AWS Supply Chain
Administrator Guide
# Table of Contents

What is AWS Supply Chain? ................................................................. 1  
Supported browsers ............................................................................ 1  
Supported languages .......................................................................... 1  
.............................................................................................................. 1  
Setting up an AWS account ............................................................... 2  
  Sign up for an AWS account .......................................................... 2  
  Create an administrative user ......................................................... 2  
  Closing an AWS account ............................................................... 3  
Getting started with AWS Supply Chain .......................................... 4  
  Prerequisites ................................................................................. 4  
  Using the console ........................................................................... 5  
  Creating an instance ....................................................................... 7  
  Enabling IAM Identity Center ....................................................... 9  
    Adding users in IAM Identity Center .......................................... 9  
  Choosing an AWS Supply Chain application owner .................... 9  
  Logging into AWS Supply Chain web application ....................... 10  
    Logging into AWS Supply Chain for the first time .................... 10  
  Updating your account profile ...................................................... 11  
  Updating your organization profile .............................................. 11  
  User permission roles ................................................................. 11  
    Adding users ............................................................................. 12  
    Updating user permissions ....................................................... 12  
    Deleting users ........................................................................... 12  
  Creating custom user permission roles ....................................... 13  
  Deleting an instance ..................................................................... 13  
Security ............................................................................................. 14  
Data protection ................................................................................ 14  
  Data handled by AWS Supply Chain .......................................... 15  
  Opt-out preference ...................................................................... 15  
  Encryption at rest ........................................................................ 15  
  Encryption in transit .................................................................... 15  
  How AWS Supply Chain uses grants in AWS KMS .................. 15  
IAM .................................................................................................. 16  
  Audience ..................................................................................... 16  
  Authenticating with identities ..................................................... 17  
  Managing access using policies .................................................. 19  
  How AWS Supply Chain works with IAM ................................ 20  
  Identity-based policy examples .................................................. 24  
  Troubleshooting ......................................................................... 25  
AWS managed policies ..................................................................... 26  
  AWSSupplyChainFederationAdminAccess .................................. 27  
  Policy updates ............................................................................. 28  
Compliance validation ....................................................................... 28  
Resilience ......................................................................................... 29  
Logging and Monitoring AWS Supply Chain .................................. 29  
  Web application APIs .................................................................. 30  
Quotas ............................................................................................... 33  
Administrative support .................................................................... 34  
Document history ............................................................................ 35
What is AWS Supply Chain?

AWS Supply Chain is a cloud-based supply chain management application that works with your existing solutions such as enterprise resource planning (ERP) and supply chain management systems. Using AWS Supply Chain, you can connect and extract your inventory, supply, and demand related data from existing ERP or supply chain systems into one unified AWS Supply Chain data model.

Topics
• Browsers supported by AWS Supply Chain (p. 1)
• Languages supported by AWS Supply Chain (p. 1)

Browsers supported by AWS Supply Chain

Before you work with AWS Supply Chain, verify that your browser is supported using the following table.

<table>
<thead>
<tr>
<th>Browser</th>
<th>Supported Versions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google Chrome</td>
<td>Latest three versions.</td>
</tr>
<tr>
<td>Mozilla Firefox ESR</td>
<td>Versions are supported until their Firefox end-of-life date. For details, see the Firefox ESR release calendar.</td>
</tr>
<tr>
<td>Mozilla Firefox</td>
<td>Latest three versions.</td>
</tr>
<tr>
<td>Microsoft Edge and Edge Chromium</td>
<td>Version 84 and later.</td>
</tr>
<tr>
<td>Safari</td>
<td>Safari 10 or later on macOS.</td>
</tr>
</tbody>
</table>

Languages supported by AWS Supply Chain

AWS Supply Chain supports the following languages:

• English (US)
• English (UK)
• German
• Spanish
• French
• Italian
• Portuguese
• Chinese (Simplified)
• Chinese (Traditional)
• Japanese
• Korean
• Indonesian
Setting up an AWS account

Use this section to create an AWS account and create an IAM user. For information on best practices to create an AWS account, see Establishing your best practice AWS environment.

Topics

• Sign up for an AWS account (p. 2)
• Create an administrative user (p. 2)
• Closing an AWS account (p. 3)

Sign up for an AWS account

If you do not have an AWS account, complete the following steps to create one.

To sign up for an AWS account

2. Follow the online instructions.

Part of the sign-up procedure involves receiving a phone call and entering a verification code on the phone keypad.

When you sign up for an AWS account, an AWS account root user is created. The root user has access to all AWS services and resources in the account. As a security best practice, assign administrative access to an administrative user, and use only the root user to perform tasks that require root user access.

AWS sends you a confirmation email after the sign-up process is complete. At any time, you can view your current account activity and manage your account by going to https://aws.amazon.com/ and choosing My Account.

Create an administrative user

After you sign up for an AWS account, create an administrative user so that you don't use the root user for everyday tasks.

Secure your AWS account root user

1. Sign in to the AWS Management Console as the account owner by choosing Root user and entering your AWS account email address. On the next page, enter your password.

For help signing in by using root user, see Signing in as the root user in the AWS Sign-In User Guide.

2. Turn on multi-factor authentication (MFA) for your root user.

For instructions, see Enable a virtual MFA device for your AWS account root user (console) in the IAM User Guide.
Create an administrative user

- For your daily administrative tasks, grant administrative access to an administrative user in AWS IAM Identity Center.

  For instructions, see Getting started in the AWS IAM Identity Center User Guide.

Sign in as the administrative user

- To sign in with your IAM Identity Center user, use the sign-in URL that was sent to your email address when you created the IAM Identity Center user.

  For help signing in using an IAM Identity Center user, see Signing in to the AWS access portal in the AWS Sign-In User Guide.

Closing an AWS account

For information on how to close an AWS account, see Closing an account.
Getting started with AWS Supply Chain

In this section, you can learn to create an AWS Supply Chain instance, grant user permission roles, log into the AWS Supply Chain web application, and create custom user permission roles. An AWS account can have up to 10 AWS Supply Chain instances in active or initializing state.

Topics
- Prerequisites (p. 4)
- Using the AWS Supply Chain console (p. 5)
- Creating an instance (p. 7)
- Enabling IAM Identity Center (p. 9)
- Choosing an AWS Supply Chain application owner (p. 9)
- Logging into AWS Supply Chain web application (p. 10)
- Updating your account profile (p. 11)
- Updating your organization profile (p. 11)
- User permission roles (p. 11)
- Creating custom user permission roles (p. 13)
- Deleting an instance (p. 13)

Prerequisites

Before you create an AWS Supply Chain instance, make sure that you complete the following steps:

- You've created an AWS account. For more information, see Setting up an AWS account (p. 2).

  Note
  If you haven't activated AWS IAM Identity Center, create an AWS organization and activate IAM Identity Center. For more information on creating an AWS organization, see Creating an organization.
  
  Turn on IAM Identity Center in the same AWS Region where you want to create your AWS Supply Chain instance. AWS Supply Chain is only supported in US East (N. Virginia), US West (Oregon), Europe (Frankfurt), Asia Pacific (Sydney) Region, and Europe (Ireland) Region Regions. For more information, see Enabling IAM Identity Center (p. 9).

  Note
  AWS Supply Chain Demand Planning is not supported in Europe (Ireland) Region.

  Note
  If you haven't activated IAM Identity Center in a Region other than those listed here, you can't create an AWS Supply Chain instance.

- You can create IAM users from the AWS Identity and Access Management (IAM) console. For more information, see Setting up an AWS account (p. 2).
- Add users who need access to AWS Supply Chain to IAM Identity Center. For more information, see Adding users in IAM Identity Center (p. 9). You can also connect your active directory to IAM
Identity Center. For more information, see Connect to a Microsoft AD directory in the AWS IAM Identity Center User Guide.

- When using Microsoft active directory, make sure the active directory sync is enabled.
- You need AWS Key Management Service (AWS KMS) to create an instance. AWS Supply Chain uses this AWS KMS key to encrypt all the data that comes into AWS Supply Chain.

### Using the AWS Supply Chain console

**Note**

If your AWS account is a member account of an AWS organization and includes a Service Control Policy (SCP), make sure the organization's SCP grants the following permissions to the member account. If the following permissions are not included in the organization's SCP policy, AWS Supply Chain instance creation will fail.

To access the AWS Supply Chain console, you must have a minimum set of permissions. These permissions must allow you to list and view details about the AWS Supply Chain resources in your AWS account. If you create an identity-based policy that is more restrictive than the minimum required permissions, the console won't function as intended for entities (users or roles) with that policy.

You don't need to allow minimum console permissions for users that are making calls only to the AWS CLI or the AWS API. Instead, allow access to only the actions that match the API operation that they're trying to perform.

To ensure that users and roles can still use the AWS Supply Chain console, also attach the AWS Supply Chain ConsoleAccess or ReadOnly AWS managed policy to the entities. For more information, see Adding permissions to a user in the IAM User Guide.

The following permissions are needed by the Console Admin to create and update AWS Supply Chain instances successfully.

```json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "scn:*",
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "s3:GetObject",
        "s3:PutObject",
        "s3:ListBucket",
        "s3:CreateBucket",
        "s3:PutBucketVersioning",
        "s3:PutBucketObjectLockConfiguration",
        "s3:PutEncryptionConfiguration",
        "s3:PutBucketPolicy",
        "s3:PutLifecycleConfiguration",
        "s3:PutBucketPublicAccessBlock",
        "s3:DeleteObject",
        "s3:ListAllMyBuckets",
        "s3:PutBucketOwnershipControls",
        "s3:PutBucketNotification",
        "s3:PutAccountPublicAccessBlock",
        "s3:PutBucketLogging"
      ],
      "Resource": "arn:aws:s3:::aws-supply-chain-*",
    }
  ]
}
```
"Effect": "Allow",
},
{
"Action": [
"cloudtrail:CreateTrail",
"cloudtrail:PutEventSelectors",
"cloudtrail:GetEventSelectors",
"cloudtrail:StartLogging"
],
"Resource": "*",
"Effect": "Allow"
},
{
"Action": [
"events:DescribeRule",
"events:PutRule",
"events:PutTargets"
],
"Resource": "*",
"Effect": "Allow"
},
{
"Action": [
"chime:CreateAppInstance",
"chime:DeleteAppInstance",
"chime:PutAppInstanceRetentionSettings",
"chime:TagResource"
],
"Resource": "*",
"Effect": "Allow"
},
{
"Action": [
"cloudwatch:PutMetricData",
"cloudwatch:Describe*",
"cloudwatch:Get*",
"cloudwatch:List*"
],
"Resource": "*",
"Effect": "Allow"
},
{
"Action": [
"organizations:DescribeOrganization",
"organizations:CreateOrganization",
"organizations:EnableAWSServiceAccess"
],
"Resource": "*",
"Effect": "Allow"
},
{
"Action": [
"kms:CreateGrant",
"kms:RetireGrant",
"kms:DescribeKey",
"kms:ListAliases"
],
"Resource": "*",
"Effect": "Allow"
},
{
"Action": [
"iam:CreateRole",
"iam:CreatePolicy",
"iam:GetRole",
"iam:PutRolePolicy",
"iam:PassRole"
Creating an instance

**Note**
You can create up to 10 instances within an AWS account. The 10 instances include active and initializing instances. If you’ve already activated IAM Identity Center (successor to AWS Single Sign-On), you must create your AWS Supply Chain instance in the same AWS Region where you’ve activated IAM Identity Center. AWS Supply Chain doesn’t support IAM Identity Center calls across Regions.

To create an AWS Supply Chain instance, follow these steps.

**Note**
Only the AWS Management Console administrator can create an instance. The AWS Management Console administrator who creates the AWS Supply Chain instance should have all permissions listed under Using the AWS Supply Chain console (p. 5). This administrator should invite an IAM user as a AWS Supply Chain administrator to manage AWS Supply Chain.

2. If necessary, change the AWS Region. In the bar at the top of the console window, open the Select a Region list and choose a Region. For more information about Regions, see Regions and endpoints in the IAM User Guide. Also, see Regions and endpoints in the Amazon Web Services General Reference.

**Note**
AWS Supply Chain is only supported in US East (N. Virginia), US West (Oregon), Europe (Frankfurt) Asia Pacific (Sydney) Region, and Europe (Ireland) Region. AWS Supply Chain Demand Planning is not supported in Europe (Ireland) Region.
3. On the AWS Supply Chain dashboard, choose **Create instance**.

4. On the **Instance properties** page, enter the following information:
   - **AWS Region** – Choose the Region where you have activated IAM Identity Center. To change the Region, choose **Select a Region** from the dropdown menu at the top right. You can’t change the Region after you create the instance.
   - **Name** – Enter the instance name.
   - (Optional) **Description** – Enter a description for the instance.

5. Under **AWS KMS Key**, enter your KMS key and update your KMS key policy with the following:
   - **Note**
     As an application administrator, when you add users to the AWS Supply Chain instance, they have access to the AWS KMS key. You can manage the user permissions to add or remove users. For more information on user permissions, see **User permission roles (p. 11)**.
   - **Note**
     Replace **YourAccountNumber** and **AWS-Region** with your AWS account and Region.

   ```json
   { 
   "Version": "2012-10-17",
   "Statement": [{
     "Sid": "Enable IAM User Permissions",
     "Effect": "Allow",
     "Principal": { 
       "AWS": "arn:aws:iam::YourAccountNumber:root"
     },
     "Action": "kms:*",
     "Resource": "*"
   },
   "Sid": "Allow access through SecretManager for all principals in the account that are authorized to use SecretManager",
   "Effect": "Allow",
   "Principal": { 
     "AWS": "*"
   },
   "Action": [
     "kms:Encrypt",
     "kms:Decrypt",
     "kms:ReEncrypt*",
     "kms:GenerateDataKey*",
     "kms:CreateGrant",
     "kms:DescribeKey"
   ],
   "Resource": "*",
   "Condition": {
     "StringEquals": {
      "kms:ViaService": "secretsmanager.Region.amazonaws.com",
      "kms:CallerAccount": "YourAccountNumber"
     }
   }
   }
   }
   }
   ```

If you don’t have a KMS key, choose **Create** to go to the AWS KMS console, where you can create this key. Use the previous KMS key policy. For detailed information on how to create KMS keys, see **Creating keys** in the **AWS Key Management Service Developer Guide**.
If you plan to use an S/4 Hana data connection, make sure that the KMS key that you provided has the `aws-supply-chain-access` tag with an associated Value of true.

6. (Optional) Under Instance tags, select Add new tag to assign a tag for your instance. You can use these tags to identify your instance. For information on tags, see Creating tags.

7. Choose Create instance.

It takes approximately 2 to 3 minutes for the AWS Supply Chain instance to be created. Once the instance is created, the Status field on the AWS Supply Chain dashboard shows as Active.

8. Once your AWS Supply Chain instance is created, update your KMS policy to allow AWS Supply Chain to access your AWS KMS key.

   **Note**
   Replace `YourInstanceID` with your AWS Supply Chain instance ID. You can find your instance ID on the AWS Supply Chain console dashboard.

   ```json
   {
     "Sid": "Allow AWS Supply Chain to access the AWS KMS Key",
     "Effect": "Allow",
     "Principal": {
       "AWS": "arn:aws:iam::YourAccountNumber:role/service-role/scn-instance-role-YourInstance ID"
     },
     "Action": [
       "kms:Decrypt",
       "kms:GenerateDataKey"
     ],
     "Resource": "*"
   }
   ```

---

### Enabling IAM Identity Center

Before you start to use AWS Supply Chain, you must connect to an identity source. For more information, see Getting started with IAM in the IAM User Guide.

### Adding users in IAM Identity Center

You can manage users for AWS Supply Chain using the IAM Identity Center service. IAM Identity Center is a cloud-based IAM Identity Center service that makes it convenient to manage IAM Identity Center access to all of your AWS accounts and cloud applications centrally. To add users, see Create an user in the IAM User Guide.

### Choosing an AWS Supply Chain application owner

**Note**
As an AWS console administrator, you are choosing an AWS Supply Chain application owner to manage the AWS Supply Chain web application access. The AWS Supply Chain application owner can add or remove user permission roles to the AWS Supply Chain web application.

After the instance is created and an identity source is connected, follow these steps to choose an AWS Supply Chain application owner.
1. On the AWS Supply Chain console dashboard, under **Application owner**, select **Assign application owner**.

2. Under **Select application owner**, select an user who will act as an AWS Supply Chain application owner. You can only search for the username and the users matching the search criteria appears.

   To add more users, select **Go to IAM Identity Center**. For more information on adding users, see [Adding users in IAM Identity Center](#) (p. 9) and for more information on user permission roles, see [User permission roles](#) (p. 11).

   **Note**
   You can only add one user at a time from the AWS Supply Chain Console. Adding groups is not supported in AWS Supply Chain.

3. Select **Send Invite**.

   On the AWS Supply Chain console dashboard, you will see the user listed under **Application owner**.

4. Choose **Manage in AWS Supply Chain** to add and remove users in the AWS Supply Chain web application.

---

**Logging into AWS Supply Chain web application**

As an AWS Supply Chain administrator, you should have received an email invite to the AWS Supply Chain web application.

1. You can either choose the link in the email or on the AWS Supply Chain console dashboard, under **Sub-domain**, select the **web URL**.

   The **AWS Supply Chain** web application login page appears.

2. Enter the AWS IAM Identity Center user credentials and select **Sign in**.

---

**Logging into AWS Supply Chain for the first time**

**Note**
You will only be asked to complete profiles for your account and organization when you log in for the first time.

After you log into the AWS Supply Chain web application as an AWS Supply Chain administrator, follow these steps to complete the setup.

1. On the **Complete your profile** page, enter your **Job Title** and **Time zone**. Choose **Next**.

2. On the **Let’s add your organization information** page, enter the **Organization name** and select the **Headquarters location**. Optionally, you can add a company logo. Choose **Next**.

3. On the **Set up your teammates on AWS Supply Chain** page, select the users who you want to have access to the AWS Supply Chain web application. Choose **Invite Users**. For information on how to add users to IAM Identity Center, see [Adding users in IAM Identity Center](#) (p. 9). For information on AWS Supply Chain user permission roles, see [User permission roles](#) (p. 11).

4. If you want to add users later, you can choose **Skip for now**.

   The **Onboarding complete** page appears.

5. Each user that you added receives an email message with a link that goes to AWS Supply Chain, or you can choose **Copy link** and send the link to the users.

6. Choose **Continue to homepage** to view the AWS Supply Chain dashboard.
Updating your account profile

You can update your account profile anytime on the AWS Supply Chain web application. Follow these steps to update the account.

1. On the AWS Supply Chain web application dashboard, from the left navigation pane, choose the **Settings** icon.
2. Choose **Account Profile**.
   
   The **Account Profile** page appears.
3. Update the account information, and choose **Save**.

Updating your organization profile

You can update the **Organization profile** anytime on the AWS Supply Chain web application. Follow these steps to update the organization profile.

1. On the AWS Supply Chain web application dashboard, from the left navigation pane, choose the **Settings** icon.
2. Choose **Organization**, and then choose **Organization Profile**.
   
   The **Organization Profile** page appears.
3. Update the organization **Logo** or **Headquarters location**, and then choose **Save**.

User permission roles

As an AWS Supply Chain administrator, you can either use the default user permission roles or create custom permission roles. AWS Supply Chain has the following default user permission roles:

- **Administrator** – Access to create, view, and manage all data and user permissions.
- **Data Analyst** – Access to create, view, and manage all data connections.
- **Inventory Manager** – Access to create, view, and manage Insights.
- **Planner** – Access to create, view and manage forecasts, overrides, and publish demand plans.

**Note**
As an AWS Supply Chain administrator, before you add users, note the following:

- Each default user permission role is defined with a set of permissions. You can add users to default user permission roles or create custom permission roles.
- A user can only be assigned to one user permission role.
- You cannot edit or delete default user permission roles.
- When you edit a custom permission role you created, the permissions for all the users under the custom permission role are updated.
- When you delete a custom permission role you created, all the users under the custom permission role will lose access to AWS Supply Chain.
- Adding groups is not supported in AWS Supply Chain.
Adding users

As an AWS Supply Chain administrator, you can add users to access the AWS Supply Chain web application. Follow these steps to add an user.

1. On the AWS Supply Chain dashboard, from the left navigation pane, choose the Settings icon.
2. Choose Permissions, and then choose Users.
   
   The Manage Users page appears.
3. Choose Add New User.
   
   The Add User page appears.
4. On the Add user(s) dropdown menu, select the users and choose Add.

Updating user permissions

You can update the user permission role for the current AWS Supply Chain users. Follow these steps to update the user permissions role.

1. On the AWS Supply Chain dashboard, from the left navigation pane, choose the Settings icon.
2. Choose Permissions, and then choose Users.
   
   The Manage Users page appears.
3. On the Manage Users page, select the user or group that you want to update the user permission role for, and from the Permissions Role dropdown menu, select one of the permission roles below:
   
   - **Administrator** – Access to create, view, and manage all data and user permissions.
   - **Data Analyst** – Access to create, view, and manage all data connections.
   - **Inventory Manager** – Access to create, view, and manage Insights.
   - **Planner** – Access to create, view, and manage forecasts, overrides, and publish demand plans.
4. Choose Save.

Deleting users

As an AWS Supply Chain administrator, you can delete users from the AWS Supply Chain web application. Follow these steps to delete users.

1. On the AWS Supply Chain dashboard, from the left navigation pane, choose the Settings icon.
2. Choose Permissions, and then choose Users.
   
   The Manage Users page appears.
3. On the Manage Users page, select the user that you want to delete and choose the Delete icon.
Creating custom user permission roles

In addition to default user permission roles, you can create custom user permission roles to include multiple permission roles and add specific locations and products. Follow these steps to create new permission roles.

**Note**
You can only choose the products and locations under **Location Access** and **Product Access** if your instance is connected to a data source. For example, you can create a custom Admin user just to manage avocados in the Seattle location, or an Insight user just to manage the insights for avocados in the Seattle location.

1. On the AWS Supply Chain dashboard, from the left navigation pane, choose the **Settings** icon. Choose **Permissions**, and then choose **Permission Roles**.
   
   The **Permission Roles** page appears.
2. Choose **Create New Role**.
3. On the **Manage Permission Role** page, under **Role Name**, enter a name.
4. Move the slider to select the user permission role.
   - **Manage** – Assigning users with manage permission can add, edit, and manage information.
   - **View** – Assigning users with view permission can only view the current information.
5. Under **Location Access**, search for the Regions as you type in the search bar and select the Regions.
6. Under **Product Access**, search for the products as you type in the search bar and select the products.
7. Choose **Save**.

Deleting an instance

To delete an instance use the following steps.

**Note**
When you delete an instance, information from the Amazon S3 bucket is not automatically deleted.

2. On the AWS Supply Chain console dashboard, from the dropdown, select the instance that you want to delete.
3. Choose **Delete**.
4. On the **Delete AWS Supply Chain Instance** page, under **Confirmation**, type `delete` to confirm that you want to delete the instance.
5. Choose **Delete**. The instance is deleted.
Security in AWS Supply Chain

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from data centers and network architectures that are AWS builds to meet the requirements of the most security-sensitive organizations.

Security is a shared responsibility between you and AWS. The shared responsibility model describes this as security of the cloud and security in the cloud:

- **Security of the cloud** – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. Third-party auditors regularly test and verify the effectiveness of our security as part of the AWS Compliance Programs. To learn about the compliance programs that apply to AWS Supply Chain, see AWS Services in Scope by Compliance Program.

- **Security in the cloud** – The AWS service that you use determines your responsibility. You are also responsible for other factors, including the sensitivity of your data, your requirements, and applicable laws and regulations.

This documentation helps you understand how to apply the shared responsibility model when you use AWS Supply Chain. The following topics show you how to configure AWS Supply Chain to meet your security and compliance objectives. You also learn how to use other AWS services that help you to monitor and secure your AWS Supply Chain resources.

Topics
- [Data protection in AWS Supply Chain](#) (p. 14)
- [IAM for AWS Supply Chain](#) (p. 16)
- [AWS managed policies for AWS Supply Chain](#) (p. 26)
- [Compliance validation for AWS Supply Chain](#) (p. 28)
- [Resilience in AWS Supply Chain](#) (p. 29)
- [Logging and Monitoring AWS Supply Chain](#) (p. 29)

Data protection in AWS Supply Chain

The AWS shared responsibility model applies to data protection in AWS Supply Chain. As described in this model, AWS is responsible for protecting the global infrastructure that runs all of the AWS Cloud. You are responsible for maintaining control over your content that is hosted on this infrastructure. This content includes the security configuration and management tasks for the AWS services that you use. For more information about data privacy, see the Data Privacy FAQ. For information about data protection in Europe, see the AWS Shared Responsibility Model and GDPR blog post on the AWS Security Blog.

For data protection purposes, we recommend that you protect AWS account credentials and set up individual users with AWS IAM Identity Center or AWS Identity and Access Management (IAM). That way, each user is given only the permissions necessary to fulfill their job duties. We also recommend that you secure your data in the following ways:

- Use multi-factor authentication (MFA) with each account.
- Use SSL/TLS to communicate with AWS resources. We require TLS 1.2 and recommend TLS 1.3.
- Set up API and user activity logging with AWS CloudTrail.
- Use AWS encryption solutions, along with all default security controls within AWS services.
• Use advanced managed security services such as Amazon Macie, which assists in discovering and securing sensitive data that is stored in Amazon S3.

• If you require FIPS 140-2 validated cryptographic modules when accessing AWS through a command line interface or an API, use a FIPS endpoint. For more information about the available FIPS endpoints, see Federal Information Processing Standard (FIPS) 140-2.

We strongly recommend that you never put confidential or sensitive information, such as your customers’ email addresses, into tags or free-form text fields such as a Name field. This includes when you work with AWS Supply Chain or other AWS services using the console, API, AWS CLI, or AWS SDKs. Any data that you enter into tags or free-form text fields used for names may be used for billing or diagnostic logs. If you provide a URL to an external server, we strongly recommend that you do not include credentials information in the URL to validate your request to that server.

Data handled by AWS Supply Chain

To limit the data that can be accessed by authorized users of a specific AWS Supply Chain instance, data held within AWS Supply Chain is segregated by your AWS account ID and your AWS Supply Chain instance ID.

AWS Supply Chain handles a variety of supply chain data such as, user information, information extracted from the data connector, and inventory details.

Opt-out preference

We may use and store Your Content that is processed by AWS Supply Chain, as noted in the AWS Service Terms. You may contact AWS Support to instruct AWS not to use and store Your Content processed by AWS Supply Chain to develop and improve the AWS Supply Chain service or technologies of AWS or its affiliates. For more information, see Get administrative support for AWS Supply Chain (p. 34).

Note

AWS Supply Chain is integrated with AWS Organizations, so you can opt-in or opt-out AWS Supply Chain to use or store Your Content on AWS Organizations. For more information, see AI services opt-out policy syntax and examples.

Encryption at rest

Contact data classified as PII, or data that represents customer content being stored by AWS Supply Chain, is encrypted at rest (that is, before it is put, stored, or saved to a disk) with a key that is time-limited and specific to the AWS Supply Chain instance.

Amazon S3 server-side encryption is used to encrypt all console and web application data with a AWS Key Management Service data key that is unique to each customer account. For information about AWS KMS keys, see What is AWS Key Management Service? in the AWS Key Management Service Developer Guide.

Encryption in transit

Data exchanged with AWS Supply Chain is protected in transit between the user's web browser and AWS Supply Chain using industry-standard TLS encryption.

How AWS Supply Chain uses grants in AWS KMS

AWS Supply Chain requires a grant to use your customer managed key.
AWS Supply Chain creates several grants using the AWS KMS key that is passed during the CreateInstance operation. AWS Supply Chain creates a grant on your behalf by sending CreateGrant requests to AWS KMS. Grants in AWS KMS are used to give AWS Supply Chain access to the AWS KMS key in a customer account.

Note

AWS Supply Chain uses its own authorization mechanism. Once an user is added to AWS Supply Chain, you cannot deny list the same user using the AWS KMS policy.

AWS Supply Chain uses the grant for the following:

• To send GenerateDataKey requests to AWS KMS to encrypt the data stored in your instance.
• To send Decrypt requests to AWS KMS in order to read your encrypted data associated with the instance.
• To add DescribeKey, CreateGrant, and RetireGrant permissions in order to keep your data secured when sending it to other AWS services like Amazon Forecast.

You can revoke access to the grant, or remove the service's access to the customer managed key at any time. If you do, AWS Supply Chain won't be able to access any of the data encrypted by the customer managed key, which affects operations that are dependent on that data.

IAM for AWS Supply Chain

AWS Identity and Access Management (IAM) is an AWS service that helps an administrator securely control access to AWS resources. IAM administrators control who can be authenticated (signed in) and authorized (have permissions) to use AWS Supply Chain resources. IAM is an AWS service that you can use with no additional charge.

Topics

• Audience (p. 16)
• Authenticating with identities (p. 17)
• Managing access using policies (p. 19)
• How AWS Supply Chain works with IAM (p. 20)
• Identity-based policy examples for AWS Supply Chain (p. 24)
• Troubleshooting AWS Supply Chain identity and access (p. 25)

Audience

How you use AWS Identity and Access Management (IAM) differs, depending on the work that you do in AWS Supply Chain.

Service user – If you use the AWS Supply Chain service to do your job, then your administrator provides you with the credentials and permissions that you need. As you use more AWS Supply Chain features to do your work, you might need additional permissions. Understanding how access is managed can help you request the right permissions from your administrator. If you cannot access a feature in AWS Supply Chain, see Troubleshooting AWS Supply Chain identity and access (p. 25).

Service administrator – If you're in charge of AWS Supply Chain resources at your company, you probably have full access to AWS Supply Chain. It's your job to determine which AWS Supply Chain features and resources your service users should access. You must then submit requests to your IAM administrator to change the permissions of your service users. Review the information on this page to
understand the basic concepts of IAM. To learn more about how your company can use IAM with AWS Supply Chain, see How AWS Supply Chain works with IAM (p. 20).

IAM administrator – If you’re an IAM administrator, you might want to learn details about how you can write policies to manage access to AWS Supply Chain. To view example AWS Supply Chain identity-based policies that you can use in IAM, see Identity-based policy examples for AWS Supply Chain (p. 24).

Authenticating with identities

Authentication is how you sign in to AWS using your identity credentials. You must be authenticated (signed in to AWS) as the AWS account root user, as an IAM user, or by assuming an IAM role.

You can sign in to AWS as a federated identity by using credentials provided through an identity source. AWS IAM Identity Center (IAM Identity Center) users, your company's single sign-on authentication, and your Google or Facebook credentials are examples of federated identities. When you sign in as a federated identity, your administrator previously set up identity federation using IAM roles. When you access AWS by using federation, you are indirectly assuming a role.

Depending on the type of user you are, you can sign in to the AWS Management Console or the AWS access portal. For more information about signing in to AWS, see How to sign in to your AWS account in the AWS Sign-In User Guide.

If you access AWS programmatically, AWS provides a software development kit (SDK) and a command line interface (CLI) to cryptographically sign your requests by using your credentials. If you don’t use AWS tools, you must sign requests yourself. For more information about using the recommended method to sign requests yourself, see Signing AWS API requests in the IAM User Guide.

Regardless of the authentication method that you use, you might be required to provide additional security information. For example, AWS recommends that you use multi-factor authentication (MFA) to increase the security of your account. To learn more, see Multi-factor authentication in the AWS IAM Identity Center User Guide and Using multi-factor authentication (MFA) in AWS in the IAM User Guide.

AWS account root user

When you create an AWS account, you begin with one sign-in identity that has complete access to all AWS services and resources in the account. This identity is called the AWS account root user and is accessed by signing in with the email address and password that you used to create the account. We strongly recommend that you don’t use the root user for your everyday tasks. Safeguard your root user credentials and use them to perform the tasks that only the root user can perform. For the complete list of tasks that require you to sign in as the root user, see Tasks that require root user credentials in the IAM User Guide.

Federated identity

As a best practice, require human users, including users that require administrator access, to use federation with an identity provider to access AWS services by using temporary credentials.

A federated identity is a user from your enterprise user directory, a web identity provider, the AWS Directory Service, the Identity Center directory, or any user that accesses AWS services by using credentials provided through an identity source. When federated identities access AWS accounts, they assume roles, and the roles provide temporary credentials.

For centralized access management, we recommend that you use AWS IAM Identity Center. You can create users and groups in IAM Identity Center, or you can connect and synchronize to a set of users and groups in your own identity source for use across all your AWS accounts and applications. For information about IAM Identity Center, see What is IAM Identity Center? in the AWS IAM Identity Center User Guide.
IAM users and groups

An **IAM user** is an identity within your AWS account that has specific permissions for a single person or application. Where possible, we recommend relying on temporary credentials instead of creating IAM users who have long-term credentials such as passwords and access keys. However, if you have specific use cases that require long-term credentials with IAM users, we recommend that you rotate access keys. For more information, see [Rotate access keys regularly for use cases that require long-term credentials](https://docs.aws.amazon.com/iam/latest/userguide/rotate-access-keys.html) in the IAM User Guide.

An **IAM group** is an identity that specifies a collection of IAM users. You can't sign in as a group. You can use groups to specify permissions for multiple users at a time. Groups make permissions easier to manage for large sets of users. For example, you could have a group named **IAMAdmins** and give that group permissions to administer IAM resources.

Users are different from roles. A user is uniquely associated with one person or application, but a role is intended to be assumable by anyone who needs it. Users have permanent long-term credentials, but roles provide temporary credentials. To learn more, see [When to create an IAM user (instead of a role)](https://docs.aws.amazon.com/iam/latest/userguide/when-to-create-iam-user.html) in the IAM User Guide.

IAM roles

An **IAM role** is an identity within your AWS account that has specific permissions. It is similar to an IAM user, but is not associated with a specific person. You can temporarily assume an IAM role in the AWS Management Console by **switching roles**. You can assume a role by calling an AWS CLI or AWS API operation or by using a custom URL. For more information about methods for using roles, see [Using IAM roles](https://docs.aws.amazon.com/iam/latest/userguide/using-iam-roles.html) in the IAM User Guide.

IAM roles with temporary credentials are useful in the following situations:

- **Federated user access** – To assign permissions to a federated identity, you create a role and define permissions for the role. When a federated identity authenticates, the identity is associated with the role and is granted the permissions that are defined by the role. For information about roles for federation, see [Creating a role for a third-party Identity Provider](https://docs.aws.amazon.com/iam/latest/userguide/creating-third-party-identity-provider-role.html) in the IAM User Guide. If you use IAM Identity Center, you configure a permission set. To control what your identities can access after they authenticate, IAM Identity Center correlates the permission set to a role in IAM. For information about permission sets, see [Permission sets](https://docs.aws.amazon.com/IAM/latest/UserGuide/id.PermissionSets.html) in the AWS IAM Identity Center User Guide.

- **Temporary IAM user permissions** – An IAM user or role can assume an IAM role to temporarily take on different permissions for a specific task.

- **Cross-account access** – You can use an IAM role to allow someone (a trusted principal) in a different account to access resources in your account. Roles are the primary way to grant cross-account access. However, with some AWS services, you can attach a policy directly to a resource (instead of using a role as a proxy). To learn the difference between roles and resource-based policies for cross-account access, see [How IAM roles differ from resource-based policies](https://docs.aws.amazon.com/iam/latest/userguide/switching-between-tagged-resources.html) in the IAM User Guide.

- **Cross-service access** – Some AWS services use features in other AWS services. For example, when you make a call in a service, it's common for that service to run applications in Amazon EC2 or store objects in Amazon S3. A service might do this using the calling principal's permissions, using a service role, or using a service-linked role.

- **Principal permissions** – When you use an IAM user or role to perform actions in AWS, you are considered a principal. Policies grant permissions to a principal. When you use some services, you might perform an action that then triggers another action in a different service. In this case, you must have permissions to perform both actions. To see whether an action requires additional dependent actions in a policy, see [Actions, Resources, and Condition Keys for AWS Supply Chain](https://docs.aws.amazon.com/IAM/latest/UserGuide/reference/latest/iam-policy-language-statementelements-actions-resources-condition-keys.html) in the Service Authorization Reference.

- **Service role** – A service role is an **IAM role** that a service assumes to perform actions on your behalf. An IAM administrator can create, modify, and delete a service role from within IAM. For more information, see [Creating a role to delegate permissions to an AWS service](https://docs.aws.amazon.com/iam/latest/userguide/creating-service-role.html) in the IAM User Guide.
AWS Supply Chain Administrator Guide
Managing access using policies

• **Service-linked role** – A service-linked role is a type of service role that is linked to an AWS service. The service can assume the role to perform an action on your behalf. Service-linked roles appear in your AWS account and are owned by the service. An IAM administrator can view, but not edit the permissions for service-linked roles.

• **Applications running on Amazon EC2** – You can use an IAM role to manage temporary credentials for applications that are running on an EC2 instance and making AWS CLI or AWS API requests. This is preferable to storing access keys within the EC2 instance. To assign an AWS role to an EC2 instance and make it available to all of its applications, you create an instance profile that is attached to the instance. An instance profile contains the role and enables programs that are running on the EC2 instance to get temporary credentials. For more information, see Using an IAM role to grant permissions to applications running on Amazon EC2 instances in the IAM User Guide.

To learn whether to use IAM roles or IAM users, see When to create an IAM role (instead of a user) in the IAM User Guide.

Managing access using policies

You control access in AWS by creating policies and attaching them to AWS identities or resources. A policy is an object in AWS that, when associated with an identity or resource, defines their permissions. AWS evaluates these policies when a principal (user, root user, or role session) makes a request. Permissions in the policies determine whether the request is allowed or denied. Most policies are stored in AWS as JSON documents. For more information about the structure and contents of JSON policy documents, see Overview of JSON policies in the IAM User Guide.

Administrators can use AWS JSON policies to specify who has access to what. That is, which principal can perform actions on what resources, and under what conditions.

By default, users and roles have no permissions. To grant users permission to perform actions on the resources that they need, an IAM administrator can create IAM policies. The administrator can then add the IAM policies to roles, and users can assume the roles.

IAM policies define permissions for an action regardless of the method that you use to perform the operation. For example, suppose that you have a policy that allows the `iam:GetRole` action. A user with that policy can get role information from the AWS Management Console, the AWS CLI, or the AWS API.

**Identity-based policies**

Identity-based policies are JSON permissions policy documents that you can attach to an identity, such as an IAM user, group of users, or role. These policies control what actions users and roles can perform, on which resources, and under what conditions. To learn how to create an identity-based policy, see Creating IAM policies in the IAM User Guide.

Identity-based policies can be further categorized as inline policies or managed policies. Inline policies are embedded directly into a single user, group, or role. Managed policies are standalone policies that you can attach to multiple users, groups, and roles in your AWS account. Managed policies include AWS managed policies and customer managed policies. To learn how to choose between a managed policy or an inline policy, see Choosing between managed policies and inline policies in the IAM User Guide.

**Resource-based policies**

Resource-based policies are JSON policy documents that you attach to a resource. Examples of resource-based policies are IAM role trust policies and Amazon S3 bucket policies. In services that support resource-based policies, service administrators can use them to control access to a specific resource. For the resource where the policy is attached, the policy defines what actions a specified principal can perform on that resource and under what conditions. You must specify a principal in a resource-based policy. Principals can include accounts, users, roles, federated users, or AWS services.
Resource-based policies are inline policies that are located in that service. You can’t use AWS managed policies from IAM in a resource-based policy.

**Access control lists (ACLs)**

Access control lists (ACLs) control which principals (account members, users, or roles) have permissions to access a resource. ACLs are similar to resource-based policies, although they do not use the JSON policy document format.

Amazon S3, AWS WAF, and Amazon VPC are examples of services that support ACLs. To learn more about ACLs, see [Access control list (ACL) overview](https://docs.aws.amazon.com/AmazonS3/latest/dev/UsingAccessControlLists.html) in the *Amazon Simple Storage Service Developer Guide*.

**Other policy types**

AWS supports additional, less-common policy types. These policy types can set the maximum permissions granted to you by the more common policy types.

- **Permissions boundaries** – A permissions boundary is an advanced feature in which you set the maximum permissions that an identity-based policy can grant to an IAM entity (IAM user or role). You can set a permissions boundary for an entity. The resulting permissions are the intersection of an entity’s identity-based policies and its permissions boundaries. Resource-based policies that specify the user or role in the `Principal` field are not limited by the permissions boundary. An explicit deny in any of these policies overrides the allow. For more information about permissions boundaries, see [Permissions boundaries for IAM entities](https://docs.aws.amazon.com/IAM/latest/userguide/permissions-boundaries.html) in the *IAM User Guide*.

- **Service control policies (SCPs)** – SCPs are JSON policies that specify the maximum permissions for an organization or organizational unit (OU) in AWS Organizations. AWS Organizations is a service for grouping and centrally managing multiple AWS accounts that your business owns. If you enable all features in an organization, then you can apply service control policies (SCPs) to any or all of your accounts. The SCP limits permissions for entities in member accounts, including each AWS account root user. For more information about Organizations and SCPs, see [How SCPs work](https://docs.aws.amazon.com/organizations/latest/userguide/whats-new.html#scp) in the *AWS Organizations User Guide*.

- **Session policies** – Session policies are advanced policies that you pass as a parameter when you programmatically create a temporary session for a role or federated user. The resulting session’s permissions are the intersection of the user or role’s identity-based policies and the session policies. Permissions can also come from a resource-based policy. An explicit deny in any of these policies overrides the allow. For more information, see [Session policies](https://docs.aws.amazon.com/IAM/latest/userguide/iam-getting-started-tutorials.html) in the *IAM User Guide*.

**Multiple policy types**

When multiple types of policies apply to a request, the resulting permissions are more complicated to understand. To learn how AWS determines whether to allow a request when multiple policy types are involved, see [Policy evaluation logic](https://docs.aws.amazon.com/IAM/latest/userguide/PolicyEvaluation.html) in the *IAM User Guide*.

**How AWS Supply Chain works with IAM**

Before you use IAM to manage access to AWS Supply Chain, learn what IAM features are available to use with AWS Supply Chain.

**IAM features you can use with AWS Supply Chain**

<table>
<thead>
<tr>
<th>IAM feature</th>
<th>AWS Supply Chain support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identity-based policies (p. 21)</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Identity-based policies for AWS Supply Chain

Identity-based policies are JSON permissions policy documents that you can attach to an identity, such as an IAM user, group of users, or role. These policies control what actions users and roles can perform, on which resources, and under what conditions. To learn how to create an identity-based policy, see Creating IAM policies in the IAM User Guide.

With IAM identity-based policies, you can specify allowed or denied actions and resources as well as the conditions under which actions are allowed or denied. You can’t specify the principal in an identity-based policy because it applies to the user or role to which it is attached. To learn about all of the elements that you can use in a JSON policy, see IAM JSON policy elements reference in the IAM User Guide.

Identity-based policy examples for AWS Supply Chain

To view examples of AWS Supply Chain identity-based policies, see Identity-based policy examples for AWS Supply Chain (p. 24).

Resource-based policies within AWS Supply Chain

Resource-based policies are JSON policy documents that you attach to a resource. Examples of resource-based policies are IAM role trust policies and Amazon S3 bucket policies. In services that support resource-based policies, service administrators can use them to control access to a specific resource. For the resource where the policy is attached, the policy defines what actions a specified principal can perform on that resource and under what conditions. You must specify a principal in a resource-based policy. Principals can include accounts, users, roles, federated users, or AWS services.

To enable cross-account access, you can specify an entire account or IAM entities in another account as the principal in a resource-based policy. Adding a cross-account principal to a resource-based policy is only half of establishing the trust relationship. When the principal and the resource are in different AWS
accounts, an IAM administrator in the trusted account must also grant the principal entity (user or role) permission to access the resource. They grant permission by attaching an identity-based policy to the entity. However, if a resource-based policy grants access to a principal in the same account, no additional identity-based policy is required. For more information, see [How IAM roles differ from resource-based policies in the IAM User Guide](#).

### Policy actions for AWS Supply Chain

- **Supports policy actions**: Yes

  Administrators can use AWS JSON policies to specify who has access to what. That is, which principal can perform actions on what resources, and under what conditions.

  The Action element of a JSON policy describes the actions that you can use to allow or deny access in a policy. Policy actions usually have the same name as the associated AWS API operation. There are some exceptions, such as permission-only actions that don’t have a matching API operation. There are also some operations that require multiple actions in a policy. These additional actions are called dependent actions.

  Include actions in a policy to grant permissions to perform the associated operation.

  Policy actions in AWS Supply Chain use the following prefix before the action:

  - `scn`

  To specify multiple actions in a single statement, separate them with commas.

  ```json
  "Action": [
    "scn:action1",
    "scn:action2"
  ]
  ```

  To view examples of AWS Supply Chain identity-based policies, see [Identity-based policy examples for AWS Supply Chain](#) (p. 24).

### Policy resources for AWS Supply Chain

- **Supports policy resources**: Yes

  Administrators can use AWS JSON policies to specify who has access to what. That is, which principal can perform actions on what resources, and under what conditions.

  The Resource JSON policy element specifies the object or objects to which the action applies. Statements must include either a Resource or a NotResource element. As a best practice, specify a resource using its Amazon Resource Name (ARN). You can do this for actions that support a specific resource type, known as resource-level permissions.

  For actions that don’t support resource-level permissions, such as listing operations, use a wildcard (*) to indicate that the statement applies to all resources.

  ```json
  "Resource": "*"
  ```
To view examples of AWS Supply Chain identity-based policies, see Identity-based policy examples for AWS Supply Chain (p. 24).

**Policy condition keys for AWS Supply Chain**

<table>
<thead>
<tr>
<th>Supports service-specific policy condition keys</th>
<th>Yes</th>
</tr>
</thead>
</table>

Administrators can use AWS JSON policies to specify who has access to what. That is, which principal can perform actions on what resources, and under what conditions.

The Condition element (or Condition block) lets you specify conditions in which a statement is in effect. The Condition element is optional. You can create conditional expressions that use condition operators, such as equals or less than, to match the condition in the policy with values in the request.

If you specify multiple Condition elements in a statement, or multiple keys in a single Condition element, AWS evaluates them using a logical AND operation. If you specify multiple values for a single condition key, AWS evaluates the condition using a logical OR operation. All of the conditions must be met before the statement's permissions are granted.

You can also use placeholder variables when you specify conditions. For example, you can grant an IAM user permission to access a resource only if it is tagged with their IAM user name. For more information, see IAM policy elements: variables and tags in the IAM User Guide.

AWS supports global condition keys and service-specific condition keys. To see all AWS global condition keys, see AWS global condition context keys in the IAM User Guide.

To view examples of AWS Supply Chain identity-based policies, see Identity-based policy examples for AWS Supply Chain (p. 24).

**Using temporary credentials with AWS Supply Chain**

<table>
<thead>
<tr>
<th>Supports temporary credentials</th>
<th>Yes</th>
</tr>
</thead>
</table>

Some AWS services don't work when you sign in using temporary credentials. For additional information, including which AWS services work with temporary credentials, see AWS services that work with IAM in the IAM User Guide.

You are using temporary credentials if you sign in to the AWS Management Console using any method except a user name and password. For example, when you access AWS using your company's single sign-on (SSO) link, that process automatically creates temporary credentials. You also automatically create temporary credentials when you sign in to the console as a user and then switch roles. For more information about switching roles, see Switching to a role (console) in the IAM User Guide.

You can manually create temporary credentials using the AWS CLI or AWS API. You can then use those temporary credentials to access AWS. AWS recommends that you dynamically generate temporary credentials instead of using long-term access keys. For more information, see Temporary security credentials in IAM.

**Cross-service principal permissions for AWS Supply Chain**

<table>
<thead>
<tr>
<th>Supports principal permissions</th>
<th>Yes</th>
</tr>
</thead>
</table>
When you use an IAM user or role to perform actions in AWS, you are considered a principal. Policies grant permissions to a principal. When you use some services, you might perform an action that then triggers another action in a different service. In this case, you must have permissions to perform both actions. To see whether an action requires additional dependent actions in a policy, see Actions, Resources, and Condition Keys for AWS Supply Chain in the Service Authorization Reference.

**Service roles for AWS Supply Chain**

| Supports service roles | Yes |

A service role is an IAM role that a service assumes to perform actions on your behalf. An IAM administrator can create, modify, and delete a service role from within IAM. For more information, see Creating a role to delegate permissions to an AWS service in the IAM User Guide.

**Warning**
Changing the permissions for a service role might break AWS Supply Chain functionality. Edit service roles only when AWS Supply Chain provides guidance to do so.

**Service-linked roles for AWS Supply Chain**

| Supports service-linked roles | No |

A service-linked role is a type of service role that is linked to an AWS service. The service can assume the role to perform an action on your behalf. Service-linked roles appear in your AWS account and are owned by the service. An IAM administrator can view, but not edit the permissions for service-linked roles.

For details about creating or managing service-linked roles, see AWS services that work with IAM. Find a service in the table that includes a Yes in the Service-linked role column. Choose the Yes link to view the service-linked role documentation for that service.

**Identity-based policy examples for AWS Supply Chain**

By default, users and roles don’t have permission to create or modify AWS Supply Chain resources. They also can’t perform tasks by using the AWS Management Console, AWS Command Line Interface (AWS CLI), or AWS API. To grant users permission to perform actions on the resources that they need, an IAM administrator can create IAM policies. The administrator can then add the IAM policies to roles, and users can assume the roles.

To learn how to create an IAM identity-based policy by using these example JSON policy documents, see Creating IAM policies in the IAM User Guide.

**Policy best practices**

Identity-based policies determine whether someone can create, access, or delete AWS Supply Chain resources in your account. These actions can incur costs for your AWS account. When you create or edit identity-based policies, follow these guidelines and recommendations:

- Get started with AWS managed policies and move toward least-privilege permissions – To get started granting permissions to your users and workloads, use the AWS managed policies that grant permissions for many common use cases. They are available in your AWS account. We recommend that
you reduce permissions further by defining AWS customer managed policies that are specific to your use cases. For more information, see AWS managed policies or AWS managed policies for job functions in the IAM User Guide.

- **Apply least-privilege permissions** – When you set permissions with IAM policies, grant only the permissions required to perform a task. You do this by defining the actions that can be taken on specific resources under specific conditions, also known as least-privilege permissions. For more information about using IAM to apply permissions, see Policies and permissions in IAM in the IAM User Guide.

- **Use conditions in IAM policies to further restrict access** – You can add a condition to your policies to limit access to actions and resources. For example, you can write a policy condition to specify that all requests must be sent using SSL. You can also use conditions to grant access to service actions if they are used through a specific AWS service, such as AWS CloudFormation. For more information, see IAM JSON policy elements: Condition in the IAM User Guide.

- **Use IAM Access Analyzer to validate your IAM policies to ensure secure and functional permissions** – IAM Access Analyzer validates new and existing policies so that the policies adhere to the IAM policy language (JSON) and IAM best practices. IAM Access Analyzer provides more than 100 policy checks and actionable recommendations to help you author secure and functional policies. For more information, see IAM Access Analyzer policy validation in the IAM User Guide.

- **Require multi-factor authentication (MFA)** – If you have a scenario that requires IAM users or a root user in your AWS account, turn on MFA for additional security. To require MFA when API operations are called, add MFA conditions to your policies. For more information, see Configuring MFA-protected API access in the IAM User Guide.

For more information about best practices in IAM, see Security best practices in IAM in the IAM User Guide.

**Troubleshooting AWS Supply Chain identity and access**

Use the following information to help you diagnose and fix common issues that you might encounter when working with AWS Supply Chain and IAM.

**Topics**

- I'm not authorized to perform an action in AWS Supply Chain (p. 25)
- I'm not authorized to perform iam:PassRole (p. 26)
- I want to allow people outside of my AWS account to access my AWS Supply Chain resources (p. 26)

**I'm not authorized to perform an action in AWS Supply Chain**

If the AWS Management Console that you're not authorized to perform an action, then you must contact your administrator for assistance. Your administrator is the person that provided you with your user name and password.

The following example error occurs when the mateojackson IAM user tries to use the console to view details about a fictional my-example-widget resource but doesn't have the fictional scn:GetWidget permissions.

User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform: scn:GetWidget on resource: my-example-widget
In this case, Mateo asks his administrator to update his policies to allow him to access the `my-example-widget` resource using the `scn:GetWidget` action.

**I'm not authorized to perform iam:PassRole**

If you receive an error that you're not authorized to perform the `iam:PassRole` action, your policies must be updated to allow you to pass a role to AWS Supply Chain.

Some AWS services allow you to pass an existing role to that service instead of creating a new service role or service-linked role. To do this, you must have permissions to pass the role to the service.

The following example error occurs when an IAM user named `marymajor` tries to use the console to perform an action in AWS Supply Chain. However, the action requires the service to have permissions that are granted by a service role. Mary does not have permissions to pass the role to the service.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform: iam:PassRole
```

In this case, Mary's policies must be updated to allow her to perform the `iam:PassRole` action.

If you need help, contact your AWS administrator. Your administrator is the person who provided you with your sign-in credentials.

**I want to allow people outside of my AWS account to access my AWS Supply Chain resources**

You can create a role that users in other accounts or people outside of your organization can use to access your resources. You can specify who is trusted to assume the role. For services that support resource-based policies or access control lists (ACLs), you can use those policies to grant people access to your resources.

To learn more, consult the following:

- To learn whether AWS Supply Chain supports these features, see [How AWS Supply Chain works with IAM](p. 20).
- To learn how to provide access to your resources across AWS accounts that you own, see [Providing access to an IAM user in another AWS account that you own](in the IAM User Guide).
- To learn how to provide access to your resources to third-party AWS accounts, see [Providing access to AWS accounts owned by third parties](in the IAM User Guide).
- To learn how to provide access through identity federation, see [Providing access to externally authenticated users (identity federation)](in the IAM User Guide).
- To learn the difference between using roles and resource-based policies for cross-account access, see [How IAM roles differ from resource-based policies](in the IAM User Guide).

**AWS managed policies for AWS Supply Chain**

An AWS managed policy is a standalone policy that is created and administered by AWS. AWS managed policies are designed to provide permissions for many common use cases so that you can start assigning permissions to users, groups, and roles.

Keep in mind that AWS managed policies might not grant least-privilege permissions for your specific use cases because they're available for all AWS customers to use. We recommend that you reduce permissions further by defining [customer managed policies](that are specific to your use cases).
You cannot change the permissions defined in AWS managed policies. If AWS updates the permissions defined in an AWS managed policy, the update affects all principal identities (users, groups, and roles) that the policy is attached to. AWS is most likely to update an AWS managed policy when a new AWS service is launched or new API operations become available for existing services.

For more information, see AWS managed policies in the IAM User Guide.

AWS managed policy:
AWSSupplyChainFederationAdminAccess

AWSSupplyChainFederationAdminAccess provides AWS Supply Chain federated users access to the AWS Supply Chain application, including the required permissions to perform actions within the AWS Supply Chain application. The policy provides administrative permissions over IAM Identity Center users and groups and is attached to a role created by AWS Supply Chain for you. You shouldn’t attach the AWSSupplyChainFederationAdminAccess policy to any other IAM entities.

Although this policy provides all access to AWS Supply Chain through the scn:* permissions, the AWS Supply Chain role determines your permissions. The AWS Supply Chain role only includes the required permissions, and don’t have permissions to the admin APIs.

Permissions details

This policy includes the following permissions:

- **Chime** – Provides access to create or delete users under an Amazon Chime AppInstance; Provides access to manage channel, channel members, and moderators; Provides access to send messages to channel. Chime operations are scoped to app instances tagged with "SCNInstanceCid".
- **AWS IAM Identity Center (AWS SSO)** – Provides permissions required to associate and disassociate user profiles and list profiles associated with IAM Identity Center application instance.
- **AppFlow** – Provides access to create, update, and delete connection profiles; Provides access to create, update, delete, start, and stop flows; Provides access to tag and untag flows and describe flow records.
- **Amazon S3** – Provides access to list all buckets. Provides GetBucketLocation, GetBucketPolicy, PutObject, GetObject, and ListBucket access to buckets with resource arn arn:aws:s3:::aws-supply-chain-data-*.
- **SecretsManager** – Provides access to creating secrets and updating secret policy.
- **KMS** – Provides Amazon AppFlow service the access to list keys and key alias. Provides DescribeKey, CreateGrant and ListGrants permissions to KMS keys tagged with key-value aws-supply-chain-access : true; Provides access to create secrets and update secret policy.

The permissions (kms:ListKeys, kms:ListAliases, kms:GenerateDataKey, and kms:Decrypt) are not restricted to Amazon AppFlow and these permissions can be granted to any AWS KMS Key in your account.

To view the permissions of this policy, see AWSSupplyChainFederationAdminAccess in the AWS Management Console.
AWS Supply Chain updates to AWS managed policies

The following table lists details about updates to AWS managed policies for AWS Supply Chain since this service began to track these changes. For automatic alerts about changes to this page, subscribe to the RSS feed on the AWS Supply Chain Document history page.

<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWSSupplyChainFederationAdminAccess</td>
<td>Updated policy</td>
<td>November 01, 2023</td>
</tr>
<tr>
<td></td>
<td>AWS Supply Chain updated the managed policy to allow federated users access to ListProfileAssociations operations in IAM Identity Center.</td>
<td></td>
</tr>
<tr>
<td>AWSSupplyChainFederationAdminAccess</td>
<td>Updated policy</td>
<td>September 21, 2023</td>
</tr>
<tr>
<td></td>
<td>AWS Supply Chain updated the managed policy to allow federated users access to the PutObject and GetObject operations on the dedicated S3 bucket with resource arn arn:aws:s3:::aws-supply-chain-data-*</td>
<td></td>
</tr>
<tr>
<td>AWSSupplyChainFederationAdminAccess</td>
<td>New policy</td>
<td>March 01, 2023</td>
</tr>
<tr>
<td></td>
<td>AWS Supply Chain added a new policy to allow federated users to access the AWS Supply Chain application. This includes permissions necessary to perform actions within the AWS Supply Chain application.</td>
<td></td>
</tr>
<tr>
<td>AWS Supply Chain started tracking changes</td>
<td>AWS Supply Chain started tracking changes for its AWS managed policies.</td>
<td>March 01, 2023</td>
</tr>
</tbody>
</table>

Compliance validation for AWS Supply Chain

Third-party auditors assess the security and compliance of AWS Supply Chain as part of multiple AWS compliance programs. These include SOC, PCI, FedRAMP, HIPAA, and others.

For a list of AWS services that fall within the scope of specific compliance programs, see [AWS Services in Scope by Compliance Program](#). For general information, see [AWS Compliance Programs](#).

You can download third-party audit reports with AWS Artifact. For more information, see [Downloading Reports in AWS Artifact](#).

Your compliance responsibility when you use AWS Supply Chain is determined by the sensitivity of your data, your company's compliance objectives, and applicable laws and regulations. AWS provides the following resources to help with compliance:
• **Security and Compliance Quick Start Guides** – These deployment guides discuss architectural considerations and provide steps to take when you deploy security-focused and compliance-focused baseline AWS environments.

• **Architecting for HIPAA Security and Compliance Whitepaper** – This whitepaper describes how companies can use AWS to create HIPAA-compliant applications.

• **AWS Compliance Resources** – This collection of workbooks and guides might apply to your industry and location.

• **Evaluating Resources with Rules** in the **AWS Config Developer Guide** – This guide assesses how well your resource configurations comply with internal practices, industry guidelines, and regulations.

• **AWS Security Hub** – This AWS service provides a comprehensive view of your security state within AWS to help you check your compliance with security industry standards and best practices.

---

**Resilience in AWS Supply Chain**

The AWS global infrastructure is built around AWS Regions and Availability Zones. AWS Regions provide multiple physically separated and isolated Availability Zones. These are connected with low-latency, high-throughput, and highly redundant networking. With Availability Zones, you can design and operate applications and databases that automatically fail over between zones without interruption. Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.

For more information about AWS Regions and Availability Zones, see [AWS Global Infrastructure](#).

In addition to the AWS global infrastructure, AWS Supply Chain offers several features to help support your data resiliency and backup needs.

---

**Logging and Monitoring AWS Supply Chain**

Logging and Monitoring is an important part of maintaining the reliability, availability, and performance of AWS Supply Chain and your other AWS solutions. AWS provides the AWS CloudTrail monitoring tool to watch AWS Supply Chain, report when something is wrong, and take automatic actions when appropriate.

**Note**

APIs called only from the AWS Supply Chain console are captured in AWS CloudTrail.

*AWS CloudTrail* captures API calls and related events made by or on behalf of your AWS account and delivers the log files to an Amazon S3 bucket that you specify. You can identify which users and accounts called AWS, the source IP address from which the calls were made, and when the calls occurred. You can view the AWS Supply Chain events under `scn.amazonaws.com`. For more information, see the [AWS CloudTrail User Guide](#).

**Note**

Note the following with AWS Supply Chain:

• When you invite users that don't have access to AWS Supply Chain, these users don't receive information in the notifications that they receive from the web application. Invited users receive an email notification with a link to the web application. They can only log in and view the content in the notification if they have the required user permissions.

• All users with or without user permissions to a particular Insight can view the Insights chat messages.

• As an application admin, when you are add users to the AWS Supply Chain instance, they have access to the AWS KMS key. You can manage the user permissions to add or remove users. For more information on user permissions, see [User permission roles (p. 11)](#).
AWS Supply Chain web application APIs

The APIs listed in this section are called by AWS Supply Chain applications on behalf of federated users. These APIs are not visible in the CloudTrail logs and are not captured in the Service Authorization Reference document, see AWS Supply Chain. Access to these APIs are controlled by AWS Supply Chain applications based on federated user role permissions. You shouldn't try to control access to these APIs to prevent disrupting the AWS Supply Chain applications.

User roles

The following APIs are used for managing users, user roles, user notifications, and chat messages in AWS Supply Chain.

```plaintext
scn:AddMembersToResourceBasedChat
scn:AssignGalaxyRoleToUser
scn:AssociateUser
scn:BatchGetUsers
scn:BatchMarkNotificationAsDelivered
scn:CreateRole
scn:DeleteRole
scn:DescribeChatForUser
scn:GetAccessDetailConfig
scn:GetChatPreferencesForUser
scn:GetMessagingSessionConnectionDetails
scn:GetNotificationsPreference
scn:GetOrCreateChimeUser
scn:GetOrCreateResourceBasedChat
scn:GetOrCreateUserBasedChat
scn:GetOrganizationInfo
scn:GetResourceBasedChatArn
scn:GetUserDetails
scn:GetChatMembers
scn:GetChatMessages
scn:GetChatModerators
scn:GetChats
scn:GetRoles
scn:GetUserNotifications
scn:GetUsersWithRole
scn:MarkNotificationAsDelivered
scn:MarkNotificationAsRead
scn:RemoveMemberFromResourceBasedChat
scn:RemoveUser
scn:SearchChimeUsers
scn:SearchUsers
scn:SendChatMessage
scn:SetNotificationsPreference
scn:UpdateChatPreferencesForUser
scn:UpdateChatReadMarker
scn:UpdateOrganizationInfo
scn:UpdateRole
scn:UpdateUser
```

Data lake

The following APIs are used for creating and managing data flows and connections in data lake.
Insights

The following APIs are used by the Insights application to manage filters, watchlists, and view inventory changes.

scn:AddModeratorToResourceBasedChat
scn:ComputePostRebalancedQuantities
scn:ComputePostRebalancedQuantitiesV1
scn:CreateInsightFilter
scn:CreateInsightSubscription
scn:DeleteInsightFilter
scn:DeleteInsightSubscription
scn:GetInsightLineItem
scn:GetInsightSubscription
scn:GetInstanceAttribute
scn:GetInstanceRequiredDatasetAvailabilityStatus
scn:GetKpiData
scn:GetModelEndpointStatus
scn:GetPIVForProduct
scn:GetPIVForSite
scn:GetPIVForSiteAndProduct
scn:GetPIVForSitesAndProducts
scn:GetProducts
scn:GetProductSummaryAggregates
scn:GetSites
scn:GetSiteSummaryAggregates
scn:IsUserAuthorizedForInsightLineItem
scn:ListCustomAttributeValue
scn:ListGeographiesAsGalaxyAdmin
scn:ListInsightFilters
scn:ListInsightLineItems
scn:ListInsightSubscriptions
AWS Supply Chain Demand Planning

The following APIs are used in AWS Supply Chain to create and manage forecasts, demand plans, or workbooks.
Quotas for AWS Supply Chain

Your AWS account has default quotas, formerly referred to as limits, for each AWS service. Unless otherwise noted, each quota is Region-specific. You can request increases for some quotas, but some quotas can't be increased.

To view the quotas for AWS Supply Chain, open the Service Quotas console. In the navigation pane, choose AWS services and select AWS Supply Chain.

To request a quota increase, see Requesting a Quota Increase in the Service Quotas User Guide. If the quota isn't yet available in Service Quotas, use the limit increase form.

Your AWS account has the following quotas related to AWS Supply Chain.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of instances</td>
<td>10</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td></td>
</tr>
<tr>
<td>You can create up to 10 instances within an AWS account.</td>
<td></td>
</tr>
<tr>
<td>Number of Amazon S3 buckets</td>
<td>100</td>
</tr>
</tbody>
</table>
Get administrative support for AWS Supply Chain

If you are an administrator and need to contact support for AWS Supply Chain, choose one of the following options:

- If you have an AWS Support account, go to Support Center and submit a ticket.
- Open the AWS Management Console and choose AWS Supply Chain, Support, Create case.

It's helpful to provide the following information:

- Your AWS Supply Chain instance ID/ARN.
- Your AWS Region.
- A detailed description of your issue.
# Document history for the AWS Supply Chain Administrator Guide

The following table describes the documentation releases for AWS Supply Chain.

<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Updated AWS managed policy (p. 35)</strong></td>
<td>AWS Supply Chain updated the managed policy to allow federated users access to ListProfileAssociations operations in IAM Identity Center.</td>
<td>November 1, 2023</td>
</tr>
<tr>
<td><strong>Updated AWS managed policy (p. 35)</strong></td>
<td>AWS Supply Chain updated the managed policy to allow federated users access to the PutObject and GetObject operations on the dedicated S3 bucket with resource arn arn:aws:s3:::aws-supply-chain-data-*.</td>
<td>September 21, 2023</td>
</tr>
<tr>
<td><strong>Updated information on regions support (p. 35)</strong></td>
<td>AWS Supply Chain Demand Planning is now also supported in Asia Pacific (Sydney) Region.</td>
<td>September 12, 2023</td>
</tr>
<tr>
<td><strong>Use AWS Console to opt-in and opt-out AWS Supply Chain (p. 35)</strong></td>
<td>AWS Supply Chain users can now use the AWS Console to opt-in and opt-out AWS Supply Chain to use or store Your Content on AWS Organizations.</td>
<td>September 7, 2023</td>
</tr>
<tr>
<td><strong>Updated information on regions support (p. 35)</strong></td>
<td>AWS Supply Chain is now also supported in Asia Pacific (Sydney) Region, and Europe (Ireland) Region.</td>
<td>July 19, 2023</td>
</tr>
<tr>
<td><strong>Updated information on how to contact AWS Support and create an instance (p. 35)</strong></td>
<td>AWS Supply Chain users can now contact AWS Support for help and updated the content on how to create an instance.</td>
<td>April 3, 2023</td>
</tr>
<tr>
<td><strong>Added AWS managed policy (p. 35)</strong></td>
<td>AWS Supply Chain added a new policy to allow federated users access to the AWS Supply Chain application, including the permissions necessary to perform actions within the AWS Supply Chain application.</td>
<td>March 1, 2023</td>
</tr>
<tr>
<td><strong>Initial release (p. 35)</strong></td>
<td>Initial release of the AWS Supply Chain Administrator Guide.</td>
<td>November 29, 2022</td>
</tr>
</tbody>
</table>