-22T16:44:48Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:parameter/MyExampleParameter"
  ],
  "detail": {
    "operation": "Update",
    "name": "MyExampleParameter",
    "type": "String",
    "description": "Sample Parameter"
  }
}
```

Hapus Parameter

```
{
  "version": "0",
  "id": "80e9b391-6a9b-413c-839a-453b528053af",
  "detail-type": "Parameter Store Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-05-22T16:45:48Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:parameter/MyExampleParameter"
```

```
  ],
  "detail": {
    "operation": "Delete",
    "name": "MyExampleParameter",
    "type": "String",
    "description": "Sample Parameter"
  }
}
```

AWS Systems ManagerOpsCenterAcara

OpsCenterOpsItembuat notifikasi

```
{
  "version": "0",
  "id": "aae66adc-7aac-f0c0-7854-7691e8c079b8",
  "detail-type": "OpsItem Create",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2023-10-19T02:48:11Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ssm:us-west-2:123456789012:opsitem/oi-123456abcdef"
  ],
  "detail": {
    "created-by": "arn:aws:iam::123456789012:user/JohnDoe",
    "created-time": "2023-10-19T02:46:53.629361Z",
    "source": "aws.ssm",
    "status": "Open",
    "ops-item-id": "oi-123456abcdef",
    "title": "An issue title",
    "ops-item-type": "/aws/issue",
    "description": "A long description may appear here"
  }
}
```

OpsCenterOpsItemperbarui pemberitahuan

```
{
  "version": "0",
  "id": "2fb5b168-b725-41dd-a890-29311200089c",
  "detail-type": "OpsItem Update",
  "source": "aws.ssm",
```

```
  "account": "123456789012",
  "time": "2023-10-19T02:48:11Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ssm:us-west-2:123456789012:opsitem/oi-123456abcdef"
  ],
  "detail": {
    "created-by": "arn:aws:iam::123456789012:user/JohnDoe",
    "created-time": "2023-10-19T02:46:54.049271Z",
    "modified-by": "arn:aws:iam::123456789012:user/JohnDoe",
    "modified-time": "2023-10-19T02:46:54.337354Z",
    "source": "aws.ssm",
    "status": "Open",
    "ops-item-id": "oi-123456abcdef",
    "title": "An issue title",
    "ops-item-type": "/aws/issue",
    "description": "A long description may appear here"
  }
}
```

AWS Systems ManagerRun CommandAcara

Run CommandPemberitahuan Perubahan Status

```
{
    "version": "0",
    "id": "51c0891d-0e34-45b1-83d6-95db273d1602",
    "detail-type": "EC2 Command Status-change Notification",
    "source": "aws.ssm",
    "account": "123456789012",
    "time": "2016-07-10T21:51:32Z",
    "region": "us-east-2",
    "resources": ["arn:aws:ec2:us-east-2:123456789012:instance/i-abcd1111"],
    "detail": {
        "command-id": "e8d3c0e4-71f7-4491-898f-c9b35bee5f3b",
        "document-name": "AWS-RunPowerShellScript",
        "expire-after": "2016-07-14T22:01:30.049Z",
        "parameters": {
            "executionTimeout": ["3600"],
            "commands": ["date"]
        },
        "requested-date-time": "2016-07-10T21:51:30.049Z",
        "status": "Success"
```

```
    }
}
```

Run CommandPemberitahuan Perubahan Status Pemanggilan

```
{
    "version": "0",
    "id": "4780e1b8-f56b-4de5-95f2-95db273d1602",
    "detail-type": "EC2 Command Invocation Status-change Notification",
    "source": "aws.ssm",
    "account": "123456789012",
    "time": "2016-07-10T21:51:32Z",
    "region": "us-east-2",
    "resources": ["arn:aws:ec2:us-east-2:123456789012:instance/i-abcd1111"],
    "detail": {
        "command-id": "e8d3c0e4-71f7-4491-898f-c9b35bee5f3b",
        "document-name": "AWS-RunPowerShellScript",
        "instance-id": "i-9bb89e2b",
        "requested-date-time": "2016-07-10T21:51:30.049Z",
        "status": "Success"
    }
}
```

AWS Systems ManagerState ManagerAcara

State ManagerPerubahan Negara Asosiasi

```
{
   "version":"0",
   "id":"db839caf-6f6c-40af-9a48-25b2ae2b7774",
   "detail-type":"EC2 State Manager Association State Change",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2017-05-16T23:01:10Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ssm:us-east-2::document/AWS-RunPowerShellScript"
   ],
   "detail":{
      "association-id":"6e37940a-23ba-4ab0-9b96-5d0a1a05464f",
      "document-name":"AWS-RunPowerShellScript",
      "association-version":"1",
      "document-version":"Optional.empty",
```

```
      "targets":"[{\"key\":\"InstanceIds\",\"values\":[\"i-12345678\"]}]",
      "creation-date":"2017-02-13T17:22:54.458Z",
      "last-successful-execution-date":"2017-05-16T23:00:01Z",
      "last-execution-date":"2017-05-16T23:00:01Z",
      "last-updated-date":"2017-02-13T17:22:54.458Z",
      "status":"Success",
      "association-status-aggregated-count":"{\"Success\":1}",
      "schedule-expression":"cron(0 */30 * * * ? *)",
      "association-cwe-version":"1.0"
   }
}
```

State ManagerPerubahan Negara Asosiasi Contoh

```
{
   "version":"0",
   "id":"6a7e8feb-b491-4cf7-a9f1-bf3703467718",
   "detail-type":"EC2 State Manager Instance Association State Change",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2017-02-23T15:23:48Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ec2:us-east-2:123456789012:instance/i-12345678",
      "arn:aws:ssm:us-east-2:123456789012:document/my-custom-document"
   ],
   "detail":{
      "association-id":"34fcb7e0-9a14-4984-9989-0e04e3f60bd8",
      "instance-id":"i-12345678",
      "document-name":"my-custom-document",
      "document-version":"1",
      "targets":"[{\"key\":\"instanceids\",\"values\":[\"i-12345678\"]}]",
      "creation-date":"2017-02-23T15:23:48Z",
      "last-successful-execution-date":"2017-02-23T16:23:48Z",
      "last-execution-date":"2017-02-23T16:23:48Z",
      "status":"Success",
      "detailed-status":"",
      "error-code":"testErrorCode",
      "execution-summary":"testExecutionSummary",
      "output-url":"sampleurl",
      "instance-association-cwe-version":"1"
   }
}
```

Skenario: Target Systems Manager di Amazon EventBridge aturan

Ketika Anda menentukan target untuk diminta di Amazon EventBridge Anda dapat memilih dari lebih dari 20 jenis target dan menambahkan hingga lima target untuk setiap aturan.

Dari berbagai target, Anda dapat memilih dari Automation,OpsCenter, danRun Command, yang merupakan kemampuanAWS Systems Manager, sebagai tindakan target saat EventBridge peristiwa terjadi.

Berikut ini adalah beberapa contoh cara Anda dapat menggunakan kemampuan ini sebagai target dari EventBridge aturan.

Contoh Otomatisasi

Anda dapat mengonfigurasi EventBridge aturan untuk memulai alur kerja Otomatisasi saat peristiwa seperti berikut terjadi:

- Saat Amazon CloudWatch alarm melaporkan bahwa node yang dikelola telah gagal pemeriksaan status (`StatusCheckFailed_Instance=1`), jalankan`AWSSupport-ExecuteEC2Rescue`Runbook otomatisasi pada node.
- Ketika peristiwa `EC2 Instance State-change Notification` terjadi karena instans Amazon Elastic Compute Cloud (Amazon EC2) baru berjalan, jalankan runbook Otomatisasi `AWS-AttachEBSVolume` pada instans.
- Ketika volume Amazon Elastic Block Store (Amazon EBS) dibuat dan tersedia, jalankan runbook Otomatisasi `AWS-CreateSnapshot` pada volume.

Contoh OpsCenter

Anda dapat mengonfigurasi EventBridge aturan untuk membuat`OpsItem`terjadi hal-hal berikut:

- Peristiwa throttling untuk Amazon DynamoDB terjadi, atau performa volume Amazon EBS telah terdegradasi.
- Grup Amazon EC2 Auto Scaling gagal untuk meluncurkan node, atau alur kerja Otomatisasi Systems Manager gagal.
- Instans EC2 mengubah tahapan dari `Running` ke `Stopped`.

Contoh Run Command

Anda dapat mengonfigurasi EventBridge aturan untuk menjalankan dokumen Systems Manager perintah perintah perintahRun Commandsaat peristiwa seperti berikut terjadi:

- Ketika grup Auto Scaling akan segera berakhir, aRun Commandskrip dapat memperoleh berkas log dari node sebelum ia berakhir.
- Ketika node baru dibuat dalam kelompok Auto Scaling, sebuahRun Commandtindakan target dapat mengaktifkan peran server web atau menginstal perangkat lunak pada node.
- Ketika node yang dikelola ditemukan tidak sesuai, aRun Commandaksi target dapat memperbarui patch pada node dengan menjalankan`AWS-RunPatchBaseline`dokumen.

Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

> **Note**
>
> Topik FIFO Layanan Pemberitahuan Sederhana Amazon tidak didukung.

Anda dapat mengonfigurasi Amazon Simple Notification Service (Amazon SNS) untuk mengirimkan notifikasi tentang status perintah yang Anda kirimkan menggunakan Run Command atauMaintenance Windows, yang merupakan kemampuan. AWS Systems Manager Amazon SNS mengoordinasikan dan mengelola pemberitahuan pengiriman ke pelanggan atau titik akhir yang berlangganan. Anda dapat menerima pemberitahuan setiap kali perintah berubah ke keadaan baru atau keadaan tertentu, seperti Gagal atau Timed Out. Dalam kasus di mana Anda mengirimkan perintah ke beberapa node, Anda dapat menerima notifikasi untuk setiap salinan perintah yang dikirimkan ke node tertentu. Setiap salinan disebut permintaan.

Amazon SNS dapat mengirimkan notifikasi sebagai HTTP atau HTTPS POST, email (SMTP, baik teks biasa maupun dalam format JSON), atau sebagai pesan yang diunggah ke antrean Amazon Simple Queue Service (Amazon SQS). Untuk informasi lebih lanjut, lihat Apa itu Amazon SNS di Panduan Developer Amazon Simple Notification Service. Untuk contoh struktur data JSON yang disertakan di notifikasi Amazon SNS yang disediakan oleh Run Command danMaintenance Windows, lihat. Contoh notifikasi Amazon SNS untuk AWS Systems Manager

Mengonfigurasi notifikasi Amazon SNS untuk AWS Systems Manager

Run Commanddan Maintenance Windows tugas yang terdaftar ke jendela pemeliharaan dapat mengirimkan notifikasi Amazon SNS untuk tugas perintah yang memasukkan status berikut:

- Dalam Progres
- Sukses
- Gagal
- Waktu Habis
- Dibatalkan

Untuk informasi tentang kondisi yang menyebabkan perintah untuk memasukkan salah satu status ini, lihat Memahami status perintah.

> **Note**
>
> Perintah yang dikirimkan menggunakan Run Command juga melaporkan status Pembatalan dan Tertunda. Status ini tidak didapatkan oleh notifikasi Amazon SNS.

Notifikasi Amazon SNS ringkasan perintah

Jika Anda mengonfigurasi Run Command atau Run Command tugas di jendela pemeliharaan untuk notifikasi Amazon SNS, Amazon SNS mengirimkan pesan ringkasan yang menyertakan informasi berikut.

Bidang	Tipe	Deskripsi
eventTime	String	Waktu acara tersebut dimulai. Stempel waktu penting karena Amazon SNS tidak menjamin urutan pengiriman pesan. Misalnya: 2016-04-26T13:15:30Z

Bidang	Tipe	Deskripsi
documentName	String	Nama dokumen SSM yang digunakan untuk menjalankan perintah ini.
commandId	String	ID yang dihasilkan oleh Run Command setelah perintah dikirimkan.
expiresAfter	Tanggal	Jika waktu ini tercapai dan perintah belum mulai mengeksekusi, ia tidak akan dijalankan.
Keluaran3 BucketName	String	Bucket Amazon Simple Storage Service (Amazon S3) tempat tanggapan terhadap eksekusi perintah harus disimpan.
Keluaran3 KeyPrefix	String	Jalur direktori Amazon S3 di dalam bucket tempat tanggapan terhadap eksekusi perintah harus disimpan.
requestedDateTime	String	Waktu dan tanggal saat permintaan dikirimkan ke node tertentu ini.

Bidang	Tipe	Deskripsi
instanceIds	StringList	Node yang ditargetkan oleh perintah. **Note** ID instans hanya disertakan di pesan ringkasan jika Run Command tugas menargetkan ID instans secara langsung. ID instans tidak disertakan di pesan ringkasan jika Run Command tugas diterbitkan menggunakan penargetan berbasis tanda.
status	String	Status perintah untuk perintah.

Notifikasi Amazon SNS berbasis permintaan

Jika Anda mengirimkan perintah ke beberapa node, Amazon SNS dapat mengirimkan pesan tentang setiap salinan atau permintaan perintah. Pesan ini mencakup informasi berikut.

Bidang	Tipe	Deskripsi
eventTime	String	Waktu acara tersebut dimulai. Stempel waktu penting karena Amazon SNS tidak menjamin urutan pengiriman pesan. Misalnya: 2016-04-26T13:15:30Z

Bidang	Tipe	Deskripsi
documentName	String	Nama dokumen Systems Manager (dokumen SSM) yang digunakan untuk menjalankan perintah ini.
requestedDateTime	String	Waktu dan tanggal saat permintaan dikirimkan ke node tertentu ini.
commandId	String	ID yang dihasilkan oleh Run Command setelah perintah dikirimkan.
instanceId	String	Instans yang ditargetkan oleh perintah.
status	String	Status perintah untuk permintaan ini.

Untuk menyiapkan notifikasi Amazon SNS saat perintah mengubah status, selesaikan tugas berikut.

> **Note**
>
> Jika Anda tidak mengonfigurasi notifikasi Amazon SNS untuk jendela pemeliharaan, maka Anda dapat melompati Tugas 5 nantinya di topik ini.

Topik

Tugas 1: Membuat dan berlangganan topik Amazon SNS

Topik Amazon SNS adalah saluran komunikasi yang Run Command dan Run Command tugas yang terdaftar ke jendela pemeliharaan digunakan untuk mengirimkan notifikasi tentang status perintah Anda. Amazon SNS mendukung berbagai protokol komunikasi, termasuk HTTP/S, email, dan lainnya Layanan AWS seperti Amazon Simple Queue Service (Amazon SQS). Untuk memulai, kami merekomendasikan agar Anda memulai dengan protokol email. Untuk informasi tentang cara membuat topik, lihat Membuat topik Amazon SNS di Panduan Developer Amazon Simple Notification Service.

> **Note**
>
> Setelah Anda membuat topik, salin atau catat ARN Topik. Anda menentukan ARN ini saat mengirimkan perintah yang dikonfigurasi untuk menampilkan notifikasi status.

Setelah Anda membuat topik, berlangganlah dengan menentukan Titik akhir. Jika Anda memilih protokol Email, titik akhirnya adalah alamat email tempat Anda ingin menerima notifikasi. Untuk informasi selengkapnya tentang cara berlangganan topik, lihat Berlangganan topik Amazon SNS di Panduan Developer Amazon Simple Notification Service.

Amazon SNS mengirimkan email konfirmasi dari Notifikasi AWS ke alamat email yang Anda tentukan. Buka email dan pilih tautan Konfirmasi langganan.

Anda akan menerima pesan pengakuan dari AWS. Amazon SNS kini dikonfigurasi untuk menerima notifikasi dan mengirimkan notifikasi sebagai email ke alamat email yang Anda tentukan.

Tugas 2: Membuat kebijakan IAM untuk notifikasi Amazon SNS

Gunakan prosedur berikut untuk membuat kebijakan AWS Identity and Access Management (IAM) kustom yang memberikan izin untuk memulai notifikasi Amazon SNS.

Untuk membuat kebijakan IAM untuk notifikasi Amazon SNS

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Pada panel navigasi, pilih Kebijakan, lalu pilih Buat kebijakan. (Jika tombol Memulai ditunjukkan, pilihlah, lalu pilih Buat Kebijakan.)
3. Pilih tab JSON.
4. Ganti konten default dengan berikut.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sns:Publish"
            ],
            "Resource": "arn:aws:sns:region:account-id:sns-topic-name"
        }
    ]
}
```

wilayah mewakili pengenal untuk Wilayah AWS didukung oleh AWS Systems Manager, seperti `us-east-2` untuk Wilayah US East (Ohio). Untuk daftar nilai *wilayah* yang didukung, lihat kolom Wilayah di endpoint layanan Systems Manager di Referensi Umum Amazon Web Services.

account-id merupakan pengenal 12 digit untuk AndaAkun AWS, dalam format. `123456789012`

*sns-topic-name*mewakili nama topik Amazon SNS yang ingin Anda gunakan untuk notifikasi penerbitan.

5. Pilih Next: Tags (Selanjutnya: Tanda).
6. (Opsional) Tambahkan satu atau beberapa pasangan nilai kunci tag untuk mengatur, melacak, atau mengontrol akses untuk kebijakan ini.
7. Pilih Next: Review (Selanjutnya: Tinjauan).
8. Di halaman Tinjau Kebijakan, untuk Nama, ketikkan nama untuk kebijakan inline tersebut. Sebagai contoh: **`my-sns-publish-permissions`**.
9. (Opsional) Untuk Deskripsi, masukkan deskripsi untuk kebijakan.
10. Pilih Buat kebijakan.

Tugas 3: Membuat IAM role untuk notifikasi Amazon SNS

Gunakan prosedur berikut untuk membuat IAM role untuk notifikasi Amazon SNS. Peran layanan ini digunakan oleh Systems Manager untuk memulai notifikasi Amazon SNS. Di semua prosedur berikutnya, peran ini disebut sebagai IAM role Amazon SNS.

Untuk membuat peran layanan IAM untuk notifikasi Amazon SNS

1. Masuk ke AWS Management Console dan buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi konsol IAM, pilih Peran, dan lalu pilih Buat peran.
3. Pilih tipe Layanan AWSperan, lalu pilih Systems Manager.
4. Pilih kasus penggunaan Systems Manager. Lalu, pilih Selanjutnya.
5. Pada halaman Lampirkan kebijakan izin, pilih kotak di sebelah kiri nama kebijakan kustom yang Anda buat di Tugas 2. Sebagai contoh: **`my-sns-publish-permissions`**.
6. (Opsional) Tetapkan batas izin. Ini adalah fitur lanjutan yang tersedia untuk peran layanan, tetapi bukan peran tertaut layanan.

   Buka bagian Batas izin dan pilih Gunakan batas izin untuk mengontrol izin peran maksimum. IAM mencakup daftar kebijakan yang AWS dikelola pelanggan dan kebijakan yang dikelola pelanggan di akun Anda. Pilih kebijakan yang akan digunakan untuk batas izin atau pilih Buat kebijakan untuk membuka tab peramban baru dan membuat kebijakan baru dari awal. Untuk informasi selengkapnya, lihat Membuat kebijakan IAM dalam Panduan Pengguna IAM. Setelah Anda membuat kebijakan, tutup tab tersebut dan kembali ke tab asli Anda untuk memilih kebijakan yang akan digunakan untuk batas izin.
7. Pilih Selanjutnya.
8. Jika memungkinkan, masukkan nama peran atau akhiran nama peran untuk membantu Anda mengidentifikasi tujuan peran ini. Nama peran harus unik di AndaAkun AWS. Grup tidak dibedakan berdasarkan huruf besar-kecil. Misalnya, Anda tidak dapat membuat peran dengan nama **`PRODROLE`** dan **`prodrole`**. Anda tidak dapat mengubah nama peran setelah dibuat karena berbagai entitas mungkin mereferensikan peran tersebut.
9. (Opsional) Untuk Deskripsi, masukkan deskripsi untuk peran baru tersebut.
10. Pilih Edit di Langkah 1: Pilih entitas tepercaya atau Langkah 2: Pilih bagian izin untuk mengedit kasus penggunaan dan izin untuk peran tersebut.
11. (Opsional) Tambahkan metadata ke pengguna dengan cara melampirkan tanda sebagai pasangan nilai kunci. Untuk informasi selengkapnya tentang menggunakan tag di IAM, lihat Menandai sumber daya IAM di Panduan Pengguna IAM.
12. Tinjau peran dan kemudian pilih Buat peran.
13. Pilih nama peran, lalu salin atau catat nilai Peran ARN. Amazon Resource Name (ARN) untuk peran tersebut digunakan saat Anda mengirimkan perintah yang dikonfigurasi untuk menampilkan notifikasi Amazon SNS.

14. Biarkan halaman Ringkasan terbuka.

Tugas 4: Mengonfigurasi akses pengguna

Jika entitas IAM (pengguna, peran, atau grup) diberi izin administrator, maka pengguna atau peran memiliki akses ke Run Command danMaintenance Windows, kemampuan. AWS Systems Manager

Untuk entitas tanpa izin administrator, administrator harus memberikan izin berikut kepada entitas IAM:

- Kebijakan yang `AmazonSSMFullAccess` dikelola, atau kebijakan yang memberikan izin yang sebanding.
- `iam:PassRole`izin untuk peran yang dibuat diTugas 3: Membuat IAM role untuk notifikasi Amazon SNS. Misalnya:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::account-id:role/sns-role-name"
        }
    ]
}
```

Untuk menyediakan akses, tambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup diAWS IAM Identity Center:

  Buat set izin. Ikuti petunjuk di Buat set izin di Panduan AWS IAM Identity Center Pengguna.
- Pengguna yang dikelola dalam IAM melalui penyedia identitas:

  Membuat peran untuk federasi identitas. Ikuti petunjuk dalam Membuat peran untuk penyedia identitas pihak ketiga (federasi) di Panduan Pengguna IAM.
- Pengguna IAM:

- Buat peran yang dapat diasumsikan pengguna Anda. Ikuti petunjuk dalam Membuat peran untuk pengguna IAM di Panduan Pengguna IAM.
- (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk dalam Menambahkan izin ke pengguna (konsol) di Panduan Pengguna IAM.

Untuk mengonfigurasi akses pengguna dan melampirkan **`iam:PassRole`** kebijakan ke akun pengguna

1. Di panel navigasi IAM, pilih Pengguna, lalu pilih akun pengguna yang ingin Anda konfigurasikan.
2. Pada tab Izin, di daftar kebijakan, pastikan kebijakan **`AmazonSSMFullAccess`** sudah tercantum atau ada kebijakan yang sebanding yang mengizinkan akun untuk mengakses Systems Manager.
3. Pilih Tambahkan kebijakan inline.
4. Di halaman Buat kebijakan, pilih tab Visual editor.
5. Pilih Pilih layanan, lalu pilih IAM.
6. Untuk Tindakan, dalam kotak teks Tindakan filter, masukkan**`PassRole`**, lalu centang kotak di samping PassRole.
7. Untuk Sumber Daya, verifikasi bahwa Spesifik dipilih, lalu pilih Tambahkan ARN.
8. Di bidang Tentukan ARN untuk peran, tempelkan ARN IAM role Amazon SNS yang Anda salin pada akhir Tugas 3. Sistem secara otomatis mengisi bidang Akun dan Nama peran dengan jalur.
9. Pilih Tambahkan.
10. Pilih Tinjau kebijakan.
11. Pada halaman Tinjau Kebijakan, masukkan nama, lalu pilih Buat kebijakan.

Tugas 5: Melampirkan iam: PassRole kebijakan ke peran jendela pemeliharaan Anda

Ketika Anda mendaftarkan Run Command tugas dengan jendela pemeliharaan, Anda menentukan peran layanan Amazon Resource Name (ARN). Peran layanan ini digunakan oleh Systems Manager untuk menjalankan tugas yang terdaftar ke jendela pemeliharaan. Untuk mengonfigurasi notifikasi Amazon SNS untuk Run Command tugas terdaftar, lampirkan `iam:PassRole` kebijakan ke peran layanan jendela pemeliharaan yang ditentukan. Jika Anda tidak bermaksud untuk mengonfigurasi tugas terdaftar untuk notifikasi Amazon SNS, maka Anda dapat melompati tugas ini.

`iam:PassRole`Kebijakan ini memungkinkan peran Maintenance Windows layanan untuk meneruskan peran Amazon SNS yang dibuat di Tugas 3 ke layanan Amazon SNS. Prosedur berikut menunjukkan cara melampirkan `iam:PassRole` kebijakan ke peran Maintenance Windows layanan.

> **Note**
>
> Gunakan peran layanan kustom untuk jendela pemeliharaan Anda guna mengirimkan notifikasi terkait Run Command tugas yang terdaftar. Untuk informasi, lihat Menyiapkan Maintenance Windows.
> Jika Anda perlu membuat peran layanan kustom untuk tugas jendela pemeliharaan, lihatGunakan konsol untuk mengonfigurasi izin untuk jendela pemeliharaan.

Untuk melampirkan **`iam:PassRole`** kebijakan ke Maintenance Windows peran Anda

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran dan pilih IAM role Amazon SNS yang dibuat di Tugas 3.
3. Salin atau catat ARN Peran dan kembalilah ke bagian Peran dari konsol IAM.
4. Pilih peran Maintenance Windows layanan kustom yang Anda buat dari daftar Nama peran.
5. Pada tab Izin, pastikan kebijakan tercantum atau ada `AmazonSSMMaintenanceWindowRole` kebijakan yang sebanding yang mengizinkan jendela pemeliharaan ke API Systems Manager. Jika tidak, pilih Tambahkan izin, Lampirkan kebijakan untuk melampirkannya.
6. Pilih Tambahkan izin, Buat kebijakan sebaris.
7. Pilih tab Editor visual.
8. Untuk Layanan, pilih IAM.
9. Untuk Tindakan, dalam kotak teks Tindakan filter, masukkan**`PassRole`**, lalu centang kotak di samping PassRole.
10. Untuk Sumber Daya, pilih Spesifik, lalu pilih Tambahkan ARN.
11. Di kotak Tentukan ARN untuk peran, tempelkan ARN dari IAM role Amazon SNS yang dibuat di Tugas 3, lalu pilih Tambahkan.
12. Pilih Tinjau kebijakan.
13. Pada halaman Tinjau kebijakan, tentukan nama untuk `PassRole` kebijakan, lalu pilih Buat kebijakan.

Contoh notifikasi Amazon SNS untuk AWS Systems Manager

Anda dapat mengonfigurasi Amazon Simple Notification Service (Amazon SNS) untuk mengirimkan notifikasi tentang status perintah yang Anda kirimkan menggunakanRun CommandatauMaintenance Windows, yang merupakan kemampuanAWS Systems Manager.

> **Note**
>
> Panduan ini tidak membahas cara mengonfigurasi notifikasi untukRun CommandatauMaintenance Windows. Untuk informasi tentang konfigurasiRun CommandatauMaintenance Windowsuntuk mengirim notifikasi Amazon SNS tentang status perintah, lihatMengonfigurasi notifikasi Amazon SNS untuk AWS Systems Manager.

Contoh berikut menunjukkan struktur output JSON yang ditampilkan oleh notifikasi Amazon SNS saat dikonfigurasi untukRun CommandatauMaintenance Windows.

Sampel pesan ringkasan Output JSON untuk Perintah menggunakan penargetan ID instans

```
{
    "commandId": "a8c7e76f-15f1-4c33-9052-0123456789ab",
    "documentName": "AWS-RunPowerShellScript",
    "instanceIds": [
        "i-1234567890abcdef0",
        "i-9876543210abcdef0"
    ],
    "requestedDateTime": "2019-04-25T17:57:09.17Z",
    "expiresAfter": "2019-04-25T19:07:09.17Z",
    "outputS3BucketName": "DOC-EXAMPLE-BUCKET",
    "outputS3KeyPrefix": "runcommand",
    "status": "InProgress",
    "eventTime": "2019-04-25T17:57:09.236Z"
}
```

Sampel pesan ringkasan Output JSON untuk Perintah menggunakan penargetan berbasis tanda

```
{
    "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"
}
```

Sampel pesan Output JSON untuk Permintaan

```
{
    "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"
}
```

Gunakan Run Command untuk mengirim perintah yang mengembalikan pemberitahuan status

Prosedur berikut menunjukkan cara menggunakan AWS Command Line Interface (AWS CLI) atau AWS Systems Manager konsol untuk mengirim perintah melaluiRun Command, kemampuanAWS Systems Manager, yang dikonfigurasi untuk mengembalikan pemberitahuan status.

Mengirim Run Command yang mengembalikan notifikasi (konsol)

Gunakan prosedur berikut untuk mengirim perintah melalui Run Command yang dikonfigurasi untuk mengembalikan pemberitahuan status menggunakan konsol Systems Manager.

Untuk mengirimkan perintah yang menampilkan notifikasi (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Run Command.

3. Pilih Run Command.
4. Di daftar Dokumen perintah, pilih dokumen Systems Manager.
5. Di bagian Parameter perintah, tentukan nilai untuk parameter yang diperlukan.
6. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

   > **Tip**
   >
   > Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

7. Untuk Parameter lainnya:

   - Untuk Komentar, ketik informasi tentang perintah ini.
   - Untuk Waktu habis (detik), tentukan jumlah detik untuk menunggu sistem sebelum gagal menjalankan perintah keseluruhan.

8. Untuk Pengendalian rate:

   - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

     > **Note**
     >
     > Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

   - Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

9. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan izin pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbedaAkun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

10. Di bagian Notifikasi SNS, pilih Aktifkan notifikasi SNS.
11. Untuk peran IAM, pilih ARN peran Amazon SNS IAM yang Anda buat di Tugas 3 di. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS
12. Untuk topik SNS, masukkan ARN topik Amazon SNS yang akan digunakan.
13. Untuk pemberitahuan Acara, pilih acara yang ingin Anda terima notifikasi.
14. Untuk pemberitahuan Ubah, pilih untuk menerima pemberitahuan hanya untuk ringkasan perintah (Perubahan status perintah) atau untuk setiap salinan perintah yang dikirim ke beberapa node (Status perintah pada setiap instance berubah).
15. Pilih Jalankan.
16. Periksa email Anda untuk melihat pesan dari Amazon SNS dan buka pesan email. Amazon SNS memerlukan waktu beberapa menit untuk mengirimkan pesan email.

Mengirim Run Command yang mengembalikan notifikasi (CLI)

Gunakan prosedur berikut untuk mengirim perintah melalui Run Command yang dikonfigurasi untuk mengembalikan pemberitahuan status menggunakanAWS CLI.

Untuk mengirimkan perintah yang menampilkan notifikasi (CLI)

1. Buka AWS CLI.
2. Tentukan parameter dalam perintah berikut untuk menargetkan berdasarkan ID node terkelola.

```
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"}'
```

Berikut adalah contohnya.

```
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"}'
```

Perintah alternatif

Tentukan parameter dalam perintah berikut untuk menargetkan instans terkelola menggunakan tanda.

```
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"}'
```

Berikut adalah contohnya.

```
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. Tekan Enter.
4. Periksa email Anda untuk melihat pesan dari Amazon SNS dan buka pesan email. Amazon SNS memerlukan waktu beberapa menit untuk mengirimkan pesan email.

Untuk informasi selengkapnya, lihat send-command dalam AWS CLI Referensi Perintah.

Gunakan jendela pemeliharaan untuk mengirimkan perintah yang menampilkan notifikasi status

Prosedur berikut menunjukkan cara mendaftarkan Run Command tugas dengan jendela pemeliharaan Anda menggunakan AWS Systems Manager konsol atau AWS Command Line Interface (AWS CLI). Run Commandadalah kemampuanAWS Systems Manager. Prosedur juga menjelaskan cara mengonfigurasi Run Command tugas untuk menampilkan notifikasi status.

Sebelum Anda memulai

Jika Anda belum membuat jendela pemeliharaan atau target terdaftar, lihat Menggunakan windows pemeliharaan (konsol) untuk langkah tentang cara membuat jendela pemeliharaan dan mendaftarkan target.

Untuk menerima notifikasi dari Amazon Simple Notification Service (Amazon SNS), lampirkan `iam:PassRole` kebijakan ke peran Maintenance Windows layanan yang ditentukan di tugas terdaftar. Jika Anda belum menambahkan `iam:PassRole` izin ke peran Maintenance Windows layanan, lihatTugas 5: Melampirkan iam: PassRole kebijakan ke peran jendela pemeliharaan Anda.

Pendaftaran Run Command tugas ke jendela pemeliharaan yang menampilkan notifikasi (konsol)

Gunakan prosedur berikut untuk mendaftarkan Run Command tugas yang dikonfigurasi untuk menampilkan notifikasi status ke jendela pemeliharaan Anda menggunakan konsol Systems Manager.

Untuk mendaftarkan Run Command tugas dengan jendela pemeliharaan yang menampilkan notifikasi (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Maintenance Windows.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Maintenance Windows.
3. Pilih jendela pemeliharaan yang ingin Anda daftarkan Run Command tugas yang dikonfigurasi untuk mengirimkan notifikasi Amazon Simple Notification Service (Amazon SNS).

4. Pilih Tindakan dan kemudian pilih Daftar Jalankan tugas perintah.
5. (Opsional) Di bidang Nama, masukkan nama untuk tugas.
6. (Opsional) Di bidang Deskripsi, masukkan deskripsi.
7. Untuk Dokumen Command, pilih dokumen Perintah.
8. Untuk Prioritas tugas, tentukan prioritas untuk tugas ini. Nol (`0`) adalah prioritas tertinggi. Tugas di jendela pemeliharaan dijadwalkan dalam urutan prioritas. Tugas yang memiliki prioritas yang sama dijadwalkan secara paralel.
9. Di bagian Target, pilih grup target terdaftar atau pilih target yang tidak terdaftar.
10. Untuk Pengendalian rate:

    - Untuk Konkurensi, tetapkan jumlah atau persentase node terkelola untuk menjalankan perintah pada saat yang sama.

      > **Note**
      >
      > Jika Anda memilih target dengan menentukan tag diterapkan ke node terkelola atau dengan menentukan AWS sumber daya grup, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada waktu yang sama dengan menentukan persentase.

    - Untuk Ambang batas kesalahan, tetapkan kapan harus berhenti menjalankan perintah pada node terkelola lainnya setelah gagal pada sejumlah atau persentase node. Misalnya, jika Anda menentukan tiga kesalahan, maka Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memroses perintah juga dapat mengirim kesalahan.
11. Di area Peran layanan IAM, pilih peran Maintenance Windows layanan yang memiliki `iam:PassRole` izin ke peran SNS.

    > **Note**
    >
    > Tambahkan `iam:PassRole` izin ke Maintenance Windows peran agar Systems Manager meneruskan peran SNS ke Amazon SNS. Jika Anda belum menambahkan izin `iam:PassRole`, lihat Tugas 5 di topik Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS.

12. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Aktifkan output penulisan ke S3. Masukkan nama bucket dan prefiks (folder) di dalam kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah dari profil instans yang ditetapkan ke node terkelola, bukan data pengguna IAM yang melaksanakan tugas ini. Untuk informasi lebih lanjut, lihat Mengkonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berada dalam yang berbedaAkun AWS, verifikasi bahwa profil instans atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

13. Di bagian notifikasi SNS, lakukan hal berikut:

    - Pilih Aktifkan Pemberitahuan SNS.
    - Untuk peran IAM, pilih peran Amazon SNS IAM role Amazon Resource Name (ARN) yang Anda buat di Tugas 3 di Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS untuk memulai Amazon SNS.
    - Untuk topik SNS, masukkan ARN topik Amazon SNS untuk digunakan.
    - Untuk Jenis acara, pilih acara yang ingin Anda terima notifikasinya.
    - Untuk Jenis notifikasi, pilih untuk menerima notifikasi agar setiap salinan perintah dikirimkan ke beberapa node (permintaan) atau ringkasan perintah.

14. Di bagian Parameter, masukkan parameter yang diperlukan berdasarkan dokumen Perintah yang Anda pilih.
15. Pilih tugas Run command.
16. Setelah jendela pemeliharaan berikutnya berjalan, periksa email Anda untuk melihat pesan dari Amazon SNS dan buka pesan email. Amazon SNS memerlukan waktu beberapa menit untuk mengirimkan pesan email.

Pendaftaran Run Command tugas ke jendela pemeliharaan yang menampilkan notifikasi (CLI)

Gunakan prosedur berikut untuk mendaftarkan Run Command tugas yang dikonfigurasi untuk menampilkan notifikasi status ke jendela pemeliharaan menggunakanAWS CLI.

Untuk mendaftarkan Run Command tugas dengan jendela pemeliharaan yang menampilkan notifikasi (CLI)

> **Note**
>
> Untuk mengelola pilhan tugas Anda dengan lebih baik, prosedur ini menggunakan pilihan perintah `--cli-input-json`, dengan nilai pilihan disimpan di file JSON.

1. Pada mesin lokal Anda, buat file bernama `RunCommandTask.json`.
2. Tempelkan konten berikut ini ke file.

```
{
    "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. Ganti nilai contoh dengan informasi tentang sumber daya Anda sendiri.

   Anda juga dapat memulihkan pilihanyang telah kami hilangkan dari contoh ini jika Anda ingin menggunakannya. Misalnya, Anda dapat menyimpan output perintah ke bucket S3.

   Untuk informasi selengkapnya, lihat register-task-with-maintenance-windowdi Referensi AWS CLI Perintah.
4. Simpan file tersebut.
5. Di direktori pada mesin lokal tempat Anda menyimpan file, jalankan perintah berikut.

   ```
   aws ssm register-task-with-maintenance-window --cli-input-json file://
   RunCommandTask.json
   ```

   > **Important**
   >
   > Pastikan untuk menyertakan `file://` sebelum nama file. Ia diperlukan dalam perintah ini.

   Jika berhasil, sistem menampilkan informasi seperti berikut ini.

   ```
   {
       "WindowTaskId": "j2l8d5b5c-mw66-tk4d-r3g9-1d4d1EXAMPLE"
   }
   ```
6. Setelah eksekusi jendela pemeliharaan berikutnya, periksa email Anda untuk melihat pesan dari Amazon SNS dan buka pesan email. Amazon SNS memerlukan waktu beberapa menit untuk mengirimkan pesan email.

Untuk informasi selengkapnya tentang mendaftarkan tugas untuk jendela pemeliharaan dari baris perintah, lihat Mendaftarkan tugas dengan jendela pemeliharaan.

Integrasi produk dan layanan dengan Systems Manager

Secara default,AWS Systems Manager berintegrasi denganLayanan AWS serta produk dan layanan lainnya. Informasi berikut dapat membantu Anda mengonfigurasi Systems Manager untuk berintegrasi dengan produk dan layanan yang Anda gunakan.

- Integrasi dengan Layanan AWS
- Integrasi dengan produk dan layanan lainnya

Integrasi dengan Layanan AWS

Melalui penggunaan dokumen Systems Manager Command (dokumen SSM) dan runbook Otomatisasi, Anda dapat menggunakanAWS Systems Manager untuk berintegrasi denganLayanan AWS. Untuk informasi lebih lanjut tentang sumber daya ini, lihat AWS Systems ManagerDokumen.

Systems Manager berintegrasi dengan yang berikutLayanan AWS.

Hitung

Amazon Elastic Compute Cloud (Amazon EC2)	Amazon EC2 menyediakan kapasitas komputasi yang dapat diskalakan di AWS Cloud. Menggunakan Amazon EC2 menghilangkan kebutuhan Anda untuk berinvestasi pada perangkat keras di awal, sehingga Anda dapat mengembangkan dan menerapkan aplikasi lebih cepat. Anda dapat menggunakan Amazon EC2 untuk meluncurkan server virtual sebanyak atau sesedikit yang Anda butuhkan, mengonfigurasi keamanan dan jaringan, dan mengelola penyimpanan. Systems Manager memungkinkan Anda untuk melakukan beberapa tugas pada instans EC2. Misalnya Anda dapat meluncurkan, mengonfigurasi, mengelola, memelihara, memecahkan masalah, dan secara aman

	terhubung ke instans EC2 Anda. Anda juga dapat menggunakan Systems Manager untuk men-deploy perangkat lunak, menentukan status kepatuhan, dan mengumpulkan inventaris dari instans EC2 Anda. Pelajari selengkapnya • Bekerja dengan node terkelola • AWS Systems Manager State Manager • AWS Systems Manager Run Command • AWS Systems Manager Patch Manager • AWS Systems Manager Session Manager • AWS Systems Manager Distributor • AWS Systems Manager Kepatuhan • AWS Systems ManagerInventaris
Amazon EC2 Auto Scaling	Auto Scaling membantu Anda memastikan agar Anda memiliki ketersediaan jumlah instans EC2 yang tepat untuk menangani beban untuk aplikasi Anda. Anda membuat koleksi instance EC2, yang disebut Grup Auto Scaling. Systems Manager memungkinkan Anda untuk mengotomatiskan prosedur umum seperti patching Amazon Machine Image (AMI) yang digunakan dalam template Auto Scaling untuk grup Auto Scaling Anda. Pelajari selengkapnya MemperbaruiAMIs untuk Auto Scaling

Amazon Elastic Container Service (Amazon ECS)	Amazon ECS adalah layanan pengelola an kontainer yang sangat dapat diskalakan dan cepat, yang memungkinkan Anda untuk menjalankan, menghentikan, dan mengelola kontainer Docker pada sebuah klaster. Systems Manager memungkinkan Anda untuk mengelola instans kontainer dari jarak jauh dan menyuntikkan data sensitif ke dalam kontainer Anda dengan menyimpan data sensitif Anda dalam parameterParameter Store, sebuah kemampuan dari Systems Manager, dan kemudian mereferensikan mereka dalam definisi kontainer Anda. Pelajari selengkapnya • Mengelola instans kontainer dari jarak jauh menggunakanAWS Systems Manager • Menentukan data sensitif menggunakan Systems ManagerParameter Store

AWS Lambda	Lambda adalah layanan komputasi yang memungkinkan Anda menjalankan kode tanpa perlu menyediakan atau mengelola server. Lambda menjalankan kode Anda hanya saat diperlukan dan menskalakan secara otomatis, dari beberapa permintaan per hari hingga ribuan per detik. Systems Manager memungkinkan Anda untuk menggunakan fungsi Lambda dalam konten runbook Otomatisasi dengan menggunakan tindakan `aws:invokeLambdaFunction` . Untuk menggunakan parameter dariParameter Store dalamAWS Lambda fungsi, Anda dapat menggunakanAWS Parameter dan Rahasia Ekstensi Lambda untuk mengambil nilai parameter dan cache mereka untuk digunakan di future. Pelajari selengkapnya Perbarui emas AMI menggunakan Otomasi,AWS Lambda, dan Parameter Store Menggunakan Parameter Store parameter dalam AWS Lambda fungsi

Internet of Things (IoT)

AWS IoT Greengrassperangkat inti perangkat inti	AWS IoT Greengrassadalah waktu aktif edge IoT sumber terbuka dan layanan cloud yang membantu Anda membangun, men-deploy, dan mengelola aplikasi IoT pada perangkat Anda. Systems Manager menawarkan dukungan asli untuk perangkatAWS IoT Greengrass inti.

	Pelajari selengkapnya AWS Systems ManagerMenyiapkan perangkat edge
AWS IoTperangkat inti perangkat inti	AWS IoTmenyediakan layanan cloud yang menghubungkan perangkat IoT Anda ke perangkat lain dan layananAWS cloud. AWS IoTmenyediakan perangkat lunak perangkat yang dapat membantu Anda mengintegrasikan perangkat IoT Anda ke dalam solusiAWS IoT berbasis. Jika perangkat Anda dapat terhubung keAWS IoT,AWS IoT dapat menghubun gkan mereka ke layanan cloud yangAWS menyediakan. Systems Manager mendukung perangkatAWS IoT inti selama perangkat tersebut dikonfigurasi sebagai node terkelola di lingkungan hybrid dan multicloud. Pelajari selengkapnya Menyiapkan Manajer Sistem untuk lingkungan hybrid dan multicloud

Penyimpanan

Amazon Simple Storage Service (Amazon S3)	Amazon S3 adalah penyimpanan untuk Internet. Ia dirancang untuk mempermudah komputasi skala web bagi developer. Amazon S3 memiliki antarmuka layanan web sederhana yang dapat Anda gunakan untuk menyimpan dan mengambil data dalam jumlah berapa pun, kapan pun, dari mana pun di web. Systems Manager memungkinkan Anda untuk menjalankan skrip jarak jauh dan dokumen SSM yang disimpan di Amazon S3. Distributor,

	sebuah kemampuanAWS Systems Manager, menggunakan Amazon S3 untuk menyimpan paket. Anda juga dapat mengirim output ke Amazon S3 untukRun Command danSession Manager, sebuah kemampuanAWS Systems Manager. Pelajari selengkapnya • Menjalankan skrip dari Amazon S3 • Menjalankan dokumen dari lokasi terpencil • AWS Systems Manager Distributor • Log data sesi menggunakan Amazon S3 (konsol)

Alat Developer

AWS CodeBuild	CodeBuildadalah layanan pembangun an terkelola penuh di cloud. CodeBuild mengumpulkan kode sumber Anda, menjalank an pengujian unit, dan menghasilkan Artifact yang siap di-deploy. CodeBuildmenghilangkan kebutuhan untuk menyediakan, mengelola, dan menskalakan server pembangunan Anda sendiri. Parameter Storememungkinkan Anda untuk menyimpan informasi sensitif untuk spesifikasi dan proyek pembangunan Anda. Pelajari selengkapnya • Membangun referensi spesifikasi untukCode Build • Membuat proyek pembangunan diAWS CodeBuild

AWS CDK	AWS Cloud Development Kit (AWS CDK)Ini adalah kerangka kerja untuk mendefinisikan infrastruktur cloud sebagai kode, dengan bahasa pemrograman, dan menerapka nnyaAWS CloudFormation. Application Managermemungkinkan Anda untuk melihat konstruksi CDK Anda dikelompo kkan sebagai aplikasi, melihat struktur aplikasi termasuk sumber daya yang mendasarinya, melihat peringatan, menyelidiki dan memulihka n masalah operasional, dan melacak biaya diApplication Manager konsol. Pelajari selengkapnya • Melihat informasi gambaran umum tentang aplikasi • Menampilkan sumber daya aplikasi

Keamanan, Identitas, dan Kepatuhan

AWS Identity and Access Management (IAM)	IAM adalah layanan web yang membantu Anda mengendalikan akses ke sumber daya AWS secara aman. Anda menggunakan IAM untuk mengontrol siapa yang dapat terautentikasi (masuk) dan berwenang (memiliki izin) untuk menggunakan sumber daya. Systems Manager memungkinkan Anda untuk mengendalikan akses ke layanan menggunak an IAM. Pelajari selengkapnya • Cara kerja AWS Systems Manager dengan IAM

	• Tindakan, sumber daya, dan kunci ketentuan untukAWS Systems Manager • Mengonfigurasi izin instans untuk Systems Manager
AWS Secrets Manager	Secrets Manager menyediakan pengelola an rahasia yang lebih mudah. Rahasia dapat berupa kredensial basis data, kata sandi, kunci API pihak ke tiga, dan bahkan teks acak. Parameter Storememungkinkan Anda untuk mengambil rahasia Secrets Manager saat menggunakan lainnyaLayanan AWS yang sudah mendukung referensi keParameter Store parameter. Pelajari selengkapnya MerujukAWS Secrets Manager rahasia dariParameter Store parameter

AWS Security Hub	Security Hub memberi Anda pandangan komprehensif tentang pemberitahuan keamanan prioritas tinggi dan status kepatuhan Anda di seluruh Akun AWS. Security Hub mengumpulkan, mengatur, dan mempriori taskan pemberitahuan atau temuan keamanan Anda, dari beberapaLayanan AWS. Ketika Anda mengaktifkan integrasi antara Security Hub danPatch Manager, sebuah kemampuanAWS Systems Manager, Security Hub memantau status patch armada Anda dari sudut pandang keamanan. Detail kepatuhan patch secara otomatis diekspor ke Security Hub. Hal ini memungkinkan Anda untuk menggunakan satu tampilan untuk memantau status kepatuhan patch Anda secara terpusat dan melacak temuan keamanan lainnya. Anda dapat menerima pemberitahuan ketika node di armada Anda keluar dari kepatuhan patch dan meninjau temuan kepatuhan patch di konsol Security Hub. Anda juga dapat mengintegrasikan Security Hub denganExplorer danOpsCenter, kemampuanAWS Systems Manager. Integrasi dengan Security Hub memungkinkan Anda menerima temuan dari Security Hub diExplore r danOpsCenter. Temuan Security Hub menyediakan informasi keamanan yang dapat Anda gunakan diExplorer danOpsCenter untuk mengumpulkan serta mengambil tindakan pada masalah keamanan, performa, dan operasional AndaAWS Systems Manager.

	Ada biaya atas penggunaan Security Hub. Untuk informasi lebih lanjut, lihat Harga Security Hub. Pelajari selengkapnya • Menerima temuan dariAWS Security HubdiExplorer • AWS Security Hub • Integrasi dengan Patch ManagerAWS Security Hub

Kriptografi dan PKI

AWS Key Management Service (AWS KMS)	AWS KMS adalah layanan terkelola yang memungkinkan Anda untuk membuat dan mengendalikan kunci terkelola konsumen, kunci enkripsi yang digunakan untuk mengenkripsi data Anda. Systems Manager memungkinkan Anda untuk menggunakanAWS KMS untuk membuat`SecureString` parameter dan mengenkripsi dataSession Manager sesi. Pelajari selengkapnya • BagaimanaAWS Systems ManagerParameter Store menggunakanAWS KMS • Aktifkan enkripsi kunci KMS data sesi (konsol)

Pengelolaan dan Tata Kelola

AWS CloudFormation	AWS CloudFormation adalah layanan yang membantu Anda membentuk dan menyiapkan sumber daya Amazon Web Services sehingga Anda dapat menghabiskan lebih sedikit waktu untuk mengelola sumber daya tersebut dan lebih banyak waktu untuk berfokus pada aplikasi Anda yang berjalan di AWS. Parameter Storeadalah sumber untuk referensi dinamis. Referensi dinamis menyediakan cara yang ringkas dan ampuh bagi Anda untuk menentukan nilai eksternal yang disimpan dan dikelola dalam layanan lainnya di templat tumpukan AWS CloudFormation Anda. Pelajari selengkapnya Penggunaan referensi dinamis untuk menentukan nilai templat
AWS CloudTrail	CloudTrailadalah sebuahLayanan AWS yang membantu Anda mengotorisasi tata kelola, kepatuhan, serta audit operasional dan risiko dari AndaAkun AWS. Tindakan yang diambil oleh pengguna, peran, atau sebuahLayanan AWS dicatat sebagai acara diCloudTrail. Acara mencakup tindakan yang diambil di SDK dan API AWS Management Console, AWS Command Line Interface (AWS CLI), dan AWS. Systems Manager terintegrasi denganClo udTrailwhich menangkap sebagian besar panggilan Systems ManagerAPI sebagai peristiwa. Hal ini termasuk panggilan API yang dimulai dari konsol Systems Manager dan

	panggilan yang dilakukan ke API Systems Manager. Pelajari selengkapnya Pencatatan panggilan AWS Systems Manager API dengan AWS CloudTrail
Amazon CloudWatch Logs	AmazonCloudWatch Logs memungkinkan Anda memusatkan log dari semua sistem, aplikasi, danLayanan AWS yang Anda gunakan. Anda kemudian dapat melihatnya, mencarinya untuk kode atau pola kesalahan tertentu, memfilternya berdasarkan bidang tertentu, atau mengarsipkannya dengan aman untuk analisis di masa mendatang. Systems Manager mendukung pengiriman log untukSSM Agent,Run Command, danSession Manager untukCloudWatch Log. Pelajari selengkapnya • Mengirim log simpul ke CloudWatch Log terpadu (CloudWatch agen) • MengonfigurasiCloudWatch Log Amazon untukRun Command • Data sesi logging menggunakan Amazon CloudWatch Logs (konsol)

Amazon EventBridge	EventBridgemenyampaikan pengaliran acara sistem mendekati waktu nyata yang menjelaskan perubahan dalam sumber daya Amazon Web Services. Dengan menggunakan aturan sederhana yang dapat Anda siapkan dengan cepat, Anda dapat mencocokkan acara dan merutekannya ke satu atau beberapa fungsi atau pengaliran target. EventBridgemenjadi sadar akan perubahan operasional yang terjadi. EventBridgemenanggapi perubahan operasional ini dan mengambil tindakan korektif yang diperlukan. Tindakan ini mencakup pengiriman pesan untuk menanggapi lingkungan, aktivasi fungsi, dan pemerolehan informasi keadaan. Systems Manager memiliki beberapa acara yang didukung denganEventBridge mengizinkan Anda untuk mengambil tindakan berdasarkan konten acara tersebut. Pelajari selengkapnya Pemantauan peristiwa Systems Manager dengan Amazon EventBridge **Note** Amazon EventBridge adalah cara terbaik untuk mengelola peristiwa Anda. CloudWatchEvents danEventBridge layanan dan API dasar yang sama, tetapiEventBridge menyediakan lebih banyak fitur. Perubahan yang Anda buat di salah satuCloudWatch atauEventBridge dicerminkan di setiap konsol. Untuk informasi lebih lanjut,

	lihat Panduan Pengguna Amazon EventBridge.
AWS Config	AWS Config menyediakan tampilan detail dari konfigurasi sumber daya AWS dalam Akun AWS. Anda. Ini mencakup cara sumber daya terkait satu sama lain dan cara sumber daya tersebut dikonfigurasi. Ini memungkinkan Anda untuk melihat cara konfigurasi dan hubungan berubah seiring waktu. Systems Manager berintegrasi dengan AWS Config, yang menyediakan beberapa aturan yang membantu Anda mendapatkan visibilit as ke dalam instans EC2 Anda. Aturan ini membantu Anda mengidentifikasi instans EC2 yang dikelola oleh Systems Manager, konfigura si sistem operasi, pembaruan tingkat sistem, aplikasi yang diinstal, konfigurasi jaringan, dan lain-lain. Pelajari selengkapnya • AWS Configjenis sumber daya yang didukung • Pencatatan konfigurasi perangkat lunak untuk instans terkelola • Melihat riwayat inventaris dan pelacakan perubahan

AWS Trusted Advisor	Trusted Advisor adalah alat online yang menyediakan panduan waktu nyata untuk membantu Anda menyediakan sumber daya dengan mengikuti praktik terbaik AWS. Systems Manager hostTrusted Advisor dan Anda dapat melihatTrusted Advisor data diExplorer. Pelajari selengkapnya • AWS Systems Manager Explorer • Memulai denganAWS Trusted Advisor

AWS Organizations	Organizations adalah layanan pengelolaan akun yang memungkinkan Anda untuk mengonsolidasikan beberapa Akun AWS ke dalam organisasi yang Anda buat dan kelola secara terpusat. Organizations mencakup pengelolaan akun dan kemampuan tagihan terkonsolidasi yang memungkinkan Anda untuk memenuhi kebutuhan anggaran, keamanan, dan kepatuhan akan bisnis Anda dengan lebih baik. Integrasi antara Change Manager, sebuah kemampuanAWS Systems Manager, dengan Organizations memungkinkan penggunaan akun administrator yang didelegasikan untuk mengelola permintaan perubahan, mengubah templat, dan persetujuan untuk keseluruhan organisasi Anda melalui akun tunggal ini. Integrasi Organizations dengan Inventory AWS Systems Manager, sebuah kemampuan , dan Explorermemungkinkan Anda untuk mengumpulkan inventaris dan data operasi (OpsData) dari beberapaWilayah AWS danAkun AWS. Integrasi antaraQuick Setup, sebuah kemampuanAWS Systems Manager, dan Organizations mengotomatiskan tugas penyiapan layanan umum, dan men-deploy konfigurasi layanan berdasarkan praktik terbaik di seluruh unit organisasi (OU) Anda.

Jaringan dan Pengiriman Konten

AWS PrivateLink	AWS PrivateLinkmemungkinkan Anda menghubungkan virtual private cloud (VPC) Anda secara privat ke layanan yang didukungL ayanan AWS dan layanan VPC endpoint tanpa memerlukan gateway internet, perangkat NAT, koneksi VPN, atauAWS Direct Connect koneksi. Systems Manager mendukung node terkelola yang menghubungkan ke API Systems Manager menggunakanAWS PrivateLink. Hal ini meningkatkan postur keamanan dari node terkelola Anda karenaAWS PrivateLink membatasi semua lalu lintas jaringan antara node terkelola Anda, Systems Manager, dan Amazon EC2 ke jaringan Amazon. Artinya node terkelola tidak harus memiliki akses ke internet. Pelajari selengkapnya Buat titik akhir VPC

Analitik

Amazon Athena	Athena adalah layanan kueri interaktif yang memungkinkan Anda untuk menganalisis data secara langsung di Amazon Simple Storage Service (Amazon S3) menggunakan SQL standar. Dengan beberapa tindakan di AWS Management Console, Anda dapat mengarahk an Athena pada data Anda yang disimpan di Amazon S3 dan mulai menggunakan SQL standar untuk menjalankan query satu kali dan mendapatkan hasil dalam hitungan detik.

	Systems Manager Inventory berintegrasi dengan Athena untuk membantu Anda mengkueri data inventaris dari beberapa Wilayah AWS dan Akun AWS. Integrasi Athena menggunakan sinkronisasi data sumber daya sehingga Anda dapat melihat data inventari s dari semua node terkelola pada halaman Tampilan Detail di konsol Systems Manager Inventory. Pelajari selengkapnya • Mengkueri data inventaris dari beberapa Wilayah dan akun • Panduan: Menggunakan sinkronisasi data sumber daya untuk mengumpulkan data inventaris
AWS Glue	AWS Glue adalah layanan ETL (extract, transform, and load) terkelola penuh yang membuat Anda dapat mengategorikan data, membersihkannya, memperkayanya, dan memindahkannya dengan andal antar berbagai penyimpanan data dan aliran data dengan sederhana dan hemat biaya. Systems Manager menggunakan AWS Glue untuk menarik data Inventory di bucket S3 Anda. Pelajari selengkapnya Mengkueri data inventaris dari beberapa Wilayah dan akun

Amazon QuickSight	AmazonQuickSight adalah layanan analitik bisnis yang dapat Anda gunakan untuk membangun visualisasi, melakukan analisis satu kali, dan mendapatkan wawasan bisnis dari data Anda. Ia secara otomatis dapat menemukan sumber data AWS dan juga sesuai untuk sumber data Anda. Sinkronisasi data sumber daya Systems Manager mengirim data inventaris yang dikumpulkan dari semua node terkelola Anda ke satu bucket S3. Anda dapat menggunak an AmazonQuickSight untuk mengkueri dan menganalisis data yang dikumpulkan. Pelajari selengkapnya • Pengonfigurasian sinkronisasi data sumber daya untuk Inventaris • Panduan: Menggunakan sinkronisasi data sumber daya untuk mengumpulkan data inventaris

Integrasi Aplikasi

Amazon Simple Notification Service (Amazon SNS)	Amazon SNS adalah layanan web yang mengoordinasikan dan mengelola penyampai an atau pengiriman pesan ke titik akhir atau klien yang berlangganan. Systems Manager menghasilkan status untuk beberapa layanan yang dapat diperoleh oleh notifikasi Amazon SNS.

	Pelajari selengkapnya • Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS • Menyiapkan notifikasi atau memicu tindakan berdasarkan Parameter Store peristiwa

AWS Management Console

AWS Resource Groups	Resource Groups mengorganisasi sumber daya AWS Anda. Resource groups mempermudah Anda untuk mengelola, memantau dan mengotomatiskan tugas pada sejumlah besar sumber daya secara sekaligus. Jenis sumber daya Systems Manager seperti node terkelola, dokumen SSM, windows pemeliharaan,Parameter Store parameter, dan dasar patch dapat ditambahkan ke resource groups. Pelajari selengkapnya Apa yangAWS Resource Groups?

Topik

- Menjalankan skrip dari Amazon S3
- MerujukAWS Secrets Manager rahasia dariParameter Store parameter
- Menggunakan Parameter Store parameter dalam AWS Lambda fungsi

Menjalankan skrip dari Amazon S3

Bagian ini menjelaskan cara mengunduh dan menjalankan skrip dari Amazon Simple Storage Service (Amazon S3). Topik berikut mencakup informasi dan terminologi yang berkaitan dengan Amazon

S3. Untuk mempelajari lebih lanjut tentang Amazon S3, lihat Apa itu Amazon S3? Anda dapat menjalankan berbagai jenis skrip, termasuk Ansible Playbooks, Python, Ruby, Shell, dan. PowerShell

Anda juga dapat mengunduh direktori yang mencakup beberapa skrip. Saat Anda menjalankan skrip utama di direktori, AWS Systems Manager juga menjalankan skrip referensi apa pun yang disertakan dalam direktori.

Perhatikan detail penting tentang menjalankan skrip dari Amazon S3 berikut ini:

- Systems Manager tidak memverifikasi bahwa skrip Anda mampu berjalan pada sebuah node. Sebelum Anda mengunduh dan menjalankan skrip, verifikasi bahwa perangkat lunak yang diperlukan diinstal pada node. Atau, Anda dapat membuat dokumen komposit yang menginstal perangkat lunak dengan menggunakan salah satu Run Command atauState Manager, kemampuan AWS Systems Manager, dan kemudian mengunduh dan menjalankan skrip.
- Verifikasi bahwa pengguna, peran, atau grup Anda telah diberikan izin AWS Identity and Access Management (IAM) yang diperlukan untuk membaca dari bucket S3.
- Pastikan profil instans pada instans Amazon Elastic Compute Cloud (Amazon EC2) Anda memiliki izin `s3:ListBucket` dan `s3:GetObject`. Jika profil instans tidak memiliki izin ini, sistem akan gagal mengunduh skrip Anda dari bucket S3. Untuk informasi lebih lanjut, lihat Menggunakan profil instance dalam Panduan Pengguna IAM.

Menjalankan skrip shell dari Amazon S3

Informasi berikut ini mencakup prosedur untuk membantu Anda menjalankan skrip dari Amazon Simple Storage Service (Amazon S3) dengan menggunakan AWS Systems Manager konsol atau (). AWS Command Line Interface AWS CLI Meskipun skrip shell digunakan dalam contoh, jenis skrip lain dapat diganti.

Menjalankan skrip shell dari Amazon S3 (konsol)

Menjalankan skrip shell dari Amazon S3

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Run Command.

3. Pilih Run Command.
4. Di daftar Dokumen perintah, pilih **`AWS-RunRemoteScript`**.
5. Di Parameter perintah, lakukan hal berikut:

   - Di Jenis Sumber, pilih S3.
   - Di kotak teks Info sumber, masukkan informasi yang diperlukan untuk mengakses sumber dalam format berikut. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri.

     > **Note**
     >
     > Ganti https://s3. *`aws-api-domain`*dengan URL untuk bucket Anda. Anda dapat menyalin URL bucket Anda di Amazon S3 pada tab Objects.

     ```
     {"path":"https://s3.aws-api-domain/path to script"}
     ```

     Berikut adalah contohnya.

     ```
     {"path":"https://mytestbucket.s3.us-west-2.amazonaws.com/scripts/shell/
     helloWorld.sh"}
     ```

   - Di bidang Baris Perintah, masukkan parameter untuk eksekusi skrip. Inilah contohnya.

     ```
     helloWorld.sh argument-1 argument-2
     ```

   - (Opsional) Di bidang Direktori Kerja, masukkan nama direktori pada node tempat Anda ingin mengunduh dan menjalankan skrip.
   - (Opsional) Di Batas Waktu Eksekusi, tentukan jumlah detik bagi sistem untuk menunggu sebelum menggagalkan eksekusi perintah skrip.
6. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

> **Tip**
>
> Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

7. Untuk Parameter lainnya:

   - Untuk Komentar, ketik informasi tentang perintah ini.
   - Untuk Waktu habis (detik), tentukan jumlah detik untuk menunggu sistem sebelum gagal menjalankan perintah keseluruhan.

8. Untuk Pengendalian rate:

   - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

     > **Note**
     >
     > Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

   - Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

9. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan milik pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang

ditentukan berbeda Akun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

10. Di bagian Notifikasi SNS, jika Anda ingin notifikasi dikirim tentang status eksekusi perintah, pilih kotak centang Aktifkan notifikasi SNS.

    Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Run Command Amazon SNS, lihat. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

11. Pilih Jalankan.

Jalankan skrip shell dari Amazon S3 (baris perintah)

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI.

2. Jalankan perintah berikut. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   > **Note**
   >
   > Ganti https://s3. *aws-api-domain*dengan URL untuk bucket Anda. Anda dapat menyalin URL bucket Anda di Amazon S3 pada tab Objects.

   Linux & macOS

```
aws ssm send-command \
    --document-name "AWS-RunRemoteScript" \
    --output-s3-bucket-name "bucket-name" \
    --output-s3-key-prefix "key-prefix" \
    --targets "Key=InstanceIds,Values=instance-id" \
    --parameters '{"sourceType":["S3"],"sourceInfo":["{\"path\":\"https://
s3.aws-api-domain/script path\"}"],"commandLine":["script name and arguments"]}'
```

Windows

```
aws ssm send-command ^
    --document-name "AWS-RunRemoteScript" ^
    --output-s3-bucket-name "bucket-name" ^
    --output-s3-key-prefix "key-prefix" ^
    --targets "Key=InstanceIds,Values=instance-id" ^
    --parameters "sourceType"="S3",sourceInfo='{\"path\":\"https://s3.aws-api-domain/script path\"}',"commandLine"="script name and arguments"
```

PowerShell

```
Send-SSMCommand `
    -DocumentName "AWS-RunRemoteScript" `
    -OutputS3BucketName "bucket-name" `
    -OutputS3KeyPrefix "key-prefix" `
    -Target @{Key="InstanceIds";Values=@("instance-id")} `
    -Parameter @{ sourceType="S3";sourceInfo='{"path": "https://s3.aws-api-domain/script path"}',; "commandLine"="script name and arguments"}
```

MerujukAWS Secrets Manager rahasia dariParameter Store parameter

AWS Secrets Managermembantu Anda mengorganisasi dan mengelola data konfigurasi penting seperti kredensyal, kata sandi, dan kunci lisensi. Parameter Store, sebuah kemampuanAWS Systems Manager, terintegrasi dengan Secrets Manager sehingga Anda dapat mengambil rahasia Secrets Manager saat menggunakan lainnyaLayanan AWS yang sudah mendukung referensi keParameter Store parameter. Layanan ini mencakup Amazon Elastic Compute Cloud (Amazon EC2), Amazon Elastic Container Service (Amazon ECS), AWS Lambda, AWS CloudFormation, AWS CodeBuild, AWS CodeDeploy, dan kemampuan Systems Manager lainnya. DenganParameter Store mereferensikan rahasia Secrets Manager, Anda membuat proses yang konsisten dan aman untuk memanggil dan menggunakan rahasia dan referensi data di skrip kode dan konfigurasi Anda.

Untuk informasi lebih lanjut tentang Secrets Manager, lihat Apa Itu AWS Secrets Manager? di Panduan Pengguna AWS Secrets Manager.

Pembatasan

Perhatikan pembatasan berikut saat menggunakan rahasia Secrets ManagerParameter Store untuk mereferensikan rahasia Secrets Manager:

- Anda hanya dapat mengambil Secrets Manager dengan menggunakan GetParameterdan operasi GetParametersAPI. Operasi modifikasi dan operasi API pengkuerian tingkat lanjut, seperti DescribeParametersdan GetParametersByPath, tidak didukung untuk Secrets Manager.
- Anda dapat menggunakanAWS Command Line Interface (AWS CLI)AWS Tools for Windows PowerShell,, dan SDK untuk mengambil rahasia dengan menggunakanParameter Store.
- Ketika Anda mengambil rahasia Secrets Manager dariParameter Store, nama harus dimulai dengan jalur pemesanan berikut: /aws/reference/secretsmanager/ Secretsmanager/ Secretsmanager/ Secretsmanager/ *Secrets* Manager.

  Ini contohnya: `/aws/reference/secretsmanager/CFCreds1`
- Parameter Storemenghormati kebijakanAWS Identity and Access Management (IAM) yang melekat pada rahasia Secrets Manager. Misalnya, jika Pengguna 1 tidak memiliki akses ke Rahasia A, maka Pengguna 1 tidak dapat mengambil Rahasia A dengan menggunakanParameter Store.
- Parameter yang mereferensikan rahasia Secrets Manager tidak dapat menggunakanParameter Store versioning atau fitur riwayat.
- Parameter Storemenghormati tahapan versi Secrets Manager. Jika Anda mereferensikan tahapan versi, ia menggunakan huruf, angka, titik (.), tanda hubung (-), atau garis bawah (_). Semua simbol lain yang ditentukan dalam tahapan versi menyebabkan referensi gagal.

Cara mereferensikan rahasia Secrets Manager dengan menggunakanParameter Store

Prosedur berikut menjelaskan cara mereferensikan rahasia Secrets Manager dengan menggunakanParameter Store API. Prosedur mereferensikan prosedur lainnya di Panduan Pengguna AWS Secrets Manager.

> **Note**
>
> Sebelum Anda memulai, pastikan Anda memiliki izin untuk mereferensikan rahasia Secrets Manager diParameter Store parameter. Jika Anda memiliki izin administrator di Secrets Manager dan Systems Manager, maka Anda dapat mereferensikan atau mengambil rahasia dengan menggunakanParameter Store API API. Jika Anda mereferensikan rahasia Secrets Manager diParameter Store parameter, dan Anda tidak memiliki izin untuk mengakses rahasia itu, maka referensi akan gagal. Untuk informasi lebiih lanjut, lihat Autentikasi dan pengendalian akses untuk AWS Secrets Manager di Panduan Pengguna AWS Secrets Manager.

> **Important**
>
> Parameter Storeberfungsi sebagai layanan pemintasan untuk referensi pada rahasia Secrets Manager. Parameter Storetidak mempertahankan data atau metadata tentang rahasia. Referensi bersifat stateless.

Mereferensikan rahasia Secrets Manager dengan menggunakanParameter Store

1. Buat rahasia di Secrets Manager. Untuk informasi selengkapnya, lihat Membuat dan mengelola rahasia denganAWS Secrets Manager.
2. Referensikan rahasia dengan menggunakan AWS CLI, AWS Tools for Windows PowerShell, atau SDK. Ketika Anda mereferensikan rahasia Secrets Manager, nama harus dimulai dengan jalur pemesanan berikut: `/aws/reference/secretsmanager/`. Dengan menentukan jalur ini, Systems Manager mengetahui bahwa rahasia diambil dari Secrets Manager dan bukanParameter Store. Berikut adalah beberapa contoh nama yang mereferensikan rahasia Secrets Manager dengan benar,`CFCreds1` dan`DBPass`, dengan benarParameter Store.

   - `/aws/reference/secretsmanager/CFCreds1`
   - `/aws/reference/secretsmanager/DBPass`

   Berikut ini adalah contoh kode Java yang mereferensikan access key dan kunci rahasia yang disimpan dalam Secrets Manager. Contoh kode ini menyiapkan klien Amazon DynamoDB. Kode tersebut mengambil data konfigurasi dan kredensi dariParameter Store. Data konfigurasi disimpan sebagai parameter string diParameter Store dan kredensyal disimpan di Secrets Manager. Meskipun data konfigurasi dan kredensyal disimpan di layanan yang terpisah, kedua kumpulan data dapat diaksesParameter Store dengan menggunakan`GetParameter` API.

```
/**
* 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();
}
```

Berikut ini adalah beberapa contoh AWS CLI. Gunakan perintah `aws secretsmanager list-secrets` untuk menemukan nama-nama rahasia Anda.

AWS CLIContoh 1: Referensi dengan menggunakan nama rahasia

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
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
{
    "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 CLIContoh 2: Referensi yang mencakup ID versi

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
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
{
    "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 CLIContoh 3: Referensi yang mencakup tahapan versi

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
```

Perintah tersebut mengembalikan informasi seperti berikut.

```
{
    "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",
}
```

Menggunakan Parameter Store parameter dalam AWS Lambda fungsi

Parameter Store, kemampuan AWS Systems Manager, menyediakan penyimpanan hierarkis yang aman untuk manajemen data konfigurasi dan manajemen rahasia. Anda dapat menyimpan data seperti kata sandi, string basis data, ID Amazon Machine Image (AMI), dan kode lisensi sebagai nilai parameter.

Untuk menggunakan parameter dari Parameter Store dalam AWS Lambda fungsi tanpa menggunakan SDK, Anda dapat menggunakan AWS Parameter dan Rahasia Ekstensi Lambda. Ekstensi ini mengambil nilai parameter dan menyimpannya di cache untuk penggunaan di masa mendatang. Menggunakan ekstensi Lambda dapat mengurangi biaya Anda dengan mengurangi jumlah panggilan API ke. Parameter Store Menggunakan ekstensi juga dapat meningkatkan latensi karena mengambil parameter yang di-cache lebih cepat daripada mengambilnya. Parameter Store

Ekstensi Lambda adalah proses pendamping yang menambah kemampuan fungsi Lambda. Ekstensi seperti klien yang berjalan secara paralel dengan pemanggilan Lambda. Klien paralel ini dapat berinteraksi dengan fungsi Anda kapan saja selama siklus hidupnya. Untuk informasi selengkapnya tentang ekstensi Lambda, lihat API Ekstensi Lambda di Panduan Pengembang.AWS Lambda

AWS Parameter dan Rahasia Lambda Extension berfungsi untuk keduanya danParameter Store. AWS Secrets Manager Untuk mempelajari cara menggunakan ekstensi Lambda dengan rahasia dari Secrets Manager, lihat Menggunakan AWS Secrets Manager rahasia dalam AWS Lambda fungsi di AWS Secrets Manager Panduan Pengguna.

Info terkait

Menggunakan ekstensi AWS Parameter dan Rahasia Lambda untuk parameter cache dan rahasia (AWS Compute Blog)

Cara kerja ekstensi

Untuk menggunakan parameter dalam fungsi Lambda tanpa ekstensi Lambda, Anda harus mengonfigurasi fungsi Lambda Anda untuk menerima pembaruan konfigurasi dengan mengintegrasikan dengan tindakan API untuk. `GetParameter` Parameter Store

Saat Anda menggunakan Ekstensi Lambda AWS Parameter dan Rahasia, ekstensi mengambil nilai parameter dari Parameter Store dan menyimpannya di cache lokal. Kemudian, nilai cache digunakan untuk pemanggilan lebih lanjut sampai kedaluwarsa. Nilai cache kedaluwarsa setelah melewati time-to-live (TTL) mereka. Anda dapat mengonfigurasi nilai TTL menggunakan variabel `SSM_PARAMETER_STORE_TTL` lingkungan, seperti yang dijelaskan nanti dalam topik ini.

Jika cache TTL yang dikonfigurasi belum kedaluwarsa, nilai parameter cache digunakan. Jika waktu telah kedaluwarsa, nilai cache tidak valid dan nilai parameter diambil dari. Parameter Store

Selain itu, sistem mendeteksi nilai parameter yang sering digunakan dan mempertahankannya di cache sambil membersihkan nilai yang kedaluwarsa atau tidak digunakan.

Detail implementasi

Gunakan detail berikut untuk membantu Anda mengonfigurasi Ekstensi Lambda AWS Parameter dan Rahasia.

Autentikasi

Untuk mengotorisasi dan mengautentikasi Parameter Store permintaan, ekstensi menggunakan kredensil yang sama seperti yang digunakan untuk menjalankan fungsi Lambda itu sendiri. Oleh karena itu, peran AWS Identity and Access Management (IAM) yang digunakan untuk menjalankan fungsi harus memiliki izin berikut untuk berinteraksi dengan: Parameter Store

- `ssm:GetParameter`— Diperlukan untuk mengambil parameter dari Parameter Store
- `kms:Decrypt`— Diperlukan jika Anda mengambil `SecureString` parameter dari Parameter Store

Untuk informasi selengkapnya, lihat peran AWS Lambda eksekusi di Panduan AWS Lambda Pengembang.

Instantiasi

Lambda membuat instance terpisah yang sesuai dengan tingkat konkurensi yang dibutuhkan fungsi Anda. Setiap instance diisolasi dan memelihara cache lokal sendiri dari data konfigurasi Anda. Untuk informasi selengkapnya tentang instans dan konkurensi Lambda, lihat Mengonfigurasi konkurensi cadangan di Panduan Pengembang.AWS Lambda

Tidak ada ketergantungan SDK

Ekstensi Lambda AWS Parameter dan Rahasia bekerja secara independen dari pustaka bahasa AWS SDK apa pun. AWS SDK tidak diperlukan untuk membuat permintaan GET keParameter Store.

Localhostpelabuhan

Gunakan `localhost` dalam permintaan GET Anda. Ekstensi membuat permintaan ke localhost port 2773. Anda tidak perlu menentukan endpoint eksternal atau internal untuk menggunakan ekstensi. Anda dapat mengkonfigurasi port dengan mengatur variabel lingkunganPARAMETERS_SECRETS_EXTENSION_HTTP_PORT.

Misalnya, dengan Python, URL GET Anda mungkin terlihat seperti contoh berikut.

```
parameter_url = ('http://localhost:' + port + '/systemsmanager/parameters/get/?
name=' + ssm_parameter_path)
```

Perubahan nilai parameter sebelum TTL kedaluwarsa

Ekstensi tidak mendeteksi perubahan pada nilai parameter dan tidak melakukan penyegaran otomatis sebelum TTL kedaluwarsa. Jika Anda mengubah nilai parameter, operasi yang menggunakan nilai parameter cache mungkin gagal hingga cache disegarkan berikutnya. Jika Anda mengharapkan perubahan yang sering terjadi pada nilai parameter, sebaiknya setel nilai TTL yang lebih pendek.

Persyaratan header

Untuk mengambil parameter dari cache ekstensi, header permintaan GET Anda harus menyertakan `X-Aws-Parameters-Secrets-Token` referensi. Setel token ke`AWS_SESSION_TOKEN`, yang disediakan oleh Lambda untuk semua fungsi yang berjalan. Menggunakan header ini menunjukkan bahwa penelepon berada dalam lingkungan Lambda.

Contoh

Contoh berikut di Python menunjukkan permintaan dasar untuk mengambil nilai parameter cache.

```
import urllib.request
import os
import json

aws_session_token = os.environ.get('AWS_SESSION_TOKEN')

def lambda_handler(event, context):
    # Retrieve /my/parameter from Parameter Store using extension cache
    req = urllib.request.Request('http://localhost:2773/systemsmanager/parameters/
get?name=%2Fmy%2Fparameter')
    req.add_header('X-Aws-Parameters-Secrets-Token', aws_session_token)
    config = urllib.request.urlopen(req).read()

    return json.loads(config)
```

Dukungan ARM

Ekstensi tidak mendukung arsitektur ARM sama di Wilayah AWS mana x86 arsitektur x86_64 dan didukung.

Untuk daftar lengkap ARN ekstensi, lihatAWS Parameter dan Rahasia Lambda Extension ARNs.

Pencatatan log

Lambda mencatat informasi eksekusi tentang ekstensi beserta fungsinya dengan menggunakan Amazon CloudWatch Logs. Secara default, ekstensi mencatat jumlah minimal informasi ke CloudWatch. Untuk mencatat detail lebih lanjut, atur variabel lingkungan `PARAMETERS_SECRETS_EXTENSION_LOG_LEVEL` ke`DEBUG`.

Menambahkan ekstensi ke fungsi Lambda

Untuk menggunakan AWS Parameter dan Rahasia Lambda Extension, Anda menambahkan ekstensi ke fungsi Lambda Anda sebagai lapisan.

Gunakan salah satu metode berikut untuk menambahkan ekstensi ke fungsi Anda.

AWS Management Console (Tambahkan opsi lapisan)

1. Buka AWS Lambda konsol di https://console.aws.amazon.com/lambda/.
2. Pilih fungsi Anda. Di area Layers, pilih Add a layer.
3. Di area Choose a layer, pilih opsi AWS layer.
4. Untuk AWS layer, pilih AWS-Parameters-and-Secrets-Lambda-Extension, pilih versi, lalu pilih Tambah.

AWS Management Console (Tentukan opsi ARN)

1. Buka AWS Lambda konsol di https://console.aws.amazon.com/lambda/.
2. Pilih fungsi Anda. Di area Layers, pilih Add a layer.
3. Di area Pilih lapisan, pilih opsi Tentukan ARN.
4. Untuk Tentukan ARN, masukkan ARN ekstensi untuk arsitektur Wilayah AWS dan Anda, lalu pilih Tambah.

AWS Command Line Interface

Jalankan perintah berikut di AWS CLI. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri.

```
aws lambda update-function-configuration \
    --function-name function-name \
    --layers layer-ARN
```

Informasi terkait

Menggunakan layer dengan fungsi Lambda Anda

Mengkonfigurasi ekstensi (arsip file.zip)

AWS Parameter dan Rahasia variabel lingkungan Ekstensi Lambda

Anda dapat mengonfigurasi ekstensi dengan mengubah variabel lingkungan berikut. Untuk melihat pengaturan saat ini, atur `PARAMETERS_SECRETS_EXTENSION_LOG_LEVEL` ke`DEBUG`. Untuk informasi selengkapnya, lihat Menggunakan variabel AWS Lambda lingkungan di Panduan AWS Lambda Pengembang.

> **Note**
>
> AWS Lambda mencatat detail operasi tentang ekstensi Lambda dan fungsi Lambda di Amazon Logs. CloudWatch

Variabel lingkungan	Detail	Dibutuhkan	Nilai valid	Nilai default
`SSM_PARAMETER_STORE_TIMEOUT_MILLIS`	Batas waktu, dalam milidetik, untuk permintaan ke. Parameter Store Nilai 0 (nol) menunjukkan tidak ada batas waktu.	Tidak	Semua bilangan bulat	0 (nol)
`SECRETS_MANAGER_TIMEOUT_MILLIS`	Batas waktu, dalam milidetik, untuk permintaan ke Secrets Manager.	Tidak	Semua bilangan bulat	0 (nol)

Variabel lingkungan	Detail	Dibutuhkan	Nilai valid	Nilai default
	Nilai 0 (nol) menunjukkan tidak ada batas waktu.			
SSM_PARAMETER_STORE_TTL	Masa pakai valid maksimum, dalam hitungan detik, dari parameter dalam cache sebelum tidak valid. Nilai 0 (nol) menunjukkan bahwa cache harus dilewati. Variabel ini diabaikan jika nilai untuk PARAMETERS_SECRETS_EXTENSION_CACHE_SIZE adalah 0 (nol).	Tidak	0 (nol) ke 300 s (Lima menit)	300 s (Lima menit)

Variabel lingkungan	Detail	Dibutuhkan	Nilai valid	Nilai default
SECRETS_MANAGER_TTL	Masa pakai valid maksimum, dalam hitungan detik, dari rahasia dalam cache sebelum tidak valid. Nilai 0 (nol) menunjukkan bahwa cache dilewati. Variabel ini diabaikan jika nilai untuk PARAMETERS_SECRETS_EXTENSION_CACHE_SIZE adalah 0 (nol).	Tidak	0 (nol) ke 300 s (Lima menit)	300 s (5 menit)
PARAMETERS_SECRETS_EXTENSION_CACHE_ENABLED	Menentukan apakah cache untuk ekstensi diaktifkan. Nilai nilai: TRUE | FALSE	Tidak	TRUE | FALSE	BETUL

Variabel lingkungan	Detail	Dibutuhkan	Nilai valid	Nilai default
PARAMETERS_SECRETS_EXTENSION_CACHE_SIZE	Ukuran maksimum cache dalam hal jumlah item. Nilai 0 (nol) menunjukkan bahwa cache dilewati. Variabel ini diabaikan jika kedua nilai cache TTL adalah 0 (nol).	Tidak	0 (nol) sampai 1000	1000
PARAMETERS_SECRETS_EXTENSION_HTTP_PORT	Port untuk server HTTP lokal.	Tidak	1 - 65535	2773

Variabel lingkungan	Detail	Dibutuhkan	Nilai valid	Nilai default
`PARAMETERS_SECRETS_EXTENSION_MAX_CONNECTIONS`	Jumlah maksimum koneksi untuk klien HTTP yang digunakan ekstensi untuk membuat permintaan Parameter Store atau Secrets Manager. Ini adalah konfigurasi per klien untuk jumlah koneksi yang dilakukan klien Secrets Manager dan Parameter Store klien ke layanan backend.	Tidak	Minimal 1; Tidak ada batas maksimum.	3

Variabel lingkungan	Detail	Dibutuhkan	Nilai valid	Nilai default
`PARAMETERS_SECRETS_EXTENSION_LOG_LEVEL`	Tingkat detail yang dilaporkan dalam log untuk ekstensi. Sebaiknya gunakan `DEBUG` untuk detail paling detail tentang konfigurasi cache Anda saat Anda mengatur dan menguji ekstensi. Log untuk operasi Lambda secara otomatis didorong ke grup CloudWatch log Log terkait.	Tidak	`DEBUG | WARN | ERROR | NONE | INFO`	`INFO`

Contoh perintah untuk menggunakan AWS Systems ManagerParameter Store dan AWS Secrets Manager Ekstensi

Contoh di bagian ini menunjukkan tindakan API untuk digunakan dengan AWS Secrets Manager ekstensi AWS Systems Manager Parameter Store dan.

Contoh perintah untuk Parameter Store

Ekstensi Lambda menggunakan akses hanya-baca ke tindakan API. GetParameter

Untuk memanggil tindakan ini, buat panggilan HTTP GET mirip dengan yang berikut ini.

```
GET http://localhost:port/systemsmanager/parameters/get?name=parameter-
path&version=version&label=label&withDecryption={true|false}
```

Dalam contoh ini, *parameter-path* mewakili nama parameter lengkap, atau jalur parameter jika parameter adalah bagian dari hierarki. *versi* dan *label* adalah pemilih yang tersedia untuk digunakan dengan `GetParameter` tindakan. Format perintah ini menyediakan akses ke parameter di tingkat parameter standar.

> **Note**
>
> Saat menggunakan panggilan GET, nilai parameter harus dikodekan untuk HTTP untuk mempertahankan karakter khusus. Misalnya, alih-alih memformat jalur hierarkis seperti`/a/b/c`, encode karakter yang dapat ditafsirkan sebagai bagian dari URL, seperti. `%2Fa%2Fb%2Fc`

```
GET http://localhost:port/systemsmanager/parameters/get/?name=MyParameter&version=5
```

Untuk memanggil parameter dalam hierarki, buat panggilan HTTP GET mirip dengan yang berikut ini.

```
GET http://localhost:port/systemsmanager/parameters/get?name=%2Fa%2Fb%2F&label=release
```

Untuk memanggil parameter publik (global), buat panggilan HTTP GET mirip dengan yang berikut ini.

```
GET http://localhost:port/systemsmanager/parameters/get/?name=%2Faws%2Fservice%20list
%2F…
```

Untuk membuat panggilan HTTP GET ke rahasia Secrets Manager dengan menggunakan Parameter Store referensi, buat panggilan HTTP GET mirip dengan berikut ini.

```
GET http://localhost:port/systemsmanager/parameters/get?name=%2Faws%2Freference
%2Fsecretsmanager%2F…
```

Untuk melakukan panggilan menggunakan Amazon Resource Name (ARN) untuk parameter, buat panggilan HTTP GET mirip dengan yang berikut ini.

```
GET http://localhost:port/systemsmanager/parameters/get?name=arn:aws:ssm:us-
east-1:123456789012:parameter/MyParameter
```

Untuk membuat panggilan yang mengakses `SecureString` parameter dengan dekripsi, buat panggilan HTTP GET mirip dengan yang berikut ini.

```
GET http://localhost:port/systemsmanager/parameters/get?
name=MyParameter&withDecryption=true
```

Anda dapat menentukan bahwa parameter tidak didekripsi dengan menghilangkan `withDecryption` atau secara eksplisit menyetelnya. `false` Anda juga dapat menentukan versi atau label, tetapi tidak keduanya. Jika Anda melakukannya, hanya yang pertama yang ditempatkan setelah tanda tanya (?) di URL yang digunakan.

AWS Parameter dan Rahasia Lambda Extension ARNs

Tabel berikut menyediakan ARN ekstensi untuk arsitektur dan Wilayah yang didukung.

Topik

ARN ekstensi untuk x86_64 dan arsitektur x86

Wilayah	ARN
AS Timur (Ohio)	`arn:aws:lambda:us-east-2:590474943231:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
AS Timur (Virginia Utara)	`arn:aws:lambda:us-east-1:177933569100:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
AS Barat (California Utara)	`arn:aws:lambda:us-west-1:997803712105:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`

Wilayah	ARN
AS Barat (Oregon)	`arn:aws:lambda:us-west-2:345057560386:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Afrika (Cape Town)	`arn:aws:lambda:af-south-1:317013901791:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Asia Pasifik (Hong Kong)	`arn:aws:lambda:ap-east-1:768336418462:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Wilayah Asia Pasifik (Hyderabad)	`arn:aws:lambda:ap-south-2:070087711984:layer:AWS-Parameters-and-Secrets-Lambda-Extension:8`
Asia Pasifik (Jakarta)	`arn:aws:lambda:ap-southeast-3:490737872127:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Asia Pasifik (Melbourne)	`arn:aws:lambda:ap-southeast-4:090732460067:layer:AWS-Parameters-and-Secrets-Lambda-Extension:1`
Asia Pasifik (Mumbai)	`arn:aws:lambda:ap-south-1:176022468876:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`

Wilayah	ARN
Asia Pasifik (Osaka)	`arn:aws:lambda:ap-northeast-3:576959938190:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Asia Pasifik (Seoul)	`arn:aws:lambda:ap-northeast-2:738900069198:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Asia Pasifik (Singapura)	`arn:aws:lambda:ap-southeast-1:044395824272:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Asia Pasifik (Sydney)	`arn:aws:lambda:ap-southeast-2:665172237481:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Asia Pasifik (Tokyo)	`arn:aws:lambda:ap-northeast-1:133490724326:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
(Canada (Central)	`arn:aws:lambda:ca-central-1:200266452380:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Kanada Barat (Calgary)	`arn:aws:lambda:ca-west-1:243964427225:layer:AWS-Parameters-and-Secrets-Lambda-Extension:1`

Wilayah	ARN
China (Beijing)	`arn:aws-cn:lambda:cn-north-1:287114880934:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Tiongkok (Ningxia)	`arn:aws-cn:lambda:cn-northwest-1:287310001119:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Eropa (Frankfurt)	`arn:aws:lambda:eu-central-1:187925254637:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Eropa (Irlandia)	`arn:aws:lambda:eu-west-1:015030872274:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Eropa (London)	`arn:aws:lambda:eu-west-2:133256977650:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Eropa (Milan)	`arn:aws:lambda:eu-south-1:325218067255:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Eropa (Paris)	`arn:aws:lambda:eu-west-3:780235371811:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`

Wilayah	ARN
Wilayah Eropa (Spanyol)	`arn:aws:lambda:eu-south-2:524103009944:layer:AWS-Parameters-and-Secrets-Lambda-Extension:8`
Eropa (Stockholm)	`arn:aws:lambda:eu-north-1:427196147048:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Israel (Tel Aviv)	`arn:aws:lambda:il-central-1:148806536434:layer:AWS-Parameters-and-Secrets-Lambda-Extension:1`
Wilayah Eropa (Zürich)	`arn:aws:lambda:eu-central-2:772501565639:layer:AWS-Parameters-and-Secrets-Lambda-Extension:8`
Timur Tengah (Bahrain)	`arn:aws:lambda:me-south-1:832021897121:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Timur Tengah (UEA)	`arn:aws:lambda:me-central-1:858974508948:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Amerika Selatan (Sao Paulo)	`arn:aws:lambda:sa-east-1:933737806257:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`

Wilayah	ARN
AWS GovCloud (AS-Timur)	`arn:aws-us-gov:lambda:us-gov-east-1:129776340158:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
AWS GovCloud (AS-Barat)	`arn:aws-us-gov:lambda:us-gov-west-1:127562683043:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`

ARN ekstensi untuk ARM64 dan arsitektur Mac with Apple silicon

Wilayah	ARN
AS Timur (Ohio)	`arn:aws:lambda:us-east-2:590474943231:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`
AS Timur (Virginia Utara)	`arn:aws:lambda:us-east-1:177933569100:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`
Wilayah AS Barat (California Utara)	`arn:aws:lambda:us-west-1:997803712105:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
AS Barat (Oregon)	`arn:aws:lambda:us-west-2:345057560386:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`

Wilayah	ARN
Wilayah Afrika (Cape Town)	`arn:aws:lambda:af-south-1:317013901791:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Wilayah Asia Pacific (Hong Kong)	`arn:aws:lambda:ap-east-1:768336418462:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Wilayah Asia Pasifik (Jakarta)	`arn:aws:lambda:ap-southeast-3:490737872127:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Asia Pasifik (Mumbai)	`arn:aws:lambda:ap-south-1:176022468876:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`
Asia Pasifik (Osaka)	`arn:aws:lambda:ap-northeast-3:576959938190:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Wilayah Asia Pasifik (Seoul)	`arn:aws:lambda:ap-northeast-2:738900069198:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Asia Pasifik (Singapura)	`arn:aws:lambda:ap-southeast-1:044395824272:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`

Wilayah	ARN
Asia Pasifik (Sydney)	`arn:aws:lambda:ap-southeast-2:665172237481:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`
Asia Pasifik (Tokyo)	`arn:aws:lambda:ap-northeast-1:133490724326:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`
Wilayah Kanada (Pusat)	`arn:aws:lambda:ca-central-1:200266452380:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Eropa (Frankfurt)	`arn:aws:lambda:eu-central-1:187925254637:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`
Eropa (Irlandia)	`arn:aws:lambda:eu-west-1:015030872274:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`
Eropa (London)	`arn:aws:lambda:eu-west-2:133256977650:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`
Wilayah Eropa (Milan)	`arn:aws:lambda:eu-south-1:325218067255:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`

Wilayah	ARN
Wilayah Eropa (Paris)	`arn:aws:lambda:eu-west-3:780235371811:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Wilayah Europe (Stockholm)	`arn:aws:lambda:eu-north-1:427196147048:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Wilayah Middle East (Bahrain)	`arn:aws:lambda:me-south-1:832021897121:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Wilayah Amerika Selatan (Sao Paulo)	`arn:aws:lambda:sa-east-1:933737806257:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`

Integrasi dengan produk dan layanan lainnya

AWS Systems Manager memiliki integrasi bawaan untuk produk dan layanan yang ditunjukkan pada tabel berikut.

Ansible	Ansibleadalah platform otomatisasi TI yang membuat aplikasi dan sistem Anda lebih mudah digunakan. Systems Manager menyediakan dokumen Systems Manager (dokumen SSM) `AWS-ApplyAnsiblePlaybooks` yang memungkinkan Anda membuat State Manager asosiasi yang menjalankan Ansible playbook.

	Pelajari selengkapnya Walkthrough: Membuat asosiasi yang menjalankan buku pedoman Ansible
Chef	Chefadalah alat otomatisasi TI yang membuat aplikasi dan sistem Anda lebih mudah digunakan. Systems Manager menyediakan dokumen `AWS-ApplyChefRecipes` SSM, yang memungkinkan Anda membuat asosiasi diState Manager, kemampuan AWS Systems Manager, yang menjalankan Chef resep. Pelajari selengkapnya Walkthrough: Membuat asosiasi yang menjalankan resep Chef Systems Manager juga terintegrasi dengan Chef InSpecprofil, memungkinkan Anda menjalankan pemindaian kepatuhan dan melihat node yang sesuai dan tidak sesuai. Pelajari selengkapnya Menggunakan Chef InSpec profil dengan Kepatuhan Systems Manager

GitHub	GitHubmenyediakan hosting untuk kontrol versi pengembangan perangkat lunak dan kolaborasi. Systems Manager menyediakan dokumen SSM`AWS-RunDocument` , yang memungkinkan Anda menjalankan dokumen SSM lain yang disimpanGitHub, dan dokumen SSM`AWS-RunRemoteScript` , yang memungkinkan Anda menjalankan skrip yang disimpan di dalamnya. GitHub Pelajari selengkapnya • Menjalankan dokumen dari lokasi terpencil • Menjalankan skrip dari GitHub
Jenkins	Jenkinsadalah server otomatisasi sumber terbuka yang memungkinkan pengembang untuk membangun, menguji, dan menyebarkan perangkat lunak mereka dengan andal. Otomatisasi, sebuah kemampuan Systems Manager, dapat digunakan sebagai langkah pasca-pembangunan untuk mem-pra-instal rilis aplikasi ke dalam Amazon Machine Images (AMIs). Pelajari selengkapnya Memperbarui AMIs menggunakan Otomasi dan Jenkins

ServiceNow	ServiceNowadalah sistem manajemen layanan perusahaan yang memungkinkan Anda mengelola layanan dan operasi TI Anda. OtomasiChange Manager,, Manajer InsidenOp sCenter, dan, semua kemampuan Systems Manager, terintegrasi ServiceNow dengan menggunakan Konektor Manajemen AWS Layanan. Dengan integrasi ini, Anda dapat melihat, membuat, memperbarui, menambahk an korespondensi, dan menyelesaikan AWS Support kasus dariServiceNow. Pelajari selengkapnya Berintegrasi dengan ServiceNow

Topik

- Menjalankan skrip dari GitHub
- Menggunakan Chef InSpec profil dengan Kepatuhan Systems Manager
- Berintegrasi dengan ServiceNow

Menjalankan skrip dari GitHub

Topik ini menjelaskan cara menggunakan dokumen Systems Manager (dokumen SSM) yang telah ditentukan sebelumnya `AWS-RunRemoteScript` untuk mengunduh skripGitHub, termasuk Ansible Playbooks, Python, Ruby, dan skrip. PowerShell Dengan menggunakan dokumen SSM ini, Anda tidak perlu lagi mem-port skrip secara manual ke Amazon Elastic Compute Cloud (Amazon EC2) atau membungkusnya dalam dokumen SSM. AWS Systems Manager integrasi dengan GitHub mempromosikan infrastruktur sebagai kode, yang mengurangi waktu yang diperlukan untuk mengelola node sambil menstandarisasi konfigurasi di seluruh armada Anda.

Anda juga dapat membuat dokumen SSM kustom yang memungkinkan Anda untuk mengunduh dan menjalankan skrip atau dokumen SSM lainnya dari lokasi berjarak jauh. Untuk informasi selengkapnya, lihat Membuat dokumen gabungan.

Anda juga dapat mengunduh direktori yang mencakup beberapa skrip. Ketika Anda menjalankan skrip utama di direktori, Systems Manager juga menjalankan skrip yang direferensikan yang disertakan dalam direktori.

Perhatikan detail penting berikut tentang menjalankan skrip dariGitHub.

- Systems Manager tidak memverifikasi bahwa skrip Anda mampu berjalan pada node. Sebelum Anda mengunduh dan menjalankan skrip, verifikasi bahwa perangkat lunak yang diperlukan diinstal pada node. Atau, Anda dapat membuat dokumen komposit yang menginstal perangkat lunak dengan menggunakan salah satu Run Command atauState Manager, kemampuan AWS Systems Manager, dan kemudian mengunduh dan menjalankan skrip.
- Anda bertanggung jawab untuk memastikan bahwa semua GitHub persyaratan terpenuhi. Ini termasuk pembaruan token akses Anda, sesuai kebutuhan. Pastikan Anda tidak melampaui jumlah permintaan yang terautentikasi atau tidak terautentikasi. Untuk informasi lebih lanjut, lihat GitHub dokumentasi.
- GitHub Enterpriserepositori tidak didukung.

Topik

Jalankan Ansible Playbooks dari GitHub

Bagian ini mencakup prosedur untuk membantu Anda menjalankan Ansible Playbook GitHub dengan menggunakan konsol atau AWS Command Line Interface (AWS CLI).

Sebelum Anda mulai

Jika Anda berencana untuk menjalankan skrip yang disimpan dalam GitHub repositori pribadi, buat AWS Systems Manager `SecureString` parameter untuk token akses GitHub keamanan Anda. Anda tidak dapat mengakses skrip di GitHub repositori pribadi dengan meneruskan token Anda secara manual melalui SSH. Token akses harus diteruskan sebagai parameter `SecureString` Systems Manager. Untuk informasi lebih lanjut tentang pembuatan parameter `SecureString`, lihat Menandai parameter Systems Manager.

Jalankan Ansible Playbook dari GitHub (konsol)

Jalankan Ansible Playbook dari GitHub

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Run Command.
3. Pilih Run Command.
4. Di daftar Dokumen perintah, pilih **`AWS-RunRemoteScript`**.
5. Di Parameter perintah, lakukan hal berikut:

   - Di Jenis Sumber, pilih GitHub.
   - Di kotak Info sumber, masukkan informasi yang diperlukan untuk mengakses sumber dalam format berikut.

   ```
   {
     "owner": "owner_name",
     "repository": "repository_name",
     "getOptions": "branch:branch_name",
     "path": "path_to_scripts_or_directory",
     "tokenInfo": "{{ssm-secure:SecureString_parameter_name}}"
   }
   ```

   Contoh ini mengunduh sebuah file bernama `webserver.yml`.

   ```
   {
       "owner": "TestUser1",
       "repository": "GitHubPrivateTest",
       "getOptions": "branch:myBranch",
       "path": "scripts/webserver.yml",
       "tokenInfo": "{{ssm-secure:mySecureStringParameter}}"
   }
   ```

> **Note**
>
> `"branch"` diperlukan hanya jika dokumen SSM Anda disimpan di cabang selain `master`.
> Untuk menggunakan versi skrip yang ada di melakukan tertentu di repositori anda, gunakan `commitID` dengan `getOptions` daripada `branch`. Sebagai contoh: `"getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",`

- Di bidang Baris Perintah, masukkan parameter untuk eksekusi skrip. Inilah contohnya.

  `ansible-playbook -i “localhost,” --check -c local webserver.yml`

- (Opsional) Di bidang Direktori Kerja, masukkan nama direktori pada node tempat Anda ingin mengunduh dan menjalankan skrip.
- (Opsional) Di Batas Waktu Eksekusi, tentukan jumlah detik bagi sistem untuk menunggu sebelum menggagalkan eksekusi perintah skrip.

6. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

> **Tip**
>
> Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

7. Untuk Parameter lainnya:

   - Untuk Komentar, ketik informasi tentang perintah ini.
   - Untuk Waktu habis (detik), tentukan jumlah detik untuk menunggu sistem sebelum gagal menjalankan perintah keseluruhan.

8. Untuk Pengendalian rate:

   - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

> **Note**
>
> Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

- Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

9. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan milik pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbeda Akun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

10. Di bagian Notifikasi SNS, jika Anda ingin notifikasi dikirim tentang status eksekusi perintah, pilih kotak centang Aktifkan notifikasi SNS.

    Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Run Command Amazon SNS, lihat. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

11. Pilih Jalankan.

Jalankan Ansible Playbook dari GitHub dengan menggunakan AWS CLI

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI.

2. Jalankan perintah berikut untuk mengunduh dan menjalankan skrip dariGitHub.

```
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"]}'
```

   Berikut adalah contoh perintah untuk dijalankan pada mesin Linux lokal.

```
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"]}'
```

Jalankan skrip Python dari GitHub

Bagian ini mencakup prosedur untuk membantu Anda menjalankan skrip Python GitHub dengan menggunakan AWS Systems Manager konsol atau (). AWS Command Line Interface AWS CLI

Jalankan skrip Python dari GitHub (konsol)

Jalankan skrip Python dari GitHub

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Run Command.

   -atau-

Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Run Command.

3. Pilih Run Command.
4. Di daftar Dokumen perintah, pilih **`AWS-RunRemoteScript`**.
5. Untuk Parameter perintah, lakukan hal berikut:

   - Di Jenis Sumber, pilih GitHub.
   - Di kotak Info sumber, masukkan informasi yang diperlukan untuk mengakses sumber dalam format berikut:

     ```
     {
         "owner": "owner_name",
         "repository": "repository_name",
         "getOptions": "branch:branch_name",
         "path": "path_to_document",
         "tokenInfo": "{{ssm-secure:SecureString_parameter_name}}"
     }
     ```

     Contoh berikut mengunduh direktori skrip bernama complex-script.

     ```
     {
         "owner": "TestUser1",
         "repository": "SSMTestDocsRepo",
         "getOptions": "branch:myBranch",
         "path": "scripts/python/complex-script",
         "tokenInfo": "{{ssm-secure:myAccessTokenParam}}"
     }
     ```

     > **Note**
     >
     > `"branch"` diperlukan hanya jika skrip Anda disimpan di cabang selain `master`. Untuk menggunakan versi skrip yang ada di melakukan tertentu di repositori anda, gunakan `commitID` dengan `getOptions` daripada `branch`. Sebagai contoh: `"getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",`

   - Untuk Baris Perintah, masukkan parameter untuk eksekusi skrip. Inilah contohnya.

```
mainFile.py argument-1 argument-2
```

Contoh ini menjalankan `mainFile.py`, yang kemudian dapat menjalankan skrip lainnya di direktori `complex-script`.

- (Opsional) Untuk Direktori Kerja, masukkan nama direktori pada node tempat Anda ingin mengunduh dan menjalankan skrip.
- (Opsional) Untuk Batas Waktu Eksekusi, tentukan jumlah detik bagi sistem untuk menunggu sebelum menggagalkan eksekusi perintah skrip.

6. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

> **Tip**
>
> Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

7. Untuk Parameter lainnya:

- Untuk Komentar, ketik informasi tentang perintah ini.
- Untuk Waktu habis (detik), tentukan jumlah detik untuk menunggu sistem sebelum gagal menjalankan perintah keseluruhan.

8. Untuk Pengendalian rate:

- Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

> **Note**
>
> Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

- Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda

menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.

9. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan milik pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbeda Akun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

10. Di bagian Notifikasi SNS, jika Anda ingin notifikasi dikirim tentang status eksekusi perintah, pilih kotak centang Aktifkan notifikasi SNS.

    Untuk informasi selengkapnya tentang mengonfigurasi notifikasi Run Command Amazon SNS, lihat. Pemantauan perubahan status Systems Manager menggunakan notifikasi Amazon SNS

11. Pilih Jalankan.

Jalankan skrip Python dari GitHub dengan menggunakan AWS CLI

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI.

2. Jalankan perintah berikut untuk mengunduh dan menjalankan skrip dariGitHub.

```
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"]}'
```

   Inilah contohnya.

```
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 "]}'
```

Contoh ini mengunduh direktori skrip bernama `complex-script`. Entri `commandLine` menjalankan `mainFile.py`, yang kemudian dapat menjalankan skrip lainnya di direktori `complex-script`.

Menggunakan Chef InSpec profil dengan Kepatuhan Systems Manager

AWS Systems Manager terintegrasi dengan Chef InSpec. Chef InSpecadalah kerangka pengujian open-source yang memungkinkan Anda membuat profil yang dapat dibaca manusia untuk disimpan atau GitHub Amazon Simple Storage Service (Amazon S3). Kemudian Anda dapat menggunakan Systems Manager untuk menjalankan pemindaian kepatuhan dan melihat node yang sesuai dan tidak sesuai. Profil adalah persyaratan keamanan, kepatuhan, atau kebijakan untuk lingkungan komputasi Anda. Misalnya, Anda dapat membuat profil yang melakukan pemeriksaan berikut saat memindai node Anda dengan Kepatuhan, kemampuan AWS Systems Manager:

- Periksa apakah port tertentu terbuka atau tertutup.
- Periksa apakah aplikasi tertentu sedang berjalan.
- Periksa apakah paket tertentu telah terinstal.
- Periksa kunci Windows Registry untuk properti tertentu.

Anda dapat membuat InSpec profil untuk instans Amazon Elastic Compute Cloud (Amazon EC2) dan server lokal atau mesin virtual (VM) yang Anda kelola dengan Systems Manager. Chef InSpecProfil sampel berikut memeriksa apakah port 22 terbuka.

```
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 mencakup kumpulan sumber daya yang membantu Anda menulis cek dan kontrol audit dengan cepat. InSpec menggunakan InSpec Domain-specific Language (DSL) untuk menulis kontrol ini di Ruby. Anda juga dapat menggunakan profil yang dibuat oleh komunitas InSpec pengguna yang besar. Misalnya, DevSec chef-os-hardeningproyek GitHub menyertakan lusinan profil untuk membantu Anda mengamankan node Anda. Anda dapat membuat dan menyimpan profil di GitHub atau Amazon S3.

Cara kerjanya

Berikut adalah cara kerja proses penggunaan InSpec profil dengan Kepatuhan:

1. Entah mengidentifikasi InSpec profil yang telah ditentukan yang ingin Anda gunakan, atau buat sendiri. Anda dapat menggunakan profil yang telah ditentukan GitHub untuk memulai. Untuk informasi tentang cara membuat InSpec profil Anda sendiri, lihat Chef InSpecProfil Chef.
2. Simpan profil di GitHub repositori publik atau pribadi, atau di bucket S3.
3. Jalankan Kepatuhan dengan InSpec profil Anda dengan menggunakan dokumen Systems Manager (dokumen SSM)`AWS-RunInspecChecks`. Anda dapat memulai pemindaian Kepatuhan dengan menggunakanRun Command, kemampuan AWS Systems Manager, untuk pemindaian sesuai permintaan, atau Anda dapat menjadwalkan pemindaian Kepatuhan reguler dengan menggunakanState Manager, kemampuan. AWS Systems Manager
4. Identifikasi node yang tidak sesuai dengan menggunakan Compliance API atau konsol Kepatuhan.

> **Note**
>
> Perhatikan informasi berikut.
>
> - Chefmenggunakan klien di node Anda untuk memproses profil. Anda tidak perlu menginstal klien. Ketika Systems Manager menjalankan dokumen SSM `AWS-RunInspecChecks`, sistem memeriksa apakah klien terinstal. Jika tidak, Systems Manager menginstal Chef klien selama pemindaian, dan kemudian menghapus instalasi klien setelah pemindaian selesai.
> - Menjalankan dokumen SSM`AWS-RunInspecChecks`, seperti yang dijelaskan dalam topik ini, menetapkan jenis entri kepatuhan `Custom:Inspec` ke setiap node yang

ditargetkan. Untuk menetapkan jenis kepatuhan ini, dokumen memanggil operasi PutComplianceItemsAPI.

Menjalankan pemindaian InSpec kepatuhan

Bagian ini mencakup informasi tentang cara menjalankan pemindaian InSpec kepatuhan menggunakan konsol Systems Manager dan AWS Command Line Interface (AWS CLI). Prosedur konsol menunjukkan cara mengkonfigurasi State Manager untuk menjalankan pemindaian. AWS CLI Prosedur ini menunjukkan cara mengkonfigurasi Run Command untuk menjalankan pemindaian.

Menjalankan pemindaian InSpec kepatuhan dengan State Manager (konsol)

Untuk menjalankan pemindaian InSpec kepatuhan State Manager dengan menggunakan AWS Systems Manager konsol

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih State Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih State Manager.
3. Pilih Buat asosiasi.
4. Di bagian Berikan detail asosiasi, masukkan nama.
5. Di daftar Dokumen, pilih **`AWS-RunInspecChecks`**.
6. Di daftar Versi dokumen, pilih Terbaru pada waktu aktif.
7. Di bagian Parameter, dalam daftar Jenis Sumber, pilih salah satu GitHubatau S3.

   Jika Anda memilih GitHub, masukkan jalur ke InSpec profil di GitHub repositori publik atau pribadi di bidang Info Sumber. Berikut adalah contoh jalur ke profil publik yang disediakan oleh tim Systems Manager dari lokasi berikut: https://github.com/awslabs/amazon-ssm/tree/master/Compliance/InSpec/PortCheck.

   ```
   {"owner":"awslabs","repository":"amazon-ssm","path":"Compliance/InSpec/
   PortCheck","getOptions":"branch:master"}
   ```

Jika Anda memilih S3, masukkan URL yang valid ke InSpec profil di bucket S3 di bidang Info Sumber.

Untuk informasi selengkapnya tentang cara Systems Manager terintegrasi dengan GitHub Amazon S3, lihat. Menjalankan skrip dari GitHub

8. Di bagian Target, pilih node terkelola tempat Anda ingin menjalankan operasi ini dengan menentukan tag, memilih instance atau perangkat tepi secara manual, atau menentukan grup sumber daya.

> **Tip**
>
> Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.

9. Di bagian Tentukan jadwal, gunakan pilihan pembangun jadwal untuk membuat jadwal yang menentukan kapan Anda menginginkan pemindaian kepatuhan dijalankan.
10. Untuk Pengendalian rate:

    - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.

    > **Note**
    >
    > Jika Anda memilih target dengan menentukan tag yang diterapkan pada node terkelola atau dengan menentukan grup AWS sumber daya, dan Anda tidak yakin berapa banyak node terkelola yang ditargetkan, maka batasi jumlah target yang dapat menjalankan dokumen pada saat yang sama dengan menentukan persentase.

    - Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Node terkelola yang masih memproses perintah mungkin juga mengirim kesalahan.
11. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih kotak Tuliskan output perintah ke bucket S3. Masukkan nama bucket dan prefiks (folder) di kotak.

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 adalah izin profil instans (untuk instans EC2) atau peran layanan IAM (mesin yang diaktifkan hibrida) yang ditetapkan ke instance, bukan milik pengguna IAM yang melakukan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager atau Membuat peran layanan IAM untuk lingkungan hibrid. Selain itu, jika bucket S3 yang ditentukan berbeda Akun AWS, pastikan bahwa profil instance atau peran layanan IAM yang terkait dengan node terkelola memiliki izin yang diperlukan untuk menulis ke bucket tersebut.

12. Pilih Buat Asosiasi. Sistem membuat asosiasi dan secara otomatis menjalankan pemindaian Kepatuhan.
13. Tunggu beberapa menit sampai pemindaian selesai, lalu pilih Kepatuhan di panel navigasi.
14. Dalam contoh terkelola yang sesuai, cari node di mana kolom Compliance Type adalah Custom:Inspec.
15. Pilih ID node untuk melihat detail status yang tidak sesuai.

Menjalankan pemindaian InSpec kepatuhan dengan Run Command (AWS CLI)

1. Instal dan konfigurasikan AWS Command Line Interface (AWS CLI), jika Anda belum melakukannya.

   Untuk selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI.
2. Jalankan salah satu perintah berikut untuk menjalankan InSpec profil dari salah satu GitHub atau Amazon S3.

   Perintah membawa parameter berikut:

   - SourceType: atau Amazon S3 GitHub
   - SourceInfo: URL ke folder profil baik InSpec di GitHub dalam atau ember S3. Folder harus berisi InSpec file dasar (*.yl) dan semua kontrol terkait (*.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"}"]}'
```

Inilah contohnya.

```
aws ssm send-command --document-name "AWS-RunInspecChecks" --targets
 '[{"Key":"tag:testEnvironment","Values":["webServers"]}]' --parameters
 '{"sourceType":["GitHub"],"getOptions":"branch:master","sourceInfo":["{\"owner\":
\"awslabs\", \"repository\":\"amazon-ssm\", \"path\": \"Compliance/InSpec/PortCheck
\"}"]}'
```

Amazon S3

```
aws ssm send-command --document-name "AWS-RunInspecChecks" --targets
 '[{"Key":"tag:tag_name","Values":["tag_value"]}]' --parameters'{"sourceType":
["S3"],"sourceInfo":["{\"path\":\"https://s3.aws-api-domain/DOC-EXAMPLE-
BUCKET/Inspec.yml_file\"}"]}'
```

Inilah contohnya.

```
aws ssm send-command --document-name "AWS-RunInspecChecks" --targets
 '[{"Key":"tag:testEnvironment","Values":["webServers"]}]' --
parameters'{"sourceType":["S3"],"sourceInfo":["{\"path\":\"https://s3.aws-api-
domain/DOC-EXAMPLE-BUCKET/InSpec/PortCheck.yml\"}"]}'
```

3. Jalankan perintah berikut untuk melihat ringkasan pemindaian Kepatuhan.

```
aws ssm list-resource-compliance-summaries --filters
 Key=ComplianceType,Values=Custom:Inspec
```

4. Jalankan perintah berikut untuk melihat detail node yang tidak sesuai.

```
aws ssm list-compliance-items --resource-ids node_ID --resource-type
 ManagedInstance --filters Key=DocumentName,Values=AWS-RunInspecChecks
```

Berintegrasi dengan ServiceNow

ServiceNowmenyediakan sistem manajemen layanan berbasis cloud untuk membuat dan mengelola alur kerja tingkat organisasi, seperti untuk layanan TI, sistem tiket, dan dukungan. Konektor Manajemen AWS Layanan terintegrasi ServiceNow dengan Systems Manager untuk menyediakan, mengelola, dan mengoperasikan AWS sumber daya dariServiceNow. Anda dapat menggunakan Konektor Manajemen AWS Layanan untuk berintegrasi ServiceNow dengan OtomasiChange Manager, Manajer InsidenOpsCenter, dan, semua kemampuan AWS Systems Manager.

Anda dapat melakukan tugas-tugas berikut menggunakanServiceNow:

- Jalankan pedoman otomatisasi dari Systems Manager.
- Melihat, memperbarui, dan menyelesaikan insiden dari Systems ManagerOpsItems.
- Melihat dan mengelola item operasional, seperti insiden, melalui Systems ManagerOpsCenter.
- Melihat dan menjalankan permintaan perubahan Systems Manager dari daftar templat perubahan yang telah disetujui sebelumnya.
- Kelola dan selesaikan insiden yang melibatkan aplikasi yang AWS dihosting dengan mengintegrasikan dengan Manajer Insiden.

> **Note**
>
> Untuk informasi tentang cara mengintegrasikan denganServiceNow, lihat Mengonfigurasi integrasi AWS layanan di Panduan Administrator Konektor Manajemen AWS Layanan.

Penandaan sumber daya Systems Manager

Tag adalah sebuah label yang Anda tetapkan ke sebuah sumber AWS. Setiap tag terdiri atas sebuah kunci dan sebuah nilai, yang keduanya Anda tentukan.

Tag memungkinkan Anda untuk mengategorikan sumber daya AWS Anda dengan berbagai cara, misalnya, berdasarkan tujuan, pemilik, atau lingkungan. Misalnya, jika Anda ingin mengatur dan mengelola sumber daya sesuai dengan apakah mereka digunakan untuk pengembangan atau produksi, Anda dapat menandai beberapa dari mereka dengan kunci `Environment` dan nilai `Production`. Anda kemudian dapat melakukan berbagai jenis kueri untuk sumber daya yang ditandai `"Key=Environment,Values=Production"`. Misalnya, Anda dapat menentukan serangkaian tanda untuk node yang dikelola akun Anda yang dapat membantu Anda melacak atau menargetkan node berdasarkan sistem operasi dan lingkungan operasi, sepertiSUSE Linux Enterprise Server dikelompokkan sebagai`development`,`staging`, dan`production`. Anda juga dapat melakukan operasi pada sumber daya dengan menentukan pasangan nilai-kunci ini dalam perintah Anda, seperti menjalankan skrip pembaruan pada semua node dalam kelompok atau meninjau status node tersebut.

Anda dapat menggunakan tag yang diterapkan pada sumber daya AWS Systems Manager dalam berbagai operasi. Misalnya, Anda dapat menargetkan hanya node terkelola yang ditandai dengan pasangan nilai-kunci tag tertentu ketika Anda jalankan perintah atau tetapkan target ke jendela pemeliharaan. Anda juga dapat membatasi akses ke sumber daya Anda berdasarkan tag yang diterapkan pada mereka.

Lebih jauh, Anda dapat membuat resource groups dengan menentukan tag yang sama untuk sumber daya AWSdari berbagai jenis, tidak hanya jenis yang sama. Setelah itu, Anda dapat menggunakan Resource Groups untuk melihat informasi tentang sumber daya dalam kelompok yang patuh dan bekerja dengan benar dan sumber daya yang memerlukan tindakan. Informasi yang Anda lihat berkaitan dengan semua jenis sumber daya AWS yang dapat ditambahkan ke resource group tidak hanya didukung jenis sumber daya Systems Manager. Untuk informasi lebih lanjut, lihat Apa itu AWS Resource Groups? dalam Panduan Pengguna AWS Resource Groups.

Sisa dari bab ini menjelaskan cara menambah dan menghapus tag dari sumber daya Systems Manager.

Topik

- Menandai asosiasi Systems Manager
- Otomatisasi penandaan
- Menandai dokumen Systems Manager
- Menandai jendela pemeliharaan
- Menandai node terkelola
- PenandaanOpsItems
- Menandai parameter Systems Manager
- Menandai dasar patch

Sumber daya Systems Manager yang dapat Anda beri label

Anda dapat menerapkan tag ke AWS Systems Manager sumber daya berikut:

- Asosiasi
- Otomatisasi
- Dokumen
- Jendela pemeliharaan
- Simpul terkelola
- OpsItems
- OpsMetadata
- Parameter
- Garis dasar patch

Anda dapat menambahkan masing-masing tipe ini, kecuali OpsItems danOpsMetadata, ke resource group.

Tergantung pada jenis sumber daya, Anda dapat menggunakan tag untuk mengidentifikasi sumber daya yang harus dimasukkan dalam operasi. Misalnya, Anda dapat menandai sekelompok simpul terkelola dan kemudian menjalankan tugas jendela pemeliharaan yang menargetkan hanya simpul dengan pasangan nilai-kunci.

Anda juga dapat membatasi akses pengguna ke jenis sumber daya ini dengan membuat kebijakan AWS Identity and Access Management (IAM) yang menentukan tag yang dapat diakses pengguna

dan melampirkan kebijakan untuk entitas IAM (pengguna, peran, atau grup). Berikut ini adalah beberapa contoh membatasi akses sumber daya menggunakan tag.

- Anda dapat menerapkan tag untuk serangkaian dokumen Systems Manager khusus (dokumen SSM) dan kemudian membuat dan menerapkan kebijakan IAM yang memberikan akses ke dokumen dengan tag tersebut tetapi tidak yang lain (atau yang melarang akses hanya ke dokumen tersebut).
- Anda dapat menetapkan tag untuk OpsItems kemudian membuat kebijakan IAM yang membatasi pengguna atau grup yang memiliki akses untuk melihat atau memperbarui sumber daya tersebut. Misalnya, direktur organisasi dapat diberikan akses penuh ke semuaOpsItems, tetapi pengembang perangkat lunak dan teknisi dukungan dapat diberikan akses hanya ke proyek atau segmen klien yang menjadi tanggung jawab mereka.
- Anda dapat menerapkan tag umum untuk sumber daya dari semua enam jenis yang didukung dan membuat kebijakan IAM yang memberikan akses ke hanya sumber daya tersebut, seperti `Key=Project,Value=ProjectA` atau `Key=Environment,Value=Development`. Anda bahkan dapat memberikan akses hanya ke sumber daya yang telah ditetapkan oleh kedua pasangan tag. Hal ini memungkinkan, misalnya, untuk membatasi pengguna untuk bekerja hanya dengan sumber daya untuk ProjectA di lingkungan Pengembangan.

Anda dapat menggunakan konsol Resource Groups Systems Manager, konsol untuk jenis sumber daya yang didukung (misalnya Maintenance Windows OpsCenter konsol atau konsol), AWS Command Line Interface (AWS CLI), danAWS Tools for PowerShell. Anda dapat menambahkan tag saat membuat atau memperbarui sebuah sumber daya. Misalnya, Anda dapat menggunakan AWS CLI add-tags-to-resourceperintah tersebut untuk menambahkan tag ke salah satu jenis sumber daya Systems Manager yang didukung setelah dibuat. Anda dapat menggunakan remove-tags-from-resourceperintah tersebut untuk menghapusnya.

Menandai asosiasi Systems Manager

Topik di bagian ini menjelaskan cara bekerja dengan tag pada State Manager asosiasi. State Manageradalah komponen dariAWS Systems Manager.

Topik

Membuat asosiasi dengan tag

Anda dapat menambahkan tag ke State Manager asosiasi saat Anda membuatnya dengan menggunakanAWS CLI. Menambahkan tag ke asosiasi saat Anda membuatnya menggunakan konsol Systems Manager tidak didukung. Untuk informasi, lihat Membuat asosiasi (baris perintah).

Menambahkan tag ke asosiasi yang ada

Gunakan prosedur berikut untuk menambahkan tag ke State Manager asosiasi yang ada dengan menggunakan baris perintah.

Topik

- Menambahkan tag ke asosiasi yang ada (AWS CLI)
- Menambahkan tag ke asosiasi yang ada (AWS Tools for PowerShell)

Menambahkan tag ke asosiasi yang ada (AWS CLI)

1. MenggunakanAWS CLI, jalankan perintah berikut untuk daftar asosiasi yang dapat Anda tag.

   ```
   aws ssm list-associations
   ```

   Perhatikan nama asosiasi yang ingin Anda tag.

2. Jalankan perintah berikut untuk menandai asosiasi. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   ```
   aws ssm add-tags-to-resource \
       --resource-type "Association" \
       --resource-id "association-ID" \
       --tags "Key=tag-key,Value=tag-value"
   ```

   Jika berhasil, perintah tidak memiliki output.

3. Jalankan perintah berikut untuk memverifikasi tag asosiasi.

   ```
   aws ssm list-tags-for-resource --resource-type "Association" --resource-id
    "association-ID"
   ```

Menambahkan tag ke asosiasi yang ada (AWS Tools for PowerShell)

1. Jalankan perintah berikut untuk daftar asosiasi yang dapat Anda tag.

```
Get-SSMAssociationList
```

2. Jalankan perintah berikut untuk menandai parameter. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

```
$tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
```

```
$tag.Key = "tag-key"
```

```
$tag.Value = "tag-value"
```

```
Add-SSMResourceTag `
    -ResourceType "Association" `
    -ResourceId "association-ID" `
    -Tag $tag `
    -Force
```

3. Jalankan perintah berikut untuk memverifikasi tag asosiasi.

```
Get-SSMResourceTag `
    -ResourceType "Association" `
    -ResourceId "association-ID"
```

Menghapus tag dari asosiasi

Anda dapat menggunakan baris perintah untuk menghapus tag dari State Manager asosiasi.

Menghapus tag dari asosiasi (baris perintah)

1. Menggunakan alat baris perintah pilihan Anda, jalankan perintah berikut untuk membuat daftar asosiasi di akun Anda.

Linux & macOS

```
aws ssm list-associations
```

Windows

```
aws ssm list-associations
```

PowerShell

```
Get-SSMAssociationList
```

Perhatikan nama asosiasi dari mana Anda ingin menghapus tag.

2. Jalankan perintah berikut untuk menghapus tag dari asosiasi. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

Linux & macOS

```
aws ssm remove-tags-from-resource \
    --resource-type "Association" \
    --resource-id "association-ID" \
    --tag-key "tag-key"
```

Windows

```
aws ssm remove-tags-from-resource ^
    --resource-type "Association" ^
    --resource-id "association-ID" ^
    --tag-key "tag-key"
```

PowerShell

```
Remove-SSMResourceTag
    -ResourceId "association-ID"
    -ResourceType "Association"
    -TagKey "tag-key
```

Jika berhasil, perintah tidak memiliki output.

3. Jalankan perintah berikut untuk memverifikasi tag asosiasi.

   Linux & macOS

   ```
   aws ssm list-tags-for-resource \
       --resource-type "Association" \
       --resource-id "association-ID"
   ```

   Windows

   ```
   aws ssm list-tags-for-resource ^
       --resource-type "Association" ^
       --resource-id "association-ID"
   ```

   PowerShell

   ```
   Get-SSMResourceTag `
       -ResourceType "Association" `
       -ResourceId "association-ID"
   ```

Otomatisasi penandaan

Topik dalam bagian ini menjelaskan cara bekerja dengan tag pada otomatisasi. Anda dapat menambahkan tag keAWS Systems Manager otomatisasi. Anda dapat menambahkan tag ke otomatisasi pada saat Anda memulai mereka baik dari konsol atau baris perintah, atau setelah mereka menjalankan dengan menggunakan baris perintah.

Menambahkan tag ke otomatisasi (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Pada panel navigasi, pilih Otomatisasi.

   -atau-

Jika halaman beranda AWS Systems Manager terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Otomatisasi.

3. Pilih runbook Otomasi yang ingin Anda jalankan.
4. Pilih Jalankan otomatisasi.
5. Di bagian Tag, pilih Edit, dan kemudian tambahkan satu pasangan nilai kunci atau lebih.
6. Pilih Simpan.

Menambahkan tag ke otomatisasi (baris perintah)

Menggunakan alat baris perintah pilihan Anda, jalankan perintah berikut untuk menambahkan tag ke otomatisasi saat dimulai. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri.

Linux & macOS

```
aws ssm start-automation-execution \
    --document-name DocumentName \
    --parameters ParametersRequiredByDocument \
    --tags "Key=ExampleKey,Value=ExampleValue"
```

Windows

```
aws ssm start-automation-execution ^
    --document-name DocumentName ^
    --parameters ParametersRequiredByDocument ^
    --tags "Key=ExampleKey,Value=ExampleValue"
```

PowerShell

```
$exampleTag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
$exampleTag.Key = "ExampleKey"
$exampleTag.Value = "ExampleValue"

Start-SSMAutomationExecution `
    -DocumentName DocumentName `
    -Parameter ParametersRequiredByDocument
```

```
    -Tag $exampleTag
```

1. Anda juga dapat tag setelah dijalankan dengan menggunakan alat baris perintah pilihan Anda. Jalankan perintah berikut untuk menambahkan tag ke otomatisasi. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

   Linux & macOS

   ```
   aws ssm add-tags-to-resource \
       --resource-type "Automation" \
       --resource-id "automation-execution-id" \
       --tags "Key=ExampleKey,Value=ExampleValue"
   ```

   Windows

   ```
   aws ssm add-tags-to-resource ^
       --resource-type "Automation" ^
       --resource-id "automation-execution-id" ^
       --tags "Key=ExampleKey,Value=ExampleValue"
   ```

   PowerShell

   ```
   $exampleTag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
   $exampleTag.Key = "ExampleKey"
   $exampleTag.Value = "ExampleValue"

   Add-SSMResourceTag `
       -ResourceType "Automation" `
       -ResourceId "automation-execution-id" `
       -Tag $exampleTag `
       -Force
   ```

   Jika berhasil, perintah tidak memiliki output.

2. Jalankan perintah berikut untuk memverifikasi tag otomatisasi.

   Linux & macOS

   ```
   aws ssm list-tags-for-resource \
       --resource-type "Automation" \
   ```

```
    --resource-id "automation-execution-id"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "Automation" ^
    --resource-id "automation-execution-id"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "Automation" `
    -ResourceId "automation-execution-id"
```

Menghapus tanda dari otomatisasi

Anda dapat menggunakan alat baris perintah untuk menghapus tag dari otomatisasi.

Menghapus tag dari otomatisasi (baris perintah)

1. Menggunakan alat baris perintah pilihan Anda, jalankan perintah berikut untuk menghapus tag dari otomatisasi. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

   Linux & macOS

   ```
   aws ssm remove-tags-from-resource \
       --resource-type "Automation" \
       --resource-id "automation-execution-id" \
       --tag-key "tag-key"
   ```

   Windows

   ```
   aws ssm remove-tags-from-resource ^
       --resource-type "Automation" ^
       --resource-id "automation-execution-id" ^
       --tag-key "tag-key"
   ```

PowerShell

```
Remove-SSMResourceTag `
    -ResourceId "automation-execution-id" `
    -ResourceType "Automation" `
    -TagKey "tag-key" `
    -Force
```

2. Jalankan perintah berikut untuk memverifikasi tag otomatisasi.

Linux & macOS

```
aws ssm list-tags-for-resource \
    --resource-type "Automation" \
    --resource-id "automation-execution-id"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "Automation" ^
    --resource-id "automation-execution-id"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "Automation" `
    -ResourceId "automation-execution-id"
```

Menandai dokumen Systems Manager

Topik dalam bagian ini menerangkan bagaimana bekerja dengan tag pada dokumen Systems Manager (dokumen SSM).

Topik

- Membuat dokumen dengan tag
- Menambahkan tag ke dokumen yang ada
- Menghapus tag dari dokumen SSM

Membuat dokumen dengan tag

Anda dapat menambahkan tag ke dokumen SSM khusus pada saat Anda membuatnya.

Untuk informasi, lihat topik berikut:

- Membuat dokumen SSM (konsol)
- Membuat dokumen SSM (baris perintah)

Menambahkan tag ke dokumen yang ada

Anda dapat menambahkan tag ke dokumen SSM khusus yang Anda miliki dengan menggunakan konsol Systems Manager atau baris perintah.

Topik

- Menambahkan tag ke sebuah dokumen SSM yang sudah ada (konsol)
- Menambahkan tag ke sebuah dokumen SSM yang sudah ada (baris perintah)

Menambahkan tag ke sebuah dokumen SSM yang sudah ada (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

   Jika halaman beranda AWS Systems Manager terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.
3. Pilih tab Dimiliki oleh saya.
4. Pilih nama dokumen untuk ditambahkan tag, dan kemudian pilih tab Detail.
5. Di bagian Tag, pilih Edit, dan kemudian tambahkan satu pasangan nilai kunci atau lebih.
6. Pilih Simpan.

Menambahkan tag ke sebuah dokumen SSM yang sudah ada (baris perintah)

Untuk menambahkan tag ke sebuah dokumen SSM yang sudah ada (baris perintah)

1. Menggunakan alat baris perintah pilihan Anda, jalankan perintah berikut untuk melihat daftar dokumen yang dapat Anda tandai.

   Linux & macOS

   ```
   aws ssm list-documents
   ```

   Windows

   ```
   aws ssm list-documents
   ```

   PowerShell

   ```
   Get-SSMDocumentList
   ```

   Perhatikan nama dokumen yang ingin Anda tandai.

2. Jalankan perintah berikut untuk menandai dokumen. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

   Linux & macOS

   ```
   aws ssm add-tags-to-resource \
       --resource-type "Document" \
       --resource-id "document-name" \
       --tags "Key=tag-key,Value=tag-value"
   ```

   Windows

   ```
   aws ssm add-tags-to-resource ^
       --resource-type "Document" ^
       --resource-id "document-name" ^
       --tags "Key=tag-key,Value=tag-value"
   ```

PowerShell

```
$tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
```

```
$tag.Key = "tag-key"
```

```
$tag.Value = "tag-value"
```

```
Add-SSMResourceTag `
    -ResourceType "Document" `
    -ResourceId "document-name" `
    -Tag $tag `
    -Force
```

Jika berhasil, perintah tidak memiliki output.

3. Jalankan perintah berikut untuk memverifikasi tag dokumen.

Linux & macOS

```
aws ssm list-tags-for-resource \
    --resource-type "Document" \
    --resource-id "document-name"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "Document" ^
    --resource-id "document-name"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "Document" `
    -ResourceId "document-name"
```

Menghapus tag dari dokumen SSM

Anda dapat menggunakan konsol Systems Manager atau baris perintah untuk menghapus tag dari dokumen SSM.

Topik

Menghapus tag dari dokumen SSM (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Dokumen.

   -atau-

   Jika halaman beranda AWS Systems Manager terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Dokumen di panel navigasi.
3. Pilih tab Dimiliki oleh saya.
4. Pilih nama dokumen untuk menghapus tag, dan kemudian pilih tab Detail.
5. Di bagian Tag, pilih Edit, lalu pilih Hapus di samping pasangan tag yang tidak lagi Anda butuhkan.
6. Pilih Simpan.

Menghapus tag dari dokumen SSM (baris perintah)

1. Menggunakan alat baris perintah pilihan Anda, jalankan perintah berikut untuk membuat daftar dokumen di akun Anda.

   Linux & macOS

   ```
   aws ssm list-documents
   ```

Windows

```
aws ssm list-documents
```

PowerShell

```
Get-SSMDocumentList
```

Catat nama dokumen yang ingin Anda hapus tagnya.

2. Jalankan perintah berikut untuk menghapus tag dari dokumen. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

Linux & macOS

```
aws ssm remove-tags-from-resource \
    --resource-type "Document" \
    --resource-id "document-name" \
    --tag-key "tag-key"
```

Windows

```
aws ssm remove-tags-from-resource ^
    --resource-type "Document" ^
    --resource-id "document-name" ^
    --tag-key "tag-key"
```

PowerShell

```
Remove-SSMResourceTag `
    -ResourceId "document-name" `
    -ResourceType "Document" `
    -TagKey "tag-key" `
    -Force
```

Jika berhasil, perintah tidak memiliki output.

3. Jalankan perintah berikut untuk memverifikasi tag dokumen.

Linux & macOS

```
aws ssm list-tags-for-resource \
    --resource-type "Document" \
    --resource-id "document-name"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "Document" ^
    --resource-id "document-name"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "Document" `
    -ResourceId "document-name"
```

Menandai jendela pemeliharaan

Topik dalam bagian ini menjelaskan cara bekerja dengan tag pada jendela pemeliharaan.

Topik

- Membuat jendela pemeliharaan dengan tag
- Menambahkan tag ke jendela pemeliharaan yang sudah ada
- Menghapus tag dari jendela pemeliharaan

Membuat jendela pemeliharaan dengan tag

Anda dapat menambahkan tag ke jendela pemeliharaan pada saat Anda membuatnya.

Untuk informasi, lihat topik berikut:

- Membuat jendela pemeliharaan (konsol)
- Tutorial: Membuat dan mengonfigurasi jendela pemeliharaan (AWS CLI)

Menambahkan tag ke jendela pemeliharaan yang sudah ada

Anda dapat menambahkan tag ke jendela pemeliharaan yang Anda miliki dengan menggunakan konsol AWS Systems Manager atau baris perintah.

Topik

Menambahkan tag ke jendela pemeliharaan yang sudah ada (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Maintenance Windows.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih pilih pilih pilih pilih pilih pilih Maintenance Windows.
3. Pilih nama jendela pemeliharaan yang telah Anda buat, lalu pilih tab Tag.
4. Pilih Edit tag, lalu pilih Tambahkan tag.
5. Untuk Kunci, masukkan kunci untuk tag, seperti **`Environment`**.
6. Untuk Nilai, masukkan nilai untuk tag, seperti **`Test`**.
7. Pilih Simpan perubahan.

Menambahkan tag ke jendela pemeliharaan yang sudah ada (AWS CLI)

1. Menggunakan alat baris perintah pilihan Anda, jalankan perintah berikut untuk melihat daftar jendela pemeliharaan yang dapat Anda tandai.

   ```
   aws ssm describe-maintenance-windows
   ```

   Perhatikan ID jendela pemeliharaan yang ingin Anda tandai.

2. Jalankan perintah berikut untuk menandai jendela pemeliharaan. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

   Linux & macOS

   ```
   aws ssm add-tags-to-resource \
       --resource-type "MaintenanceWindow" \
       --resource-id "window-id" \
       --tags "Key=tag-key,Value=tag-value"
   ```

   Windows

   ```
   aws ssm add-tags-to-resource ^
       --resource-type "MaintenanceWindow" ^
       --resource-id "window-id" ^
       --tags "Key=tag-key,Value=tag-value"
   ```

   Jika berhasil, perintah tidak memiliki output.

3. Jalankan perintah berikut untuk memverifikasi tag jendela pemeliharaan.

   Linux & macOS

   ```
   aws ssm list-tags-for-resource \
       --resource-type "MaintenanceWindow" \
       --resource-id "window-id"
   ```

   Windows

   ```
   aws ssm list-tags-for-resource ^
       --resource-type "MaintenanceWindow" ^
       --resource-id "window-id"
   ```

Menandai jendela pemeliharaan (AWS Tools for PowerShell)

1. Jalankan perintah berikut untuk menampilkan jendela pemeliharaan yang dapat Anda tandai.

   ```
   Get-SSMMaintenanceWindow
   ```

2. Jalankan perintah berikut untuk menandai jendela pemeliharaan.

```
$tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
```

```
$tag.Key = "tag-key"
```

```
$tag.Value = "tag-value"
```

```
Add-SSMResourceTag `
    -ResourceType "MaintenanceWindow" `
    -ResourceId "window-id" `
    -Tag $tag
```

windows-id adalah ID dari jendela pemeliharaan yang ingin Anda tandai.

tag-key adalah nama kunci khusus yang Anda berikan. Misalnya, Lingkungan atau Proyek.

tag-value adalah konten khusus untuk nilai yang ingin Anda berikan untuk kunci itu. Misalnya, Produksi atau Q321.

3. Jalankan perintah berikut untuk memverifikasi tag jendela pemeliharaan.

```
Get-SSMResourceTag `
    -ResourceType "MaintenanceWindow" `
    -ResourceId "window-id"
```

Menghapus tag dari jendela pemeliharaan

Anda dapat menggunakan konsol Systems Manager atau baris perintah untuk menghapus tag dari jendela pemeliharaan.

Topik

- Menghapus tag dari jendela pemeliharaan (konsol)
- Menghapus tag dari jendela pemeliharaan (baris perintah)

Menghapus tag dari jendela pemeliharaan (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Maintenance Windows.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih pilih pilih pilih pilih pilih pilih Maintenance Windows.
3. Pilih nama jendela pemeliharaan untuk menghapus tag, lalu pilih tab Tag.
4. Pilih Edit tag, lalu pilih Hapus tag di samping pasangan tag yang tidak lagi Anda butuhkan.
5. Pilih Simpan perubahan.

Menghapus tag dari jendela pemeliharaan (baris perintah)

1. Menggunakan alat baris perintah yang Anda inginkan, jalankan perintah berikut untuk mencantumkan jendela pemeliharaan di akun Anda.

   Linux & macOS

   ```
   aws ssm describe-maintenance-windows
   ```

   Windows

   ```
   aws ssm describe-maintenance-windows
   ```

   PowerShell

   ```
   Get-SSMMaintenanceWindows
   ```

   Perhatikan ID jendela pemeliharaan yang ingin Anda hapus tandanya.
2. Jalankan perintah berikut untuk menghapus tag dari jendela pemeliharaan. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

Linux & macOS

```
aws ssm remove-tags-from-resource \
    --resource-type "MaintenanceWindow" \
    --resource-id "window-id" \
    --tag-key "tag-key"
```

Windows

```
aws ssm remove-tags-from-resource ^
    --resource-type "MaintenanceWindow" ^
    --resource-id "window-id" ^
    --tag-key "tag-key"
```

PowerShell

```
Remove-SSMResourceTag `
    -ResourceType "MaintenanceWindow" `
    -ResourceId "window-id" `
    -TagKey "tag-key
```

Jika berhasil, perintah tidak memiliki output.

3. Jalankan perintah berikut untuk memverifikasi tag jendela pemeliharaan.

Linux & macOS

```
aws ssm list-tags-for-resource \
    --resource-type "MaintenanceWindow" \
    --resource-id "window-id"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "MaintenanceWindow" ^
    --resource-id "window-id"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "MaintenanceWindow" `
    -ResourceId "window-id"
```

Menandai node terkelola

Topik di bagian ini menjelaskan cara bekerja dengan tag pada node terkelola.

Node terkelola adalah mesin apa pun yang dikonfigurasi untukAWS Systems Manager. Ini termasuk instans Amazon Elastic Compute Cloud (Amazon EC2) dan mesin non-EC2 dalam lingkungan hybrid dan multicloud yang dikonfigurasi untuk Systems Manager.

Petunjuk dalam topik ini berlaku untuk setiap mesin yang dikelola menggunakan Systems Manager.

Topik

- Membuat atau mengaktifkan node terkelola dengan tag
- Menambahkan tag ke node terkelola yang ada
- Menghapus tag dari node terkelola

Membuat atau mengaktifkan node terkelola dengan tag

Anda dapat menambahkan tag ke instans EC2 pada saat Anda membuatnya. Anda dapat menambahkan tag ke server on-premise dan mesin virtual (VM) pada saat Anda mengaktifkannya.

Untuk informasi, lihat topik berikut:

- Untuk instans EC2, lihat Menandai sumber daya Amazon EC2 Anda di Panduan Pengguna Amazon EC2 untuk Instans Linux. (Konten berlaku untuk kedua instans EC2 untuk Linux dan untuk Windows)
- Untuk server lokal dan VM, lihat Membuat aktivasi node terkelola untuk lingkungan hibrid.

Menambahkan tag ke node terkelola yang ada

Anda dapat menambahkan tag ke node terkelola dengan menggunakan konsol Systems Manager atau baris perintah.

Topik

Menambahkan tag ke node terkelola yang ada (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih ID node terkelola untuk menambahkan tag, lalu pilih tab Tag.

   > **Note**
   >
   > Jika node terkelola yang Anda harapkan tidak terdaftar, lihat Memecahkan masalah ketersediaan node terkelola untuk tips pemecahan masalah.
4. Di bagian Tag, pilih Edit, dan kemudian tambahkan satu pasangan nilai kunci atau lebih.
5. Pilih Simpan.

Menambahkan tag ke node terkelola yang ada (baris perintah)

Untuk menambahkan tag ke node terkelola yang ada (baris perintah)

1. Menggunakan alat baris perintah pilihan Anda, jalankan perintah berikut untuk melihat daftar node terkelola yang dapat Anda tag.

Linux & macOS

```
aws ssm describe-instance-information
```

Windows

```
aws ssm describe-instance-information
```

PowerShell

```
Get-SSMInstanceInformation
```

Perhatikan ID node terkelola yang ingin Anda tag.

> **Note**
>
> Mesin non-EC2 yang telah terdaftar untuk digunakan dengan Systems Manager di lingkungan hybrid dan multicloud dimulai dengan`mi-`, seperti. `mi-0471e04240EXAMPLE` Instans EC2 memiliki ID yang dimulai dengan `i-`, seperti `i-02573cafcfEXAMPLE`.

2. Jalankan perintah berikut untuk menandai node terkelola. Ganti setiap *`placeholder sumber daya contoh`* dengan informasi Anda sendiri.

Linux & macOS

```
aws ssm add-tags-to-resource \
    --resource-type "ManagedInstance" \
    --resource-id "instance-id" \
    --tags Key=tag-key,Value=tag-value
```

Windows

```
aws ssm add-tags-to-resource ^
    --resource-type "ManagedInstance" ^
    --resource-id "instance-id" ^
    --tags "Key=tag-key,Value=tag-value"
```

PowerShell

```
$tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
```

```
$tag.Key = "tag-key"
```

```
$tag.Value = "tag-value"
```

```
Add-SSMResourceTag `
    -ResourceType "ManagedInstance" `
    -ResourceId "instance-id" `
    -Tag $tag `
    -Force
```

Jika berhasil, perintah tidak memiliki output.

3. Jalankan perintah berikut untuk memverifikasi tag node terkelola.

Linux & macOS

```
aws ssm list-tags-for-resource \
    --resource-type "ManagedInstance" \
    --resource-id "instance-id"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "ManagedInstance" ^
    --resource-id "instance-id"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "ManagedInstance" `
    -ResourceId "instance-id"
```

Menghapus tag dari node terkelola

Anda dapat menggunakan konsol Systems Manager atau baris perintah untuk menghapus tag dari node terkelola.

Topik

Menghapus tag dari node terkelola (konsol)

1. Buka konsol AWS Systems Manager di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Fleet Manager.

   -atau-

   Jika AWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Fleet Managerdi panel navigasi.
3. Pilih nama node terkelola untuk menghapus tag, lalu pilih tab Tag.
4. Di bagian Tag, pilih Edit, lalu pilih Hapus di samping pasangan tag yang tidak lagi Anda butuhkan.
5. Pilih Simpan.

Menghapus tag dari node yang dikelola (baris perintah)

1. Menggunakan alat baris perintah pilihan Anda, jalankan perintah berikut untuk membuat daftar node terkelola di akun Anda.

   Linux & macOS

   ```
   aws ssm describe-instance-information
   ```

   Windows

   ```
   aws ssm describe-instance-information
   ```

PowerShell

```
Get-SSMInstanceInformation
```

Perhatikan nama node terkelola tempat Anda ingin menghapus tag.

2. Jalankan perintah berikut untuk menghapus tag dari node terkelola. Ganti setiap *placeholder sumber daya contoh* dengan informasi Anda sendiri.

   Linux & macOS

   ```
   aws ssm remove-tags-from-resource \
       --resource-type "ManagedInstance" \
       --resource-id "instance-id" \
       --tag-key "tag-key"
   ```

   Windows

   ```
   aws ssm remove-tags-from-resource ^
       --resource-type "ManagedInstance" ^
       --resource-id "instance-id" ^
       --tag-key "tag-key"
   ```

   PowerShell

   ```
   Remove-SSMResourceTag `
       -ResourceId "instance-id" `
       -ResourceType "ManagedInstance" `
       -TagKey "tag-key" `
       -Force
   ```

   Jika berhasil, perintah tidak memiliki output.

3. Jalankan perintah berikut untuk memverifikasi tag node terkelola.

   Linux & macOS

   ```
   aws ssm list-tags-for-resource \
       --resource-type "ManagedInstance" \
   ```

```
    --resource-id "instance-id"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "ManagedInstance" ^
    --resource-id "instance-id"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "ManagedInstance" `
    -ResourceId "instance-id"
```

PenandaanOpsItems

Topik dalam bagian ini menjelaskan cara bekerja dengan tag padaOpsItems.

Topik

- MembuatOpsItems dengan tag
- Menambahkan tag ke yang sudah adaOpsItems
- Menghapus tag dari Systems ManagerOpsItems

MembuatOpsItems dengan tag

Anda dapat menambahkan tag ke khususAWS Systems ManagerOpsItems pada saat Anda membuatnya jika Anda menggunakan alat baris perintah.

Untuk informasi, lihat topik berikut:

Menambahkan tag ke yang sudah adaOpsItems

Anda dapat menambahkan tag keOpsItems dengan menggunakan alat baris perintah.

Topik

- Menambahkan tag ke yang sudah adaOpsItem (baris perintah)

Menambahkan tag ke yang sudah adaOpsItem (baris perintah)

Untuk menambahkan tag ke yang sudah adaOpsItem (baris perintah)

1. Menggunakan alat baris perintah pilihan Anda, jalankan perintah berikut untuk melihat daftarOpsItem yang dapat Anda tandai.

   Linux & macOS

   ```
   aws ssm describe-ops-items
   ```

   Windows

   ```
   aws ssm describe-ops-items
   ```

   PowerShell

   ```
   Get-SSMOpsItemSummary
   ```

   Perhatikan IDOpsItem yang ingin Anda tandai.

2. Jalankan perintah berikut untuk menandaiOpsItem. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri.

   Linux & macOS

   ```
   aws ssm add-tags-to-resource \
       --resource-type "OpsItem" \
       --resource-id "ops-item-id" \
       --tags "Key=tag-key,Value=tag-value"
   ```

   Windows

   ```
   aws ssm add-tags-to-resource ^
       --resource-type "OpsItem" ^
       --resource-id "ops-item-id" ^
       --tags "Key=tag-key,Value=tag-value"
   ```

PowerShell

```
$tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
```

```
$tag.Key = "tag-key"
```

```
$tag.Value = "tag-value"
```

```
Add-SSMResourceTag `
    -ResourceType "OpsItem" `
    -ResourceId "ops-item-id" `
    -Tag $tag `
    -Force
```

Jika berhasil, perintah tidak memiliki output.

3. Jalankan perintah berikut untuk memverifikasiOpsItem tag.

Linux & macOS

```
aws ssm list-tags-for-resource \
    --resource-type "OpsItem" \
    --resource-id "ops-item-id"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "OpsItem" ^
    --resource-id "ops-item-id"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "OpsItem" `
    -ResourceId "ops-item-id"
```

Menghapus tag dari Systems ManagerOpsItems

Anda dapat menggunakan alat baris perintah untuk menghapus tag dari Systems ManagerOpsItems.

Topik

Menghapus tag dariOpsItems (baris perintah)

1. Menggunakan alat baris perintah pilihan Anda, jalankan perintah berikut untuk membuat daftarOpsItems di akun Anda.

   Linux & macOS

   ```
   aws ssm describe-ops-items
   ```

   Windows

   ```
   aws ssm describe-ops-items
   ```

   PowerShell

   ```
   Get-SSMOpsItemSummary
   ```

   Catat namaOpsItem yang ingin Anda hapus tagnya.

2. Jalankan perintah berikut untuk menghapus tag dariOpsItem Op.Replace setiap *`contoh sumber daya`* dengan informasi Anda sendiri.

   Linux & macOS

   ```
   aws ssm remove-tags-from-resource \
       --resource-type "OpsItem" \
       --resource-id "ops-item-id" \
       --tag-key "tag-key"
   ```

Windows

```
aws ssm remove-tags-from-resource ^
    --resource-type "OpsItem" ^
    --resource-id "ops-item-id" ^
    --tag-key "tag-key"
```

PowerShell

```
Remove-SSMResourceTag `
    -ResourceId "ops-item-id" `
    -ResourceType "OpsItem" `
    -TagKey "tag-key" `
    -Force
```

Jika berhasil, perintah tidak memiliki output.

3. Jalankan perintah berikut untuk memverifikasiOpsItem tag.

Linux & macOS

```
aws ssm list-tags-for-resource \
    --resource-type "OpsItem" \
    --resource-id "ops-item-id"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "OpsItem" ^
    --resource-id "ops-item-id"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "OpsItem" `
    -ResourceId "ops-item-id"
```

Menandai parameter Systems Manager

Topik dalam bagian ini menjelaskan cara bekerja dengan tag pada parameter AWS Systems Manager (parameter SSM).

Topik

Menciptakan parameter dengan tag

Anda dapat menambahkan tag ke parameter SSM pada saat Anda membuatnya.

Untuk informasi, lihat topik berikut:

- Membuat parameter Systems Manager (konsol)
- Membuat parameter Systems Manager (AWS CLI)
- Membuat parameter Systems Manager (Tools for WindowsPowerShell)

Menambahkan tag ke parameter yang sudah ada

Anda dapat menambahkan tag untuk parameter SSM khusus yang Anda miliki dengan menggunakan konsol Systems Manager atau baris perintah.

Topik

Menambahkan tag ke parameter yang sudah ada (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Parameter Store.

-atau-

JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Parameter Store.

3. Pilih nama parameter yang telah Anda buat, lalu pilih tab Tag.
4. Pada kotak pertama, masukkan kunci untuk tag, seperti **`Environment`**.
5. Di kotak kedua, masukkan nilai untuk tag, seperti **`Test`**.
6. Pilih Simpan.

Menambahkan tag ke parameter yang sudah ada (AWS CLI)

1. Menggunakan alat baris perintah pilihan Anda, jalankan perintah berikut untuk melihat daftar parameter yang dapat Anda tandai.

```
aws ssm describe-parameters
```

Perhatikan nama parameter yang ingin Anda tandai.

2. Jalankan perintah berikut untuk menandai parameter. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri.

```
aws ssm add-tags-to-resource \
    --resource-type "Parameter" \
    --resource-id "parameter-name" \
    --tags "Key=tag-key,Value=tag-value"
```

Jika berhasil, perintah tidak memiliki output.

3. Jalankan perintah berikut untuk memverifikasi tag parameter.

```
aws ssm list-tags-for-resource --resource-type "Parameter" --resource-id
 "parameter-name"
```

Menambahkan tag ke parameter yang sudah ada (AWS Tools for PowerShell)

1. Jalankan perintah berikut untuk membuat daftar parameter yang dapat Anda tandai.

```
Get-SSMParameterList
```

2. Jalankan perintah berikut untuk menandai parameter. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

```
$tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
```

```
$tag.Key = "tag-key"
```

```
$tag.Value = "tag-value"
```

```
Add-SSMResourceTag `
    -ResourceType "Parameter" `
    -ResourceId "parameter-name" `
    -Tag $tag `
    -Force
```

3. Jalankan perintah berikut untuk memverifikasi tag parameter.

```
Get-SSMResourceTag `
    -ResourceType "Parameter" `
    -ResourceId "parameter-name"
```

Menghapus tag dari parameter SSM

Anda dapat menggunakan konsol Systems Manager atau baris perintah untuk menghapus tag dari parameter SSM.

Topik

- Menghapus tag dari parameter SSM (konsol)
- Menghapus tag dari parameter SSM (baris perintah)

Menghapus tag dari parameter SSM (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Parameter Store.

-atau-

JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Parameter Store.

3. Pilih nama parameter untuk menghapus tag, lalu pilih tab Tag.
4. Pilih Hapus di samping pasangan tag yang tidak lagi Anda butuhkan.
5. Pilih Simpan.

Menghapus tag dari parameter SSM (baris perintah)

1. Menggunakan alat baris perintah yang Anda inginkan, jalankan perintah berikut untuk membuat daftar parameter di akun Anda.

   Linux & macOS

   ```
   aws ssm describe-parameters
   ```

   Windows

   ```
   aws ssm describe-parameters
   ```

   PowerShell

   ```
   Get-SSMParameterList
   ```

   Perhatikan nama parameter yang ingin Anda hapus tagnya.
2. Jalankan perintah berikut untuk menghapus tag dari parameter. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

   Linux & macOS

   ```
   aws ssm remove-tags-from-resource \
       --resource-type "Parameter" \
       --resource-id "parameter-name" \
       --tag-key "tag-key"
   ```

Windows

```
aws ssm remove-tags-from-resource ^
    --resource-type "Parameter" ^
    --resource-id "parameter-name" ^
    --tag-key "tag-key"
```

PowerShell

```
Remove-SSMResourceTag
    -ResourceId "parameter-name"
    -ResourceType "Parameter"
    -TagKey "tag-key
```

Jika berhasil, perintah tidak memiliki output.

3. Jalankan perintah berikut untuk memverifikasi tag dokumen.

Linux & macOS

```
aws ssm list-tags-for-resource \
    --resource-type "Parameter" \
    --resource-id "parameter-name"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "Parameter" ^
    --resource-id "parameter-name"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "Parameter" `
    -ResourceId "parameter-name"
```

Menandai dasar patch

Topik dalam bagian ini menjelaskan cara bekerja dengan tag pada dasar patch.

Topik

Membuat dasar patch dengan tag

Anda dapat menambahkan tag ke dasar patch AWS Systems Manager saat Anda membuatnya.

Untuk informasi, lihat topik berikut:

- Bekerja dengan dasar patch kustom
- Membuat dasar patch
- Buat dasar patch dengan repositori kustom untuk versi OS yang berbeda

Menambahkan tag ke dasar patch yang sudah ada

Anda dapat menambahkan tag untuk dasar patch yang Anda miliki sendiri dengan menggunakan konsol Systems Manager atau baris perintah.

Topik

Menambahkan tag ke sebuah dasar patch yang sudah ada (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Patch Manager.

   -atau-

JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.

3. Pilih nama dasar patch khusus yang telah Anda buat, gulir ke bawah ke bagian Tabel tag, lalu pilih Edit tag.
4. Pilih Tambahkan tanda.
5. Untuk Kunci, masukkan kunci untuk tag, seperti **`Environment`**.
6. Untuk Nilai, masukkan nilai untuk tag, seperti **`Test`**.
7. Pilih Simpan perubahan.

Menambahkan tag ke dasar patch yang sudah ada (AWS CLI)

1. Menggunakan alat baris perintah pilihan Anda, jalankan perintah berikut untuk melihat daftar dasar patch yang dapat Anda tandai.

```
aws ssm describe-patch-baselines --filters "Key=OWNER,Values=[Self]"
```

   Perhatikan ID dasar patch yang ingin Anda tandai.

2. Jalankan perintah berikut untuk menandai dasar patch. Ganti setiap *`contoh placeholder sumber daya`* dengan informasi Anda sendiri.

   Linux & macOS

```
aws ssm add-tags-to-resource \
    --resource-type "PatchBaseline" \
    --resource-id "baseline-id" \
    --tags "Key=tag-key,Value=tag-value"
```

   Windows

```
aws ssm add-tags-to-resource ^
    --resource-type "PatchBaseline" ^
    --resource-id "baseline-id" ^
    --tags "Key=tag-key,Value=tag-value"
```

Jika berhasil, perintah tidak memiliki output.

3. Jalankan perintah berikut untuk memverifikasi tag dasar patch.

   Linux & macOS

   ```
   aws ssm list-tags-for-resource \
       --resource-type "PatchBaseline" \
       --resource-id "baseline-id"
   ```

   Windows

   ```
   aws ssm list-tags-for-resource ^
       --resource-type "PatchBaseline" ^
       --resource-id "patchbaseline-id"
   ```

Menandai dasar patch (AWS Tools for PowerShell)

1. Jalankan perintah berikut untuk membuat daftar dasar patch yang dapat Anda tandai.

   ```
   Get-SSMPatchBaseline
   ```

2. Jalankan perintah berikut untuk menandai dasar patch. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

   ```
   $tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
   ```

   ```
   $tag.Key = "tag-key"
   ```

   ```
   $tag.Value = "tag-value"
   ```

   ```
   Add-SSMResourceTag `
       -ResourceType "PatchBaseline" `
       -ResourceId "baseline-id" `
       -Tag $tag `
       -Force
   ```

3. Jalankan perintah berikut untuk memverifikasi tag dasar patch.

```
Get-SSMResourceTag `
    -ResourceType "PatchBaseline" `
    -ResourceId "baseline-id"
```

Menghapus tag dari dasar patch

Anda dapat menggunakan konsol Systems Manager atau baris perintah untuk menghapus tag dari dasar patch.

Topik

- Menghapus tag dari dasar patch (konsol)
- Menghapus tag dari dasar patch (baris perintah)

Menghapus tag dari dasar patch (konsol)

1. Buka konsol AWS Systems Manager pada https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Patch Manager.

   -atau-

   JikaAWS Systems Manager halaman beranda terbuka terlebih dahulu, pilih ikon menu (☰) untuk membuka panel navigasi, lalu pilih Patch Manager.
3. Pilih nama dari dasar patch untuk menghapus tag, gulir ke bawah ke bagian Tabel tag, lalu pilih tab Edit tag.
4. Pilih Hapus tag di samping pasangan tag yang tidak lagi Anda butuhkan.
5. Pilih Simpan perubahan.

Menghapus tag dari dasar patch (baris perintah)

1. Menggunakan alat baris perintah pilihan Anda, jalankan perintah berikut untuk membuat daftar dasar patch di akun Anda.

Linux & macOS

```
aws ssm describe-patch-baselines
```

Windows

```
aws ssm describe-patch-baselines
```

PowerShell

```
Get-SSMPatchBaseline
```

Perhatikan ID dari dasar patch yang ingin Anda hapus tagnya.

2. Jalankan perintah berikut untuk menghapus tag dari dasar patch. Ganti setiap *contoh placeholder sumber daya* dengan informasi Anda sendiri.

Linux & macOS

```
aws ssm remove-tags-from-resource \
    --resource-type "PatchBaseline" \
    --resource-id "baseline-id" \
    --tag-key "tag-key"
```

Windows

```
aws ssm remove-tags-from-resource ^
    --resource-type "PatchBaseline" ^
    --resource-id "baseline-id" ^
    --tag-key "tag-key"
```

PowerShell

```
Remove-SSMResourceTag `
    -ResourceType "PatchBaseline" `
    -ResourceId "baseline-id" `
    -TagKey "tag-key
```

Jika berhasil, perintah tidak memiliki output.

3. Jalankan perintah berikut untuk memverifikasi tag dasar patch.

Linux & macOS

```
aws ssm list-tags-for-resource \
    --resource-type "PatchBaseline" \
    --resource-id "baseline-id"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "PatchBaseline" ^
    --resource-id "baseline-id"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "PatchBaseline" `
    -ResourceId "baseline-id"
```

AWS Systems Manager referensi

Informasi dan topik berikut dapat membantu Anda menerapkan solusi AWS Systems Manager dengan lebih baik.

Kepala Sekolah

Dalam AWS Identity and Access Management (IAM), Anda dapat memberikan atau menolak akses layanan ke sumber daya menggunakan elemen kebijakan Principal. Nilai elemen kebijakan Prinsipal untuk Systems Manager adalah `ssm.amazonaws.com`.

Didukung Wilayah AWS dan titik akhir

Lihat titik akhir layanan Systems Manager di. Referensi Umum Amazon Web Services

Service Quotas

Lihat kuota layanan Systems Manager di. Referensi Umum Amazon Web Services

Referensi API

Lihat Referensi API AWS Systems Manager.

AWS CLI Referensi Perintah

Lihat AWS Systems Manager bagian Referensi AWS CLI Perintah.

AWS Tools for PowerShell Cmdlet Referensi

Lihat AWS Systems Manager bagian Referensi AWS Tools for PowerShell Cmdlet.

SSM AgentRepositori di GitHub

Lihat aws/ amazon-ssm-agent.

Ajukan Pertanyaan

Masalah Systems Manager di AWS Re:post

AWS Blog Berita

Alat Manajemen

Topik referensi lainnya

- Referensi: Pola dan jenis EventBridge acara Amazon untuk Systems Manager
- Referensi: Ekspresi cron dan rate untuk Systems Manager
- Referensi: ec2messages, ssmmessages, dan operasi API lainnya
- Referensi: Membuat string tanggal dan waktu yang diformat untuk Systems Manager

Referensi: Pola dan jenis EventBridge acara Amazon untuk Systems Manager

> **Note**
>
> Amazon EventBridge adalah cara yang lebih disukai untuk mengelola acara Anda. CloudWatch Acara dan EventBridge merupakan layanan dan API dasar yang sama, tetapi EventBridge menyediakan lebih banyak fitur. Perubahan yang Anda buat di salah satu CloudWatch atau EventBridge tercermin di setiap konsol. Untuk informasi selengkapnya, lihat Panduan EventBridge Pengguna Amazon.

Menggunakan Amazon EventBridge, Anda dapat membuat aturan yang cocok dengan peristiwa yang masuk dan merutekkannya ke target untuk diproses.

Sebuah peristiwa menunjukkan perubahan dalam lingkungan dalam aplikasi Anda sendiri, perangkat lunak sebagai layanan (SaaS) aplikasi, atau aplikasi. Layanan AWS Kejadian dihasilkan atas dasar upaya terbaik. Setelah jenis peristiwa yang ditentukan dalam aturan terdeteksi, EventBridge rutekan ke target tertentu untuk diproses. Target dapat mencakup instans Amazon Elastic Compute Cloud (Amazon EC2), fungsi AWS Lambda, aliran Amazon Kinesis, tugas Amazon Elastic Container Service (Amazon ECS), mesin berstatus AWS Step Functions, topik Amazon Simple Notification Service (Amazon SNS), antrean Amazon Simple Queue Service (Amazon SQS), target tertanam dan masih banyak lagi.

Untuk informasi tentang membuat EventBridge aturan, lihat topik berikut:

- Pemantauan peristiwa Systems Manager dengan Amazon EventBridge
- Contoh EventBridge acara Amazon untuk Systems Manager
- Memulai Amazon EventBridge di Panduan EventBridge Pengguna Amazon

Sisa topik ini menjelaskan jenis peristiwa Systems Manager yang dapat Anda sertakan dalam EventBridge aturan Anda.

Jenis kejadian: Otomatisasi

Nama jenis kejadian	Deskripsi kejadian yang dapat Anda tambahkan ke aturan
Notifikasi perubahan Status Eksekusi Otomatisasi EC2	Status keseluruhan sebuah alur kerja Otomatisasi berubah. Anda dapat menambahk an satu atau lebih perubahan status berikut ke aturan kejadian: • Disetujui • Dibatalkan • Gagal • PendingApproval • PendingChangeCalendarOverride • Ditolak • Terjadwal • Berhasil • TimedOut
Notifikasi perubahan Status Langkah Otomatisa si EC2	Status sebuah langkah tertentu dalam alur kerja Otomatisasi berubah. Anda dapat menambahkan satu atau lebih perubahan status berikut ke aturan kejadian: • Dibatalkan • Gagal • Berhasil • TimedOut

Jenis acara: Change Calendar

Nama jenis kejadian	Deskripsi kejadian yang dapat Anda tambahkan ke aturan
Perubahan Status Kalender	Keadaan suatu Change Calendar perubahan . Anda dapat menambahkan satu atau kedua perubahan status berikut ke aturan kejadian: • TERBUKA • TERTUTUP Perubahan status untuk kalender yang dibagikan dari Akun AWS tidak didukung.

Jenis acara: Change Manager

Nama jenis kejadian	Deskripsi kejadian yang dapat Anda tambahkan ke aturan
Ubah Pembaruan Status Permintaan	Keadaan permintaan Change Manager perubahan. Anda dapat menggunakan status berikut dalam aturan acara: • Disetujui • Ditolak • InProgress

Jenis kejadian: Kepatuhan Konfigurasi

Nama jenis kejadian	Deskripsi kejadian yang dapat Anda tambahkan ke aturan
Perubahan Status Kepatuhan Konfigurasi	Status node terkelola berubah, baik untuk kepatuhan asosiasi atau kepatuhan patch. Anda dapat menambahkan satu atau lebih perubahan status berikut ke aturan kejadian: • patuh • tidak_patuh

Jenis kejadian: Inventaris

Nama jenis kejadian	Deskripsi kejadian yang dapat Anda tambahkan ke aturan
Perubahan Status Sumber Daya Inventaris	Penghapusan inventaris kustom dan PutInventorypanggilan yang menggunakan versi skema lama. Anda dapat menambahkan satu atau lebih perubahan status berikut ke aturan kejadian: • Jenis inventaris kustom dihapus peristiwa pada node tertentu. EventBridge mengirimkan satu acara per node per kustom Inventory Type. • Jenis inventaris kustom dihapus peristiwa untuk semua node. • PutInventory panggilan dengan acara versi skema lama. EventBridgemengirimkan acara ini ketika versi skema kurang dari skema saat ini. Kejadian ini berlaku untuk semua jenis inventaris.

Nama jenis kejadian	Deskripsi kejadian yang dapat Anda tambahkan ke aturan
	Untuk informasi selengkapnya, lihat Tentang EventBridgepemantauan peristiwa Inventaris.

Jenis kejadian: Maintenance Window

Nama jenis kejadian	Deskripsi kejadian yang dapat Anda tambahkan ke aturan
Notifikasi Perubahan Status Maintenance Window	Status keseluruhan satu atau lebih jendela pemeliharaan berubah. Anda dapat menambahkan satu atau lebih perubahan status berikut ke aturan kejadian: • DINONAKTIFKAN • DIAKTIFKAN
Notifikasi Pendaftaran Target Maintenance Window	Status dari satu atau lebih target jendela pemeliharaan berubah. Anda dapat menambahkan satu atau lebih perubahan status berikut ke aturan kejadian: • DIBATALKAN PENDAFTARANNYA • TERDAFTAR • DIPERBARUI
Notifikasi perubahan Status Eksekusi Maintenance Window	Status keseluruhan jendela pemeliharaan berubah saat sedang berjalan. Anda dapat menambahkan satu atau lebih perubahan status berikut ke aturan kejadian: • DIBATALKAN • MEMBATALKAN • GAGAL

Nama jenis kejadian	Deskripsi kejadian yang dapat Anda tambahkan ke aturan
	• SEDANG_BERLANGSUNG • TERTUNDA • DILOMPATI_TUMPANGTINDIH • BERHASIL • HABIS_WAKTU
Notifikasi perubahan Status Eksekusi Tugas Maintenance Window	Status sebuah tugas dalam jendela pemeliharaan berubah saat sedang berjalan. Anda dapat menambahkan satu atau lebih perubahan status berikut ke aturan kejadian: • DIBATALKAN • MEMBATALKAN • GAGAL • SEDANG_BERLANGSUNG • BERHASIL • HABIS_WAKTU

Nama jenis kejadian	Deskripsi kejadian yang dapat Anda tambahkan ke aturan
Notifikasi perubahan Status Pemanggilan Target Tugas Maintenance Window	Status sebuah tugas jendela pemeliharaan pada target tertentu berubah. Pemberitahuan ini didukung penuh hanya untuk Run Command tugas. Untuk jenis tugas ini, Anda dapat menambahkan satu atau beberapa perubahan status berikut ke aturan acara: • DIBATALKAN • MEMBATALKAN • GAGAL • SEDANG_BERLANGSUNG • BERHASIL • HABIS_WAKTU Untuk OtomasiAWS Lambda,, dan AWS Step Functions tugas, hanya EventBridge melaporkan negara bagian `IN_PROGRESS` dan`COMPLETE`. `COMPLETED`Dilaporkan apakah tugas itu berhasil atau tidak.
Notifikasi Pendaftaran Tugas Maintenance Window	Status dari satu atau lebih tugas jendela pemeliharaan berubah. Anda dapat menambahkan satu atau lebih perubahan status berikut ke aturan kejadian: • DIBATALKAN PENDAFTARANNYA • TERDAFTAR • DIPERBARUI

Jenis acara: OpsCenter

Nama jenis kejadian	Deskripsi kejadian yang dapat Anda tambahkan ke aturan
OpsItemBuat	Terjadi ketika OpsItem sebuah dibuat. Anda dapat menambahkan aturan untuk salah satu OpsItem jenis berikut: • /aws/masalah • /aws/tugas • /aws/wawasan • /aws/item tindakan
OpsItemPerbarui	Terjadi ketika OpsItem sebuah diperbarui. Anda dapat menambahkan aturan untuk salah satu OpsItem jenis berikut: • /aws/masalah • /aws/tugas • /aws/wawasan • /aws/item tindakan

Jenis acara: Parameter Store

Nama jenis kejadian	Deskripsi kejadian yang dapat Anda tambahkan ke aturan
Perubahan Parameter Store	Status sebuah parameter berubah. Anda dapat menambahkan satu atau lebih perubahan status berikut ke aturan kejadian: • Buat • Perbarui • Hapus

Nama jenis kejadian	Deskripsi kejadian yang dapat Anda tambahkan ke aturan
	• LabelParameterVersion Untuk informasi selengkapnya, lihat Mengkonfigurasi EventBridge aturan untuk parameter dan kebijakan parameter.
Tindakan Kebijakan Parameter Store	Syarat perubahan kebijakan parameter lanjutan terpenuhi. Anda dapat menambahkan satu atau lebih perubahan status berikut ke aturan kejadian: • Kedaluwarsa • ExpirationNotification • NoChangeNotification Untuk informasi selengkapnya, lihat Mengkonfigurasi EventBridge aturan untuk parameter dan kebijakan parameter.

Jenis acara: Run Command

Nama jenis kejadian	Deskripsi kejadian yang dapat Anda tambahkan ke aturan
Notifikasi perubahan Status Pemanggilan Perintah EC2	Status sebuah perintah yang dikirim ke instans terkelola individu berubah. Anda dapat menambahkan satu atau lebih perubahan status berikut ke aturan kejadian: • Berhasil • InProgress • TimedOut

Nama jenis kejadian	Deskripsi kejadian yang dapat Anda tambahkan ke aturan
	• Dibatalkan • Gagal
Notifikasi perubahan Status Perintah EC2	Status keseluruhan perintah berubah. Anda dapat menambahkan satu atau lebih perubahan status berikut ke aturan kejadian: • Berhasil • InProgress • TimedOut • Dibatalkan • Gagal

Jenis acara: State Manager

Nama jenis kejadian	Deskripsi kejadian yang dapat Anda tambahkan ke aturan
Perubahan Negara State Manager Asosiasi EC2	Status keseluruhan suatu Asosiasi berubah saat sedang diterapkan. Anda dapat menambahkan satu atau lebih perubahan status berikut ke aturan kejadian: • Gagal • Tertunda • Berhasil
Perubahan Negara Asosiasi State Manager Instans EC2	Status instans terkelola tunggal yang ditargetkan oleh Asosiasi berubah. Anda dapat menambahkan satu atau lebih perubahan status berikut ke aturan kejadian: • Gagal

Nama jenis kejadian	Deskripsi kejadian yang dapat Anda tambahkan ke aturan
	• Tertunda • Berhasil

Referensi: Ekspresi cron dan rate untuk Systems Manager

Saat Anda membuat State Manager asosiasi atau jendela pemeliharaan diAWS Systems Manager, Anda menentukan jadwal kapan jendela atau asosiasi harus berjalan. Anda dapat menentukan penjadwal baik sebagai entri berbasis waktu, disebut Cron expression, atau entri berbasis frekuensi, disebut rate expression.

Informasi umum tentang cron dan ekspresi rate

Informasi berikut berlaku untuk ekspresi cron dan rate untuk kedua jendela pemeliharaan dan asosiasi.

Jadwal lari tunggal

Ketika Anda membuat associate atau pemeliharaan windows, Anda dapat menentukan timestamp dalam format Coordinated Universal Time (UTC) sehingga berjalan sekali pada waktu yang ditentukan. Sebagai contoh: `"at(2020-07-07T15:55:00)"`

Jadwalkan offset

Asosiasi dan jendela pemeliharaan mendukung offset jadwal untuk ekspresi cron saja. Offset jadwal adalah jumlah hari untuk menunggu setelah tanggal dan waktu yang ditentukan oleh ekspresi cron sebelum menjalankan jendela asosiasi atau pemeliharaan.

Maintenance window example

Dalam perintah berikut, ekspresi cron menjadwalkan jendela pemeliharaan untuk menjalankan Selasa ketiga setiap bulan pada pukul 11:30. Namun, karena jadwal offset adalah2, jendela pemeliharaan tidak akan berjalan sampai 11:30 PM dua hari kemudian.

```
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

Dalam perintah berikut, ekspresi cron menjadwalkan asosiasi untuk menjalankan Kamis kedua setiap bulan. Namun, karena jadwal offset adalah3, asosiasi tidak akan berjalan sampai hari Minggu berikutnya, tiga hari kemudian.

```
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**
>
> Untuk menggunakan offset dengan asosiasi, Anda harus menentukan opsi. `--apply-only-at-cron-interval` Opsi ini memberitahu sistem untuk tidak menjalankan asosiasi segera setelah Anda membuatnya.

Jika Anda membuat asosiasi atau jendela pemeliharaan dengan ekspresi cron yang menargetkan hari yang telah berlalu dalam periode saat ini, tetapi menambahkan tanggal offset jadwal yang jatuh di masa depan, jendela asosiasi atau pemeliharaan tidak akan berjalan dalam periode tersebut. Itu akan berlaku pada periode berikut. Misalnya, jika Anda menentukan ekspresi cron yang akan menjalankan pemeliharaan windows kemarin dan menambahkan penjadwal offset dua hari, pemeliharaan windows tidak akan berjalan besok.

Bidang wajib

Ekspresi cron untuk jendela pemeliharaan memiliki enam bidang wajib. Ekspresi cron untuk asosiasi memiliki lima. (saat ini State Manager tidak mendukung menentukan bulan dalam ekspresi cron untuk asosiasi.) Bidang tambahan, `Seconds` bidang (yang pertama dalam ekspresi cron), adalah opsional. Field dipisahkan oleh space.

Contoh ekspresi cron

Menit	Jam	Hari dalam sebulan	Bulan	Hari dalam seminggu	Tahun	Arti
0	10	*	*	?	*	Jalankan pada pukul 10:00 pagi (UTC) setiap hari
15	12	*	*	?	*	Jalankan pada pukul 12:15 malam (UTC) setiap hari
0	18	?	*	MON-FRI	*	Jalankan pada pukul 6:00 sore (UTC) setiap Senin hingga Jumat
0	8	1	*	?	*	Jalankan pada pukul 8:00 AMS (UTC) setiap tanggal 1 pada bulan tersebut

Nilai yang didukung

Tabel berikut menunjukkan nilai support untuk entri cron diperlukan.

Nilai support untuk ekspresi cron

Bidang	Nilai	Wildcard
Menit	`0-59`	`, - * /`
Jam	`0-23`	`, - * /`
D ay-of-month	`1-31`	`, - * ? / L W`
Bulan (hanya jendela pemeliharaan)	`1- 12 atau JAN - DEC`	`, - * /`
D ay-of-week	`1- 7 atau SUN - SAT`	`, - * ? / L #`
Tahun	`1970-2199`	`, - * /`

> **Note**
>
> Anda tidak dapat menentukan nilai di day-of-month dan di day-of-week bidang dalam ekspresi cron yang sama. Jika Anda menentukan nilai di salah satu fields, gunakan ? (tanda tanya) di field lain.

Wildcard untuk ekspresi cron

Tabel berikut menunjukkan nilai wildcard yang support ekspresi cron.

> **Note**
>
> Ekspresi cron yang mengarah ke tingkat lebih cepat dari (5) menit tidak support. Support untuk menentukan nilai day-of-week dan day-of-month nilai tidak lengkap. Gunakan karakter tanda tanya (?) di salah satu field ini.

Wildcard support untuk ekspresi cron

Wildcard	Deskripsi
,	Wildcard , (koma) mencakup nilai tambahan. Di field Bulan, JAN, FEB, MAR akan mencakup Januari, Februari, dan Maret.
-	Wildcard - (dash) menentukan rentang. Di bidang Tanggal, 1-15 akan mencakup tanggal 1 hingga 15 pada bulan yang ditentukan.
*	Wildcard * (asterisk) mencakup semua nilai di lapangan. Di field Jam, * akan mencakup setiap jam.
/	Wildcard / (garis miring maju) menentuka n kenaikan. Di field menit, Anda bisa memasukkan 1/10 untuk menentukan setiap menit kesepuluh, mulai dari menit pertama jam. Jadi 1/10 menentukan pertama, menit 11, 21, dan 31, dan seterusnya.
?	Wildcard ? (tanda tanya) menentukan satu atau yang lain. Di ay-of-month bidang D Anda bisa memasukkan 7 dan jika Anda tidak peduli hari apa dalam minggu ke-7, Anda bisa masuk? di ay-of-week bidang D.
L	LWildcard di ay-of-week bidang D ay-of-mon th atau D menentukan hari terakhir bulan atau minggu.
W	WWildcard di ay-of-month bidang D menentukan hari kerja. Di ay-of-month bidang D, 3W menentukan hari yang paling dekat dengan hari kerja ketiga setiap bulan.

Wildcard	Deskripsi
#	#Wildcard di day-of-week lapangan diikuti oleh angka antara satu dan lima menentukan hari tertentu dalam sebulan. 5 #3 menentukan Kamis ke-3 setiap bulan.

Nilai ekspresi

Ekspresi rate memiliki dua field wajib berikut. Fields dipisahkan oleh spasi.

Kolom yang diperlukan untuk ekspresi rate

Bidang	Nilai
Nilai	angka positif, seperti 1 atau 15
Unit	`minute` `minutes` `hour` `hours` `day` `days`

Jika nilai sama dengan 1, maka unit harus tunggal. Demikian pula, untuk nilai lebih besar dari 1, unit harus jamak. Misalnya, `rate(1 hours)` dan `rate(5 hour)` tidak validasi, tapi `rate(1 hour)` dan `rate(5 hours)` validasi.

Topik

- Ekspresi cron dan rate untuk associate
- Ekspresi cron dan rate untuk pemeliharaan windows

Ekspresi cron dan rate untuk associate

Bagian ini mencakup contoh ekspresi cron dan rate untuk State Manager asosiasi. Sebelum Anda membuat salah satu ungkapan ini, perhatikan informasi berikut:

- Asosiasi mendukung ekspresi cron berikut: Setiap 1/2, 1, 2, 4, 8, atau 12 jam; setiap hari, setiap minggu, atau setiap hari dan waktu tertentu dalam seminggu; hari tertentu dalam minggu tertentu dalam sebulan, atau x hari terakhir bulan pada waktu tertentu.
- Associate support ekspresi rate berikut: interval 30 menit atau lebih dan kurang dari 31 hari.
- Jika Anda menentukan opsional `Seconds` field, nilai dapat menjadi 0 (nol). Sebagai contoh: `cron(0 */30 * * * ? *)`
- Untuk associate yang mengumpulkan metadata untuk Inventory, kemampuan dari AWS Systems Manager, kami rekomendasi menggunakan ekspresi rate.
- State Managersaat ini tidak mendukung menentukan bulan dalam ekspresi cron untuk asosiasi.

Asosiasi mendukung ekspresi cron yang mencakup hari dalam seminggu dan tanda angka (#) untuk menunjuk hari ke-n dalam sebulan untuk menjalankan asosiasi. Berikut adalah contoh yang menjalankan jadwal cron pada hari Selasa ketiga setiap bulan pukul 23:30 UTC:

```
cron(30 23 ? * TUE#3 *)
```

Berikut adalah contoh yang berjalan pada hari Kamis kedua setiap bulan pada tengah malam UTC:

```
cron(0 0 ? * THU#2 *)
```

Asosiasi juga mendukung tanda (L) untuk menunjukkan hari X terakhir setiap bulan. Berikut adalah contoh yang menjalankan jadwal cron pada hari Selasa terakhir setiap bulan pada tengah malam UTC:

```
cron(0 0 ? * 3L *)
```

Untuk mengontrol lebih lanjut saat asosiasi berjalan, misalnya jika Anda ingin menjalankan asosiasi dua hari setelah patch Selasa, Anda dapat menentukan offset. Offset mendefinisikan berapa hari untuk menunggu setelah hari yang dijadwalkan untuk menjalankan asosiasi. Misalnya, jika Anda menentukan jadwal cron`cron(0 0 ? * THU#2 *)`, Anda dapat menentukan angka 3 di bidang Offset Jadwal untuk menjalankan asosiasi setiap hari Minggu setelah Kamis kedua setiap bulan.

Untuk menggunakan offset, Anda harus memilih asosiasi Terapkan hanya pada opsi interval Cron yang ditentukan berikutnya di konsol atau Anda harus menentukan `--apply-only-at-cron-`

`interval` parameter penggunaan dari baris perintah. Opsi ini memberitahu untuk State Manager tidak menjalankan asosiasi segera setelah Anda membuatnya.

Tabel berikut menyajikan contoh cron untuk asosiasi.

Contoh cron untuk associate

Contoh	Detail
cron(0/30 * * * ? *)	Setiap 30 menit
cron(0 0/1 * * ? *)	Setiap jam
cron(0 0/2 * * ? *)	Setiap 2 jam
cron(0 0/4 * * ? *)	Setiap 4 jam
cron(0 0/8 * * ? *)	Setiap 8 jam
cron(0 0/12 * * ? *)	Setiap 12 jam
cron(15 13 ? * * *)	Setiap hari pukul 1:15 siang
cron(15 13 ? * MON *)	Setiap Senin pukul 1:15 siang
cron(30 23 ? * TUE#3 *)	Selasa ketiga setiap bulan pukul 11:30

Berikut adalah beberapa contoh rate untuk associate.

Contoh rate untuk associate

Contoh	Detail
rate(30 minutes)	Setiap 30 menit
rate(1 hour)	Setiap jam
rate(5 hours)	Setiap 5 jam
rate(15 days)	Setiap 15 hari

AWS CLIcontoh untuk asosiasi

Untuk membuat State Manager asosiasi menggunakanAWS CLI, Anda menyertakan `--schedule-expression` parameter dengan ekspresi cron atau rate. Contoh berikut menggunakan AWS CLI pada mesin Linux lokal.

> **Note**
>
> Secara default, ketika Anda membuat associate baru, sistem berjalan segera setelah dibuat lalu mengikuti dengan penjadwal yang Anda tentukan. Tentukan `--apply-only-at-cron-interval` sehingga associate tidak berjalan segera setelah Anda membuatnya. Ini parameter tidak support untuk ekspresi rate.

```
aws ssm create-association \
    --association-name "My-Cron-Association" \
    --schedule-expression "cron(0 2 ? * SUN *)" \
    --targets Key=tag:ServerRole,Values=WebServer \
    --name AWS-UpdateSSMAgent
```

```
aws ssm create-association \
    --association-name "My-Rate-Association" \
    --schedule-expression "rate(7 days)" \
    --targets Key=tag:ServerRole,Values=WebServer \
    --name AWS-UpdateSSMAgent
```

```
aws ssm create-association \
    --association-name "My-Rate-Association" \
    --schedule-expression "at(2020-07-07T15:55:00)" \
    --targets Key=tag:ServerRole,Values=WebServer \
    --name AWS-UpdateSSMAgent \
    --apply-only-at-cron-interval
```

Ekspresi cron dan rate untuk pemeliharaan windows

Bagian ini mencakup contoh ekspresi cron dan rate untuk pemeliharaan windows.

Tidak seperti State Manager asosiasi, jendela pemeliharaan mendukung semua ekspresi cron dan rate. Ini termasuk support untuk nilai di field detik.

Misalnya, 6-field ekspresi cron berikut menjalankan pemeliharaan windows pada pukul 9:30 AMS setiap hari.

```
cron(30 09 ? * * *)
```

Dengan menambahkan nilai ke `Seconds` field, ekspresi 7-field berikut menjalankan pemeliharaan windows pada pukul 9:30:24 AMS setiap hari.

```
cron(24 30 09 ? * * *)
```

Tabel berikut memberikan tambahan 6-field contoh cron untuk pemeliharaan windows.

Contoh cron untuk pemeliharaan windows

Contoh	Detail
cron (0 2 ? * KAM#3 *)	Pukul 02:00 AMS Kamis ketiga setiap bulan
cron (15 10 ? * * *)	Pukul 10:15 AMS setiap hari
cron (15 10 ? * SENIN-JUM *)	Pukul 10:15 AMS setiap hari Senin, Selasa, Rabu, Kamis dan Jumat
cron (0 2 L * ? *)	Pukul 02:00 AMS pada hari terakhir setiap bulan
cron (15 10 ? * 6L *)	Pukul 10:15 AMS pada hari Jumat terakhir setiap bulan

Tabel berikut memberikan contoh rate untuk pemeliharaan windows.

Contoh rate untuk pemeliharaan windows

Contoh	Detail
rate(30 minutes)	Setiap 30 menit
rate(1 hour)	Setiap jam
rate(5 hours)	Setiap 5 jam

Contoh	Detail
rate(25 days)	Setiap 25 hari

AWS CLIcontoh untuk jendela pemeliharaan

Untuk membuat pemeliharaan windows menggunakan AWS CLI, Anda menyertakan parameter `--schedule` dengan ekspresi cron atau rate atau timestamp. Contoh berikut menggunakan AWS CLI pada mesin Linux lokal.

```
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
```

Info lebih lanjut

- Ekspresi CRON di situs web Wikipedia

Referensi: ec2messages, ssmmessages, dan operasi API lainnya

Jika Anda memantau operasi API, Anda mungkin melihat panggilan ke yang berikut:

- `ec2messages:AcknowledgeMessage`
- `ec2messages:DeleteMessage`
- `ec2messages:FailMessage`
- `ec2messages:GetEndpoint`
- `ec2messages:GetMessages`
- `ec2messages:SendReply`
- `ssmmessages:CreateControlChannel`
- `ssmmessages:CreateDataChannel`
- `ssmmessages:OpenControlChannel`
- `ssmmessages:OpenDataChannel`
- `ssm:DescribeInstanceProperties`
- `ssm:DescribeDocumentParameters`
- `ssm:ListInstanceAssociations`
- `ssm:RegisterManagedInstance`
- `ssm:UpdateInstanceAssociationStatus`
- `ssm:UpdateInstanceInformation`
- `ssm:GetManifest`
- `ssm:PutConfigurePackageResult`
- `ssm:GetCalendar`
- `ssm:PutCalendar`

Ini adalah operasi khusus yang digunakan oleh AWS Systems Manager.

Operasi API terkait agen (ssmmessages dan titik akhir ec2messages)

operasi API ssmmessages

Systems Manager menggunakan `ssmmessages` endpoint untuk dua jenis operasi API berikut:

- Operasi dari SSM Agent keSession Manager, kemampuan AWS Systems Manager, di cloud. Titik akhir ini diperlukan untuk membuat dan menghapus saluran sesi dengan Session Manager layanan di cloud. Selain itu, jika konektivitas diizinkan, SSM Agent terima `Command` dokumen melalui iniAmazon Message Gateway Service. Jika konektivitas tidak diizinkan, SSM Agent terima `Command` dokumen melaluiAmazon Message Delivery Service. Untuk informasi lebih lanjut, lihat Tindakan, sumber daya, kunci syarat untuk Amazon Session Manager Message Gateway Service.
- Operasi dari Systems Manager Agent (SSM Agent) ke layanan Systems Manager di cloud.

operasi API ec2messages

`ec2messages:*`Operasi API dibuat ke Amazon Message Delivery Service titik akhir. Systems Manager menggunakan endpoint ini untuk operasi API dari Systems Manager Agent (SSM Agent) ke layanan Systems Manager di cloud.

Prioritas koneksi titik akhir

Dimulai dengan versi 3.3.40.0 dari, Systems SSM Agent Manager mulai menggunakan `ssmmessages:*` endpoint (Amazon Message Gateway Service) kapan pun tersedia, bukan endpoint (). `ec2messages:*` Amazon Message Delivery Service

Jika Anda memberikan akses ke `ssmmessages:*` dalam kebijakan izin AWS Identity and Access Management (IAM), SSM Agent sambungkan ke `ssmmessages:*` titik akhir, meskipun profil instans IAM Anda dikonfigurasi untuk mengizinkan kedua titik akhir. Ini termasuk kebijakan untuk profil instans IAM dan peran layanan IAM yang telah Anda buat sendiri, dan untuk profil instans IAM yang dibuat oleh konfigurasi manajemen Host dan Konfigurasi Manajemen Quick Setup Host Default.

Jika Anda telah memberikan izin untuk titik akhir dan memantau operasi API menggunakan, misalnya, CloudWatch Metrik, Anda tidak akan melihat panggilan ke. `ec2messages:*`

Namun, Anda dapat dengan aman meninggalkan `ec2messages:*` izin dalam kebijakan Anda saat ini.

Failover koneksi titik akhir

Jika profil instans IAM Anda tidak memberikan izin untuk `ssmmessages:*` saat agen memulai, tetapi hanya`ec2messages:*`, SSM Agent terhubung ke titik akhir`ec2messages:*`. Jika Anda memiliki keduanya `ssmmessages:*` dan `ec2messages:*` pada saat SSM Agent dimulai, tetapi hapus `ssmmessages:*` setelah agen dimulai, SSM Agent segera alihkan koneksi ke `ec2messages:*` titik akhir.

Untuk informasi selengkapnya tentang `ssmmessages` dan `ec2messages:*` titik akhir, lihat topik berikut di Referensi Otorisasi AWS Layanan.

- Tindakan, sumber daya, dan kunci kondisi untuk Amazon Message Gateway Service (`ssmmessages`).
- Tindakan, sumber daya, dan kunci kondisi untuk Amazon Message Delivery Service (`ec2messages:*`)

Operasi API terkait instans

`UpdateInstanceInformation`: SSM Agent memanggil layanan Systems Manager di cloud setiap 5 menit untuk memberikan informasi detak jantung. Panggilan ini diperlukan untuk menjaga heartbeat dengan agen sehingga layanan tahu agen berfungsi seperti yang diharapkan.

`UpdateInstanceAssociationStatus`: Agen menjalankan operasi API ini untuk memperbarui asosiasi. Operasi API ini diperlukan untukState Manager, kemampuan AWS Systems Manager, untuk berfungsi.

`ListInstanceAssociations`: Agen menjalankan operasi API ini untuk melihat apakah State Manager asosiasi baru tersedia. Operasi API ini diperlukan State Manager untuk berfungsi.

`DescribeInstancePropertiesdanDescribeDocumentParameters`: Systems Manager menjalankan operasi API ini untuk merender node tertentu di konsol Amazon EC2. Hasil `DescribeInstanceProperties` operasi ditampilkan di Fleet Manager node. Hasil `DescribeDocumentParameters` operasi ditampilkan di node Documents.

`GetCalendardanPutCalendar`: Systems Manager menjalankan operasi API ini untuk merender dan memperbarui dokumen Change Calendar tipe di Change Calendar konsol.

`RegisterManagedInstance`: SSM Agent menjalankan operasi API ini untuk mendaftarkan server lokal atau mesin virtual (VM) dengan Systems Manager sebagai instance terkelola menggunakan kode aktivasi dan ID, atau untuk mendaftarkan AWS IoT Greengrass Version 2 kredensional. Operasi ini juga disebut oleh instans Amazon EC2 yang menjalankan SSM Agent versi 3.1.x atau yang lebih baru.

Operasi API terkait distributor

SSM Agentberjalan `GetManifest` untuk menentukan persyaratan sistem untuk menginstal atau memperbarui versi AWS Systems Manager Distributor paket tertentu. Ini adalah operasi API lama dan tidak tersedia di Wilayah AWS diluncurkan setelah 2017.

SSM Agentberjalan `PutConfigurePackageResult` untuk mempublikasikan kesalahan instalasi dan metrik latensi untuk paket Distributor publik ke akun pemilik paket.

Referensi: Membuat string tanggal dan waktu yang diformat untuk Systems Manager

Operasi API AWS Systems Manager menerima filter untuk membatasi jumlah hasil yang dikembalikan oleh permintaan. Beberapa operasi API ini menerima filter yang memerlukan string yang diformat untuk mewakili tanggal dan waktu tertentu. Misalnya, operasi API `DescribeSessions` menerima kunci `InvokedAfter` dan `InvokedBefore` sebagai beberapa nilai yang valid untuk suatu objek `SessionFilter`. Contoh lainnya adalah operasi API `DescribeAutomationExecutions`, yang menerima kunci `StartTimeBefore` dan `StartTimeAfter` sebagai beberapa nilai yang valid untuk suatu objek `AutomationExecutionFilter`. Nilai yang Anda berikan untuk kunci ini saat mem-filter permintaan harus sesuai dengan standar ISO 8601. Untuk informasi tentang ISO 8601, lihat ISO 8601.

String tanggal dan waktu yang diformat ini tidak terbatas pada filter. Ada juga operasi API yang memerlukan string dengan format ISO 8601 untuk mewakili tanggal dan waktu tertentu ketika memberikan nilai untuk parameter permintaan. Misalnya, parameter permintaan `AtTime` untuk operasi `GetCalendarState`. String ini sulit dibuat. Gunakan contoh dalam topik ini untuk membuat string tanggal dan waktu yang diformat untuk digunakan dengan operasi API Systems Manager.

Memformat string tanggal dan waktu untuk Systems Manager

Berikut ini adalah contoh dari string tanggal dan waktu string yang diformat ISO 8601.

```
2020-05-08T15:16:43Z
```

Ini mewakili 8 Mei 2020 pukul 15:16 Coordinated Universal Time (UTC). Bagian tanggal kalender pada string diwakili oleh empat digit tahun, dua digit bulan, dan dua digit hari dipisahkan oleh tanda hubung. Hal ini dapat diwakili dalam format berikut.

```
YYYY-MM-DD
```

Bagian waktu pada string dimulai dengan huruf "T" sebagai pembatas, dan kemudian diwakili oleh dua digit jam, dua digit menit, dan dua digit detik yang dipisahkan oleh titik dua. Hal ini dapat diwakili dalam format berikut.

```
hh:mm:ss
```

Bagian waktu pada string diakhiri dengan huruf "Z", yang menunjukkan standar UTC.

Membuat string tanggal dan waktu kustom untuk Systems Manager

Anda dapat membuat string tanggal dan waktu kustom dari mesin lokal Anda menggunakan alat baris perintah pilihan Anda. Sintaks yang Anda gunakan untuk membuat string tanggal dan waktu yang diformat ISO 8601 akan berbeda tergantung pada sistem operasi mesin lokal Anda. Berikut ini adalah contoh bagaimana Anda dapat menggunakan`date` coreutils GNU di Linux, atau PowerShell pada Windows untuk membuat string tanggal dan waktu berformat ISO 8601.

coreutils

```
date '+%Y-%m-%dT%H:%M:%SZ'
```

PowerShell

```
(Get-Date).ToString("yyyy-MM-ddTH:mm:ssZ")
```

Ketika bekerja dengan operasi API Systems Manager, Anda mungkin perlu membuat string tanggal dan waktu historis untuk tujuan pelaporan atau pemecahan masalah. Berikut ini adalah contoh bagaimana Anda dapat membuat dan menggunakan string tanggal dan waktu berformat ISO 8601 historis kustom untuk AWS Tools for PowerShell dan AWS Command Line Interface (AWS CLI).

AWS CLI

- Mengambil minggu terakhir riwayat perintah untuk dokumen 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
```

- Mengambil minggu terakhir riwayat eksekusi otomatisasi.

```
lastWeekStamp=$(date '+%Y-%m-%dT%H:%M:%SZ' -d '7 days ago')

aws ssm describe-automation-executions \
    --filters Key=StartTimeAfter,Values=$lastWeekStamp
```

- Mengambil bulan terakhir riwayat sesi.

```
lastWeekStamp=$(date '+%Y-%m-%dT%H:%M:%SZ' -d '30 days ago')

aws ssm describe-sessions \
    --state History \
    --filters key=InvokedAfter,value=$lastWeekStamp
```

AWS Tools for PowerShell

- Mengambil minggu terakhir riwayat perintah untuk dokumen 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
```

- Mengambil minggu terakhir riwayat eksekusi otomatisasi.

```
$lastWeekStamp = (Get-Date).AddDays(-7).ToString("yyyy-MM-ddTH:mm:ssZ")

Get-SSMAutomationExecutionList `
    -Filters @{Key="StartTimeAfter";Values=$lastWeekStamp}
```

- Mengambil bulan terakhir riwayat sesi.

```
$lastWeekStamp = (Get-Date).AddDays(-30).ToString("yyyy-MM-ddTH:mm:ssZ")

Get-SSMSession `
    -State History `
    -Filters @{Key="InvokedAfter";Value=$lastWeekStamp}
```

Kasus penggunaan dan praktik terbaik

Topik ini mencantumkan kasus penggunaan umum dan praktik terbaik untuk kemampuan AWS Systems Manager. Jika tersedia, topik ini juga mencakup tautan ke posting blog dan dokumentasi teknis yang relevan.

> **Note**
>
> Judul setiap bagian di sini adalah tautan aktif ke bagian yang sesuai dalam dokumentasi teknis.

Otomatisasi

- Buat runbook Otomatisasi layanan mandiri untuk infrastruktur.
- Gunakan Otomatisasi, kemampuan dari AWS Systems Manager, untuk menyederhanakan pembuatan Amazon Machine Images (AMIs) dari AWS Marketplace atau AMIs kustom, menggunakan dokumen Systems Manager publik (dokumen SSM) atau dengan mengotorisasi alur kerja Anda sendiri.
- Menyusun dan mempertahankan AMIs menggunakan `AWS-UpdateLinuxAmi` dan runbook Otomatisasi `AWS-UpdateWindowsAmi`, atau menggunakan runbook Otomatisasi kustom yang Anda buat.

Inventaris

- Gunakan Inventaris, kemampuan dari AWS Systems Manager, dengan AWS Config untuk mengaudit konfigurasi aplikasi Anda dari waktu ke waktu.

Maintenance Windows

- Tentukan jadwal untuk melakukan tindakan yang berpotensi mengganggu pada node Anda seperti patch sistem operasi (OS), pembaruan driver, atau instalasi perangkat lunak.
- Untuk informasi tentang perbedaan antara State Manager danMaintenance Windows, kemampuanAWS Systems Manager, lihatMemilih antara State Manager dan Maintenance Windows.

Parameter Store

- GunakanParameter Store, kemampuanAWS Systems Manager, untuk mengelola pengaturan konfigurasi global secara terpusat.
- Bagaimana AWS Systems ManagerParameter Store menggunakan AWS KMS.
- AWS Secrets ManagerRahasia referensi dari Parameter Store parameter.

Patch Manager

- GunakanPatch Manager, kemampuanAWS Systems Manager, untuk meluncurkan tambalan dalam skala besar dan meningkatkan visibilitas kepatuhan armada di seluruh node Anda.
- Integrasikan Patch Manager dengan AWS Security Hub untuk menerima peringatan saat node di armada Anda tidak sesuai dan pantau status patching armada Anda dari sudut pandang keamanan. Ada biaya atas penggunaan Security Hub. Untuk informasi selengkapnya, lihat Harga.
- Gunakan hanya satu metode pada satu waktu untuk memindai node terkelola untuk kepatuhan patch untuk menghindari penimpaan data kepatuhan secara tidak sengaja.

Run Command

- Kelola Instans dalam Skala Besar tanpa Akses SSH Menggunakan Run Command EC2.
- Audit semua panggilan API yang dilakukan oleh atau atas namaRun Command, kemampuanAWS Systems Manager, penggunaanAWS CloudTrail.
- Saat Anda mengirim perintah menggunakanRun Command, jangan sertakan informasi sensitif yang diformat sebagai teks biasa, seperti kata sandi, data konfigurasi, atau rahasia lainnya. Semua aktivitas Systems Manager API di akun Anda dicatat dalam bucket S3 untuk AWS CloudTrail log. Ini berarti bahwa setiap pengguna dengan akses ke bucket S3 dapat melihat nilai plaintext dari rahasia tersebut. Untuk alasan ini, kami sarankan untuk membuat dan menggunakan `SecureString` parameter untuk mengenkripsi data sensitif yang Anda gunakan dalam operasi Systems Manager Anda.

  Untuk informasi selengkapnya, lihat Membatasi akses ke parameter Systems Manager menggunakan kebijakan IAM.

> **Note**
>
> Secara default, file log yang dikirimkan CloudTrail ke bucket Anda dienkripsi oleh enkripsi sisi server Amazon dengan kunci enkripsi yang dikelola Amazon S3 (SSE-S3). Untuk menyediakan lapisan keamanan yang dapat dikelola secara langsung, Anda dapat menggunakan enkripsi sisi server dengan AWS KMS —managed keys (SSE-KMS) untuk file log Anda. CloudTrail
> Untuk informasi selengkapnya, lihat Mengenkripsi file CloudTrail log dengan AWS KMS — kunci terkelola (SSE-KMS) di Panduan Pengguna. AWS CloudTrail

- Gunakan target dan fitur kontrol tingkat Run Command untuk melakukan operasi perintah bertahap.
- Gunakan izin akses berbutir halus untuk Run Command (dan semua kemampuan Systems Manager) dengan menggunakan kebijakan AWS Identity and Access Management (IAM).

Session Manager

- Audit aktivitas sesi di Akun AWS menggunakan AWS CloudTrail.
- Log data sesi di Anda Akun AWS menggunakan Amazon CloudWatch Logs atau Amazon S3.
- Kontrol akses sesi pengguna ke instans.
- Batasi akses ke perintah dalam sesi.
- Nonaktifkan atau aktifkan izin administratif akun pengguna ssm.

State Manager

- Perbarui SSM Agent setidaknya sebulan sekali menggunakan `AWS-UpdateSSMAgent` dokumen yang telah dikonfigurasi sebelumnya.
- (Windows) Unggah modul PowerShell atau DSC ke Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3), dan gunakan. `AWS-InstallPowerShellModule`
- Gunakan tag untuk membuat grup aplikasi untuk node Anda. Dan kemudian target node menggunakan `Targets` parameter alih-alih menentukan ID node individu.
- Secara otomatis memulihkan temuan yang dihasilkan oleh Amazon Inspector dengan menggunakan Systems Manager.

- Gunakan repositori konfigurasi terpusat untuk dokumen SSM Anda, dan bagikan dokumen di seluruh organisasi.
- Untuk informasi selengkapnya tentang perbedaan antara State Manager dan Maintenance Windows, lihat Memilih antara State Manager dan Maintenance Windows.

Node terkelola

- Systems Manager memerlukan referensi waktu yang akurat untuk melakukan operasinya. Jika tanggal dan waktu node Anda tidak disetel dengan benar, mereka mungkin tidak cocok dengan tanggal tanda tangan permintaan API Anda. Hal ini dapat menyebabkan kesalahan atau fungsionalitas yang tidak lengkap. Misalnya, node dengan pengaturan waktu yang salah tidak akan disertakan dalam daftar node terkelola Anda.

  Untuk informasi tentang pengaturan waktu pada node Anda, lihat topik berikut:

  - Atur waktu untuk instance Linux Anda
  - Mengatur waktu untuk instance Windows
- Pada node yang dikelola Linux, verifikasi tanda tangan SSM Agent.

Info lebih lanjut

- Praktik terbaik keamanan untuk Systems Manager

Menghapus sumber daya dan artefak Systems Manager

Sebagai praktik terbaik, kami menyarankan Anda menghapus sumber daya dan artefak Systems Manager jika Anda tidak perlu lagi melihat data tentang sumber daya tersebut atau menggunakan artefak dengan cara apa pun. Tabel berikut mencantumkan setiap kemampuan atau artefak Systems Manager dan tautan ke informasi selengkapnya tentang menghapus sumber daya atau artefak yang dibuat oleh Systems Manager.

Kemampuan atau artefak	Detail
Application Manager	Anda tidak dapat menghapus aplikasiApplication Manager, tetapi Anda dapat menghapus aplikasi dari layanan dengan menghapus

Kemampuan atau artefak	Detail
	tag, Resource Groups, atau AWS CloudFormationtumpukan yang mendasarinya.
Otomatisasi	Jika Anda membuat AWS sumber daya dengan menggunakan Systems Manager Automation, Anda harus menghapus sumber daya tersebut secara manual dengan menggunakan yang sesuaiAWS Management Console. Jika Anda membuat runbook kustom, Anda dapat menghapus dokumen SSM yang mendasarinya. Untuk informasi selengkapnya, lihat Menghapus dokumen SSM kustom.
Change Calendar	Anda dapat menghapus kalender perubahan dan acara kalender perubahan. Untuk informasi selengkapnya, lihat Menghapus kalender perubahan dan MenghapusChange Calendar acara.
Change Manager	Anda dapat menghapus template perubahan. Untuk informasi selengkapnya, lihat Menghapus templat perubahan.
Kepatuhan	Kepatuhan Systems Manager secara otomatis menampilkan data kepatuhan tentang Patch Manager patching dan State Manager asosiasi. Anda tidak dapat menghapus data ini. Jika Anda mengonfigurasi sinkronisasi data sumber daya untuk memusatkan data kepatuhan dalam bucket S3, Anda dapat menghapus sinkronisasi tersebut. Untuk informasi selengkapnya, lihat Menghapus sinkronisasi data sumber daya untuk Kepatuhan.

Kemampuan atau artefak	Detail
Distributor	Anda dapat menghapus paket diDistrib utor. Untuk informasi selengkapnya, lihat Menghapus paket.
Explorer	Anda dapat memutuskan sambungan dari sumber yang Explorer dikumpulkan OpsData. Untuk informasi selengkapnya, lihat Mengedit sumber data Systems Manager Explorer. Anda juga dapat menghapus sinkronisasi data sumber daya yang digunakan Explorer untuk menggabungkan OpsData dan OpsItems dari beberapa akun Wilayah AWS dan ke satu bucket Amazon Simple Storage Service (Amazon S3). Untuk informasi selengkapnya, lihat Menghapus sinkronisasi data sumber daya untuk Systems Manager Explorer. Untuk informasi tentang menghapus bucket S3, lihat Menghapus bucket di Panduan Pengembang Layanan Email Sederhana Amazon.
Fleet Manager	Anda tidak dapat menghapus node terkelola dengan menggunakanFleet Manager. Anda harus menggunakan Amazon Elastic Compute Cloud (Amazon EC2). Untuk informasi selengkapnya, lihat Menghentikan instans Anda (Linux) dan Mengakhiri instans Anda (Windows).

Kemampuan atau artefak	Detail
Inventaris	Anda dapat menghentikan pengumpulan data Inventaris dengan menghapus State Manager asosiasi yang menentukan jadwal dan sumber daya untuk mengumpulkan metadata. Untuk informasi selengkapnya, lihat Menghentikan pengumpulan data dan menghapus data inventaris. Jika Anda tidak lagi ingin menggunakan AWS Systems Manager Inventaris untuk melihat metadata tentang AWS sumber daya Anda, sebaiknya hapus sinkronisasi data sumber daya yang digunakan untuk pengumpulan data inventaris. Untuk informasi selengkapnya, lihat Menghapus sinkrondata sumber daya data sumber daya data sumber daya.
Maintenance Windows	Anda dapat menghapus jendela pemeliharaan, target jendela pemeliharaan, dan tugas jendela pemeliharaan. Untuk informasi selengkapnya, lihat Memperbarui atau menghapus sumber daya jendela pemeliharaan (konsol).
OpsCenter	Anda dapat menghapus individu OpsItem dengan memanggil operasi Delete OpsItem API menggunakan AWS Command Line Interface atau AWS SDK. Anda tidak dapat menghapus file OpsItem diAWS Management Console. Untuk informasi selengkapnya, lihat Hapus OpsItems.
Parameter Store	Anda dapat menghapus parameter yang telah Anda buat. Untuk informasi selengkapnya, lihat Menghapus parameter Systems Manager.

Kemampuan atau artefak	Detail
Patch Manager	Anda dapat menghapus baseline patch kustom. Untuk informasi selengkapnya, lihat Memperbarui atau menghapus baseline patch kustom.
Pengaturan Cepat	Anda dapat menghapus asosiasi yang dibuat oleh Quick Setup. Asosiasi disimpan dan diproses olehState Manager. Untuk informasi selengkapnya, lihat Menghapus asosiasi.
Run Command	Setelah perintah selesai diproses, informasi tentangnya disimpan di tab Riwayat perintah. Anda tidak dapat menghapus informasi dari tab Riwayat perintah.
Peran tertaut layanan	Systems Manager secara otomatis membuat peran terkait layanan untuk beberapa kemampuan. Anda dapat menghapus peran ini. Untuk informasi selengkapnya, lihat MenghapusAWSServiceRoleForAmazonSSM peran terkait layanan untukSystems Manager.
Session Manager	Session Managertidak menyimpan data tentang sumber daya Anda setelah Anda mengakhiri sesi. Untuk mengakhiri sesi, lihatMengakhiri sesi.

Kemampuan atau artefak	Detail
SSM Agent	Anda dapat menghapus secara manual SSM Agent dari node Anda. Untuk informasi lain, lihat topik berikut. • Linux: Menghapus instalasi SSM Agent dari instance Linux • macOS: Menghapus instalasiSSM Agentdari macOScontoh • Windows Server: Buka Panel kontrol dan kemudian pilih Tambah/hapus program.
State Manager	Anda dapat menghapus asosiasi. Untuk informasi selengkapnya, lihat Menghapus asosiasi.
Layanan dokumen Systems Manager	Anda tidak dapat menghapus runbook yang disediakan oleh AWS atauAWS Support, tetapi Anda dapat menghapus runbook kustom. Untuk informasi selengkapnya, lihat Menghapus dokumen SSM kustom.

Memilih antara State Manager dan Maintenance Windows

State ManagerdanMaintenance Windows, kedua kemampuan AWS Systems Manager, dapat melakukan beberapa jenis pembaruan serupa pada node terkelola Anda. Pemilihan bergantung pada apakah Anda perlu mengotomatiskan kepatuhan sistem atau melakukan tugas prioritas tinggi dan mendesak selama periode yang ditentukan.

State ManagerdanMaintenance Windows: Kasus penggunaan utama

State Manager, kemampuan AWS Systems Manager, menetapkan dan mempertahankan konfigurasi status yang ditargetkan untuk node terkelola dan AWS sumber daya dalam Anda Akun AWS. Anda dapat menentukan kombinasi konfigurasi dan target sebagai objek asosiasi. State Manageradalah kemampuan yang disarankan jika Anda ingin mempertahankan semua node terkelola di akun Anda

dalam keadaan konsisten, menggunakan Auto Scaling Amazon EC2 untuk menghasilkan node baru, atau memiliki persyaratan pelaporan kepatuhan yang ketat untuk node terkelola di akun Anda.

Kasus penggunaan utama untuk State Manager adalah sebagai berikut:

- Skenario Auto Scaling: State Manager dapat memantau semua node baru yang diluncurkan dalam akun baik secara manual atau melalui grup Auto Scaling. Jika ada asosiasi dalam akun yang menargetkan node baru itu (melalui tag atau semua node), maka asosiasi tertentu secara otomatis diterapkan ke node baru.
- Pelaporan kepatuhan: State Manager dapat mendorong pelaporan kepatuhan status yang diperlukan untuk sumber daya di akun Anda.
- Mendukung semua node: State Manager dapat menargetkan semua node dalam akun tertentu.

Jendela pemeliharaan mengambil satu atau lebih tindakan di sumber daya AWS dalam jendela waktu tertentu. Anda dapat menentukan satu jendela pemeliharaan dengan waktu mulai dan waktu selesai. Anda dapat menentukan beberapa tugas untuk dijalankan dalam jendela pemeliharaan ini. GunakanMaintenance Windows, kemampuan AWS Systems Manager, jika operasi prioritas tinggi Anda termasuk menambal node terkelola Anda, menjalankan beberapa jenis tugas pada node Anda selama periode pembaruan, atau mengontrol kapan operasi pembaruan dapat dijalankan pada node Anda.

Kasus penggunaan utama untuk Maintenance Windows adalah sebagai berikut:

- Menjalankan beberapa dokumen: Jendela pemeliharaan dapat menjalankan beberapa tugas. Setiap tugas dapat menggunakan jenis dokumen yang berbeda. Sebagai hasilnya, Anda dapat membuat alur kerja yang kompleks menggunakan tugas yang berbeda dalam satu jendela pemeliharaan.
- Patching: Jendela pemeliharaan dapat memberikan dukungan patching untuk semua node terkelola dalam satu Wilayah yang ditandai dengan tag atau grup sumber daya tertentu. Karena menambal biasanya melibatkan menurunkan node (misalnya, menghapus node dari penyeimbang beban), menambal, dan pasca pemrosesan (menempatkan node kembali ke produksi), penambalan dapat dicapai sebagai serangkaian tugas dalam jendela waktu patch tertentu.

> **Note**
>
> Menggunakan jendela pemeliharaan, operasi patching Anda terbatas pada satu Wilayah dalam satu akun. Dengan menggunakan kebijakan tambalan yang dibuat diQuick Setup,

kemampuan Systems Manager, Anda dapat mengonfigurasi penambalan untuk beberapa atau semua akun dan Wilayah dalam organisasi yang dibuat. AWS Organizations Untuk informasi selengkapnya, lihat Menggunakan kebijakan Quick Setup tambalan.

- Tindakan Window: Jendela pemeliharaan dapat membuat satu atau lebih rangkaian tindakan dimulai dalam jendela waktu tertentu. Jendela pemeliharaan tidak akan dimulai di luar jendela itu. Tindakan sudah mulai berlanjut sampai selesai, bahkan jika selesai di luar jendela waktu.

Tabel berikut membandingkan fitur utama State Manager danMaintenance Windows.

Fitur	State Manager	Maintenance Windows
AWS CloudFormation integrasi	AWS CloudFormation template mendukung State Manager asosiasi.	AWS CloudFormation template mendukung pemeliharaan jendela, target jendela, dan tugas jendela.
Kepatuhan	Setiap State Manager asosiasi melaporkan kepatuhan sehubungan dengan keadaan yang diperlukan dari sumber daya yang ditargetkan. Anda dapat menggunakan Dasbor Kepatuhan untuk mengumpulkan dan melihat kepatuhan yang dilaporkan.	Tidak berlaku.
Integrasi Manajemen Konfigurasi	State Managermendukung solusi status bertarget eksternal seperti Microsoft Desired PowerShell State Configuration (DSC), Ansible playbook, dan Chef resep. Anda dapat menggunakan State Manager asosiasi untuk menguji apakah solusi Manajemen Konfigurasi	Tidak berlaku.

Fitur	State Manager	Maintenance Windows
	berfungsi dan menerapkan perubahan konfigurasinya ke node saat Anda siap.	
Dokumen	State Managerkonfigurasi dapat didefinisikan sebagai dokumen Kebijakan (untuk mengumpulkan informasi inventaris), runbook Otomasi, untuk AWS sumber daya seperti bucket Amazon Simple Storage Service (Amazon S3), atau dokumen Command Systems Manager (dokumen SSM) untuk node terkelola.	Maintenance Windowsko nfigurasi dapat didefinisikan sebagai dokumen otomatisa si (tindakan multi-langkah dengan alur kerja persetuju an opsional) atau dokumen SSM (status wajib untuk node terkelola).
Pemantauan	State Managermemantau perubahan konfigurasi, asosiasi, atau status node (misalnya, node baru datang online). Ketika State Manager mendeteksi perubahan ini, asosiasi yang diberikan diterapkan kembali ke node yang awalnya ditargetkan dengan asosiasi itu.	Tidak berlaku.

Fitur	State Manager	Maintenance Windows
Prioritas dalam tugas	Tidak berlaku.	Tugas dalam jendela pemeliharaan dapat diberi prioritas. Semua tugas dengan prioritas yang sama dijalankan secara paralel. Tugas dengan prioritas yang lebih rendah dijalankan setelah tugas dengan prioritas yang lebih tinggi untuk mencapai status akhir. Tidak ada cara untuk menjalankan tugas secara kondisional. Setelah tugas dengan prioritas lebih tinggi mencapai status akhirnya, tugas prioritas berikutnya akan berjalan, terlepas dari status tugas sebelumnya.

Fitur	State Manager	Maintenance Windows
Kontrol keamanan	State Managermendukung dua kontrol keselamatan saat menyebarkan konfigurasi di armada besar. Anda dapat menggunakan konkurensi maksimum untuk menentuka n berapa banyak node atau sumber daya bersamaan yang harus memiliki konfigurasi yang diterapkan. Anda dapat menentukan tingkat kesalahan maksimum yang dapat digunakan untuk menjeda State Manager asosiasi jika sejumlah atau persentase kesalahan tertentu terjadi di seluruh armada.	Jendela pemeliharaan mendukung dua kontrol keselamatan saat men-deploy konfigurasi di seluruh armada besar. Anda dapat menggunakan konkurensi maksimum untuk menentuka n berapa banyak node atau sumber daya bersamaan yang harus memiliki konfigura si yang diterapkan. Anda dapat menentukan tingkat kesalahan maksimum yang dapat digunakan untuk menjeda tindakan di jendela pemeliharaan jika sejumlah atau persentase kesalahan tertentu terjadi di seluruh armada.

Fitur	State Manager	Maintenance Windows
Penjadwalan	Anda dapat menjalankan State Manager asosiasi sesuai permintaan, pada interval cron tertentu, pada tingkat tertentu, atau setelah dibuat. Ini berguna jika Anda ingin mempertahankan status sumber daya yang diperluka n secara konsisten dan tepat waktu. **Important** Ekspresi cron untuk State Manager asosiasi tidak mendukung bidang bulan, seperti `03` atau `MAR` untuk bulan Maret. Jika Anda memerlukan pembaruan konfigurasi bulanan atau triwulana n, jendela pemelihar aan dapat memenuhi kebutuhan Anda dengan baik. Untuk informasi selengkap nya, lihat Referensi : Ekspresi cron dan rate untuk Systems Manager.	Jendela pemeliharaan mendukung beberapa opsi penjadwalan termasuk ekspresi `at` (misalnya, `"at(2021-07-07T13: 15:30)"`), ekspresi cron dan laju, cron dengan offset, dan waktu mulai serta waktu selesai untuk kapan jendela pemeliharaan harus dijalanka n, dan waktu batas untuk menentukan kapan harus menghentikan penjadwalan dalam jangka waktu tertentu.

Fitur	State Manager	Maintenance Windows
Penargetan	State Managerasosiasi dapat menargetkan satu atau lebih node dengan menggunakan ID node, tag, atau grup sumber daya. State Managerdapat menargetkan semua node yang dikelola dalam akun tertentu.	Jendela pemeliharaan dapat menargetkan satu atau beberapa node menggunak an ID node, tag, atau grup sumber daya.
Tugas dalam jendela pemeliharaan	Tidak berlaku.	Jendela pemeliharaan dapat mendukung satu atau beberapa tugas di mana setiap tugas menargetkan runbook Otomatisasi tertentu atau tindakan dokumen Command. Semua tugas dalam jendela pemelihar aan berjalan secara paralel kecuali prioritas yang berbeda ditetapkan untuk tugas yang berbeda. Secara keseluruhan, jendela pemeliharaan mendukung empat jenis tugas: • AWS Systems Manager Run Commandperintah • AWS Systems Manager Alur kerja otomatisasi • AWS Lambda fungsi • AWS Step Functions tugas

Informasi terkait

Sumber daya terkait berikut dapat membantu Anda ketika bekerja dengan layanan ini.

Harga

Beberapa kemampuan Systems Manager mengenakan biaya. Untuk informasi selengkapnya, lihat harga AWS Systems Manager.

AWS Systems Managerperpustakaan dokumentasi

AWS Systems ManagerDokumentasi — Akses semua dokumentasi pengguna untuk Systems Manager, termasukAWS AppConfig, Incident Manager, dan AWS Systems Manager untuk SAP.

AWS re:Post

AWS re:Post – Layanan tanya jawab (Q&A) terkelola AWS yang menawarkan jawaban yang bersumber dari banyak orang dan ditinjau oleh ahli untuk pertanyaan teknis Anda.

AWSBlog & Podcast

Baca posting blog tentang Systems Manager di Kategori Alat AWS Manajemen, dan posting lain yang ditandai dengan `#Systems Manager`.

Kuota layanan

Tinjau kuota layanan Systems Manager di. Referensi Umum Amazon Web Services Kecuali dinyatakan lain, setiap kuota berlaku untuk satu Wilayah dalam suatuAkun AWS.

Referensi Otorisasi Layanan untuk Systems Manager

Dalam Referensi Otorisasi AWS Layanan, lihat informasi tentang tindakan, sumber daya, dan kunci konteks kondisi yang dapat Anda gunakan dalam kebijakan AWS Identity and Access Management (IAM) untuk Systems Manager.

AWS Systems ManagerPerjanjian Tingkat Layanan

Perjanjian Tingkat AWS Systems Manager Layanan (SLA) adalah kebijakan yang mengatur penggunaan Systems Manager dan berlaku secara terpisah untuk masing-masing Akun AWS menggunakan Systems Manager.

Sumber daya AWS umum

Sumber daya umum berikut dapat membantu Anda ketika bekerja dengan AWS.

- Kelas dan Lokakarya – Tautan ke kursus khusus dan berbasis peran, selain laboratorium mandiri, untuk membantu mempertajam keterampilan AWS Anda dan mendapatkan pengalaman praktis.
- Pusat Developer AWS – Jelajahi tutorial, unduh peralatan, dan pelajari tentang acara developer AWS.
- Alat Developer AWS – Tautan ke alat, SDK, kit alat IDE, dan alat baris perintah developer untuk mengembangkan serta mengelola aplikasi AWS.
- Memulai Pusat Sumber Daya – Pelajari cara menyiapkan Akun AWS, bergabung dengan komunitas AWS, dan meluncurkan aplikasi pertama Anda.
- Tutorial Hands-On - Ikuti step-by-step tutorial untuk meluncurkan aplikasi pertama Anda. AWS
- Laporan Resmi AWS – Tautan ke daftar laporan resmi teknis AWS yang komprehensif, yang mencakup topik seperti arsitektur, keamanan, dan ekonomi serta ditulis oleh Arsitek Solusi AWS atau ahli teknis lainnya.
- Pusat AWS Support – Hub untuk membuat dan mengelola kasus AWS Support Anda. Juga mencakup tautan ke sumber daya yang bermanfaat lainnya, seperti forum, FAQ teknis, status kondisi layanan, dan AWS Trusted Advisor.
- AWS Support— Halaman web utama untuk informasi tentangAWS Support, saluran dukungan respons cepat untuk membantu Anda membangun dan menjalankan aplikasi di cloud. one-on-one
- Hubungi Kami – Titik kontak pusat untuk pertanyaan tentang tandaihan AWS, akun, peristiwa, penyalahgunaan, dan masalah lainnya.
- Persyaratan Situs AWS – Informasi detail tentang hak cipta dan merek dagang kami; akun, lisensi, dan akses situs Anda; serta topik lainnya.

Riwayat dokumen

Tabel berikut menjelaskan perubahan penting pada dokumentasi sejak rilis terakhir AWS Systems Manager. Untuk notifikasi tentang pembaruan dokumentasi ini, Anda dapat berlangganan ke umpan RSS.

- Versi API: 2014-11-06

Perubahan	Deskripsi	Tanggal
Pembaruan topik: kebijakan AWS terkelola untuk AWS Systems Manager	Topik kebijakan AWS terkelola untuk AWS Systems Manager telah memberikan informasi tentang empat kebijakan terkelola untuk Systems Manager yang telah diperkenalkan atau diperbarui sejak 12 Maret 2021. Kami telah menambahkan bagian ke topik ini dengan informasi tentang 12 kebijakan terkelola lainnya untuk digunakan dengan Systems Manager yang dibuat atau terakhir diperbarui sebelum tanggal tersebut. Untuk detailnya, lihat Kebijakan terkelola tambahan untuk Systems Manager.	Maret 1, 2024
Parameter Storesekarang mendukung berbagi lintas akun	Sekarang Anda dapat membagikan parameter lanjutan secara aman dan efisien di seluruh Akun AWS atau di dalam AWS Organisasi Anda dengan menyiapkan pembagian	Februari 21, 2024

	sumber daya. Berbagi sumber daya memungkinkan Anda memusatkan manajemen konfigurasi aplikasi dan mengurangi biaya operasional berbagi parameter dengan setiap akun yang Anda miliki. Parameter dapat dibagikan di seluruh akun menggunakan Parameter Store konsol, AWS RAM konsol, atau file AWS CLI. Untuk informasi selengkapnya, lihat Bekerja dengan parameter bersama.	
Peningkatan tindakan otomatisasi	Anda sekarang dapat menggunakan `isCritical` properti `onFailure` dan dengan `aws:approve` tindakan. Untuk informasi selengkapnya tentang `aws:approve` tindakan, lihat aws:approve — Menjeda otomatisasi untuk persetujuan manual.	Februari 12, 2024
Dukungan versi operasi tambahan untuk Patch Manager	Kami telah menambahkan ke daftar versi sistem operasi yang didukung untuk Patch Manager. Support telah ditambahkan sebagai berikut: • Debian Server11.x dan 12.x • macOS14.0 (Sonoma) • SUSE Linux Enterprise Server(SLES) 15,5 • Ubuntu Server23.04	4 Januari 2024

Konfigurasikan SSM Agent pembaruan otomatis menggunakan Application Manager konsol	Anda sekarang dapat menggunakan Application Manager konsol untuk mengotomatiskan SSM Agent pembaruan untuk instance aplikasi Anda. Untuk informasi selengkapnya, lihat Bekerja dengan instance aplikasi Anda.	21 Desember 2023
Proses yang diperbarui untuk mendaftarkan mesin non Amazon EC2 di lingkungan hybrid dan multicloud	Systems Manager sekarang menyediakan `ssm-setup-cli` untuk membantu Anda mendaftarkan mesin non Amazon Elastic Compute Cloud (Amazon EC2) di lingkungan hybrid dan multicloud. Untuk informasi selengkapnya, lihat Install SSM Agent for a hybrid environment (Linux) dan Install SSM Agent for a hybrid environment (Windows).	Desember 20, 2023

Kelola volume Amazon EBS menggunakan Fleet Manager	Anda sekarang dapat menggunakanFleet Manager, kemampuan AWS Systems Manager, untuk mengelola volume Amazon Elastic Block Store pada instans terkelola Anda. Misalnya, Anda dapat menginisialisasi volume EBS, memformat partisi, dan memasang volume agar tersedia untuk digunakan. Untuk informasi selengkapnya, lihat manajemen volume EBS.	14 Desember 2023
Session Managerpeningkatan plugin	Menambahkan dukungan untuk meneruskan respons StartSessionAPI sebagai variabel lingkungan ke session-manager-plugin.	Desember 4, 2023
Pengalaman desain visual baru untuk runbook Otomasi	Anda sekarang dapat membuat dan mengedit runbook menggunakan pengalaman desain visual baru yang dikembangkan oleh Systems Manager Automation. Pengalaman desain visual menyediakan drag-and-drop antarmuka kode rendah sehingga Anda dapat membuat dan mengedit runbook dengan lebih mudah. Untuk informasi selengkap nya, lihat Pengalaman desain visual untuk runbook Otomasi.	26 November 2023

Tindakan Otomasi Systems Manager baru, elemen data, dan penyempurnaan fungsional untuk runbook	Anda sekarang dapat mengulang beberapa tindakan dalam runbook menggunak an `aws:loop` tindakan. Tindakan baru ini mendukung `do while` dan `for each` menggayakan loop. Selain itu, dengan menggunakan elemen data variabel baru, Anda dapat menentukan, mereferen sikan, dan memperbarui nilai secara dinamis dalam konteks runbook. Untuk memperbarui nilai variabel di runbook Anda, gunakan `aws:updateVariable` tindakan baru. Otomasi juga telah menambahkan dukungan untuk konversi tipe data dinamis untuk output. Ini berarti bahwa jika nilai output tidak cocok dengan tipe data yang Anda tentukan, Automation mencoba mengonversi tipe data. Misalnya, jika nilai yang dikembalikan adalah`Integer`, tetapi yang `Type` ditentukan adalah`String`, nilai output akhir adalah `String` nilai. Terakhir, Automation sekarang mendukung ekspresi filter JsonPath untuk penyeleksi. Untuk informasi selengkapnya, lihat topik berikut:	17 November 2023

	• aws:loop - Ulangi langkah-langkah dalam otomatisasi • AWS: UpdateVariable - Memperbarui nilai untuk variabel runbook • Elemen dan parameter data - Elemen data tingkat atas • Menggunakan output tindakan sebagai input. • Menggunakan JsonPath di runbook.	
Dukungan Wilayah yang diperbarui untuk koneksi Remote Desktop Protocol (RDP)	Fleet ManagerRemote Desktop, yang didukung oleh NICE DCV, memberi Anda konektivitas aman ke Windows Server instans langsung dari konsol Systems Manager. Tiga Wilayah tambahan berikut telah diaktifkan untuk koneksi Fleet Manager Remote Desktop: • Africa (Cape Town) (af-south-1) • Asia Pasifik (Jakarta) (ap-tenggara 3) • Israel (Tel Aviv) (tengah-1)	15 November 2023

Patch Manager: Dukungan versi OS yang diperluas untuk RHEL dan macOS	Patch Managersekarang mendukung versi sistem operasi tambahan berikut: • Red Hat Enterprise Linux: versi 8.8 • macOS: 11.5— 11:7 (Big Sur) • macOS: 12.0—12.6 (Monterey) • macOS: 13.0—13.5 (Ventura)	23 Oktober 2023
OpsCenterAPI Baru - Hapus OpsItem	OpsCentersekarang menawarkan Delete OpsItem API untuk menghapus individuOpsItems. Untuk informasi selengkapnya, lihat DeleteOpsItem di dalam Referensi API AWS Systems Manager .	20 Oktober 2023
Jenis Quick Setup konfigurasi baru: SSM Agent pembaruan untuk seluruh organisasi	Jenis konfigurasi baru Konfigurasi Manajemen Host Default memungkin kan administrator organisas i, sebagaimana didefinisikan dalam AWS Organizations, untuk meminta pemeriksa an otomatis dan pembaruan SSM Agent pada semua instans EC2 di akun dan Wilayah organisasi. Untuk informasi selengkapnya, lihat Manajemen Host Default untuk organisasi.	16 Oktober 2023

Format judul dan deskripsi baru untuk OpsItems dibuat oleh CloudWatch Application Insights	Judul dan deskripsi untuk OpsItems dibuat oleh CloudWatch Application Insights berubah menjadi format yang ditingkatkan pada 16 Oktober 2023. Untuk melihat format baru, lihat Wawasan CloudWatch Aplikasi Amazon.	September 29, 2023

Support untuk beberapa resolusi tampilan dalam koneksi Fleet Manager RDP	Saat Anda terhubung ke node Windows Server terkelola menggunakan opsi Remote Desktop protocol (RDP) diFleet Manager, Anda sekarang dapat memilih resolusi tampilan. Sebelumnya, semua koneksi menggunakan resolusi tetap 720P (1366 x 768). Anda sekarang dapat memilih dari yang berikut untuk setiap koneksi: • Beradaptasi Secara Otomatis (menentukan resolusi optimal berdasarkan ukuran layar yang terdeteksi) • 1920 x 1080 • 1400 x 900 • 1366 x 768 • 800 x 600 Untuk selengkapnya, lihat Connect ke node terkelola menggunakan Remote Desktop.	September 22, 2023

Topik baru: ID dasar patch acak dalam operasi kebijakan tambalan	Kami telah menambahkan konten untuk menjelaskan bagaimana kebijakan Quick Setup tambalan menggunakan `Baseline0verride` parameter dalam dokumen Perintah `AWS-RunPatchBaseline` SSM untuk menghasilkan ID acak untuk garis dasar tambalan setiap kali operasi kebijakan tambalan dijalankan. Untuk selengkapnya, lihat ID dasar patch acak dalam operasi kebijakan tambalan.	September 22, 2023
Wawasan operasional baru untuk mengelola OpsItems	OpsCentersekarang termasuk wawasan operasional yang disebut Sumber Daya menghasilkan paling banyak OpsItems. Wawasan jenis ini dihasilkan ketika AWS sumber daya memiliki lebih dari 10 terbukaOpsItems. Gunakan wawasan ini untuk menemukan sumber daya yang bermasalah. Gunakan `AWS-BulkResolveOpsItems` runbook dari dalam wawasan untuk menyelesaikan dengan cepat OpsItems terkait dengan sumber daya. Untuk informasi selengkapnya, lihat Menganalisis wawasan operasional untuk mengurangi OpsItems.	September 22, 2023

Kunci publik GPG diperbarui	Kunci publik baru telah dibuat untuk memverifikasi tanda tanganSSM Agent. Untuk informasi selengkapnya, lihat Memverifikasi tanda tangan SSM Agent.	5 September 2023
Support ditambahkan untuk versi tambahan AlmaLinux ,Oracle Linux,RHEL, dan Rocky Linux	Daftar sistem operasi yang didukung untuk AWS Systems Managerdan Patch Managertelah diperbarui untuk mencerminkan dukungan versi OS tambahan berikut: • AlmaLinux: 9.2 • Oracle Linux: 8.7 dan 9.2 • Red Hat Enterprise Linux(RHEL): 8.7, 9.1, dan 9.2 • Rocky Linux: 8.6 dan 8.7, 9.0-9.2	Agustus 30, 2023

OpsCentermenambahkan dukungan untuk pemformatan Markdown di bidang OpsItem deskripsi.	OpsCentersekarang mendukung pemformatan Markdown di bidang OpsItem deskripsi. Jenis pemformatan Markdown berikut didukung: • Paragraf • Jarak baris • Garis Horisontal • Judul • Format Teks • Tautan • Daftar Untuk informasi selengkap nya, lihat Menggunakan Penurunan Harga di Konsol di Panduan Memulai dengan AWS Management Console Memulai.	18 Agustus 2023
Versi baru dari AWS Parameter dan Rahasia Lambda Extension	Versi baru dari AWS Parameter dan Rahasia Lambda Extension sekarang tersedia. Selain itu, dukungan ekstensi telah ditambahkan untuk Wilayah Asia Pasifik (Melbourne) (ap-tenggara 4) dan Israel (Tel Aviv) (il-central-1) (dan arsitektu r saja.) x86_64 x86 Untuk informasi selengkapnya, lihat Menggunakan Parameter Store parameter dalam AWS Lambda fungsi.	16 Agustus 2023

Pembaruan: Menambahkan informasi tentang izin yang diperlukan untuk bucket kebijakan Quick Setup tambalan	Saat Anda membuat kebijakan tambalan, Quick Setup buat bucket Amazon S3 yang berisi file bernama. `baseline_overrides.json` File ini menyimpan informasi tentang garis dasar tambalan yang Anda tentukan untuk kebijakan tambalan Anda. Saat mengonfigurasi kebijakan tambalan, Anda memiliki opsi untuk memilih kotak centang Tambahkan kebijakan IAM yang diperlukan ke profil instans yang ada yang dilampirkan ke instance Anda. Jika Anda memilih untuk tidak memilih opsi ini, Anda harus secara manual memberikan sumber daya tertentu dengan izin untuk mengakses bucket ini atau operasi kebijakan Anda mungkin gagal. Untuk informasi selengkapnya, lihat topik berikut: • Izin untuk bucket S3 kebijakan patch • Masalah: “Memohon-PatchBaselineOperation : Akses Ditolak” kesalahan atau kesalahan “Tidak dapat mengunduh file dari S3" untuk`baseline_overrides.json`	6 Juli 2023

Gunakan Quick SetupOpsCenter untuk mengkonfigurasi manajemen multi-akun OpsItem	Quick Setupuntuk OpsCenter membantu Anda menyelesaikan tugas-tugas berikut untuk mengelola OpsItems seluruh akun: • Menentukan akun administrator yang didelegasikan • Membuat kebijakan dan peran wajib AWS Identity and Access Management (IAM) • Menentukan AWS Organizations organisasi, atau subset akun anggota, tempat administrator yang didelegasikan dapat mengelola seluruh akun OpsItems Untuk informasi selengkapnya, lihat (Opsional) Mengkonfigurasi OpsCenter untuk mengelola OpsItems seluruh akun dengan menggunakan Quick Setup.	19 Juni 2023

Perbarui agen peluncuran Amazon EC2 menggunakan Quick Setup	Anda sekarang dapat mengizinkan Systems Manager untuk memeriksa setiap 30 hari untuk versi baru agen peluncuran yang diinstal pada instans Anda. Jika versi baru tersedia, Systems Manager memperbarui agen pada instans Anda. Untuk informasi selengkapnya, lihat Manajemen Quick Setup Host.	19 Juni 2023
Patch Managersekarang mendukung Ubuntu Server 22,04 LTS	Anda sekarang dapat menggunakan Patch Manager untuk menambal Ubuntu Server 22,04 node LTS. Seperti versi lain yang didukungUbuntu Server, versi 22.04 LTS, menggunakan baseline `AWS-UbuntuDefaultPatchBaseline` patch AWS terkelola.	15 Mei 2023

Systems Manager sekarang mendukung AlmaLinux, termasuk Patch Manager	Anda sekarang dapat menggunakan Systems Manager untuk mengelola AlmaLinux 8.3-8.7; 9.0-9.1 node. Banyak aturan yang berlaku untuk RHEL 8 untuk patching juga berlaku untuk AlmaLinux. AlmaLinux menggunakan yang baru`AWS-DefaultAlmaLinuxPatchBaseline` . Untuk informasi selengkapnya, lihat topik berikut: • Instal secara manual SSM Agent pada AlmaLinux instance • Bagaimana patch keamanan dipilih • Bagaimana tambalan dipasang • Cara kerja aturan dasar patch AlmaLinux,RHEL, dan. Rocky Linux	8 Mei 2023
Terapkan agen EC2launch v2 menggunakan Quick Setup	Anda sekarang dapat menggunakan agen EC2launch v2 menggunakan. Quick Setup Untuk informasi selengkapnya, lihat Menerapkan Distributor paket dengan Quick Setup.	13 April 2023

Systems Manager sekarang mendukung Amazon Linux 2023	Systems Manager sekarang mendukung jenis instans EC2 Amazon Linux 2023 (AL2023) yang baru, termasuk dukungan untuk operasi. Patch Manager Banyak aturan untuk patching yang berlaku untuk Amazon Linux 2 juga berlaku untuk Amazon Linux 2023. (Patch Managerjuga terus mendukung rilis pratinjau Amazon Linux 2022.) Untuk informasi selengkapnya, lihat topik berikut: • Bagaimana patch keamanan dipilih • Bagaimana tambalan dipasang • Cara kerja aturan dasar tambalan di Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022, dan Amazon Linux 2023	Maret 23, 2023
Menyiapkan konten yang direvisi untuk instans Amazon EC2	Kami telah merevisi konten pengaturan untuk instans Amazon EC2. Sekarang disarankan untuk menggunakan Konfigurasi Manajemen Host Default yang baru dirilis untuk izin misalnya. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans untuk Systems Manager.	15 Februari 2023

Manajemen instans otomatis dengan Konfigurasi Manajemen Host Default	Anda sekarang dapat secara otomatis mengelola instans Amazon EC2 secara keseluruhan menggunakan Systems Wilayah AWS Manager. Untuk informasi selengkapnya, lihat Konfigurasi Manajemen Host Default.	15 Februari 2023
Tambahkan dokumen SSM ke favorit Anda	Untuk membantu Anda menemukan dokumen SSM yang sering digunakan, Anda sekarang dapat menambahkan dokumen ke favorit Anda. Anda dapat memfavoritkan hingga 20 dokumen per jenis dokumen, per Akun AWS dan Wilayah AWS. Anda dapat memilih, memodifikasi, dan melihat favorit Anda dari konsol Systems Manager Documents. Untuk informasi selengkapnya, lihat Menambahkan dokumen ke favorit Anda.	7 Februari 2023
Menerapkan kontrol perubahan untuk Otomatisasi menggunakan Change Calendar	Dengan mengintegrasikan Otomasi denganChange Calendar, Anda sekarang dapat menerapkan kontrol perubahan ke semua otomatisasi di Anda. Akun AWS Untuk informasi selengkapnya, lihat Menerapkan kontrol perubahan untuk Otomasi.	Januari 24, 2023

Alur kerja Change Manager persetujuan baru	Alur kerja Change Manager persetujuan sekarang mendukung persetujuan per tingkat, bukan persetujuan per baris. Sebelumnya, setiap pemberi persetujuan yang Anda tambahkan ke tingkat persetujuan harus menyetuju i permintaan perubahan. Kalau tidak, levelnya tidak disetujui. Sekarang, Anda menentukan berapa banyak persetujuan yang diperluka n untuk level dan dapat menambahkan banyak atau lebih pemberi persetujuan. Misalnya, Anda dapat meminta tiga persetujuan untuk satu level tetapi menentukan hingga lima pemberi persetuju an. Persetujuan dari ketiga pemberi persetujuan tersebut cukup untuk menyetujui level tersebut. Untuk informasi selengkapnya, lihat Tentang persetujuan di templat perubahan Anda.	23 Januari 2023

Baru: Konfigurasikan patching untuk seluruh organisasi menggunakan kebijakan patch di Quick Setup	Dengan Quick Setup kemampuan Systems Manager, Anda sekarang dapat membuat kebijakan tambalan yang didukung olehPatch Manager. Kebijakan patch mendefinisikan jadwal dan patch baseline yang akan digunakan saat secara otomatis menambal node terkelola Anda. Dengan menggunakan konfigurasi kebijakan tambalan tunggal, Anda dapat menentukan penambalan untuk semua akun di semua Wilayah di organisasi Anda, hanya untuk akun dan Wilayah yang Anda pilih, atau untuk satu pasangan Account-region. Untuk informasi selengkapnya, lihat topik berikut. • Menggunakan kebijakan Quick Setup tambalan • Mengotomatiskan penambalan di seluruh organisasi menggunakan kebijakan tambalan Quick Setup	22 Desember 2022

Application Managerte rintegrasi dengan Amazon EC2 untuk menampilkan informasi tentang instans Anda dalam konteks aplikasi.	Application Managerme nampilkan status instans, status, dan kesehatan Auto Scaling Amazon EC2 untuk aplikasi yang dipilih dalam format grafis. Tab Instances juga menyertakan tabel dengan informasi berikut untuk setiap instance dalam aplikasi Anda. • Status instans (Tertunda, Berhenti, Berlari, Berhenti) • Status ping untuk SSM Agent • Status dan nama runbook Systems Manager Automation terakhir yang diproses pada instans • Hitungan alarm Amazon CloudWatch Logs per negara bagian. • `ALARM` – Metrik atau ekspresi berada di luar ambang batas yang telah ditetapkan sebelumnya. • `OK` – Metrik atau ekspresi berada dalam ambang batas yang telah ditetapkan sebelumnya. • `INSUFFICIENT_DATA` – Alarm baru saja dimulai, metrik tidak tersedia, atau tidak ada data yang memadai yang	22 Desember 2022

	tersedia bagi metrik untuk menentukan status alarm. • Kesehatan grup Auto Scaling untuk grup penskalaan otomatis induk dan individu	
Jadwalkan awal dan penghentian instans Amazon EC2 Anda menggunakan Quick Setup	Sekarang Anda dapat menerapkan solusi Resource Scheduler untuk mengotomatiskan awal dan penghentian instans Amazon EC2 Anda menggunakan. Quick Setup Untuk informasi selengkapnya, lihat Resource Scheduler.	19 Desember 2022

OpsCentersekarang mendukung bekerja dengan OpsItems lintas akun	OpsCentermendukung bekerja dengan OpsItems dari akun manajemen (baik akun AWS Organizations manajemen atau akun administrator yang didelegasikan Systems Manager) dan akun anggota selama sesi. Setelah dikonfigurasi, pengguna dapat melakukan jenis tindakan berikut: • Buat, lihat, dan perbarui OpsItems di akun anggota • Melihat informasi rinci tentang AWS sumber daya yang ditentukan OpsItems dalam akun anggota • Mulai runbook Automation Systems Manager untuk memulihkan masalah dengan AWS sumber daya di akun anggota Untuk informasi selengkapnya, lihat Menyiapkan OpsCenter untuk bekerja dengan OpsItems seluruh akun.	16 November 2022

Lacak detail permintaan Change Manager perubahan menggunakan AWS CloudTrail Lake	Anda sekarang dapat menggunakan penyimpanan data acara di AWS CloudTrail Lake untuk menangkap dan meninjau detail tentang permintaan perubahan yang dijalankan Change Manager untuk organisasi atau akun Anda. Informasi ini mencakup rincian yang dapat diaudit tentang identitas pengguna yang membuat permintaan perubahan, alamat IP dari mana permintaan dibuat, Wilayah AWS di mana perubahan dibuat, sumber daya yang ditargetkan, dan banyak lagi. Untuk selengkapnya, lihat Memantau peristiwa permintaan perubahan dan Meninjau detail, tugas, dan jadwal permintaan perubahan.	11 November 2022

Kontrol tugas Otomasi Systems Manager tambahan menggunakan CloudWatch alarm	Anda sekarang dapat menerapkan kontrol tambahan saat menjalankan otomatisasi di beberapa akun dan Wilayah dengan menggunakan CloudWatch alarm. Dengan menerapkan CloudWatch alarm metrik atau komposit ke otomatisasi, Anda dapat mengontrol kapan otomatisasi berhenti berdasarkan metrik yang Anda tentukan. Untuk informasi selengkapnya tentang menerapkan CloudWatch alarm ke otomatisasi yang berjalan di beberapa akun dan Wilayah, lihat Menjalankan otomatisasi di beberapa Wilayah dan akun (konsol)	9 November 2022

Diperbarui: 'Menggunakan Parameter Store parameter dalam AWS Lambda fungsi'	Kami telah memberikan informasi tambahan untuk membantu Anda menggunakan Ekstensi Lambda AWS Parameter dan Rahasia untuk mengambil nilai parameter dan menyimpannya untuk digunakan di masa mendatang dalam fungsi Lambda. Menggunakan ekstensi Lambda dapat mengurangi biaya Anda dengan mengurangi jumlah panggilan API ke. Parameter Store Untuk selengkapnya, lihat Menggunakan Parameter Store parameter dalam AWS Lambda fungsi.	25 Oktober 2022

Kontrol tugas Systems Manager tambahan menggunakan CloudWatch alarm	Anda sekarang dapat menerapkan kontrol tambahan saat menjalankan otomatisasi dan perintah dengan menggunakan CloudWatch alarm. CloudWatch Alarm juga dapat ditambahkan ke otomatisasi atau perintah ketika terdaftar dengan tugas jendela State Manager asosiasi atau pemeliharaan. Dengan menerapkan CloudWatch alarm komposit ke otomatisasi atau perintah, Anda dapat mengontrol kapan otomatisasi atau perintah berhenti berdasarkan metrik yang Anda tentukan. Untuk informasi selengkapnya tentang menerapkan CloudWatch alarm ke otomatisasi atau perintah, lihat prosedur berikut: • Bagaimana patch keamanan dipilih • Bagaimana tambalan dipasang • Cara kerja aturan dasar tambalan di Amazon Linux 1, Amazon Linux 2, dan Amazon Linux 2022.	26 September 2022

Kontrol tugas Systems Manager tambahan menggunakan CloudWatch alarm	Anda sekarang dapat menerapkan kontrol tambahan saat menjalankan otomatisasi dan perintah dengan menggunakan CloudWatch alarm. CloudWatch Alarm juga dapat ditambahkan ke otomatisasi atau perintah ketika terdaftar dengan tugas jendela State Manager asosiasi atau pemeliharaan. Dengan menerapkan CloudWatch alarm komposit ke otomatisasi atau perintah, Anda dapat mengontrol kapan otomatisasi atau perintah berhenti berdasarkan metrik yang Anda tentukan. Untuk informasi selengkapnya tentang menerapkan CloudWatch alarm ke otomatisasi atau perintah, lihat prosedur berikut: • Menjalankan otomatisasi sederhana • Menjalankan perintah dari konsol • Buat asosiasi • Tetapkan tugas ke jendela pemeliharaan	26 September 2022

Mengklarifikasi persyaratan tingkat contoh lanjutan	Berdasarkan umpan balik pelanggan, kami telah mengklarifikasi skenario yang mengharuskan Anda mengaktifkan tingkatan instance lanjutan di Mengonfigurasi tingkatan instans.	21 September 2022
Menerapkan CloudWatch Agen Amazon menggunakan Quick Setup	Anda sekarang dapat menggunakan CloudWatch agen Amazon menggunak anQuick Setup. Untuk informasi selengkapnya, lihat Menerapkan Distributor paket dengan Quick Setup.	September 20, 2022

Kunci 'PatchGroup' sekarang didukung untuk grup tambalan ketika metadata instans EC2 diizinkan	Saat Anda mengizinkan tag dalam metadata instans EC2, kunci tag yang Anda buat tidak boleh berisi spasi apa pun. Sebelumnya, ini mencegah pelanggan menambahk an beberapa instans EC2 mereka ke grup tambalan Patch Manager karena kunci tag `Patch Group` harus diterapkan ke instance. Patch Managersekarang mendukung keduanya `Patch Group` (dengan spasi) dan `PatchGroup` (tanpa spasi) sebagai kunci tag untuk mengidentifikasi instance untuk grup tambalan. Contoh EC2 di mana tag diizinkan dalam metadata instance sekarang dapat ditambahkan ke grup tambalan di. Patch Manager Untuk selengkapnya, lihat Tentang grup tambalan.	31 Agustus 2022

Topik baru: “Bagaimana tanggal rilis paket dan tanggal pembaruan dihitung”	Dalam garis dasar tambalan yang dikelola oleh AWS, tambalan baru disetujui secara otomatis 7 hari setelah dirilis atau diperbarui. Dalam baseline patch kustom yang Anda buat, Anda dapat secara opsional menentukan berapa hari untuk menunggu setelah dirilis atau diperbarui untuk menyetujui instalasi secara otomatis. Untuk Amazon Linux 1 dan Amazon Linux 2, berbagai faktor mempengar uhi bagaimana tanggal rilis terbaru dan tanggal pembaruan dihitung. Untuk membantu Anda menghinda ri hasil yang tidak terduga saat memilih penundaan persetujuan otomatis, faktor-faktor ini dijelaskan dalam topik Bagaimana tanggal rilis paket dan tanggal pembaruan dihitung.	Agustus 24, 2022
Konten yang diperbaru i: Menambal AMI dan memperbarui grup Auto Scaling	Kami telah memperbarui panduan grup Pembaruan AMIs untuk Auto Scaling untuk menggunakan templat peluncuran alih-alih konfigura si peluncuran. Selain itu, kami telah menerapkan tindakan dan runtime Otomasi terbaru di konten runbook.	Juni 22, 2022

Change Manager: Mencegah pengguna membuat permintaan yang dapat disetujui secara otomatis	Anda dapat mengonfigurasi templat perubahan Change Manager untuk mendukung persetujuan otomatis, yang berarti bahwa pengguna dengan izin IAM yang diperlukan dapat memilih untuk memulai permintaan perubahan tanpa memerlukan persetujuan tambahan. Sekarang, Anda juga dapat membatasi pengguna individu, grup, atau peran IAM agar tidak mengirimkan permintaan persetujuan otomatis, meskipun templat perubahan mendukungnya. Ini dicapai melalui penggunaan kunci kondisi IAM baru,`ssm:AutoApprove` . Untuk informasi selengkapnya, lihat Mengontrol akses ke alur kerja buku runbook persetujuan otomatis	15 Juni 2022

Panduan yang diperbarui untuk peran tugas jendela pemeliharaan	Sebelumnya, konsol Systems Manager memberi Anda kemampuan untuk memilih peran tertaut layanan IAM AWS terkelola yang akan digunakan sebagai peran `AWSServiceRoleForAmazonSSM` pemeliharaan untuk tugas Anda. Menggunakan peran ini dan kebijakan terkaitnya`AmazonSSM ServiceRolePolicy` ,, untuk tugas jendela pemeliharaan tidak lagi disarankan. Sebagai gantinya, Anda harus membuat kebijakan dan peran khusus untuk tugas jendela pemeliharaan. Untuk informasi selengkapnya, lihat Menyiapkan Maintenance Windows.	9 Juni 2022
Port forwarding ke dukungan host jarak jauh untuk Session Manager	Session Managersekarang mendukung sesi penerusan port ke host jarak jauh. Host jarak jauh tidak perlu dikelola oleh Systems Manager. Untuk informasi selengkapnya, lihat Memulai sesi (penerusan port ke host jarak jauh).	25 Mei 2022

Konten yang diperbarui: Petunjuk untuk menginstal secara manual SSM Agent pada instans Amazon EC2 Linux	Menanggapi umpan balik pelanggan, kami telah merombak topik yang memberikan instruksi untuk menginstal secara manual di instans Amazon SSM Agent EC2. Topik-topik ini sekarang menyediakan perintah menggunakan file yang tersedia secara global yang dapat Anda salin dan tempel untuk instalasi cepat pada instans EC2 di mana pun. Wilayah AWS Topik-topik ini juga memberikan informasi untuk membantu Anda membuat perintah instalasi yang menggunakan file yang tersedia di Wilayah kerja Anda sendiri. Pendekatan terakhir direkomendasikan ketika Anda menginstal agen pada beberapa contoh menggunakan skrip atau template. Untuk informasi selengkapnya, lihat petunjuk untuk sistem operasi Linux Anda di bagian Menginstal secara manual SSM Agent pada instans EC2 untuk Linux.	9 Mei 2022

Topik baru: Amazon Machine Images (AMIs) dengan SSM Agent prainstal	Menanggapi umpan balik pelanggan, kami memiliki informasi terpusat tentang yang AWS dikelola AMIs termasuk yang sudah diinstal SSM Agent sebelumnya. Topik ini juga memberikan petunjuk tentang cara memverifikasi bahwa instans Amazon EC2 yang dibuat dari instans ini berhasil diinstal dan AMIs sedang berjalan. Untuk kasus yang jarang terjadi di mana agen mungkin tidak berhasil menginstal, atau menginstal tetapi tidak memulai, kami juga memberikan informasi tentang memulai atau menginstal agen secara manual pada instance ini. Untuk detailnya , lihat Amazon Machine Images(AMIs) dengan SSM Agent prainstal.	8 Mei 2022
State ManagerBagian baru	Menambahkan bagian baru yang menjelaskan rincian kapan State Manager menjalankan asosiasi. Untuk informasi selengkapnya, lihat Tentang penjadwalan asosiasi.	27 April 2022

Patch Managersekarang mendukung Rocky Linux	Anda sekarang dapat menggunakan Patch Manager untuk menambal Rocky Linux node. Banyak aturan yang berlaku untuk RHEL 8 untuk patching juga berlaku untukRocky Linux. Rocky Linux8 menggunakan yang baru`AWS-DefaultRockyLinuxPatchBaseline` . Untuk informasi selengkapnya, lihat topik berikut: • Bagaimana patch keamanan dipilih • Bagaimana tambalan dipasang • Cara kerja aturan dasar patchRHEL,CentOS Stream, dan. Rocky Linux	April 14, 2022

Patch Managersekarang mendukung CentOS Stream 8	Anda sekarang dapat menggunakan Patch Manager untuk menambal CentOS Stream 8 instance dan Red Hat Enterprise Linux (RHEL) 4.4-4.5 instance. Banyak aturan yang berlaku untuk RHEL 8 untuk patching juga berlaku CentOS Stream 8. CentOS Stream8 menggunak an`AWS-DefaultCentOSP atchBaseline` . Untuk informasi selengkapnya, lihat topik berikut: • Bagaimana patch keamanan dipilih • Bagaimana tambalan dipasang • Cara kerja aturan dasar tambalan dan RHELCentOS Stream	4 April 2022

Buat peran asumsi untuk Change Manager	Bagian baru mengklarifikasi persyaratan untuk membuat dan mengimplementasikan peran asumsi untukChange Manager. Peran asumsi adalah peran layanan AWS Identity and Access Management (IAM) yang memungkinkan Change Manager untuk menjalankan alur kerja buku runbook yang ditentukan dengan aman dalam permintaan perubahan yang disetujui atas nama Anda. Peran tersebut memberikan AWS Systems Manager (AWS STS) `AssumeRole` kepercayaan kepadaChange Manager. Untuk selengkapnya, lihat Mengonfigurasi peran dan izin untuk. Change Manager	18 Maret 2022
Menyetujui atau menolak permintaan Change Manager perubahan secara massal	Di konsol Systems Manager, Anda sekarang dapat memilih beberapa permintaan perubahan untuk menyetujui atau menolak dalam satu operasi. Untuk selengkapnya, lihat Meninjau dan menyetujui atau menolak permintaan perubahan (konsol).	8 Maret 2022

Support untuk Rocky Linux dan node terkelola Windows Server 2022	Systems Manager mendukung Rocky Linux dan node terkelola Windows Server 2022, termasuk perangkat edge dan mesin hybrid yang berlokasi di lokasi atau dengan penyedia cloud lainnya. Untuk menggunakan Systems Manager dengan sistem operasi ini, Anda harus menyelesaikan semua prosedur penyiapan Systems Manager yang diperlukan, termasuk prosedur untuk lingkungan hybrid atau perangkat edge, jika berlaku. Untuk informasi selengkapnya, lihat Menyiapkan Systems Manager. Untuk Rocky Linux mesin, Anda juga harus menginstal secara manualSSM Agent. Untuk informasi selengkapnya, lihat Instal secara manual SSM Agent pada Rocky Linux instance. Untuk SSM Agent instans Amazon Elastic Compute Cloud (Amazon EC2) Windows Server 2022, diinstal secara default.	1 Maret 2022

Izinkan Otomasi beradaptasi dengan kebutuhan konkurensi Anda dan lihat metrik penggunaan Otomasi	Sekarang Anda dapat mengizinkan Otomasi untuk secara otomatis menyesuaikan kuota otomatisasi bersamaan, dan melihat metrik penggunaan Otomasi yang dipublikasikan. CloudWatch Untuk informasi selengkapnya tentang konkurensi adaptif, lihat Mengizinkan Otomasi beradaptasi dengan kebutuhan konkurensi Anda. Untuk informasi selengkapnya tentang cara melihat metrik penggunaan Otomasi, lihat Metrik Pemantauan Otomasi menggunakan Amazon. CloudWatch	27 Januari 2022

Izinkan Otomasi beradaptasi dengan kebutuhan konkurensi Anda dan lihat metrik penggunaan Otomasi	Sekarang Anda dapat mengizinkan Otomasi untuk secara otomatis menyesuaikan kuota otomatisasi bersamaan, dan melihat metrik penggunaan Otomasi yang dipublikasikan. CloudWatch Untuk informasi selengkapnya tentang konkurensi adaptif, lihat Mengizinkan Otomasi beradaptasi dengan kebutuhan konkurensi Anda. Untuk informasi selengkapnya tentang cara melihat metrik penggunaan Otomasi, lihat Metrik Pemantauan Otomasi menggunakan Amazon. CloudWatch	27 Januari 2022
Dokumen Systems Manager diatur berdasarkan kategori	Dokumen Systems Manager milik Amazon sekarang diatur berdasarkan jenis dan kategori untuk membantu Anda menemukan dokumen yang Anda butuhkan.	Januari 13, 2022

Membuat dan memanggil integrasi untuk Otomasi	Anda sekarang dapat mengirim pesan menggunakan webhook selama otomatisasi dengan membuat integrasi. Integrasi dapat dipanggil selama otomatisasi menggunakan `aws:invokeWebhook` tindakan baru di runbook Anda. Untuk informasi selengkapnya tentang membuat integrasi, lihat Membuat integrasi webhook untuk Otomasi. Untuk mempelajari lebih lanjut tentang `aws:invokeWebhook` tindakan tersebut, lihat `aws:invokeWebhook` — Memanggil integrasi webhook Otomasi.	Januari 13, 2022
Kemampuan tidak tersedia di baru Wilayah AWS	Kemampuan Systems Manager berikut saat ini tidak tersedia di Wilayah Asia Pasifik (Jakarta) yang baru. • Application Manager • Change Calendar • Change Manager • Explorer • Fleet Manager • Incident Manager • Quick Setup	13 Desember 2021

Lihat detail biaya sumber daya untuk aplikasi	Application Managerte rintegrasi dengan AWS Billing and Cost Management melalui widget Cost Explorer. Setelah Anda mengaktifkan Cost Explorer di konsol Billing and Cost Management, widget Application Manager Cost Explorer akan menampilkan data biaya untuk aplikasi non-container atau komponen aplikasi tertentu. Anda dapat menggunakan filter di widget untuk melihat data biaya sesuai dengan periode waktu, perincian, dan jenis biaya yang berbeda baik dalam bagan batang atau garis. Untuk informasi selengkapnya, lihat Melihat informasi ikhtisar tentang aplikasi.	Desember 7, 2021
Mengelola proses menggunakan Fleet Manager	Anda sekarang dapat menggunakan Fleet Manager untuk mengelola proses pada node Anda. Untuk informasi selengkapnya, lihat Bekerja dengan proses.	Desember 6, 2021

Perubahan terminologi: instance terkelola sekarang menjadi node yang dikelola	Dengan dukungan untuk perangkat AWS IoT Greengrass inti, frasa instance terkelola telah diubah menjadi node terkelola di sebagian besar dokumentasi Systems Manager. Konsol Systems Manager, panggilan API, pesan kesalahan, dan dokumen SSM masih menggunakan istilah instance.	29 November 2021

Support untuk perangkat edge	Systems Manager mendukung konfigurasi perangkat edge berikut. • AWS IoT Greengras sSystems Manager sekarang mendukung perangkat apa pun yang dikonfigurasi untuk AWS IoT Greengrass dan menjalank an perangkat lunak AWS IoT Greengrass Core. Untuk onboard perangkat AWS IoT Greengrass inti Anda, Anda harus membuat peran layanan AWS Identity and Access Management (IAM). Anda juga harus menggunak an AWS IoT Greengrass konsol untuk digunakan SSM Agent sebagai AWS IoT Greengrass komponen di perangkat Anda. Untuk informasi selengkapnya, lihat AWS Systems Manager Menyiapkan perangkat edge. • Perangkat Edge dalam lingkungan hybrid: Systems Manager juga mendukung perangkat AWS IoT Core dan perangkat AWS non-IoT setelah Anda mengonfig urasinya sebagai mesin lokal. Untuk onboard	29 November 2021

	perangkat Anda, Anda harus membuat peran layanan IAM, membuat aktivasi node terkelola untuk lingkunga n hybrid, dan menginstal SSM Agent secara manual di perangkat Anda. Untuk informasi selengkapnya, lihat AWS Systems Manager Menyiapkan lingkungan hibrida	
Connect ke instans terkelola menggunakan Remote Desktop	Anda sekarang dapat menggunakan Fleet Manager untuk terhubung ke instance Windows yang dikelola menggunakan Remote Desktop Protocol (RDP). Sesi Remote Desktop ini didukung oleh NICE DCV menyediakan koneksi aman ke instans Anda langsung dari browser Anda. Untuk informasi selengkapnya, lihat Connect menggunakan Remote Desktop.	23 November 2021

Tentukan durasi sesi maksimum dan berikan alasan untuk sesi	Anda sekarang dapat menentukan durasi sesi maksimum untuk semua Session Manager sesi di Wilayah AWS dalam sesi Anda Akun AWS. Ketika sesi mencapai mencapai durasi yang Anda tentukan, sesi tersebut akan dihentikan. Anda sekarang juga dapat menambahkan alasan secara opsional saat memulai sesi. Untuk informasi selengkapnya, lihat Menentukan durasi sesi maksimum.	November 16, 2021
Patch Managersekarang mendukung sistem Raspberry Pi OS operasi	Anda sekarang dapat menggunakan Patch Manager untuk menambal Raspberry Pi OS instance. Patch Managermendukung patching Raspberry Pi OS 9 (Stretch) dan 10 (Buster). Karena OS berbasis Debian, banyak aturan patching yang sama berlaku untuk itu. Raspberry Pi OS Debian Server Untuk informasi selengkapnya, lihat topik berikut: • Cara pemilihan patch keamanan • Cara menginstal patch • Cara kerja aturan dasar tambalan dan Debian ServerRaspberry Pi OS	November 16, 2021

Akses portal Red Hat Knowledgebase	Gunakan Fleet Manager untuk mengakses portal RHEL Knowledgebase untuk menemukan solusi, artikel, dokumentasi, dan video tentang penggunaan produk Red Hat. Untuk informasi lebih lanjut, lihat Mengakses portal Red Hat Knowledgebase.	3 November 2021
Suntingan massal OpsItems	OpsCentersekarang mendukung pengeditan massalOpsItems. Anda dapat memilih beberapa OpsItems dan mengedit salah satu bidang berikut: Status, Prioritas, Keparahan, Kategori. Untuk informasi selengkapnya, lihat Mengedit OpsItems.	Oktober 15, 2021
Buat parameter masukan yang mengisi sumber daya AWS	Anda sekarang dapat membuat parameter input di runbook Otomasi yang mengisi AWS sumber daya di file. AWS Management Console Untuk selengkapnya, lihat Membuat parameter input yang mengisi AWS sumber daya.	Oktober 14, 2021

Opsi cutoff pemanggilan tugas baru untuk jendela pemeliharaan	Anda sekarang dapat memilih untuk memblokir pemanggilan tugas baru dari memulai setelah batas waktu yang ditentukan untuk jendela pemeliharaan tercapai. Untuk selengkapnya, lihat Menetapkan tugas ke jendela pemeliharaan (konsol).	13 Oktober 2021
Patch Managerdukungan untuk macOS 11.3.1 dan 11.4 (Big Sur)	Instans Amazon Elastic Compute Cloud (Amazon EC2) macOS untuk 11.3.1 dan 11.4 (Big Sur) sekarang dapat ditambal menggunakan. Patch Manager Ini merupakan tambahan dukungan yang ada untuk macOS 10.14.x (Mojave) dan 10.15.x (Catalina). Untuk informasi tentang bekerja denganPatch Manager, lihat AWS Systems ManagerPatch Manager.	1 Oktober 2021

Wawasan aplikasi di Application Manager	Application Managerterintegrasi dengan Amazon CloudWatch Application Insights. Application Insights mengidentifikasi dan menyiapkan metrik utama, log, dan alarm di seluruh sumber daya aplikasi dan tumpukan teknologi Anda. Application Insights terus memantau metrik dan log untuk mendeteksi dan mengkorelasikan anomali dan kesalahan. Ketika sistem mendeteksi kesalahan atau anomali, Application Insights menghasilkan CloudWatch Peristiwa yang dapat Anda gunakan untuk mengatur notifikasi atau mengambil tindakan. Anda dapat mengaktifkan dan melihat Wawasan Aplikasi pada tab Ikhtisar dan Pemantauan di. Application Manager Untuk informasi selengkapnya tentang Wawasan Aplikasi, lihat Apa itu Wawasan CloudWatch Aplikasi Amazon di CloudWatch Panduan Pengguna Amazon.	September 21, 2021

Impor acara dari kalender lain ke Change Calendar	Anda sekarang dapat mengimpor acara dari kalender pihak ketiga ke kalender diChange Calendar. Sebelumnya, setiap acara harus dimasukkan secara manual ke dalam kalender. Setelah Anda mengekspo r kalender dari penyedia kalender pihak ketiga yang didukung ke file iCalendar `.ics` (), impor Change Calendar ke dalam, dan acaranya disertakan dalam aturan untuk kalender terbuka atau tertutup di Systems Manager. Penyedia yang didukung termasuk Kalender iCloud, Kalender Google, dan Microsoft Outlook. Untuk informasi selengkapnya, lihat Mengimpor dan mengelola acara dari kalender pihak ketiga.	8 September 2021

Fitur penandaan dan runbook baru di Application Manager	Peningkatan penandaan mencakup kemampuan untuk menambahkan tag ke atau menghapus tag dari sumber daya tertentu atau semua sumber daya dalam aplikasi. Application Manager Penyempurnaan Runbook mencakup kemampuan untuk melihat daftar runbook yang difilter untuk jenis sumber daya tertentu atau memulai runbook pada semua sumber daya dari jenis yang sama. Untuk informasi selengkap nya, lihat Bekerja dengan tag di Application Manager dan Bekerja dengan runbook di Application Manager.	31 Agustus 2021
Contoh baru: Buat permintaa n perubahan menggunakan AWS CLI	Contoh membuat permintaan perubahan dengan AWS CLI telah ditambahkan ke Change Manager chapter. Contoh menggunakan template `AWS-HelloWorldChan geTemplate` perubahan sampel dan`AWS-Hello World runbook` : • Membuat permintaan perubahan (AWS CLI)	Agustus 20, 2021

Bagian baru: Gunakan parameter di Amazon EKS	Bagian baru telah ditambahkan ke bagian iniParameter Store. Topik ini adalah panduan tentang cara menggunakan parameter Anda di kluster Amazon EKS. Untuk informasi selengkapnya, lihat Menggunakan Parameter Store parameter di Amazon Elastic Kubernetes Service.	19 Agustus 2021
Kait Patch Manager siklus hidup yang diperbarui	Patch Managersekarang menyediakan kait siklus hidup — kemampuan untuk menjalankan dokumen Systems Manager Command — untuk titik tambahan selama operasi Patch now patching. Jika Anda menjadwalkan reboot instance setelah menjalankan Patch sekarang, Anda dapat menentukan hook siklus hidup untuk dijalankan setelah reboot selesai. Untuk informasi selengkapnya, lihat Menggunakan kait siklus hidup 'Patch now' dan Tentang dokumen SSM. `AWS-RunPatchBaselineWithHooks`	9 Agustus 2021

Persetujuan otomatis sekarang didukung untuk permintaan Change Manager	Sekarang Anda dapat mengonfigurasi templat perubahan Change Manager untuk mendukung persetujuan otomatis, yang berarti bahwa pengguna dengan izin IAM yang diperlukan dapat memilih untuk memulai permintaan perubahan tanpa memerlukan persetujuan tambahan. Pengguna yang memiliki akses ke templat persetujuan otomatis masih dapat memilih untuk menentukan pemberi persetujuan jika mereka memilih. Untuk membantu Anda mengontrol Change Manager proses, persetujuan masih diperlukan untuk semua permintaan selama periode pembekuan perubahan. Untuk informasi selengkapnya, lihat topik berikut: • Membuat template perubahan • Membuat permintaan perubahan • Coba template `Hello World` perubahan AWS terkelola	30 Juli 2021

OpsCenterwawasan operasional	OpsCentersecara otomatis menganalisis OpsItems di akun Anda dan menghasilkan wawasan. Wawasan mencakup informasi untuk membantu Anda memahami berapa banyak duplikat OpsItems di akun Anda dan sumber mana yang membuatnya. Wawasan juga menyediakan praktik terbaik yang direkomendasikan dan runbook Otomasi untuk membantu Anda menyelesaikan duplikat. OpsItems Untuk informasi selengkapnya, lihat Bekerja dengan wawasan operasional.	13 Juli 2021
Lihat instance yang dihentikan di Fleet Manager	Anda sekarang dapat melihat instance mana `running` dan instance mana yang `stopped` berasal dari konsol. Fleet Manager Untuk informasi selengkapnya, lihat AWS Systems ManagerFleet Manager.	12 Juli 2021
Topik baru: Menulis runbook Otomasi	Topik baru, Authoring Automation runbook, memberikan panduan dan contoh naratif tentang cara membuat konten untuk runbook Otomasi kustom.	8 Juli 2021

AWS CloudFormation tumpukan dan pembuatan template di Application Manager	Application Managermembantu Anda menyediakan dan mengelola sumber daya untuk aplikasi Anda dengan mengintegrasikan dengan CloudFormation. Anda dapat membuat, mengedit, dan menghapus AWS CloudFormation template dan tumpukan diApplication Manager. Application Managerjuga mencakup pustaka templat tempat Anda dapat mengkloning, membuat, dan menyimpan templat. Application Managerdan CloudFormation menampilkan informasi yang sama tentang status tumpukan saat ini. Template dan pembaruan template disimpan di Systems Manager hingga Anda menyediakan tumpukan, pada saat itu perubahan juga ditampilkan CloudFormation. Untuk informasi selengkapnya, lihat Bekerja dengan AWS CloudFormation Tumpukan di Application Manager.	8 Juli 2021

Topik baru: Putar kunci pribadi secara otomatis untuk SSM Agent instance hybrid	Topik baru, Menyiapkan rotasi otomatis kunci pribadi, memberikan instruksi tentang cara memperkuat postur keamanan Anda dengan mengonfigurasi SSM Agent untuk memutar kunci pribadi lingkungan hibrida secara otomatis.	15 Juni 2021
Session Managerplugin untuk AWS CLI versi 1.2.205.0	Versi baru dari Session Manager plugin untuk AWS CLI telah dirilis. Untuk informasi selengkapnya, lihat Session Managerplugin versi terbaru dan riwayat rilis.	10 Juni 2021
Peran terkait layanan IAM baru	Saat Anda mengaktifkan wawasan OpsCenter operasional, Systems Manager membuat peran terkait layanan AWS Identity and Access Management (IAM) baru yang disebut. `AWSSSMOpsInsightsServiceRolePolicy` Untuk informasi selengkapnya tentang peran ini, lihat Menggunakan peran untuk membuat wawasan operasional OpsItems di Systems ManagerOpsCenter: `AWSSSMOpsInsightsServiceRolePolicy`.	9 Juni 2021

Konten Patch Manager pemecahan masalah baru untuk Linux	Topik baru, Kesalahan saat menjalankan `AWS-RunPatchBaseline` pada Linux, menyediakan deskripsi dan solusi untuk beberapa masalah yang mungkin ditemui saat patching instans terkelola dengan sistem operasi Linux.	8 Juni 2021
Peningkatan dukungan untuk tugas jendela pemeliharaan yang tidak memerlukan target tertentu (konsol)	Anda sekarang dapat membuat tugas jendela pemeliharaan di konsol tanpa harus menentukan target dalam tugas jika tidak diperlukan. Sebelumnya, opsi ini hanya tersedia saat menggunakan API AWS CLI atau. Opsi ini berlaku untuk Otomasi, AWS Lambda, dan jenis AWS Step Functions tugas. Sebagai contoh, jika Anda membuat tugas Otomatisasi dan sumber daya untuk diperbarui ditentukan dalam parameter dokumen Otomatisasi, Anda tidak perlu lagi menentukan target dalam tugas itu sendiri. Untuk informasi selengkapnya, lihat Mendaftarkan tugas jendela pemeliharaan tanpa target, Menetapkan tugas ke jendela pemeliharaan (konsol), dan Menjadwalkan otomatisasi dengan jendela pemeliharaan.	28 Mei 2021

Referensi buku runbook otomatisasi dipindahkan	Referensi runbook otomatisasi telah dipindahkan ke lokasi baru. Untuk informasi lebih lanjut, lihat Referensi runbook Otomatisasi Systems Manager.	10 Mei 2021
AWS Systems Manager Incident Manager peluncuran	Incident Manager adalah konsol manajemen insiden yang dirancang untuk membantu pengguna mengurangi dan memulihkan dari insiden yang memengaruhi aplikasi yang dihosting mereka AWS . Untuk informasi selengkapnya, silakan lihat Panduan Pengguna AWS Systems Manager Incident Manager.	10 Mei 2021
State Managermendukung Change Calendar	Sekarang Anda dapat menentukan Change Calendar nama atau Nama Sumber Daya Amazon (ARN) saat membuat atau memperbarui State Manager asosiasi. State Managermenerapkan asosiasi hanya ketika kalender perubahan terbuka, bukan saat ditutup. Untuk informasi selengkapnya, lihat Membuat asosiasi dan Mengedit dan membuat versi baru asosiasi.	6 Mei 2021

Dokumen Clone Systems Manager	Menggunakan konsol Dokumen Systems Manager, Anda sekarang dapat menyalin konten dari dokumen yang ada ke dokumen baru yang dapat Anda modifikasi. Untuk mempelajari selengkap nya, lihat Mengkloning dokumen SSM.	4 Mei 2021
Integrasikan Security Hub dengan Explorer dan OpsCenter	Anda sekarang dapat mengintegrasikan Explorer dan OpsCenter dengan AWS Security Hub. Security Hub memberikan pandangan komprehensif tentang status keamanan Anda AWS dan membantu Anda memeriksa lingkungan Anda terhadap standar industri keamanan dan praktik terbaik. Saat terintegr asiExplorer, Anda dapat melihat temuan keamanan di widget Security Hub di Explorer dasbor. Ketika terintegrasi denganOps Center, Anda dapat membuat OpsItems untuk temuan Security Hub. Untuk informasi lebih lanjut, lihat Menerima temuan dari AWS Security Hub dalam Explorer dan Menerima temuan dari AWS Security Hub dalam OpsCenter.	27 April 2021

Topik baru: Konvensi dokumen	Kami telah menambahkan topik baru untuk membantu pengguna memahami konvensi tipografi umum untuk Panduan Pengguna AWS Systems Manager . Untuk informasi selengkapnya, lihat Konvensi dokumen.	21 April 2021
Topik yang diperbarui: Tentang menambal aplikasi yang dirilis oleh Microsoft pada Windows Server	Topik Tentang menambal aplikasi yang dirilis oleh Microsoft Windows Server sekarang menjelaskan bahwa, agar dapat Patch Manager menambal aplikasi yang dirilis oleh Microsoft pada instance Windows Server terkelola Anda, opsi pembaruan Windows Beri saya pembaruan untuk produk Microsoft lainnya ketika saya memperbarui Windows harus diizinkan pada instance.	12 April 2021
Reorganisasi referensi buku runbook otomatisasi	Untuk membantu Anda menemukan runbook yang Anda butuhkan dan menavigasi referensi dengan lebih efisien, kami mengatur ulang konten dalam referensi runbook Otomasi menurut yang relevan. Layanan AWS Untuk melihat perubahan ini, lihat Referensi runbook Otomatisasi Systems Manager.	12 April 2021

Patch Manager: Hasilkan laporan kepatuhan tambalan .csv	Patch Managersekarang mendukung kemampuan untuk menghasilkan laporan kepatuhan tambalan untuk instans Anda dan menyimpan laporan dalam bucket S3 pilihan Anda, dalam format.csv. Kemudian, menggunakan alat seperti Amazon QuickSight, Anda dapat menganalisis data laporan kepatuhan patch. Anda dapat membuat laporan kepatuhan patch untuk instans tunggal, atau untuk semua instans di Akun AWS Anda. Anda dapat membuat laporan satu kali sesuai permintaan, atau mengatur jadwal agar laporan dibuat secara otomatis. Anda juga dapat menentukan topik Amazon Simple Notification Service untuk memberikan notifikasi ketika laporan dibuat. Untuk informasi selengkapnya, lihat Membuat laporan kepatuhan patch CSV.	9 April 2021
Hapus label Parameter Store parameter	Anda sekarang dapat menghapus label Parameter Store parameter dengan menggunakan konsol Systems Manager atau AWS CLI. Untuk informasi lebih lanjut, lihat Bekerja dengan label parameter.	6 April 2021

Jadwalkan reboot instance saat menggunakan Patch Now	Patch Managersekarang mendukung penjadwal an waktu untuk instance Anda untuk reboot setelah patch diinstal menggunakan fitur Patch Now. Ini adalah tambahan untuk opsi yang ada untuk me-reboot instans hanya jika diperlukan untuk menyelesaikan instalasi patch atau untuk melompati semua proses reboot setelah operasi patching. Untuk informasi, lihat Patching instans sesuai permintaan.	1 April 2021
Topik baru: Temukan parameter publik	Parameter Storeparameter publik sekarang dapat ditemukan menggunak an konsol AWS CLI atau Systems Manager. Untuk informasi selengkapnya, lihat Menemukan parameter publik.	1 April 2021

Patch sekarang diperbarui: Simpan log di S3 & dan jalankan kait siklus hidup	Saat menjalankan operasi Patch Manager Patch now, Anda dapat memilih bucket S3 untuk menyimpan log patching secara otomatis. Selain itu, Anda dapat memilih untuk menjalankan dokumen Perintah Systems Manager (dokumen SSM) sebagai kait siklus hidup pada tiga titik selama operasi: Sebelum instalasi, Setelah instalasi, dan Saat keluar. Untuk informasi lebih lanjut, lihat Patching instans sesuai permintaan.	31 Maret 2021
Systems Manager sekarang melaporkan perubahan pada kebijakan yang AWS dikelola	Mulai 24 Maret 2021, perubahan kebijakan terkelola dilaporkan dalam Systems Managerpembaruan topik kebijakan AWS terkelola. Perubahan pertama yang tercantum adalah penambahan dukungan untuk Explorer kemampuan melaporkan OpsData dan OpsItems dari beberapa akun dan Wilayah.	24 Maret 2021

Explorersecara otomatis memungkinkan semua OpsData sumber untuk sinkronisasi data sumber daya berdasarkan akun di AWS Organizations	Saat Anda membuat sinkronisasi data sumber daya, jika Anda memilih salah satu AWS Organizations opsi, Systems Manager secara otomatis mengizinkan semua OpsData sumber yang dipilih Wilayah AWS untuk semua Akun AWS di organisasi Anda (atau di unit organisasi yang dipilih). Ini berarti, misalnya, bahwa meskipun Anda belum mengizinkan Explorer Wilayah AWS, jika Anda memilih AWS Organizations opsi untuk sinkronisasi data sumber daya Anda, maka Systems Manager secara otomatis mengumpulkan OpsData dari Wilayah tersebut. Untuk informasi selengkapnya, lihat Tentang sinkronisasi data sumber daya beberapa akun dan Region.	24 Maret 2021
Systems Manager Automation menyediakan variabel sistem baru untuk runbook Anda	Dengan variabel `global:AWS_PARTITION` sistem baru, Anda dapat menentukan AWS partisi tempat sumber daya berada saat membuat runbook Anda. Untuk informasi selengkapnya, lihat Variabel sistem Otomatisasi.	18 Maret 2021

Izinkan beberapa tingkat persetujuan untuk permintaan Change Manager perubahan	Saat Anda membuat templat Change Manager perubahan, Anda sekarang dapat meminta lebih dari satu tingkat pemberi persetujuan memberikan izin agar permintaan perubahan dijalankan. Misalnya, Anda mungkin memerlukan peninjau teknis untuk menyetujui permintaan perubahan yang dibuat dari templat perubahan terlebih dahulu, dan kemudian memerlukan persetujuan tingkat kedua dari satu pengelola atau lebih. Untuk informasi lebih lanjut, lihat Membuat templat perubahan.	4 Maret 2021
Patch Managersekarang mendukung Oracle Linux 8.x	Anda sekarang dapat menggunakan Patch Manager untuk menambal instance Oracle Linux 8.x, melalui versi 8.3. Untuk informasi selengkapnya, lihat topik berikut: • Bagaimana patch keamanan dipilih • Bagaimana tambalan dipasang • Cara kerja aturan dasar tambalan Oracle Linux	1 Maret 2021

OpsCentermenampilkan lainnya OpsItems untuk sumber daya yang dipilih	Untuk membantu Anda menyelidiki masalah dan menyediakan konteks untuk masalah, Anda dapat melihat daftar OpsItems AWS sumber daya tertentu. Daftar ini menampilkan status, tingkat keparahan, dan judul masing-masingOpsItem. Daftar ini juga mencakup tautan dalam ke masing-masingOpsItem. Untuk informasi selengkapnya, lihat Melihat sumber lain OpsItems untuk sumber daya tertentu.	1 Maret 2021
Tentukan preferensi tambalan saat runtime	Anda sekarang dapat menentukan preferensi patching pada saat waktu aktif menggunakan fitur baseline override. Untuk selengkapnya, lihat Menggunakan BaselineOverride parameter.	25 Februari 2021

Jenis dokumen Systems Manager baru	AWS CloudFormation template sekarang dapat disimpan sebagai dokumen Systems Manager. Menyimpan CloudFormation template sebagai dokumen Systems Manager memungkinkan Anda memanfaatkan fitur dokumen Systems Manager seperti pembuatan versi, membandingkan konten versi, dan berbagi dengan akun. Untuk informasi selengkapnya, lihat Dokumen AWS Systems Manager.	9 Februari 2021

Contoh tambalan menggunakan kait opsional	Dokumen SSM yang baru `AWS-RunPatchBaselineWithHooks` menyediakan kait yang dapat Anda gunakan untuk menjalankan dokumen SSM pada tiga titik selama siklus patching instans. Untuk informasi tentang `AWS-RunPatchBaselineWithHooks`, lihat Tentang dokumen SSM `AWS-RunPatchBaselineWithHooks`. Untuk contoh panduan operasi patching yang menggunakan seluruh tiga kait, lihat Panduan: Memperbarui dependensi aplikasi, mem-patch instans, dan melakukan pemeriksaan kesehatan khusus aplikasi.	2 Februari 2021

Topik baru: Memvalidasi server lokal dan mesin virtual menggunakan sidik jari perangkat keras	SSM Agentmemverifikasi identifikasi server lokal dan mesin virtual dan VM yang Anda daftarkan dengan layanan menggunakan sidik jari yang dihitung. Sidik jari adalah string buram, disimpan dalam Vault yang diteruskan agen ke API Systems Manager tertentu. Untuk informasi tentang sidik jari perangkat keras dan petunjuk untuk mengkonfi gurasi ambang kesamaan untuk membantu verifikas i mesin, lihat Memvalidasi server dan mesin virtual on-premise menggunakan sidik jari perangkat keras.	25 Januari 2021
Topik baru: referensi SSM Agent teknis	Referensi SSM Agent teknis topik menyatukan informasi untuk membantu Anda menerapkan AWS Systems Manager SSM Agent dan memahami cara kerja agen. Topik ini mencakup bagian yang semuanya baru, pembaruan SSM Agent bergulir oleh Wilayah AWS.	21 Januari 2021

SSM Agentpada Windows Server 2008	Per 14 Januari 2020, Windows Server 2008 tidak lagi didukung untuk pembaruan fitur atau keamanan dari Microsoft. Windows Server2008 AMIs memang termasukSSM Agent, tetapi agen tidak lagi diperbarui untuk sistem operasi ini.	5 Januari 2021
Peningkatan dukungan untuk tugas jendela pemeliharaan yang tidak memerlukan target tertentu (AWS CLI dan hanya API)	Anda sekarang dapat membuat tugas jendela pemeliharaan tanpa harus menentukan target dalam tugas jika tidak diperlukan (AWS CLI dan hanya API). Ini berlaku untuk Otomasi, AWS Lambda dan jenis AWS Step Functions tugas. Sebagai contoh, jika Anda membuat tugas Otomatisasi dan sumber daya untuk diperbaru i ditentukan dalam parameter runbook Otomatisasi, Anda tidak perlu lagi menentukan target dalam tugas itu sendiri. Untuk informasi selengkap nya, lihat Mendaftarkan tugas jendela pemeliharaan tanpa target dan Menjadwalkan otomatisasi dengan jendela pemeliharaan.	23 Desember 2020

Fitur Otomasi Baru	Properti bersama baru telah ditambahkan ke runbook Otomatisasi Systems Manager. Properti `onCancel` memungkinkan Anda untuk menentukan langkah apa yang dituju otomatisasi jika ada kejadian pengguna membatalkan otomatisasi. Untuk informasi lebih lanjut, lihat Properti yang dibagi oleh semua tindakan otomatisasi.	21 Desember 2020
Topik baru: Bekerja dengan asosiasi menggunakan IAM	Topik baru telah ditambahkan ke bagian Systems Manager State Manager yang menjelaskan praktik terbaik untuk membuat asosiasi menggunakan IAM. Untuk informasi selengkapnya, lihat Bekerja dengan asosiasi menggunakan IAM.	18 Desember 2020
State Managersekarang mendukung multi-wilayah dan multi-akun	Asosiasi sekarang dapat dibuat atau diperbarui dengan beberapa region atau akun. Untuk informasi selengkapnya, lihat Membuat asosiasi.	15 Desember 2020

Kemampuan baru: Fleet Manager	Fleet Manager, kemampuan AWS Systems Manager, adalah pengalaman antarmuka pengguna terpadu (UI) yang membantu Anda mengelola armada server yang berjalan di AWS, atau lokal dari jarak jauh. DenganFleet Manager, Anda dapat melihat status kesehatan dan kinerja seluruh armada server Anda dari satu konsol. Anda juga dapat mengumpulkan data dari masing-masing instans untuk melakukan pemecahan masalah umum dan tugas manajemen dari konsol. Untuk informasi, lihat AWS Systems ManagerFleet Manager.	15 Desember 2020

Kemampuan baru: Change Manager	Amazon Web Services telah merilis Change Manager kerangka kerja manajemen perubahan perusahaan untuk meminta, menyetujui, menerapkan, dan melaporkan perubahan operasional pada konfigurasi dan infrastruktur aplikasi Anda. Dari satu akun administrator yang didelegasikan, jika Anda menggunakan AWS Organizations, Anda dapat mengelola perubahan di beberapa Akun AWS dalam beberapa Wilayah AWS. Atau, dengan menggunakan akun lokal, Anda dapat mengelola perubahan untuk satu Akun AWS. Gunakan Change Manager untuk mengelola perubahan pada AWS sumber daya dan sumber daya lokal. Untuk informasi, lihat AWS Systems ManagerChange Manager.	15 Desember 2020

Kemampuan baru: Application Manager	Application Managermembantu Anda menyelidiki dan memulihkan masalah dengan AWS sumber daya Anda dalam konteks aplikasi Anda. Application Managermengumpulkan informasi operasi dari beberapa kemampuan Layanan AWS dan Systems Manager menjadi satu AWS Management Console. Untuk informasi, lihat AWS Systems ManagerApplication Manager.	15 Desember 2020

AWS Systems Manager mendukung instans Amazon EC2 untuk macOS	Seiring dengan perilisan support Amazon Elastic Compute Cloud (Amazon EC2) untuk instans macOS, Systems Manager sekarang mendukung banyak operasi pada instans EC2 untuk macOS. Versi yang didukung mencakup macOS 10.14.x (Mojave) dan 10.15.x (Catalina). Untuk informasi selengkap nya, lihat topik berikut. • Untuk informasi tentang penginstalan SSM Agent pada instans EC2macOS, lihat Menginstal dan mengonfigurasi SSM Agent instans EC2 untuk. macOS . • Untuk informasi tentang menambal instans EC2macOS, lihat Cara tambalan diinstal, dan Membuat baseline patch kustom (). macOS • Untuk informasi umum tentang support untuk instans EC2 untuk macOS, lihat instans Mac Amazon EC2 di Panduan Pengguna Amazon EC2 untuk Instans Linux.	30 November 2020

Parameter semu jendela pemeliharaan: Jenis sumber daya baru yang didukung untuk `{{TARGET_ID}}` dan `{{RESOURCE_ID}}`	Jenis sumber daya tambahan sekarang tersedia untuk digunakan dengan parameter semu `{{TARGET_ID}}` dan `{{RESOURCE_ID}}`. Sekarang Anda dapat menggunakan jenis sumber daya `AWS::RDS::DBCluster` dengan kedua parameter semu ini. Untuk informasi tentang parameter semu jendela pemeliharaan, lihat Menggunakan parameter semu saat mendaftarkan tugas jendela pemeliharaan.	27 November 2020
Session Managerplugin untuk AWS CLI versi 1.2.30.0	Versi baru dari Session Manager plugin untuk AWS CLI telah dirilis. Untuk informasi selengkapnya, lihat Session Managerplugin versi terbaru dan riwayat rilis.	24 November 2020
Topik baru: Membandingkan versi dokumen SSM	Anda sekarang dapat membandingkan perbedaan dalam konten antar beberapa versi dokumen SSM di konsol dokumen Systems Manager. Untuk informasi selengkapnya, lihat Membandingkan versi dokumen SSM.	24 November 2020

Systems Manager sekarang mendukung kebijakan titik akhir VPC	Anda sekarang dapat membuat kebijakan untuk titik akhir antarmuka VPC untuk Systems Manager. Untuk informasi selengkapnya, lihat Membuat kebijakan VPC endpoint antarmuka.	18 November 2020
Topik baru: Tentukan nilai batas waktu sesi idle	Anda sekarang dapat menentukan jumlah waktu untuk memungkinkan pengguna menjadi tidak aktif sebelum sesi berakhir denganSession Manager. Untuk informasi selengkap nya, lihat Tentukan nilai waktu habis sesi siaga.	18 November 2020
Fitur Session Manager logging baru	Anda sekarang dapat mengirim aliran terus menerus log data sesi berformat JSON ke Amazon Logs. CloudWatc h Untuk informasi selengkap nya, lihat Streaming data sesi menggunakan Amazon CloudWatch Logs.	18 November 2020
Topik baru: Verifikasi tanda tangan SSM Agent	Anda sekarang dapat memverifikasi tanda tangan kriptografi dari paket installer untuk instance SSM Agent di Linux. Untuk informasi selengkapnya, lihat Skema dan fitur dokumen SSM.	17 November 2020

Topik baru: Memahami status otomatisasi	Topik baru telah ditambahkan ke bab Otomatisasi Systems Manager yang menjelaskan status untuk tindakan dan otomatisasi. Untuk informasi selengkapnya, lihat Memahami status otomatisasi.	17 November 2020
Jenis sumber baru untuk `aws:downloadContent` plugin	Git dan HTTP sekarang didukung sebagai jenis sumber untuk plugin `aws:downloadContent` . Untuk informasi selengkapnya, lihat `aws:downloadContent` .	17 November 2020
Fitur skema dokumen Systems Manager baru (dokumen SSM)	Dalam dokumen SSM dengan skema versi 2.2 atau yang lebih baru, parameter `precondition` sekarang mendukung mereferensikan parameter input dokumen Anda. Untuk informasi selengkapnya, lihat Skema dan fitur dokumen SSM.	17 November 2020

Sumber data baru diExplorer: AWS Config	Explorersekarang menampilkan informasi tentang AWS Config kepatuhan, termasuk ringkasan keseluruhan AWS Config aturan yang sesuai dan tidak patuh, jumlah sumber daya yang sesuai dan tidak sesuai, dan detail spesifik tentang masing-masing (saat Anda menelusuri aturan atau sumber daya yang tidak sesuai). Untuk informasi selengkapnya, lihat Mengedit sumber data Explorer Systems Manager.	11 November 2020
Topik baru: Menjalankan grup Auto Scaling dengan asosiasi	Bagian baru telah ditambahkan State Manager yang menjelaskan praktik terbaik untuk membuat asosiasi untuk menjalankan grup Auto Scaling. Untuk informasi selengkapnya, lihat Menjalankan grup Auto Scaling dengan asosiasi.	10 November 2020
Quick Setupsekarang mendukung penargetan grup sumber daya	Quick Setupsekarang mendukung memilih grup sumber daya sebagai target untuk jenis pengaturan lokal. Untuk informasi selengkapnya, lihat Memilih Target untuk Quick Setup.	5 November 2020

Patch Managermenambahkan dukungan untuk Debian Server 10 LTS, Oracle Linux 7.9 LTS, dan 20.10 STR Ubuntu Server	Anda sekarang dapat menggunakan Patch Manager untuk menambal Debian Server 10 instans LTS, Oracle Linux 7.9 LTS, dan Ubuntu Server 20.10 STR. Untuk informasi selengkapnya, lihat topik berikut: • Patch Managerprasyarat • Bagaimana patch keamanan dipilih • Bagaimana tambalan dipasang • Cara kerja aturan dasar tambalan Debian Server • Cara kerja aturan dasar tambalan Oracle Linux • Cara kerja aturan dasar tambalan Ubuntu Server	4 November 2020

EventBridge Dukungan baru untuk AWS Systems ManagerChange Calendar	Amazon EventBridge sekarang menyediakan dukungan untuk Change Calendar acara acara dalam aturan acara. Ketika status kalender berubah, EventBridge dapat memulai tindakan target yang Anda tetapkan EventBridge aturan. Untuk informasi tentang bekerja dengan EventBridge dan acara Systems Manager, lihat topik berikut. • Mengkonfigurasi EventBridge untuk acara Systems Manager • Referensi: Pola dan jenis EventBridge acara Amazon untuk Systems Manager	4 November 2020
CloudWatch Konfigurasikan untuk membuat OpsItems dari alarm	Anda dapat mengonfigurasi Amazon CloudWatch untuk secara otomatis membuat OpsItem di Systems Manager OpsCenter saat alarm memasuki `ALARM` status. Melakukannya memungkinkan Anda untuk dengan cepat mendiagnosis dan memulihkan masalah dengan AWS sumber daya dari satu konsol. Untuk informasi selengkapnya, lihat Mengkonfigurasi CloudWatch untuk membuat OpsItems dari alarm.	4 November 2020

Support untuk Ubuntu Server 20.10	AWS Systems Manager sekarang mendukung Ubuntu Server 20.10 rilis jangka pendek (STR). Untuk informasi selengkapnya, lihat topik berikut: • Sistem operasi yang didukung • Instal SSM Agent untuk lingkungan hybrid (Linux) • Instal secara manual SSM Agent pada Ubuntu Server instance • Memeriksa SSM Agent status dan memulai agen	22 Oktober 2020
Topik baru: Izinkan profil shell yang dapat dikonfigurasi	Anda sekarang dapat mengizinkan profil shell yang dapat dikonfigurasi denganSession Manager. Dengan mengizinkan profil shell yang dapat dikonfigurasi, Anda dapat menyesuaikan preferensi dalam sesi seperti preferensi shell, variabel lingkungan, direktori kerja, dan menjalankan beberapa perintah ketika sesi dimulai. Untuk informasi selengkapnya, lihat Mengizinkan profil shell yang dapat dikonfigurasi.	21 Oktober 2020

Hasil kepatuhan tambalan sekarang melaporkan CVE mana yang diselesaikan dengan tambalan mana	Untuk sebagian besar sistem Linux yang didukung, ketika Anda melihat hasil kepatuhan patch untuk instans terkelola Anda, detail yang dapat Anda lihat sekarang melaporkan masalah buletin Common Vulnerability and Exposure (CVE) apa yang diselesai kan dengan patch tertentu. Informasi ini dapat membantu Anda menentukan seberapa mendesak Anda perlu menginstal patch yang hilang atau gagal. Untuk informasi selengkapnya, lihat Melihat hasil kepatuhan patch.	20 Oktober 2020

Dukungan yang diperluas untuk metadata patch Linux	Anda sekarang dapat melihat banyak detail tentang patch Linux yang tersedia diPatch Manager. Anda dapat memilih untuk melihat data patch seperti arsitektur, jangka waktu, versi, ID CVE, ID Penasihat, ID Bugzilla, repositori, dan banyak lagi. Selain itu, operasi DescribeAvailablePatchesAPI telah diperbarui untuk mendukung sistem operasi Linux dan pemfilteran sesuai dengan jenis metadata patch yang baru tersedia ini. Untuk informasi selengkapnya, lihat topik berikut: • Melihat tambalan yang tersedia • DescribeAvailablePatchesdan Patch di Referensi AWS Systems Manager API • describe-available-patchesdi AWS Systems Manager bagian Referensi AWS CLI Perintah	16 Oktober 2020
Session Managerplugin untuk AWS CLI versi 1.2.7.0	Versi baru dari Session Manager plugin untuk AWS CLI telah dirilis. Untuk informasi selengkapnya, lihat Session Managerplugin versi terbaru dan riwayat rilis.	15 Oktober 2020

Topik baru: Skema dokumen sesi	Topik baru Skema dokumen sesi menjelaskan elemen skema untuk dokumen Sesi. Informasi ini dapat membantu Anda membuat dokumen Sesi kustom di mana Anda menentukan preferensi untuk jenis sesi yang Anda gunakanSession Manager.	15 Oktober 2020
Topik baru: Pencarian teks gratis untuk dokumen SSM	Kotak pencarian di halaman Dokumen Systems Manager sekarang mendukung pencarian teks bebas. Pencarian teks bebas membandingkan istilah pencarian atau istilah yang Anda masukkan terhadap nama dokumen di setiap dokumen SSM. Untuk informasi selengkapnya, lihat Menggunakan pencarian teks bebas.	15 Oktober 2020
Topik baru: Memecahkan masalah ketersediaan instans terkelola Amazon EC2	Topik baru Pemecahan masalah ketersediaan instans Amazon EC2 terkelola membantu Anda menyelidiki mengapa suatu instans Amazon EC2 yang telah dikonfirmasi berjalan tidak tersedia dalam daftar instans terkelola yang tersedia di Systems Manager.	6 Oktober 2020

Parameter Storereorganisasi pasal	Untuk membantu Anda menemukan informasi yang Anda butuhkan secara lebih efisien, kami mengatur ulang konten Parameter Store di bagian Panduan AWS Systems Manager Pengguna. Sebagian besar konten sekarang diatur di bagian Menyiapkan Parameter Store dan Bekerja dengan Parameter Store. Selain itu, topik AWS Systems ManagerParameter Storetelah diperluas untuk mencakup bagian-bagian berikut: • Bagaimana bisa Parameter Store menguntungkan organisasi saya? • Siapa yang harus menggunakanParameter Store? • Apa saja fitur-fiturnyaPara meter Store? • Apa itu parameter?	1 Oktober 2020

Topik terkait kepatuhan tambalan baru	Topik berikut ini telah ditambahkan untuk membantu Anda mengidentifikasi instans terkelola yang berada di luar kepatuhan patch, memahami berbagai jenis pemindaian kepatuhan patch, dan mengambil langkah-langkah yang tepat untuk membawa instans Anda ke dalam kepatuhan. • Mengidentifikasi contoh yang tidak sesuai • Menambal contoh yang tidak sesuai • Melihat hasil kepatuhan tambalan	24 September 2020
SSM Agentversi 3.0	Systems Manager meluncurkan versi baruSSM Agent.	21 September 2020

Topik baru dan diperbarui: Amazon EventBridge menggantikan CloudWatch Acara untuk manajemen acara	CloudWatch Acara dan EventBridge merupakan layanan dan API dasar yang sama, tetapi EventBridge menyediakan lebih banyak fitur dan sekarang menjadi cara yang lebih disukai untuk mengelola acara Anda AWS. (Perubahan yang Anda buat di salah satu CloudWatch atau EventBridge tercermin di setiap konsol.) Referensi untuk CloudWatch Acara dan prosedur yang ada di seluruh Panduan AWS Systems Manager Pengguna telah diperbarui untuk mencerminkan EventBridge dukungan. Di samping itu, topik baru berikut ini telah ditambahkan. • Acara Monitoring Systems Manager • Mengkonfigurasi EventBridge untuk acara Systems Manager • Contoh tipe target Systems Manager • Referensi: Pola dan jenis EventBridge acara Amazon untuk Systems Manager	18 September 2020

Integrasi dan AWS Security HubPatch Manager	Anda sekarang dapat berintegrasi Patch Manager dengan AWS Security Hub. Security Hub memberikan pandangan komprehensif tentang status keamanan Anda AWS dan membantu Anda memeriksa lingkungan Anda terhadap standar industri keamanan dan praktik terbaik. Ketika terintegrasi denganPatch Manager, Security Hub memantau status patching armada Anda dari sudut pandang keamanan. Untuk informasi selengkapnya, lihat Mengintegrasikan Patch Manager dengan AWS Security Hub.	17 September 2020

Parameter semu jendela pemeliharaan: Jenis sumber daya baru yang didukung untuk `{{TARGET_ID}}` dan `{{RESOURCE_ID}}`	Ketika Anda mendaftarkan sebuah tugas jendela pemeliharaan, Anda menggunakan pilihan `--task-invocation-parameters` untuk menentukan parameter yang bersifat unik untuk masing-masing dari keempat jenis tugas. Anda juga dapat mereferensikan nilai tertentu menggunakan sintaks parameter semu, seperti `{{TARGET_ID}}` dan `{{RESOURCE_ID}}` . Ketika tugas jendela pemeliharaan berjalan, ia meneruskan nilai yang benar dan bukan placeholder parameter semu. Dua jenis sumber daya tambahan sekarang tersedia untuk digunakan dengan parameter semu `{{TARGET_ID}}` dan `{{RESOURCE_ID}}` . Sekarang Anda dapat menggunakan jenis sumber daya `AWS::RDS::DBInstance` dan `AWS::SSM::ManagedInstance` dengan kedua parameter semu ini. Untuk informasi tentang parameter semu jendela pemeliharaan, lihat Menggunakan parameter semu saat mendaftarkan tugas jendela pemeliharaan.	14 September 2020

Instans patch sesuai permintaan dengan opsi 'Patch now' baru	Anda sekarang dapat menggunakan konsol Systems Manager untuk mem-patch instans, atau memindai patch yang hilang, kapan saja. Anda dapat melakukan ini tanpa harus membuat atau memodifikasi jadwal, atau menentukan opsi konfigura si patching lengkap untuk mengakomodasi kebutuhan patching segera. Anda hanya perlu menentukan apakah akan memindai atau menginstal tambalan dan mengidentifikasi instance target untuk operasi. Patch Managersecara otomatis menerapkan baseline patch default saat ini untuk jenis instans Anda dan menerapka n opsi praktik terbaik untuk berapa banyak instance yang ditambal sekaligus, dan berapa banyak kesalahan yang diizinkan sebelum operasi gagal. Untuk informasi lebih lanjut, lihat Patching instans sesuai permintaan.	9 September 2020

Topik baru: Memeriksa SSM Agent status dan memulai agen	Topik baru Memeriksa SSM Agent status dan memulai agen memberikan perintah untuk memeriksa SSM Agent apakah berjalan pada setiap sistem operasi pendukung. Hal ini juga menyediakan perintah untuk memulai agen jika tidak berjalan.	7 September 2020
Patch Managersekarang mendukung Ubuntu Server 20,04 LTS	Anda sekarang dapat menggunakan Patch Manager untuk menambal Ubuntu Server 20.04 instance LTS. Untuk informasi selengkapnya, lihat topik berikut: • Bagaimana patch keamanan dipilih • Bagaimana tambalan dipasang • Cara kerja aturan dasar tambalan Ubuntu Server	31 Agustus 2020
Topik baru untuk kasus Penggunaan dan praktik terbaik	Kami telah menambahkan topik baru untuk membantu pengguna dengan cepat memahami perbedaan antara Maintenance Windows danState Manager. Untuk informasi selengkapnya, lihat Memilih antara State Manager dan Maintenance Windows.	28 Agustus 2020

OpsCenterFitur baru	OpsCentersertakan fitur baru untuk membantu Anda menemukan dan menjalank an runbook Otomasi dengan cepat untuk memperbaiki masalah. Untuk informasi selengkapnya, lihat Fitur buku runbook otomatisasi di OpsCenter.	19 Agustus 2020
Sumber data baru dalamExpl orer: AWS Support kasus	Explorersekarang menampilk an informasi tentang AWS Support kasus. Anda harus memiliki akun Perusahaan atau Bisnis yang disiapkan AWS Support. Untuk informasi selengkapnya, lihat Mengedit sumber data Explorer Systems Manager.	13 Agustus 2020
Distributorsekarang menyediakan paket pihak ketiga dari Trend Micro.	Distributorsekarang termasuk paket pihak ketiga dari Trend Micro. Anda dapat menggunak an Distributor untuk menginsta l agen Trend Micro Cloud One pada instans terkelola Anda. Trend Micro Cloud One membantu Anda mengamank an beban kerja Anda di cloud. Untuk informasi selengkapnya, lihat AWSDistributor.	12 Agustus 2020

Plugin `aws:configurePackage` dokumen sekarang menyertakan parameter additionalArguments.	plugin dokumen Perintah Systems Manager `aws:configurePackage` sekarang mendukung menyediakan parameter tambahan untuk skrip Anda (instal, bongkar, dan perbarui) dengan parameter `additionalArguments` yang baru. Untuk informasi lebih lanjut, lihat topiknya `aws:configurePackage`.	11 Agustus 2020
AppConfigkonten dipindahkan ke panduan pengguna terpisah	Informasi tentang AWS AppConfig telah dipindahkan ke panduan pengguna yang terpisah. Untuk informasi lebih lanjut, lihat Apa itu AWSAppConfig? AppConfig juga memiliki halaman arahan dokumentasi terpisah dengan tautan ke panduan pengguna, referensi AppConfig API, dan AppConfig lokakarya baru.	3 Agustus 2020

Quick Setupsekarang mendukung AWS Organizations	Quick Setupsekarang mendukung AWS Organizations memungkinkan Anda untuk dengan cepat mengkonfigurasi peran keamanan yang diperlukan dan kemampuan Systems Manager yang umum digunakan di beberapa akun dan Wilayah. Untuk informasi selengkapnya, lihat AWS Systems ManagerQuick Setup.	23 Juli 2020
Sumber data baru diExplorer: kepatuhan asosiasi	Explorersekarang menampilkan data kepatuhan asosiasi dariState Manager. Untuk informasi selengkapnya, lihat Mengedit sumber data Explorer Systems Manager.	23 Juli 2020
Dokumen Systems Manager Command baru untuk menghidupkan dan mematikan Kernel Live Patching	Dokumen `AWS-ConfigureKernelLivePatching` ini sekarang tersedia untuk digunakan Run Command ketika Anda ingin mengaktifkan atau mematikan Kernel Live Patching pada instans Amazon Linux 2. Dokumen ini menggantikan kebutuhan untuk membuat dokumen Perintah kustom Anda sendiri untuk tugas-tugas ini. Untuk informasi lebih lanjut, lihat Menggunakan Kernel Live Patching pada instans Amazon Linux 2	22 Juli 2020

Kuota Otomasi yang Diperbarui	Service Quota untuk Otomatisasi telah diperbarui termasuk antrean terpisah untuk otomatisasi pengendali rate. Untuk informasi selengkapnya, lihat Otomatisasi AWS Systems Manager.	20 Juli 2020
Tentukan jumlah hari offset jadwal untuk jendela pemeliharaan menggunakan konsol	Menggunakan konsol Systems Manager, Anda sekarang dapat menentukan jumlah hari untuk menunggu setelah tanggal dan waktu yang ditentukan oleh ekspresi CRON sebelum menjalankan jendela pemeliharaan. (Sebelumnya, opsi ini hanya tersedia saat menggunakan AWS SDK atau alat baris perintah.) Misalnya, jika ekspresi CRON Anda menjadwalkan jendela pemeliharaan untuk berjalan pada Selasa ketiga setiap bulan pada 11:30 siang – `cron(0 30 23 ? * TUE#3 *)` – dan Anda menentukan jadwal offset 2, jendela tidak akan berjalan sampai dua hari kemudian pada 11:30 siang. Untuk informasi selengkapnya, lihat Ekspresi Cron dan Rate untuk Systems Manager dan Tentukan jumlah jadwal offset days untuk jendela pemeliharaan.	17 Juli 2020

Perbarui PowerShell menggunakan Run Command	Untuk membantu Anda memperbarui PowerShel l ke versi 5.1 pada instans R2 Windows Server 2012 dan 2012, kami menambahk an panduan ke Panduan Pengguna. AWS Systems Manager Untuk informasi selengkapnya, lihat Memperbarui PowerShell menggunakan Run Command.	30 Juni 2020
Patch Managersekarang mendukung CentOS 8.0 dan 8.1	Anda sekarang dapat menggunakan Patch Manager untuk menambal instance CentOS 8.0 dan 8.1. Untuk informasi selengkapnya, lihat topik berikut: • Bagaimana patch keamanan dipilih • Bagaimana tambalan dipasang • Cara kerja aturan dasar tambalan di CentOS • Instal secara manual SSM Agent pada instance CentOS • Instal SSM Agent untuk lingkungan hybrid (Linux)	27 Juni 2020

AppConfigIntegrasi dengan AWS CodePipeline	AppConfigadalah tindakan penerapan terintegrasi untuk AWS CodePipeline (CodePipeline). CodePipeline adalah layanan pengiriman berkelanjutan yang dikelola sepenuhnya yang membantu Anda mengotomatiskan saluran pipa rilis Anda untuk pembaruan aplikasi dan infrastruktur yang cepat dan andal. CodePipel inemengotomatiskan fase build, test, dan deploy dari proses rilis Anda setiap kali ada perubahan kode, berdasarkan model rilis yang Anda tentukan. Integrasi AppConfig dengan CodePipeline menawarkan manfaat berikut. Untuk informasi selengkapnya, lihat AppConfig integrasi dengan CodePipeline. • Pelanggan yang menggunakan CodePipeline untuk mengelola orkestrasi sekarang memiliki cara ringan untuk menerapkan perubahan konfigurasi ke aplikasi mereka tanpa harus menerapkan seluruh basis kode mereka. • Pelanggan yang ingin menggunakan AppConfig untuk mengelola penerapan	25 Juni 2020

	konfigurasi tetapi terbatas karena AppConfig tidak mendukung kode atau penyimpanan konfigurasi mereka saat ini, sekarang memiliki opsi tambahan. CodePipeline mendukung AWS CodeCommit,GitHub, dan BitBucket (untuk beberapa nama).	
Bab baru: Integrasi produk dan layanan	Untuk membantu Anda memahami bagaimana Systems Manager terintegrasi dengan Layanan AWS produk dan layanan lainnya, babak baru telah ditambahkan ke Panduan AWS Systems Manager Pengguna. Untuk informasi selengkapnya, lihat Integrasi produk dan layanan dengan Systems Manager.	23 Juni 2020
Reorganisasi Bab Otomasi	Untuk membantu Anda menemukan apa yang Anda butuhkan, kami menata ulang topik di bagian Otomatisasi dalam Panduan Pengguna AWS Systems Manager . Sebagai contoh, referensi tindakan Otomatisasi dan runbook Otomatisasi sekarang menjadi bagian teratas dalam bab tersebut. Untuk informasi selengkapnya, lihat Otomatisasi AWS Systems Manager.	23 Juni 2020

Tentukan jumlah hari offset jadwal untuk jendela pemeliharaan	Menggunakan alat baris perintah atau AWS SDK, Anda sekarang dapat menentukan beberapa hari untuk menunggu setelah tanggal dan waktu yang ditentukan oleh ekspresi CRON sebelum menjalankan jendela pemeliharaan. Misalnya, jika ekspresi CRON Anda menjadwalkan jendela pemeliharaan untuk berjalan pada Selasa ketiga setiap bulan pada 11:30 siang – `cron(0 30 23 ? * TUE#3 *)` – dan Anda menentukan jadwal offset 2, jendela tidak akan berjalan sampai dua hari kemudian pada 11:30 siang. Untuk informasi selengkapnya, lihat Ekspresi Cron dan Rate untuk Systems Manager dan Tentukan jumlah jadwal offset days untuk jendela pemeliharaan.	19 Juni 2020

Patch Managerdukungan untuk Kernel Live Patching di Amazon Linux 2 instans	Kernel Live Patching untuk Amazon Linux 2 memungkinkan Anda menerapkan patch kerentanan dan bug penting untuk kernel Linux yang berjalan, tanpa reboot atau gangguan pada aplikasi yang berjalan. Anda sekarang dapat mengizinkan fitur dan menerapkan patch langsung kernel menggunakanPatch Manager. Untuk informasi, lihat Menggunakan Kernel Live Patching pada instans Amazon Linux 2.	16 Juni 2020
Patch Managermeningkatkan dukungan Oracle Linux versi	Sebelumnya, hanya Patch Manager didukung versi 7.6 dari. Oracle Linux Seperti yang tercantum dalam Patch Managerprasyarat, dukungan sekarang mencakup versi 7.5-7.8.	16 Juni 2020

Contoh skenario untuk menggunakan `InstallOverrideList` parameter dalam operasi penambalan	Topik baru Contoh skenario untuk menggunakan parameter `InstallOverrideList` menjelaskan strategi untuk menggunakan parameter `InstallOverrideList` dalam `AWS-RunPatchBaseline` untuk menerapkan jenis patch berbeda ke suatu grup target, pada jadwal jendela pemeliharaan yang berbeda, sambil tetap menggunakan dasar patch tunggal.	11 Juni 2020
Strategi penyebaran yang telah ditentukan untuk AppConfig	AppConfigsekarang menawarkan strategi penyebaran yang telah ditentukan. Untuk informasi selengkapnya, lihat Membuat strategi deployment.	10 Juni 2020

Patch Managersekarang mendukung Red Hat Enterprise Linux (RHEL) 7.8-8.2	Anda sekarang dapat menggunakan Patch Manager untuk menambal instance RHEL 7.8-8.2. Untuk informasi selengkapnya, lihat topik berikut: • Bagaimana patch keamanan dipilih • Bagaimana tambalan dipasang • Cara kerja aturan dasar tambalan RHEL • Instal secara manual SSM Agent pada Red Hat Enterprise Linux instance • Instal SSM Agent untuk lingkungan hybrid (Linux)	9 Juni 2020

Explorermendukung administrasi yang didelegasikan	Jika Anda menggabungkan Explorer data dari beberapa Wilayah AWS dan Akun AWS dengan menggunakan sinkronisasi data sumber daya dengan AWS Organizations, maka kami sarankan Anda mengonfigurasi administrator yang didelegasikan untuk. Explorer Administrator yang didelegasikan meningkatkan Explorer keamanan dengan membatasi jumlah Explorer administrator yang dapat membuat atau menghapus sinkronisasi data sumber daya multi-akun dan Wilayah hanya untuk satu individu. Anda juga tidak perlu lagi masuk ke akun AWS Organizations manajemen untuk mengelola sinkronisasi data sumber daya. Explorer Untuk informasi selengkapnya, lihat Mengkonfigurasi Administrator yang Didelegasikan.	3 Juni 2020

Terapkan State Manager asosiasi hanya pada interval Cron yang ditentukan berikutnya	Jika Anda tidak ingin State Manager asosiasi berjalan segera setelah Anda membuatnya, Anda dapat memilih asosiasi Terapkan hanya pada opsi interval Cron yang ditentukan berikutnya di konsol Systems Manager. Untuk informasi selengkapnya, lihat Membuat asosiasi.	3 Juni 2020
Sumber data baru diExplorer: AWS Compute Optimizer	Explorersekarang menampilkan data dari AWS Compute Optimizer. Ini termasuk jumlah instans EC2 kurang dari ditentukan dan Lebih dari yang ditentukan, temuan optimasi, detail harga sesuai permintaan, dan rekomendasi untuk tipe instans dan harga instans. Untuk informasi selengkapnya, lihat detail penyiapan AWS Compute Optimizer di Menyiapkan layanan terkait.	26 Mei 2020

Bab baru: Menandai Sumber Daya Systems Manager	Bab baru Penandaan sumber day Systems Manager menyediakan gambaran umum tentang cara Anda dapat menggunakan tag dengan enam jenis sumber daya yang dapat ditandai di Systems Manager. Bab ini juga menyediakan instruksi komprehensif untuk menambahkan dan menghapus tag dari jenis sumber daya ini: • Dokumen • Jendela pemeliharaan • Instans terkelola • OpsItems • Parameter-parameter • Dasar patch	25 Mei 2020

Instal Paket Layanan Windows dan upgrade versi minor Linux menggunakan Patch Manager	Topik baru Tutorial: Buat baseline patch untuk menginstal Paket Layanan Windows (konsol) menunjukkan bagaimana Anda dapat membuat baseline patch yang ditujukan khusus untuk menginstal Paket Layanan Windows. Topik Membuat dasar patch kustom (Linux) telah diperbarui dengan informasi tentang termasuk pemutakhiran versi minor untuk sistem operasi Linux di dasar patch.	21 Mei 2020
Parameter Storereorganisasi pasal	Semua topik yang berhubungan dengan konfigurasi atau pengaturan opsi untuk Parameter Store operasi telah dikonsolidasikan ke dalam bagian Pengaturan Parameter Store. Ini termasuk topik Mengelola tingkatan parameter dan Meningkatkan Parameter Store throughput, yang telah dipindahkan dari bagian lain dari chapter ini.	18 Mei 2020

Topik baru untuk membuat string tanggal dan waktu untuk berinteraksi dengan operasi API Systems Manager.	Topik baru Membuat string tanggal dan waktu yang diformat untuk Systems Manager menjelaskan cara membuat string tanggal dan waktu yang diformat untuk berinteraksi dengan operasi API Systems Manager.	13 Mei, 2020
Tentang izin untuk mengenkripsi parameter SecureString	Topik baru Membatasi akses ke parameter Systems Manager menggunakan kebijakan IAM menjelaskan perbedaan antara mengenkripsi `SecureString` parameter Anda menggunakan AWS KMS key dan menggunakan yang disediakan oleh. Kunci yang dikelola AWS AWS	13 Mei, 2020

Patch Managersekarang mendukung Debian Server dan Oracle Linux 7.6 sistem operasi	Anda sekarang dapat menggunakan Patch Manager untuk menambal Debian Server dan Oracle Linux instance. Patch Managerme ndukung patching Debian Server 8.x dan 9.x dan Oracle Linux 7.6 versi. Untuk informasi selengkapnya, lihat topik berikut: • Bagaimana patch keamanan dipilih • Bagaimana tambalan dipasang • Cara kerja aturan dasar tambalan Debian Server • Cara kerja aturan dasar tambalan Oracle Linux	7 Mei 2020
Buat State Manager asosiasi yang menargetkan AWS Resource Groups	Selain menargetkan tag, instance individual, dan semua instance di Anda Akun AWS, kini Anda dapat membuat State Manager asosiasi yang menargetkan instance. AWS Resource Groups Untuk informasi selengkapnya, lihat Tentang target dan kontrol tarif dalam State Manager asosiasi	7 Mei 2020

Tipe `aws:ec2:image` data baru Parameter Store untuk memvalidasi ID AMI	Saat Anda membuat parameter `String`, Anda dapat menentukan tipe data sebagai `aws:ec2:image` untuk memastikan bahwa nilai parameter yang Anda masukkan adalah format ID Amazon Machine Image (AMI) yang valid. Support untuk format ID AMI memungkinkan Anda untuk tidak harus memperbarui semua skrip dan templat Anda dengan ID baru setiap kali AMI yang ingin Anda gunakan dalam proses berubah. Anda dapat membuat parameter dengan tipe data `aws:ec2:image` , dan untuk nilainya, masukkan ID AMI. Ini adalah AMI yang ingin Anda gunakan untuk membuat instans baru. Anda kemudian mereferensi parameter ini dalam templat, perintah Anda. Saat Anda siap menggunakan yang berbedaAMI, perbarui nilai parameter. Parameter Storememvalidasi AMI ID baru, dan Anda tidak perlu memperbarui skrip dan templat Anda. Untuk informasi selengkapnya, lihat Dukungan parameter native untuk ID Amazon Machine Image.	5 Mei 2020

Mengelola kode keluar dalam Run Command perintah	Run Commandmemungkinkan Anda untuk menentukan bagaimana kode keluar ditangani dalam skrip Anda. Secara default, kode keluar dari perintah terakhir yang dijalankan dalam skrip dilaporkan sebagai kode keluar untuk seluruh skrip. Namun, Anda dapat menyertakan pernyataan bersyarat shell untuk keluar dari skrip jika perintah sebelum yang terakhir gagal menggunakan pendekatan berikut. Sebagai contoh, lihat topik baru Mengelola kode keluar dalam Run Command perintah.	5 Mei 2020
Parameter publik baru dirilis untuk zona ketersediaan dan zona lokal	Parameter publik telah dirilis untuk membuat informasi tentang AWS availability zone dan local zone tersedia secara terprogram. Ini merupakan tambahan dari parameter publik infrastruktur global yang ada untuk Layanan AWS dan Wilayah AWS. Untuk informasi selengkapnya, lihat Memanggil parameter publik untuk Layanan AWS, Wilayah, titik akhir, Availability Zone, local zone, dan Wavelength Zones.	4 Mei 2020

Sumber data baru diExplorer: AWS Trusted Advisor	Explorersekarang menampilkan data dari AWS Trusted Advisor. Ini termasuk status pemeriksaan praktik terbaik dan rekomendasi di bidang-bidang berikut: optimasi biaya, keamanan, toleransi kesalahan, kinerja, dan kuota layanan. Untuk informasi selengkapnya, lihat detail penyiapan Trusted Advisor di Menyiapkan layanan terkait.	4 Mei 2020

Buat State Manager asosiasi yang menjalankan Chef resep	Anda dapat membuat State Manager asosiasi yang menjalankan Chef buku masak dan resep dengan menggunakan `AWS-ApplyChefRecipes` dokumen. Dokumen ini menawarkan manfaat berikut untuk menjalankan Chef resep: • Mendukung beberapa rilis Chef (Chef11 hingga Chef 14). • Secara otomatis menginstal perangkat lunak Chef klien pada instance target. • Secara opsional menjalankan pemeriksaan kepatuhan Systems Manager pada instans target, dan menyimpan hasil pemeriksaan kepatuhan di bucket S3. • Menjalankan beberapa buku masak dan resep dalam satu dokumen. • Secara opsional menjalankan resep dalam mode `why-run`, untuk menunjukkan resep mana yang akan berubah pada instans target tanpa membuat perubahan. • Secara opsional menerapkan atribut JSON kustom ke berjalannya `chef-client` .	19 Maret 2020

	Untuk informasi selengkapnya, lihat Membuat asosiasi yang menjalankan Chef resep	
Sinkronkan data inventaris dari beberapa Akun AWS ke bucket Amazon S3 pusat	Anda dapat menyinkronkan data Systems Manager Inventory dari beberapa Akun AWS ke bucket S3 pusat. Akun harus didefinisikan dalam AWS Organizations. Untuk informasi selengkap nya, lihat Membuat sinkronis asi data sumber daya Inventari s untuk beberapa akun yang didefinisikan dalam AWS Organizations.	16 Maret 2020
Simpan AppConfig konfigurasi di Amazon S3	Sebelumnya, AppConfig hanya mendukung konfigurasi aplikasi yang disimpan dalam dokumen atau Parameter Store parameter Systems Manager (SSM). Selain opsi ini, AppConfig sekarang mendukung penyimpanan konfigurasi di Amazon S3. Untuk informasi lebih lanjut, lihat Tentang konfigurasi yang disimpan dalam Amazon S3.	13 Maret 2020
SSM Agentdiinstal secara default di Amazon ECS yang dioptimalkan AMIs	SSM Agentsekarang diinstal secara default di Amazon ECS AMIs yang dioptimalkan. Untuk informasi selengkap nya, lihat Bekerja dengan SSM Agent.	25 Februari 2020

Buat AppConfig konfigurasi di konsol	AppConfigsekarang memungkinkan Anda untuk membuat konfigurasi aplikasi di konsol pada saat Anda membuat profil konfigurasi. Untuk informasi selengkapnya, lihat Membuat konfigurasi dan profil konfigurasi.	13 Februari 2020
Persetujuan otomatis hanya patch yang dirilis hingga tanggal yang ditentukan	Selain opsi untuk secara otomatis menyetujui tambalan untuk penginstalan dalam jumlah hari tertentu setelah dirilis, Patch Manager sekarang mendukung kemampuan untuk menyetuju i otomatis hanya tambalan yang dirilis pada atau sebelum tanggal yang Anda tentukan. Sebagai contoh, jika Anda menentukan 7 Juli 2020 sebagai tanggal cutoff di dasar patch Anda, tidak ada patch yang dirilis pada atau setelah 8 Juli 2020 yang diinstal secara otomatis. Untuk informasi selengkapnya, lihat Tentang baseline kustom dan Bekerja dengan baseline patch kustom (konsol).	12 Februari 2020

Gunakan parameter semu {{RESOURCE_ID}} dalam tugas jendela pemeliharaan	Ketika Anda mendaftarkan sebuah tugas jendela pemeliharaan, Anda menentukan parameter yang bersifat unik untuk jenis tugas tersebut. Anda dapat mereferensikan nilai tertentu menggunakan sintaks parameter semu, seperti `{{TARGET_ID}}` , `{{TARGET_TYPE}}` , dan `{{WINDOW_TARGET_ID}}` . Ketika tugas jendela pemeliharaan berjalan, ia meneruskan nilai yang benar dan bukan placeholder parameter semu. Untuk mendukung sumber daya yang merupakan bagian dari sebuah resource group sebagai target, Anda dapat menggunakan parameter semu `{{RESOURCE_ID}}` untuk meneruskan nilai untuk sumber daya seperti tabel DynamoDB, bucket S3, dan tipe lain yang didukung. Untuk informasi selengkapnya, lihat topik berikut di Tutorial: Membuat dan mengkonfigurasi jendela pemeliharaan (AWS CLI): • Menggunakan parameter semu saat mendaftarkan tugas jendela pemeliharaan	6 Februari 2020

	• Contoh: Daftarkan tugas dengan jendela pemeliharaan	
Jalankan kembali perintah dengan cepat	Systems Manager menyertakan dua opsi untuk membantu Anda menjalankan kembali perintah dari Run Commandhalaman di AWS Systems Manager konsol. Jalankan kembali: Tombol ini memungkinkan Anda untuk menjalankan perintah yang sama tanpa membuat perubahan. Salin ke baru: Tombol ini menyalin pengaturan dari satu perintah ke perintah baru dan memberikan Anda pilihan untuk mengedit pengaturan tersebut sebelum Anda menjalankannya. Untuk informasi lebih lanjut, lihat Menjalankan kembali perintah.	5 Februari 2020

Mengembalikan dari tingkat instans lanjutan ke tingkat instans standar	Jika sebelumnya Anda mengkonfigurasi semua instans on-premise yang berjalan di lingkungan hibrid Anda untuk menggunakan tingkat instans lanjutan, kini Anda dapat dengan cepat mengkonfigurasi instans tersebut untuk menggunak an tingkat instans standar. Kembali ke tingkat instans standar berlaku untuk semua instance hybrid dalam satu dan satu. Akun AWS Wilayah AWS Mengembalikan ke tingkat instans standar mempengaruhi ketersedi aan beberapa kemampuan Systems Manager. Untuk informasi selengkapnya, lihat Mengembalikan dari tingkat instans lanjutan ke tingkat instans standar.	16 Januari 2020

Opsi baru untuk melewati reboot instance setelah instalasi patch	Sebelumnya, instance terkelola selalu di-boot ulang setelah patch Patch Manager diinstal pada mereka. parameter `RebootOption` baru dalam dokumen SSM `AWS-RunPatchBaseline` memungkinkan Anda untuk menentukan apakah Anda ingin instans Anda untuk di-reboot secara otomatis setelah patch yang baru diinstal. Untuk informasi selengkap nya, lihat Nama parameter : RebootOption dalam topik Tentang dokumen `AWS-RunPatchBaseline` SSM.	15 Januari 2020
Topik baru: 'Menjalankan PowerShell skrip pada instance Linux'	Topik baru yang menjelask an Run Command cara menggunakan PowerShel l skrip pada instance Linux. Untuk informasi selengkapnya, lihat Menjalankan PowerShell skrip pada instance Linux.	10 Januari 2020
Pembaruan untuk 'mengkonf igurasi SSM Agent untuk menggunakan proksi'	Nilai yang akan ditentukan saat mengonfigurasi SSM Agent untuk menggunakan proxy telah diperbarui untuk mencerminkan opsi untuk server proxy HTTP dan server proxy HTTPS. Untuk informasi selengkapnya, lihat SSM AgentMengkonfigurasi untuk menggunakan proxy.	9 Januari 2020

Bab “Keamanan” baru menguraikan praktik untuk mengamankan sumber daya Systems Manager	Bab Keamanan yang baru di Panduan Pengguna AWS Systems Manager membantu Anda memahami cara menerapkan model tanggung jawab bersama saat menggunakan Systems Manager. Topik dalam bab tersebut menunjukkan kepada Anda cara mengkonfigurasi Systems Manager untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga mempelajari cara menggunakan Layanan AWS yang lain yang membantu Anda memantau dan mengamankan sumber daya Systems Manager Anda. **Note** Sebagai bagian dari pembaruan ini, panduan pengguna bab "Autentikasi dan Kontrol Akses" telah diganti dengan bagian baru yang lebih sederhana, Identity and access management untuk AWS Systems Manager.	24 Desember 2019

Contoh baru runbook Otomasi kustom	Satu set contoh runbook Otomatisasi kustom telah ditambahkan ke panduan pengguna. Contoh ini menunjukkan cara menggunakan berbagai tindakan Otomatisasi untuk menyederhanakan tugas deployment, pemecahan masalah, dan pemeliharaan, dan dimaksudkan untuk membantu Anda menulis runbook Otomatisasi kustom Anda sendiri. Untuk informasi lebih lanjut, lihat Contoh runbook Otomatisasi kustom. Anda juga dapat melihat konten runbook Otomatisasi yang dikelola Amazon di konsol Systems Manager. Untuk informasi lebih lanjut, lihat Referensi runbook Otomatisasi Systems Manager.	23 Desember 2019

Support untuk Oracle Linux	Systems Manager sekarang mendukung Oracle Linux 7.5 dan 7.7. Untuk informasi tentang menginstal SSM Agent instans EC2 secara manual untuk Oracle Linux instans, lihat. Oracle Linux Untuk informasi tentang menginstal SSM Agent pada Oracle Linux server di lingkungan hybrid, lihat Langkah 6: Instal Agen SSM untuk lingkungan hybrid (Linux).	19 Desember 2019

Luncurkan Session Manager sesi dari konsol Amazon EC2	Anda sekarang dapat memulai Session Manager sesi dari konsol Amazon Elastic Compute Cloud (Amazon EC2). Bekerja dengan tugas yang berhubungan dengan sesi dari konsol Amazon EC2 memerlukan izin IAM yang berbeda untuk pengguna dan administrator. Anda dapat memberikan izin untuk menggunakan Session Manager konsol dan AWS CLI hanya, untuk menggunakan konsol Amazon EC2 saja, atau untuk menggunakan ketiga alat. Untuk informasi selengkapnya, lihat topik berikut. • Quickstart kebijakan IAM default untuk Session Manager • Memulai sesi (konsol Amazon EC2)	18 Desember 2019

CloudWatch dukungan untuk Run Command metrik dan alarm	AWS Systems Manager sekarang menerbitkan metrik tentang status Run Command perintah ke CloudWatch, memungkinkan Anda untuk mengatur alarm berdasarkan metrik tersebut. Nilai status terminal untuk perintah yang dapat Anda lacak metriknya mencakup `Success`, `Failed`, dan `Delivery Timed Out`. Untuk informasi selengkap nya, lihat Run CommandMe trik pemantauan menggunak an Amazon CloudWatch.	17 Desember 2019
Kemampuan Systems Manager baru: Change Calendar	Gunakan Systems Manager Change Calendar untuk menentukan periode waktu (peristiwa) di mana Anda ingin membatasi atau mencegah perubahan kode (seperti dari runbook atau AWS Lambda fungsi Systems Manager Automation) ke sumber daya. Change Calendar adalah jenis dokumen Systems Manager baru yang menyimpan data iCalendar 2.0 dalam format teks biasa. Untuk informasi selengkapnya, lihat AWS Systems Manager Change Calendar.	11 Desember 2019

Kemampuan Systems Manager baru: AWSAppConfig	Gunakan AppConfig untuk membuat, mengelola, dan menyebarkan konfigura si aplikasi dengan cepat. AppConfigmendukung penerapan terkontrol untuk aplikasi dari berbagai ukuran. Anda dapat menggunakan AppConfig dengan aplikasi yang dihosting di instans EC2, wadah AWS Lambda, aplikasi seluler, atau perangkat IoT. Untuk mencegah kesalahan saat menerapkan konfigurasi aplikasi, AppConfig sertakan validator. Sebuah validator menyediakan pemeriksaan sintaksis atau semantik untuk memastikan bahwa konfigura si yang ingin Anda deploy berfungsi sebagaimana yang dimaksud. Selama penerapan konfigurasi, AppConfig monitor aplikasi untuk memastikan bahwa penerapan berhasil. Jika sistem mengalami kesalahan atau jika penerapan memulai alarm, AppConfig memutar kembali perubahan untuk meminimalkan dampak bagi pengguna aplikasi Anda. Untuk informasi selengkapnya, lihat AWSAppConfig.	25 November 2019

Kemampuan Systems Manager Baru: Systems Manager Explorer	AWS Systems Manager Explorer adalah dasbor operasi yang dapat disesuaikan yang melaporkan informasi tentang sumber daya Anda AWS . Explorermenampilkan tampilan agregat data operasi (OpsData) untuk Anda Akun AWS dan seluruh Wilayah AWS. DiExplorer, OpsData sertakan metadata tentang instans EC2 Anda, detail kepatuhan tambalan, dan item pekerjaan operasional (). OpsItems Explorerm emberikan konteks tentang bagaimana OpsItems didistribusikan di seluruh unit bisnis atau aplikasi Anda, bagaimana tren mereka dari waktu ke waktu, dan bagaimana mereka bervariasi menurut kategori. Anda dapat mengelompokkan dan memfilter informasi Explorer untuk fokus pada item yang relevan bagi Anda dan yang memerlukan tindakan. Saat mengidentifikasi masalah prioritas tinggi, Anda dapat menggunakan Systems Manager OpsCenter untuk menjalankan runbook Otomasi dan menyelesaikan masalah tersebut dengan cepat. Untuk informasi lihat,	18 November 2019

AWS Systems ManagerExplorer.

> **Note**
>
> Pengaturan untuk Systems Manager OpsCenter terintegrasi dengan set up forExplorer. Jika Anda sudah menyiapkanOpsCenter, Anda masih perlu menyelesaikan Penyiapan Terpadu untuk memverifikasi pengaturan dan opsi. Jika Anda belum menyiapkanOpsCenter, maka Anda dapat menggunakan Pengaturan Terpadu untuk memulai dengan kedua kemampuan tersebut. Untuk informasi selengkapnya, lihat Memulai dengan Explorer dan OpsCenter.

Peningkatan kemampuan pencarian parameter	Alat untuk mencari parameter sekarang memudahkan untuk menemukan parameter ketika Anda memiliki sejumlah besar parameter di akun Anda atau ketika Anda tidak ingat nama parameter yang tepat. Dengan alat pencarian, Anda mem-filter berdasarkan `contains`. Sebelumnya, alat pencarian mendukung pencarian nama parameter hanya dengan `equals` dan `begins-with` . Untuk informasi lebih lanjut, lihat Mencari parameter Systems Manager.	15 November 2019

Pembuat Dokumen Berbasis Konsol Baru untuk Otomatisasi | Support untuk menjalankan skrip dalam langkah-langkah Otomasi	Anda sekarang dapat menggunakan Systems Manager Automation untuk membangun dan berbagi buku pedoman operasional standar untuk memastikan konsistensi di seluruh pengguna, Akun AWS dan. Wilayah AWS Dengan kemampuan untuk menjalankan skrip ini dan menambahkan dokumentasi inline ke runbook Otomatisasi Anda menggunakan Markdown, Anda dapat mengurangi kesalahan dan menghilangkan langkah-langkah manual seperti menavigasi prosedur tertulis di wiki dan menjalankan perintah terminal. Untuk informasi selengkapnya, lihat topik berikut. • Walkthrough: Menggunakan Pembuat Dokumen untuk membuat runbook Otomasi khusus • `aws:executeScript` (Referensi tindakan otomatisasi) • Membuat runbook Otomasi menggunakan Document Builder	14 November 2019

	• Fitur Otomatisasi baru Dalam Systems Manager pada Blog Berita AWS	
Lakukan pembaruan paket di tempat menggunakan Distributor	Sebelumnya, ketika Anda ingin menginstal pembaruan ke paket menggunakanDistributor, satu-satunya pilihan Anda adalah menghapus seluruh paket dan menginstal ulang versi baru. Sekarang Anda dapat memilih untuk melakukan pembaruan di tempat sebagai gantinya. Selama pembaruan di tempat, Distributor instal hanya file yang baru atau diubah sejak instalasi terakhir, sesuai dengan skrip pembaruan yang Anda sertakan dalam paket Anda. Dengan opsi ini, aplikasi paket Anda dapat tetap tersedia dan tidak dijadikan offline selama pembaruan. Untuk informasi lebih lanjut, lihat topik berikut. • Buat paket • Instal atau perbarui paket	11 November 2019

Fitur pembaruan SSM Agent auto baru	Dengan satu klik, Anda dapat mengonfigurasi semua instance di Anda Akun AWS untuk secara otomatis memeriksa dan mengunduh versi baru. SSM Agent Untuk melakukannya, pilih Pembaruan otomatis agen di halaman Instans terkelola di AWS Systems Manager konsol. Untuk selengkap nya, lihat Mengotomatiskan pembaruan ke SSM Agent.	5 November 2019
Batasi Session Manager akses menggunakan tag AWS yang disediakan	Metode kedua untuk mengendalikan akses pengguna ke tindakan sesi sekarang tersedia. Dengan metode baru ini, Anda membuat kebijakan akses IAM menggunakan tag sesi yang disediakan AWS bukannya menggunak an variabel `{aws:user name}` . Menggunakan tag sesi AWS yang disediakan ini memungkinkan organisas i yang menggunakan ID federasi untuk mengontro l akses pengguna ke sesi. Untuk informasi, lihat Izinkan pengguna untuk mengakhir i hanya sesi yang mereka mulai.	2 Oktober 2019

Dokumen SSM Command baru untuk menerapkan Playbooks Ansible	Anda dapat membuat State Manager asosiasi yang menjalankan Ansible Playbooks dengan menggunakan `AWS-ApplyAnsiblePlaybooks` dokumen. Dokumen ini menawarkan manfaat berikut untuk menjalankan Playbook: • Support untuk menjalankan Playbook yang kompleks • Support untuk mengunduh Playbook dari GitHub dan Amazon Simple Storage Service (Amazon S3) • Support untuk struktur Playbook yang terkompresi • Pencatatan yang ditingkatkan • Kemampuan untuk menentukan Playbook mana yang akan dijalankan saat Playbook dipaketkan Untuk informasi selengkapnya, lihat Membuat asosiasi yang menjalankan buku Ansible pedoman	24 September 2019

Dukungan penerusan port untuk Session Manager	Session Managerse karang mendukung sesi penerusan port. Penerusan port memungkinkan Anda membuat terowongan dengan aman di antara instans yang di-deploy di subnet privat, tanpa perlu memulai layanan SSH di server, membuka port SSH di grup keamanan, atau menggunakan host bastion. Mirip dengan terowonga n SSH, penerusan port memungkinkan Anda untuk meneruskan lalu lintas antara laptop Anda ke port terbuka pada instans Anda. Setelah penerusan port dikonfigurasi, Anda dapat terhubung ke port lokal dan mengakses aplikasi server yang berjalan di dalam instans. Untuk informasi selengkapnya, lihat topik berikut: • Port Forwarding Menggunak an AWS Systems ManagerSession Manager di Blog Berita AWS • Memulai sesi (port forwardin g)	29 Agustus 2019

Tentukan tingkat parameter default atau otomatiskan pemilihan tingkat	Sekarang Anda dapat menentukan tingkat parameter default yang akan digunakan untuk permintaan untuk membuat atau memperbar ui parameter yang tidak menentukan tingkat. Anda dapat mengatur tingkat default ke parameter standar, parameter lanjutan, atau opsi baru, Intelligent-Tiering. Intelligent-Tiering mengevalu asi setiap PutParameter permintaan dan membuat parameter lanjutan hanya jika diperlukan. (Parameter lanjutan diperlukan jika ukuran nilai parameter lebih dari 4 KB, kebijakan parameter dikaitkan dengan parameter, atau maksimum 10.000 parameter yang didukung untuk tingkat standar sudah dibuat.) Untuk informasi lebih lanjut tentang menentukan tingkat default dan menggunakan Intellige nt-Tiering, lihat Menentukan tingkat parameter default.	27 Agustus 2019

Bekerja dengan bagian asosiasi diperbarui dengan CLI dan prosedur PowerShell	Bagian Bekerja dengan Asosiasi telah diperbarui untuk memasukkan dokumentasi prosedural untuk mengelola asosiasi menggunakan AWS CLI or AWS Tools for PowerShell. Untuk informasi, lihat Bekerja dengan asosiasi di Systems Manager.	26 Agustus 2019
Bekerja dengan bagian eksekusi Otomasi diperbarui dengan PowerShell CLI dan prosedur	Bagian Bekerja dengan Eksekusi Otomasi telah diperbarui untuk menyertakan dokumentasi prosedural untuk menjalankan alur kerja Otomasi menggunakan or. AWS CLI AWS Tools for PowerShell Untuk informasi lihat, Bekerja dengan Eksekusi Otomatisasi.	20 Agustus 2019
OpsCenterterintegrasi dengan wawasan aplikasi	OpsCenterterintegrasi dengan Amazon CloudWatch Application Insights untuk .NET dan SQL Server. Ini berarti Anda dapat secara otomatis membuat OpsItems masalah yang terdeteksi dalam aplikasi Anda. Untuk informasi tentang cara mengonfigurasi Wawasan Aplikasi untuk dibuatOpsItems, lihat Menyiapkan, mengonfigurasi, dan mengelola aplikasi untuk pemantauan di Panduan CloudWatch Pengguna Amazon.	7 Agustus 2019

Fitur konsol baru: AWS Systems ManagerQuick Setup	Quick Setupadalah fitur baru di konsol Systems Manager yang membantu Anda mengonfig urasi beberapa komponen Systems Manager dengan cepat pada instans EC2 Anda. Khususnya, Quick Setup membantu Anda mengkonfi gurasi komponen-komponen berikut pada instans yang Anda pilih atau targetkan dengan menggunakan tag: • Peran profil instans AWS Identity and Access Management (IAM) untuk Systems Manager. • Pembaruan dua bulanan yang dijadwalkan. SSM Agent • Pengumpulan metadata Inventaris terjadwal setiap 30 menit. • Pemindaian harian instans Anda untuk mengidentifikasi patch yang hilang. • Instalasi dan konfigurasi satu kali dari CloudWatch agen Amazon. • Pembaruan CloudWatch agen yang dijadwalkan dan bulanan.	7 Agustus 2019

Untuk informasi lebih lanjut, lihat AWS Systems Manager Quick Setup.

Daftarkan grup sumber daya sebagai target jendela pemeliharaan	Selain mendaftarkan instance terkelola sebagai target jendela pemeliharaan, Anda sekarang dapat mendaftarkan grup sumber daya sebagai target jendela pemeliharaan. Maintenance Windowsmendukung semua jenis AWS sumber daya yang didukung oleh AWS Resource Groups termasuk`AWS::EC2::Instance` ,,`AWS::DynamoDB::Table` , `AWS::OpsWorks::Instance` `AWS::Redshift::Cluster` , dan banyak lagi. Dengan rilis ini Anda juga dapat mengirim perintah ke grup sumber daya, misalnya dengan menggunakan Run Command konsol atau AWS CLI send-commandperintah. Untuk informasi selengkapnya, lihat topik berikut: • Tetapkan target ke jendela pemeliharaan (konsol) • Contoh: Daftarkan target dengan jendela pemeliharaan • Menggunakan target dan kontrol tingkat untuk mengirim perintah ke armada	23 Juli 2019

Pembuatan dan pembuatan versi paket yang disederhanakan dengan AWS Systems ManagerDistributor	Distributormemiliki alur kerja pembuatan paket baru yang disederhanakan yang dapat menghasilkan manifes paket, skrip, dan hash file untuk Anda. Anda juga dapat menggunakan alur kerja yang disederhanakan saat Anda menambahkan versi ke paket yang sudah ada.	22 Juli 2019
Panel kategori dokumen baru untuk Automasi Systems Manager	Systems Manager mencakup panel kategori Dokumen baru ketika Anda menjalankan Otomatisasi di konsol. Gunakan panel ini untuk memfilter runbook Otomatisasi berdasarkan tujuan mereka.	18 Juli 2019

Verifikasi izin pengguna untuk mengakses dokumen Session Manager konfigurasi default	Ketika pengguna di akun Anda menggunakan AWS CLI untuk memulai Session Manager sesi dan tidak menentukan dokumen konfigurasi dalam perintah, Systems Manager menggunakan dokumen konfigurasi default`SSM-SessionManagerRunShell` . Anda sekarang dapat memverifikasi bahwa pengguna telah diberikan izin untuk mengakses dokumen ini dengan menambahkan elemen kondisi `ssm:SessionDocumentAccessCheck` untuk kebijakan untuk dokumen.AWS Identity and Access Management (IAM) entitas (pengguna, grup, atau peran). Untuk informasi, lihat Melaksanakan pemeriksaan izin dokumen untuk skenario CLI default.	9 Juli 2019

Support untuk memulai Session Manager sesi menggunakan kredensi pengguna sistem operasi	Secara default, Session Manager sesi diluncurkan menggunakan kredensya l `ssm-user` akun yang dihasilkan sistem yang dibuat pada instance terkelola. Pada mesin Linux, Anda sekarang dapat meluncurkan sesi menggunakan kredensial akun sistem operasi. Untuk informasi, lihat Mengaktif kan dukungan Run As untuk instans Linux.	9 Juli 2019
Support untuk memulai Session Manager sesi menggunakan SSH	Anda sekarang dapat menggunakan AWS CLI untuk memulai sesi SSH pada instance terkelola menggunak anSession Manager. Untuk informasi tentang mengizink an sesi SSHSession Manager, lihat (Opsional) Mengaktifkan sesi SSH Session Manager. Untuk informasi tentang memulai sesi SSH menggunak anSession Manager, lihat Memulai sesi (SSH).	9 Juli 2019

Support untuk mengubah kata sandi pada instans terkelola	Anda sekarang dapat mengatur ulang kata sandi pada mesin yang Anda kelola menggunakan Systems Manager (instans terkelola). Anda dapat mengatur ulang kata sandi dengan menggunakan konsol Systems Manager atau AWS CLI. Untuk informasi, lihat Menyetel ulang kata sandi pada instans terkelola.	9 Juli 2019
Revisi untuk “Apa itu AWS Systems Manager?”	Konten pengenalan dalam Apa yang dimaksud AWS Systems Manager? telah diperluas untuk memberikan pengenalan yang lebih luas untuk layanan dan mencermin kan kemampuan Systems Manager yang telah dirilis baru-baru ini. Selain itu, konten lain di bagian ini telah dipindahkan ke topik individual agar dapat ditemukan dengan lebih baik.	10 Juni 2019

Kemampuan Systems Manager baru: OpsCenter	OpsCentermenyediakan lokasi pusat di mana insinyur operasi dan profesional TI dapat melihat, menyelidiki, dan menyelesaikan item kerja operasional (OpsItems) yang terkait dengan AWS sumber daya. OpsCenter dirancang untuk mengurangi waktu rata-rata untuk menyelesaikan masalah yang memengaruhi AWS sumber daya. Kemampuan Systems Manager ini mengumpulkan dan menstandarisasi OpsItems seluruh layanan sambil memberikan data investigasi kontekstual tentang masing-masing sumber daya terkaitOpsItem, dan terkaitOpsItems. OpsCenter juga menyediakan runbook Systems Manager Automation yang dapat Anda gunakan untuk menyelesaikan masalah dengan cepat. Anda dapat menentukan data kustom yang dapat dicari untuk masing-masing data. OpsItem Anda juga dapat melihat laporan ringkasan yang dibuat secara otomatis berdasarkan status dan sumber. OpsItems Untuk informasi selengkapnya, lihat AWS Systems ManagerOpsCenter.	6 Juni 2019

Perubahan pada panel navigasi kiri Systems Manager di AWS Management Console	Systems Manager meninggalkan panel navigasi di judul baru AWS Management Console termasuk, termasuk judul baru untuk Ops Center, yang menyediakan pengelompokan kemampuan Systems Manager yang lebih logis.	6 Juni 2019

Tutorial yang direvisi untuk membuat dan mengkonfigurasi jendela pemeliharaan menggunakan AWS CLI	Tutorial: Membuat dan mengkonfigurasi jendela pemeliharaan (AWS CLI) telah dirombak untuk menyediakan jalur yang sederhana melalui langkah-langkah praktek. Anda membuat satu jendela pemeliharaan, mengidentifikasi satu target, dan mengatur tugas sederhana untuk jendela pemeliharaan untuk dijalankan. Sepanjang jalan, kami menyediakan informasi dan contoh yang dapat Anda gunakan untuk membuat perintah pendaftaran tugas Anda sendiri, termasuk informasi untuk menggunakan parameter semu seperti `{{TARGET_ID}}` . Untuk informasi selengkapnya dan contoh, lihat topik berikut ini: • Contoh: Daftarkan target dengan jendela pemeliharaan • Contoh: Daftarkan tugas dengan jendela pemeliharaan • Tentang register-task-with-maintenance opsi -windows • Menggunakan parameter semu saat mendaftarkan tugas jendela pemeliharaan	31 Mei 2019

Pemberitahuan tentang SSM Agent pembaruan	Untuk diberi tahu tentang SSM Agent pembaruan, berlangganan halaman Catatan SSM Agent Rilis diGitHub.	24 Mei 2019
Menerima pemberitahuan atau memicu tindakan berdasarkan perubahan Parameter Store	Topik Menyiapkan notifikasi atau memicu tindakan berdasarkan Parameter Store peristiwa kini membantu Anda mengatur EventBridge aturan Amazon untuk merespons perubahan Parameter Store. Anda dapat menerima notifikasi atau memicu tindakan lain ketika salah satu hal berikut terjadi: • parameter dibuat, diperbarui, atau dihapus. • Label versi parameter dibuat, diperbarui, atau dihapus. • parameter berakhir, akan berakhir, atau tidak berubah dalam jangka waktu tertentu.	22 Mei 2019

Revisi besar untuk menyiapkan dan memulai konten	Kami telah memperluas dan menata ulang konten Menyiapkan dan Memulai di Panduan Pengguna AWS Systems Manager . Konten Menyiapkan telah dibagi menjadi dua bagian. Satu bagian berfokus pada tugas untuk menyiapkan Systems Manager untuk mengkonfigurasi dan mengelola instans EC2 Anda. Yang lain berfokus pada tugas untuk menyiapkan Systems Manager untuk mengkonfigurasi dan mengelola server on-premise dan mesin virtual (VM) di lingkungan hibrid. Kedua bagian sekarang menyajikan semua topik penyiapan sebagai langkah bernomor utama, dalam urutan penyelesaian yang direkomendasikan. Bab Memulai yang baru berfokus pada membantu pengguna akhir memulai dengan Systems Manager setelah tugas konfigurasi akun dan layanan telah selesai. • Menyiapkan AWS Systems Manager • Menyiapkan AWS Systems Manager untuk lingkungan hybrid	15 Mei 2019

- Memulai dengan AWS Systems Manager

Sertakan patch untuk aplikasi yang dirilis oleh Microsoft di patch baseline (Windows)	Patch Managersekarang mendukung pembaruan patch untuk aplikasi yang dirilis oleh Microsoft pada Windows Server instance. Sebelumnya, hanya tambalan untuk sistem Windows Server operasi yang didukung. Patch Managermenyediakan dua baseline patch yang telah ditentukan untuk instance. Windows Server Dasar patch `AWS-WindowsPredefinedPatchBaseline-OS` hanya berlaku untuk patch sistem operasi. `AWS-WindowsPredefinedPatchBaseline-OS-Applications` berlaku untuk sistem operasi dan aplikasi Windows Server yang dirilis oleh Microsoft pada Windows. Untuk informasi tentang cara membuat dasar patch kustom yang mencakup patch untuk aplikasi yang dirilis oleh Microsoft, lihat prosedur pertama di Membuat dasar patch kustom. Juga, sebagai bagian dari pembaruan ini, nama-nama baseline patch AWS yang telah ditentukan -provided sedang diubah. Untuk informasi lebih lanjut, lihat Baseline yang telah ditetapkan.	7 Mei 2019

Contoh untuk mendaftarkan target jendela pemeliharaan menggunakan AWS CLI	Topik baru Contoh: Mendaftarkan target dengan jendela pemeliharaan menyediakan tiga contoh perintah untuk menunjukkan cara yang berbeda Anda dapat menentukan target untuk jendela pemeliharaan ketika Anda menggunakan AWS CLI. Topik ini juga menjelaskan kasus penggunaan terbaik untuk masing-masing contoh perintah.	3 Mei 2019

Pembaruan untuk menambal topik grup	Topik Tentang grup patch telah diperbarui untuk menyertakan bagian tentang cara instans terkelola menentukan dasar patch yang sesuai untuk digunakan selama operasi patching. Selain itu, petunjuk telah ditambahkan untuk menggunakan konsol AWS CLI atau Systems Manager untuk menambahkan Grup Patch atau PatchGrouptag ke instance terkelola Anda, dan cara menambahkan Grup Patch atau PatchGroupke baseline patch. (Anda harus menggunakan**`PatchGroup`** , tanpa spasi, jika Anda telah mengizinkan tag dalam metadata instans EC2.) Untuk informasi lebih lanjut, lihat Membuat grup patch dan Menambahkan grup patch ke dasar patch.	1 Mei 2019

Parameter StoreFitur baru	Parameter Storemenawarkan fitur-fitur baru berikut: • Parameter lanjutan: Parameter Store sekarang memungkinkan Anda mengonfigurasi parameter secara individual untuk menggunakan tingkat parameter standar (tingkat default) atau tingkat parameter lanjutan. Parameter lanjutan menawarkan kuota ukuran yang lebih besar untuk nilai parameter, kuota yang lebih tinggi untuk jumlah parameter yang dapat Anda buat per Akun AWS dan Wilayah AWS, dan kemampuan untuk menggunakan kebijakan parameter. Untuk informasi selengkapnya tentang parameter lanjutan, lihat Tentang parameter lanjutan Systems Manager. • Kebijakan parameter : Kebijakan parameter membantu Anda mengelola serangkaian parameter yang berkembang dengan memungkinkan Anda menetapkan kriteria tertentu ke sebuah parameter, seperti tanggal kedaluwar	25 April 2019

	sa atau waktu untuk tayang. Kebijakan parameter sangat membantu dalam memaksa Anda memperbarui atau menghapus kata sandi dan data konfigurasi yang tersimpan di dalamnyaP arameter Store. Kebijakan parameter hanya tersedia untuk parameter yang menggunakan tingkat parameter lanjutan. Untuk informasi lebih lanjut, lihat Bekerja dengan kebijakan parameter. • Throughput yang lebih tinggi: Anda sekarang dapat meningkatkan kuota Parameter Store throughput hingga maksimal 1.000 transaksi per detik. Untuk informasi selengkap nya, lihat Meningkatkan Parameter Store throughput.	
Pembaruan ke bagian Otomasi	Bagian Otomatisasi telah diperbarui agar lebih mudah ditemukan. Selain itu, tiga topik baru telah ditambahkan ke bagian Otomatisasi: • Jalankan otomatisasi secara manual • Jalankan otomatisasi dengan pemberi persetujuan • Penjadwalan otomatisasi	17 April 2019

Enkripsi data sesi menggunakan kunci AWS KMS	Secara default, Session Manager menggunakan TLS 1.2 untuk mengenkripsi data sesi yang dikirimkan antara mesin lokal pengguna di akun Anda dan instans EC2 Anda. Sekarang Anda dapat memilih untuk mengenkripsi lebih lanjut data tersebut menggunakan data AWS KMS key yang telah dibuat di AWS Key Management Service. Anda dapat menggunakan kunci KMS yang telah dibuat di Akun AWS Anda atau yang telah dibagikan dengan Anda dari akun lain. Untuk informasi tentang menentukan kunci KMS untuk mengenkripsi data sesi, lihat Mengaktifkan enkripsi AWS KMS kunci data sesi (konsol), Membuat Session Manager preferensi (), atau Memperbarui Session Manager preferensi (AWS CLI).AWS CLI	4 April 2019

Mengonfigurasi notifikasi Amazon SNS untuk AWS Systems Manager	Menambahkan instruksi untuk menggunakan konsol AWS CLI atau Systems Manager untuk mengonfigurasi notifikasi Amazon SNS Run Command dan Run Command tugas yang terdaftar ke jendela pemeliharaan. Untuk informasi lebih lanjut lihat Mengkonfigurasi notifikasi Amazon SNS untuk AWS Systems Manager.	6 Maret 2019

Instans lanjutan untuk server dan VM di lingkungan hybrid	AWS Systems Manager menawarkan tingkat instans standar dan tingkat instans lanjutan untuk server dan VM di lingkungan hybrid Anda. Tingkat instans standar memungkinkan Anda mendaftarkan maksimum 1.000 server atau VM per per per server. Akun AWS Wilayah AWS Jika Anda perlu mendaftarkan lebih dari 1.000 server atau VM dalam satu akun dan Wilayah, gunakan tingkat instans lanjutan. Anda dapat membuat instance sebanyak yang Anda suka di tingkat instans lanjutan, tetapi semua instance yang dikonfigu rasi untuk Systems Manager tersedia berdasarkan satu basis. pay-per-use Instans lanjutan juga memungkin kan Anda untuk terhubung ke mesin hybrid Anda dengan menggunakan AWS Systems Manager Session Manager. Session Managerme nyediakan akses shell interakti f ke instance Anda. Untuk informasi selengkapnya tentang mengizinkan instans lanjutan, lihat Menggunakan tingkat instans lanjutan.	4 Maret 2019

Buat State Manager asosiasi yang menggunakan dokumen SSM bersama	Anda dapat membuat State Manager asosiasi yang menggunakan runbook SSM Command and Automatio n yang dibagikan dari yang lain. Akun AWS Membuat asosiasi dengan menggunak an dokumen SSM bersama membantu untuk menjaga Amazon EC2 dan infrastruktur hibrid Anda dalam keadaan konsisten bahkan ketika instans tidak berada dalam akun yang sama. Untuk informasi tentang berbagi dokumen SSM, lihat Dokumen AWS Systems Manager. Untuk informasi tentang membuat State Manager asosiasi, lihat Membuat asosiasi.	28 Februari 2019
Melihat daftar peristiwa Systems Manager yang didukung untuk EventBridge aturan Amazon	Topik baru Acara Monitorin g Systems Manager dengan Amazon EventBridge memberikan ringkasan berbagai peristiwa yang dipancarkan oleh Systems Manager tempat Anda dapat mengatur aturan pemantauan acara. EventBridge	25 Februari 2019

Menambahkan tag saat Anda membuat sumber daya Systems Manager	Systems Manager sekarang mendukung kemampuan untuk menambahkan tag ke jenis sumber daya tertentu ketika Anda membuatnya. Sumber daya yang dapat Anda beri tag saat membuatnya dengan AWS CLI atau SDK mencakup jendela pemeliharaan, garis dasar tambalan, Parameter Store parameter, dan dokumen SSM. Anda juga dapat menetapkan tag ke instans terkelola saat Anda membuat aktivasi untuk itu. Ketika Anda menggunakan konsol Systems Manager, Anda dapat menambahkan tag ke jendela pemeliharaan, dasar patch, dan parameter.	24 Februari 2019

Pembuatan peran IAM otomatis untuk Systems Manager Inventory	Sebelumnya Anda harus membuat peran AWS Identity and Access Managemen t (IAM) dan melampirkan kebijakan terpisah ke peran ini untuk melihat data inventari s di halaman Tampilan Detail Inventaris di konsol. Anda tidak perlu lagi membuat peran ini atau melampirkan kebijakan. Saat Anda memilih Sinkronisasi Data Jarak Jauh di halaman Tampilan Detail Inventaris, Systems Manager secara otomatis membuat `Amazon-GlueService PolicyForSSM` peran tersebut dan menetapkan kebijakan GlueServicePolicyF orAmazon-SSM- {S3 bucket name} dan AWSGlueSe rviceRolekebijakan tersebut. Untuk informasi lebih lanjut, lihat Mengkueri data inventari s dari beberapa Region dan akun.	14 Februari 2019

Maintenance Windowspanduan untuk memperbarui SSM Agent	Menambahkan dua panduan baru ke dokumentasi. Maintenance Windows Penelusuran ini merinci cara menggunakan konsol Systems Manager atau AWS CLI untuk membuat jendela pemeliharaan yang disimpan secara otomatis. SSM Agent up-to-date Untuk informasi lebih lanjut, lihat Maintenance Windowspanduan.	11 Februari 2019
Menggunakan parameter Parameter Store publik	Ditambahkan bagian pendek yang menjelaskan parameter Parameter Store publik. Untuk informasi lebih lanjut, lihat Menggunakan parameter publik Systems Manager.	31 Januari 2019
Gunakan AWS CLI untuk membuat Session Manager preferensi	Menambahkan petunjuk penggunaan AWS CLI untuk membuat Session Manager preferensi, seperti CloudWatch Log, opsi pencatatan bucket S3, dan setelan enkripsi sesi. Untuk informasi selengkapnya, lihat Menggunakan AWS CLI untuk membuat Session Manager preferensi.	22 Januari 2019

Menjalankan alur kerja otomatisasi Systems Manager dengan menggunakan State Manager	AWS Systems Manager State Managersekarang mendukung pembuatan asosiasi yang menggunakan runbook Otomasi SSM. State Managersebelumnya hanya didukung `command` dan `policy` dokumen, yang berarti bahwa Anda hanya dapat membuat asosiasi yang menargetkan instance terkelola. Dengan dukungan untuk runbook Otomatisasi SSM, Anda sekarang dapat membuat asosiasi yang menargetkan berbagai jenis sumber daya AWS . Untuk informasi selengkapnya, lihat Menjalankan alur kerja Otomasi Systems Manager menggunakan. State Manager	22 Januari 2019

Pembaruan referensi untuk ekspresi cron dan rate serta opsi penjadwalan jendela pemeliharaan	Topik referensi Ekspresi cron dan rate untuk Systems Manager telah direvisi. Versi baru memberikan lebih banyak contoh dan penjelasan yang lebih baik tentang cara menggunakan ekspresi cron dan rate untuk menjadwalkan jendela dan State Manager asosiasi pemeliharaan Anda. Selain itu, Maintenance Windowspenjadwalan topik baru dan opsi periode aktif menjelaskan bagaimana berbagai opsi terkait jadwal untuk jendela pemeliharaan (Tanggal mulai, tanggal akhir, zona waktu, frekuensi Jadwal) berhubungan satu sama lain.	6 Desember 2018
Aktifkan SSM Agent logging debug	Anda dapat mengaktifkan logging SSM Agent debug dengan mengedit file seelog.xml.template pada instance terkelola. Untuk informasi selengkapnya, lihat Mengaktifkan logging SSM Agent debug.	30 November 2018

Support untuk arsitektur prosesor ARM64	AWS Systems Manager sekarang mendukung versi ARM64 dari sistem operasi Amazon Linux 2, Red Hat Enterprise Linux 7.6, dan Ubuntu Server (18.04 LTS dan 16.04 LTS). Untuk informasi selengkapnya, lihat petunjuk untuk menginstal Amazon Linux 2, RHEL, dan Ubuntu Server18.04 dan 16.04 LTS dengan paket Snap. Untuk informasi selengkapnya tentang tipe instans A1, lihat Instans tujuan umum dalam Panduan Pengguna Amazon EC2 untuk Instans Linux.	26 November 2018

Membuat dan menyebarkan paket dengan menggunakan AWS Systems ManagerDistributor	Menggunakan AWS Systems Manager Distributor, Anda mengemas perangkat lunak Anda sendiri—atau menemukan paket perangkat lunak agen AWS yang disediakan, seperti `AmazonCloudWatchAgent` —untuk menginstal pada instance terkelola. AWS Systems Manager Distributormenerbitkan sumber daya, seperti paket perangkat lunak, ke instance yang AWS Systems Manager dikelola. Menerbitkan paket mengiklankan versi tertentu dari dokumen paket—dokumen Systems Manager yang Anda buat saat menambahkan paket Distributor di —ke instance terkelola yang Anda identifikasi berdasarkan ID instans terkelola Akun AWS , ID, tag, atau sebuah. Wilayah AWS Untuk informasi selengkapnya, lihat AWS Systems ManagerDistributor.	20 November 2018

Menjalankan alur kerja AWS Systems Manager Otomasi secara bersamaan di beberapa Wilayah AWS dan Akun AWS dari akun pusat	Anda dapat menjalankan alur kerja AWS Systems Manager otomatisasi secara bersamaan di beberapa Wilayah AWS dan Akun AWS atau Unit AWS Organisasi (OU) dari akun manajemen Otomasi. Secara bersamaan menjalankan Otomatisasi di beberapa Region dan akun atau OU untuk mengurangi waktu yang diperlukan untuk mengelola sumber daya AWS Anda sambil meningkatkan keamanan lingkungan komputasi Anda. Untuk informasi selengkapnya, lihat Menjalankan alur kerja Otomasi dalam beberapa Wilayah AWS dan. Akun AWS	19 November 2018

Kueri data inventaris dari beberapa Wilayah AWS dan Akun AWS	Systems Manager Inventory terintegrasi dengan Amazon Athena untuk membantu Anda menanyakan data inventaris dari Wilayah AWS beberapa dan. Akun AWS Integrasi Athena menggunakan sinkronisasi data sumber daya sehingga Anda dapat melihat data inventaris dari semua instans terkelola di halaman Tampilan Detail Inventaris di konsol. AWS Systems Manager Untuk informasi lebih lanjut lihat Mengkueri data Inventaris dari beberapa Region dan akun.	15 November 2018

Buat State Manager asosiasi yang menjalankan file MOF	Anda dapat menjalankan file Managed Object Format (MOF) untuk menerapkan status yang ditargetkan pada instance terkelola Windows Server State Manager dengan menggunakan dokumen SSM. `AWS-ApplyDSCMofs` Dokumen `AWS-Apply DSCMofs` memiliki dua mode eksekusi. Dengan mode pertama, Anda dapat mengonfigurasi asosiasi untuk memindai dan melaporkan jika instance terkelola saat ini berada dalam status target yang ditentukan dalam file MOF yang ditentukan. Dalam modus kedua, Anda dapat menjalankan file MOF dan mengubah konfigurasi instans Anda berdasarkan sumber daya dan nilai-nilai mereka yang didefinisikan dalam file MOF. Dokumen `AWS-ApplyDSCMofs` mengizinkan Anda untuk mengunduh dan menjalankan file konfigurasi MOF dari Amazon Simple Storage Service (Amazon S3), berbagi lokal, atau dari situs web aman dengan domain HTTPS. Untuk informasi selengkapnya, lihat Membuat asosiasi yang menjalankan file MOF.	15 November 2018

Batasi akses administratif dalam sesi Session Manager	Session Managersesi diluncurkan menggunakan kredensil akun pengguna yang dibuat dengan root default atau izin administrator yang disebut. `ssm-user` Informasi tentang membatasi kontrol administratif untuk akun ini sekarang tersedia dalam topik Mengaktifkan atau menonaktifkan izin administratif akun ssm-user.	13 November 2018
Contoh YAMG dalam referensi tindakan Otomasi	Referensi tindakan Otomatisasi sekarang mencakup sampel YAML untuk setiap tindakan yang sudah mencakup sampel JSON.	31 Oktober 2018
Tetapkan tingkat keparahan kepatuhan ke asosiasi	Anda sekarang dapat menetapkan tingkat keparahan kepatuhan ke State Manager asosiasi. Tingkat kepelikan ini dilaporkan di Dasbor Kepatuhan dan juga dapat digunakan untuk mem-filter laporan kepatuhan Anda. Tingkat kepelikan yang dapat Anda tetapkan mencakup Kritis, Tinggi, Medium, Rendah, dan Tidak Ditentukan. Untuk informasi selengkapnya, lihat Membuat asosiasi (konsol).	26 Oktober 2018

Gunakan target dan kontrol nilai dengan Otomasi dan State Manager	Kontrol eksekusi Otomasi dan State Manager asosiasi di seluruh armada sumber daya Anda dengan menggunakan target, konkurensi, dan ambang kesalahan. Untuk informasi selengkapnya, lihat Menggunakan target dan kontrol kecepatan untuk menjalankan alur kerja Otomasi pada armada dan Menggunakan target dan kontrol tarif dengan State Manager asosiasi.	23 Oktober 2018
Tentukan rentang waktu aktif dan zona waktu internasional untuk jendela pemeliharaan	Anda juga dapat menentukan tanggal agar jendela pemeliharaan tidak dijalankan sebelum atau sesudahnya (tanggal mulai dan tanggal akhir), dan Anda dapat menentukan zona waktu internasional untuk mendasarkan jadwal jendela pemeliharaan. Untuk informasi lebih lanjut lihat Membuat jendela pemeliharaan (konsol) dan Memperbarui jendela pemeliharaan (AWS CLI).	9 Oktober 2018

Pertahankan daftar patch khusus untuk baseline patch Anda di bucket S3	Dengan parameter 'InstallOverrideList' baru di dokumen perintah SSM`AWS-RunPatchBaseline`, Anda dapat menentukan URL https atau URL gaya jalur Amazon Simple Storage Service (Amazon S3) ke daftar tambalan yang akan diinstal. Daftar instalasi patch ini, yang Anda pertahankan dalam sebuah bucket S3 dalam format YAML, menggantikan patch yang ditentukan oleh dasar patch default saat ini. Untuk informasi selengkapnya, lihat Nama parameter: InstallOverrideList.	5 Oktober 2018
Kontrol yang diperluas atas apakah dependensi patch diinstal	Sebelumnya, jika sebuah patch di daftar path yang Ditolak Anda diidentifikasi sebagai dependensi dari patch lain, itu akan tetap diinstal. Sekarang Anda dapat memilih apakah akan menginstal dependensi ini atau memblokirnya agar tidak diinstal. Untuk informasi lebih lanjut, lihat Membuat dasar patch.	5 Oktober 2018

Buat alur kerja otomatisasi dinamis dengan percabangan bersyarat	Tindakan Otomatisasi `aws:branch` mengizinkan Anda membuat alur kerja Otomatisasi dinamis yang mengevaluasi pilihan yang berbeda dalam satu langkah dan kemudian melompat ke langkah berbeda di runbook Otomatisasi berdasarkan hasil evaluasi tersebut. Untuk informasi selengkapnya, lihat Menggunakan pernyataan bersyarat di buku runbook.	26 September 2018
Gunakan AWS CLI untuk memperbarui Session Manager preferensi	Petunjuk penggunaan CLI untuk memperbarui Session Manager preferensi, seperti opsi pencatatan bucket CloudWatch Log dan S3, telah ditambahkan ke Panduan Pengguna.AWS Systems Manager Untuk selengkapnya, lihat Menggunakan Session Manager preferensi AWS CLI untuk memperbarui.	25 September 2018
SSM AgentPersyaratan yang diperbarui untuk Session Manager	Session Managersekarang membutuhkan SSM Agent versi 2.3.68.0 atau yang lebih baru. Untuk informasi lebih lanjut tentang Session Manager prasyarat, lihat Prasyarat lengkap. Session Manager	17 September 2018

Kelola instance tanpa membuka port masuk atau mempertahankan host bastion menggunakan Session Manager	Dengan menggunakanSession Manager, kemampuan yang dikelola sepenuhnya AWS Systems Manager, Anda dapat mengelola instans EC2 Anda melalui shell berbasis browser satu klik interaktif atau melalui file. AWS CLISessio n Managermenyediakan manajemen instans yang aman dan dapat diaudit tanpa perlu membuka port masuk, memelihara host bastion, atau mengelola kunci SSH. Session Managerjuga memungkin kan Anda untuk mematuhi kebijakan perusahaan yang memerlukan akses terkontrol ke instans, praktik keamanan yang ketat, dan log yang dapat diaudit sepenuhnya dengan detail akses instans, sambil tetap memberikan pengguna akhir akses lintas platform satu klik sederhana ke instans EC2 Anda. Untuk informasi selengkapnya, lihat Pelajari lebih lanjut Session Manager.	11 September 2018

Memanggil yang lain Layanan AWS dari alur kerja Otomasi Systems Manager	Anda dapat memanggil kemampuan Systems Manager lainnya Layanan AWS dan lainnya dalam alur kerja Otomasi Anda dengan menggunakan tiga tindakan Otomasi (atau plugin) baru di runbook Otomasi Anda. Untuk informasi selengkapnya, lihat Untuk informasi selengkap nya, lihat Menggunakan output tindakan sebagai input.	28 Agustus 2018
Gunakan kunci kondisi khusus Manajer Sistem dalam kebijakan IAM	Topik Menentukan syarat dalam kebijakan telah diperbarui untuk daftar kunci syarat IAM untuk Systems Manager yang dapat Anda masukkan dalam kebijakan . Anda dapat menggunakan kunci ini untuk menentukan syarat pemberlakuan suatu kebijakan. Topik ini juga mencakup tautan ke contoh kebijakan dan topik terkait lainnya.	18 Agustus 2018

Agregat data inventaris dengan grup untuk melihat instance mana yang dikonfigurasi dan tidak dikonfigurasi untuk mengumpulkan jenis inventaris	Grup memungkinkan Anda dengan cepat melihat jumlah instans terkelola yang dikonfigurasi dan yang tidak dikonfigurasi untuk mengumpulkan satu atau beberapa jenis Inventaris. Dengan grup, Anda menentukan satu atau beberapa jenis Inventaris dan filter yang menggunakan operator `exists`. Untuk informasi lebih lanjut, lihat Menggabungkan data Inventaris.	16 Agustus 2018
Lihat riwayat dan pelacakan perubahan untuk Kepatuhan Inventaris dan Konfigurasi	Sekarang Anda dapat melihat riwayat dan mengubah pelacakan untuk Inventaris yang dikumpulkan dari instans terkelola Anda. Anda juga dapat melihat riwayat dan mengubah pelacakan untuk Patch Manager tambalan dan State Manager asosiasi yang dilaporkan oleh Kepatuhan Konfigurasi. Untuk informasi lebih lanjut, lihat Melihat riwayat Inventaris dan pelacakan perubahan.	9 Agustus 2018

Parameter Storeterintegrasi dengan Secrets Manager	Parameter Storesekarang terintegrasi dengan AWS Secrets Manager sehingga Anda dapat mengambil rahasia Secrets Manager saat menggunakan rahasia lain Layanan AWS yang sudah mendukung referensi ke Parameter Store parameter . Layanan ini mencakup Amazon EC2, Amazon Elastic Container Service,, AWS Lambda,, AWS CloudForm ation, AWS CodeBuild AWS CodeDeploy, dan kemampuan Systems Manager lainnya. Dengan menggunakan Parameter Store untuk mereferensikan rahasia Secrets Manager, Anda membuat proses yang konsisten dan aman untuk memanggil dan menggunak an rahasia dan data referensi dalam kode dan skrip konfigurasi Anda. Untuk selengkapnya, lihat Merujuk AWS Secrets Manager rahasia dari Parameter Store parameter.	26 Juli 2018

Lampirkan label ke Parameter Store parameter	Label parameter adalah alias yang ditetapkan pengguna untuk membantu Anda mengelola versi parameter yang berbeda. Saat Anda memodifikasi sebuah parameter, Systems Manager secara otomatis menyimpan versi baru dan menambah nomor versi dengan satu. Label dapat membantu Anda mengingat tujuan dari sebuah versi parameter ketika ada beberapa versi. Untuk informasi, lihat Pelabelan parameter.	26 Juli 2018

Buat alur kerja Otomasi dinamis	Secara default, langkah-langkah (atau tindakan) yang Anda tentukan di bagian mainSteps runbook Otomatisasi dijalankan secara berurutan. Setelah satu tindakan selesai, tindakan berikutnya yang ditentukan dalam bagian mainSteps akan dimulai. Dengan rilis ini, Anda sekarang dapat membuat alur kerja Otomatisasi yang melakukan Percabangan bersyarat. Ini berarti bahwa Anda dapat membuat alur kerja Otomatisasi yang secara dinamis menanggapi perubahan kondisi dan melompat ke langkah tertentu. Untuk selengkapnya, lihat Menggunakan pernyataan bersyarat di buku runbook.	18 Juli 2018

SSM Agentsekarang sudah diinstal sebelumnya pada Ubuntu Server 16.04 AMIs menggunakan Snap	Dimulai dengan instance yang dibuat dari Ubuntu Server 16.04 yang AMIs diidentif ikasi`20180627`, SSM Agent sudah diinstal sebelumnya menggunakan paket Snap. Pada instans yang dibuat dari AMIs sebelumnya, Anda harus terus menggunakan paket penginstal deb. Untuk selengkapnya, lihat Tentang SSM Agent instalasi pada instance 64-bit Ubuntu Server 16.04.	7 Juli 2018
Tinjau izin S3 minimum yang diperlukan oleh SSM Agent	Topik baru Izin bucket Minimum S3 untuk SSM Agent memberikan informasi tentang bucket Amazon Simple Storage Service (Amazon S3) yang mungkin perlu diakses sumber daya untuk melakukan operasi Systems Manager. Anda dapat menentukan bucket ini dalam kebijakan kustom jika Anda ingin membatasi akses bucket S3 untuk profil instans atau VPC endpoint untuk minimum yang diperlukan untuk menggunakan Systems Manager.	5 Juli 2018

Melihat riwayat eksekusi lengkap untuk ID State Manager asosiasi tertentu	Topik baru Melihat riwayat asosiasi menjelaskan cara melihat semua eksekusi untuk ID asosiasi tertentu dan kemudian melihat detail eksekusi untuk satu atau lebih sumber daya.	2 Juli 2018
Patch Managermemperkenalkan dukungan untuk Amazon Linux 2	Anda sekarang dapat menggunakan Patch Manager untuk menerapkan tambalan ke instans Amazon Linux 2. Untuk informasi umum tentang dukungan sistem Patch Manager operasi, lihat Patch Managerprasyarat. Untuk informasi tentang pasangan nilai kunci yang didukung untuk Amazon Linux 2 saat mendefinisikan filter tambalan, lihat PatchFilterdi Referensi API.AWS Systems Manager	26 Juni 2018
Kirim output perintah ke Amazon CloudWatch Logs	Topik baru Mengonfigurasi CloudWatch Log Amazon untuk Run Command menjelaskan cara mengirim Run Command output ke CloudWatch Log.	18 Juni 2018

Buat atau hapus sinkronisasi data sumber daya untuk Inventaris dengan cepat menggunakan AWS CloudFormation	Anda dapat menggunakan AWS CloudFormation untuk membuat atau menghapus sinkronisasi data sumber daya untuk Systems Manager Inventory. Untuk menggunakan AWS CloudFormation, tambahkan sumber daya AWS::SSM::ResourceDataSync ke AWS CloudFormation template Anda. Untuk informasi selengkapnya, lihat Bekerja dengan Templat AWS CloudFormation dalam Panduan Pengguna AWS CloudFormation . Anda juga dapat secara manual membuat sinkronisasi data sumber daya untuk Inventaris seperti yang dijelaskan di Mengkonfigurasi sinkronisasi data sumber daya untuk Inventaris.	11 Juni 2018

AWS Systems Manager Pemberitahuan pembaruan Panduan Pengguna sekarang tersedia melalui RSS	Versi HTML Panduan Pengguna Systems Manager sekarang mendukung RSS feed pembaruan yang didokumentasikan di halaman Riwayat pembaruan Dokumentasi Systems Manager. RSS feed mencakup pembaruan yang dibuat pada bulan Juni 2018 dan setelahnya. Pembaruan yang diumumkan sebelumnya masih tersedia di halaman Riwayat pembaruan dokumentasi Systems Manager. Gunakan tombol RSS di panel menu atas untuk berlangganan feed.	6 Juni 2018
Tentukan kode keluar dalam skrip untuk me-reboot instance terkelola	Topik baru Mem-boot ulang instance terkelola dari skrip menjelaskan cara menginstruksikan Systems Manager untuk me-reboot instance terkelola dengan menentukan kode keluar dalam skrip yang Anda jalankan. Run Command	3 Juni 2018
Buat acara di Amazon EventBridge setiap kali inventaris kustom dihapus	Topik baru Melihat tindakan penghapusan inventaris di EventBridge menjelaskan cara mengonfigurasi Amazon EventBridge untuk membuat acara kapan saja pengguna menghapus Inventaris kustom.	1 Juni 2018

Pembaruan sebelum Juni 2018

Tabel berikut menjelaskan perubahan penting dalam setiap rilis Panduan Pengguna AWS Systems Manager sebelum Juni 2018.

Perubahan	Deskripsi	Tanggal rilis
Inventarisasi semua instans terkelola di Akun AWS	Anda dapat menginventarisasi semua instans terkelola Akun AWS dengan membuat asosiasi inventaris global. Untuk informasi selengkapnya, lihat Inventarisasi semua node terkelola di Akun AWS. **Note** Asosiasi inventaris global tersedia dalam SSM Agent versi 2.0.790.0 atau yang lebih baru. Untuk informasi tentang cara memperbarui SSM Agent instans Anda, lihatMemperbarui SSM Agent penggunaan Run Command.	3 Mei 2018
SSM Agentdiinstal secara default pada Ubuntu Server 18	SSM Agentdiinstal, secara default, pada Ubuntu Server 18.04 LTS 64-bit dan 32-bit. AMIs	2 Mei 2018
Topik baru	Topik baru Menjalankan perintah menggunakan versi dokumen tertentu menjelaskan cara menggunakan parameter versi dokumen untuk menentukan versi dokumen SSM yang digunakan ketika perintah berjalan.	1 Mei 2018
Topik baru	Topik baru Penghapusan inventaris kustom menjelaskan cara menghapus data Inventaris kustom dari Amazon S3 dengan menggunakan AWS CLI. Topik ini juga menjelaskan cara menggunakan `SchemaDeleteOption` untuk mengelola inventaris kustom dengan menonaktifkan atau menghapus jenis inventaris kustom. Fitur baru ini menggunakan operasi DeleteInventoryAPI.	19 April 2018

Perubahan	Deskripsi	Tanggal rilis
Notifikasi Amazon SNS untuk SSM Agent	Anda dapat berlangganan topik Amazon SNS untuk menerima pemberitahuan saat versi baru tersediaSSM Agent. Untuk informasi selengkapnya, lihat Berlangganan notifikasi SSM Agent.	9 April 2018
support patching CentOS	Systems Manager sekarang mendukung patching instans CentOS. Untuk informasi tentang versi CentOS yang didukung, lihat Prasyarat Patch Manager. Untuk informasi selengkapnya tentang cara kerja patching, lihat Bagaimana Patch Manager operasi bekerja.	29 Maret 2018
Bagian baru	Untuk menyediakan satu sumber untuk informasi referensi di Panduan Pengguna AWS Systems Manager , bagian baru telah diperkenalkan, AWS Systems Manager referensi . Konten tambahan akan ditambahkan ke bagian ini ketika sudah tersedia.	15 Maret 2018
Topik baru	Topik baru Tentang format nama paket untuk daftar patch yang disetujui dan ditolak mendetail format nama paket yang dapat Anda masukkan dalam daftar patch yang disetujui dan patch yang ditolak untuk dasar patch kustom. Format sampel disediakan untuk setiap jenis sistem operasi yang didukung olehPatch Manager.	9 Maret 2018
Topik baru	Systems Manager sekarang terintegrasi dengan Chef Chef InSpec. InSpec adalah kerangka kerja runtime open-sour ce yang memungkinkan Anda membuat profil yang dapat dibaca manusia di atau Amazon S3. GitHub Kemudian Anda dapat menggunakan Systems Manager untuk menjalankan pemindaian kepatuhan dan melihat instans yang patuh dan tidak patuh. Untuk informasi selengkapnya, lihat Menggunakan Chef InSpec profil dengan Kepatuhan Systems Manager.	7 Maret 2018

Perubahan	Deskripsi	Tanggal rilis
Topik baru	Topik baru Menggunakan peran terkait layanan untuk Systems Manager menjelaskan cara menggunakan peran terkait layanan AWS Identity and Access Management (IAM) dengan Systems Manager. Saat ini, peran terkait layanan hanya diperlukan bila menggunakan Inventaris Systems Manager untuk mengumpulkan metadata tentang tag dan Resource Groups.	27 Februari 2018

Perubahan	Deskripsi	Tanggal rilis
Topik baru dan diperbarui	Anda sekarang dapat menggunakan Patch Manager untuk menginstal tambalan yang berada di repositori sumber yang berbeda dari yang default yang dikonfigurasi pada instance. Ini berguna untuk menambal instance dengan pembaruan yang tidak terkait dengan keamanan; dengan konten Personal Package Archives (PPA) untukUbuntu Server; dengan pembaruan untuk aplikasi internal perusahaan; dan sebagainya. Anda menentukan repositori sumber patch alternatif ketika membuat dasar patch kustom. Untuk informasi selengkapnya, lihat topik berikut: • Cara menentukan repositori sumber patch alternatif (Linux) • Bekerja dengan dasar patch kustom • Buat dasar patch dengan repositori kustom untuk versi OS yang berbeda Selain itu, Anda sekarang dapat menggunakan Patch Manager untuk menambal SUSE Linux Enterprise Server instance. Patch Managermendukung patching SLES 12.* versi (64-bit saja). Untuk informasi lebih lanjut, lihat informasi spesifik SLES dalam topik berikut: • Cara pemilihan patch keamanan • Cara menginstal patch • Cara kerja aturan dasar patch pada SUSE Linux Enterprise Server	6 Februari 2018

Perubahan	Deskripsi	Tanggal rilis
Topik baru	Topik baru Memutakhirkan modul permintaan Python di Amazon Linux 1 instance yang menggunakan server proxy memberikan instruksi untuk memastikan bahwa instance yang dibuat menggunakan Amazon Linux 1 AMI telah diperbarui dengan versi modul `requests` Python saat ini. Persyaratan ini untuk memastikan kompatibilitas denganPatch Manager.	12 Januari 2018
Topik baru	Topik baru Tentang dokumen SSM untuk patching node terkelola menjelaskan tujuh dokumen SSM yang tersedia untuk membantu Anda menjaga instans terkelola Anda di-patch dengan pembaruan terbaru terkait keamanan.	10 Januari 2018
Pembaruan penting mengenai dukungan Linux	Pembaruan berbagai topik dengan informasi berikut: • SSM Agentdiinstal, secara default, di Amazon Linux 1 basis AMIs tertanggal 2017.09 dan yang lebih baru. • Instal secara manual SSM Agent pada versi Linux lainnya, termasuk gambar non-basis seperti Amazon ECS AMIs yang dioptimalkan.	9 Januari 2018
Topik baru	Topik baru, Tentang dokumen SSM `AWS-RunPatchBaseline`, menyediakan detail tentang bagaimana dokumen SSM ini beroperasi pada sistem Windows dan Linux. Ini juga menyediakan informasi tentang dua parameter yang tersedia di dokumen `AWS-RunPatchBaseline`, `Operation`, dan `Snapshot ID`.	5 Januari 2018
Topik baru	Bagian baru,BagaimanaPatch Manager operasi bekerja, memberikan rincian teknis yang menjelaskan bagaimana Patch Manager menentukan patch keamanan mana yang akan diinstal dan bagaimana menginstalnya pada setiap sistem operasi yang didukung. Bagian ini juga menyediakan informasi tentang cara aturan dasar patch bekerja pada distribusi sistem operasi Linux yang berbeda	2 Januari 2018

Perubahan	Deskripsi	Tanggal rilis
Perubahan judul dan pemindahan Referensi Tindakan Otomatisasi Systems Manager	Berdasarkan umpan balik pelanggan, referensi tindakan Otomatisasi sekarang disebut referensi runbook Otomatisasi Systems Manager. Selain itu, kami memindahkan referensi ke Sumber Daya Bersama > simpul Dokumen sehingga lebih dekat dengan Referensi plugin dokumen perintah. Untuk informasi selengkapnya, lihat Referensi tindakan Otomatisasi Systems Manager.	20 Desember 2017
Bab dan konten Pemantauan Baru	Bab baru,Pemantauan AWS Systems Manager, memberikan instruksi untuk mengirim metrik dan data log ke Amazon CloudWatch Logs. Topik baru,Mengirim log simpul ke CloudWatch Log terpadu (CloudWatch agen), menyediakan instruksi untuk memigrasikan tugas pemantauan on-instance, hanya pada Windows Server instance 64-bit, dari SSM Agent ke agen. CloudWatch	14 Desember 2017
Bab baru	Bab baru,Identity and access management untuk AWS Systems Manager, memberikan informasi komprehensif tentang penggunaan AWS Identity and Access Management (IAM) dan AWS Systems Manager untuk membantu mengamankan akses ke sumber daya Anda melalui penggunaan kredensil. Kredensil ini memberikan izin yang diperlukan untuk mengakses AWS sumber daya, seperti mengakses data yang disimpan dalam bucket S3 dan mengirim perintah ke dan membaca tag pada instans EC2.	11 Desember 2017
Perubahan pada navigasi kiri	Kami mengubah tajuk di navigasi kiri panduan pengguna ini untuk mencocokkan tajuk di Konsol AWS Systems Manager.	8 Desember 2017

Perubahan	Deskripsi	Tanggal rilis
Beberapa perubahan untuk re:Invent 2017	• Peluncuran resmi AWS Systems Manager: AWS Systems Manager (sebelumnya Amazon EC2 Systems Manager) adalah antarmuka terpadu yang memungkinkan Anda memusatkan data operasional dan mengotomatiskan tugas di seluruh sumber daya Anda. AWS Anda dapat mengakses AWS Systems Manager konsol baru di sini. Untuk informasi selengkapnya, lihat Apakah AWS Systems Manager itu? • Support YAML: Anda dapat membuat dokumen SSM di YAML. Untuk informasi selengkapnya, lihat AWS Systems ManagerDokumen.	29 November 2017
Menggunakan Run Command untuk Mengambil Snapshot Volume EBS yang diaktifkan VSS	Dengan menggunakanRun Command, Anda dapat mengambil snapshot yang konsisten dengan aplikasi dari semua volume Amazon Elastic Block Store (Amazon EBS) yang dilampirkan ke instans Windows Amazon EC2 Anda. Proses snapshot menggunakan Volume Shadow Copy Service (VSS) Windows untuk mengambil cadangan tingkat citra aplikasi VSS-aware, termasuk data dari transaksi yang tertunda antara aplikasi ini dan disk. Selain itu, Anda tidak perlu mematikan instans atau memutusnya saat Anda perlu mencadangkan semua volume yang terpasang . Untuk informasi selengkapnya, lihat Mengambil Snapshot berkemampuan Microsoft VSS Menggunakan di Panduan Pengguna AWS Systems Manager Amazon EC2 untuk Instans Windows.	20 November 2017

Perubahan	Deskripsi	Tanggal rilis
Keamanan Systems Manager yang Disempurnakan dengan Menggunakan VPC Endpoint	Anda dapat meningkatkan postur keamanan instans terkelola (termasuk instans terkelola di lingkungan hibrid) dengan mengkonfigurasi Systems Manager untuk menggunakan VPC endpoint antarmuka. Endpoint antarmuka didukung oleh PrivateLink, teknologi yang memungkinkan Anda mengakses Amazon EC2 dan API Systems Manager secara pribadi dengan menggunakan alamat IP pribadi. PrivateLink membatasi semua lalu lintas jaringan antara instans terkelola, Systems Manager, dan EC2 ke jaringan Amazon (instans terkelola tidak memiliki akses ke Internet). Selain itu, Anda tidak memerlukan gateway internet, perangkat NAT, atau virtual private gateway. Untuk informasi selengkapnya, lihat Membuat titik akhir VPC.	7 November 2017

Perubahan	Deskripsi	Tanggal rilis
Support Inventaris untuk File, Layanan, Peran Windows, dan Registri Windows	Inventaris SSM sekarang mendukung pengumpulan informasi berikut dari instans terkelola Anda. • File: Nama, ukuran, versi, tanggal terinstal, modifikasi dan waktu akses terakhir, dan sebagainya. • Layanan: Nama, nama tampilan, status, layanan dependen, jenis layanan, jenis mulai, dan sebagainya. • Registri Windows: Jalur kunci registri, nama nilai, jenis nilai, dan nilai. • Peran Windows: Nama, nama tampilan, jalur, jenis fitur, status terinstal, dan sebagainya. Sebelum Anda mencoba mengumpulkan informasi untuk jenis inventaris ini, SSM Agent perbarui instans yang ingin Anda inventarisasi. Dengan menjalankan versi terbaruSS M Agent, Anda memastikan bahwa Anda dapat mengumpul kan metadata untuk semua jenis inventaris yang didukung. Untuk informasi tentang cara memperbarui SSM Agent dengan menggunakanState Manager, lihatWalkthrough: Perbarui secara otomatis (SSM AgentCLI). Untuk informasi lebih lanjut tentang Inventaris, lihat Pelajari selengkapnya tentang Inventaris Systems Manager.	6 November 2017
Pembaruan untuk dokumentasi Otomatisasi	Memperbaiki beberapa masalah dalam informasi tentang pengaturan dan konfigurasi akses untuk Otomatisasi Systems Manager. Untuk informasi selengkapnya, lihat Menyiapkan Otomatisasi.	31 Oktober 2017

Perubahan	Deskripsi	Tanggal rilis
GitHubdan Integrasi Amazon S3	Jalankan skrip jarak jauh: Systems Manager sekarang mendukung pengunduhan dan menjalankan skrip dari GitHub repositori pribadi atau publik, dan dari Amazon S3. Dengan menggunakan dokumen SSM yang `AWS-RunRemoteScript` telah ditentukan sebelumnya atau `aws:downloadContent` plugin dalam dokumen SSM khusus, Anda dapat menjalankan Ansible Playbooks dan skrip dengan Python, Ruby, atau, untuk beberapa nama. PowerShell Perubahan ini lebih meningkatkan infrastruktur sebagai kode ketika Anda menggunakan Systems Manager untuk mengotomatisasi konfigurasi dan deployment instans EC2 dan instans terkelola on-premise di lingkungan hibrid Anda. Untuk informasi lebih lanjut, lihat Menjalankan skrip dari GitHub dan Menjalankan skrip dari Amazon S3. Membuat dokumen SSM komposit: Systems Manager sekarang mendukung menjalankan satu atau lebih dokumen SSM sekunder dari dokumen SSM primer. Dokumen-dokumen primer yang menjalankan dokumen-dokumen lain disebut dokumen komposit. Dokumen komposit memungkinkan Anda untuk membuat dan berbagi satu set standar dokumen SSM sekunder Akun AWS untuk tugas-tugas umum seperti perangkat lunak anti-virus boot-strapping atau instance penggabungan domain. Anda dapat menjalankan dokumen komposit dan sekunder yang disimpan di Systems ManagerGitHub, atau Amazon S3. Setelah Anda membuat dokumen komposit, Anda dapat menjalankannya dengan menggunakan dokumen SSM yang ditetapkan `AWS-RunDocument` . Untuk informasi lebih lanjut, lihat Membuat dokumen gabungan dan Menjalankan dokumen dari lokasi terpencil. Referensi plugin dokumen SSM: Untuk akses yang lebih mudah, kami memindahkan Referensi Plugin SSM untuk dokumen SSM dari referensi API Systems Manager dan ke	26 Oktober 2017

Perubahan	Deskripsi	Tanggal rilis
	dalam Panduan Pengguna. Untuk informasi selengkapnya, lihat Referensi plugin dokumen perintah.	
Support untuk Versi Parameter di Parameter Store	Saat Anda mengedit parameter, Parameter Store sekarang secara otomatis mengulangi nomor versi dengan 1. Anda dapat menentukan nama parameter dan nomor versi tertentu dalam panggilan API dan dokumen SSM. Jika Anda tidak menentukan nomor versi, sistem akan secara otomatis menggunakan versi terbaru. Versi parameter menyediakan lapisan perlindungan sekiranya parameter tidak sengaja diubah. Anda dapat melihat nilai semua versi, dan mereferensi versi lama jika perlu. Anda juga dapat menggunakan versi parameter untuk melihat berapa kali parameter berubah selama periode waktu tertentu. Untuk informasi selengkapnya, lihat Bekerja dengan versi parameter.	24 Oktober 2017
Support untuk Menandai Dokumen Systems Manager	Anda sekarang dapat menggunakan AddTagsTo ResourceAPI, file AWS CLI, atau AWS Tools for PowerShell untuk menandai dokumen Systems Manager dengan pasangan nilai kunci. Penandaan membantu Anda mengidentifikasi sumber daya spesifik dengan cepat berdasarkan tag yang telah Anda tetapkan. Ini merupakan tambahan dari dukungan penandaan yang ada untuk instance terkelola, jendela pemeliharaan, Parameter Store parameter, dan baseline patch. Untuk informasi, lihat Menandai dokumen Systems Manager.	3 Oktober 2017

Perubahan	Deskripsi	Tanggal rilis
Berbagai Pembaruan Dokumentasi untuk Memperbaiki Kesalahan atau Memperbarui Konten Berdasarkan Umpan Balik	• Pembaruan Menyiapkan Manajer Sistem untuk lingkungan hybrid dan multicloud dengan informasi untuk Raspbian Linux. • Diperbarui Menyiapkan Systems Manager untuk instans EC2 dengan persyaratan baru untuk Windows Server instance. SSM Agentmemerlukan Windows PowerShell 3.0 atau yang lebih baru untuk menjalankan Dokumen SSM tertentu pada Windows Server instance (misalnya, dokumen `AWS-ApplyPatchBaseline` SSM lama). Verifikasi bahwa instans Windows Server Anda menjalankan Windows Management Framework 3.0 atau yang lebih baru. Kerangka kerja mencakup PowerShell. Untuk informasi lebih lanjut, lihat Windows Management Framework 3.0.	2 Oktober 2017
Memecahkan Masalah Instans Windows Tidak Terjangkau dengan Menggunakan Alur Kerja Otomatisasi EC2Rescue	EC2Rescue dapat membantu Anda mendiagnosis dan memecahkan masalah di instans Windows Server Amazon EC2. Anda dapat menjalankan alat sebagai alur kerja Systems Manager Automation dengan menggunakan dokumen AWSSupport-ExecuteEC2Rescue. Dokumen AWSSupport-ExecuteEC2Rescue dirancang untuk melakukan kombinasi tindakan, tindakan AWS CloudFormation , dan fungsi Lambda Systems Manager yang mengotomatiskan langkah-langkah yang biasanya diperlukan untuk menggunakan EC2Rescue. Untuk informasi selengkapnya, lihat Jalankan alat EC2Rescue pada instans yang tidak dapat dijangkau.	29 September 2017
SSM AgentDiinstal Secara Default di Amazon Linux	SSM Agentdiinstal, secara default, di Amazon Linux AMIs tertanggal 2017.09 dan yang lebih baru. Instal secara manual SSM Agent pada versi Linux lainnya, seperti yang dijelaskan dalamBekerja denganSSM Agent instans EC2 untuk Linux.	27 September 2017

Perubahan	Deskripsi	Tanggal rilis
Run CommandPenyempurnaan	Run Commandtermasuk perangkat tambahan berikut. • Anda dapat membatasi eksekusi perintah ke instance tertentu dengan membuat dan menetapkan kebijakan IAM yang menyertakan kondisi bahwa pengguna hanya dapat menjalankan perintah pada instance yang ditandai dengan tag Amazon EC2 tertentu. Untuk informasi selengkapnya, lihat MembatasiRun Command akses akses berdasarkan tag. • Anda memiliki lebih banyak opsi untuk menargetkan instans dengan menggunakan tag Amazon EC2. Anda sekarang dapat menentukan beberapa kunci tag dan beberapa nilai tag saat mengirim perintah. Untuk informasi selengkapnya, lihat Menjalankan perintah saat skala.	12 September 2017
Systems Manager Didukung pada Raspbian	Systems Manager sekarang dapat berjalan pada perangkat Raspbian Jessie dan Raspbian Stretch, termasuk Raspberry Pi (32-bit).	7 September 2017
Secara otomatis Kirim SSM Agent Log ke Amazon CloudWatch Logs	Anda sekarang dapat membuat perubahan konfigurasi sederhana pada instance Anda untuk SSM Agent mengirim file log ke CloudWatch. Untuk informasi selengkapnya, lihat MengirimSSM Agent log ke CloudWatch Log.	7 September 2017
Mengenkripsi sinkronisasi data sumber daya	Dengan sinkronisasi data sumber daya Systems Manager, Anda dapat menggabungkan data inventaris yang dikumpulkan pada puluhan atau ratusan instans terkelola dalam bucket S3 pusat. Anda sekarang dapat mengenkripsi sinkronisasi data sumber daya dengan menggunakan kunci AWS Key Management Service . Untuk informasi selengkapnya, lihat Panduan: Menggunakan sinkronisasi data sumber daya untuk mengumpulkan data inventaris.	1 September 2017

Perubahan	Deskripsi	Tanggal rilis
State ManagerPenelusuran Baru	Menambahkan dua panduan baru ke dokumentasi: State Manager Walkthrough: Perbarui secara otomatis (SSM AgentCLI) Panduan: Secara otomatis memperbarui driver PV pada instans EC2 untuk Windows Server (konsol)	31 Agustus 2017
Kepatuhan Konfigurasi Systems Manager	Menggunakan Kepatuhan Konfigurasi untuk memindai armada instans terkelola Anda untuk kepatuhan patch dan inkonsistensi konfigurasi. Anda dapat mengumpulkan dan mengumpulkan data dari beberapa Akun AWS dan Wilayah AWS, lalu menelusuri sumber daya tertentu yang tidak sesuai. Secara default, Kepatuhan Konfigurasi menampilkan data kepatuhan tentang Patch Manager penambalan dan State Manager asosiasi. Anda juga dapat menyesuaikan layanan dan membuat jenis kepatuhan Anda sendiri berdasarkan persyaratan IT atau bisnis Anda. Untuk informasi selengkapnya, lihat AWS Systems Manager Kepatuhan.	28 Agustus 2017
Tindakan Otomasi Baru: `aws:executeAutomation`	Menjalankan alur kerja Otomatisasi sekunder dengan memanggil runbook Otomatisasi sekunder. Dengan tindakan ini, Anda dapat membuat runbook Otomatisasi untuk alur kerja yang paling umum Anda, dan mereferensi dokumen tersebut selama eksekusi Otomatisasi. Tindakan ini dapat menyederhanakan runbook Otomatisasi Anda dengan menghapus kebutuhan untuk menduplikasi langkah-langkah di runbook serupa. Untuk informasi selengkapnya, lihat `aws:executeAutomation` – Jalankan otomatisasi lain.	22 Agustus 2017

Perubahan	Deskripsi	Tanggal rilis
Otomatisasi sebagai Target CloudWatch Acara	Anda dapat memulai alur kerja Otomasi dengan menentukan runbook Otomasi sebagai target acara Amazon. CloudWatch Anda dapat memulai alur kerja sesuai jadwal, atau ketika peristiwa AWS sistem tertentu terjadi. Untuk informasi selengkapnya, lihat Jalankan otomatisasi berdasarkan peristiwa.	21 Agustus 2017
State ManagerVersi Asosiasi dan Pembaruan Umum	Anda sekarang dapat membuat versi State Manager asosiasi yang berbeda. Ada kuota 1.000 versi untuk setiap asosiasi. Anda juga dapat menentukan nama untuk asosiasi Anda. Juga, State Manager dokumentasi telah diperbarui untuk mengatasi informasi yang sudah ketinggalan zaman dan inkonsistensi. Untuk informasi selengkapnya, lihat AWS Systems Manager State Manager.	21 Agustus 2017

Perubahan	Deskripsi	Tanggal rilis
Perubahan ke Maintenance Windows	Maintenance Windowstermasuk perubahan atau penyempurnaan berikut: • Sebelumnya, hanya Maintenance Windows bisa melakukan tugas dengan menggunakanRun Command. Anda sekarang dapat melakukan tugas dengan menggunakan Systems Manager Automation, AWS Lambda, dan AWS Step Functions. • Anda dapat mengubah target jendela pemeliharaan, menentukan nama target, deskripsi, dan pemilik. • Anda dapat mengedit tugas di jendela pemeliharaan, termasuk menentukan dokumen SSM baru untuk Run Command dan tugas Otomasi. • Semua Run Command parameter sekarang didukung, termasuk DocumentHash, DocumentHashType, TimeoutSeconds, Komentar, dan NotificationConfig. • Sekarang Anda dapat menggunakan bendera `safe` ketika Anda mencoba untuk membatalkan pendaftaran target. Jika diaktifkan, sistem mengembalikan kesalahan jika target direferensikan oleh tugas apa pun. Untuk informasi selengkapnya, lihat AWS Systems Manager Maintenance Windows.	16 Agustus 2017
Tindakan Otomasi Baru: `aws:approve`	Tindakan baru untuk runbook Otomatisasi ini menghentikan Otomatisasi untuk sementara waktu sampai prinsip utama yang ditunjuk menyetujui atau menolak tindakan. Setelah jumlah persetujuan yang diperlukan tercapai, eksekusi Otomatisasi dilanjutkan. Untuk informasi selengkapnya, lihat Referensi tindakan Otomatisasi Systems Manager.	10 Agustus 2017

Perubahan	Deskripsi	Tanggal rilis
Mengambil Peran Otomatisasi Tidak Lagi Diperlukan	Otomatisasi sebelumnya mengharuskan Anda menentukan peran layanan (atau mengambil peran) sehingga layanan memiliki izin untuk melakukan tindakan atas nama Anda. Otomatisasi tidak lagi memerlukan peran ini karena layanan sekarang beroperasi dengan menggunakan konteks pengguna yang memanggil eksekusi. Namun, situasi berikut ini masih mengharuskan Anda menentukan peran layanan untuk Otomatisasi: • Saat Anda ingin membatasi izin pengguna pada sumber daya, tetapi Anda ingin pengguna menjalankan Otomatisasi yang memerlukan izin yang ditingkatkan. Dalam skenario ini, Anda dapat membuat peran layanan dengan izin yang ditingkatkan dan memungkinkan pengguna untuk menjalankan alur kerja. • Operasi yang Anda perkirakan akan berjalan lebih dari 12 jam memerlukan peran layanan. Untuk informasi selengkapnya, lihat Menyiapkan Otomatisasi.	3 Agustus 2017
Kepatuhan Konfigurasi	Menggunakan Kepatuhan Konfigurasi Amazon EC2 Systems Manager untuk memindai armada instans terkelola Anda untuk kepatuhan patch dan inkonsistensi konfigurasi. Anda dapat mengumpulkan dan mengumpulkan data dari beberapa Akun AWS dan Wilayah AWS, lalu menelusuri sumber daya tertentu yang tidak sesuai. Untuk informasi selengkapnya, lihat AWS Systems Manager Kepatuhan.	8 Agustus 2017

Perubahan	Deskripsi	Tanggal rilis
Penyempurnaan Dokumen SSM	Dokumen Perintah dan Kebijakan SSM sekarang menawarkan dukungan lintas platform. Ini berarti bahwa satu dokumen SSM dapat memproses plugin untuk sistem operasi Windows dan Linux. Dukungan lintas-platform memungkinkan Anda mengkonsolidasikan jumlah dokumen yang Anda kelola. Dukungan lintas-platform ditawarkan dalam dokumen SSM yang menggunakan skema versi 2.2 atau yang lebih baru. Dokumen Perintah SSM yang menggunakan skema versi 2.0 atau yang lebih baru sekarang dapat mencakup beberapa plugin dari jenis yang sama. Misalnya, Anda dapat membuat dokumen Perintah yang memanggil plugin `aws:runRunShellScript` beberapa kali. Untuk informasi selengkapnya tentang perubahan skema versi 2.2, lihat AWS Systems Manager dokumen. Untuk informasi selengkapnya tentang plugin SSM, lihat Referensi plugin dokumen perintah.	12 Juli 2017

Perubahan	Deskripsi	Tanggal rilis
Patching Linux	Patch Managersekarang dapat menambal distribusi Linux berikut: Sistem 64-bit dan 32-bit • Amazon Linux 2014.03, 2014.09, atau yang lebih baru • Ubuntu Server16,04 LTS, 14,04 LTS, atau 12,04 LTS • Red Hat Enterprise Linux (RHEL) 6.5 atau yang lebih baru Hanya sistem 64-bit • Amazon Linux 2015.03, 2015.09, atau yang lebih baru • Red Hat Enterprise Linux (RHEL) 7.x atau yang lebih baru Untuk informasi selengkapnya, lihat AWS Systems Manager Patch Manager. Note • Untuk menambal instance Linux, instance Anda harus menjalankan SSM Agent versi 2.0.834.0 atau yang lebih baru. Untuk informasi tentang memperbarui agen, lihat bagian berjudul Contoh: Perbarui SSM Agent diMenjalankan perintah dari konsol. • Dokumen SSM `AWS-ApplyPatchBaseline` digantikan oleh dokumen `AWS-RunPatchBaseline` .	6 Juli 2017

Perubahan	Deskripsi	Tanggal rilis
Sinkronisasi data sumber daya	Anda dapat menggunakan sinkronisasi data sumber daya Systems Manager untuk mengirim data Inventaris yang dikumpulkan dari semua instans terkelola Anda ke satu bucket Amazon S3. Sinkronisasi data sumber daya kemudian secara otomatis memperbarui data terpusat saat data Inventaris baru dikumpulkan. Dengan semua data Inventaris yang disimpan dalam bucket S3 target, Anda dapat menggunakan layanan seperti Amazon Athena dan QuickSight Amazon untuk menanyakan dan menganali sis data gabungan. Untuk informasi selengkapnya, lihat. Pengonfigurasian sinkronisasi data sumber daya untuk Inventaris Untuk contoh cara bekerja dengan sinkronisasi data sumber daya, lihat Panduan: Menggunakan sinkronis asi data sumber daya untuk mengumpulkan data inventari s.	29 Juni 2017
Hierarki Parameter Systems Manager	Mengelola puluhan atau ratusan parameter Systems Manager sebagai daftar datar memakan waktu dan rentan terhadap kesalahan. Anda dapat menggunakan hierarki parameter untuk membantu Anda menata dan mengelola parameter Systems Manager. Hierarki adalah nama parameter yang menyertakan jalur yang Anda tentukan dengan menggunakan garis miring. Contoh berikut ini menggunakan tiga tingkat hierarki dalam nama untuk mengidentifikasi hal berikut: /Lingkungan/Jenis Komputer/Aplikasi/Data `/Dev/DBServer/MySQL/db-string13` Untuk informasi selengkapnya, lihat Bekerja dengan hierarki parameter. Untuk contoh bagaimana untuk bekerja dengan hierarki parameter, lihat Bekerja dengan hierarki parameter.	22 Juni 2017

Perubahan	Deskripsi	Tanggal rilis
SSM Agent Dukungan untuk SUSE Linux Enterprise Server	Anda dapat menginstal SSM Agent pada 64-bit SUSE Linux Enterprise Server (SLES). Untuk informasi selengkapnya, lihat Bekerja denganSSM Agent instans EC2 untuk Linux.	14 Juni 2017

Konvensi dokumen

Berikut ini adalah konvensi tipografi umum untuk PanduanAWS Systems Manager Pengguna.

Contoh dibedakan untuk sistem operasi lokal atau bahasa baris perintah

Kami menggunakan tab untuk menampilkan contoh perintah berdasarkan jenis sistem operasi lokal pengguna pengguna lokal pengguna. Untuk contoh Linux dan macOS, kita menggunakan karakter garis miring terbalik (\) untuk memecah perintah panjang menjadi beberapa baris. Untuk contoh Windows Server, kita menggunakan karakter tanda sisipan (^) untuk memecah perintah menjadi beberapa baris.

Contoh:

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
```

Elemen dalam antarmuka pengguna

Pemformatan: Teks dalam huruf tebal

Contoh: Pilih File, Properties (Properti).

Input pengguna (teks yang diketik pengguna)

Pemformatan: Teks dalam font monospace

Contoh: Untuk nama, ketik **`my-new-resource`**.

Teks placeholder untuk nilai yang diperlukan

Pemformatan: Teks dalam *huruf miring*

Contoh:

```
aws ec2 register-image --image-location my-s3-bucket/image.manifest.xml
```

AWS Glosarium

Untuk AWS terminologi terbaru, lihat AWS glosarium di Referensi.Glosarium AWS

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.