aws

Administrator Guide

AWS Service Management Connector

AWS Service Management Connector: Administrator Guide

Table of Contents

What is AWS Service Management Connector?

AWS Service Management Connector and its integration connectors enable you to provision, manage, and operate native AWS resources and capabilities in familiar IT Service Management (ITSM) tooling, such as ServiceNow and Atlassian.

These integrations enable Enterprises to accelerate migration and AWS adoption at scale through oversight and governance in their declared operational tooling and system of record.

For an interactive introduction to AWS Service Management Connector, review the ServiceNow Connector workshop in the *AWS workshop studio*.

Security in AWS Service Management Connector

Service Management Connector uses the roles and permissions that an IAM user requires to access your specific AWS resources and services. Service Management Connector requires two IAM user roles, *SyncUser* and *EndUser*, to perform various integration operations. For more information, see your chosen Connector to identify the IAM permissions for a specific integration.

Service Management Connector is not within the scope of any AWS compliance programs. Using Service Management Connector to access a service does not alter that service's compliance.

Encryption at rest — Service Management Connector does not store any customer data. The connector installs Tables and Schemas on third-party platforms that can store credentials in the platform's database. All credentials are encrypted and masked to comply with platform best practices.

Encryption in transit — By default, AWS encrypts all data transmitted between external platforms and Service Management Connector by sending data through a HTTPS/TLS connection.

AWS Service Management Connector for ServiceNow

The AWS Service Management Connector for ServiceNow (formerly the AWS Service Catalog Connector) enables ServiceNow end users to provision, manage, and operate AWS resources natively through ServiceNow.

ServiceNow administrators can:

- Provide pre-approved, secured, and governed AWS resources to end users through Service Catalog.
- Execute automation playbooks through AWS Systems Manager.
- View and manage operational items as incidents through AWS Systems Manager OpsCenter.
- Use AWS Config to track resources in the CMDB seamlessly on ServiceNow with the AWS Service Management Connector.
- Define new resource types based on ServiceNow CMDB tables and synchronize these with AWS Config custom resources.
- Sync AWS Security Hub findings to ServiceNow incidents or problems.

ServiceNow end users can:

- Browse, request, and provision pre-secured AWS solutions.
- View AppRegistry applications, attribute groups, and related resource details with AppRegistry.
- View, update, and resolve Incidents from AWS Systems Manager OpsItems.
- View configuration item details.
- Execute workflows in ServiceNow on AWS resources.
- View, update, and resolve ServiceNow incidents or problems through AWS Security Hub findings.
- View, create, add correspondence and resolve Support cases from ServiceNow (including AMS Accelerate support cases).
- View and execute AWS Systems Manager Change Requests from a curated list of pre-approved AWS Change templates.
- View resource performance and the availability of AWS services and account through AWS Health dashboard.
- Manage and resolve incidents affecting AWS-hosted applications through the integration with AWS Systems Manager Incident Manager.

These features minimize direct AWS platform access, simplify AWS product request and operational actions for ServiceNow users. They also provide streamlined Service Management governance and oversight over AWS resources and services.

The AWS-supplied connector is available at no charge in the ServiceNow store. It supports ServiceNow platform releases San Diego(S), Rome (R), and Quebec (Q - Patch 5 going forward). These new features are generally available in all AWS Regions where AWS Service Catalog, AWS Config, and AWS Systems Manager services are available. For list of regions and service quotas of AWS services, see Service endpoints and quotas.

> **Note**
>
> For the ServiceNow Quebec release, we only support Quebec Patch 5 going forward due to a deprecated ServiceNow REST API call, `getDeprecatedValue()`, which inhibited end users' ability to request AWS Service Catalog products and AWS Systems Manager automation documents in the Connector. ServiceNow resolved the issue in Quebec Patch 5, so we now support only Patch 5 going forward.

The following AWS services integrate into this Connector:

- Service Catalog allows you to centrally manage commonly deployed AWS services and provisioned software products. It helps your organization achieve consistent governance and compliance requirements, while enabling users to quickly deploy only the approved AWS services they need. It also offers AppRegistry, which creates a repository of your applications and associated resources.
- AWS Config enables you to assess, audit, and evaluate the configurations of your AWS resources. AWS Config continuously monitors and records your AWS resource configurations. It also lets you automate the evaluation of recorded configurations against desired configurations.
- AWS Systems Manager gives you visibility and control of your infrastructure on AWS. Systems Manager provides a unified user interface so you can view operational data from multiple AWS services, investigate and resolve operational issues through OpsCenter and Incident Manager, and automate operational tasks across your AWS resources.
- AWS Security Hub gives you a comprehensive view of your security alerts and security posture across your AWS accounts. With AWS Security Hub, there is a single place that aggregates, organizes, and prioritizes your security alerts, or findings.

- AWS Health provides personalized information about events that can affect your AWS infrastructure, guides you through scheduled changes, and accelerates the troubleshooting of issues that affect your AWS resources and accounts.
- Support provides multiple tooling mechanisms, people, and programs designed to proactively help you optimize performance, lower costs, and innovate faster. Support enables you to be successful on your cloud journey. It addresses requests that range from answering best practices questions to providing guidance on configuration and break-fix and problem resolution.
- ServiceNow is an enterprise service management platform that places a service-oriented lens on the activities, tasks, and processes that enable day-to-day work life and a modern work environment. ServiceNow Service Catalog is a self-service application that end users can use to order IT services based on request fulfillment approvals and workflows. The ServiceNow CMDB provides resource transparency and relationships for the logical components of a service.

Align the ServiceNow Connector to industry best practices

This Connector aligns to industry best practices such as ITIL®'s service management areas by enabling tools (services) with the intersection of people, processes and partners. The Connector also addresses a baseline set of service management practices customers use within existing operational tooling:

Service Management Area	AWS service(s) integration
Service Catalog Management Deployment Management (Provisioning)	AWS Service Catalog or AWS CloudFormation (Requesting and provisioning vetted or predictable products and performing post-provision actions)
Incident Management (ticketing)	Support (AWS services or platform incidents) AWS Systems Manager OpsCenter (Operational incidents derived or detected for solutions built on AWS platform) AWS Security Hub (Incidents derived from security Findings)

Service Management Area	AWS service(s) integration
	AWS Systems Manager Incident Manager (Incidents generated according to response plans)
Service Configuration Management (CMDB)	AWS Config(AWS resource or configuration items tracking and detective control compliance)
Change Enablement (management)	AWS Systems Manager Change Manager (Standard changes with automated runbooks as implementation task(s))
Measurement & Reporting	AWS Health Dashboard (Visibility into resource performance)

Setting up AWS Service Management Connector for ServiceNow

Before installing the AWS Service Management Connector for ServiceNow, verify that you have the necessary permissions in your AWS account and ServiceNow instance.

Topics

AWS Service Management Connector for ServiceNow prerequisites

Make sure you have AWS and ServiceNow prerequisites configured before you get started.

- **AWS Service Catalog with the Connector** — You must have an AWS account to configure your AWS portfolios and products. For details, refer to Setting up for Service Catalog and Using AppRegistry.
- **AWS Config details** — Configure the service settings to record data for the resource types of interest. We recommend you include provisioned products and AWS CloudFormation stacks, in addition to the major resource types that your team uses. For more information, see Setting up

AWS Config with the console. This version of the Connector enables the import of aggregated Config data in a single AWS account from more than one AWS Region or account. To use this feature, you must configure an aggregator in AWS. For more information, see Setting up an aggregator using the console.

- **AWS Systems Manager Automation with the Connector** — This feature requires no AWS-side set up. As standard, AWS provides a number of automation documents (runbooks). If you want additional automation documents (runbook), retrieve them in the Connector. For more information, see Working with Automation Runbooks.
- **AWS Systems Manager OpsCenter with the Connector** — You must enable the service in all Regions and accounts where you want to sync OpsItems. For more information, see Getting started with OpsCenter
- **AWS Security Hub with the Connector** — You must enable the service in all Regions and accounts where you want to sync Findings. For details, see Setting up Security Hub. We recommend you connect ServiceNow with the primary (main) AWS account for AWS Security Hub. For more information, see Managing administrator and member accounts.
- **Support with the Connector** — Your account must have a Business or Enterprise Support plan to use support integration with the Connector.
- **AWS Systems Manager Change Manager with the Connector** — You must enable the service in all Regions and accounts where you want to sync change templates. The AWS Systems Manager Change Manager integration of AWS Service Management Connector introduces a curated version of the integration. It allows customers to execute pre-approved change templates that contain at least one Automation Runbook and does not require approvals during execution from ServiceNow. For more information, see Setting up Change Manager.
- **AWS Systems Manager Incident Manager with the Connector** — You must enable Incident Manager in all AWS Regions and accounts from where you want to sync the incidents. For details, see Setting up for AWS Systems Manager Incident Manager.
- **AWS Health with the Connector** — Your account must have a Business or Enterprise Support plan to use AWS Health integration with the Connector.
- **ServiceNow instance** — You need a ServiceNow instance to install the ServiceNow Connector scoped application. The initial installation should occur in either an enterprise sandbox or a ServiceNow Personal Developer Instance (PDI), depending on your organization's technology governance requirements. The ServiceNow administrator needs the admin role to install the Connector for ServiceNow scoped application.

Setting baseline permissions for AWS Service Management Connector for ServiceNow

This section describes how to configure Identity and Access Management (IAM) permissions, AWS Service Catalog, and other AWS services to use AWS Service Management Connector for ServiceNow.

To use an AWS CloudFormation template to set up the AWS configurations of the Connector for ServiceNow, refer to the AWS configurations for Connector for ServiceNow AWS commercial Regions , AWS GovCloud Regions, and AWS China Regions.

> **Note**
>
> The AWS CloudFormation template creates IAM users with permissions to all existing integrations, and *is intended to enable all supported integrations in a sandbox or developer ServiceNow instance*. For quality-assurance and production, you must apply least-privilege permissions based on the integrations enabled through the connector. Review the Creating users section for additional information.

> **Note**
>
> If you choose to use the Connector for ServiceNow AWS Configuration template, go to the AWS Service Catalog Administrator Guide .

Creating Connector for ServiceNow users

For each AWS account, the Connector for ServiceNow requires two users:

- **AWS Sync User**: A user to sync AWS resources (such as portfolios, products, automation documents (runbook), Ops Items, Incident Manager incidents, change templates and requests, configuration items, and security Findings), sync AWS support cases, and AWS Health events and resources to ServiceNow .
- **AWS End User**: A user who can provision products as an end user, execute requests, and view resources that ServiceNow exposes. This role includes any required roles to provision and execute.

> **Note**
>
> To align with best practices, AWS recommends periodically rotating IAM user access keys. For more information, refer to Manage IAM user access keys properly.

Creating the AWS Service Management Connector Sync user

This section describes how to create the AWS Sync user and associate the appropriate IAM permission. To perform this task, you need IAM permissions to create new users. The following steps to create a Sync user and End user are not required if you use the AWS CloudFormation template to deploy the permissions. Review Setting baseline permissions for AWS Service Management Connector for ServiceNow for more information.

> **Note**
>
> The AWS CloudFormation template to set up the AWS configurations of the Connector for ServiceNow creates the Sync user and End user with the required permissions for all the supported integrations.

To create AWS Service Management Connector sync user

1. Follow the instructions in Creating an IAM user in your AWS account to create a sync user (SMSyncUser). The user needs programmatic and AWS Management Console access to follow the Connector for ServiceNow installation instructions.
2. Set permissions for your sync user (SMSyncUser). Choose **Attach existing policies directly** and select:
   - **`AWSServiceCatalogAdminReadOnlyAccess`** (AWS managed policy)
   - **`AmazonSSMReadOnlyAccess`** (AWS managed policy)
   - **`AWSConfigUserAccess`** (AWS managed policy)
   - **`AWSSupportAccess`** (AWS managed policy)
3. Create this policy: `ConfigBidirectionalPolicy`. Then follow the instructions in Creating IAM Policies, and add this code in the JSON editor:

```
{
```

```
   "Version": "2012-10-17",
   "Statement": [
     {
         "Action": [
         "cloudformation:RegisterType",
         "cloudformation:DescribeTypeRegistration",
         "cloudformation:DeregisterType",
         "config:PutResourceConfig"
     ],
     "Resource": "*",
     "Effect": "Allow"
     }
   ]
}
```

The provided AWS Configuration template consists of two policies: `ConfigBiDirectionalPolicy` and `SecurityHubPolicy`.

4. Create this policy: `SecurityHubPolicy`. Then follow the instructions in Creating IAM policies, and add this code in the JSON editor:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "sqs:ReceiveMessage",
                "sqs:DeleteMessage"
            ],
            "Resource": "<add sqs ARN here> ",
            "Effect": "Allow"
        },
        {
            "Action": [
                "securityhub:BatchUpdateFindings"
            ],
            "Resource": "*",
            "Effect": "Allow"
        }
    ]
}
```

5. Create this policy: `OpsCenterExecutionPolicy`. Then follow the instructions in Creating IAM Policies and add this code in the JSON editor:

```
  {
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:CreateOpsItem",
                "ssm:GetOpsItem",
                "ssm:UpdateOpsItem",
                "ssm:DescribeOpsItems"
             ],
            "Resource": "*"
        }
    ]
}
```

6. Create this policy: `AWSIncidentBaselinePolicy`. Then follow the instructions in Creating IAM Policies and add this code in the JSON editor:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "ssm-incidents:ListIncidentRecords",
                "ssm-incidents:GetIncidentRecord",
                "ssm-incidents:UpdateRelatedItems",
                "ssm-incidents:ListTimelineEvents",
                "ssm-incidents:GetTimelineEvent",
                "ssm-incidents:UpdateIncidentRecord",
                "ssm-incidents:ListRelatedItems",
                "ssm:ListOpsItemRelatedItems"
            ],
            "Resource": "*",
            "Effect": "Allow"
        }
    ]
}
```

7. [Optional] Create this policy: `AWSChangeManagerCloudtrailPolicy`. Then follow the instructions in Creating IAM Policies and add this code in the JSON editor:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "cloudtrail:DescribeQuery",
                "cloudtrail:ListEventDataStores",
                "cloudtrail:StartQuery",
                "cloudtrail:GetQueryResults"
            ],
            "Resource": "*",
            "Effect": "Allow"
        }
    ]
}
```

8. Create this policy: `DescribeWorkSpacesPolicy`. Then follow the instructions in Creating IAM Policies and add this code in the JSON editor:

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

9. Add a policy that allows `budgets:ViewBudget` on all resources (*).
10. Review and choose **Create User**.
11. Note the access and secret access information. Download the .csv file that contains the user credential information.

> **Note**
>
> To align with best practices, AWS recommends periodically rotating IAM user access keys. For more information, refer to Manage access keys for IAM users.

Creating the AWS Service Management Connector end user

This section describes how to create the AWS Service Management Connector end user and associates the appropriate IAM permission. To perform this task, you need IAM permissions to create new users.

To create AWS Service Management Connector end user

1. Follow the instructions in Creating an IAM user in your AWS account to create a user (SMEndUser). The user needs programmatic and AWS Management Console access to follow the Connector for ServiceNow installation instructions.

   For products using AWS CloudFormation StackSets, you need to create a StackSet inline policy. With AWS CloudFormation StackSets, you are able to create products across multiple accounts and Regions.

   Using an administrator account, you define and manage a Service Catalog product. You also use it to provision stacks into selected target accounts across specified Regions. You need to have the necessary permissions defined in your AWS accounts.

   To set up the necessary permissions, see Granting Permissions for Stack Set Operations. Follow the instructions to create an `AWSCloudFormationStackSetAdministrationRole` and an `AWSCloudFormationStackSetExecutionRole`.
2. Add the following permissions (policies) to the user:

   - `AWSServiceCatalogEndUserFullAccess` (AWS managed policy)
   - `StackSet` (inline policy) - For Service Catalog products with stack sets, you need to modify the SMEndUser to include the Read Only permissions for the services you want to provision. For example, to provision an Amazon S3 bucket, include the `AmazonS3ReadOnlyAccess` policy to the `SMEndUser`.
   - `OpsCenterExecutionPolicy`
   - `AmazonEC2ReadOnlyAccess` (AWS managed policy)

- `AmazonS3ReadOnlyAccess` (AWS managed policy)

Creating the SCConnectLaunch role

The `SCConnectLaunch` role is an IAM role that places baseline AWS service permissions into the AWS Service Catalog launch constraints. Configuring this role enables segregation of duty through provisioning product resources for ServiceNow end users.

The `SCConnectLaunch` role baseline contains permissions to Amazon EC2 and Amazon S3 services. If your products contain more AWS services, you must either include those services in the `SCConnectLaunch` role or create new launch roles.

This section describes how to create the `SCConnectLaunch` role. This role places baseline AWS service permissions in the Service Catalog launch constraints. For more information, see Service Catalog Launch Constraints.

To create SCConnectLaunch role

1. Create this policy: `AWSCloudFormationFullAccess` policy. Choose **create policy** and add this code in the JSON editor:

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "cloudformation:DescribeStackResource",
            "cloudformation:DescribeStackResources",
            "cloudformation:GetTemplate",
            "cloudformation:List*",
            "cloudformation:DescribeStackEvents",
            "cloudformation:DescribeStacks",
            "cloudformation:CreateStack",
            "cloudformation:DeleteStack",
            "cloudformation:DescribeStackEvents",
            "cloudformation:DescribeStacks",
            "cloudformation:GetTemplateSummary",
            "cloudformation:SetStackPolicy",
            "cloudformation:ValidateTemplate",
```

```
            "cloudformation:UpdateStack",
            "cloudformation:CreateChangeSet",
            "cloudformation:DescribeChangeSet",
            "cloudformation:ExecuteChangeSet",
            "cloudformation:DeleteChangeSet",
            "s3:GetObject"
         ],
         "Resource":"*"
      }
   ]
}
```

> **Note**
>
> `AWSCloudFormationFullAccess` includes additional permissions for ChangeSets.

2. Create this policy: `ServicecodeCatalogSSMActionsBaseline`. Follow the instructions in Creating IAM policies, and add this code in the JSON editor:

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"Stmt1536341175150",
         "Action":[
            "servicecatalog:AssociateResource",
            "servicecatalog:DisassociateResource",
            "servicecatalog:ListServiceActionsForProvisioningArtifact",
            "servicecatalog:ExecuteprovisionedProductServiceAction",
            "ssm:DescribeDocument",
            "ssm:GetAutomationExecution",
            "ssm:StartAutomationExecution",
            "ssm:StopAutomationExecution",
            "ssm:StartChangeRequestExecution",
            "cloudformation:ListStackResources",
            "ec2:DescribeInstanceStatus",
            "ec2:StartInstances",
            "ec2:StopInstances"
         ],
         "Effect":"Allow",
         "Resource":"*"
```

```
      },
      {
         "Effect":"Allow",
         "Action":"iam:PassRole",
         "Resource":"*",
         "Condition":{
            "StringEquals":{
               "iam:PassedToService":"ssm.amazonaws.com"
            }
         }
      }
   ]
}
```

3. Create the SCConnectLaunch role. Then assign the trust relationship to Service Catalog.

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

4. Attach the relevant policies to the SCConnectLaunch role.

   We recommend you customize and scope your launch policies to the specific AWS Services, which are in the associated CloudFormation template for the given Service Catalog product.

   For example, to provision EC2 and S3 products, your role policies are as follows:

   - AmazonEC2FullAccess AWS managed policy)
   - AmazonS3FullAccess AWS managed policy)
   - AWSCloudFormationFullAccess (custom managed policy)

- `ServiceCatalogSSMActionsBaseline` (custom managed policy)

Configuring core ServiceNow components

This section describes how to configure core components in ServiceNow.

> **Note**
>
> Before installing the AWS Service Management scoped app, we recommend you clear the ServiceNow platform and your browser cache.
> Ensure that you install the update set in a non-production or sandbox environment. Consult a ServiceNow system administrator if you need approval to clear the ServiceNow platform cache.

Topics

Activating ServiceNow plugins

AWS Service Management Connector uses three ServiceNow plugins to provide useful components to the integration features:

- User Criteria Scoped API (for AWS Service Catalog integration)
- Discovery and Service Mapping Patterns (for AWS Config integration)
- Change Management – Change Model Foundation Data (for AWS Systems Manager Change Manager integration)

To activate the User Criteria Scoped API plugin

1. In your ServiceNow dashboard, enter **`plugins`** into the navigation panel in the upper left.
2. When the **System Plugins** page populates, next to the **Name** dropdown, search for **User Criteria**.
3. Choose **User Criteria Scoped API** and then choose **Activate**.

To activate the Discovery and Service Mapping Patterns plugin

1. In your ServiceNow dashboard, enter **`plugins`** into the navigation panel in the upper left.
2. When the **System Plugins** page populates, next to the **Name** dropdown, search for **Discovery**.
3. Choose **Discovery and Service Mapping Patterns** and then choose **Activate**.

> **Note**
>
> This plugin is free and aligns to the CMDB tables outside of ServiceNow's family release CMDB updates.

To activate the Change Management – Change Model Foundation Data plugin

1. In your ServiceNow dashboard, enter **`plugins`** in the navigation panel in the upper left.
2. When the System Plugins page populates, next to the **Name** dropdown, search for **Change Management**.
3. Choose **Change Management - Change Model Foundation Data** and then choose **Activate**.

Installing ServiceNow Connector scoped application

The AWS Service Management Connector for ServiceNow is a conventional, scoped application that was developed and released through a ServiceNow update set. Update sets are code changes to the base platform that lets developers move code across ServiceNow instances.

Download and install a certified version of the connector for no additional cost from the following locations:

- ServiceNow store

- ServiceNow update set: AWS Service Management Connector offers an update set for users who want to install the connector application in a ServiceNow Personal Developer Instance (PDI) or sandbox environment.

If you don't already have a ServiceNow instance, start with the following first step. If you already have a ServiceNow instance, use the previous links to download and install the connector.

To install the connector, complete the following steps.

Obtain a ServiceNow instance

1. Open Obtaining a Personal Developer Instance.
2. Create ServiceNow developer program credentials.
3. Follow the instructions for requesting a ServiceNow instance.
4. Capture your instance details, including URL, administrative ID, and temporary password credentials.

To install the update set

1. In your ServiceNow dashboard, enter **`update sets`** into the navigation panel in the upper left.
2. Choose **Retrieved Update Sets** from the results.
3. Choose **Import Update Set from XML** and upload the release XML file.
4. Choose the **AWS Service Management Connector for ServiceNow** update set.
5. Choose **Preview Update Set**, which makes ServiceNow validate the Connector update set.
6. Choose **Update**.
7. Choose **Commit Update Set** to apply the update set and create the application. This procedure should complete 100%.

Configuring Connector using Guided Setup

The Connector for ServiceNow includes a Guided Setup mechanism to enable customers to configure and mark complete ServiceNow installation components for the AWS Service Management Connector.

Guided Setup enables the customers to plan the roll-out of the Connector and perform the basic configurations of the Connector to launch it across ServiceNow staged environments.

The Connector Guided Setup:

- Provides a direct set of links to the pages in the ServiceNow instance where you can perform the configuration.
- Tracks completed tasks so you can stop and start again where you left off.
- Enables less maneuvering between AWS documentation and the ServiceNow instance.
- Coordinates the deployment and configuration of the Connector for individuals and teams.

> **Note**
>
> Only ServiceNow admin users can access the Guided Setup to configure the Connectors.

To configure Connector using Guided Setup

1. Log in to your ServiceNow instance as an admin user.
2. Enter **`AWS Service Management Connector`** in the left filter navigator.
3. Choose **Guided Setup**.
4. Review details on the Guided Setup homepage and choose **Get Started**.
5. Review details on each section.
6. To perform a task, select the task and choose **Configure**.
7. After completion of the task, choose **Mark as Complete**.

   To skip sections or tasks that do not apply to you, choose Skip.

Platform system administrator components

To enable the AWS Service Management Connector scoped application named **AWS Service Management**, the system admin must create a discovery source, and configure specific platform tables, forms, and views.

Create a discovery source AWS Service Management Connector entry

You must create a new discovery data source, AWS Service Management Connector.

To enable AWS to report discovered CIs into your CMDB

1. Choose **System Definition**. Then select **Choice Lists**.
2. Choose **New**.
3. Create a new entry with these details:
   - **Table: `Configuration Item [cmdb_ci]`**
   - **Element: `discovery_source`**
   - **Label: `AWS Service Management Connector`**
   - **Value: `AWS Service Management Connector`**

> **Note**
>
> Make sure you are in Global mode in ServiceNow System Settings to modify System Definitions.

Administering AWS Service Management Connector Dashboard

As the system administrator, you can restrict access to the dashboard and its reports for specific users, roles or groups.

To restrict access to the connector dashboard

1. In the ServiceNow instance, navigate to the AWS Service Management Connector dashboard.
2. Choose the **Share** icon and then select **Add users, groups, or roles**.
3. Add the users, groups, or roles that require access to the dashboard.
4. (optional) You can also restrict access to the reports available in the dashboard. For detailed instructions, review Administering reports in the *ServiceNow product documentation*.

Enabling permissions on ServiceNow Platform

For AWS products to display under AWS portfolios as sub-categories in the ServiceNow Service Catalog, you need to modify the Application Access form for Catalog Item Category tables. This action is necessary because a ServiceNow scoped API is not available for the Catalog Item Category table.

To view AWS Service Catalog products (Catalog Item Category)

1. Enter **`Tables`** in the Navigator and choose **System Definition**, then choose **Tables**.
2. In the list of tables, search for a table with label **Catalog Item Category** (or with the name `sc_cat_item_category`). The list of tables displays.
3. Choose **Category** to view the form defining the table.
4. Choose the **Application Access** tab on the form and select **Can Create**, **Can Update**, and **Can Delete** on the form.
5. Choose **Update**.

To enable the connector to control visibility of Service Catalog products on Service Portal through Allowed Groups

> **Note**
>
> This step is only required if the Application Access is not already enabled in your ServiceNow instance. Additionally, Service Management Connector recommends that you enable the `User Criteria Scope API` plugin.

1. Enter **`Tables`** in the Navigator and choose **System Definition**, then choose **Tables**.
2. In the list of tables, search for a table with label **Catalog Item Available for** (or with the name `sc_cat_item_user_criteria_mtom`). The list of tables displays.
3. Choose **Category** to view the form defining the table.
4. Choose the **Application Access** tab on the form and select **Can Create** and **Can Update** on the form.
5. Choose **Update**.

ServiceNow permissions for administrators of the Connector scoped app

The AWS Service Management scoped app has two ServiceNow roles that enable access to configure the application. This feature enables system admins to grant one or more user's privileges to administer the application, without having to open full sysadmin access to them. System admins can assign these roles to either individual users or to one administrator user.

To set up Connector application administrator privileges

1. Enter **`Users`** in the navigator and select **System Security – Users**.
2. Choose a user to grant one or both previous roles (such as admin). You can also Administer the Now Platform.
3. Choose **Edit** on the **Roles** tab of the form.
4. Filter the collection of roles by the prefix **`x_126749_aws_sc`**.
5. Choose one or more of the following and add them to the user: **x_126749_aws_sc_account_admin**, **x_126749_aws_sc_portfolio_manager**, **x_126749_ aws_sc.appregistry_manager, x_126749_ aws_sc.automation_manager**, **x_126749_aws_sc.finding_manager**, **x_126749_aws_sc.opscenter_manager**, **x_126749_aws_sc.support_case_manager** , **x_126749_aws_sc.change_manager_manager**, **x_126749_aws_sc.productsearchaccess**, **x_126749_aws_sc.cloudtrail_event_user**, and **x_126749_aws_sc.health_dashboard_viewer.**
6. Choose **Save**.

To add Service Catalog to ServiceNow Service Catalog categories

1. Choose **Self Service | Service Catalog** and select the **Add content** icon in the upper right.
2. Choose the **AWS Service Catalog Product** entry. To add it to your catalog home page, choose the first **Add Here** link on the second row of the selection panel at the bottom of the page.

To add AWS Systems Manager automation documents (runbook) to ServiceNow Service Catalog categories

1. Choose **Self Service | Service Catalog** and select the **Add content** icon in the upper right.
2. Select the **AWS Systems Manager** entry. To add it to your catalog home page, choose the first **Add Here** link on the second row of the selection panel at the bottom of the page.

> **Note**
>
> This Connector release displays all AWS Systems Manager documents in the AWS account that has AWS Systems Manager selected.

System administrators can deactivate AWS Systems Manager document requests. To deactivate requests, choose **AWS Systems Manager**, **Automation Documents**, and deselect **Active**. After deactivation of the document, you no longer see the document in the ServiceNow Service Catalog.

The Connector creates closed change requests on post provision actions (such as update, terminate and self-service) for AWS Service Catalog products visible in ServiceNow.

To achieve a closed change request from post provisioned actions, add a change request type and configure the `sys_id` for the group assigned to the closed change records in the Connector AWS Service Catalog system properties.

To add a change request type for closed change request from post provisioned actions

1. If you upgrade from a previous version of the AWS Service Management scoped app, you must remove the **AWS Product Termination** change request type before you create a new change request type.
2. You must add a new change request type called **AWS Provisioned Product Event** for the scoped application to trigger an automated change request in Change Management. For more information, see IT Service Management.
3. Open an existing change request.
4. Open (right-click) the context menu for **Type** and then choose **Show Choice List**.
5. Choose **New** and complete these fields:
   - **Table**: **`Change Request`**
   - **Label**: **`AWS Provisioned Product Event`**
   - **Value**: **`AWSProvisionedProductEvent`**
   - **Sequence**: pick the next unused value
6. Submit the form.

To add a change request type for executing AWS Systems Manager Change Manager change templates

You must add a new change request type called `AWSChangeRequest` for the scoped application to view and execute AWS Change Manager change templates in ServiceNow Change Management. For more information, see IT Service Management.

1. Open an existing change request.

2. Open (right-click) the context menu for **Type** and then choose **Show Choice List**.
3. Choose **New** and complete these fields:
   - Table: **`Change Request`**
   - Label: **`AWS Change Request`**
   - Value: **`AWSChangeRequest`**
   - Sequence: pick the next unused value
4. Submit the form.

To enable AWS Systems Manager Change Manager integration Change models

AWS Systems Manager Change Manager integration in ServiceNow requires Change Model feature in ServiceNow.

1. In the navigator, enter **`sys_properties.list`**.
2. Enter **`*change_model`** in the **Search** panel to view and edit the properties.
3. Review the available settings and recommendations in the table below.

> **Note**
>
> For more information on Change model system properties, see IT Service Management.

Available settings	Desired value
`com.snc.change_management.change_model.hide`	false
`com.snc.change_management.change_model.type_compatibility`	true

ServiceNow Permissions Recap

ServiceNow Persona	Scoped App Permissions	ServiceNow Permission Type	Description
Admin	x_126749_aws_sc_portfolio_manager	Role (scoped app)	Manage AWS Service Catalog portfolios and product access
	x_126749_aws_sc_account_admin	Role (scoped app)	Onboard and manage AWS accounts
	x_126749_aws_sc.appregistry_manager	Role (scoped app)	View AppRegistry applications and attribute groups
	x_126749_aws_sc.automation_manager	Role (scoped app)	Manage Automation Documents and view Automation executions
	x_126749_aws_sc.finding_manager	Role (scoped app)	View AWS Security Hub findings
	x_126749_aws_sc.opscenter_manager	Role (scoped app)	Default access control for OpsItem integration.
	x_126749_aws_sc.change_manager_manager	Role (scoped app)	Manage AWS Systems Manager Change Manager change templates

ServiceNow Persona	Scoped App Permissions	ServiceNow Permission Type	Description
	x_126749_aws_sc.support_case_manager	Role (scoped app)	Manage Support services and categories
	x_126749_aws_sc.productsearchaccess	Role (scoped app)	End user role for searching AWS Service Catalog products using the search widget
	x_126749_aws_sc.cloudtrail_event_user	Role (scoped app)	Default ACL for CloudTrail events on AWS Systems Manager Change Manager
	x_126749_aws_sc.health_dashboard_viewer	Role (scoped app)	View AWS Health dashboard
End User (i.e., Abel Tuter)	Order_AWS_Products	Group	

Configuring AWS Service Management Connector scoped application

After installing and configuring the AWS Service Management Connector, you must configure the scoped application and applicable roles.

To configure the AWS Service Management Connector scoped application permissions

1. In your ServiceNow instance, create a user group called **Order_AWS_Products**.

Members of this group can order Service Catalog products. For instructions, see Administer the Now Platform.

2. Grant ServiceNow permissions to these users:

   - **System Administrator (admin)**: For simplicity in this example, user **admin** is the administrator of the AWS Service Management scoped application. Grant this user both of the administrative permissions from the adapter: **x_126749_aws_sc_account_admin, x_126749_aws_sc_portfolio_manager**, **x_126749_ aws_sc.appregistry_manager**, **x_126749_ aws_sc.automation_manager**, **x_126749_aws_sc.finding_manager**, **x_126749_aws_sc.opscenter_manager**, **x_126749_aws_sc.support_case_manager** and **x_126749_aws_sc.change_manager_manager**, **x_126749_aws_sc.productsearchaccess**, **x_126749_aws_sc.cloudtrail_event_user**, and **x_126749_aws_sc.health_dashboard_viewer**.

     Add **System Administrator** to the new ServiceNow group **Order_AWS_Products**. In a real scenario, these roles would likely be granted to different users or groups.
   - **Abel Tuter**: The user **abel.tuter** is an illustrative end user. Grant Abel the new role **Order_AWS_Products**. This permission allows Abel to order products from AWS.

Configuring AWS accounts to synchronize in the Connector

Learn how to configuring AWS accounts to synchronize in the Connector.

1. Log in as the system administrator.
2. Enter **AWS** in the navigator. Choose the **AWS Service Management** scoped app.
3. In the **Accounts** menu, create one entry for every AWS account. Use the keys and secret keys from the users you created in AWS.

To create an account entry

1. Enter the name as an account entry identifier, such as **Connector_Demo** (for Commercial Region), or **Connector_Demo_GovCloud** (for GovCloud Region).
2. Enter the access key and secret access key from the AWS account *sync user* IAM configurations.
3. Enter the access key and secret access key from the AWS account *end user* IAM configurations.
4. Choose the visible AWS service integrations for this AWS account. The choices include:

   - Integrate with Service Catalog (including AppRegistry)

- Integrate with AWS Config

  Choose AWS Config if you plan to integrate AWS Config cloud resources per each AWS account or through the latest AWS Config aggregator integration feature. The Connector for ServiceNow includes an AWS Config aggregator feature that enables ServiceNow administrators to align aggregated AWS Config details into one AWS account.

  If you plan to view AppRegistry related resources details, choose **AWS Config** with **AWS Service Catalog**.

- Integrate with AWS Systems Manager Automation

  Choose AWS Systems Manager Automation if you want to execute automation documents (runbook) to remediate incidents from OpsItems.

- Integrate with AWS Systems Manager OpsCenter
- Integrate with AWS Security Hub
- Integrate with Support
- Integrate with AWS Systems Manager Change Manager
- Integrate with AWS Health
- Integrate with AWS Systems Manager Incident Manager

5. Choose **Account Regions**. Select the **Commercial** or **GovCloud Region**. To see the AWS account Regions, double-click **Insert a new row...**.

   > **Note**
   >
   > AWS Support API uses a specific GovCloud endpoint for GovCloud accounts to enable Support integration for GovCloud accounts. Choose a GovCloud Region in Account Regions when you onboard the account in ServiceNow.

6. Repeat the step above to insert additional Regions.
7. Save or update the account entries.
8. Validate AWS account connectivity by following the steps in Validating connectivity to AWS Regions. Note that in this Connector for ServiceNow, **Validate Accounts** only appears once after you submit or update the account entry.

> **Note**
>
> AWS Service Management Connector allows synchronization of updated keys using any automation or integration through a REST endpoint. For more information, see Syncing updated keys programatically in ServiceNow.

Validating ServiceNow connectivity to AWS Regions

You can now validate connectivity to AWS accounts between the ServiceNow **Connector_Demo** account and the AWS IAM `SMSyncUser` and `SMEndUser`.

To validate connectivity to AWS account

1. In the AWS Service Management scoped app, choose **Setup**, then **AWS Accounts**.
2. Choose **Connector_Demo** and select **Validate Account**.

   A successful connection results in the message, *Successfully validating AWS account in each referenced Region*.

If the AWS IAM access key or secret access key are incorrect, you receive an error message.

Manually syncing scheduled jobs

The Connector for ServiceNow includes nine sync jobs related to AWS services integrations. During the initial setup, manually execute the sync job for your AWS service integration instead of waiting for Scheduled Jobs to run.

To sync AWS service integrations or accounts manually

1. Log in as system administrator.
2. Find **Scheduled Jobs** in the navigator panel.
3. Search the following AWS Service Management Connector scheduled jobs (including default sync intervals) in the table below:

AWS Service Management Scheduled Job Name	Brief description	Default Sync Interval
Sync all Automation Execution	Syncs execution of AWS Systems Manager Automation runbooks (documents)	5 minutes
Sync all provisioned AWS Service Catalog products	Syncs latest status of provisioned AWS Service Catalog products	5 minutes
Sync all ServiceNow resources to AWS Config	Syncs ServiceNow resources mapped to AWS Config custom resources	6 Hours
Synchronize changes to all AWS Accounts	Syncs changes to AWS services opted into each AWS account associated to the Connector	1 Day
Synchronize AWS Config	Syncs resource details or relationships from AWS Config into the ServiceNow CMDB	31 minutes
Synchronize AWS Security Hub	Syncs bi-directionally security findings from AWS Security Hub to ServiceNow incidents or problems	31 minutes
Synchronize AWS Service Catalog	Syncs AWS Service Catalog products into ServiceNo w Service Catalog request items	31 minutes

AWS Service Management Scheduled Job Name	Brief description	Default Sync Interval
Synchronize AWS Systems Manager Automation	Syncs AWS Systems Manager Automation runbooks (documents) into ServiceNow Service Catalog request items	31 minutes
Synchronize AWS Systems Manager OpsCenter	Syncs bi-directionally OpsItems from AWS Systems Manager OpsCenter to ServiceNow incidents	31 minutes
Synchronize AWS Support Cases through SQS	Syncs Support Cases created or updated from AWS into ServiceNow	1 min
Synchronize status of synced Support Cases	Syncs status of Closed Incidents from AWS to ServiceNow	6 hours
Synchronize AWS Systems Manager Change Manager	Syncs pre-approved Change templates and Change Requests from AWS to ServiceNow	31 min
Synchronize AWS Systems Manager Incident Manager	Syncs Incident Manager incidents from AWS to ServiceNow	1 min
Synchronize AWS Health	Syncs Health events and resource information from AWS to ServiceNow	5 min
Synchronize Amazon WorkSpaces	Syncs Amazon WorkSpaces resource type from AWS Config	31 min

4. Choose the desired sync job, and choose **Execute Now**.

> **Note**
>
> If you do not see **Execute Now** in the upper left corner, choose **Configure Job Definition**. **Execute Now** is visible. ServiceNow Administrator can adjust the Scheduled Job repeat interval as required.

Data is visible in the AWS Service Management scoped app menus after the Connector's scheduled synchronization job has run.

AWS Service Catalog in ServiceNow

After you create two IAM users with baseline permissions in each account, the next step is to configure AWS Service Catalog.

Use the Amazon S3 template in Creating an Amazon S3 Bucket for Website Hosting for your preliminary product. Copy and save the Amazon S3 template to your device.

For an interactive workshop using ServiceNow, review the ServiceNow Connector workshop in the *AWS workshop studio*.

Topics

- Configuring AWS Service Catalog
- Configuring AWS Service Catalog in ServiceNow
- Using service integration features to validate AWS Service Catalog integration in ServiceNow
- Viewing products in the Standard User Interface (Fulfiller View)
- Ordering Service Catalog products through the ServiceNow Service portal

Configuring AWS Service Catalog

This section provides the configurations you need to integrate AWS services in ServiceNow.

To configure Service Catalog

1. Follow the steps to create a Service Catalog portfolio.

2. To add the Amazon S3 bucket product to the portfolio you created in Step 1, go to the Service Catalog console. In the **Upload new product** page, enter the product details.
3. For **Select template**, choose the Amazon S3 bucket AWS CloudFormation template you saved to your device.
4. Set **Constraint type** to **Launch** for the product that you created now with the `SCConnectLaunch` role in the baseline permissions. For additional launch constraint instructions, see AWS Service Catalog Launch Constraints.

   > **Note**
   >
   > The AWS configuration design requires each Service Catalog product to have a launch constraint. Failure to follow this step could result in an *Unable to Retrieve Parameter* message in the ServiceNow Service Catalog.

5. Add the SMEndUser user to the Service Catalog portfolio. For additional user access instructions, see Granting Access to Users.

> **Note**
>
> The AWS configuration design requires each Service Catalog product to have either a launch constraint or a stack set constraint. Failure to follow this step could result in an *Unable to Retrieve Parameter* error in the ServiceNow Service Catalog.

Creating StackSet constraints

AWS CloudFormation StackSets enable users to create and deploy products across multiple accounts and Regions.

To apply a stack set constraint to a Service Catalog product

1. As a catalog admin in Service Catalog, choose the portfolio that contains the product.
2. Expand **Constraints** and choose **Add constraints**.
3. Choose the product from **Product** and set **Constraint type** to **Stack Set**. Choose **Continue**.
4. On the StackSet constraint page, enter a description.
5. Choose the account(s) in which you want to create products.

6. Choose the Region(s) in which you want to deploy products. Products deploy in these Regions in the order you specify.
7. Choose the following:

   `AWSCloudFormationStackSetAdministrationRole` to manage your target accounts.

   `AWSCloudFormationStackSetExecutionRole` for the role the Administrator will assume.
8. Choose **Submit**.

Relating budgets to products and portfolios

The Connector for ServiceNow enables ServiceNow administrators to view budgets related to Service Catalog products and portfolios. Service Catalog administrators can create or associate existing budgets to products and portfolios.

For more information on creating and associating budgets, see Managing Budgets.

Service Catalog Terraform Open Source product type support

AWS Service Management Connector supports AWS Service Catalog's Terraform open source product type. For more information, review Getting started with Terraform product in the *AWS Service Catalog admin guide*.

As of the 4.8.5 release, you can provision AWS Service Catalog products and their resources using either AWS CloudFormation or Hashicorp Terraform (Terraform open source).

The **AWS CloudFormation** product type in AWS Service Catalog allows you to request provisioning, create provisioned product plans, perform self-service actions, and request termination or update for the provisioned product. The connector also dynamically makes API calls to list available parameters such as VPC ID, Subnet IDs, and Security Groups in a drop down format.

When provisioning fails for a AWS CloudFormation product, the provisioned product **Status** changes to `TERMINATED`.

The **Terraform open source** product type in AWS Service Catalog allows you to request provisioning for Terraform open source products as well as request termination or update for the provisioned product.

> **Note**
>
> The Terraform open source product type does not support self-service actions and provisioned product plans.

When the provisioning fails for a Terraform open source product, the provisioned product **Status** changes to `TAINTED`.

Configuring AppRegistry

To configure AppRegistry, follow the steps in the AWS Service Catalog AppRegistry Administrator Guide.

Configuring AWS Service Catalog in ServiceNow

This section provides the configurations you need to integrate AWS Service Catalog in ServiceNow.

Topics

- Configuring the AWS Service Catalog product widget components and assignment group for closed change records
- Granting access to AWS Service Catalog portfolios
- Configuring AWS tags for provisioned products
- Adding the My AWS Products widget to the Service Portal view
- Activate AWS Service Catalog portfolio categorization in ServiceNow Service Portal
- Viewing budgets related to Service Catalog portfolios and products

Configuring the AWS Service Catalog product widget components and assignment group for closed change records

To address the varying personas of end users requesting AWS products, the Connector for ServiceNow includes a scoped app setting to enable or disable components of the AWS product widget. By default, all AWS product components are active.

To modify the AWS product view

1. In the navigator, enter **`System Properties`** and select **Service Catalog**.

> **Note**
>
> Make sure you are in the AWS Service Management Connector scoped application mode.

2. Deselect any AWS product component to enable:
   - Editing of the Service Catalog product name.
   - Selection of launch options for Service Catalog Products. (This component is only visible if the AWS product has more than one launch path.)
   - Selection of product versions for Service Catalog. (This component is only visible if the AWS product has more than one product version.)
   - Tags for Service Catalog products.
   - Plans (ChangeSet) creation for product. (If set to false the plan section is not visible.)
3. Choose **Save**.

The AWS Service Catalog system properties also include a section that identifies an assignment group. This group associates with closed change records from post provision actions of products (such as terminate, update, or self-service actions).

To associate the assignment group for change records from AWS Service Catalog post provision actions

1. In the navigator, enter **`System Properties`** and choose **AWS Service Catalog**. Make sure you are in the AWS Service Management Connector scoped application mode.
2. Choose the section **Set the 'assignment group' sys_id or name that the connector will use when creating change requests**.
3. Enter the assignment **`group sys_id`**.

   If you need to find the `group sys_id`, enter **`System Security`** in the left navigator.
4. Choose **Groups** module.
5. Search for the **Group** name.
6. Choose the group that you want to associate to close changed records and choose **Copy sys_id**. You are now able to paste the copied `sys_id` into the AWS Service Catalog Properties

for the Connector under **Set the 'assignment group' sys_id or name that the connector will use when creating change requests**.

If the `sys_id` is blank, the change record sends a message that no assignment group exists for the record, which causes change requests created from the Connector to be in an open state.

Granting access to AWS Service Catalog portfolios

This release of the Connector does not require you to link AWS identities to ServiceNow roles. To grant access to Service Catalog products in ServiceNow, you must establish a link between the Service Catalog portfolios and the ServiceNow group (for example, **Order_AWS_Products** from an earlier installation example).

To grant access to Service Catalog portfolios in ServiceNow

1. In the AWS Service Management scoped app, choose **Service Catalog**, then the **Portfolios** module.
2. Choose the desired Portfolio ARN. You can double-click the Service Catalog portfolio name.
3. Choose the **Allowed Groups** tab.
4. Choose **New** and enter the **Group** named **`Order_AWS_Products`**.
5. Choose **Submit**.

Configuring AWS tags for provisioned products

The AWS Service Management Connector enables ServiceNow administrators to add tags (metadata) to provisioned products globally across the scoped app or granularly at the portfolio level. These tags are not visible to end users.

Three tag types are available in this release:

- Generic tags in which the administrator can enter the key and value.
- ServiceNow Request Item tags in which the admin can enter the syntax for Key and Value in the table below.
- ServiceNow table(s) values that end users can select as tags for provisioned AWS resources. This release now enables administrators to identify any ServiceNow tables, such as Cost center or Department, and makes values from that table selectable for end users.

> **Note**
>
> Generic tags (from administrators) and ServiceNow Request Item tags are not viewable by end users.

Key	Value
Requested Item Number	${REQUEST_NUMBER}
User	${USERNAME}
Requested for	${REQUESTED_FOR}
Opened by	${OPENED_BY}

To add generic AWS tags to Service Catalog provisioned products in ServiceNow

1. In the AWS Service Management scoped app, choose **Setup**, then the **Automated Tags** module.
2. Choose **New**.
3. For Global tags, enter the Key and Value entries and choose **Submit**.
4. For Portfolio tags, deselect **Global check**. The **Portfolio** field appears.

   Choose the Service Catalog portfolio, enter the Key and Value entries, and choose **Submit**.

To add in-scope ServiceNow request item AWS tags to Service Catalog provisioned products derived from ServiceNow

1. In the AWS Service Management scoped app, choose **Setup**, then the **Automated Tags** module.
2. Choose **New**.
3. For Global tags, enter the specific Key and Value entries for either User or Request Item Number, and choose **Submit**.

4. For Portfolio tags, deselect **Global check**. The Portfolio field appears. Select the AWS Service Catalog portfolio, enter the Key and Value entries, and choose **Submit**.

To add tags to AWS provisioned products from ServiceNow tables and fields that are selectable by end users

1. In the AWS Service Management scoped app, choose **Setup**, then the **Automated Tags** module.
2. Choose **New**.
3. Choose **Selectable by End User**.
4. Choose a table from the dropdown list: **Table Name**.
5. Choose a field from the dropdown list: **Table Field**.
6. [Optional] Add a filter for the table selected using the **Table Filter** field.
7. For Global tags, enter the Key and Value entries and choose **Submit**.
8. For Portfolio tags, deselect **Global check**. The **Portfolio** field appears.

   Select the AWS Service Catalog portfolio, enter the Key and Value entries, and choose **Submit**.

   The ServiceNow table and field value appear on the AWS Product (ServiceNow catalog item). It is a required value prior to ordering. After product provisioning, you can see in the AWS console that these tags associate with the resource.

Adding the My AWS Products widget to the Service Portal view

We recommend ServiceNow administrators add the **My AWS Products** widget to the ServiceNow Portal view. The widget enables users to view their AWS product requests, view outputs, and perform post-operational actions such as update, terminate, and service actions (AWS Systems Manager documents).

To include the My AWS Products widget on the Service Portal view

1. Log in as system administrator in the ServiceNow standard user interface (Fulfiller view).
2. In the navigator panel, find **Service Portal**.
3. Choose **Service Portal Configuration**.
4. Choose **Designer**.

5. Search for **Service Portal** in the filter.
6. Choose the **Service Portal** box with a house image and the word **Index** in the lower right corner.
7. In the left panel in **Widgets**, enter **`My AWS Products`** in the **Filter Widget.**
8. Drag the widget to the Service Portal edit view to your desired location.
9. Preview your changes.

To include the Search AWS Products widget on the Service Portal view

1. Log in as system administrator in the ServiceNow standard user interface (Fulfiller view).
2. In the navigator panel, find **Service Portal**.
3. Choose **Service Portal Configuration**.
4. Choose **Designer**.
5. Search for Service Portal in the filter.
6. Choose the Service Portal box with a house image and the word Index in the lower right corner.
7. In the left panel in **Widgets,** enter **`AWS Custom Product Search in the Filter Widget`**.
8. Drag the widget to the Service Portal edit view to your desired location.
9. Preview your changes.

> **Note**
>
> Ensure that the end user has **x_126749_aws_sc.productsearchaccess** to view and use the widget.

Activate AWS Service Catalog portfolio categorization in ServiceNow Service Portal

AWS Service Management Connector can display portfolios with an additional categorization of AWS Account and Region names in the ServiceNow Service Portal. This allows you to identify the account and region a portfolio and its product belongs to if the end user has access to multiple portfolios with the same name.

To activate Portfolio categorization in ServiceNow Portal

1. Log in as system administrator.
2. In the **System Properties** menu, choose **AWS Service Catalog**.
3. In the option **If set to Account/Region/Portfolio, the hierarchy of categories created will be set to portfolio, region and account. If set to Portfolio, only portfolio category will be created**, choose **Account/Region/Portfolio**.
4. In the **System Definition** menu, choose **Scheduled Jobs**.

To activate Portfolio categorization for existing users

1. In the **System Definition** menu, choose **Scheduled Jobs**.
2. Select the scheduled job, and then choose **Synchronize AWS Service Catalog**.
3. In the **Active** field, choose **False**, and then choose **Update**.
4. In the **System Definition** menu, choose **Fix Script**.
5. Select the fix script, and then choose **AWS Service Catalog Category Delete**, and then choose **Run Fix script**.
6. Follow the steps in *To activate Portfolio categorization in ServiceNow Portal* above.

Viewing budgets related to Service Catalog portfolios and products

ServiceNow administrators can view budgets and actual costs related to Service Catalog portfolios and products in the ServiceNow standard user interface.

To view portfolio budgets

1. Log in as system administrator.
2. In the navigator panel, search for **Service Catalog**.
3. Choose the **Portfolios** module.
4. Choose the Service Catalog portfolio that contains an associated budget.
5. Choose the **Budget** tab.

To view product budgets

1. Log in as system administrator.

2. In the navigator panel, search for **Service Catalog**.
3. Choose the **Products** module.
4. Choose the Service Catalog product that contains an associated budget.
5. Choose the **Budget** tab.

Using service integration features to validate AWS Service Catalog integration in ServiceNow

This section describes how you can use service integration features to validate AWS Service Management Connector for ServiceNow installation.

To order a Service Catalog product

1. Log in to your ServiceNow instance as the end user (for this example, Abel Tuter).
2. Enter **`Service Catalog`** in the navigation filter and choose **Service Catalog**.
3. Choose the **AWS Service Catalog S3 Storage** product to provision.
4. Enter the product request details, including product name, parameters, and tags.
5. Choose **Order Now** to submit the ServiceNow request and provision the Service Catalog product.

   After approximately one minute, you receive an order status acknowledging the submission.

To view provisioned products

End users can view products in two places on the ServiceNow portal: **request items (Requests)** or **My AWS Service Catalog Products** widgets.

To view products in Service Portal Requests

1. Choose **Requests** in the home page navigation bar.
2. Choose the request item with the Service Catalog product and request the item number.

> **Note**
>
> AWS product events and outputs update the request item. When you terminate the AWS product, the ServiceNow request item enters a state of **Closed Complete**.

To view products in the My AWS Products widget Service Portal Requests

1. In the **My AWS Products** widget, choose the AWS Select product name on the request form.
2. View **Status and Product Events**.
3. If you want to perform post-provisioned operational actions, choose **Request Update**, **Request Self-Service Action**, or **Terminate**.

To override workflows on Portfolios

1. Log in to your ServiceNow fulfiller view (standard user interface).
2. Enter **`AWS Service Catalog`** in the navigation filter and choose **Portfolios**.
3. Choose **Display Name** to open a portfolio.
4. Select the required workflow from the search to set **Workflow Override.**
5. Choose **Update**.

To view AppRegistry applications

1. Log in to your ServiceNow fulfiller view (standard user interface).
2. Enter **`AWS Service Catalog`** in the navigation filter and choose **AppRegistry Applications**.
3. Choose the AppRegistry application.

To view AppRegistry attribute groups

1. Log in to your ServiceNow fulfiller view (standard user interface).
2. Enter **`AWS Service Catalog`** in the navigation filter and choose **AppRegistry Attribute Groups**.
3. Choose the AppRegistry attribute group.

Video: Integrate AWS Products into Your ServiceNow Portal with the AWS Service Management Connector

This video (18:33) describes how to integrate AWS products in your ServiceNow Portal with the AWS Service Management Connector.

Integrate AWS Products into Your ServiceNow Portal with the AWS Service Management Connector

Viewing products in the Standard User Interface (Fulfiller View)

View provisioned products as an end user and from the scoped app as an administrator.

To view provisioned products as an end user

1. Choose **My Assets** in the ServiceNow standard user interface.
2. In **My Asset Requests**, view the requests.
3. To view the product, personalize the list view to show the associated configuration item.

   To show the items, choose **Settings** in the header row of the table of asset requests.
4. Choose **Configuration item (configuration_item)**. Then use the **>** icon to add it to the view.

   Move the configuration item to below **Stage** in the list. You can see it (the ordered product) in the list of assets.
5. To view the product, choose the configuration item name.
6. In the **Outputs** tab of the form, view the **Outputs** for the provisioned product.
7. In the **Product Events** tab of the form, view the provisioning history of the product .

To view provisioned products from the scoped app as an administrator

1. Log in to your ServiceNow instance as the end user (for example, Abel Tuter).
2. Enter **`Service Catalog`** in the navigation filter and choose **Provisioned Products**. The user interface view displays the provisioned products.
3. Choose a provisioned product to view the current status. You can also select post provisioned actions such as **Request Update**, **Request Termination**, as well as associated service actions.

Ordering Service Catalog products through the ServiceNow Service portal

The Connector for ServiceNow supports the ordering of Service Catalog products through Service Portal. You can use the **Service Catalog** and **Order Something** views. The release also includes pages and widgets you can add to Service Portal that enable users to view their provisioned products.

> **Note**
>
> The audience for the Service Portal Features section is a ServiceNow administrator or equivalent. The ServiceNow user requires permissions to modify the Service Portal.

Service portal widgets

The Connector for ServiceNow includes widgets you can add to your Service Portal. It also includes two alternative view Portal Pages for the following:

- **My AWS Products** – Overview of all provisioned products the user owns
- **AWS Product Details** – Details of a single provisioned product
- **Search AWS Products** – Search for AWS Service Catalog products by providing AWS account, Region, and portfolio details. To access the new widgets, update the Service Portal Designer.

To access the new widgets, update the Service Portal Designer.

To update the Service Portal Designer

1. Go to Create and edit a page using the Service Portal Designer.
2. Following the instructions, choose the **Service Portal Index** page.
3. Under the **Order Something** container, add the **My AWS** widget.

   The new widget appears on your main Service Portal view.

Service portal pages

This section describes the two new pages available in the Service Portal Beta release of the AWS Service Management Connector: **My AWSProducts** and **AWS Product Details**. You can add links to these pages on the Service Portal home page or other pages by using the usual page configuration mechanism in Service Portal.

My AWS Products

An overview of all provisioned products that the user owns. Terminated products display separately from current products in a collapsed panel on the initial page load.

Use the following format to access the **My AWS Products** page.

```
http://<insertinstancename>.service-now.com/sp?id=aws_sc_pp
```

AWS Product Details

Details of a single provisioned product.

Use the following format to access the **AWS Product Details** page:

```
http://<insertinstancename>.service-now.com/sp?id=aws_sc_pp_details&sys_id=<provisioned
 product id>
```

Search AWS Products

Search feature for AWS Service Catalog products

Use the following format to access the **Search AWS Products** page:

```
        http://<insertinstancename>.service-now.com/sp?id=aws_sc_product_search>
```

> **Note**
>
> Ensure that the end user has **x_126749_aws_sc.productsearchaccess** to view and use this service portal

AWS Config in ServiceNow

This section shows you how to use AWS Config to integrate to ServiceNow.

To allow the Connector to synchronize Config data for a given Region, you must enable AWS Config in that Region. For more information, see Setting Up AWS Config with the Console.

AWS Service Management Connector for ServiceNow enables ServiceNow administrators to specify select ServiceNow tables as custom resources within AWS Config.

To set up these resources, use the preconfigured files in the Connector. These required files include the custom resource schema.

Topics

Configuring system properties, aggregators, and custom resources

This version of the AWS Service Management Connector enables ServiceNow administrators to configure system properties, Config Aggregators, and AWS Config custom resources from select ServiceNow tables.

To configure the new AWS Config integration System properties

1. In the navigator, enter **`AWS Service Management`**.
2. Choose **System Properties**, and then choose **AWS Config**.
3. Review the available settings and recommendations in the table below.

Available settings	Description
The name of the S3 bucket from where to get the resource provider ZIP files	The S3 bucket for custom resources from ServiceNow that populates AWS Config. Default and hard coded value: `cmdb-resource-providers` **Note** Service Management Connector recommends that you do not change this setting.
Name of the Discovery source for synchronization with AWS Config	The setting that correlates the Discovery source in ServiceNow. Default and hard coded value: AWS Service Management Connector

Available settings	Description
	Note Service Management Connector recommends you do not change this setting.
What field to use for correlation ID	Administrators use this setting to specify which column contains the correlation ID for each AWS Config. The correlation ID disambiguates AWS Config item that might have the same resource ID (such as SQS queues). It consists of the comma separated string of: • Source account number • Source Region • Resource type, such as AWS::EC2::Instance • Resource ID Default: `correlation_id`
What field to use for AWS capture time	Administrators use this setting to specify which column contains the capture time (such as capture time from AWS Config) for each AWS Config item. Default: `last_discovered`
What field to use for last sync time	Administrators use this setting to specify which column contains the last sync time (such as the last time AWS Config integration performed a synchroni zation for a given item) for each AWS Config item. Default: `checked_in`

Available settings	Description
Enable the creation of a relationship for state sync	Administrators use this setting to enable the creation of a relationship to a special *state sync* configuration item. When enabled, each synchronized item links to a particular state sync, or execution. By enabling this feature, it allows the SMC to identify stale items. **Warning**: This action creates an additional relationship per synchronized item. Depending on the number of items, it might have a performance impact. Default: No
Enable the deletion of the previous relationship for state sync	Administrators use this setting to enable the deletion of previous relationships to a special *state sync* configuration item. When enabled, a successful synchronization to a given AWS Config time deletes the previous relationships to state sync configuration item. **Warning:** This action performs `GlideAggregate` queries for each group of synchronized accounts, Regions, or Aggregators. Depending on the number of items, it might have a performance impact. Default: No

Available settings	Description
What *Install status* to put stale config item into	Administrators use this setting to automatically change the *install_status* of configuration items identified as stale. This action ensures that the status of stale resources correctly updates when using an Aggregator. Be aware this feature works only if you set *What field to use for last sync time* and enable *Enable the creation of a relationship for state sync*. Allowed values: • Installed • Retired • Absent • Do nothing Default: Do nothing
Interval in minutes between the execution of full Config synchronization	Administrators use this setting to control the time between full syncs of Config data. The default is 720 minutes or 12 hours.
Use MTM for managing stale status	This setting ensures the use of separate tables for handing relationships for sync status instead of using the `cmdb_rel_ci` table. AWS Service Management Connector recommends using the default setting. Default: Yes

Validating the synchronization of Amazon WorkSpaces from AWS Config

Validate the synchronization of Amazon WorkSpaces in AWS Config by executing a scheduled job.

To validate the synchronization of Amazon WorkSpaces in AWS Config

1. Execute the scheduled job **synchronize Amazon WorkSpaces** manually.
2. Navigate to **AWS Config**, and then choose **WorkSpaces**.
3. Validate the data.

> **Note**
>
> Amazon WorkSpaces synchronization is only supported for stand-alone accounts, not for AWS Config Aggregator accounts.
> The **SyncUser** role must include the `DescribeWorkSpacesPolicy` for the synchronization to execute successfully.

Addressing stale AWS Config items in the ServiceNow CMDB

> **Note**
>
> ServiceNow administrators are the target audience for this section.

In addition to the AWS Config settings, AWS SMC for ServiceNow now exposes a global API to identify stale config items from the AWS Config integration.

Stale Config items are the existing AWS Config items that did not update during the most recent sync for the same source (such as account, Region, and Aggregator name).

> **Note**
>
> This feature requires you to enable the creation relationship to sync the status setting in the AWS Config System Properties in the ServiceNow scoped app.

The script includes `x_126749_aws_sc.AwsSmc` and exposes a public API. You can use this script to access any application scope, including *global* scope. As an example, run this script:

```
   x_126749_aws_sc.AwsSmc.asSyncUser().getStaleConfigItems().forAll(function(object)
{
  gs.info(
       object.accountNumber + '/' + object.region + ' '
       + (object.aggregatorName ? 'aggregator: ' + object.aggregatorName + ' ' :
'')
       + 'ci: ' + object.ci.name
       + ' - ' + object.ci.getDisplayValue('install_status')
  );
});
```

As a background script, it would log the following:

```
Info: 11111111/us-east-1 ci: i-1234567fg6j8 - Installed
Info: 11111111/us-west-1 ci: i-9876541fdgfd - Installed
Info: 22222222/eu-west-1 aggregator: all-dev ci: i-1df5235ftt55 - Installed
```

Each *object* contains the properties below:

Property	Type	Description
`accountNumber`	String	The account number from which the stale config item originates.
`region`	String	The Region from which the stale config item originates.
`aggregatorName`	String	The Aggregator name (if applicable) from which the stale config item originates.
`lastSynced`	`GlideDateTime`	The `GlideDateTime` of the when the last synchronization occurred.
`CI`	`GlideRecord`	The `GlideRecord` of the stale config item.

Optionally, you can also pass an `options` object as the second argument to the `forAll` method that allows you to customize the search for stale items.

Property	Type	Description
`lowerTimeLimit`	`GlideDateTime`	The threshold `GlideDateTime` from when you should search items. Any stale item last updated prior to that date does not return.
`upperTimeLimit`	`GlideDateTime`	The threshold `GlideDateTime` until you should search for items. Any item last updated after that date does not return.
`excludeStatus`	Number	The `install_status` to filter on.

Timestamps of sync resources:

- `LastSyncTimeField`(default `checked_in`): The start of the current sync process.
- `first_discovered` (for new records): The current time. We set the `LastDiscoveredField` (default `last_discovered`) to the `configurationItemCaptureTime` of the resource, if it exists or is undefined.

Additional notes on stale records

When AWS Service Management Connector reads AWS Config records that refer to other resources, it often creates a relationship to those resources.

In some cases, the related resource does not have an entry in the ServiceNow CMDB. In these cases, the Connector creates a record for that relationship, with an install status of *absent*. When the Connector reads the AWS Config record for the related resource, that record populates.

To see active resources, you should filter ServiceNow records synced from AWS Config by an install status of *not Absent*.

Disclaimer

Because the script compares items linked to stale sync records, it is unable to identify stale resources synced before the installation of this SMC version. When switching to sync with an Aggregator or switching from Aggregator sync to non-Aggregator sync, the script also fails to detect items that became stale between the last non-Aggregator sync and the first Aggregator sync.

Configuring synchronization of AWS Config data using an Aggregator in ServiceNow CMDB

Prerequisite: You need to opt-in and configure the AWS account that contains the aggregated AWS Config resources details prior to performing the steps below. For more information, see Configuring AWS Accounts to Synchronize in the Connector.

To configure the Connector to use an Aggregator to synchronize AWS Config data

1. In the AWS Service Management scoped app, choose the **Setup** module.
2. Choose **Aggregators for AWS Config**.
3. Choose **New**.
4. Enter the name of the new Config Aggregator.
5. Choose the Region where you created the new Config Aggregator.
6. Choose the AWS account that should use the new Aggregator. Only AWS accounts opted into the Connector for ServiceNow that have **Integrate with AWS Config** are viewable.
7. Choose **Submit**.

   If you define an Aggregator for an AWS account and Region, the Aggregator integration becomes the only AWS Config to ServiceNow CMDB synchronization mechanism for that AWS account.

The Connector can now synchronize Config data from multiple accounts and Regions using an Aggregator. You must configure the Config Aggregator in AWS before using this feature. For more information, see Setting up an Aggregator in the console.

> **Note**
>
> The Config Aggregator view in AWS displays only current config item resources in AWS Config. Thus, terminated resources are not available in the Config Aggregator view.

To minimize stale config item records from rendering in the ServiceNow CMDB from the AWS Config Aggregator, we recommend you remove Config rules associated to terminated resources. For more information, see Evaluating Resources with AWS Config Rules

Configuring available ServiceNow tables to sync as AWS Config custom resources

In this Connector for ServiceNow release, you can now sync a set of ServiceNow tables in the CMDB to AWS Config as custom resources.

The ServiceNow tables and AWS Config custom resource mapping are as follows:

ServiceNow CMDB table	AWS custom resource
`cmdb_ci_apache_web_server`	Apache Web Server
`cmdb_ci_app_server`	Application Server
`cmdb_ci_app_server_java`	Java Server
`cmdb_ci_app_server_tomcat`	Tomcat Server
`cmdb_ci_app_server_tomcat_war`	Tomcat Web Application
`cmdb_ci_app_server_websphere`	IBM Websphere Application
`cmdb_ci_app_server_ws_ear`	Websphere Enterprise Archive
`cmdb_ci_appl`	Application
`cmdb_ci_appl_dot_net`	A .Net Application
`cmdb_ci_appl_now_app_comp`	ServiceNow Application Component
`cmdb_ci_appl_sap`	SAP Application
`cmdb_ci_appl_sap_hana_db`	SAP Hana Database
`cmdb_ci_appl_sap_system`	SAP System

ServiceNow CMDB table	AWS custom resource
cmdb_ci_appl_sharepoint	Microsoft Sharepoint Application
cmdb_ci_application_cluster	Application Cluster
cmdb_ci_application_server_resource	Application Server Resource
cmdb_ci_application_software	Application Software
cmdb_ci_db_mssql_database	MySql Database
cmdb_ci_db_mysql_instance	MySql Instance
cmdb_ci_kubernetes_cluster	Kubernetes Cluster

To configure ServiceNow tables as AWS Config custom resources

> **Note**
>
> When you configure ServiceNow tables as AWS Config custom resources you might encounter an increase in your billing statement for the creation of additional resources.

1. In the navigator, enter **`AWS Service Management`**.
2. Choose **Setup**, then **Tables Sync to AWS Config**.
3. Choose **New**.
4. Choose an in scope ServiceNow table.
5. Choose an account and Region for the new resource type. You can select any supported Region, in addition to preconfigured Regions for the account.
6. Click **Submit**.
7. Repeat steps above to include additional ServiceNow tables available to sync as AWS Config custom resources.

   The amount of time to create new AWS Config resources depends on the number of ServiceNow tables you selected. You can see resources in the **Schema version** field upon

successful completion. The period synchronization of resources automatically includes the new AWS Config custom resource type. As details in the ServiceNow table update, this information syncs to AWS Config custom resource.

Validating AWS Config integration in ServiceNow

To see AWS Config details, configure the service settings to record data for the resource types of interest. For more information, see Setting Up AWS Config with the Console.

To view configuration item details from AWS Config in the ServiceNow CMDB

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (Standard user interface view).
2. In the navigator, enter **`AWS Service Management`**.
3. Choose **AWS Config**. Select and view the relationships for available AWS resources.

This table illustrates the available AWS resources, ServiceNow CMDB label, and table name.

AWS resources (AWS Config)	ServiceNow CMDB/Scoped App Table Label	ServiceNow CMDB/Scoped App Table Name
Accounts	CMDB CI Cloud Service Accounts	`cmdb_ci_cloud_service_account`
VPCs	Cloud Networks	`cmdb_ci_network`
Availability Zones	Availability Zone	`cmdb_ci_availability_zone`
EC2 Instances	Virtual Machine Instance	`cmdb_ci_vm_instance`
EBS Volumes	Storage Volume	`cmdb_ci_storage_volume`
Security Groups	Compute Security Group	`cmdb_ci_compute_security_group`

AWS resources (AWS Config)	ServiceNow CMDB/Scoped App Table Label	ServiceNow CMDB/Scoped App Table Name
Auto Scaling Group	Auto Scaling Groups	x_126749_aws_sc_cmdb_ci_autoscaling_group
Network Interfaces	Cloud Mgmt Network Interface	cmdb_ci_nic
RDS Instances	Cloud DataBase	cmdb_ci_cloud_database
Subnets	Cloud Subnet	cmdb_ci_cloud_subnet
Load Balancers (V2)	Cloud Load Balancer	cmdb_ci_cloud_load_balancer
S3 Buckets	Cloud Object Storages	cmdb_ci_cloud_object_storage
CloudFormation Stacks	CloudFormation Stack	x_126749_aws_sc_cmdb_ci_cloudformation_stack
CloudFormation Provisioned Products	CloudFormation Provisioned Product	x_126749_aws_sc_cmdb_ci_config_pp
Tags	Key Value	cmdb_key_value
Lambdas	Cloud Function	cmdb_ci_cloud_function
Dynamo DB	DynamoDB Table	cmdb_ci_dynamodb_table
OS images	Images	cmdb_ci_os_template

AWS resources (AWS Config)	ServiceNow CMDB/Scoped App Table Label	ServiceNow CMDB/Scoped App Table Name
AppRegistry Applications	AppRegistry Application	`x_126749_aws_sc_cmdb_ci_appregistry_application`
AppRegistry Attribute Groups	AppRegistry Attribute Group	`x_126749_aws_sc_cmdb_ci_appregistry_attribute_group`
AppRegistry Resources	AppRegistryResource	`x_126749_aws_sc_cmdb_ci_appregistry_resource`
RDS Cluster	Cloud Database Clusters	`cmdb_ci_cloud_db_cluster`
API Gateway	Cloud Gateways	`cmdb_ci_cloud_gateway`
Amazon Workspaces	Virtual Desktop	`cmdb_ci_virtual_desktop`
Amazon Elastic Container Service (ECS)	AWS Cloud ECS Cluster	`cmdb_ci_cloud_ecs_cluster`
Amazon Elastic Kubernetes Service (EKS)	Kubernetes Cluster	`cmdb_ci_kubernetes_cluster`
Amazon Elastic File System (EFS)	File System	`cmdb_ci_file_service`

Updating the AWS Load Balancer resource details in the ServiceNow CMDB

AWS Load Balancer resources map to the ServiceNow table: Cloud Load Balancer (`cmdb_ci_cloud_load_balancer`).

The previous table in the Connector was Load Balancer Service (`cmdb_ci_lb_service`). This change aligns with ServiceNow's cloud resource best practices.

> **Note**
>
> The following transition steps are required only if you are upgrading from version 3 of the Connector to version 4.

Fix Scripts to address changes to ELB mappings in ServiceNow CMDB

If you are using AWS Config integration before version 4, the Connector includes two fix scripts that migrate existing Connector resources in the Load Balancer Service (`cmdb_ci_lb_service`) table to the Cloud Load Balancer (`cmdb_ci_cloud_load_balancer`) table.

Fix Script 1: AWS SMC - Migrate ELB data

This fix script migrates ELBv2 data from the legacy Load Balancer Service (`cmdb_ci_lb_service`) table with `discovery_source` *AWS Service Management Connector* to the new Cloud Load Balancer (`cmdb_ci_cloud_load_balancer`) table with all the relationships. (Legacy records remain undeleted for audit).

> **Note**
>
> The **AWS SMC - Migrate ELB data fix script** migrates all existing relationships of the ELBv2 resource in Load Balancer Service (`cmdb_ci_lb_service`), where the discovery source is *AWS Service Management Connector* to the newly created resource in the Cloud Load Balancer (`cmdb_ci_cloud_load_balancer`) table.

Fix Script 2: AWS SMC - Delete ELB legacy relationship (optional)

This fix script deletes the relationships where a child or parent is a resource in the original Load Balancer Service (`cmdb_ci_lb_service`) table, and the discovery source of the resource is *AWS Service Management Connector*.

> **Note**
>
> We recommend you execute **AWS SMC - Delete ELB legacy relationship fix script** after executing **AWS SMC - Migrate ELB data fix script**, and receiving approvals from your ServiceNow admin based on your organization's data retention policies.

To run a fix script in ServiceNow

1. Log in to your ServiceNow instance as an admin user (for example, System Administrator) in the fulfiller view (Standard user interface view).
2. In the filter navigator, enter **`System Definition`**.
3. Choose **Fix Scripts**.
4. To migrate resources to the new Cloud Load Balancer table, choose **AWS SMC - Migrate ELB data**.

   To delete relationships from the Load Balancer Service table, choose **AWS SMC - Delete ELB legacy relationship fix script**.
5. Open the fix script to execute.
6. Choose **Run Fix Script**.

AWS Security Hub in ServiceNow

AWS Security Hub enables users to view security Findings from AWS services such as Amazon Guard Duty and Amazon Inspector, as well as AWS Partner solutions.

If you use both AWS Security Hub and ServiceNow ITSM, the AWS Service Management Connector for ServiceNow allows you to create an automated, bidirectional integration between Security Hub and ServiceNow ITSM. This two-way integration synchronizes your Security Hub findings and ServiceNow tickets.

Specifically, as a ServiceNow administrator, you can use this integration to automatically create ServiceNow incident or problem tickets from AWS Security Hub findings. When you update those tickets in ServiceNow, the changes are automatically replicated back to the original Security Hub findings. For example, when you resolve the ticket in ServiceNow, the workflow status of the Security Hub finding also changes to `RESOLVED`. This action ensures that Security Hub always has up-to-date information about your security posture.

View the following video, *AWS Security Hub - Bidirectional integration with ServiceNow ITSM*, for an overview of the AWS Security Hub integration to the Connector for ServiceNow.

Share (https://www.youtube.com/embed/OYTi0sjEggEShare) AWS Security Hub - Bidirectional integration with ServiceNow ITSM

Configuring AWS Security Hub in ServiceNow

This section describes how to configure your AWS services in ServiceNow.

To configure AWS Security Hub integration features

1. Enable AWS Security Hub. For more information, see Setting up AWS Security Hub with the Console.
2. Set up an SQS queue to receive updated Findings. Name the queue, `AwsServiceManagementConnectorForSecurityHubQueue`, to align with the default name in the ServiceNow System Properties for the AWS Security Hub integration. For more information, see Getting started with Amazon SQS.
3. Set up an Amazon EventBridge rule to detect changes to Findings and push these to the queue. For more information, see Getting started with Amazon EventBridge.

   The rule should have this event pattern and point to the SQS queue created in Step 2.

   ```
   "EventPattern": {

          "source": [

           "aws.securityhub"

           ]
   }
   ```

4. You can also customize this CloudWatch Events rule to only pull in Security Hub findings that have specific finding types, severity labels, workflow statuses, or compliance statuses. For details about how to filter the event pattern, see Configuring an EventBridge rule for automatically sent findings in the *AWS Security Hub User Guide*.

> **Note**
>
> You can use the AWS CloudFormation templates for the Connector for ServiceNow to automate the AWS Config custom resource and AWS Security Hub integration features. For more information, see Baseline Permissions.

Synchronizing AWS Security Hub to the Connector in ServiceNow

This section shows you how to synchronize AWS Security Hub to the Connector in ServiceNow.

To configure AWS Security Hub synchronization behavior to the Connector in ServiceNow

1. In the ServiceNow filter navigator in the fulfiller (stand user interface) view, enter **`AWS Service Management Connector`**.
2. Choose **System Properties**, then **AWS Security Hub**.
3. Set these configuration items:

   - Choose the types of AWS Security Hub Findings to sync in ServiceNow: **CRITICAL**, **HIGH**, **MEDIUM**, **LOW**, and **INFORMATIONAL**.
   - Choose an action for a newly synced Finding to the Connector in ServiceNow:
     - **Do Nothing**. This action only imports Security Finding types for the scoped app. Users with scoped app permissions can view and choose to create an Incident or Problem. **Do Nothing** is the default value in the Connector.
     - **Create Incident**. This action automatically creates Incidents from Security Findings and syncs updates in ServiceNow to AWS Security Hub.
     - **Create Problem**. This action automatically creates Incidents from Security Findings and syncs updates in ServiceNow to AWS Security Hub.
     - **Create Incident and Problem**. This action automatically creates Incidents and Problems from Security Findings and syncs updates in ServiceNow to AWS Security Hub.
   - Adjust the maximum number of messages to fetch from the SQS queue per sync, account, or Region (default 50). By default, the sync process runs every five minutes.
   - Change the SQS Queue name if you're not using the default that the Connector created. The CloudFormation template supplies the Connector.

> **Note**
>
> We recommend you not change the SQS name in the ServiceNow scoped app (`AwsServiceManagementConnectorForSecurityHubQueue`) unless you change the SQS name in the AWS account.

4. Choose **Save** after any changes.

Fields synchronized from AWS Security Hub Findings to the ServiceNow scoped app AWS Security Hub Findings module in ServiceNow

Region	The Region that generated the Finding.
Account Id	The account that generated the Finding.
Company Name	The company that generated the Finding (e.g. AWS).
Compliance	Whether a resource passes the configured compliance criteria. Contains status (PASSED, WARNING, FAILED, NOT_AVAILABLE). If the resource does not pass, it will contain information about the reason.
Created At	The creation time of the Finding.
Description	A description of the Finding.
Criticality	The level of importance for the resource associated with the Finding.
First Observed At	First observation of when Findings captured any potential security issues.
Last Observed at	The most recent time Findings captured any potential security issues.
Product Name	The name of the product that generates the Finding (such as Security Hub).
Product Arn	The ARN of the product that generates the Finding.

Record State	Either ACTIVE or ARCHIVED.
Severity (normalized)	A value from 0 to 100 that indicates the severity of the problem associated with the Finding.
Status	PASSED, WARNING, FAILED, or NOT AVAILABLE.
Title	The title of the Finding.
Updated At	When the Finding provider last updated the record.
Workflow Status	The workflow status can be: NEW, ASSIGNED, IN PROGRESS, RESOLVED, DEFERRED, or DUPLICATE.
Remediation Text	A description of suggested action to resolve the discovered issue.
Remediation Url	A link to a resource that can resolve the discovered issue.

Note

ServiceNow does not duplicate findings. If a Security Hub finding is sent to ServiceNow with the same finding ID as one previously sent to ServiceNow, we update the ticket with the most recent information in the finding.

Validating AWS Security Hub integration in ServiceNow

This section describes how to validate AWS Security Hub integration in ServiceNow.

To view Findings from AWS Security Hub

To view AWS Security Hub Findings, you must have the role, **x_126749_aws_sc.finding_manager**, from the Connector scope app.

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter **`AWS Service Management`**.

3. Choose **AWS Security Hub**.
4. Choose **Findings** to show a list of all synced Findings.
5. Choose a Finding to open the record.
6. The **Incident and Problem** fields show the Incident and Problem related to the Finding if these exist.
7. Choose the ⓘ symbol to the right of the field to preview the Incident or Problem.
8. Choose **Open Record** on the preview form to open the Incident or Problem.
9. If the Connector does not automatically create a ServiceNow Incident or Problem when a new Finding syncs, choose the link at the bottom of the form to create one manually.

This table shows how fields map from ServiceNow Findings records to ServiceNow as Incident or Problem records.

Finding	Incident	Problem
Created at	Opened at	Opened at
Company Name	Company	Company
Description	Description	Description
Criticality	Impact	Impact
Severity	Urgency	Urgency
Hardcoded to *software*	Category	Category
Id of record in *cmdb_ci_service* with name **AWS Security Hub**	Business service	Business service
Description	Short description	Short description
Reference to related Problem if it exists	problem_id	n/a

This table shows how fields synchronize between AWS Security Findings and ServiceNow Incidents or Problems.

AWS Security Hub value	ServiceNow Incident	ServiceNow Problem
Severity Label	Urgency	Urgency
Criticality	Impact	Impact

Fields synchronized between AWS Security Findings, Incidents, and Problems in ServiceNow

- Finding severity label → Problem/Incident urgency
  - INFORMATIONAL or LOW → LOW
  - MEDIUM → MEDIUM
  - HIGH or CRITICAL → HIGH
- Finding criticality → Problem/Incident impact
  - 0 - 29 → LOW
  - 30 - 69 → MEDIUM
  - 70 - 100 → HIGH

Fields synchronized from Findings to AWS Security Hub

- Severity (Label and Normalized)
- WorkflowStatus

AWS Systems Manager OpsCenter in ServiceNow

To allow the Connector to synchronize AWS Systems Manager OpsCenter data for a specific Region, you must enable OpsCenter in that account and Region.

For more information, see AWS Systems Manager OpsCenter.

Topics

Configuring ServiceNow for AWS Systems Manager OpsCenter

This section shows you how to integrate AWS Systems Manager OpsCenter in ServiceNow.

To configure the AWS Systems Manager OpsCenter integration system properties

1. In the navigator, enter **`AWS Service Management`**.
2. Choose **System Properties**, then **AWS Systems Manager - OpsCenter**.
3. Review the available settings and recommendations in the table below.

Available settings	Description
Synchronizing a new OpsItem with a severity 1	**Do Nothing**. This action only imports selected OpsItems for the scoped app. Users with scoped app permissions can view and choose to create an Incident or Problem. **Create Incident**. This action automatically creates Incidents from OpsItems and syncs updates in ServiceNow to AWS Systems Manager - OpsCenter. **Default value**: Create Incident
Synchronizing a new OpsItem with a severity 2	**Do Nothing**. This action only imports selected OpsItems for the scoped app. Users with scoped app permissions can view and choose to create Incident or Problem. **Create Incident**. This action automatically creates Incidents from OpsItems and syncs updates in ServiceNow to AWS Systems Manager - OpsCenter. **Default value**: Create Incident
Synchronizing a new OpsItem with a severity 3	**Do Nothing**. This action only imports selected OpsItems for the scoped app. Users with scoped app permissions can view and choose to create Incident or Problem.

Available settings	Description
	Create Incident. This action automatically creates Incidents from OpsItems and syncs updates in ServiceNow to AWS Systems Manager - OpsCenter. **Default value**: Do Nothing
Synchronizing a new OpsItem with a severity 4	**Do Nothing**. This action only imports selected OpsItems for the scoped app. Users with scoped app permissions can view and choose to create Incident or Problem. **Create Incident**. This action automatically creates Incidents based on OpsItems and syncs updates in ServiceNow to AWS Systems Manager - OpsCenter. **Default value**: Do Nothing

Available settings	Description
Assignment Group (SYS_ID) for created Incidents	ServiceNow Incidents from AWS OpsItems need assignment group. **To associate the assignment group for ServiceNow Incidents from AWS OpsItems** 1. Choose the section **Set the assignment group sys_id** or name that the Connector uses when creating Incidents. 2. Enter the Assignment group `sys_id`. If you need to find the group `sys_id`, enter **`System Security`** in the left navigator. 3. Choose the **Groups** module and search for the Group name. 5. Choose the group to associate to ServiceNow Incidents generated from AWS OpsItems and choose **Copy sys_id**. You can now paste the copied `sys_id` into AWS Systems Manager – OpsCenter System Properties.

Validating AWS Systems Manager OpsCenter integration in ServiceNow

This section describes how to validate AWS Systems Manager OpsCenter integration in ServiceNow.

To view OpsItems from AWS Systems Manager - OpsCenter

To view AWS OpsItem, you must have the role, `x_126749_aws_sc.opscenter_manager`, with the Connector scope app.

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (Standard user interface view).
2. In the navigator, enter **`AWS Service Management`**.
3. Choose **AWS Systems Manager - OpsCenter**.

4. Choose **OpsItems** to show a list of all synced Findings.
5. Choose an OpsItems to open the record.

   The **Incident** and **Problem** fields show the Incident for the OpsItems, if these exist.
6. Choose the ⓘ icon to the right of the field to preview the Incident.
7. Choose **Open Record** on the preview form to open the Incident.

   If the Connector configuration does not to automatically create a ServiceNow Incident when a new Finding syncs, you can create one manually. To do so, choose the link at the bottom of the form.

To execute an AWS Systems Manager – Automation Document from an AWS OpsItems associated to a ServiceNow Incident

One of the following conditions must be true to view or execute automation documents (runbooks):

- The user has the role Account Manager or Automation Manager.
- The user has a linked Incident.
- The system parameter **Assignment Group (SYS_ID) for created incidents** is set to a valid group and a linked Incident whose Assignment group is set to that group, and the user is a member of that group.

> **Note**
>
> To enable this feature, you must activate AWS Systems Manager Automation in the AWS Account and opt in to the Connector.

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter **`AWS Service Management`**. Then choose **AWS Systems Manager - OpsCenter**.
3. Choose OpsItems to show a list of all synced Findings. Then choose **Execute Automation Document**.
4. Choose your Automation Document.

> **Note**
>
> You can configure an OpsItem with Automation Documents and mark it as *Associated*.

5. Choose **Order Execution** next to the Automation Document you want to execute. You'll see the ServiceNow catalog item associated with the Automation Document.
6. Enter the necessary AWS parameters and choose **Order Now**.
7. In OpsItems in the scoped app, choose the OpsItem in the Automation Document where you executed it.
8. In **OpsItem Automation Executions**, review the success or failure status.
9. Follow your organization's Incident management procedures to determine related Incident resolution actions.

Fields mapped from OpsCenter OpsItem records to ServiceNow Incident records

This table shows how AWS OpsItems map to ServiceNow Incidents.

AWS Ops Center	ServiceNow Incident
Title	short_description
Description	description
CreatedTime	opened_at
Status	incident_state
Severity	impact/urgency
Priority	priority
CreatedBy	Not synced
LastModifiedTime	Not synced
LastModifiedBy	Not synced

AWS Ops Center	ServiceNow Incident
Source	Not synced
OpsItemId	Not synced
OperationalData	Not synced
Category	Software

Incident Status is an integer in ServiceNow. We map OpsItem status values to values.

ServiceNow Incident Status	OpsCenter Status
New (primary)	Open
On Hold	Open
In Progress	In Progress
Resolved (primary)	Resolved
Closed	Resolved
Cancelled	Resolved

In this type of subjective mapping, we only change the target value if it is incompatible. An example of subjective mapping would be if *New* and *On Hold* in ServiceNow both map to *Open* in AWS. An example of an incompatible target would be if the Incident is *On Hold*, while we're synchronizing from AWS an OpsItem that is *Open*, and we don't change *On Hold*.

Priority - In Incident, you can’t set the Priority field directly. The values of the **Impact** and **Urgency** fields calculate the **Priority** field. When synchronizing from AWS, we set by default the fields shown in the table below:

OpsItem Priority	ServiceNow Incident		
	Impact	**Urgency**	**Priority (Calculated)**

OpsItem Priority	ServiceNow Incident		
1	High	High	Critical (1)
2	Medium	High	High (2)
3	Medium	Medium	Moderate (3)
4	Low	Medium	Low (4)
5	Low	Low	Planning (5)

You can find these mappings in a ServiceNow table *Priority Data Lookup*. While we can use this table to find the required values of **Impact** and **Urgency**, note that you can customize the mappings and also define new priority values. Additionally, you might want a specific priority in AWS to map to an entirely different priority in an Incident or Problem.

Integrating AWS Systems Manager Automation in ServiceNow

To allow the Connector to execute Automation Documents, you must ensure that the Connector Sync and End user has the permissions required to sync and execute Automation Documents.

For more information, see Setting up Automation.

This table describes the available settings to configure Support integration system properties.

Available settings	Description
Name of the Systems Manager category to assign to Automation Documents from AWS Systems Manager	The setting allows the Automation Documents to be created under the specified category. By default, the category sets to AWS Systems Manager Automation.
Name of a workflow that starts the execution of an Automation Document from AWS Systems Manager	The setting allows you to use custom workflow with the AWS Systems Manager Automation integration.

Validating AWS Systems Manager Automation integration in ServiceNow

This section describes how to validate AWS Systems Manager Automation integration in ServiceNow.

To request an AWS Systems Manager Automation document (runbook) execution

1. Log in to your ServiceNow instance as the end user (for this example, Abel Tuter).
2. In the navigation filter, enter **`AWS Systems Manager`**, then choose **Systems Manager**.
3. Choose an AWS Systems Manager document to execute.
4. Enter the request details, including parameters and tags.
5. Choose **Order Now** to submit the ServiceNow request and execute the AWS Systems Manager document.

   You receive an order status acknowledging your request submission.

To view AWS Systems Manager document executions

1. Log in to your ServiceNow instance as the end user (for example, Abel Tuter).
2. In the navigation filter, enter **`AWS Systems Manager`**, then choose **Automation Executions**.

   The user interface view displays the latest executions and provides the status.

Support in ServiceNow

To allow the Connector to synchronize Support tickets, the account should have a Business or Enterprise Support plan. For more information, see Getting started with Support.

> **Note**
>
> AWS Service Management Connector allows AWS Managed Services (AMS) Accelerate users to create Incidents and Service Requests through ServiceNow. To ensure that your account has the required permissions to create AMS Accelerate support cases, make sure you onboard your account to Accelerate. For more information, see Getting Started with AWS Managed Services.

Topics

Configuring Support integration in ServiceNow

This section describes how to configure Support integration in ServiceNow.

To configure AWS Support integration features

1. Set up an SQS queue (in N.Virginia (us-east-1) for Commercial regions and US West (us-gov-west-1) for GovCloud regions) to sync AWS Support cases. Name the queue, **AwsServiceManagementConnectorForSupportQueue**, to align with the default name in the ServiceNow System Properties for the AWS Support integration. For more information, see Getting started with Amazon SQS.
2. Set up an Amazon EventBridge rule to detect changes to AWS Support Cases and push these to the queue. For more information, see Getting started with Amazon EventBridge.

The rule should have this event pattern and point to the SQS queue created in Step 1.

```
            "EventPattern": {
{
    "detail-type": ["Support Case Update"],
    "source": ["aws.support"]
}
}
```

> **Note**
>
> You can use baseline AWS CloudFormation tempates for the Connector for ServiceNow to automate the Support integration features. For more information, see Baseline Permissions.

To create the required SQS queue and EventBridge rule, use Connector for ServiceNow - AWS Support Commercial Regions, and Connector for Service Management - AWS Support GovCloud West Region.

Configuring ServiceNow for integration with Support

This section shows you how to integrate Support in ServiceNow.

To configure the Support integration System Properties

1. In the navigator, enter **`AWS Service Management`**.
2. Choose **System Properties**, then **Support**.
3. Set the system property, as required.

Available settings	Description
Interval, in minutes, between the execution of full synchronization	Default: **1440 min**
SQS Name created by the AWS CloudFormation stack. The same name must be used for all accounts	Default: **AwsServiceManagementConnectorForSupportQueue**
(Advanced mode) Enable an *intermediate table* (SMC Support Case table) to synchronize data to and from Support. Use caution; enabling an intermediate table replaces the default Incident table.	Default: **False**

Advanced Mode for Support integration (optional)

AWS Service Management Connector allows you to enable an intermediate table for the creation of Support Cases. This allows you to add custom logic using ServiceNow business rules and workflows to align with your internal Incident or Case Management process.

For more information about enabling advanced mode, refer to the *Advanced mode* row in the above table.

After you create an Support Case, the API only allows specific changes by an end user. The allowable changes for design considerations while using Support integration are:

- Adding a correspondence to the case
- Resolving the case
- Reopening a case, which occurs if you add correspondence to a previously resolved support case

Validating Support in ServiceNow

This section describes how to create, view, and manage integration features for Support in order to validate integration.

To view Cases from Support

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter **`AWS Service Management`**.
3. To show a list of all synched Support Cases, choose **Incidents** under **Support**.

To manually sync a Support Case

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter **`AWS Service Management`**.
3. To show a list of all synched Support Cases, choose **Incidents** under **Support**.
4. Choose an Incident to open the record.
5. Choose **Sync From AWS**.

To create a general Support Case

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter **`AWS Service Management`**.

3. To show a list of all synched Support Cases, choose **Incidents** under **Support**.
4. Choose **New** from list header.
5. Complete the mandatory fields on the form.

   - **Subject**- Brief summary of the question or issue
   - **Description** – Detailed account of the question or issue
   - **AWS Account** – AWS account against which the support case is initiated
   - **AWS Service** – AWS Service related to the support case
   - **AWS Category** – Category of the case under the related service
   - **Caller** – ServiceNow field that indicates who created the support ticket
6. Choose **Submit**.
7. Choose the Incident you created from the list.

   The **AWS Case Id** and **AWS Case Status** displays.

For AWS Managed Services Accelerate customer to create AMS Accelerate Service Request

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter **`AWS Service Management`**.
3. To show a list of all synched Support Cases, choose **Incidents** under **Support**.
4. Choose **New** from list header.
5. Complete the mandatory fields on the form.

   - **Subject**- Brief summary of the question or issue
   - **Description** – Detailed account of the question or issue
   - **AWS Account** – AWS account against which the support case is initiated
   - **AWS Service** – AWS Service related to the support case (Select **AMS Operations – Service Request**)
   - **AWS Category** – Category of the case under the related service
   - **Caller** – ServiceNow field that indicates who created the support ticket
6. Choose **Submit**.
7. Choose the Incident you created from the list.

The **AWS Case Id** and **AWS Case Status** displays.

For AWS Managed Services Accelerate customer to create AMS Accelerate Report Incident

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter **`AWS Service Management`**.
3. To show a list of all synched Support Cases, choose **Incidents** under **Support**.
4. Choose **New** from list header.
5. Complete the mandatory fields on the form.
   - **Subject**- Brief summary of the question or issue
   - **Description** – Detailed account of the question or issue
   - **AWS Account** – AWS account against which the support case is initiated
   - **AWS Service** – AWS Service related to the support case (Select **AMS Operations – Report Incident**)
   - **AWS Category** – Category of the case under the related service
   - **Caller** – ServiceNow field that indicates who created the support ticket
6. Choose **Submit**.
7. Choose the Incident you created from the list.

   The **AWS Case Id** and **AWS Case Status** displays.

To add a correspondence to an existing Support Case

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter **`AWS Service Management`**.
3. To show a list of all synched Support Cases, choose **Incidents** under **Support**.
4. Choose an Incident to open the record.
5. In the Incident form, scroll to the middle of the page to view and open the **Notes** tab.
6. Add correspondence on the **Additional Comments** (Customer visible) field.
7. Choose **Post**.

To add an attachment to an existing Support Case

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter **`AWS Service Management`**.
3. To show a list of all synched Support Cases, choose **Incidents** under **Support**.
4. Choose an Incident to open the record.
5. On the Incident form header, choose paper clip icon to add attachment.
6. Choose the file from your disk to add as an attachment.

To resolve a Support Case

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter **`AWS Service Management`**.
3. To show a list of all synched Support Cases, choose **Incidents** under **Support**.
4. Choose an Incident to open the record.
5. In the Incident form, scroll to the middle of the page to view and open the Resolution Information tab.
6. Complete the **Resolution Code** and **Resolution Notes** fields.
7. On the Incident form header, choose **Resolve**.

Fields mapped from Support Case records to ServiceNow Incident records

This table shows how Support Case map to ServiceNow Incidents.

Support case	ServiceNow incident
Subject	short_description
First correspondence	description
Case ID	x_126749_aws_sc_awssupportcaseid
Status	x_126749_aws_sc_awscasestatus

Support case	ServiceNow incident
Service	x_126749_aws_sc_awsservice
Category	x_126749_aws_sc_awscategory
Additional contacts	x_126749_aws_sc_awscasecommunication emails
AWS account	x_126749_aws_sc_awsaccount

Incident State is an integer in ServiceNow. We map Support case status values to ServiceNow state.

ServiceNow incident Status	Support case status
New	Unassigned
New	Open
In Progress	Work in progress
In Progress	Reopened
On Hold	Pending customer action
Resolved	Resolved
Resolved	Closed
Resolved	Closed

Priority: In Incident, you can't set the Priority field directly.

The values of the **Impact** and **Urgency** fields calculate the **Priority** field. When synchronizing from AWS, we set by default the fields shown in the table below.

Support Case Severity label	Support Case Severity value	ServiceNow Incident priority label	ServiceNow Incident priority value
Business Critical System Down (Enterprise support plan only)	critical	1 – Critical	1
Production System Down	urgent	2 – High	2
Production System Impaired	high	3 – Moderate	3
System Impaired	normal	4 – Low	4
General Guidance	low	5 – Planning	5

Support integration also enables you to customize the priority values, and maps Support Case Severity to ServiceNow Incident Priority.

To create custom priority mappings

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter **`AWS Service Management`**.
3. Under **Setup**, choose **Priority Mappings**. Then choose **New**.
4. Choose **AWS Record Type** as **Support Case**.
5. For mapping, choose **Support Case Severity** and **ServiceNow Incident Priority**.
6. Choose **Submit**.

AWS Systems Manager Change Manager in ServiceNow

AWS Service Management Connector includes a curated version of the Change Manager integration. To allow the Connector to synchronize change templates, the change templates should be:

- An Approved status in AWS
- At least one Automation Runbook associated with it
- Enabled as auto-approval

For more information, see AWS Systems Manager Change Manager.

You can also view resources affected by the changes that were executed on their AWS accounts from the AWS CloudTrail events available on the AWS change request.

> **Note**
>
> Currently, only the first level events that occurred in the execution of an automation document will be tracked and synched. Steps which have nested automations will not have the events synced. This can however be traced separately in the AWS CloudTrail console using Lake feature by their unique automation execution ID.

Configuring AWS for AWS Systems Manager Change Manager in ServiceNow

AWS Systems Manager uses the service-linked role named `AWSServiceRoleForAmazonSSM`. AWS Systems Manager uses this IAM service role to manage AWS resources on your behalf. For more information, see Using service-linked roles for AWS Systems Manager.

To create a service-linked role for AWS Systems Manager

1. Follow the instructions in Creating a service-linked role (console) to create the role.
2. Choose **AWS Service as Systems Manager** and the use case as **Systems Manager – Inventory and Maintenance Window**.
3. Review the details and be sure to attach `AmazonSSMServiceRolePolicy`. Then choose **Create Role**.

To create AutomationAssumeRole

1. Follow the instructions in Creating an IAM role in your AWS account to create a role, `ServiceNowChangeManagerRole`.

2. Add permissions for `ServiceNowChangeManagerRole.` Choose the use case as Systems Manager and choose `AmazonSSMAutomationRole` (AWS managed policy).

> **Note**
>
> You can use baseline AWS CloudFormation tempates to create the `ServiceNowChangeManagerRole` role. For more information, see Setting baseline permissions for AWS Service Management Connector for ServiceNow.

> **Note**
>
> `ServiceNowChangeManagerRole` contains the minimum baseline permissions to execute change templates that contain automation runbooks on EC2 instances. To invoke automation runbooks on other services, you need to attach additional policies. For more information, see Create a service role for Automation.

To create an event data store (optional)

To create AWS CloudTrail Lake, follow the instructions outlined in Create an event data store in your AWS account to create the event data store.

Configuring Support integration system properties with ServiceNow

The AWS Systems Manager Change Manager integration for AWS Service Management Connector aligns with the Change Management process in ServiceNow. It enables you to align the internal Change Management process for executing pre-approved change templates directly from a ServiceNow instance.

To configure the AWS Support integration system properties

1. In the navigator, enter **`AWS Service Management`**.
2. Choose **System Properties**, then **AWS Systems Manager Change Manager**.
3. Review the available settings and recommendations in the table below.

Available settings	Description
Name of the Change Manager category to assign to AWS Change Template from AWS Systems Manager Change Manager	The setting correlates to the Catalog item category in ServiceNow to which the synchroni zed AWS Change templates are associated.
Assignment Group (SYS_ID) to use when creating Change Requests from Change Template	The setting automatically assigns the change requests created from the change templates to the Assignment Group that relates to the `sys_id`.
Default role name that allows the Automation to perform the actions on your behalf	The setting contains the default role to create change requests from AWS change templates. The setting is available if the user does not fill in the `AutomationAssumeRole` field when requesting a change from AWS Systems Manager Change Manager. The value is case-sensitive and must exist in every account using the AWS Systems Manager Change Manager.
AWS CloudTrail Lake: Event Data Store Name	Defines the Name of the AWS CloudTrail Lake: Event Data Store Name to target. Note that to use AWS Systems Manager Change Manager's CloudTrail Lake Event integration an Event Data Store with this Name MUST exist in all regions defined in AWS Accounts with AWS Systems Manager Change Manager enabled.
AWS CloudTrail Lake: Maximum number of events to retrieve per synchronization	Default : 1000

Validating AWS Systems Manager Change Manager integration in ServiceNow

This section describes how to validate AWS Systems Manager Change Manager integration in ServiceNow.

To view AWS Systems Manager Change templates

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter **`AWS Service Management Connector`**.
3. To show a list of all synched Change templates, choose **Change Templates** under **Systems Manager**.

To view Systems Manager Change Request

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter **`AWS Service Management Connector`**.
3. To show a list of all synched Change Requests created from ServiceNow, choose **Change Requests** under **Systems Manager**.
4. Choose a Change Request to open the record.

To view AWS Systems Manager Change Request Ops Items

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter **`AWS Service Management Connector`**.
3. To show a list of all synched Change Requests created from ServiceNow, choose **Change Request Ops Items** under **Systems Manager**.
4. Choose an Ops Item to open the record.

To create AWS Systems Manager Change Manager change

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter **Change**. Then choose **Create New** to view the various Change options.
3. Choose **Create AWS Systems Manager Change Manager Change: Make changes to AWS resources using Change Manager Templates**.
4. Choose the runbook you want to execute and complete all the required fields.
5. Choose **Submit** to create a ServiceNow Change Request.
6. Choose **Request Approval** to send approval requests to members of the Assignment group.

   After change approval, it moves to a *Scheduled state*.
7. Choose **Implement**.
8. Scroll to the bottom and view Change Tasks under related lists to view the Change task associated with Automation Execution.

   After the Change Execution is complete, the change moves to a *Closed state*.

To view AWS CloudTrail events for the Change execution

This procedure requires you to create and configure AWS CloudTrail Lake on AWS and configure the Lake name on the AWS Systems Manager Change Manager system properties in ServiceNow

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter**`AWS Service Management Connector`**.
3. To show a list of all synched Change Requests created from ServiceNow, choose **Change Requests** under **AWS Systems Manager**.
4. Choose a Change Request to open the record.
5. Use UI Action, **Sync CloudTrail Events**, to start the synchronization of events.
6. Choose the same Change Request to reopen the record.
7. Scroll to the bottom of the Change Request form and use **CloudTrail Events** related list to review the events of the Change execution.

Fields mapped from AWS Change Request Ops Item records to ServiceNow Change Request records

This table shows how AWS Change Request Ops items map to ServiceNow Change Request.

AWS Change Request Ops Item	ServiceNow Change Request
AWS Account	x_126749_aws_sc_awsaccount
AWS Request ID	x_126749_aws_sc_awsrequestid
AWS Region	x_126749_aws_sc_awsregion
AWS Status	x_126749_aws_sc_awsstatus

AWS Systems Manager Incident Manager in ServiceNow

To allow the Connector to synchronize Incidents from AWS Systems Manager Incident Manager for a specific Region, you must enable Incident Manager in that account and Region.

For more information, see What is AWS Systems Manager Incident Manager.

Configuring ServiceNow for integration with AWS Systems Manager Incident Manager

This section shows you how to integrate AWS Systems Manager Incident Manager in ServiceNow.

To configure the AWS Systems Manager Incident Manager integration system properties

1. In the navigator, enter **`AWS Service Management Connector`**.
2. Choose **System Properties**, then **AWS Systems Manager Incident Manager.**
3. Review the available settings and recommendations in the table below.

Available settings	Description
Assignment Group value (SYS_ID) to use when creating ServiceNow Incidents from AWS Systems Manager Incident Manager synchronization	sys_id of the assignment group that the Connector uses when synching Incidents from AWS Systems Manager Incident Manager Default value: <empty>
Synchronization of the resolved status	Bidirectional. Sync Resolve status of the incident from AWS to ServiceNow and ServiceNow to AWS Unidirectional: AWS to ServiceNow. Sync Resolve status of the incident only from AWS to ServiceNow Unidirectional: ServiceNow to AWS. Sync Resolve status of the incident only from ServiceNow to AWS None. Resolve status are not synched Default value: Bidirectional

Validating AWS Systems Manager Incident Manager integration

This section describes how to validate AWS Systems Manager Incident Manager integration in ServiceNow.

To view Incident Manager incidents

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (Standard user interface view).
2. In the navigator, enter **`AWS Service Management`**.
3. Choose **AWS Systems Manager Incident Manager**.
4. Choose **Incidents** to show a list of all synced Incidents.

To view Incident Manager incident details

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (Standard user interface view).
2. In the navigator, enter **`AWS Service Management`** .
3. Choose **AWS Systems Manager Incident Manager**.
4. Choose **Incidents** to show a list of all synced Incidents.
5. To open the record, choose the **Number** field of an Incident.
6. Open the AWS Systems Manager Incident Manager tab to display details of the IM Incident.
7. To open the Incident Manager incident on the AWS Incident Management console, choose the AWS Incident URL.

To resolve an Incident Manager incident

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (Standard user interface view).
2. In the navigator, enter **`AWS Service Management`**.
3. Choose **AWS Systems Manager Incident Manager**.
4. Choose **Incidents** to show a list of all synced Incidents.
5. To open the record, choose the **Number** field of an Incident
6. In the **Resolution Information** tab, complete **Resolution Code and Resolution Notes**.
7. Choose **Resolve**.

To view AWS Systems Manager Incident Manager Ops Items

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (Standard user interface view).
2. In the navigator, enter **`AWS Service Management`**.
3. Choose **AWS Systems Manager Incident Manager**.
4. Choose **Incidents** to show a list of all synced Incidents.
5. To open the record, choose the **Number** field of an Incident.
6. Scroll to the bottom of the Incident form and use the AWS OpsItems related list to see associated OpsItems.

Fields mapped from Incident Manager incident to ServiceNow Incident records

This table shows how AWS Incident Manager Incidents map to ServiceNow Incidents.

AWS Incident Manager incident	ServiceNow Incident
Title	short_description
Summary	description
Incident ARN	x_126749_aws_sc_awsincidentarn
AWS Account	x_126749_aws_sc_awsaccount
AWS Region	x_126749_aws_sc_awsregion
Status	x_126749_aws_sc_awsstatus
Start time	x_126749_aws_sc_awscreationtime
Resolved time	x_126749_aws_sc_awsresolvetime
Updated time	x_126749_aws_sc_awslastupdatedtime
Incident Sync time	x_126749_aws_sc_awslastsynctime
AWS incident URL	x_126749_aws_sc_awsincidenturl
Impact	impact

Incident Status is an integer in ServiceNow. We map Incident Manager incident status values to ServiceNow status values.

Incident Manager Incident Status	ServiceNow Incident Status
Open	New
Resolved	Resolved

Incident Manager Incident Status	ServiceNow Incident Status
Resolved	Cancelled

Priority - In ServiceNow Incident, you can't set the Priority field directly. The values of the **Impact** and **Urgency** fields calculate the **Priority** field. When synchronizing from AWS, we set the default priorities as below:

Incident Manager Incident	ServiceNow Incident		
	Impact	**Urgency**	**Priority (Calculated)**
Critical	High	High	Critical (1)
High	High	High	Critical (1)
Medium	Medium	High	High (2)
Low	Low	High	Moderate (3)
No Impact	Low	High	Moderate (3)

AWS Health in ServiceNow

AWS Health integration includes a dashboard that provides ongoing visibility into your resource performance and the availability of your AWS services and accounts. This enables deeper visibility into resource issues, upcoming changes, and important notifications.

To allow the Connector to synchronize AWS Health events and resource information, the account should have a Business or Enterprise Support plan. For more information, refer to What is AWS Health?

Topics

Configuring AWS

This section describes how to configure AWS Health integration in ServiceNow.

Configure AWS for health-integration features

1. Set up an Amazon SQS queue to sync AWS Health events. Name the queue, **AwsServiceManagementConnectorForHealthDashboardQueue**, to align with the default name in the ServiceNow System Properties for the AWS Health integration. For more information, refer to Getting started with Amazon SQS.
2. Set up an Amazon EventBridge rule to detect **Health Event** changes and push them to the queue. For more information, refer to Getting started with Amazon EventBridge. The rule should have the following event pattern and point to the Amazon SQS queue from step 1:

```
"EventPattern":
{
    "source":
    [
        "aws.health"
    ]
}
```

> **Note**
>
> The SQS queue synchronizes every five minutes. To change this threshold, navigate to **Scheduled Jobs**, and modify the **Repeat Interval** value of the **Synchronize AWS Health** job.

> **Note**
>
> You can use baseline AWS CloudFormation tempates to automate AWS Health integration features. For more information, refer to Setting baseline permissions for AWS Service Management Connector for ServiceNow.

Synchronizing AWS Health events with ServiceNow

This section shows you how to synchronize AWS Health events with ServiceNow.

1. In the ServiceNow filter navigator in the fulfiller (stand user interface) view, enter **`AWS Service Management Connector`**.
2. Choose **System Properties** and then **AWS Health**.

   Configure the SQS name created by the CloudFormation stack. Note that a queue with this name must exist in all Regions defined in any AWS accounts with the AWS Health integration enabled. The default value is **AwsServiceManagementConnectorForHealthDashboardQueue**.

   > **Note**
   >
   > Unless you change the SQS name in the AWS account, don't change the Amazon SQS name in the ServiceNow scoped app (`AwsServiceManagementConnectorForHealthDashboardQueue`).

3. Review and modify the following settings as needed:

ServiceNow settings

Setting	Description	Default value	
SQS queue name	Name of the queue to fetch messages from. Only change this setting if you change the CloudFormation template that creates the queue.	`AwsServiceManagementConnectorForHealthDashboardQueue`	
Enable auto-creation for `issue` and `investigation`	Automatically creates a ServiceNow incident for new health events for `issue` and `investigation` types. If this setting is disabled, users can manually create incidents through the health dashboard.	*none*	

Setting	Description	Default value	
Enable auto-creation for `accountNotification`	Automatically creates a ServiceNow change request for new health events of type `accountNotification` . If this setting is disabled, users can manually create change requests through the health dashboard.	*none*	
Enable auto-creation for `scheduledChange`	Automatically creates a ServiceNow change request for new health events of type `scheduledChange` . If this setting is disabled, users can manually create change requests through the health dashboard.	*none*	
Assignment group	System ID of the default assignment group, which is the ServiceNow group that automatically assigns incidents and change requests. If this field is blank, no default group is assigned.	*none*	

Note

The types of change requests are `Standard`, `Normal`, and `Emergency`, but custom types are also available. The default type is `Standard`.

Validating AWS Health integration

View AWS Health dashboard

> **Note**
>
> To view the the AWS Health dashboard, you must use the role **x_126749_aws_sc.health_dashboard_viewer**.

1. Log in to your ServiceNow instance in the fulfiller (standard) view.
2. In the search box, enter **`AWS Service Management Connector`**.
3. Choose **AWS Health** and then **Dashboards**.
4. At the top-right, select your account from the **Select an AWS account** dropdown list. The following four tabs are available:
   - **Open and recent issues** (opens by default) displays health events that were updated within the past seven days. Choose an event to display its details and a list of affected resources.
   - **Scheduled changes** displays future health events with start times after the current date and time.
   - **Other notifications** displays health events that were updated within the past seven days.
   - **Event log** displays all health events for the selected AWS account.

View AWS Health incidents

1. Log in to your ServiceNow instance in the fulfiller (standard) view.
2. In the navigator, enter **`AWS Service Management Connector`**.
3. Under **AWS Health**, choose **AWS Health Incidents**.

View AWS Health change requests

1. Log in to your ServiceNow instance in the fulfiller (standard) view.
2. In the navigator, enter **`AWS Service Management Connector`**.
3. Under **AWS Health**, choose **AWS Health Requests**.

Manually create an AWS Health incident

1. Log in to your ServiceNow instance in the fulfiller (standard) view.
2. In the navigator, enter **`AWS Service Management Connector`**.
3. Choose **AWS Health** and then **Dashboards**.
4. Choose an event that doesn't already have an incident linked to it.
5. Choose **Create a New Incident**. You are redirected to the new-incident form, which has prefilled data fields for the selected health event.

Manually create an AWS Health change

1. Log in to your ServiceNow instance in the fulfiller (standard) view.
2. In the navigator, enter **`AWS Service Management Connector`**.
3. Choose **AWS Health** and then **Dashboards**.
4. Choose an event that doesn't already have a change linked to it.
5. Choose **Create a New Change**. You are redirected to the new-incident form, which has prefilled data fields for the selected health event.

Validate the automatic creation of AWS Health incidents and changes

1. Log in to your ServiceNow instance in the fulfiller (standard) view.
2. In the navigator, enter **`AWS Service Management Connector`**.
3. Navigate to **AWS Health** system properties, and enable automatic creation for health event types.
4. Generate new health events, and then sync AWS Health.

AWS Service Management Connector for ServiceNow Pricing

The AWS Service Management Connector for ServiceNow is a conventional ServiceNow scoped application developed and released through a ServiceNow Update Set. This application is available for no-cost download and use in your ServiceNow instance. You may still incur costs related to the use of AWS services integrated with the connector, and any licensing for Information Technology Service Management (ITSM) tools.

The certified version of the AWS Service Management Connector is available for no-cost install from the ServiceNow store.

AWS Service Management Connector (SMC) for ServiceNow uses security approved public APIs of the AWS service for all supported integrations. See the product pages of the AWS service to view pricing details. Contact the account manager or AWS Sales representatives for more information.

AWS Service	Pricing details
AWS Service Catalog	https://aws.amazon.com/servicecatalog/pricing/
AWS Config	https://aws.amazon.com/config/pricing
AWS Systems Manager	https://aws.amazon.com/systems-manager/pricing
AWS Security Hub	https://aws.amazon.com/security-hub/pricing/?nc=sn&loc=3
AWS Health and AWS Support	https://aws.amazon.com/premiumsupport/pricing/

AWS Service Management Connector is a ServiceNow scoped application certified and released through the ServiceNow store. SMC includes custom tables as part of the connector for the various integrations. For more information on your custom table limits and cost implications, contact your ServiceNow account manager.

SMC has dependency on ServiceNow plugins for managing visibility of resources and aligning with ServiceNow best practices. For more information, see the plugin documentation in the table below.

ServiceNow plugin	Documentation
User Criteria Scoped API	https://docs.servicenow.com/bundle/washingtondc-application-development/page/build/custom-application/concept/build-applications.html

ServiceNow plugin	Documentation
Discovery and Service Mapping Patterns	https://docs.servicenow.com/bundle/store-release-notes/page/release-notes/store/it-operations-management/store-rn-itom-patterns.html

Release notes for AWS Service Management Connector for ServiceNow

The latest version includes support for Xanadu and minor fixes to existing AWS Security Hub integrations. The prior version included enhancements to the existing AWS Health integration.

Version 5.1.3

AWS ServiceNow Connector Core Features

- Supports the latest ServiceNow platform releases Xanadu (X), Washington DC (W), and Vancouver (V).

AWS Security Hub

- Fix an issue with date and timestamp for AWS Security Hub findings to show the correct format.

> **Note**
>
> To maintain the integration capabilities of the Xanadu ServiceNow release, upgrade the connector to version 5.1.3.

Version 5.0.0

AWS Health

- Create incidents, including changes, from AWS Health events.
- Supports affected resource tracking for planned lifecycle events.

- Supports pagination by syncing health events with visual information.
- Supports AWS Organizations to view and consolidate multiple AWS accounts via Amazon EventBridge.
- Updated dashboard that allows selecting accounts and events.
- Support for ServiceNow Vancouver release.
- Support for ServiceNow Washington DC release.

Version 4.8.5

AWS ServiceNow Connector Core Features

- Dashboard that displays reports/charts for AWS Service Catalog, AWS Config, and AWS Security Hub integrations in the ServiceNow platform.
- Support for China Regions (Beijing and Ningxia) for all AWS services compatible with China Regions.
- Support for ServiceNow Utah release.

AWS Service Catalog

- Support for the Terraform open source product type, enabling self-service provisioning with governance for your Terraform configurations within AWS from Service Catalog at scale.
- Fix validation issue with mandatory parameters input on catalog item submission.

AWS Config

- Support for the following new resource types: Amazon WorkSpaces, Amazon Elastic Container Service (ECS), Amazon Elastic Kubernetes Service (EKS), Amazon Elastic File System (EFS), and Amazon RDS Cluster.
- Ability to change synchronization to use many-to-many (MTM) table in the connector.

AWS Systems Manager OpsCenter

- Synchronize **Action Item** type OpsItems from AWS Systems Manager Incident Manager.

Version 4.7.5

AWS ServiceNow Connector Core Features

- Supports latest ServiceNow platform releases for Tokyo (T), San Diego (S), and Rome (R).
- Enables conditional dependency on ServiceNow plugins based on the AWS integrations in use.

AWS Service Catalog

- Ability to filter Service Catalog synced portfolios in the ServiceNow Service Portal using AWS accounts and regions.

AWS Systems Manager Incident Manager

- Displays formatted Timeline Events of an incident in ServiceNow incident comments.
- Provides a new Open Incident module to display in-progress incidents.

Support

- Ability to configure Support cases through automatic incident creation or staged support cases, allowing you to create custom ServiceNow Business Rules and workflow logic.

Version 4.5.5

AWS Systems Manager OpsCenter

- Prevents duplicate incidents created for OpsItems synched to ServiceNow.

Version 4.5.0

AWS Health

- Syncs AWS Health events and resource information.
- Provides a dashboard to view AWS Health status of AWS accounts.

AWS Systems Manager Incident Manager

- Syncs AWS Systems Manager Incident Manager incidents as ServiceNow Incidents.
- Creates relationship between synched incident from Incident Manager and the associated Ops Item.
- Provides configuration to allow bidirectional or unidirectional synchronization of the 'resolved' status between ServiceNow incident and corresponding AWS incident.

AWS ServiceNow Connector Core Features

- Displays AWS account number for validated accounts.
- Supports latest ServiceNow platform releases for Quebec (Q - Patch 5 going forward), Rome (R), and San Diego (S).

AWS Service Catalog

- Provides Service Portal widget to search AWS Service Catalog products from ServiceNow Service Portal.
- Configures independent workflows for different portfolios.
- Provides feature to set a table filter for user selectable Automated Tags.

Support

- Offers near real-time sync of Support cases to ServiceNow using Amazon EventBridge and Amazon SQS queue.
- Syncs Support case severity back into ServiceNow incident.
- Supports AWS accounts with different service accesses.

AWS Security Hub

- Provides revised AWS Security Hub Findings form to show remediation information.

AWS Systems Manager Change Manager

- Syncs AWS CloudTrail events and resource information related to the AWS Change Request.

AWS Config

- Supports Amazon API Gateway resource type.
- Creates relationship between RDS Instances and RDS Cluster, if present.
- Introduces new attribute mappings and relationships on existing resource types.

Version 4.0.1

AWS ServiceNow Connector Core Features

- Supports the latest ServiceNow platform releases for Quebec (Q - Patch 5 going forward), Rome (R), and San Diego (S).

AWS Service Catalog

- Accurately retrieves launch paths/parameters for catalog items in order guides.

Support

- Uses GovCloud accounts with Support integration.

AWS Security Hub

- Syncs ServiceNow Incident state updates to AWS Security Hub Findings.

Version 4.0.0

AWS ServiceNow Connector Core Features

- Uses Guided Setup to enable you to configure and mark complete ServiceNow install components for the AWS Service Management Connector.
- Supports the latest ServiceNow platform releases for Rome (R), Quebec (Q - Patch 5 going forward).

Support

- Views, creates, updates, adds correspondence, and resolves Support cases from ServiceNow as incidents.

- Tracks and manages AWS cases (incidents) within ServiceNow as incidents to ascertain the health of their AWS services and resources as opposed to swiveling between multiple platforms.

AWS Systems Manager Change Manager

- Creates Change Requests from a curated list of AWS Change Templates that are vetted in AWS Systems Manager Change Manager.
- Enables you to customize the change workflow in ServiceNow and streamline and align the maintenance and Service Management governance of AWS resources with your existing Change Management process.

AWS Systems Manager Automation

- Updates mappings to accurately display Status values of Automation document execution in ServiceNow.

Version 3.8.5

AWS ServiceNow Connector Core Features

- Enhances AWS services (AWS Service Catalog, AWS Config, AWS Systems Manager, AWS Security Hub) synchronization to ServiceNow into separate, distinct scheduled jobs.
- Renames 'Sync all Accounts' scheduled job to 'Synchronize changes to all AWS accounts' based on synchronization enhancements.
- Supports the latest ServiceNow platform releases for Rome (R), Quebec (Q - Patch 5 going forward), Paris (P) and Orlando (O).

AWS Service Catalog

- Views AppRegistry applications, attribute groups and linked resources in the ServiceNow CMDB.
- Enables support for ServiceNow order guides for AWS Service Catalog products and AWS Systems Manager automation documents.
- Supports NoEcho parameters when viewing AWS Service Catalog Provisioned Products parameters through ServiceNow Requested Item.

AWS Config

- Adds a configurable ServiceNow system property for AWS Config integration to automatically copy the AWS Resource Id (Object ID in ServiceNow) into ServiceNow's Name field to make AWS resources visible as configuration items.
- Updates ELB resource mapping from cmdb_ci_lb_service table to cmdb_ci_cloud_load_balancer table.
- Updates relationships visible in the ServiceNow CMDB for AWS resources such as Cloud Subnet, DynamoDB, EC2, ELB, RDS, Storage volume, Security groups, and VPC.

AWS Security Hub

- Synchronizes UserDefinedFields JSON blob for Security Hub Findings.

Reference: AWS API calls for the AWS Service Management Connector

The following provides the reference AWS API calls for AWS Service Management Connector.

- `AWSBudgets.describeBudget`
- `AWSCloudFormation.registerType`
- `AWSCloudFormation.deregisterType`
- `AWSCloudFormation.describeTypeRegistration`
- `AmazonConfig.describeConfigurationRecorders`
- `AmazonConfig.getResourceConfigHistory`
- `AmazonConfig.listDiscoveredResources`
- `AmazonConfig.putResourceConfig`
- `AmazonConfig.selectResourceConfig`
- `AmazonConfig.selectAggregateResourceConfig`
- `AWSSecurityHub.batchUpdateFindings`
- `AWSSecurityTokenService.getCallerIdentity`
- `AWSServiceCatalog.createProvisionedProductPlan`
- `AWSServiceCatalog.deleteProvisionedProductPlan`

- AWSServiceCatalog.describePortfolio
- AWSServiceCatalog.describeProduct
- AWSServiceCatalog.describeProductAsAdmin
- AWSServiceCatalog.describeProductView
- AWSServiceCatalog.describeProvisionedProduct
- AWSServiceCatalog.describeProvisionedProductPlan
- AWSServiceCatalog.describeProvisioningParameters
- AWSServiceCatalog.describeRecord
- AWSServiceCatalog.executeProvisionedProductPlan
- AWSServiceCatalog.executeProvisionedProductServiceAction
- AWSServiceCatalog.listBudgetsForResource
- AWSServiceCatalog.listLaunchPaths
- AWSServiceCatalog.listPortfolioAccess
- AWSServiceCatalog.listPortfolios
- AWSServiceCatalog.listProvisionedProductPlans
- AWSServiceCatalog.listServiceActionsForProvisioningArtifact
- AWSServiceCatalog.listStackInstancesForProvisionedProduct
- AWSServiceCatalog.provisionProduct
- AWSServiceCatalog.searchProducts
- AWSServiceCatalog.searchProductsAsAdmin
- AWSServiceCatalog.terminateProvisionedProduct
- AWSServiceCatalog.updateProvisionedProduct
- AWSSimpleQueueService.DeleteMessage
- AWSSimpleQueueService.DeleteMessageBatch
- AWSSimpleQueueService.ReceiveMessage
- AWSSimpleSystemsManagement.describeAutomationExecutions
- AWSSimpleSystemsManagement.describeDocument
- AWSSimpleSystemsManagement.getAutomationExecution
- AWSSimpleSystemsManagement.getDocument

- AWSSimpleSystemsManagement.listDocuments
- AWSSimpleSystemsManagement.startAutomationExecution
- AWSSimpleSystemsManagement.describeOpsItems
- AWSSimpleSystemsManagement.getOpsItem
- AWSSimpleSystemsManagement.updateOpsItem
- AWSServiceCatalogAppRegistry.ListAttributeGroups
- AWSServiceCatalogAppRegistry.GetAttributeGroup
- AWSServiceCatalogAppRegistry.ListApplications
- AWSServiceCatalogAppRegistry.GetApplication
- AWSServiceCatalogAppRegistry.ListAssociatedAttributeGroups
- AWSServiceCatalogAppRegistry.ListAssociatedResources
- Support:DescribeAttachment
- Support:DescribeCommunications
- Support:AddAttachmentsToSet
- Support:AddCommunicationToCase
- Support:CreateCase
- Support:ResolveCase
- Support:DescribeCases
- Support:DescribeServices
- Cloudtrail:DescribeQuery
- Cloudtrail:ListEventDataStores
- Cloudtrail:StartQuery
- Cloudtrail:GetQueryResults
- AWSSimpleSystemsManagementIncident:ListIncidentRecords
- AWSSimpleSystemsManagementIncident:GetIncidentRecord
- AWSSimpleSystemsManagementIncident:UpdateRelatedItems
- AWSSimpleSystemsManagementIncident:ListTimelineEvents
- AWSSimpleSystemsManagementIncident:GetTimelineEvent
- AWSSimpleSystemsManagementIncident:UpdateIncidentRecord

- `AWSSimpleSystemsManagement:ListOpsItemRelatedItems`

Updated key synchronization in ServiceNow

AWS Service Management Connector for ServiceNow allows synchronization of updated keys using any automation or integration through a new REST endpoint.

You can send requests to sync updated keys for one or more AWS accounts registered in the AWS Service Management Connector for either the *Sync User* or *End User* role.

For instructions and information about synching updated keys syntax, see Syncing Updated Keys Programmatically in ServiceNow.

Contacting Service Management Connector specialist team

In AWS Service Management Connector, you can now directly contact the AWS SMC Specialist team through an Support case directly from the Connector.

> **Note**
>
> You must have a Business or Enterprise plan and enable the Support integration while setting up AWS Accounts in the Connector.

To create a support case with Connector team from ServiceNow

1. Log in to your ServiceNow instance as a user (for example, System Administrator) in the fulfiller view (standard user interface view).
2. In the navigator, enter **`AWS Service Management`**.
3. Choose **Incidents** under **Support** to show a list of all synched support cases.
4. Choose **New** from the list header.
5. Complete the mandatory fields on the form.
   - **Subject**- Brief summary of the question or issue
   - **Description** – Detailed account of the question or issue
   - **AWS Account** – AWS account you selected as the support case

- **AWS Service** – AWS Service related to the support case
- **AWS Category** – Category of the case under the related service
- **Caller** – ServiceNow field that identifies the creator of the support ticket

6. Choose **Submit**.
7. Choose the Incident you created from the list.

   The **AWS Case Id** and **AWS Case Status** display.

> **Note**
>
> Alternatively, you can create the support case from Support console.
>
> 1. In the console, choose **Technical Support**.
> 2. Complete the required fields on the form:
>   - Service – **`Service Catalog`**
>   - Category – **`Service Management Connectors`**
>   - Severity – **`General Guidance or System Impaired`** (based on your need)
>   - Subject – Brief summary of the question or issue; include the name of the Connector you use.
>   - Description – Detailed account of the question or issue.
> 3. In **Contact Options**, choose **Web**.
> 4. Choose **Submit**.
>
>   An SMC specialist team member will reach out through the support case.

Upgrading to AWS Service Management Connector from a previous version

To upgrade to AWS Service Management Connector from a previous Connector version in a ServiceNow Production instance, you must:

- Install the Connector in a ServiceNow sandbox instance.
- Follow the Connector installation instructions starting at baseline permissions.

> **Note**
>
> There is a known issue with committing update sets that have a previous version of the Connector installed.
> Previewing the update set is successful. However, at the conclusion of the committing update, an error appears that states: “Version loading was stopped by DictionaryUpdateLoader.…”
> We consider these errors as false positives. After further testing, we determined there is no impact on the update set. AWS logs a ServiceNow support case and provides a new release if needed.

- Compare the two versions to plan how you manage your ServiceNow Development.
- Determine how you want to address Service Catalog provisioned products in previous releases.
- Create a check list of all your transition action items that include, but are not limited to:
  - Transition plan
    - Decision point on Service Catalog provisioned products
    - Steps to update or install the Connector in ServiceNow development to production environments
  - ServiceNow platform admin communications
  - End user communications

Delete application files

(Optional) When you upgrade to the latest connector version, you may have application files that are no longer required. While these files don't pose any risks to the feature set, you can delete them by completing the following steps:

1. Navigate to **System Definition** and then **Fix Scripts**.
2. Open the context (right-click) menu for **Name**, and then choose **Import XML**.
3. Upload the Fix Script.
4. Select `AWSConnector-RemoveDeletedAppFiles`.
5. Choose **Run Fix Script**.

Using AWS Service Management Connector for Jira Service Management Data Center

The AWS Service Management Connector for Jira Service Management (Connector) (formerly the AWS Service Catalog Connector) enables Jira Service Management end users to provision, manage, and operate AWS resources natively through Atlassian's Jira Service Management.

It enables Jira Service Management administrators to:

- Provide preapproved, secured, and governed AWS resources to end users through AWS Service Catalog.
- Create and manage operational items through AWS Systems Manager OpsCenter.
- Execute automation playbooks through AWS Systems Manager Automation.
- Track resources in a configuration item view powered by AWS Config.
- View, create, investigate, add correspondence, and resolve Support cases through Jira Service Management (including AMS Accelerate support cases).
- Manage and resolve incidents affecting AWS-hosted applications through integrations with AWS Systems Manager Incident Manager.

These integrations streamline AWS native services by making it easier for you to consume and provide Jira Service Management governance and oversight over AWS products.

The AWS-supplied connector is available at no charge in the Atlassian Marketplace. This new feature is generally available in all AWS Regions where AWS Service Catalog, AWS Config, and AWS Systems Manager services are available.

Service management alignment

This Connector aligns to industry best practices, such as ITIL®'s service management areas by enabling tools (services) with the intersection of people, processes and partners. The Connector also addresses a baseline set of service management practices you can use in existing operational tooling:

Service management area	AWS service(s) integration
Service Catalog management deployment management (Provisioning)	AWS Service Catalog, AWS CloudFormation, and AWS Systems Manager Automation requests and provisions vetted and predictable products and performs post-provision actions.
Incident management (Ticketing)	Support (AWS services and platform incidents). AWS Systems Manager OpsCenter (Jira operational Issues derived and detected for solutions built on AWS platform). AWS Security Hub (Jira Issues from security Findings). AWS Systems Manager Incident Manager (AWS services and platform incidents).
Service configuration management (CMDB)	AWS Config (Track AWS resources related to the Jira Issue).

In addition, Atlassian Jira Service Management (JSM) is service desk software for modern IT teams. Jira Service Management request types enable self-service for developers and end users to order IT services based on request fulfillment approvals and workflows.

Jira Service Management supported versions

The AWS Service Management Connector (connector) for Jira Service Management Data Center supports Jira software (Jira Service Management) release for both the current and single prior version in each of the major, minor, and point release streams for:

Jira Data Center 7.13.18 to 9.16.1

A Jira Service Management Connector (connector) for Jira Service Management Cloud is also available in the Atlassian Marketplace. For more information, see AWS Service Management Connector for Jira Service Management Cloud.

Release notes

Version 2.0.8 includes updates to core features. Version 2.0.5 of the AWS Service Management Connector for Jira Service Management introduces an integration with AWS Systems Manager Incident Manager cases and Jira incidents.

Version 2.0.8 core features

- Updated package dependencies.

Version 2.0.7 core features

- Updated version of **aws-sdk** library.
- Fix for XML parser issue.

AWS Systems Manager Incident Manager

- Allows Jira Service Management end users to view and resolve a Jira issue when AWS Systems Manager Incident Manager creates or updates an incident.
- Automatically relate an AWS incident to the associated AWS OpsItem when AWS Systems Manager OpsCenter integration is enabled.
- Allows bidirectional or unidirectional synchronization of the 'resolved' status between a Jira issue and a corresponding AWS incident.

The latest version also includes prior integrations to AWS services, such as Support, AWS Security Hub, AWS Service Catalog, AWS Config, AWS Systems Manager automation, and AWS Systems Manager OpsCenter.

Support

- Configure dual synchronization of Support cases with Jira Service Management incidents.
- View, create, resolve and add correspondences to Support tickets directly from Jira Incident.

AWS Security Hub integration

- Configure synchronization of AWS Security Hub Findings within Jira Service Management.
- Create, view, investigate and resolve AWS Security Hub Findings as Jira issues.

- View updates from synced security Findings Jira Issues in AWS Security Hub.

AWS Service Catalog

- Render AWS Service Catalog portfolios and products in the Jira Service Management Customer Portal and Jira Agent views.
- Associate Jira Service Management approval groups to AWS Service Catalog portfolios to require approvals for Jira Service Management user product requests.
- Assign the default Jira user that the Jira workflow engine uses.
- Configure AWS product request form components available for end users to view.
- Create AWS Tags across provisioned products.
- View AWS specific parameters on EC2 resources, such as Availability Zones, Image ID, Instance Id, KeyPair, Security Group, and VPC.

AWS Config

- Render AWS Config configuration item details on provisioned AWS products through Jira Service Management request.
- View the configuration item relationships in a tree structure.
- Associate AWS Config items details to Jira issues.

AWS Systems Manager Automation

- Render AWS Systems Manager automation documents in the Jira Service Management Customer Portal and Jira Agent views.
- Request and execute AWS Systems Manager automation documents through Jira Service Management.
- Create Jira issues (incidents) that provide actionable remediation suggestions through a Connector-specific AWS Systems Manager automation document.

AWS Systems Manager OpsCenter

- Create and update a Jira Issue when you create and update an operational item (OpsItem) in AWS Systems Manager OpsCenter.

- Update OpsItems in AWS Systems Manager OpsCenter when you update the Jira issue in Jira Service Management.
- View and execute automation runbooks to resolve OpsItems and view execution results from the Jira Issue.
- Support multiple AWS accounts.
- Support FIPS endpoints and usage in the AWS GovCloud East and GovCloud West Regions.
- Support the latest releases of Jira Service Management Server and data center versions.

Prerequisites for Jira Service Management Data Center

Before installing the AWS Service Management Connector for Jira Service Management, you need an AWS account and an Atlassian instance with Jira Service Management pre-installed. Verify that you have the necessary permissions in your AWS account and Jira Service Management software.

For a zip file containing Connector add-on code as well as AWS Configuration files, download and extract the AWS Service Management Connector for JSM configuration files.

AWS prerequisites

- To use Service Catalog with the Connector, you need an AWS account to configure your AWS portfolios and products. For more information, see Setting Up AWS Service Catalog.
- To see AWS Config details, configure the service settings to record data for the resource types of interest. We recommend including provisioned products and AWS CloudFormation stacks, in addition to the major resource types your team uses. For more information, see Setting Up AWS Config with the Console.
- To use AWS Systems Manager Automation with the Connector, you don't need AWS-side setup. A number of automation documents are available from AWS as standard. If you want to use additional automation documents, they are available in the Connector. For more information, see Working with Automation Documents (Playbooks).
- To use AWS Systems Manager OpsCenter with the Connector, enable OpsCenter in the AWS Systems Manager console. For more information, see AWS Systems Manager OpsCenter. The Connector also enables viewing resources and automation documents (runbooks) associated to OpsItem. For more information to associate resources to OpsItems in AWS OpsCenter, see Working with Related Resources . For more information to associate automation documents to OpsItems in AWS OpsCenter, see Remediating OpsItem issues using Systems Manager automation.

- To use AWS Security Hub with the Connector, you must enable the service in all Regions and accounts where you want to sync Findings. For more information, see Setting up Security Hub. We recommend you connect Jira Service Management with the primary AWS account for AWS Security Hub. For more information, see Managing administrator and member accounts.
- To use Support with the Connector, your account must have a Business or Enterprise Support plan to use support integration.
- To use AWS Systems Manager Incident Manager with the Connector and allow the Connector to synchronize Incidents for a specific Region, you must enable Incident Manager in that account and Region. For details on the service endpoint, see AWS Systems Manager Incident Manager endpoints and quotas.

> **Note**
>
> AWS Service Management Connector allows AWS Managed Services (AMS) Accelerate users to create Incidents and Service Requests through Jira Service Management. To ensure that your account has the required permissions to create AMS Accelerate support cases, make sure you onboard your account to Accelerate. For more information, see Getting Started with AMS Accelerate.

For each AWS account, the Connector for Jira Service Management also requires API access with Baseline permissions.

Jira Service Management prerequisites

In addition to your AWS account, you need the Jira Service Management software installed on your Atlassian instance before you can install the AWS Service Management Connector add-on. The Jira Service Management administrator needs the *admin* role to install the AWS Service Management Connector add-on.

Before configuring your AWS connector, ensure you follow Atlassian recommendations for securing your Jira Service Management instances. For more information, see Preventing Security Attacks.

The Connector for Jira Service Management add-on is available to download in the Atlassian Marketplace.

Setting up baseline AWS users and permissions

This section provides instructions on how to set up the baseline AWS users and permissions for the AWS Service Management Connector for Jira Service Management.

Topics

Available template for baseline permissions

To use an AWS CloudFormation template to set up the AWS configurations of the Connector for Jira Service Management, see the AWS configurations for Connector for Jira Service Management - AWS Commercial Regions and Connector for Jira Service Management - AWS GovCloud West Region.

> **Note**
>
> If you use the Connector for Jira Service Management AWS Configuration template, go to the Service Catalog Administrator Guide.

For each AWS account, the Connector for Jira Service Management requires two sets of an access key identifier and a secret key for API access. These correspond to users in AWS Identity and Access Management (IAM). Specifically, you should set up:

- An IAM user to sync AWS resources and to sync and manage Support cases through Jira Service Management.
- An IAM user able to perform end user functionality to provision and execute requests exposed through Jira Service Management, including any roles required to perform the provisioning and execution. We recommend launch roles for Service Catalog to comply with IAM best practices.

These can be the same user and can be an existing user. We recommend you assign two new users for Connector.

> **Note**
>
> To align with best practices, AWS recommends periodically rotating IAM user access keys. For more information, refer to Manage access keys for IAM users.

Creating AWS Service Management Connector Sync User

The following section describes how to create the AWS Connector sync user and associate the appropriate IAM permissions. To perform this task, you need IAM permissions to create new users.

To create AWS Service Management Connector sync user

1. Follow the instructions in **Creating IAM Policies** to create the policy, **SSMOpsItemActionPolicy**. This policy enables Jira administrators to create and manage AWS Systems Manager OpsItems.

   Copy this policy and paste it into **Policy Document**:

```
{

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

    "Statement": [

        {

            "Effect": "Allow",

            "Action": [

                "ssm:CreateOpsItem",

                "ssm:GetOpsItem",

                "ssm:UpdateOpsItem",

                "ssm:DescribeOpsItems",

                "ssm:CreateOpsItem"

            ],
```

```
            "Resource": "*"

        }

    ]

}
```

2. Follow the instructions in Creating IAM policies and create the policy, **ConfigBidirectionalSecurityHubSQSBaseline**.

   Copy this policy and paste it in the JSON editor.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"VisualEditor0",
         "Effect":"Allow",
         "Action":[
            "cloudformation:RegisterType",
            "cloudformation:DescribeTypeRegistration",
            "cloudformation:DeregisterType",
            "sqs:ReceiveMessage",
            "sqs:DeleteMessage",
            "securityhub:BatchUpdateFindings"
         ],
         "Resource":"*"
      }
   ]
}
```

3. Follow the instructions in **Creating IAM policies** to create the policy, **AWSIncidentBaselinePolicy**.

   Copy this policy and paste it in the JSON editor.

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

```
   "Statement":[
{
   "Effect":"Allow",
    "Action":[
       "ssm-incidents:ListIncidentRecords",
       "ssm-incidents:GetIncidentRecord",
       "ssm-incidents:UpdateRelatedItems",
       "ssm-incidents:ListTimelineEvents",
       "ssm-incidents:GetTimelineEvent",
       "ssm-incidents:UpdateIncidentRecord",
       "ssm-incidents:ListRelatedItems",
       "ssm:ListOpsItemRelatedItems"
     ],
        "Resource":"*"
     }
   ]
}
```

4. Follow the instructions in **Creating an IAM User in your AWS Account** to create a sync user (SCSyncUser). The user needs programmatic access and AWS Management Console access to follow the Connector for Jira Service Management installation instructions.

   Set permissions for your sync user (SCSyncUser). Choose **Attach the following policies directly** and select **AWSServiceCatalogAdminReadOnlyAccess**, **AmazonSSMReadOnlyAccess**, **SSMOpsItemActionPolicy**, **AWSSupportAccess**, **AWSIncidentBaselinePolicy**, and **ConfigBidirectionalSecurityHubSQSBaseline.**

5. Add a policy that allows **budgets:ViewBudget** on all resources (*).
6. Review and choose **Create User**.
7. Note the access and secret access information. Download the .csv file that contains the user credential information.

Creating AWS Service Management Connector End User

The following section describes how to create the AWS Service Management Connector end user and associate the appropriate IAM permissions. To perform this task, you need IAM permissions to create new users.

To create AWS Service Management Connector end user

1. Follow the instructions in Creating an IAM user in your AWS Account to create a user (such as SCEndUser). The user needs programmatic and AWS Management Console access to follow the Connector for Jira Service Management installation instructions.
2. For products with AWS CloudFormation StackSets, you need to create a stack set inline policy. With AWS CloudFormation StackSets, you can create products to deploy across multiple accounts and Regions.

   Using an administrator account, you define and manage a Service Catalog product and use it as the basis for provisioning stacks into selected target accounts across specified Regions. You need to have the necessary permissions defined in your AWS accounts.

   To set up the necessary permissions, follow the instructions in Granting Permissions for Stack Set Operations to create an **AWSCloudFormationStackSetAdministrationRole** and an **AWSCloudFormationStackSetExecutionRole**.
3. Create the stack set inline policy to enable the provisioning of a product across multiple Regions in one account, replacing the `arn` number string with your account number.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
        "Action": [
        "sts:AssumeRole"
        ],
        "Resource": [
        "arn:aws:iam::123456789123:role/AWSCloudFormationStackSetExecutionRole"
        ],
        "Effect": "Allow"
        },
        {
        "Effect": "Allow",
        "Action": [
        "iam:GetRole",
        "iam:PassRole"
        ],
        "Resource":       "arn:aws:iam::123456789123:role/
AWSCloudFormationStackSetAdministrationRole"
        }
```

```
    ]
}
```

4. Add the following permissions (policies) to the user **SCEndUser**:

   - **AWSServiceCatalogEndUserFullAccess** - (AWS managed policy)
   - **StackSet** - (inline policy)
   - **AmazonS3ReadOnlyAccess** - (AWS managed policy)
   - **AmazonEC2ReadOnlyAccess** - (AWS managed policy)
   - **AWSConfigUserAccess** - (AWS managed policy)
   - **SSMOpsItemActionPolicy** - (inline policy)
   - **ConfigBidirectionalSecurityHubSQSBaseline** - (inline policy)

   > **Note**
   >
   > For Service Catalog products with AWS CloudFormation StackSets, you need to include the read only permissions for the services you want to provision. For example, to provision an Amazon S3 bucket, include the **AmazonS3ReadOnlyAccess** policy to the **SCEndUser** role.

5. Also add a policy that allows the following on all resources (*): **ssm:DescribeAutomationExecutions**, **ssm:DescribeDocument**, and ssm:**StartAutomationExecution**.
6. Review and choose **Create User**.
7. Note the access and secret access information. Download the .csv file that contains the user credential information.

Creating SCConnectLaunch Role

The following section describes how to create the **SCConnectLaunch** role. This role places baseline AWS service permissions into the Service Catalog launch constraints. For more information, see CORRECT LINK.

To create SCConnectLaunch role

1. Create the **AWSCloudFormationFullAccess** policy. Choose **create policy** and then paste the following in the JSON editor.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
            "cloudformation:DescribeStackResource",
            "cloudformation:DescribeStackResources",
            "cloudformation:GetTemplate",
            "cloudformation:List*",
            "cloudformation:DescribeStackEvents",
            "cloudformation:DescribeStacks",
            "cloudformation:CreateStack",
            "cloudformation:DeleteStack",
            "cloudformation:DescribeStackEvents",
            "cloudformation:DescribeStacks",
            "cloudformation:GetTemplateSummary",
            "cloudformation:SetStackPolicy",
            "cloudformation:ValidateTemplate",
            "cloudformation:UpdateStack",
            "cloudformation:CreateChangeSet",
            "cloudformation:DescribeChangeSet",
            "cloudformation:ExecuteChangeSet",
            "cloudformation:DeleteChangeSet",
            "s3:GetObject"
            ],
            "Resource": "*"
        }
    ]
}
```

2. Create a policy called **ServiceCatalogSSMActionsBaseline**. Follow the instructions in Creating IAM Policies, and paste the following into the JSON editor.

```
 {
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Stmt1536341175150",
            "Action": [
                "servicecatalog:ListServiceActionsForProvisioningArtifact",
                "servicecatalog:ExecuteprovisionedProductServiceAction",
                "ssm:DescribeDocument",
                "ssm:GetAutomationExecution",
                "ssm:StartAutomationExecution",
                "ssm:StopAutomationExecution",
                "cloudformation:ListStackResources",
                "ec2:DescribeInstanceStatus",
                "ec2:StartInstances",
                "ec2:StopInstances"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

3. Create the **SCConnectLaunch** role. Assign the trust relationship to Service Catalog.

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

4. Attach the relevant policies to the **SCConnectLaunch** role. Attach the following baseline IAM policies:

   - **AmazonEC2FullAccess** (AWS managed policy)
   - **AmazonS3FullAccess** (AWS managed policy)
   - **AWSCloudFormationFullAccess** (custom managed policy)
   - **ServiceCatalogSSMActionsBaseline** (custom managed policy)

> **Note**
>
> You can use the available AWS CloudFormation templates for the JSM connector to configure your AWS account to enable AWS Service Catalog integration. This stack includes the *Sync user* and *End user* roles, which attach the required permissions for all available integrations. For more information, see Baseline Permissions.

Configuring Service Catalog Integration

After you create two IAM users with baseline permissions in each account, you can now configure Service Catalog. This section describes how to configure Service Catalog to have a portfolio that includes an Amazon S3 bucket product. Use the Amazon S3 template in Creating an Amazon S3 Bucket for Website Hosting for your preliminary product. Copy and save the Amazon S3 template to your device.

To configure Service Catalog

1. Follow the steps in Step 3: Create an AWS Service Catalog Portfolio to create a portfolio.
2. To add the Amazon S3 bucket product to the portfolio you just created, enter the product details in the Service Catalog console on the **Upload new product** page.
3. For **Select template**, choose the Amazon S3 bucket AWS CloudFormation template you saved to your device.
4. Set **Constraint type** to **Launch** for the product that you just created with the **SCConnectLaunch** role in the baseline permissions. For additional launch constraint instructions, see AWS Service Catalog Launch Constraints.

> **Note**
>
> The AWS configuration design requires each Service Catalog product to have either a launch or StackSet constraint. Failure to follow this step can result in an *Unable to Retrieve Parameter* message within Jira Service Management Service Catalog.

Creating Stack Set Constraint

AWS CloudFormation StackSets enable users to create products that deploy across multiple accounts and Regions. In Service Catalog, a stack set constraint allows you to configure product deployment options.

To apply a stack set constraint to a Service Catalog product

1. As an AWS Service Catalog administrator, choose the portfolio that contains the product you want to apply a constraint.
2. Expand **Constraints** and choose **Add constraints**.
3. Choose the product from **Product** and set **Constraint type** to **Stack Set**. Then choose **Continue**.
4. On the **Stack set constraint** page, enter a description.
5. Choose the accounts in which you want to create products.
6. Choose the Regions in which you want to deploy products. Products deploy in these Regions in the order that you specify.
7. Choose the **AWSCloudFormationStackSetAdministratorRole** role to manage your target accounts.
8. Choose the **AWSCloudFormationStackSetExecutionRole** role that the administrator role will assume.
9. Choose **Submit**.

> **Note**
>
> You can use the available AWS CloudFormation templates for the JSM connector to configure your AWS account to enable AWS Service Catalog integration. For more information, see Baseline Permissions.

Example stack set outputs:

```
SCStackSetAdministratorRoleARN
arn:aws:iam::123456789123:role/AWSCloudFormationStackSetAdministrationRole
 SCIAMStackSetExecutionRoleName
AWSCloudFormationStackSetExecutionRole
SCIAMAdminRoleARN
arn:aws:iam::123456789123:role/AWSCloudFormationStackSetAdministrationRole
```

Note that Service Catalog products can have either a stack set or a launch constraint, but not both.

Video: Integrate AWS products in your Jira Service Management portal

This video (11:22) describes how to integrate AWS products into your Jira Service Management portal. Jira Service Management enables end users to provision, manage, and operate AWS resources natively with Jira Service Management from Atlassian.

Integrate AWS Products into Your Jira Service Management Portal

Configuring AWS Security Hub Integration

AWS Security Hub enables users to view security findings from AWS services, such as Amazon Guard Duty, Amazon Inspector, as well as AWS Partner solutions.

If you use both AWS Security Hub and Jira Service Management (JSM), the AWS Service Management Connector for JSM allows you to create an automated, bidirectional integration between Security Hub and JSM. This two-way integration synchronizes your Security Hub findings and Jira issues.

Specifically, as a Jira administrator, you can use this integration to automatically create Jira issues from Security Hub findings. When you update those tickets in Jira, the changes are automatically replicated back to the original Security Hub findings. For example, when you resolve the issue in Jira, the workflow status of the Security Hub finding also changes to RESOLVED. This action ensures Security Hub always has up-to-date information about your security posture.

To configure AWS Security Hub integration features

1. Enable AWS Security Hub. For more information, see Accessing Security Hub.
2. Set up an SQS queue to receive updated Findings. Name the queue **AwsSmcJsmSecurityHubQueue** to align with the default name in the JSM Connector Settings for the AWS Security Hub integration. For more information, see Getting started with Amazon SQS.
3. Set up a Amazon EventBridge rule to detect changes to Findings and push these to the queue. For more information, see Getting started with Amazon EventBridge.

   The CloudWatch rule should have the following event pattern and should point to the SQS queue created in Step 2.

   ```
   "EventPattern": {

          "source": [

           "aws.securityhub"

           ]
   }
   ```

4. You can also customize this CloudWatch Events rule to only pull in Security Hub findings that have specific finding types, severity labels, workflow statuses, or compliance statuses. For details about how to filter the event pattern, see Configuring an EventBridge rule for automatically sent findings in the *AWS Security Hub User Guide*.

> **Note**
>
> You can use the available AWS CloudFormation templates for the JSM connector to configure your AWS account to enable AWS Service Catalog integration. For more information, see Baseline Permissions.

Video: Bidirectional integration with Atlassian Jira Service Management

This video (8:40) describes how to set up a bidirectional integration with Atlassian Jira Service Management. This feature makes it easier for AWS Security Hub users to automatically create and update issues in Jira Service Management from AWS Security Hub findings and ensure that updates to those tickets are synced with the findings.

Integrate AWS Products into Your Jira Service Management Portal

Configuring Support Integration

To enable the Connector to synchronize Support tickets, the account should have a Business or Enterprise Support plan. For more information, see Getting started with Support.

> **Note**
>
> AWS Service Management Connector allows AWS Managed Services (AMS) Accelerate users to create Incidents and Service Requests through Jira Service Management. To ensure that your account has the required permissions to create AMS Accelerate (Accelerate) support cases, make sure you onboard your account to Accelerate. For more information, see Getting Started with AMS Accelerate.

To configure Support integration features

1. Set up an SQS queue (in N.Virginia (us-east-1) for Commercial regions and US West (us-gov-west-1) for GovCloud regions) to receive updates on Support cases. Name the queue **AWSServiceManagementConnectorSupportQueue** to align with the default name within the JSM Connector Settings for the Support integration. For more information, see Getting started with Amazon SQS.
2. Set up an Amazon EventBridge rule to detect changes to Support case and push these to the queue. For more information, see Getting Started with Amazon EventBridge.

   The Amazon EventBridge rule should have the following event pattern and should point to the SQS queue created in Step 2.

   ```
   EventPattern":{
   ```

```
   "source":[
      "aws.support"
   ],
}
```

> **Note**
>
> You can use the available AWS CloudFormation templates for the JSM connector to configure your AWS account to enable AWS Service Catalog integration. For more information, see Baseline Permissions.

For creation of SQS queue and EventBridge rule, use Connector for Jira Service Management - AWS Support Commercial Regions and Connector for Jira Service Management AWS Support GovCloud West Region.

Configuring AWS Systems Manager Incident Manager Integration

To allow the Connector to synchronize Incidents from AWS Systems Manager Incident Manager for a specific Region, you must enable Incident Manager in that account and Region. For more information, see What is AWS Systems Manager Incident Manager.

Configuring Jira Service Management

The AWS Service Management Connector for Jira Service Management is a conventional Jira Service Management add-on. Add-ons are code changes to the Jira software that extend its functionality or extend the functionality of Jira Service Management software. The Connector for Jira Service Management add-on is available to download in the Atlassian Marketplace.

After completing the IAM and Service Catalog configurations, clear your web browser cache to remove previously rendered Jira Service Management forms, and then configure Jira Service Management. Installation tasks within Jira Service Management include:

Topics

- Installing Jira Service Management Connector add-on

- Configuring AWS Accounts and Regions
- Configuring Service Catalog portfolios in Jira

Installing Jira Service Management Connector add-on

Follow these steps to install the Jira Service Management Connector add-on.

1. Log in to your Jira instance as an admin.
2. From the admin menu, choose **Add-ons**.
3. On the **Manage add-ons** screen, choose **Find new apps** or **Find new add-ons** from the left side of the page.
4. Find **AWS Service Management Connector for JSM**. The search results should include app versions compatible with your Jira instance.
5. Choose **Install** to download and install your app.
6. Proceed to **Configuring AWS Accounts and Regions**.

Alternatively, download the AWS Service Management Connector for Jira Service Management file.

1. Go to **Manage apps**.
2. Select **Upload app** and upload the OBR file.
3. Proceed to **Configuring AWS Accounts and Regions**.

You can apply the Connector for Jira Service Management add-on to the supported Jira software (Jira Service Management) releases noted above.

Configuring AWS Accounts and Regions

After you install the AWS Service Management Connector, you need to configure it. To do so, choose the Jira administration icon in the top right, then choose **Add-ons**.

1. From the Service Catalog section on the left navigation menu, choose **AWS Accounts**.
2. Choose **Connect new account**.
3. Enter the account alias (used to identify the AWS account in the Connector).
4. Enter the credentials for SC-sync-user. It is the access key identity and credentials for a sync user saved from the AWS configuration. SC-sync-user credentials can retrieve portfolios and

products to make them available through Jira Service Management. You can set the allowed groups that can access them.

5. Enter the credentials for SC-end-user. It is the access key identity and credentials for the end user saved from the AWS configuration. The SC-end-user credentials provision products on behalf of a Jira user.
6. Add **AWS Regions**. It contains Service Catalog products and portfolios you want available in Jira Service Management.
7. Choose **Test Connectivity**.
8. Upon successful connection status, choose **Connect**.

> **Note**
>
> We recommend the Sync user and End user be new users in AWS, used only with AWS Service Management Connector. These users should have minimum required privileges. You can use the available AWS CloudFormation templates for your sandbox and development AWS accounts to configure and enable available integrations. For more information, see Baseline Permissions.

Configuring Service Catalog portfolios in Jira

This section describes how to configure AWS Service Catalog portfolios within Jira.

Once your account or accounts are set up and connectivity is successful, use the **AWS Account** page to manage, for each account, which groups can access each portfolio in each Region. You can expand and collapse each Region and edit and add groups for each portfolio. Only users in the designated groups have access to those products. By default, no groups have access.

> **Note**
>
> At least one group must be associated to a Service Catalog portfolio for Jira Service Management end users to request AWS products.

To provision products and portfolios

1. Choose **AWS Accounts**.

2. Choose **Manage** for the AWS account in which you want to configure portfolios.
3. Under **Portfolios**, expand the Region associated with the account. Portfolios display under each Region.
4. In the **Permission to request** column, choose **Add groups** for the portfolios that you want to make visible in Jira Service Management. Select the group you want to see and request Service Catalog products.

> **Note**
>
> Because the AWS Service Management Connector for Jira Service Management allows Jira users to provision AWS products in the portfolios their groups have access to, and to control those provisioned products, users should maintain security in their Jira accounts.

5. If products in this portfolio do not require approvals, choose **Save**.

Jira Service Management Approvals for Products in Service Catalog Portfolios

The AWS Service Management Connector for Jira Service Management enables administrators to configure approvals for products at the portfolio level. All products in a portfolio that contain approval permissions require approval, so AWS and Jira administrators might need to collaborate on the Service Catalog portfolio structure.

To configure the approval process

1. Choose **AWS Accounts**.
2. Choose **Manage** on the AWS account for which you want to configure portfolio approvals.
3. In the **Permission to approve** column, choose **Add groups** for the portfolios that require product approvals.
4. Select **Require approval for provisioning**.
5. Under **Permission to approve**, choose **Add group**.
6. Choose **Save**.

> **Note**
>
> If a portfolio only has a group associated with **Permissions to request**, products in the portfolio immediately provision when you submit the product request.

Products and budgets

For reference, two other tabs in the **Admin - AWS Accounts - Manage** section let you view information on portfolios.

The **Available Products** tab lists the products in the portfolio and budgetary information on each. The **Budgets** tab gives overall budgetary information on the portfolio.

> **Note**
>
> Find details about additional configurations for the AWS Service Catalog request form and Automated Tags in the next section Configuring Connector Settings.

Configuring Connector Settings (Jira Project Enablement and Request Type)

In addition to configuring AWS accounts, the AWS Service Management Connector contains AWS services and UI settings (AWS Service Catalog) that enable projects and configure AWS Systems Manager OpsCenter.

> **Note**
>
> There are no per-account settings for AWS Config and AWS Systems Manager Automation through the JSM Connector.

Connector features enabled by default

To configure the default Connector features for specific AWS services

For a new installation of Connector, we enable the default project configuration for all Connector features (AWS Service Catalog, AWS Config, AWS Systems Manager Automation, AWS Systems

Manager OpsCenter, and AWS Security Hub). If you are upgrading an existing installation, for security reasons, we do not intially enable new features.

> **Note**
>
> If you are using the AWS Security Hub integration, we recommend you also turn on AWS Config.
> If you use the AWS Config integration with JSM, this might add more resource details in JSM issues created for AWS Security Hub Findings. For example, if the original Finding has limited resource details, the Config resource enrichment provides fuller information.
> Also, if the resource no longer exists, the Config enrichment provides information about the resource status. If the resource details changed since the creation of the Finding, the Config enrichment provides the latest details, but it does not overwrite the original details.

1. In the left navigation menu, under **AWS Service Management**, select **Connector settings.**
2. At the top, under **Connector features enabled by default**, select each feature depending whether you want projects using the default configuration to be able to use them or not.
3. Choose **Save**.

Configuring UI Settings (AWS Service Catalog)

Configure the AWS Service Catalog product widget components to make them viewable to end users.

To address the varying personas of end users requesting AWS products, the Connector for Jira Service Management includes an add-on app setting to enable or disable components of the AWS product widget. By default, we enable AWS product components.

To modify the AWS product view

1. In the left navigation menu, under **AWS Service Management**, choose **AWS Connector settings**.
2. In the **UI settings** (Service Catalog) section, deselect any AWS product component such as:
    1. Allow the product name to be edited. (If unchecked, we provide an autogenerated name the user cannot edit.)

2. Allow the user to select a launch option. (If unchecked, we select the default launch option and hide it.)
3. Allow the user to select a product version. (If unchecked, we select the default product version and hide it.)
4. Allow the user to add or edit tags. (If unchecked, we select the default values for tag options and hide it.)
5. Allow user to create a plan for creation or update of a provisioned product. (If unchecked, we hide the plans section.)

3. Choose **Save**.

Configuring projects enabled for the Connector

The AWS Service Management Connector for Jira Service Management requires the add-on to be associated to one or more Jira projects and for JSM request types. You can configure which Connector features are enabled for each Jira project.

To configure the Jira projects for AWS Service Catalog, AWS Config, AWS Systems Manager Automation, AWS Systems Manager OpsCenter, AWS Security Hub, Support, and AWS Systems Manager Incident Manager.

1. In the left navigation menu, under **AWS Service Management Connector**, choose **Connector settings**.
2. Under **Projects enabled for Connector**, you must enable at least one Jira project. You can create a new Jira Service Management project or add an existing one. Only users with access to the associated project can access the Connector. When you apply this update, the Connector adds the necessary issue types and other Jira items for AWS Service Catalog products to be available in those projects. You can return to this screen and add or remove projects at any time.
3. Projects initially take the default configuration for which Connector features are enabled. Choose **Edit** in a project row to change the configuration for individual projects. We permit projects to use more features than the default.
4. Choose **Save.**

> **Note**
>
> For end-users to be able to request AWS Service Catalog products, one or more projects must be enabled and users must have Jira permissions to create issues in the Jira project and Permission to Request in the Jira settings for the AWS Account for at least one portfolio with products.

AWS Systems Manager Automation enablement considerations

We currently do not support fine-grained permissions in Jira for which users and groups should be allowed to access which AWS Systems Manager automation documents. If you enable a project for Systems Manager Automation, then any user with permission to create issues in that project can run any of the automations. You can restrict access by limiting which users have access to projects with AWS Systems Manager Automation enabled.

Associate Jira projects to the AWS Systems Manager OpsCenter integration

Once you've enabled projects for the Connector, AWS Systems Manager OpsCenter requires Jira admins to associate Jira project(s) to this integration, as well as determine the full sync and delta sync intervals.

To associate the Jira projects enabled for the Connector to the AWS Systems Manager OpsCenter integration features

1. In the left navigation menu, under **AWS Service Management Connector**, choose **Connector settings**.
2. Create a new Jira Service Management Project. Under **OpsCenter Configuration**, you must enable at least one Jira project. You can create a new Jira Service Management project or add an existing one. Only users with access to the associated project can access the Connector. When you apply this update, the Connector adds the necessary issue type to associated project(s). You can return to this screen and add or remove projects at any time.
3. Under **AWS Systems Manager OpsCenter Configuration**, in the **Full Sync Interval** and **Delta Sync Interval** fields, you can change the sync interval if you want. The **Full Sync** and **Delta** interval determines how often Jira Service Management conducts syncs all or changes to OpsItems details with AWS Systems Manager OpsCenter respectively. Increasing this number

reduces the number of API calls to AWS, but increases the time for OpsItems updates to reflect in the Connector.

4. Choose **Save**.

Associating Jira projects to the AWS Security Hub integration

After you've enabled projects for the Connector, AWS Security Hub requires Jira admins to associate Jira project(s) to this integration, and configurations to manage the Security Hub integration.

To associate the Jira projects enabled for the Connector to the AWS Security Hub integration features

1. In the left navigation menu under **AWS Service Management Connector**, choose **Connector settings**.
2. Create a new Jira Service Management Project.

   Under **Security Hub Configuration**, you must enable at least one Jira project. You can create a new Jira Service Management project or add an existing project. Only users with access to the associated project can access the Connector.

   When you apply this update, the Connector adds the necessary issue type to associated project(s). You can return to this screen and add or remove projects at any time.
3. Under **AWS Security Hub Configuration,** in the **Sync Interval** field, you can change the sync interval if you want. **SQS Queue Name** and **Number of messages to pull from SQS** set the Amazon SQS queue and the polling size, respectively. **Synchronize AWS Security Hub Findings according to their Severity value** determines the Findings with specific severities that sync to the JSM project.
4. Choose **Save**.

Associate Jira projects to the Support integration

After you enable projects for the Connector, Support integration requires Jira admins to associate Jira project(s) to this integration, as well as determine the SQS Queue Name and sync intervals.

To associate the Jira projects enabled for the Connector to the AWS Systems Manager OpsCenter integration features

1. In the left navigation menu, under **AWS Service Management Connector**, choose C**onnector settings**.
2. Create a new Jira Service Management Project.

   Under **Support Configuration**, you must enable at least one Jira project. You can create a new Jira Service Management project or add an existing one. Only users with access to the associated project can access the Connector.

   When you apply this update, the Connector adds the necessary issue type to associated project(s). You can return to this screen and add or remove projects at any time.
3. Under **Support Configuration**, in the **Sync Interval,** you can change the sync interval if you want. The **Sync Interval** determines how often Jira Service Management conducts syncs for all **AWS Services** and **AWS Categories**. **SQS Queue Name** identifies the Amazon SQS queue from which the Support case events sync to JSM
4. Choose **Save**.

Associating Jira projects to the AWS Systems Manager Incident Manager integration

Once you've enabled projects for the Connector, AWS Systems Manager Incident Manager integration requires Jira admins to associate Jira project(s) to this integration, as well as determine the full sync and delta sync intervals.

To associate the Jira projects enabled for the Connector to the AWS Systems Manager Incident Manager integration features

1. In the left navigation menu, under **AWS Service Management Connector**, choose **Connector settings**.
2. Create a new Jira Service Management Project. Under **Incident Manager Configuration**, you must enable at least one Jira project. You can create a new Jira Service Management project or add an existing one. Only users with access to the associated project can access the Connector. When you apply this update, the Connector adds the necessary issue type to associated project(s). You can return to this screen and add or remove projects at any time.

3. Under **AWS Systems Manager Incident Manager Configuration**, the **Synchronization of the resolved status**, determine whether a resolution of an Incident from AWS should transition the corresponding Jira issue to the **Resolved** Status or the inverse. The default sync interval for this integration is one minute.
4. Choose **Save**.

Configuring core operational settings

To configure operational settings for the AWS Service Management Connector for Jira Service Management

1. In the left navigation menu, under **AWS Service Management Connector**, choose **Connector settings**.
2. Under **Core operational settings**, in the **Synchronization interval** field, you can change the sync interval if you want.

   This interval determines how often Jira Service Management syncs with AWS. Increasing this number reduces the number of API calls to AWS, but increases the time for updates in AWS portfolios and automation documents to reflect in the Connector. Information on actively provisioning products and ongoing automation executions updates are more frequent.
3. Under **Core operational settings**, in the **JIRA Administrator to run as** field, you can change the admin user assigned to perform automated operations within JIRA.

   > **Important**
   >
   > The Connector performs many actions within Jira, and needs to do those actions as a Jira user. By default, Connector chooses the Jira Admin user with the lowest ID, which works for many environments.

   However, that approach might be the wrong strategy if the initial admin user has been disabled, or if there is a different admin user. For clarity within the Connector, it can be a good idea to create a new user called, for example, "AWS Connector Admin", and select that as the default user.

   We record actions performed automatically by the Connector as being performed by this user, such as synchronizing OpsItems from AWS or adding a comment for changes to an AWS

provisioned product. These actions do not affect actions that end users perform, such as requesting a provisioned product or manually creating an OpsItem in Jira, which we record as the end user performing the action.

This user should have global admin permissions, JSM permissions, and admin access to each of the AWS-enabled projects.

4. Choose **Save**.

> **Note**
>
> We recommend no changes to entities that the plugin created, such as the addition of fields, workflows, issue types, screens, and so on.

Configuring automated tags for AWS Service Catalog

The AWS Service Management Connector v1.9.0 enables Jira administrators to add tags (metadata) to AWS Service Catalog provisioned products globally across the add-on or granularly at the portfolio level. These tags are not visible to end users.

Two tag types are available in this release:

- Generic tags in which the admin can enter the key and value.
- AWS Service Catalog Request Type tags in which the admin can enter the following syntax for key and value:

AWS Service Catalog Request Type tags

Key	Value
Project Code	${PROJECT_CODE}
Project Name	${PROJECT_NAME}
Project Name	${ISSUE_ID}
Username	${USERNAME}
Opened By	${OPENED_BY}

To add generic AWS tags to AWS Service Catalog provisioned products in Jira Service Management

1. In the left navigation menu, under **AWS Service Management**, select **Automated Tags**.
2. For Global level tags, enter the Key and Value entries. Under **Portfolio**, select **Global** (set by default). Choose the **+** icon to insert.
3. For Portfolio level tags, enter the Key and Value entries. Under **Portfolio**, select the Portfolio dropdown to choose the portfolio associated to associate tag. Choose the **+** icon to insert.

To add in-scope request type AWS tags to AWS Service Catalog provisioned products derived from Jira Service Management

1. In the left navigation menu, under **AWS Service Management**, choose **Automated Tags**.
2. For Global level tags, enter the Key and Value entries. Under **Portfolio**, select **Global** (set by default). Select the **+** icon to insert.
3. For Portfolio level tags, enter the Key and Value entries. Under **Portfolio**, select the Portfolio dropdown to choose the portfolio to associate with the tag. Choose the **+** icon to insert.

   After the product provisions, you can see in the AWS console that these tags are associated to the resource.

Configuring project request type groups

The AWS request type must be in a group for users to be able to access it in Jira Service Management. Enabling Jira projects, as described in Configuring Connector Settings (Jira Project Enablement and Request Type), makes AWS product request types available, but Jira Service Management users won't see the request type until you add it to a **Request Type Group**.

To configure request types

1. In the AWS Service Management Connector for Jira Service Management, go to the **Connector settings** page.
2. In the **Projects** section, choose **add the AWS request type**.
3. Choose **Add existing request type** in the upper right-hand corner.
4. Choose **Request AWS product** from the available request type.
5. Choose **Edit Groups** for the **Request AWS product** request type.

6. On the **Edit groups** form, choose **General**, then choose **Save**.

> **Note**
>
> When you create a custom **Request AWS Product** request type for the Connector for Jira Service Management, you do not need to edit to the **Request AWS Product** request type. You can add a request type to an existing group. If you don't have a group, create a new group and add the request type to it.

Setting up AWS resources through Jira Service Management to natively manage resources

The AWS Service Management Connector for Jira Service Management allows Jira Service Management end users to provision, manage, and operate AWS resources natively through Atlassian's Jira Service Management.

- AWS Config linked resources
- Suggested AWS Systems Manager remediations for an issue

The Connector provides two fields to use for any issue.

- *AWS Config Linked Resources*: enables any resource with an entry in AWS Config to have its AWS Config information displayed on the issue in Jira. You can expand and see the information. You can link multiple AWS resources to an issue.
- *AWS Systems Manager Automation Suggested Remediation*: enables SSM automation documents to be recorded against an issue. They then display, as suggested, ways to correct the issue. When a Jira user views the issue, they can see these suggested remediations and choose to apply them. You can attach multiple suggested remediations to an issue.

You can use the two fields individually, but they work very well together. Upon detecting an incident on an AWS resource or set of resources, setting both allows a Jira user to see the configuration information to confirm or better understand the problem, apply remediations to fix common problems, and then confirm in the AWS Config information that the problem has been fixed.

To add AWS fields to an existing issue

1. You must enable the project or projects for the Connector in **Connector Settings** under **Admin -> Manage Add-Ons**, as described in the Connector setup guide.
2. In **Admin**, **Projects**, open the project you want to use these fields.
3. Choose the issue type you want to use in the menu at left.
4. Choose to view **Fields** in the top right (if not already selected). It should then show a list of fields enabled for the screen.
5. Scroll to the bottom where there should be a textbox where you can enter additional fields. Enter **AWS**, then choose the AWS field you want to use.
6. Choose **Add** to apply.
7. Repeat the previous step for the other field if you want to use it.
8. Repeat these steps for each issue type you want to use these fields. Some issue types might share screens so the field might already be added for some.

It is important also to make a note of the field ID for the field or fields you are using. Choose **Admin -> Issues -> Custom fields** and select **Configure** on each field.

Inspect the opened URL to see the numeric field ID. It should be a 5-digit number.

Alternatively, for any issue in a project where you've added the field (following the instructions above), the REST API at `/rest/api/2/issue/PRJ-1/editmeta` (for example, `http://localhost:2990/jira/rest/api/2/issue/PRJ-1/editmeta`) will include information on the fields.

The REST API should contain an entry `customfield_#####: { ..., name: "AWS Config Linked Resources", ... }`, where `#####` is the numeric field ID.

Once these fields are enabled for projects and issue types, use the Jira REST API to create or update issues with values for these fields. You can use tools such as CloudWatch, AppDynamics, Jenkins, or a Systems Manager Automation Document (provided in the next section).

The REST API endpoint to update an issue is `/rest/api/2/issue/issue-key` and the general schema to pass to set a value is as follows:

```
                    { "update": {
      "customfield_field-ID": [ {
        "set": "value"
      } ]
  } }
```

See the examples below, or for more information on the REST API, see JIRA Developer Documentation : Updating an Issue through the JIRA REST APIs.

AWS Config Linked Resources

The **AWS Config Linked Resources** field should be set to the JSON string representation of a list of objects (maps) corresponding to the linked resources, each with the following keys:

- *resourceId*: the ID of the resource in AWS Config
- *resourceType*: the type of the resource in AWS Config
- *accountName*: the name or alias of the AWS account configured in Jira that should be used to access this resource
- *region*: the Region where AWS Config should be accessed to get information on this resource

For example, the following value would show information on the S3 bucket `my-bucket` in `eu-central-1`, using the account and end user credentials specified in Jira for the AWS account identified in Jira as `MyAccount1`:

```
            [ { "resourceId": "my-bucket",
      "resourceType": "AWS::S3::Bucket",
      "accountName": "MyAccount1",
      "region": "eu-central-1" } ]
```

AWS Systems Manager Automation Suggested Remediation

The **AWS Systems Manager Automation Suggested Remediation** field should be set to the JSON string that represents a list of objects (maps) that correspond to the automation documents as remediations, each with the following keys:

- *documentName*: the name of the Systems Manager automation document

- *description*: a description of the remediation to display in Jira; this may be different to the document description in AWS and might explain why it is a good remediation for the issue where this is being set
- *accountName*: the name or alias of the AWS account configured in Jira that should be used to access this resource
- *region*: the Region where AWS Config should be accessed to get information on this resource

For example, the following value would suggest the `AWS-DisableS3BucketPublicReadWrite` automation document, with a description to show in Jira, to apply in `eu-central-1`, using the account and end-user credentials that is specified in Jira for the AWS account identified in Jira as `MyAccount1`:

```
                              [ { "documentName": "AWS-
DisableS3BucketPublicReadWrite",
        "description": "This will make the bucket private, resolving the issue.",
        "accountName": "MyAccount1",
        "region": "eu-central-1" } ]
```

Scripting Field Creation

As an example, the following bash script using curl links the above-noted resource to an issue and attaches a suggested remediation. The values used below assume Jira is at *localhost:2990/jira* with login *admin:admin*, the issue is *PRJ-1*, and the field IDs are 10011 (AWS Config linked resources) and 10010 (suggested remediation). These should be changed to reflect your environment.

1. Set the following to correspond to your environment and issue:

   JIRA_BASE_URL=http://localhost:2990/jira

   JIRA_USER_PASS=admin:admin

   ISSUE_KEY=PRJ-1
2. Set the field ID and edit the JSON record for an AWS Config resource to link.

   ```
                              CUSTOM_FIELD_ID=customfield_10011
   ```

```
cat > value.json  EOF
    [ { "resourceId": "my-bucket",
        "resourceType": "AWS::S3::Bucket",
        "accountName": "MyAccount1",
        "region": "eu-central-1" } ]
EOF
```

3. Define a helper function to escape the JSON.

```
                        json_escape () {
    printf '%s' "$1" | python -c \
      'import json,sys; print(json.dumps(sys.stdin.read()))'
}
```

4. Make the REST call to set the AWS Config Linked Resource field.

```
                        curl -v -D- -X PUT  -H "Content-Type: application/json" \
  --data '{ "update": { "'${CUSTOM_FIELD_ID}'": [ {"set": '"$(
     json_escape "$(cat value.json)")"' } ] } }' \
  -u admin:admin ${JIRA_BASE_URL}/rest/api/2/issue/${ISSUE_KEY}
```

5. Set the field ID and edit the JSON record for a suggested remediation to attach.

```
                        CUSTOM_FIELD_ID=customfield_10010
cat > value.json  EOF
    [ { "documentName": "AWS-DisableS3BucketPublicReadWrite",
        "description": "This will make the bucket private, resolving the issue.",
        "accountName": "MyAccount1",
        "region": "eu-central-1" } ]
EOF
```

6. Make the REST call to set the **AWS Systems Manager Automation Suggested Remediations** field.

```
                      curl -v -D- -X PUT  -H "Content-Type: application/json" \
--data '{ "update": { "'${CUSTOM_FIELD_ID}'": [ {"set": '"$(
   json_escape "$(cat value.json)")"' } ] } }' \
-u ${JIRA_USER_PASS} ${JIRA_BASE_URL}/rest/api/2/issue/${ISSUE_KEY}
```

The issue should then show AWS Config for the bucket and a suggested remediation to make it private.

Creating issues with suggestions and a linked AWS resource from AWS Systems Manager

A Systems Manager Automation Document can automatically create a Jira issue with the fields set to have a linked AWS resource and up to three suggested remediation documents.

To install this automation document, download and extract the JSM Connector Create Remediation Issue Automation and IT Lifecycle Demo.zip that contains two files:

- *JSMConnector-CreateRemediationIssue.ssmdoc.yaml*
- *JSMConnector-function.zip*

Follow these steps

1. Upload the file *JSMConnector-function.zip* to a bucket. In the following command, replace ${BUCKET} with the appropriate bucket:

```
aws s3 cp JSMConnector-function.zip s3://${BUCKET}/function.zip
```

2. Create the Systems Manager Automation Document, called **JSMConnector-CreateRemediationIssue**, with the contents from the file *JSMConnector-CreateRemediationIssue.ssmdoc.yam*l and an attachment *Key=SourceUrl,Values=s3://*

${BUCKET}/, using the bucket name from the previous step as ${BUCKET}. The following command replaces ${BUCKET}):

```
aws ssm create-document --name "JSMConnector-CreateRemediationIssue" --content
 "file://JSMConnector-CreateRemediationIssue.ssmdoc.yaml" --document-type
 "Automation" --document-format "YAML" --attachments "Key=SourceUrl,Values=s3://
${BUCKET}/"
```

Once installed, enter the parameters and run it. Note that it requires many of the same parameters, as described previously to connect to Jira.

You should then see an issue in Jira with AWS Config information and the suggested remediation shown.

Sample Use Case: Automatically Creating Issues for IT Lifecycle Management - Remediating non-compliant public S3 buckets

Once you enable the fields to an issue and create the Systems Manager Automation Document, you can set up rules to automatically create Jira issues for common problem categories in AWS. You can also include suggested remediations to make it easy for Jira agents and end users to see problems and fix them.

This demo creates a Config Rule in AWS, which detects public S3 buckets and makes it possible for Jira agents or end users to disable public access directly from Jira.

You should set up prerequisites, roles for the automation and lambda to execute, and the Jira password as a secure string in Systems Manager Parameter Store.

To store the Jira password securely in Parameter Store

1. Open the AWS Console and go to **Systems Manager -> Parameter Store**.
2. Choose **Create parameter**.
3. Set the name as **`jira_password`**.
4. Set the type as **`SecureString`**.
5. Set the value as the password for the Jira user to create issues.
6. To save, choose **Create parameter**.

An AWS CloudFormation template assists setting up the role and configuration rule: **JSMConnector-CreateRemediationIssue-MakePublicBucketsPrivateConfigRule.cfn.yaml**

Install the template, setting the following parameters:

- **JiraURL**: the base URL to your Jira, such that appending */rest/...* after it accesses the REST API
- **JiraUsername**: the username to log in to Jira (with the password specified in *jira_password*)
- **SSMParameterName**: *jira_password* (the parameter containing the Jira password)
- **ProjectKey**: the key of the project (the token before the *-n an issue*), such as *PRJ*.
- **IssueTypeName**: must exactly match the name of the issue type on the project in Jira
- **JiraAwsAccountName**: the name of the AWS Account as configured in the Connector in Jira
- **JiraAwsAccountRegion**: the Region of this violating resource, e.g. *us-east-1*
- **JiraAwsResourceFieldId**: the field ID of the AWS Config Linked Resources field in Jira, such as *customfield_10011*.
- **JiraRemediationsFieldId**: the field ID of the **AWS Systems Manager Automation Suggested Remediation** field in Jira, such as *customfield_10010*.

The Config Rule runs automatically within the period specified. To see it in action immediately:

1. Create a public Amazon S3 bucket.
2. Open the Config Rule in AWS Config and choose **Re-evaluate**. The rule and the automation can take a short while to run, but within a few minutes you should see a new issue in Jira with AWS Config information for the bucket, which is in violation and suggests the **DisableS3BucketPublicReadWrite** automation document as a remediation.

Validating AWS Service Management Connector configurationsfor for Jira Service Management

You can validate the AWS Service Management Connector for Jira Service Management installation procedures.

Topics

- Validating AWS Systems Manager OpsCenter integration
- Validating Support integration
- Validating AWS Systems Manager Incident Manager integration
- Validating AWS Security Hub integration

Validationg Service Catalog integration

To validate Service Catalog integration, order a Service Catalog product or view provisioned products.

To order a Service Catalog product

1. Log in to your Jira Service Management customer portal as the end user.
2. In the Jira Service Management customer portal, choose **Request AWS product**.
3. Enter **Summary** details.
4. Open the **AWS product request detail** menu and select a product to provision.
5. Fill in the product request details, including product reference name, parameters, and tags.
6. Choose **Create** to submit the Jira Service Management request and provision the Service Catalog product.
7. After the request processes, a message appears indicating that your request was created. When the product is ready to provision, the end user receives a notification that the product is launching.

To view provisioned products

1. In the Jira Service Management customer portal, choose **Requests** in the upper right corner.
2. Choose **My Requests** in the Jira Service Management customer portal view.
3. Choose the AWS product you requested.
4. The AWS product details display, including the status of the product request, product events, and activities.
5. If that Connector feature is available, AWS Config information appears. You can expand **Configuration Items** or **Relationships** to see more information. Related resources can be loaded by continuing to expand them underneath the **Relationships** section.

6. Once the product is in the **Available** status, end users can request post-provision operations actions such as **Request update**, **Request termination**, and **Request self-service actions**. These actions render additional product events and activities within the request. Once the product terminates, the request closes in a resolved state.

Validating AWS Systems Manager Automation integration

To validate AWS Systems Manager Automation integration, execute an automation document and view automation executions.

To execute an automation document

1. Log in to your Jira Service Management customer portal as the end user.
2. In the Jira Service Management customer portal, choose **Request AWS automation**.
3. Enter **Summary** details.
4. Open the **AWS automation request detail** menu and choose an automation document to execute.
5. Enter the automation request details, parameters, and tags.
6. Choose **Create** to submit the Jira Service Management request and execute the AWS Systems Manager Automation Document.
7. After the request processes, a message indicates the completion of the request. As the automation executes, the end user receives a notification of progress.

To view automation executions

1. In the Jira Service Management customer portal, choose **Requests** in the upper right corner.
2. Choose **My Requests** in the Jira Service Management customer portal view.
3. Choose the AWS automation execution you requested. The AWS automation execution details displays and includes the status of the execution, request details, and steps.

Validating AWS Systems Manager OpsCenter integration

To validate AWS Systems Manager OpsCenter integration, view or create OpsItems.

To view OpsItems in Jira Service Management from AWS Systems Manager

1. Log in to your **Jira Agent** view as an end user.
2. In the **Jira Service Management Jira Agent** view, choose the Jira project associated to OpsCenter
3. Choose **Open Issues** and select the **OpsItem** from AWS that you want to view.

To create AWS Systems Manager OpsItems in Jira Service Management

1. Log in to your **Jira Agent** view as an end user.
2. In the **Jira Service Management Jira Agent** view, choose **Create**.
3. In the **Create Issue** field input the following details:
   - **Project**: Auto-populated.
   - **Issue Type**: Choose **AWS OpsItem** if you have multiple issue types.
   - **Summary**: Input Summary Details.
   - **Description**: Input Description.
   - **Priority**: Choose the appropriate Priority (default value is Low).
   - **Severity**: Choose the appropriate Severity (required for AWS OpsItem).
   - **Category**: Choose the appropriate Category (required for AWS OpsItem).
   - **Region**: Choose the appropriate AWS Region (required for AWS OpsItem).
4. Choose **Create**.

> **Note**
>
> The newly created OpsItem from Jira Service Management displays in the AWS account view of OpsItem on the next sync between AWS and Jira Service Management.

To update AWS Systems Manager OpsItems in Jira Service Management

1. Log in to your **Jira Agent** view as an end user.
2. In the **Jira Service Management Jira Agent** view, choose the Jira project associated to OpsCenter.
3. Choose **Open Issues** and select the **OpsItem** from AWS that you want to update.

4. Choose **Edit Issue**.
5. Update fields available such as Summary, Description, Priority, Severity, Category. The **Resolved** button in the OpsItem issue is also available to select upon resolution.

> **Note**
>
> Updates to OpsItem fields from Jira Service Management displays in the AWS account view of OpsItem on the next sync between AWS and Jira Service Management.

To view AWS related resources in AWS Systems Manager OpsItems through Jira Service Management

1. Log in to your **Jira Agent** view as an end user.
2. In the **Jira Service Management Jira Agent** view, choose the Jira project associated to OpsCenter.
3. Choose **Open Issues** and select the **OpsItem** from the OpsItem from AWS.
4. Choose the AWS related resource section of the OpsItem selected. This section displays the related resource details.

To execute runbooks on AWS Systems Manager OpsItems through Jira Service Management

1. Log in to your **Jira Agent** view as an end user.
2. In the **Jira Service Management Jira Agent** view, choose the Jira project associated to OpsCenter.
3. Choose **Open Issues** and select the **OpsItem**.
4. Choose the OpsItem section of AWS Runbooks. The OpsItem that contains the associated runbooks display a list of automation documents available. (See them next to the star shaped symbol.)

   - Choose **Execute** on the desired runbook. An **Execute Runbook from OpsItem** screen displays.
   - Enter the workflow parameter details associated to the runbook. The runbook will not execute successfully without the correct parameter inputs.
   - Enter metadata tags details if applicable.

- **Select Create**. An **Execute AWS Systems Manager Automation Request** issue generates and provides the execution status.

OpsItems without associated runbooks are still able to run automated documents.

To run automated documents not associated with runbooks

1. In the OpsItem, choose **Show All Runbooks**. A list on AWS Runbooks display.
2. To narrow the list of runbooks available, enter details into the search bar above the first listed runbook.
3. Choose **Execute** on the desired runbook. An **Execute Runbook from OpsItem** screen displays.
4. Enter the workflow parameter details associated to the runbook. The runbook will not execute successfully without the correct parameter inputs.
5. Enter metadata tags details if applicable.
6. Choose **Create**. An **Execute AWS Systems Manager Automation Request** issue displays and provides the execution status.

Validating Support integration

This section describes how to create, view, and manage integration features for Support.

To view Support cases from Support as Jira incidents

1. Log in to your **Jira Agent** view as an end user.
2. In the **Jira Service Management Jira Agent** view, choose the Jira project associated to Support
3. Choose **Incidents** and select the Incident related to the Support case in AWS

To create a general Support case as a Jira incident

1. Log in to your Jira Agent view as an end user.
2. In the Jira Service Management Jira Agent view, choose the Jira project associated to Support.
3. Choose **Create** from list header and select Issue Type as **Incident**.
4. Complete the mandatory fields on the form.

   Under the Jira Issue Fields section

- **Summary**- Brief summary of the question or issue
- **Description** – Detailed account of the question or issue
- **Priority** – Severity of the AWS Support case

Under Support fields section

- **Create Support case** – Check this box to create support case
- **Support Service and Category** – AWS Service and Category of the support case
- **AWS Cc Email Addresses** – Add cc email addresses to the Support case (not mandatory)

5. Choose **Create**.
6. Choose the Incident you created from the list. The **AWS Case Id** and **AWS Case Status** displays.

For AWS managed services Accelerate customers to create AMS Accelerate Report Incident in Jira

1. Log in to your **Jira Agent** view as an end user.
2. In the **Jira Service Management Jira Agent** view, choose the Jira project associated to Support.
3. Choose **Create** from list header and select Issue Type as **Incident**.
4. Complete the mandatory fields on the form.

Under **Jira Issue Fields** section

- **Summary**- Brief summary of the question or issue
- **Description** – Detailed account of the question or issue
- **Priority** – Severity of the Support case

Under **Support fields** section

- **Create Support case** – Check this box to create support case
- **AWS Support Service and Category** – Select AMS Operations – Service Request and choose category
- **AWS Cc Email Addresses** – Add cc email addresses to the Support case (not mandatory)

5. Choose **Create**.

6. Choose the Incident you created from the list. The **AWS case Id** and **AWS case status** displays.

To add a correspondence and attachment to an existing Support case in Jira incident

1. Log in to your **Jira Agent** view as an end user
2. In the **Jira Service Management Jira Agent** view, choose the Jira project associated to Support.
3. Choose **Incidents** and select the Incident related to the Support case in AWS.
4. Use **Add Comment** action or scroll to the bottom of the form and **Click to add comment** to add a correspondence with or without attachments
5. Choose **Share with customer**.

To resolve an Support case in Jira

1. Log in to your **Jira Agent** view as an end user.
2. In the **Jira Service Management Jira Agent** view, choose the Jira project associated to Support.
3. Choose **Incidents** and select the Incident related to the Support case in AWS.
4. In the Jira Incident form, choose an action from **Workflow**, **Resolve**.
5. Complete the required mandatory fields.
6. Choose **Resolve**.

Fields mapped from Support case records to Jira Service Management Incident records

Status: We map Support case status values to JSM state.

JSM incident status	Support case status
OPEN	Unassigned
OPEN	Opened
WORK IN PROGRESS	Work in progress
WORK IN PROGRESS	Reopened

JSM incident status	Support case status
PENDING	Pending customer action
COMPLETED	Resolved

Priority: We map Support case severity to JSM Incident Priority

AWS severity	JSM incident priority
General Guidance	Minor
System Impaired	Low
Production System Impaired	Medium
Production system down	High
Business Critical system down	Blocker

Validating AWS Systems Manager Incident Manager integration

This section describes how to validate AWS Systems Manager Incident Manager integration in Jira.

To view Incident Manager incidents

1. Log in to your **Jira Agent** view as an end user.
2. In the **Jira Service Management Jira Agent** view, choose the Jira project associated to AWS Systems Manager Incident Manager
3. Use Jira filters to show only issues with the Issue Type **AWS Incident**

The resulting list displays all synced Incidents.

To view Incident Manager incident details

1. Log in to your **Jira Agent view** as an end user.

2. In the **Jira Service Management Jira Agent view**, choose the Jira project associated to AWS Systems Manager Incident Manager.
3. Use Jira filters to show only issues with the Issue Type **AWS Incident**.
4. Choose **Issue Id (Key)** to open the AWS Incident.
5. Review the details of the AWS Incident from the issue.
6. (Optional) Chose the AWS Incident URL to open the incident in the AWS Incident Manager console.

If AWS Systems Manager integration is enabled, an OpsItem is linked to the AWS Incident.

To resolve an Incident Manager incident

1. Log in to your **Jira Agent view** as an end user.
2. In the **Jira Service Management Jira Agent view**, choose the Jira project associated to AWS Systems Manager Incident Manager.
3. Use Jira filters to show only issues with the Issue Type **AWS Incident**.
4. Choose **Issue Id (Key)** to open the AWS Incident you want to resolve.
5. Choose **Resolve**.

Fields mapped from Incident Manager incidents to Jira issue records

This table shows how AWS Incident Manager Incidents map to a Jira issue.

AWS Incident Management Incident	Jira AWS Incident
TITLE	Summary
SUMMARY	Description
INCIDENT ARN	AWS Incident ARN
AWS ACCOUNT	AWS Account ID
AWS REGION	AWS Region
STATUS	AWS Incident Status

AWS Incident Management Incident	Jira AWS Incident
START TIME	AWS Creation Time
RESOLVED TIME	AWS Resolved Time
UPDATED TIME	AWS Last Updated Time
AWS INCIDENT URL	AWS Incident URL
IMPACT	Priority

Incident Status is an integer in Jira Service Management. Jira Service Management Connector maps Incident Manager incident status values to Jira status values.

AWS Incident Management Incident Status	Jira AWS Incident Status
Open	OPEN
Resolved	RESOLVED

Jira Service Management Connector maps **Priority - Imact** of an AWS Incident to the priority of the corresponding JIRA issue.

AWS Incident Management Incident Impact	Jira AWS Incident Priority
Critical	Blocker
High	High
Medium	Medium
Low	Low
No Impact	Minor

Validating AWS Security Hub integration

This section describes how to view AWS Security Hub Findings, update AWS Systems Manager OpsItems, and view AWS related resources in AWS Systems Manager OpsItems in Jira Service Management.

To view AWS Security Hub Findings in Jira Service Management from AWS Systems Manager

1. Log in to your **Jira Agent** view as an end user.
2. In the **Jira Service Management Jira Agent** view, choose the Jira project associated to the AWS Security Hub Finding.
3. Choose **Open Issues** and select the **AWS Security Hub Finding** from AWS that you want to view.

To update AWS Security Hub Finding in Jira Service Management

1. Log in to your **Jira Agent** view as an end user.
2. In the **Jira Service Management Jira Agent** view, choose the Jira project associated to AWS Security Hub Finding.
3. Choose **Open Issues** and select the AWS Security Hub Finding from AWS that you want to update.
4. Choose **Edit Issue**.
5. Update the fields available, such as **Severity**, **Priority**, and **Criticality**.
6. Choose **Update** to save the details.

> **Note**
>
> Updates to Security Hub Finding fields from Jira Service Management displays in the AWS account view of Findings on the next sync between AWS and Jira Service Management. Only the fields Severity, Priority, and Criticality update in the AWS account from Jira Service Management.

To view AWS related resources in AWS Security Hub Findings through Jira Service Management

1. Log in to your **Jira Agent** view as an end user.

2. In the **Jira Service Management Jira Agent** view, choose the Jira project associated to AWS Security Hub Finding.
3. Choose **Open Issues** and select the AWS Security Hub Finding.
4. In the selected AWS resources section of the AWS Security Hub Finding, you see the related resource details. If the resources relate and the AWS Config integration is active in the Connector, you can drill down on the Config resource details and relationships. The section remains empty if AWS resources do not relate in AWS Security Hub.

   AWS Security Hub findings follow the AWS Security Finding Format (ASFF). Here's a mapping of fields from AWS Security Hub findings to JSM Incident records.

JIRA issue field	Security Hub ASFF field
Created	CreatedAt
Updated	UpdatedAt
Summary	Title
Priority	Severity.Label
Status	Workflow.Status

> **Note**
>
> Jira does not duplicate findings. If a Security Hub finding is sent to Jira with the same finding ID as one previously sent to Jira, Jira updates the ticket with the most recent information in the finding.

Jira approvals and access controls

The following sections describe approvals and access controls that are available in Jira.

Approvals

The approval agent has access to a screen with the options to approve or reject the product request. For a rejection, the agent can add a comment explaining the rejection of the request. The requester is able to see the status of the request, such as *Waiting for Approval*, *Scheduled*, *Launching*, or *Available*.

Changes to approver group members do not impact approvers identified for pre-existing issues, but do affect whether we permit approval. Only approver users assigned to the issue at the time of issue creation can approve the request. The approver user must still be a member of the group to issue an approval. Otherwise, we reject the request.

As with Service Catalog, all post-provision actions, including termination, receive pre-approval for the user or group approved to provision it.

Access controls

You can set access controls on portfolios, as described earlier in this guide. Those access controls are in addition to the per-project enablement: users must have access to an AWS Connector-enabled project and belong to the groups enabled for a portfolio to provision products in that portfolio.

AWS Service Management Connector for Jira Service Management Cloud

The AWS Service Management Connector (SMC) streamlines cloud operations of AWS resources with your existing operational IT Service Management (ITSM) tooling. The AWS Service Management Connector for Atlassian's Jira Service Management Cloud enables internal customers and Jira agents to provision, manage, and operate AWS resources natively through Atlassian's Jira Service Management. Using the Connector for Jira Service Management Cloud improves the efficiency of Service Management governance and oversight for AWS resources and services.

The Connector for Jira Service Management Cloud enables role-specific tasks for Jira internal customers and Jira agents.

Jira Service Management **administrators** can

- Provide pre-approved, secured, and governed AWS resources to Jira agents and internal customers through AWS Service Catalog.
- Configure synchronization and associate Jira projects for AWS Security Hub integration.
- Configure incident resolution behavior and associate Jira projects for AWS Systems Manager Incident Manager.
- Configure synchronization and associate Jira projects for Support integration.
- Provide access to Jira agents to execute AWS Systems Manager Automation Documents.

Jira Service Management **internal customers and Jira agents** can

- Browse, request, and provide pre-secured AWS solutions.
- View, update, and resolve AWS Security Hub findings as Jira issues.
- View and resolve incidents affecting AWS-hosted applications through AWS Systems Manager Incident Manager.
- View, create, add correspondences, and resolve Support cases from Jira Service Management (including AMS Accelerate support cases).
- View and execute AWS Systems Manager Automation Documents.

These features minimize direct AWS console access and simplify AWS product requests and operational actions for Jira Service Management Cloud internal customers and Jira agents. This ensures efficient service management governance and oversight over AWS resources and services.

AWS Service Management Connector is built using Forge for Atlassian's Jira Service Management and is available at no charge in the Atlassian Marketplace. This feature is generally available in all AWS Regions where AWS Service Catalog, AWS Security Hub, AWS Systems Manager Incident Manager Support, and AWS Systems Manager Automation services are available.

The following AWS services are integrated with this Connector:

AWS Service Catalog provides a way to manage commonly deployed AWS services and provisioned software. It can help your organization establish consistent governance and compliance requirements while limiting users to deploying only approved AWS services.

AWS Security Hub provides a comprehensive view of security alerts and security posture across your AWS accounts. Security Hub provides a single location that aggregates, organizes, and prioritizes alerts (findings).

AWS Systems Manager Incident Manager helps you mitigate and recover from incidents that affect AWS applications. It improves incident resolutions by notifying responders of the impact, highlighting relevant troubleshooting data, and providing collaboration tools.

AWS Systems Manager Automation provides a way to automate common and repetitive IT operations and management tasks. You can use predefined or custom playbooks to configure AWS resources across multiple accounts and AWS Regions.

Support provides multiple tools, people, and programs to help you optimize performance, lower costs, and innovate faster. It addresses best practices, configuration details, and fixes.

AWS Health provides personalized information about events that affect your AWS infrastructure. It can also guide you through scheduled changes and help you troubleshoot issues that affect AWS resources and accounts.

AWS Systems Manager OpsCenter provides a central location for operations engineers and IT professionals to manage work items (OpsItems) related to AWS resources.

Atlassian's Jira Service Management is an IT service management tool that places developers, IT personnel, and business teams on the same platform so they can deliver services together. Jira Service Management has request types that provide self-service options and Jira agents that can deliver IT services like fulfillment approvals and workflows.

Service management alignment

This Connector aligns with industry best practices, such as ITIL service management areas, and addresses a baseline set of service management practices you can use in existing tools:

Service management area	AWS service(s) integration
Service Catalog deployment management (provisioning)	AWS Service Catalog or AWS CloudForm ation: Requesting and provisioning vetted or predictable products and performing post-provisioning actions. AWS Systems Manager Automation: Allows users to safely automate common and repetitive tasks using a predefined or custom-built automated runbacks.
Incident management (ticketing)	AWS Systems Manager Incident Manager: Generating incidents according to response plans. Support (AWS incidents, service requests, and support cases).
Security event and incident management	AWS Security Hub: Managing incidents resulting from security findings.

Pricing

The AWS Service Management Connector for Atlassian's Jira Service Management Cloud is available for no-cost download and use from your Atlassian site. You may still incur costs related to the use of AWS services integrated with the connector, and any licensing for Information Technology Service Management (ITSM) tools.

The certified version of AWS Service Management Connector is available for no-cost install from the Atlassian Marketplace.

AWS

AWS Service Management Connector for Jira Service Management Cloud uses security-approved public APIs of AWS services that support Jira Service Management Cloud integration. For pricing details, review the individual product pages of each supported AWS service below. Contact your account manager or an AWS Sales representative for more information.

- AWS Service Catalog
- AWS Security Hub
- AWS Systems Manager Incident Manager
- AWS Systems Manager Automation
- Support
- AWS Systems Manager
- AWS Health

Atlassian

An *Atlassian's Jira Service Management Cloud* license is required to use the AWS Service Management Connector app. Visit Atlassianfor more information. For licensing costs, contact your Atlassian account manager.

Prerequisites for AWS Service Management Connector for Jira Service Management Cloud

Before installing the AWS Service Management Connector for Atlassian's Jira Service Management Cloud, you must have an AWS account and an Atlassian site with Jira Service Management pre-installed. You must also verify that you have the necessary permissions in your AWS account and on the Jira Service Management website.

AWS prerequisites

To start, use the following integrations:

- **Service Catalog**

  You need an AWS account to configure your AWS portfolios and products. For details, refer to Setting up for Service Catalog.

- **AWS Security Hub**

  You must enable the service in all Regions and accounts where you want to sync Findings. For details, refer to Setting up Security Hub. We recommend you connect Jira Service Management with the primary (main) AWS account for AWS Security Hub. For more information, refer to Managing administrator and member accounts.

- **AWS Systems Manager Incident Manager**

  You must enable Incident Manager in all AWS Regions and accounts from which you want to sync incidents. For more information, refer to AWS Systems Manager Incident Manager.

- **AWS Systems Manager Automation with the Connector**

  This feature requires no setup in AWS. AWS provides a number of automation documents (runbooks). If you want additional runbooks, you can retrieve them in the Connector. For more information, see Creating your own runbooks in the *AWS Systems Manager user guide*.

- **Support with the Connector**

  Your account must have a Business or Enterprise Support plan to use support integration with the Connector.

Jira Service Management Cloud prerequisites

In addition to the AWS account, you must have an existing Jira Project, or create a new Project. The initial installation should occur in either an enterprise sandbox or a Atlassian Jira Service Management site, depending on your organization's technology governance requirements.

The Jira administrator must have the Admin role to install the Connector for Jira Service Management Cloud.

For details about Jira Service Management agent onboarding, refer to the Quick Start Guide.

Configuring baseline permissions for Jira Service Management Cloud

This section describes how to configure AWS Identity and Access Management (IAM) permissions, AWS Service Catalog, and other AWS services to use AWS Service Management Connector for Jira Service Management Cloud.

> **Note**
>
> To align with best practices, AWS recommends periodically rotating IAM user access keys. For more information, refer to Manage access keys for IAM users.

Topics

Available template for baseline permissions

For an AWS CloudFormation template to configure Jira Service Management, refer to AWS commercial Regions and AWS GovCloud (US) Regions. For each AWS account, the connector for Jira Service Management requires two IAM users:

- **AWS Sync User**: An IAM user to sync AWS resources (such as portfolios, products, Incident Manager Incidents, security Findings, and Automation Documents) to Jira.
- **AWS End User**: An IAM user who can provision products and execute automation documents as an end user. This role includes any required roles to provision and execute.

These can be the same user, and can be an existing user. Service Management Connector recommends that you assign two new users for the Connector.

> **Note**
>
> The baseline AWS CloudFormation template creates the **Sync User** and **End User** with required permissions and configures the AWS account for all available integrations.

Creating AWS Service Management Connector Sync user

This section describes how to create the AWS Sync user and associate the appropriate IAM permission. To perform this task, you must have IAM permissions to create new users. The

following steps to create a Sync user and End user are not required if you use the CloudFormation template to deploy the permissions. Refer to the AWS configurations for Connector for Jira Service Management AWS Commercial Regions and AWS GovCloud Regions.

To create AWS Service Management Connector sync user

1. Follow the instructions in Creating an IAM user in your AWS account to create a sync user (SMSyncUser). This user needs programmatic and AWS Management Console access to follow the Connector for Jira installation instructions.
2. Set permissions for your sync user (SMSyncUser). Choose **Attach existing policies directly** and select:

   `AWSServiceCatalogAdminReadOnlyAccess` (AWS managed policy)
3. Create this policy: `AWSSecurityHubPolicy`. Then follow the instructions in Creating IAM Policies, and add this code in the JSON editor:

```
{
        "Version": "2012-10-17",
        "Statement": [
            {"Action": [
                "sqs:ReceiveMessage",
                "sqs:DeleteMessage"
            ],
            "Resource": "<add sqs ARN here>",
            "Effect": "Allow"
            },
            {"Action": [
                 "securityhub:BatchUpdateFindings"
                 ],
                 "Resource": "*",
                 "Effect": "Allow"
      }
   ]
}
```

4. Create this policy: `ConfigHealthSQSBaseline`. Then follow the instructions in Creating IAM Policies, and add this code in the JSON editor:

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

```
            {"Action": [
                "sqs:ReceiveMessage",
                "sqs:DeleteMessage"
            ],
            "Resource": "<add sqs ARN here> ",
            "Effect": "Allow"
        }
    ]
}
```

5. Create this policy: `OpsCenterExecutionPolicy`. Then follow the instructions in Creating IAM Policies, and add this code in the JSON editor:

```
{
        "Version": "2012-10-17",
        "Statement": [
            {"Effect": "Allow",
            "Action": [
                "ssm:CreateOpsItem",
                "ssm:GetOpsItem",
                "ssm:UpdateOpsItem",
                "ssm:DescribeOpsItems"
             ],
            "Resource": "*"
        }
    ]
}
```

6. Create this policy: `AWSIncidentBaselinePolicy`. Then follow the instructions in Creating IAM policies, and add this code in the JSON editor:

```
{
        "Version": "2012-10-17",
        "Statement": [
            {"Action": [
                 "ssm-incidents:ListIncidentRecords",
                 "ssm-incidents:GetIncidentRecord",
                 "ssm-incidents:UpdateRelatedItems",
                 "ssm-incidents:ListTimelineEvents",
                 "ssm-incidents:GetTimelineEvent",
                 "ssm-incidents:UpdateIncidentRecord",
                 "ssm-incidents:ListRelatedItems",
                 "ssm:ListOpsItemRelatedItems",
```

```
                "ssm-incidents:ListRelatedItems",
                "ssm-incidents:ListResponsePlans",
                "ssm-incidents:StartIncident"
            ],
            "Resource": "*",
            "Effect": "Allow"
        }
    ]
}
```

7. Choose **Attach existing policies directly** and then select the following policies:
   - **AmazonSSMReadOnlyAccess** (AWS managed policy)
   - **AWSSupportAccess** (AWS managed policy)
8. Add a policy that allows `budgets:ViewBudget` on all resources (*).
9. Review and then choose **Create User**. Note the access and secret access information, and then download the .csv file containing the user credential information.

> **Note**
>
> To align with best practices, AWS recommends periodically rotating IAM user access keys. For more information, refer to Manage access keys for IAM users.

Creating AWS Service Management Connector end user

This section describes how to create the AWS Service Management Connector end user and associates the appropriate IAM permission. To perform this task, you need IAM permissions to create new users.

To create AWS Service Management Connector end user

1. Follow the instructions in Creating an IAM user in your AWS account to create a user (EndUser). The user needs programmatic and AWS Management Console access to follow the Connector for Jira installation instructions.

   For products using AWS CloudFormation StackSets, you need to create a StackSet inline policy. With AWS CloudFormation StackSets, you are able to create products across multiple accounts and Regions.

Using an administrator account, you define and manage a Service Catalog product. You also use this account to provision stacks into selected target accounts across specified Regions. You need to have the necessary permissions defined in your AWS accounts.

To set up the necessary permissions, see Granting Permissions for Stack Set Operations. Follow the instructions to create an `AWSCloudFormationStackSetAdministrationRole` and an `AWSCloudFormationStackSetExecutionRole`.

2. Add the following permissions (policies) to the user:

   - `AWSServiceCatalogEndUserFullAccess` (AWS managed policy)
   - `StackSet` (inline policy) - For Service Catalog products with stack sets, you need to modify the EndUser to include the Read Only permissions for the services you want to provision. For example, to provision an Amazon S3 bucket, include the `AmazonS3ReadOnlyAccess` policy to the `EndUser`.
   - `AmazonEC2ReadOnlyAccess` (AWS managed policy)
   - `AmazonS3ReadOnlyAccess` (AWS managed policy)

Creating SCConnectLaunch role

This section describes how to create the `SCConnectLaunch` role. This role places baseline AWS service permissions in the AWS Service Catalog launch constraints. For more information, refer to AWS Service Catalog launch constraints.

The `SCConnectLaunch` role is an IAM role that places baseline AWS service permissions into the AWS Service Catalog launch constraints. Configuring this role enables segregation of duty through provisioning product resources for Jira internal customers, Jira agents, and end users.

The `SCConnectLaunch` role baseline contains permissions to Amazon EC2 and Amazon S3 services. If your products contain additional AWS services, you must either include those services in the `SCConnectLaunch` role or create a new launch role.

To create SCConnectLaunch role

1. Create this policy: `AWSCloudFormationFullAccess` policy and then follow the instructions in Creating IAM policies. Choose **create policy** and add this code in the JSON editor:

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "cloudformation:DescribeStackResource",
                "cloudformation:DescribeStackResources",
                "cloudformation:GetTemplate",
                "cloudformation:List*",
                "cloudformation:DescribeStackEvents",
                "cloudformation:DescribeStacks",
                "cloudformation:CreateStack",
                "cloudformation:DeleteStack",
                "cloudformation:DescribeStackEvents",
                "cloudformation:DescribeStacks",
                "cloudformation:GetTemplateSummary",
                "cloudformation:SetStackPolicy",
                "cloudformation:ValidateTemplate",
                "cloudformation:UpdateStack",
                "cloudformation:CreateChangeSet",
                "cloudformation:DescribeChangeSet",
                "cloudformation:ExecuteChangeSet",
                "cloudformation:DeleteChangeSet",
                "s3:GetObject"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Note

`AWSCloudFormationFullAccess` includes additional permissions for ChangeSets.

2. Create this policy: `ServicecodeCatalogSSMActionsBaseline` policy and then follow the instructions in Creating IAM policies. Choose **create policy** and add this code in the JSON editor:

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

```
        {
            "Sid": "Stmt1536341175150",
            "Action": [
                "servicecatalog:AssociateResource",
                "servicecatalog:DisassociateResource",
                "servicecatalog:ListServiceActionsForProvisioningArtifact",
                "servicecatalog:ExecuteprovisionedProductServiceAction",
                "ssm:DescribeDocument",
                "ssm:GetAutomationExecution",
                "ssm:StartAutomationExecution",
                "ssm:StopAutomationExecution",
                "ssm:StartChangeRequestExecution",
                "cloudformation:ListStackResources",
                "ec2:DescribeInstanceStatus",
                "ec2:StartInstances",
                "ec2:StopInstances"
            ],
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "ssm.amazonaws.com"
                }
            }
        }
    ]
}
```

3. Create the SCConnectLaunch role. Then assign the trust relationship to AWS Service Catalog using this code in the JSON editor:

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

```
            "Principal": {
                "Service": "servicecatalog.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

4. Attach the relevant policies to the SCConnectLaunch role.

   Service Management Connector recommends that you customize and scope your launch policies to the specific AWS services, which are in the associated AWS CloudFormation template for the given Service Catalog product.

   For example, to provision Amazon EC2 and Amazon S3 products, the recommended policies are as follows:

   - AmazonEC2FullAccess (AWS managed policy)
   - AmazonS3FullAccess (AWS managed policy)
   - AWSCloudFormationFullAccess (custom managed policy)
   - ServiceCatalogSSMActionsBaseline (custom managed policy)

Configuring Jira Service Management Cloud

AWS Service Management Connector for Jira Service Management is based on Forge. Download the connector from Atlassian Marketplace.

After configuring IAM and AWS Service Catalog, clear browser cache, and then configure Jira Service Management. Installation tasks within Jira Service Management include the following:

Topics

Installing AWS Service Management Connector

Learn how to install Service Management Connector in the Atlassian site.

1. Log in to your Atlassian site as an administrator.
2. Navigate to the **Settings** menu, and then choose **Apps**.
3. In the **Atlassian Marketplace menu**, choose **Find new Apps**.
4. Select **AWS Service Management Connector for Jira Service Management**, and then choose **Get App**.

Configuring AWS Accounts and Regions

After installing the AWS Service Management Connector, you must configure AWS accounts and Regions in the connector.

Configure AWS accounts and Regions in the connector

1. Navigate to the **Settings** menu, and then choose **Apps**.
2. In the **Apps** menu, navigate to **AWS Service Management Connector**, and then choose **AWS accounts**.
3. Choose **Connect new account**.
4. Enter the account alias (used to identify the AWS accounts in the connector).
5. Enter the credentials for an SC-sync-user. It is the access key identity and credentials for a sync user saved from the AWS configuration. SC-sync-user credentials can retrieve portfolios and products to make them available through Jira Service Management. You can set the allowed groups that can access them.
6. Enter the credentials for a SC-end-user. It is the access key identity and credentials for the end user saved from the AWS configuration. The SC-end-user credentials provision products on behalf of a Jira user.
7. Add an **AWS Regions**. The region contains the Service Catalog products and portfolios that you also want available in Jira Service Management.
8. Choose **Test Connectivity**.
9. Upon successful connection status, choose **Connect**.

> **Note**
>
> Service Management Connector recommend the Sync user and End user be new users in AWS, used only with AWS Service Management Connector. These users should have minimum required privileges. You can use the available AWS CloudFormation templates for your sandbox and development AWS accounts to configure and enable available integrations. For more information, see Setting baseline permissions for AWS Service Management Connector for ServiceNow.

Configuring Service Catalog Portfolios in Jira

This section describes how to configure AWS Service Catalog portfolios within Jira.

AWS product access

Once your account or accounts are set up and connectivity is successful, use the **AWS Account** page to manage, for each account, which groups can access each portfolio in each Region. You can expand and collapse each Region and edit and add groups for each portfolio. Only internal customers and Jira agents in the designated groups have access to those products. By default, no groups have access.

> **Note**
>
> At least one group must be associated to a Service Catalog portfolio for Jira Service Management internal customers and Jira agents to request AWS products.

To provision products and portfolios

1. Choose **AWS Accounts**.
2. Choose **Manage** for the AWS account in which you want to configure portfolios.
3. Under **Portfolios**, expand the Region associated with the account. Portfolios display under each Region.
4. In the **Permission to request** column, choose **Add groups** for the portfolios that you want to make visible in Jira Service Management. Select the group you want to see and request Service Catalog products.

> **Note**
>
> Because the AWS Service Management Connector for Jira Service Management allows Jira internal customers and Jira agents to provision AWS products in the portfolios their groups have access to, and to control those provisioned products, internal customers and Jira agents should maintain security in their Jira accounts.

5. If products in this portfolio do not require approvals, choose **Save**.

Configuring Jira Service Management approvals for products in Service Catalog Portfolios

The AWS Service Management Connector for Jira Service Management enables administrators to configure approvals for products at the portfolio level. All products in a portfolio that contain approval permissions require approval, so AWS and Jira administrators might need to collaborate on the Service Catalog portfolio structure.

To configure the approval process

1. Choose **AWS Accounts**.
2. Choose **Manage** on the AWS account for which you want to configure portfolio approvals.
3. In the **Permission to approve** column, choose **Add groups** for the portfolios that require product approvals.
4. Select **Require approval for provisioning**.
5. Under **Permission to approve**, choose **Add group**.
6. Choose **Save**.

> **Note**
>
> If a portfolio only has a group associated with **Permissions to request**, products in the portfolio immediately provision when you submit the product request.

Viewing products and budgets

The **Available Products** tab lists the products in the portfolio and budgetary information on each. The **Budgets** tab gives overall budgetary information on the portfolio.

> **Note**
>
> Find details about additional configurations for the AWS Service Catalog request form and Automated Tags in the next section Configuring Connector Settings.

Configuring Connector Settings (Jira Project Enablement and Request Type)

In addition to configuring AWS accounts, the AWS Service Management Connector contains AWS services and UI settings for enabling and associating Jira projects and configuring integration behavior.

> **Note**
>
> There is no project-account association for AWS Service Catalog. Project-account visibility is determined by the permissions groups that are granted permission to provision.

Connector Features Enabled by Default

To configure the default Connector features for specific AWS services

For a new installation of Connector, Service Management Connector enables the default project configuration for all Connector features (AWS Service Catalog, AWS Systems Manager Incident Manager, and AWS Security Hub). If you are upgrading an existing installation, Service Management Connector does not initially enable new features.

1. In the left navigation menu, under **AWS Service Management**, select **Connector settings.**
2. At the top, under **Connector features enabled by default**, select each feature depending whether you want projects using the default configuration to be able to use them or not.
3. Choose **Save**.

UI Settings (AWS Service Catalog)

Configure the AWS Service Catalog product widget components to make them viewable to internal customers and Jira agents.

To address the varying personas of internal customers and Jira agents requesting AWS products, the Connector for Jira Service Management includes an add-on app setting to enable or disable components of the AWS product widget. By default, we enable AWS product components.

To modify the AWS product view

1. Navigate to the **Settings** menu, and then choose **Apps**.
2. Choose **AWS Service Management Connector**, and then navigate to **Connector settings**.
3. In the **UI settings** (Service Catalog) section, deselect any AWS product component such as:
    1. Allow the product name to be edited. (If unchecked, we provide an autogenerated name the user cannot edit.)
    2. Allows internal customers and Jira agents to select a launch option. (If unchecked, we select the default launch option and hide it.)
    3. Allows internal customers and Jira agents to select a product version. (If unchecked, we select the default product version and hide it.)
    4. Allows internal customers and Jira agents to add or edit tags. (If unchecked, we select the default values for tag options and hide it.)
    5. Allows internal customers and Jira agents to create a plan for creation or update of a provisioned product. (If unchecked, we hide the plans section.)
4. Choose **Save**.

Configuring AWS TagOptions for Provisioned Products

The AWS Service Management Connector enables Jira administrators to add tags (metadata) to provisioned products globally across the connector application, or granularly at the portfolio level. These tags are not visible to internal customers and Jira agents.

Two tag types are available

- Generic tags where the administrator can enter the **Key** and **Value**.

- Jira issues metadata tags where the administrator can enter the syntax for the **Key** and **Value** in the table below.

> **Note**
>
> Generic tags from administrators are not visible to internal customers or Jira agents during provisioning, but are available in the provisioned product in Service Catalog.

Key	Value
Requester name	${OPENED_BY}
Requester user name	${USERNAME}
Issue ID	${ISSUE_ID}
Project name	${PROJECT_NAME}
Project code	${PROJECT_CODE}

To add TagOptions to Service Catalog integration in Jira Service Management

1. Navigate to the **Settings** menu, and then choose **Apps**.
2. Choose **AWS Service Management Connector**, and then navigate to **Automated Tags**.
3. Enter the **Key** and **Value** fields.
4. Select a portfolio option.
    a. **Glocal** if the tag should be available in all synced portfolios, or
    b. **Portfolio** to restrict tags to only the specified portfolio.
5. Choose **Add**.

Projects Enabled for the Connector

The AWS Service Management Connector for Jira Service Management must be associated with one or more Jira projects and Jira Service Management request types. You can configure which features are enabled for each Jira project.

Configure Jira projects for AWS Service Catalog,AWS Systems Manager Incident Manager, AWS Security Hub, Support, AWS Systems Manager Automation, AWS Systems Manager OpsCenter and AWS Health

To configure the Jira projects for AWS Service Catalog, AWS Systems Manager Incident Manager, AWS Security Hub, Support, and AWS Systems Manager Automation

1. Navigate to the **Settings** menu, and then choose **Apps**.
2. Choose **AWS Service Management Connector**, and then navigate to **Connector settings**.
3. Under **Projects enabled for Connector**, you must enable at least one Jira project. You can create a new Jira Service Management project or add an existing one. Only Jira internal customers and Jira agents with access to the associated project can access the Connector. When you apply this update, the Connector adds the necessary issue types and other Jira items for AWS Service Catalog products to be available in those projects. You can return to this screen and add or remove projects at any time.
4. Projects initially take the default configuration for which Connector features are enabled. Choose **Edit** in a project row to change the configuration for individual projects. We permit projects to use more features than the default.
5. Choose **Save.**

> **Note**
>
> For internal customers and Jira agents to be able to request AWS Service Catalog products, one or more projects must be enabled. Internal customers and Jira agents must have Jira permissions to create issues in the Jira project and Permission to Request in the Jira settings for the AWS Account for at least one portfolio with products.

AWS Security Hub configuration

1. Navigate to the **Settings** menu, and then choose **Apps**.

2. Choose **AWS Service Management Connector**, and then navigate to **Connector settings**.
3. Under **Security Hub configuration**, choose **CRITICAL**, **HIGH**, **MEDIUM**, **LOW**, or **INFORMATIONAL** to configure the findings synched to Jira Service Management.

   SQS queue name is the queue from which Security Hub findings are synched. The default value is `AwsSmcJsmCloudForgeSecurityHubQueue`. The configured queue is available in all AWS accounts and regions and where you have configured the integration.
4. (optional) Enable **Recreate Jira Issues** to indicate if Jira Issues will be created for updated findings where the original Jira Issue deleted.
5. Assign onboarded AWS accounts to Jira projects.
6. Choose **Save**.

AWS Systems Manager Incident Manager configuration

1. Configure incident resolution behavior between AWS and Jira Service Management. The default value is **Bidirectional**.
2. Assign onboarded AWS accounts to Jira projects.
3. Choose **Save**.

Support configuraiton

1. Navigate to the **Settings** menu, and then choose **Apps**.
2. Choose **AWS Service Management Connector**, and then navigate to **Connector settings**.
3. In the **Support configuration** pane, choose **SQS gueue name** from where you want to sync the Support case. The default value is **AwsSmcJsmCloudForgeSupportQueue**. The queue must be available in `us-east-1` for commercial and `us-gov-west-1` for GovCloud accounts.
4. Assign onboarded AWS accounts to Jira projects.
5. Choose **Save**.

AWS Systems Manager Automation configuraiton

1. Navigate to the **Settings** menu, and then choose **Apps**.
2. Choose **AWS Service Management Connector**, and then navigate to **Connector settings**.

3. In the **AWS Systems Manager Automation configuration** pane, select the **Jira groups that can request automation** execution.
4. Choose **Save**.

AWS Systems Manager OpsCenter configuration

1. Navigate to the **Settings** menu, and then choose **Apps**.
2. Choose **AWS Service Management Connector**, and then navigate to **Connector settings**.
3. Under **Systems Manager OpsCenter configuration**, assign onboarded AWS accounts to Jira projects.
4. Choose **Save**.

AWS Health configuration

1. Navigate to the **Settings** menu, and then choose **Apps**.
2. Choose **AWS Service Management Connector**, and then navigate to **Connector settings**.
3. Under **AWS Health configuration**, provide the SQS queue name from where you want to sync health events. The default name is **AwsSmcJsmCloudForgeHealthQueue**.
4. Choose default severity levels for Jira issues for health event types (**Issue**, **Account Notification**, **Scheduled Change**).
5. Assign onboarded AWS accounts to Jira projects.
6. Choose **Save**.

Enabling the AWS Service Catalog request type in the Jira Customer Portal

The Jira Customer Portal enables registered Atlassian site internal customers and Jira agents to provision resources using the Jira Service Management (JSM) AWS Service Catalog integration. The Customer Portal does not require Jira Agent permissions.

To enable the AWS Service Catalog Request Type in Jira Customer Portal

1. Log in to your Atlassian site as an administrator.
2. For **Projects**, choose the desired Project.

3. In the navigation pane, choose **Project Settings**.
4. On the Portal Settings page, choose the **Portal groups** tab.
5. Choose **Create group**, and then enter the group name. For example, *AWS Service Catalog products*.
6. Choose **Add request form** and then select **AWS Service Catalog** from the available options.
7. Choose **Save**.

Enabling the Support request type in the Jira Customer Portal

The connector enables registered Atlassian site internal customers and Jira agents to create and manage Support cases using the Jira Service Management (JSM) Customer portal. The Customer Portal does not require Jira Agent permissions.

To enable the Support case request type in Jira Customer Portal

1. Log in to your Atlassian site as an administrator.
2. Select the desired **Project**.
3. In the navigation panel, choose **Project Settings**.
4. On the **Portal Settings** page, choose the **Portal groups** tab.
5. - To use an existing group, choose **Add request** from within the existing group, and then choose **Support Case** from the available options.
   - To create a new group, choose **Create group** and then enter the **Group name**. For example, **`Support Requests`**. Then, choose **Add request form** and choose **Support Case** from the available options.
6. (Optional) Add a **CC email address** to the case.
   a. Navigate to the **Project Settings** page and then choose the **Request types** tab.
   b. Choose the **Unassigned** tab and then choose **Support Case**.
   c. Choose the **Support Case CC Emails** field and move it from the right side panel into the **Request type** fields list.
   d. (Optional) Organize the field order as desired.
   e. Choose **Save**.

Integrating AWS Service Catalog in Jira Service Management Cloud

After you create two IAM users with baseline permissions in each account, the next step is to configure Service Catalog.

This section describes how to configure Service Catalog to have a portfolio with an Amazon S3 bucket product.

Use the Amazon S3 template in Creating an Amazon S3 Bucket for Website Hosting for your preliminary product. Copy and save the Amazon S3 template to your device.

Configuring AWS Service Catalog integration

This section provides the configurations you need to integrate AWS services in Jira Service Management Cloud.

To configure Service Catalog

1. Follow the steps to create a Service Catalog portfolio to create a portfolio.
2. To add the Amazon S3 bucket product to the portfolio you created in Step 1, go to the Service Catalog console. In the **Upload new product** page, enter the product details.
3. For **Select template**, choose the Amazon S3 bucket AWS CloudFormation template you saved to your device.
4. Set **Constraint type** to **Launch** for the product that you created now with the `SCConnectLaunch` role in the baseline permissions. For additional launch constraint instructions, see AWS Service Catalog Launch Constraints.

   > **Note**
   >
   > The AWS configuration design requires each Service Catalog product to have a launch constraint. Failure to follow this step could result in an *Unable to Retrieve Parameter* message in the ServiceNow Service Catalog.

5. Add the *SMEndUser* IAM user to the Service Catalog portfolio. For additional user access instructions, see Granting Access to Users.

Note

The AWS configuration design requires each Service Catalog product to have either a launch constraint or a stack set constraint. Failure to follow this step could result in an *Unable to Retrieve Parameter* error in the ServiceNow Service Catalog.

Creating stack set constraints

AWS CloudFormation StackSets enable users to create and deploy products across multiple accounts and Regions.

To apply a stack set constraint to a Service Catalog product

1. As a catalog admin in Service Catalog, choose the portfolio that contains the product.
2. Expand **Constraints** and choose **Add constraints**.
3. Choose the product from **Product** and set **Constraint type** to **Stack Set**. Choose **Continue**.
4. On the StackSet constraint page, enter a description.
5. Choose the account(s) in which you want to create products.
6. Choose the Region(s) in which you want to deploy products. Products deploy in these Regions in the order you specify.
7. Choose the following:

   `AWSCloudFormationStackSetAdministrationRole` to manage your target accounts.

   `AWSCloudFormationStackSetExecutionRole` for the role the Administrator will assume.
8. Choose **Submit**.

Note

The available template for baseline permissions creates the permissions as well as the outputs needed for stack set constraints.

Example stack set outputs

```
                  SCStackSetAdministratorRoleARN
                  arn:aws:iam::123456789123:role/
AWSCloudFormationStackSetAdministrationRole
                  SCIAMStackSetExecutionRoleName
                  AWSCloudFormationStackSetExecutionRole
                  SCIAMAdminRoleARN
                  arn:aws:iam::123456789123:role/
AWSCloudFormationStackSetAdministrationRole
```

The AWS Service Catalog products can have either a set set or a launch constraint, but not both.

Relating budgets to products and portfolios

The Connector for Jira Service Management enables Jira administrators to view budgets related to Service Catalog products and portfolios. Service Catalog administrators can create or associate existing budgets to products and portfolios.

For more information on creating and associating budgets, see Managing Budgets.

Validating AWS Service Catalog integration in Jira Service Management Cloud

This section describes how you can use service integration features to validate AWS Service Management Connector for Jira Service Management Cloud installation.

To order a Service Catalog product using the Jira Customer Portal

> **Note**
>
> You can only order a Service Catalog product using the Jira Customer Portal if you have enabled Jira projects for the connector and added the Service Catalog request form to the portal. For more information about the Service Catalog request form, review Enable the AWS Service Catalog Request Type in Jira Customer Portal.

1. Log in to your Jira Service Management Customer Portal.
2. Select the portal group that corresponds with the Service Catalog request form.
3. Select the product you want to provision.

4. Enter the product request details, including the **product reference name**, **parameters**, and **tags**.
5. Choose **Send** to submit the JSM request and provision the Service Catalog product.

When the product is ready to provision, users receive a notification that the product is launching.

To view provisioned products using the Jira Customer Portal

1. Log in to your Jira Service Management Customer Portal.
2. Choose **Requests** at the top right corner.
3. Select the desired provisioned product to open the issue.
4. Review the provisioned product details, including the **Status** of the product request, **Product events**, **Activities**, and any available **Self-service actions**.

To perform post-provisioning actions

1. Log in to your Jira Service Management Customer Portal.
2. Choose **Requests** at the top right corner.
3. Select a **service action** from the Self-service actions list, and then choose **Execute**.

When the product is in the `Available` status, internal customers and Jira agents can request post-provision operations, including **Request update** and **Request termination** from the **Actions** menu at the top right corner of the Issues page.

To order a Service Catalog product using the Jira Agent view

1. Log in to the Jira Service Management agent view as the internal customer or Jira agent.
2. Open the Jira project and navigate to apps **AWS Service Catalog - Order Product**.
3. Select a product to provision.
4. Fill in the product request details, including the product reference name, parameters, and tags.
5. Choose **Order** to submit the Jira Service Management request and provision the AWS Service Catalog product
6. After the request processes, a message appears indicating that your request was created. When the product is ready to provision, the internal customers or Jira agents receives a notification that the product is launching.

To view provisioned products using the Jira Agent view

1. Log in to your Jira Service Management Agent View as the internal customer or Jira agent.
2. Use Jira filters to show only issues with the Issue Type **AWS Service Catalog Request**.
3. Open a Jira issue.
4. Choose the **AWS Service Catalog** panel.
5. Review the AWS provisioned product details, including the status of the product request, product events, activities, and available Self-Service Actions.
6. If Self-Service Actions are available, you can select a service action from the list, and then choose **Execute**.
7. After the product is in the `Available` status, internal customers and Jira agents can request post-provision operations including **Request update** and **Request termination** from the **Actions** menu at the top right corner of the issue page.

Integrating AWS Security Hub in Jira Service Management Cloud

AWS Security Hub enables users to view security Findings from AWS services such as Amazon Guard Duty and Amazon Inspector, as well as AWS Partner solutions.

If you use both AWS Security Hub and Jira Service Management, the AWS Service Management Connector for Jira Service Management allows you to create an automated, bidirectional integration between Security Hub and Jira Service Management. This two-way integration synchronizes your Security Hub Findings and Jira Issues.

Specifically, as a Jira administrator, you can use this integration to automatically create Jira Issues from AWS Security Hub Findings. When you update those tickets in Jira, the changes are automatically replicated back to the original Security Hub Findings. For example, when you resolve the issue in Jira, the workflow status of the Security Hub finding also changes to `RESOLVED`. This action ensures that Security Hub always has up-to-date information about your security posture.

> **Note**
>
> If you are aggregating your Security Hub findings to a single management AWS account and have onboarded management to the connector, internal customers and Jira agents updates on the Finding issue will **not** be synched to the finding in Security Hub.

Configuring AWS Security Hub integration

This section describes how to configure your AWS services in Jira Service Management Cloud.

To configure AWS Security Hub integration features

1. Enable AWS Security Hub. For more information, refer to Setting up AWS Security Hub with the Console.
2. Set up an SQS queue to receive updated Findings. Name the queue, `AwsSmcJsmCloudForgeSecurityHubQueue`, to align with the default name in the Jira Service Management Connector Settings for the AWS Security Hub integration. For more information, refer to Getting started with Amazon SQS.
3. Set up an Amazon EventBridge rule to detect changes to Findings and push these to the queue. For more information, refer to Getting started with Amazon EventBridge.

   The CloudWatch rule should have this event pattern and point to the SQS queue created in Step 2.

   ```
   "EventPattern": {"source": [

       "aws.securityhub"

       ]
   }
   ```

4. You can also customize this CloudWatch Events rule to only pull in Security Hub Findings that have specific Finding types, severity labels, workflow statuses, or compliance statuses. For details about how to filter the event pattern, refer to Configuring an EventBridge rule for automatically sent findings in the *AWS Security Hub User Guide*.

> **Note**
>
> You can use the AWS CloudFormation templates for the Connector for Jira Service Management to automate the AWS Config custom resource and AWS Security Hub integration features. For more information, refer to Setting baseline permissions for AWS Service Management Connector for ServiceNow.

Validating AWS Security Hub integration in Jira Service Management Cloud

This section describes how to validate AWS Security Hub Findings, update AWS Systems Manager OpsItems, and view AWS related resources in Jira Service Management.

To view AWS Security Hub Findings in Jira Service Management from AWS Systems Manager

1. Log in to your **Jira Agent** view as an internal customer or Jira agent.
2. In the **Jira Service Management Jira Agent** view, choose the Jira project associated with the AWS Security Hub Finding.
3. Use Jira filters to show only issues with the Issue Type **AWS Security Hub Finding**.

To update AWS Security Hub Findings in Jira Service Management

1. Log in to your **Jira Agent** view as an internal customer or Jira agent.
2. In the **Jira Service Management Jira Agent** view, choose the Jira project associated to the AWS Security Hub Finding.
3. Use Jira filters to show only issues with the Issue Type **AWS Security Hub Finding**.
4. Choose **Edit Issue**.
5. Update the available fields, including **Severity**, **Priority**, and **Criticality**.
6. Choose **Update** to save the details.

> **Note**
>
> Updates to Security Hub Finding fields from Jira Service Management display in the AWS account view of Findings on the next sync between AWS and Jira Service Management.

Only the fields Severity, Priority, and Criticality update in the AWS account from Jira Service Management.

To view AWS related resources in AWS Security Hub Findings through Jira Service Management

1. Log in to your **Jira Agent** view as an internal customer or Jira agent.
2. In the **Jira Service Management Jira Agent** view, choose the Jira project associated to the AWS Security Hub Finding.
3. Use Jira filters to show only issues with the Issue Type **AWS Security Hub Finding**.
4. Choose the **Security Hub Findings** panel.
5. In the selected AWS resources section of the AWS Security Hub Finding, you can review the related resource details. If the resources relate and the AWS Config integration is active in the Connector, you can filter on the AWS Config-specific resource details and relationships. The section remains empty if AWS resources do not relate in AWS Security Hub. Security Hub Findings follow the AWS Security Finding format (ASFF). Review the following mapping of fields from AWS Security Hub Findings to Jira Service Management Incident records.

Jira Issue field	Security Hub ASFF field
Created	CreatedAt
Updated	UpdatedAt
Summary	Title
Priority	Severity.Label
Status	Workflow.Status

Integrating AWS Systems Manager Incident Manager

To allow the Connector to synchronize Incidents from AWS Systems Manager Incident Manager for a specific Region, you must enable Incident Manager in that account and Region.

For more information, refer to What is AWS Systems Manager Incident Manager.

Configuring AWS Systems Manager Incident Manager integration

To allow the connector to synchronize Incidents from AWS Systems Manager Incident Manager for a specific AWS Region, you must first enable Incident Manager in that AWS account and Region. For information about enabling Incident Manager, refer to What is AWS Systems Manager Incident Manager.

Validating AWS Systems Manager Incident Manager integration

This section describes how to validate AWS Systems Manager Incident Manager integration in Jira.

To view Incident Manager incidents

1. Log in to your **Jira Agent** view as a Jira agent.
2. In the **Jira Service Management Jira Agent** view, choose the Jira Project associated to AWS Systems Manager Incident Manager.
3. Use Jira filters to show only Issues with the Issue Type **AWS Incident**.

The resulting list displays all synced Incidents.

To view Incident Manager incident details

1. Log in to your **Jira Agent** view as a Jira agent.
2. In the **Jira Service Management Jira Agent** view, choose the Jira Project associated to AWS Systems Manager Incident Manager.
3. Use Jira filters to show only Issues with the Issue Type **AWS Incident**.
4. Choose **Issue ID (key)** to open the AWS Incident.
5. Review the details of the AWS Incident from the issue.
6. (Optional) Choose the AWS Incident URL to open the incident in the Incident Manager console.

If AWS Systems Manager integration is enabled, an OpsItem is linked to the AWS Incident.

To resolve an Incident Manager Incident

1. Log in to your **Jira Agent** view as a Jira agent.
2. In the **Jira Service Management Jira Agent** view, choose the Jira Project associated to AWS Systems Manager Incident Manager.

3. Use Jira filters to show only Issues with the Issue Type **AWS Incident**.
4. Choose **Issue ID (key)**to open the AWS Incident.
5. Choose **Resolve**.

Fields mapped from Incident Manager Incidents to Jira Issue records

The following table displays the mapping between Incident Manager Incidents and Jira Issues.

AWS Systems Manager Incident status	Jira AWS Issue Status
Open	OPEN
Resolved	RESOLVED

Jira Service Management Connector maps **Priority - Impact** of an AWS Incident to the priority of the corresponding Jira Issue.

AWS Systems Manager Incident Manager Incident impact	Jira AWS Issue priority
Critical	Blocker
High	High
Medium	Medium
Low	Low
No impact	Minor

Integrating Support in Jira Service Management Cloud

To allow the Connector to synchronize Support tickets, the account must have a Business or Enterprise Support plan. For more information, review Getting started with Support.

> **Note**
>
> AWS Service Management Connector allows AWS Managed Services (AMS) Accelerate users to create Incidents and Service Requests through JSM Cloud. To ensure that your account has the required permissions to create AMS Accelerate support cases, you must first onboard your account to AMS Accelerate. For more information, review Getting Started with AMS Accelerate.

Configuring Support integration

This section describes how to configure Support in Jira Service Management Cloud.

To configure Support integration features

1. Set up an Amazon SQS queue in us-east-1 for Commercial regions and AWS GovCloud (US-West) for AWS GovCloud (US) to sync Support cases.
2. Enter **`AwsSmcJsmCloudForgeSupportQueue`** for the queue name, which aligns with the default name in the JSM Cloud connector settings for the Support integration. For more information, review Getting started with Amazon SQS.
3. Create an Amazon EventBridge rule to detect changes to Support cases and push those changes to the queue. For more information, review Getting started with Amazon EventBridge.
4. The rule you created must have the following event pattern and point to the Amazon SQS queue you created in step 1:

```
"EventPattern":{
     {
        "detail-type":[
            "Support Case Update"
        ],
        "source":[
            "aws.support"
       ]
    }
}
```

Note

You can use baseline AWS CloudFormation templates for the Connector for JSM Cloud to automate the Support integration features. For more information, see Setting baseline permissions for AWS Service Management Connector for ServiceNow.
To create the required Amazon SQS queue and EventBridge rule, use Connector for JSM Cloud - AWS Support Commercial Regions and Connector for Service Management - AWS Support for GovCloud West Region.

Validating Support integration in Jira Service Management Cloud

This section describes how to create, view, and manage integration features of Support.

To view Cases from Support

1. Log in to your Jira Agent view.
2. In the Jira Service Management Jira Agent view, choose the **Jira Project** associated to Support.
3. Use the Jira filters to only view Issues with the **Support Case** Issue Type.

To create a general Support case as a Jira Incident

1. Log in to your Jira Agent view.
2. Choose **Requests** at the top right corner.
3. In the Jira Service Management Jira Agent view, choose the **Jira Project** associated to Support.
4. Choose **Create** from the list header and then select the **Support Case** Issue Type.
5. Complete the following mandatory fields in the form:
   - **Summary**— A brief summary of the question or issue
   - **Description**— A detailed summary of the question or issue
   - **Priority**— The severity of the Support case
   - **Support Service and Category**— The AWS service and category of the Support case
   - **AWS Account**— The account to create the Support case
   - (optional)**AWS CC Email Addresses**— Additional email addresses for the Support case
6. Choose **Create**.

7. Choose the Incident you created from the list. Service Management Connector displays the **AWS Case ID** and the **AWS Case Status**.

To create AMS Accelerate report incidents in Jira (for AMS Accelerate customers)

1. Log in to the Jira Agent view.
2. In the Jira Service Management Jira Agent view, choose the **Jira Project** associated to Support.
3. Choose **Create** from the list header and then select the **Support Case** Issue Type.
4. Complete the following mandatory fields in the form:

   - **Summary**— A brief summary of the question or issue
   - **Description**— A detailed summary of the question or issue
   - **Priority**— The severity of the Support case
   - **Support Service and Category**— AMS Operations - Report Incident and the chosen category
   - **AWS Account**— The account to create the Support case
   - (optional)**AWS CC Email Addresses**— Additional email addresses for the Support case
5. Choose **Create**.
6. Choose the Incident you created from the list. Service Management Connector displays the **AWS Case ID** and the **AWS Case Status**.

To create AMS Accelerate service requests in Jira (for AMS Accelerate customers)

1. Log in to the Jira Agent view.
2. In the Jira Service Management Jira Agent view, choose the **Jira Project** associated to Support.
3. Choose **Create** from the list header and then select the **Support Case** Issue Type.
4. Complete the following mandatory fields in the form:

   - **Summary**— A brief summary of the question or issue
   - **Description**— A detailed summary of the question or issue
   - **Priority**— The severity of the Support case
   - **Support Service and Category**— AMS Operations - Service Request and the chosen category
   - **AWS Account**— The account to create the Support case

- (optional)**AWS CC Email Addresses**— Additional email addresses for the Support case

5. Choose **Create**.
6. Choose the Incident you created from the list. Service Management Connector displays the **AWS Case ID** and the **AWS Case Status**.

To add a correspondence and attach it to an existing Support case in Jira incident

1. Log in to the Jira Agent view.
2. In the Jira Service Management Jira Agent view, choose the **Jira Project** associated to Support.
3. Choose **Create** from the list header and then select the **Support Case** Issue Type.
4. Open the required **Support case**.
5. At the bottom of the form, choose **Reply to customer** to add a correspondence. You can attach a maximum of three attachments, with a size limit of 5MB per attachment per correspondence.
6. Choose **Save**.

To resolve an Support case in Jira

1. Log in to the Jira Agent view.
2. In the Jira Service Management Jira Agent view, choose the **Jira Project** associated to Support.
3. Use Jira filters to display only issues with the **Support Case** Issue Type.
4. Open the required Support case.
5. Choose **Issue status** and then choose **Mark as Resolved**.

> **Note**
>
> AWS Service Management Connector also enables Jira internal customers without an Agent license to create Support cases. The validation steps above are applicable and valid for interactions with the **Support Case** issue type through the Jira customer portal.

Validating AWS Systems Manager Automation in Jira Service Management Cloud

To allow the Connector to execute Automation Documents, you must ensure that the Connector's sync user and end user have the required permissions. For more information, review Setting up Automation in the *AWS Systems Manager user guide*.

To execute a AWS Systems Manager Automation Document from Jira agent view

1. Log in to your Jira Agent view.
2. Open the desired **Jira project** and then navigate to the **AWS Service Management Connector** app.
3. Choose the **Systems Manager Automation** tab.
4. Enter the required **automation execution parameters** and add optional **Tags**.
5. Choose **Execute** to submit the Jira Service Management request and execute the automation document.

After Jira processes the request, Jira displays a message indicating that the request was created. When the automation document execution starts, you are able to view the details in the Automation panel within the Jira issue.

To view provisioned products using the Jira Agent view

1. Log in to your Jira Agent view.
2. Use Jira filters to display only issues with the **Support Automation Request** Issue Type.
3. Open the Jira issue.
4. Choose the **Automation Details** panel.

   Review the Automation Execution details, including the status of the execution, parameters, and step functions.

When the execution is complete, the issue moves to the **Execution complete** status.

AWS Systems Manager OpsCenter

To allow the Connector to synchronize AWS Systems Manager OpsCenter data for a specific Region, you must enable OpsCenter in that account and Region. For more information, refer to AWS Systems Manager OpsCenter.

Topics

Configuring AWS Systems Manager OpsCenter integration

This section describes how to configure the AWS Systems Manager OpsCenter integration in Jira Service Management. For the connector to synchronize AWS Systems Manager OpsCenter data in a specific Region, you must enable OpsCenter in that account and Region. For more information, refer to AWS Systems Manager OpsCenter.

Configuring AWS Systems Manager OpsCenter integration

This section describes how to validate the AWS Systems Manager OpsCenter integration in Jira.

> **Note**
>
> To view an AWS OpsItem, you must have access to the relevant Jira projects.

1. Log in to your Jira Agent as an internal customer or Jira agent.
2. In the Jira Service Management (Agent) view, choose the Jira project associated with the AWS OpsCenter OpsItem.
3. Use Jira filters to show only issues with type *AWS OpsCenter OpsItem*.

Validating AWS Systems Manager OpsCenter integration

This section describes how to validate the AWS Systems Manager OpsCenter integration in Jira.

Run an AWS Systems Manager automation document from an AWS OpsItems associated with a Jira incident

To view or execute automation documents (runbooks), the user must belong to the Jira permissions group assigned to the AWS Systems Manager automation integration. This group can be set on the **Connector Settings** page.

> **Note**
>
> To enable this feature, you must activate AWS Systems Manager automation in the AWS account and opt in to the connector.

1. Log in to your Jira Agent view.
2. Open your Jira project, and choose an **OpsItem** issue.
3. From the **Actions** menu at the top-right of the **Issues** page, choose **Request runbook execution**.
4. Choose your automation document.

Create a Jira Incident from AWS OpsItems

1. Log in to your Jira Agent view.
2. Open the desired Jira project, and choose an OpsItem issue.
3. From the **Actions** menu at the top-right corner of the **Issues** page, choose **Create** Incident
4. Choose a response plan, and then choose **Confirm**.

View related OpsItems or AWS Incidents from an AWS OpsItems

> **Note**
>
> There isn't a field for **RelatedOpsItems** because Jira already offers a native feature that can link Jira issues. Upon synchronization from AWS, AWS Service Management Connector looks up any Jira issues that correspond to the related OpsItems and links them. Similarly, if an end user in Jira links an issue of type *AWS OpsItem* to another issue of type *AWS OpsItem*, then AWS Service Management Connector marks the corresponding AWS OpsItems as related.

1. Log in to Jira Agent view.

2. Open your Jira project, and choose an OpsItem issue.
3. View related OpsItems at the bottom of the form.

Integrating AWS Health

This section describes how you can use AWS Health for Jira Service Management.

> **Note**
>
> To allow the connector to synchronize AWS Health events and resource information, the account should have a business or enterprise support plan. For more information, refer to What is AWS Health?

Topics

- Configuring AWS Health integration
- Validating AWS Health integration

Configuring AWS Health integration

This section describes how to configure AWS Health integration in Jira Service Management.

> **Note**
>
> To allow the connector to synchronize AWS Health data for a specific Region, you must enable AWS Health in that account and Region. For more information, refer to What is AWS Health?

Configuring AWS Health integration

1. Set up an SQS queue to sync AWS Health events. Name the queue **AwsSmcJsmCloudForgeHealthQueue** to align it with the default name in the connector settings. For more information, refer to Getting started with Amazon SQS.
2. Set up an Amazon EventBridge rule to detect changes to findings and push them to the queue. For more information, refer to Getting started with Amazon EventBridge. The rule should have the following event pattern and point to the SQS queue from step 1.

```
"EventPattern": {
    "source": ["aws.health"]
}
```

> **Note**
>
> A queue with this name must exist in all Regions defined by the AWS account that has the AWS Health integration enabled. The default value is **AwsSmcJsmCloudForgeHealthQueue**.

Configuring AWS Health

1. Navigate to the **Settings** menu, and then choose **Apps**.
2. Choose **AWS Service Management Connector**, and then navigate to **Connector** settings.
3. In the **AWS Health** section, enter the SQS queue name from where you want to sync the AWS Health. The default value is **AwsSmcJsmCloudForgeHealthQueue**. The configured queue is available in all AWS accounts and Regions where the integration is configured.
4. Assign onboarded AWS accounts to Jira projects.
5. Choose **Save**.

Validating AWS Health integration

This section describes how to validate the AWS Health integration in Jira Service Management.

Validating AWS Health integration

1. Log in to your Jira Agent view.
2. Open your Jira project, and choose a Jira issue with type *AWS Health Event*.
3. Select the **AWS Health Affected Resources** panel to view event resources.

Reference: AWS API calls for the AWS Service Management Connector

The following provides the reference AWS API calls for AWS Service Management Connector.

- AWSBudgets.describeBudget
- AWSCloudFormation.registerType
- AWSCloudFormation.deregisterType
- AWSCloudFormation.describeTypeRegistration
- AWSSecurityHub.batchUpdateFindings
- AWSSecurityTokenService.getCallerIdentity
- AWSServiceCatalog.createProvisionedProductPlan
- AWSServiceCatalog.deleteProvisionedProductPlan
- AWSServiceCatalog.describePortfolio
- AWSServiceCatalog.describeProduct
- AWSServiceCatalog.describeProductAsAdmin
- AWSServiceCatalog.describeProductView
- AWSServiceCatalog.describeProvisionedProduct
- AWSServiceCatalog.describeProvisionedProductPlan
- AWSServiceCatalog.describeProvisioningParameters
- AWSServiceCatalog.describeRecord
- AWSServiceCatalog.executeProvisionedProductPlan
- AWSServiceCatalog.executeProvisionedProductServiceAction
- AWSServiceCatalog.listBudgetsForResource
- AWSServiceCatalog.listLaunchPaths
- AWSServiceCatalog.listPortfolioAccess
- AWSServiceCatalog.listPortfolios
- AWSServiceCatalog.listProvisionedProductPlans
- AWSServiceCatalog.listServiceActionsForProvisioningArtifact
- AWSServiceCatalog.listStackInstancesForProvisionedProduct
- AWSServiceCatalog.provisionProduct
- AWSServiceCatalog.searchProducts
- AWSServiceCatalog.searchProductsAsAdmin

- `AWSServiceCatalog.terminateProvisionedProduct`
- `AWSServiceCatalog.updateProvisionedProduct`
- `AWSSimpleQueueService.DeleteMessage`
- `AWSSimpleQueueService.DeleteMessageBatch`
- `AWSSimpleQueueService.ReceiveMessage`
- `AWSSimpleSystemsManagementIncident:ListIncidentRecords`
- `AWSSimpleSystemsManagementIncident:GetIncidentRecord`
- `AWSSimpleSystemsManagementIncident:UpdateRelatedItems`
- `AWSSimpleSystemsManagementIncident:ListTimelineEvents`
- `AWSSimpleSystemsManagementIncident:GetTimelineEvent`
- `AWSSimpleSystemsManagementIncident:UpdateIncidentRecord`
- `AWSSimpleSystemsManagement:ListOpsItemRelatedItems`
- `AWSSimpleSystemsManagement:ListRelateditems`

Contacting the Service Management Connector specialist team

You can contact the AWS Service Management Connector (SMC) specialist team directly from the connector using an Support case.

To create a support case with the SMC specialist team from Support console

1. In the console, choose **Technical Support**.
2. Complete the form's required fields:
   - Service - `Service Catalog`
   - Category - `Service Management Connectors`
   - Severity - `General Guidence` or `System Impaired` (based on your need).
   - Subject - A brief summary of the question or issue; include the name of the Connector in use.
   - Description - A detailed account of the question or issue.
3. In **Console Options**, choose **Web**.
4. Choose **Submit**. A SMC specialist team member will contact you through the support case.

Jira approvals and access controls

This section describes approvals and access controls that are available in Jira.

Approvals

The approval agent has access to a screen with the options to approve or reject the product request. For a rejection, the agent can add a comment explaining the rejection of the request. The requester can view the status of the request, which includes **Waiting for Approval**, **Scheduled**, **Launching**, or **Available**. Changes to approver group members does not impact approvers identified for pre-existing issues, but does affect whether AWS permits approval. Only approver users assigned to the issue at the time of issue creation can approve the request. The approver user must also be a member of the group to issue an approval. If the approver user is not a member of the group, AWS may reject the request. All post-provision actions, including termination, receive pre-approval for the user or group approved to provision it.

Access controls

You can set access controls on portfolios, as described earlier in this guide. Those access controls are in addition to the per-project enablement: users must have access to an AWS Connector-enabled project and belong to the groups enabled for a portfolio to provision products in that portfolio.

Release notes

The AWS Service Management Connector is for Atlassian's Jira Service Management Cloud, an application based on Forge. The connector is available from Atlassian Marketplace. The latest version integrates with AWS Systems Manager OpsCenter and AWS Health.

Version 6.6.0

Core features

- Improved reliability of resource installation process.

AWS Systems Manager OpsCenter integration

- Create and update Jira issues when you create and update operational items (AWS OpsItems) in AWS Systems Manager OpsCenter.

- Update OpsItems in AWS Systems Manager OpsCenter when you update the Jira issue in Jira Service Management Cloud.
- View and run AWS Systems Manager automation runbooks to resolve OpsItems and view results of the Jira issues.
- Synchronize action-item type OpsItems from AWS Systems Manager Incident Manager.
- Creates a relationship between synced incidents from Incident Manager and the associated OpsItem.

AWS Health integration

- Creates Jira issues from AWS Health events.
- Supports affected resource tracking for planned lifecycle events.
- Supports pagination by syncing health events with visual information about the progress.
- Supports AWS Organizations to view and consolidate multiple AWS accounts through Amazon EventBridge.

Version 6.0.0

Core features

- Resolves an issue during installation that creates workflows, issue types, and other resources.
- Upgrades packages to address vulnerabilities.

AWS Systems Manager Incident Manager integration

- Resolves an intermittent issue with the Incident Manager integration that delays ticket creation.

AWS Security Hub integration enhancement

- Resolves an issue with duplicate fields that causes an *Error in Data* issue when viewing the Security Hub details panel.
- Enhances logging for Security Hub integration.

Version 5.7.0

Core features

- To avoid timeouts, the connector installation now uses the Forge Async events API.

Version 5.6.0

Core features

- Resolved site-specific issues with AWS Security Hub sync
- Improved throttling exception handing for AWS Systems Manager Automation sync
- Package dependency update

Version 5.0.0

Support integration

- Configure dual synchronization of Support cases as Jira issues
- View, create, resolve, and add correspondences to Support tickets directly from Jira issues

AWS Systems Manager Automation integration

- Render AWS Systems Manager automation documents in the Jira Service Management Agent views
- Request and execute AWS Systems Manager automation documents through Jira Service Management

Version 4.4.0

AWS Security Hub integration

- Corrected the invalid request type message to appear on the "update product issue" action only, and excluded from display on the main portal view

Version 4.2.0

AWS Security Hub integration

- Enhanced logging for AWS Security Hub integration

Core features

- Improved the connector configurations filter to allow only selection of Jira Service Desk project types

Version 4.0.0

AWS Service Catalog integration

- Render AWS Service Catalog portfolios and products in Jira Service Management using the Customer Portal view

Core features

- Implement appropriate endpoint to support AWS Service Catalog integration for China Regions into the Connector for Jira Service Management

Version 3.9.0

AWS Security Hub integration enhancement

- Additional error trapping and enhanced logging for configuration errors

Version 3.8.0

AWS Service Catalog integration

- Render AWS Service Catalog portfolios and products in Jira Service Management using the Jira Agent view
- Associate Jira Service Management approval groups to AWS Service Catalog portfolios to require approvals for Jira Service Management user product requests

- Configure AWS product request form components available for internal customers and Jira agents to view
- Create AWS Tags across provisioned products
- View AWS-specific parameters on Amazon EC2 resources, such as Availability Zones, Image ID, Instance ID, KeyPair, Security Group, and VPC

AWS Security Hub integration

- Configure synchronization behavior of AWS Security Hub Findings within Jira Service Management Cloud
- Create, view, update, investigate and resolve AWS Security Hub Findings as Jira issues

AWS Systems Manager Incident Manager integration

- Sync Incident Manager incidents as Jira Issues
- Provide configuration to allow bidirectional or unidirectional synchronization of the `resolved` status between a Jira Issue and the corresponding AWS incident

Release history

Review the release history for AWS Service Management Connector for Atlassian's Jira Service Management Cloud.

Version	Description	Date
v6.6.0	AWS Systems Manager OpsCenter integration AWS Health integration	November 6, 2023
v6.0.0	AWS Systems Manager Incident Manager integration enhancements	June 22, 2023
v5.0.0	Support integration	April 12, 2023

Version	Description	Date
	AWS Systems Manager Automation integration	
v4.2.0	AWS Security Hub enhanced logging and improved configurations filter	February 23, 2023
v4.0.0	AWS Service Catalog integration with Jira Customer Portal	February 7, 2023
v3.9.0	AWS Security Hub integration enhancements including additional logging to capture errors associated with the AWS Security Hub Findings sync	January 4, 2023
v3.8.0	AWS Service Catalog integration AWS Security Hub integration AWS Systems Manager Incident Manager integration	November 17, 2022

Document history for the AWS Service Management Connector User Guide

The following table describes the documentation releases for AWS Service Management Connector.

Change	Description	Date
First release	Initial release of the AWS Service Management Connector for Jira Service Management, Cloud version.	November 17, 2022
Initial release	Initial release of the AWS Service Management Connector Administrator Guide. It includes version 4.5.0 of AWS Service Management Connector for ServiceNow and version 1.9.0 of AWS Service Managemen t Connector for Jira Service Management.	June 9, 2022