AWS IAM Identity Center (successor to AWS Single Sign-On)

User Guide

AWS IAM Identity Center (successor to AWS Single Sign-On): User Guide

Copyright © 2022 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Table of Contents

What is IAM Identity Center?

AWS IAM Identity Center (successor to AWS Single Sign-On) expands the capabilities of AWS Identity and Access Management (IAM) to provide a central place that brings together administration of users and their access to AWS accounts and cloud applications. Although the service name AWS Single Sign-On has been retired, the term *single sign-on* is still used throughout this guide to describe the authentication scheme that allows users to sign in one time to access multiple applications and websites.

With IAM Identity Center you can manage sign-in security for your workforce by creating or connecting your users and groups to AWS in one place. With *multi-account permissions* you can assign your *workforce identities* access to AWS accounts. You can use *application assignments* to assign your users access to software as a service (SaaS) applications. With a single click, IAM Identity Center enabled application admins can assign access to your workforce users, and can also use application assignments to assign your users access to software as a service (SaaS) applications.

IAM Identity Center features

IAM Identity Center includes the following core features:

Workforce identities

Human users that are members of your organization are also known as workforce identities or *workforce users*. You can create workforce users and groups in IAM Identity Center, or connect and synchronize to an existing set of users and groups in your own identity source for use across all your AWS accounts and applications. Supported identity sources include Microsoft Active Directory Domain Services, and external identity providers such as Okta Universal Directory or Microsoft Azure AD.

Application assignments for SAML applications

With application assignments, you can grant your workforce users in IAM Identity Center single sign-on access to SAML 2.0 applications, such as Salesforce and Microsoft 365. Your users can access these applications in a single place, without the need for you to set up separate federation.

Identity Center enabled applications

AWS applications and services, such as Amazon Managed Grafana, Amazon Monitron, and Amazon SageMaker Studio Notebooks, discover and connect to IAM Identity Center automatically to receive sign-in and user directory services. This provides users with a consistent single sign-on experience to these applications with no additional configuration of the applications. Because the applications share a common view of users, groups, and group membership, users also have a consistent experience when sharing application resources with others.

Multi-account permissions

With multi-account permissions you can plan for and centrally implement IAM permissions across multiple AWS accounts at one time without needing to configure each of your accounts manually. You can create fine-grained permissions based on common job functions or define custom permissions that meet your security needs. You can then assign those permissions to workforce users to control their access over specific accounts.

AWS access portal

The AWS access portal provides your workforce users with one-click access to all their assigned AWS accounts and cloud applications through a simple web portal.

IAM Identity Center rename

On July 26, 2022, AWS Single Sign-On was renamed to AWS IAM Identity Center (successor to AWS Single Sign-On). For existing customers, the following table is meant to describe some of the more common term changes that have been updated throughout this guide as a result of the rename.

Legacy term	Current term
AWS SSO user *or* SSO user	workforce user *or* user
AWS SSO user portal *or* user portal	AWS access portal
AWS SSO-integrated applications	Identity Center enabled applications
AWS SSO directory	Identity Center directory
AWS SSO store or AWS SSO identity store	identity store used by IAM Identity Center

The following table describes the applicable user, developer and API reference guide name changes that also took place as a result of this rename.

Legacy guide	Current guide
AWS Single Sign-On User Guide	IAM Identity Center User Guide
AWS Single Sign-On SCIM Implementation Developer Guide	IAM Identity Center SCIM Implementation Developer Guide
AWS Single Sign-On API Reference Guide	IAM Identity Center API Reference
AWS Single Sign-On Identity Store API Reference Guide	Identity Store API Reference
AWS Single Sign-On OIDC API Reference Guide	IAM Identity Center OIDC API Reference
AWS Single Sign-On Portal API Reference Guide	IAM Identity Center Portal API Reference

Legacy namespaces remain the same

The `sso` and `identitystore` API namespaces along with the following related namespaces **remain unchanged** for backward compatibility purposes.

- CLI commands
  - `aws configure sso`
  - `identitystore`
  - `sso`
  - `sso-admin`
  - `sso-oidc`
- Managed policies containing `AWSSSO` and `AWSIdentitySync` prefixes
- Service endpoints containing `sso` and `identitystore`
- AWS CloudFormation resources containing `AWS::SSO` prefixes

- Service-linked role containing `AWSServiceRoleForSSO`
- Console URLs containing `sso` and `singlesignon`
- Documentation URLs containing `singlesignon`

Getting started

The topics in this section will help you get started with IAM Identity Center.

Topics

Prerequisites and considerations

The following topics provide information about prerequisites and other considerations for setting up IAM Identity Center.

Topics

Are you new to AWS?

If you don't have an AWS account, sign up for one. After you create an AWS account, proceed to Step 1: Enable IAM Identity Center (p. 6).

Prerequisites and considerations for specific environments

The following topics provide guidance for setting up IAM Identity Center for specific environments. Review the guidance that applies to your environment before you proceed to Step 1: Enable IAM Identity Center (p. 6).

Topics

Active Directory or an external IdP

If you're already managing users and groups in Active Directory or an external IdP, we recommend that you consider connecting this identity source when you enable IAM Identity Center and choose

your identity source. Doing this before you create any users and groups in the default Identity Center directory will help you avoid the additional configuration that is required if you change your identity source later.

If you want to use Active Directory as your identity source, your configuration must meet the following prerequisites:

- You must have an existing AD Connector or AWS Managed Microsoft AD directory set up in AWS Directory Service, and it must reside within your AWS Organizations management account. You can connect only one AD Connector directory or one directory in AWS Managed Microsoft AD at a time. If you need to support multiple domains or forests, use AWS Managed Microsoft AD. For more information, see:
  - Connect a directory in AWS Managed Microsoft AD to IAM Identity Center (p. 33)
  - Connect a self-managed directory in Active Directory to IAM Identity Center (p. 34)
- If you're using AWS Managed Microsoft AD, you must enable IAM Identity Center in the same AWS Region where your AWS Managed Microsoft AD directory is set up. IAM Identity Center stores the assignment data in the same Region as the directory. To administer IAM Identity Center, you might need to switch to the Region where IAM Identity Center is configured. Also, note that the AWS access portal uses the same access URL as your directory.

AWS Organizations

Your AWS account must be managed by AWS Organizations. If you haven't set up an organization, you don't have to. When you enable IAM Identity Center, you will choose whether to have AWS create an organization for you.

If you've already set up AWS Organizations, make sure that all features are enabled. For more information, see Enabling All Features in Your Organization in the *AWS Organizations User Guide*.

To enable IAM Identity Center, you must sign in to the AWS Management Console by using the credentials of your AWS Organizations management account. You can't enable IAM Identity Center while signed in with credentials from an AWS Organizations member account. For more information, see Creating and Managing an AWS Organization in the *AWS Organizations User Guide*.

IAM roles

If you've already configured IAM roles in your AWS account, we recommend that you check whether your account is approaching the quota for IAM roles. For more information, see IAM object quotas.

If you're nearing the quota, consider requesting a quota increase. Otherwise, you might experience problems with IAM Identity Center when you provision permission sets to accounts that have exceeded the IAM role quota. For information about how to request a quota increase, see Requesting a Quota Increase in the *Service Quotas User Guide*.

Next-generation firewalls and secure web gateways

If you filter access to specific AWS domains or URL endpoints by using a web content filtering solution such as NGFWs or SWGs, you must add the following domains or URL endpoints to your web-content filtering solution allow-lists. Doing so enables IAM Identity Center to function correctly.

Specific DNS domains

- *.awsapps.com (http://awsapps.com/)
- *.signin.aws

Specific URL endpoints

- https://*[yourdirectory]*.awsapps.com/start
- https://*[yourdirectory]*.awsapps.com/login
- https://*[yourregion]*.signin.aws/platform/login

Step 1: Enable IAM Identity Center

To perform the following steps, you'll sign in to the AWS Management Console as the AWS account root user. To enhance the security of your root user credentials, make sure that multi-factor authentication (MFA) is activated for your root user before you proceed. For more information, see Activate MFA on the AWS account root user in the *AWS Account Management Reference Guide*.

To enable IAM Identity Center

1. Sign in to the AWS Management Console with your AWS account root user credentials. For more information, see Sign in as the root user.
2. Open the IAM Identity Center console.
3. Under **Enable IAM Identity Center**, choose **Enable**.
4. IAM Identity Center requires AWS Organizations. If you haven't set up an organization, you must choose whether to have AWS create one for you. Choose **Create AWS organization** to complete this process.

   AWS Organizations automatically sends a verification email to the address that is associated with your management account. There might be a delay before you receive the verification email. Verify your email address within 24 hours.

   Note
   If you are using a multi-account environment, we recommend that you configure delegated administration. With delegated administration, you can limit the number of people who require access to the management account in AWS Organizations. For more information, see Delegated administration (p. 91).

Step 2: Choose your identity source

Your identity source in IAM Identity Center defines where your users and groups are managed. You can choose one of the following as your identity source:

- **Identity Center directory –** When you enable IAM Identity Center for the first time, it is automatically configured with an Identity Center directory as your default identity source. This is where you create your users and groups, and assign their level of access to your AWS accounts and applications.
- **Active Directory –** Choose this option if you want to continue managing users in either your AWS Managed Microsoft AD directory using AWS Directory Service or your self-managed directory in Active Directory (AD).
- **External identity provider –** Choose this option if you want to manage users in an external identity provider (IdP) such as Okta or Azure Active Directory.

After you enable IAM Identity Center, you must choose your identity source. The identity source that you choose determines where IAM Identity Center searches for users and groups that need single sign-on access. After you choose your identity source, you'll create or specify a user and assign them administrative permissions to your AWS account.

Important
If you're already managing users and groups in Active Directory or an external identity provider (IdP), we recommend that you consider connecting this identity source when you enable IAM Identity Center and choose your identity source. This should be done before you create any users and groups in the default Identity Center directory and make any assignments. If you're already managing users and groups in one identity source in IAM Identity Center, changing to a different identity source might remove all user and group assignments that you configured in IAM Identity Center. If this occurs, all users, including the administrative user in IAM Identity Center, will lose single sign-on access to their AWS accounts and applications. For more information, see Considerations for changing your identity source (p. 22).

Topics

- Connect Active Directory or an external IdP and specify a user (p. 7)
- Use the default directory and create a user in IAM Identity Center (p. 8)

Connect Active Directory or an external IdP and specify a user

If you're already using Active Directory or an external identity provider (IdP), the following topics will help you connect your directory to IAM Identity Center.

You can connect an AWS Managed Microsoft AD directory, a self-managed directory in Active Directory, or an external IdP with IAM Identity Center. If you plan to connect an AWS Managed Microsoft AD directory or a self-managed directory in Active Directory, make sure that your Active Directory configuration meets the prerequisites in Active Directory or an external IdP (p. 4).

Note
As a security best practice, we strongly recommend that you enable multi-factor authentication. If you plan to connect an AWS Managed Microsoft AD directory or a self-managed directory in Active Directory and you're not using RADIUS MFA with AWS Directory Service, enable MFA in IAM Identity Center. If you plan to use an external identity provider, note that the external IdP, not IAM Identity Center, manages MFA settings. MFA in IAM Identity Center is not supported for use by external IdPs. For more information, see Enable MFA (p. 83).

AWS Managed Microsoft AD

1. Review the guidance in Connect to a Microsoft AD directory (p. 32).
2. Follow the steps in Connect a directory in AWS Managed Microsoft AD to IAM Identity Center (p. 33).
3. Configure Active Directory to synchronize the user to whom you want to grant administrative permissions into IAM Identity Center. For more information, see Synchronize an administrative user into IAM Identity Center (p. 8).

Self-managed directory in Active Directory

1. Review the guidance in Connect to a Microsoft AD directory (p. 32).
2. Follow the steps in Connect a self-managed directory in Active Directory to IAM Identity Center (p. 34).
3. Configure Active Directory to synchronize the user to whom you want to grant administrative permissions into IAM Identity Center. For more information, see Synchronize an administrative user into IAM Identity Center (p. 8).

External IdP

AWS IAM Identity Center (successor
to AWS Single Sign-On) User Guide
Use the default directory and
create a user in IAM Identity Center

1. Review the guidance in Connect to an external identity provider (p. 45).

2. Follow the steps in How to connect to an external identity provider (p. 45).

3.
    Configure your IdP to provision users into IAM Identity Center.

    > **Note**
    > Before you set up automatic, group-based provisioning of all your workforce identities from
    > your IdP into IAM Identity Center, we recommend that you sync the one user to whom you
    > want to grant administrative permissions into IAM Identity Center.

Synchronize an administrative user into IAM Identity Center

After you connect your directory to IAM Identity Center, you can specify a user to whom you want to
grant administrative permissions, and then synchronize that user from your directory into IAM Identity
Center.

1. Open the IAM Identity Center console.

2. Choose **Settings**.

3. On the **Settings** page, choose the **Identity source** tab, choose **Actions**, and then choose **Manage
    Sync**.

4. On the **Manage Sync** page, choose the **Users** tab, and then choose **Add users and groups**.

5. On the **Users** tab, under **User**, enter the exact user name and choose **Add**.

6. Under **Added Users and Groups**, do the following:

    a. Confirm that the user to whom you want to grant administrative permissions is specified.

    b. Select the check box to the left of the user name.

    c. Choose **Submit**.

7. In the **Manage sync** page, the user that you specified appears in the **Users in sync scope** list.

8. In the navigation pane, choose **Users**.

9. On the **Users** page, it might take some time for the user that you specified to appear in the list.
    Choose the refresh icon to update the list of users.

Next step: Create an administrative permission set

At this point, your user doesn't have access to the management account. You will set up administrative
access to this account by creating an administrative permission set and assigning the user to that
permission set. For more information, see Step 3: Create an administrative permission set (p. 9).

Use the default directory and create a user in IAM Identity Center

When you enable IAM Identity Center for the first time, it is automatically configured with an Identity
Center directory as your default identity source. Complete the following steps to create a user in IAM
Identity Center and activate the user account.

1. Open the IAM Identity Center console using your AWS account root user credentials.

2. Follow the steps in Add users (p. 29) to create a user.

    When you specify the user details, you can either send an email with the password setup instructions
    (this is the default option) or generate a one-time password. If you send an email, make sure that
    you specify an email address that you can access.

3.   After you add the user, return to this procedure. If you kept the default option to send an email with the password setup instructions, do the following to activate the user account:

   a.   You'll receive an email with the subject **Invitation to join AWS Single Sign-On**. Open the email and choose **Accept invitation**.

   b.   On the **New user sign up** page, enter and confirm a password, and then choose **Set new password**.

      Note
      Make sure to save your password. You'll need it later to sign into the AWS access portal using your administrative credentials (p. 10).

Next step: Create an administrative permission set

At this point, your user doesn't have access to the management account. You will set up administrative access to this account by creating an administrative permission set and assigning the user to that permission set. For more information, see Step 3: Create an administrative permission set (p. 9).

Step 3: Create an administrative permission set

Permission sets are stored in IAM Identity Center and define the level of access that users and groups have to an AWS account. Perform the following steps to create a permission set that grants administrative permissions.

1.   Open the IAM Identity Center console using your AWS account root user credentials.
2.   In the IAM Identity Center navigation pane, under **Multi-account permissions**, choose **Permission sets**.
3.   Choose **Create permission set**.
4.   **For Step 1: Select permission set type**, on the **Select permission set type** page, keep the default settings and choose **Next**. The default settings grant full access to AWS services and resources using the **AdministratorAccess** predefined permission set.

      Note
      The predefined **AdministratorAccess** permission set uses the **AdministratorAccess** AWS managed policy.
5.   **For Step 2: Specify permission set details**, on the **Specify permission set details** page, keep the default settings and choose **Next**. The default setting limits your session to one hour.
6.   **For Step 3: Review and create**, on the **Review and create** page, do the following:

   1. Review the permission set type and confirm that it is **AdministratorAccess**.
   2. Review the AWS managed policy and confirm that it is **AdministratorAccess**.
   3. Choose **Create**.

Step 4: Set up AWS account access for an administrative user

To set up AWS account access for an administrative user in IAM Identity Center, you must assign the user to the **AdministratorAccess** permission set.

1.   Open the IAM Identity Center console using your AWS account root user credentials.
2.   In the navigation pane, under **Multi-account permissions**, choose **AWS accounts**.

3. On the **AWS accounts** page, a tree view list of your organization appears. Select the check box next to the AWS account to which you want to assign administrative access. If you have multiple accounts in your organization, select the check box next to the management account.

4. Choose **Assign users or groups**.

5. For **Step 1: Select users and groups**, on the **Assign users and groups to "*AWS-account-name*"** page, do the following:

    1. On the **Users** tab, select the user to whom you want to grant administrative permissions.

        To filter the results, start typing the name of the user that you want in the search box.

    2. After you confirm that the correct user is selected, choose **Next**.

6. For **Step 2: Select permission sets**, on the **Assign permission sets to "*AWS-account-name*"** page, under **Permission sets**, select the **AdministratorAccess** permission set.

7. Choose **Next**.

8. For **Step 3: Review and Submit**, on the **Review and submit assignments to "*AWS-account-name*"** page, do the following:

    1. Review the selected user and permission set.

    2. After you confirm that the correct user is assigned to the **AdministratorAccess** permission set, choose **Submit**.

        Important
        The user assignment process might take a few minutes to complete. Leave this page open until the process successfully completes.

9. If either of the following applies, follow the steps in Enable MFA (p. 83) to enable MFA for IAM Identity Center:

    -   You're using the default Identity Center directory as your identity source.
    -   You're using an AWS Managed Microsoft AD directory or a self-managed directory in Active Directory as your identity source and you're not using RADIUS MFA with AWS Directory Service.

        Note
        If you're using an external identity provider, note that the external IdP, not IAM Identity Center, manages MFA settings. MFA in IAM Identity Center is not supported for use by external IdPs.

When you set up account access for the administrative user, IAM Identity Center creates a corresponding IAM role. This role, which is controlled by IAM Identity Center, is created in the relevant AWS account, and the policies specified in the permission set are attached to the role.

Step 5: Sign in to the AWS access portal with your administrative credentials

Complete the following steps to confirm that you can sign in to the AWS access portal by using the credentials of the administrative user, and that you can access the AWS account.

1. Open the IAM Identity Center console using your AWS account root credentials.
2. In the navigation pane, choose **Dashboard**.
3. On the **Dashboard** page, under **Settings summary**, copy the AWS access portal URL.
4. Open a separate browser, paste in the AWS access portal URL that you copied, and press **Enter**.

AWS IAM Identity Center (successor
to AWS Single Sign-On) User Guide
Step 6: Set up AWS account access
for additional users (optional)

5. Sign in by using either of the following:

    - If you're using Active Directory or an external identity provider (IdP) as your identity source, sign in by using the credentials of the Active Directory or IdP user that you assigned to the **AdministratorAccess** permission set in IAM Identity Center.
    - If you're using the default Identity Center directory as your identity source, sign in by using the user name that you specified when you created the user and the new password that you specified when you activated the user account. For more information, see Use the default directory and create a user in IAM Identity Center (p. 8).

6. After you are signed in, an **AWS account** icon appears in the portal.

7. When you select the **AWS account** icon, the account name, account ID, and email address associated with the account appear.

8. Choose the name of the account to display the **AdministratorAccess** permission set, and select the **Management Console** link to the right of **AdministratorAccess**.

    When you sign in, the name of the permission set to which the user is assigned appears as an available role in the AWS access portal. Because you assigned this user to the `AdministratorAccess` permission set, the role will appear in the AWS access portal as: `AdministratorAccess/`*`username`*

9. If you are redirected to the AWS Management Console, you successfully finished setting up administrative access to the AWS account. Proceed to step 10.

10. Switch to the browser that you used to sign into the AWS Management Console and set up IAM Identity Center, and sign out from your AWS account root user.

    Important
    We strongly recommend that you use the credentials of the administrative user when you sign in to the AWS access portal. Safeguard your root user credentials and use them to perform the tasks that only the root user can perform. To enable other users to access your accounts and applications, and to administer IAM Identity Center, create and assign permission sets only through IAM Identity Center.

Step 6: Set up AWS account access for additional users (optional)

Now that you've created an administrative user in IAM Identity Center, you can add other users by doing any of the following:

- Synchronizing your users from Active Directory (p. 38)
- Synchronizing your users from your external identity provider (IdP) (p. 45)
- Creating your users in IAM Identity Center (p. 29)

After you add other users, create permission sets for these users and assign the users to the new permission sets as needed to grant them single sign-on access to one or more AWS accounts in your organization.

If you are using the default Identity Center directory as an identity source, after your users accept their invitation (p. 77) to activate their account and they sign into the AWS access portal, the only icons that appear in the portal are for the AWS accounts to which the users are assigned. Users who are assigned to multiple permission sets can sign in to the AWS access portal, choose an account, and then choose a role that was created from an assigned permission set.

For information about how to assign additional users single sign-on access to your AWS accounts by using the console, see Assign user access to AWS accounts (p. 95). Alternatively, you can use AWS

CloudFormation to create and assign permission sets and assign users to those permission sets. Users can then sign in to the AWS access portal (p. 78) or use AWS Command Line Interface (AWS CLI) commands.

Step 7: Set up single sign-on access to your applications (optional)

With IAM Identity Center, you can use AWS applications that are integrated with IAM Identity Center, cloud applications for which AWS provides preintegration, and custom SAML 2.0 applications. The configuration steps for setting up single sign-on access to applications vary based on the application type.

- Add and configure an Identity Center enabled application (p. 112)
- Add and configure a cloud application (p. 115)
- Add and configure a custom SAML 2.0 application (p. 116)

For more information about supported application types, see Application assignments (p. 111).

After you follow the guidance in the relevant topics, your users can access your applications from within their AWS access portal based on the permissions that you assigned.

Key concepts

You'll get more out of AWS IAM Identity Center (successor to AWS Single Sign-On) if you become familiar with key concepts relating to SAML federation, user authentication, and IAM permissions.

Topics

Users, groups, and provisioning

IAM Identity Center manages access to all your AWS Organizations accounts, Identity Center enabled applications, and other business applications that support the Security Assertion Markup Language (SAML) 2.0 standard.

User name and email address uniqueness

When working in IAM Identity Center, users must be uniquely identifiable. IAM Identity Center implements a user name that is the primary identifier for your users. Although most people set the user name equal to a user's email address, IAM Identity Center and the SAML standard do not require this. However, a large percentage of SAML-based applications use an email address as the unique identifier for users. They obtain this from assertions that a SAML identity provider sends during authentication. Such applications depend upon the uniqueness of email addresses for each user. As such, IAM Identity Center allows you to specify something other than an email address for user sign-in. IAM Identity Center requires that all user names and email addresses for your users are non-NULL and unique.

Groups

Groups are a logical combination of users that you define. You can create groups and add users to the groups. IAM Identity Center does not support adding a group to a group (nested groups). Groups are useful when assigning access to AWS accounts and applications. Rather than assign each user individually, you give permissions to a group. Later, as you add or remove users from a group, the user dynamically gets or loses access to accounts and applications that you assigned to the group.

User and group provisioning

You can create users and groups directly in IAM Identity Center, or work with users and groups you have in Active Directory or an another external identity provider. Before IAM Identity Center can be used to assign users and groups access permissions in an AWS account, IAM Identity Center must first be aware of the users and groups. Similarly, Identity Center enabled applications can work with users and groups for which IAM Identity Center is aware. Provisioning is the process of making user and group information available for use by IAM Identity Center and Identity Center enabled applications.

Provisioning in IAM Identity Center varies based on the identity source that you use. For more information, see Manage your identity source (p. 21).

Identity Center enabled applications

IAM Identity Center provides support for integration by other AWS applications and services. These applications can use IAM Identity Center to perform authentication and can access information about users and groups. For example, a user might sign into an application that generates performance dashboards for resources that the user controls. The user might then share the dashboard by looking up a group in IAM Identity Center.

To enable this capability, IAM Identity Center provides an identity store which contains user and group attributes, excluding sign-in credentials.

You can use either of the following methods to keep the users and groups in your IAM Identity Center identity store updated:

- Use the IAM Identity Center identity store as your main identity source. If you choose this method, you manage your users and groups from within the IAM Identity Center console or AWS CLI.
- Set up provisioning (synchronization) of users and groups coming from either of the following identity sources to your IAM Identity Center identity store:
  - **Active Directory** - For more information, see .
  - **External identity provider** - For more information, see .

  If you choose this provisioning method, you continue managing your users and groups from within your identity source and those changes would get synced to the IAM Identity Center identity store.

Regardless of which identity source you choose, IAM Identity Center has the ability to share the user and group information with Identity Center enabled applications. This capability makes it possible to connect an identity source to IAM Identity Center once and then share identity information with multiple applications in the AWS Cloud. This eliminates the need to set up federation and identity provisioning with each application independently. This sharing feature also makes it easy to give users in your workforce access to many applications in different AWS accounts.

Considerations for sharing identity information in AWS accounts

The attributes contained in IAM Identity Center are the basic attributes commonly used across applications. These attributes include information such as first and last name, phone number, email address, address, and preferred language. You might want to consider which applications and which accounts can use this personally identifiable information.

To control access to this information, you have two options. First, you can choose to enable access in only the AWS Organizations management account or in all AWS Organizations accounts. Second, you can use service control policies (SCPs) to control which applications can access the information in which AWS Organizations accounts. For example, if you enable access in the AWS Organizations management account only, then applications in member accounts have no access to the information. However, if you enable access in all accounts, you can use SCPs to disallow access by all applications except those you want to permit.

Allow Identity Center enabled applications in AWS accounts

When you enable IAM Identity Center for the first time, AWS allows use of Identity Center enabled applications automatically in all AWS Organizations accounts. To constrain applications, you must implement SCPs.

If you enabled IAM Identity Center prior to November 25, 2019, IAM Identity Center disables the use of Identity Center enabled applications in all AWS Organizations accounts. To use Identity Center enabled applications, you must enable them in the management account and optionally enable them in member accounts. If you enable them in the management account only, you can enable them in member accounts in the future. To enable these applications, use the **Enable access** option in the IAM Identity Center **Settings** page in the Identity Center enabled applications section.

SAML federation

IAM Identity Center supports identity federation with SAML (Security Assertion Markup Language) 2.0. SAML 2.0 is an industry standard used for securely exchanging SAML assertions that pass information about a user between a SAML authority (called an identity provider or IdP), and a SAML consumer (called a service provider or SP). IAM Identity Center uses this information to provide federated single sign-on access for those users who are authorized to use applications within the AWS access portal.

IAM Identity Center adds SAML IdP capabilities to either your AWS Managed Microsoft AD or your IAM Identity Center identity store. Users can then single sign-on into services that support SAML, including the AWS Management Console and third-party applications such as Microsoft 365, SAP Concur, and Salesforce.

User authentications

A user signs in to the AWS access portal using their user name. When they do, IAM Identity Center redirects the request to the IAM Identity Center authentication service based on the directory associated with the user email address. Once authenticated, users have single sign-on access to any of the AWS accounts and third-party software-as-a-service (SaaS) applications that show up in the portal without additional sign-in prompts. This means that users no longer need to keep track of multiple account credentials for the various assigned AWS applications that they use on a daily basis.

Authentication sessions

There are two types of authentication sessions maintained by IAM Identity Center: one to represent the users' sign in to IAM Identity Center, and another to represent the users' access to IAM Identity Center enabled applications, such as Amazon SageMaker Studio or Amazon Managed Grafana. Each time a user signs in to IAM Identity Center, a sign in session is created for the duration configured in Identity Center, which can be up to 7 days. For more information, see Manage IAM Identity Center integrated application sessions (p. 25). Each time the user accesses an Identity Center enabled application, the IAM Identity Center sign in session is used to obtain an IAM Identity Center application session for that application. IAM Identity Center application sessions have a refreshable 1-hour lifetime – that is, IAM Identity Center application sessions are automatically refreshed every hour as long as the IAM Identity Center sign in session from which they were obtained is still valid. When the user uses IAM Identity Center to access the AWS Management Console or CLI, the IAM Identity Center sign in session is used to obtain an IAM session, as specified in the corresponding IAM Identity Center permission set (more specifically, IAM Identity Center assumes an IAM role, which IAM Identity Center manages, in the target account).

When you disable or delete a user in IAM Identity Center, that user will immediately be prevented from signing in to create new IAM Identity Center sign in sessions. IAM Identity Center sign in sessions are cached for one hour, which means that when you disable or delete a user while they have an active IAM Identity Center sign in session, their existing IAM Identity Center sign in session will continue for up to an hour, depending on when the sign in session was last refreshed. During this time, the user can initiate new IAM Identity Center application and IAM role sessions.

After the IAM Identity Center sign in session expires, the user can no longer initiate new IAM Identity Center application or IAM role sessions. However, IAM Identity Center application sessions can also be cached for up to an hour, such that the user may retain access to an Identity Center enabled application for up to an hour after the IAM Identity Center sign in session has expired. Any existing IAM role sessions will continue based on the duration configured in the IAM Identity Center permission set (admin-configurable, up to 12 hours).

The table below summarizes these behaviors:

User experience / system behavior	Time after user is disabled / deleted
User can no longer sign in to IAM Identity Center; user cannot obtain a new IAM Identity Center sign in session	None (effective immediately)
User can no longer start new application or IAM role sessions via IAM Identity Center	Up to 1 hour
User can no longer access any Identity Center enabled applications (all app sessions are terminated)	Up to 2 hours (up to 1 hour for IAM Identity Center sign in session expiry, plus up to 1 hour for IAM Identity Center app session expiry)
User can no longer access any AWS accounts through IAM Identity Center	Up to 13 hours (up to 1 hour for IAM Identity Center sign in session expiry, plus up to 12 hours for administrator-configured IAM role session expiry per the IAM Identity Center session duration settings for the permission set)

For more information about sessions, see Set session duration (p. 101).

Permission sets

A permission set is a template that you create and maintain that defines a collection of one or more IAM policies. Permission sets simplify the assignment of AWS account access for users and groups in your organization. For example, you can create a *Database Admin* permission set that includes policies for administering AWS RDS, DynamoDB, and Aurora services, and use that single permission set to grant access to a list of target AWS accounts within your AWS Organization for your database administrators.

IAM Identity Center assigns access to a user or group in one or more AWS accounts with permission sets. When you assign a permission set, IAM Identity Center creates corresponding IAM Identity Center-controlled IAM roles in each account, and attaches the policies specified in the permission set to those roles. IAM Identity Center manages the role, and allows the authorized users you've defined to assume the role, by using the IAM Identity Center User Portal or AWS CLI. As you modify the permission set, IAM Identity Center ensures that the corresponding IAM policies and roles are updated accordingly.

You can add AWS managed policies, customer managed policies, inline policies, and AWS managed policies for job functions to your permission sets. You can also assign an AWS managed policy or a customer managed policy as a permissions boundary.

To create a permission set, see Create and manage permission sets (p. 98).

Topics

- Predefined permissions (p. 17)
- Custom permissions (p. 17)

Predefined permissions

You can create a permission set with either **Predefined permissions** or **Custom permissions (p. 17)**.

When you create a permission set with predefined permissions, you choose one policy from a list of AWS managed policies. Within the available policies, you can choose from **Common permission policies** and **Job function policies**.

Common permission policies

Choose from a list of AWS managed policies that make it possible to access resources in your entire AWS account. You can add one of the following policies:

- AdministratorAccess
- PowerUserAccess
- ReadOnlyAccess
- ViewOnlyAccess

Job function policies

Choose from a list of AWS managed policies that make it possible to access resources in your AWS account that might be relevant to a job within your organization. You can add one of the following policies:

- Billing
- DataScientist
- DatabaseAdministrator
- NetworkAdministrator
- SecurityAudit
- SupportUser
- SystemAdministrator

For detailed descriptions of the available common permission policies and job function policies, see AWS managed policies for job functions in the *AWS Identity and Access Management user guide*.

Custom permissions

When you create a permission set with **Custom permissions**, you can combine any of the AWS managed and customer managed policies that you have in AWS Identity and Access Management (IAM) with *inline policies*, and a *permissions boundary* that sets the maximum possible permissions that any other policy can grant to users of your permission set.

For a detailed walkthrough of the process to create a permission set, see Create and manage permission sets (p. 98).

Policy types that you can attach to your permission set

Topics

- Inline policies (p. 18)

- AWS managed policies (p. 18)
- Customer managed policies (p. 18)
- Permissions boundaries (p. 19)

Inline policies

You can attach an *inline policy* to a permission set. An inline policy is a block of text formatted as an IAM policy that you add directly to your permission set. You can paste in a policy, or generate a new one with the policy creation tool in the IAM Identity Center console when you create a new permission set. You can also create IAM policies with the AWS Policy Generator.

When you deploy a permission set with an inline policy, IAM Identity Center creates an IAM policy in the AWS accounts where you assign your permission set. IAM Identity Center creates the policy when you assign the permission set to the account. The policy is then attached to the IAM role in your AWS account that your user assumes.

When you create an inline policy and assign your permission set, IAM Identity Center configures the policies in your AWS accounts for you. When you build your permission set with Customer managed policies (p. 18), you must create the policies in your AWS accounts yourself before you assign the permission set.

AWS managed policies

You can attach *AWS managed policies* to your permission set. AWS managed policies are IAM policies that AWS maintains. In contrast, Customer managed policies (p. 18) are IAM policies in your account that you create and maintain. AWS managed policies address common least privilege use cases in your AWS account. You can assign an AWS managed policy as permissions for the role that IAM Identity Center creates, or as a permissions boundary.

AWS maintains AWS managed policies for job functions that assign job-specific access permissions to your AWS resources. You can add one job-function policy when you choose to use **Predefined permissions** with your permission set. When you choose **Custom permissions**, you can add more than one job-function policy.

Your AWS account also contains a large number of AWS managed IAM policies for specific AWS services and combinations of AWS services. When you create a permission set with **Custom permissions**, you can choose from many additional AWS managed policies to assign to your permission set.

AWS populates every AWS account with AWS managed policies. To deploy a permission set with AWS managed policies, you don't need to first create a policy in your AWS accounts. When you build your permission set with Customer managed policies (p. 18), you must create the policies in your AWS accounts yourself before you assign the permission set.

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

Customer managed policies

You can attach *customer managed policies* to your permission set. Customer managed policies are IAM policies in your account that you create and maintain. In contrast, AWS managed policies (p. 18) are IAM policies in your account that AWS maintains. You can assign an customer managed policy as permissions for the role that IAM Identity Center creates, or as a permissions boundary.

When you create a permission set with a customer managed policy, you must create an IAM policy with the same name and path in each AWS account where IAM Identity Center assigns your permission set. If you are specifying a custom path, make sure to specify the same path in each AWS account. For more information, see Friendly names and paths in the *IAM User Guide*. IAM Identity Center attaches the IAM

policy to the IAM role that it creates in your AWS account. As a best practice, apply the same permissions to the policy in each account where you assign the permission set. For more information, see Use IAM policies in permission sets (p. 100).

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

Permissions boundaries

You can attach a *permissions boundary* to your permission set. A permissions boundary is an AWS managed or customer managed IAM policy that sets the maximum permissions that an identity-based policy can grant to an IAM principal. When you apply a permissions boundary, your Inline policies (p. 18), Customer managed policies (p. 18), and AWS managed policies (p. 18) can't grant any permissions that exceed the permissions that your permissions boundary grants. A permissions boundary doesn't grant any permissions, but instead makes it so that IAM ignores all permissions beyond the boundary.

When you create a permission set with a customer managed policy as a permissions boundary, you must create an IAM policy with the same name in each AWS account where IAM Identity Center assigns your permission set. IAM Identity Center attaches the IAM policy as a permissions boundary to the IAM role that it creates in your AWS account .

For more information, see Permissions boundaries for IAM entities in the IAM User Guide.

Workforce identities

AWS Identity and Access Management (IAM) helps you securely manage identities and access to AWS services and resources. As an IAM service, AWS IAM Identity Center (successor to AWS Single Sign-On) is where you create, or connect, your *workforce identities* in AWS once and manage access centrally to your multiple AWS accounts and applications.

For IAM Identity Center customers, there is no change to how you centrally manage access to multiple AWS accounts or applications. For new customers to IAM Identity Center, you can flexibly configure IAM Identity Center to run alongside or replace single AWS account access management using IAM.

Topics

Use cases

Review the following use case to determine how best to use IAM Identity Center for your business needs.

Topics

Enable single sign-on access to your AWS applications (Application admin role)

This use case provides guidance for application administrators who manage Identity Center enabled applications (p. 111) such as Amazon SageMaker or AWS IoT SiteWise, and need to provide single sign-on access to their end users.

Before you get started, consider the following questions:

- Should you create a test or production environment in your own separate AWS organization?
- Is IAM Identity Center already enabled in your AWS organization? Do you have permissions to enable IAM Identity Center in the management account of AWS Organizations?

Depending on your specific business needs, review the guidance below.

Configure my AWS application in a standalone AWS account

If you need to provide single sign-on access to an AWS application and know that your IT department does not yet use IAM Identity Center, you may need to create a standalone AWS account so that you can get started. By default, when you create your own AWS account, you'll have the necessary permissions, such as Root user, to create and manage your own AWS organization. This permission is required to enable IAM Identity Center. For more information about the permissions necessary to enable IAM Identity Center, see Identity and access management for IAM Identity Center (p. 130).

AWS IAM Identity Center (successor
to AWS Single Sign-On) User Guide
Enable single sign-on access to
your AWS EC2 Windows instances

If they are not already enabled, IAM Identity Center and AWS Organizations can be enabled automatically during the process of setting up some AWS applications (for example Amazon Managed Grafana does this). If your AWS application does not provide the option to enable them, you'll need to first setup AWS Organizations and IAM Identity Center before you can provide single sign-on access to your application.

IAM Identity Center is currently not configured in my organization

In your role as an application admin, you might not be able to enable IAM Identity Center as it requires specific permissions in the AWS Organizations management account. In this case, you will need to coordinate with your IT or cloud administrators to get IAM Identity Center enabled in the Organizations management account.

If you do have sufficient permissions to enable IAM Identity Center within your organization, do so first, then proceed with the application setup. For more information, see Getting started (p. 4).

IAM Identity Center is currently configured in my organization

In this scenario, you can continue to deploy your AWS application without any further steps or requirements.

> **Note**
> If your organization enabled IAM Identity Center in the management account before November 25th, 2019, you must also enable Identity Center enabled applications in the management account and optionally in the member accounts. If you enable them in the management account only, you can enable them in member accounts later. To enable these applications, choose **Enable access** in the IAM Identity Center console's **Settings** page in the Identity Center enabled applications section. For more information, see Identity Center enabled applications (p. 14).

Enable single sign-on access to your AWS EC2 Windows instances

This use case provides guidance for application administrators who manage users in an AWS IAM Identity Center (successor to AWS Single Sign-On) identity store or supported external identity provider, and need to provide IAM Identity Center access to their EC2 Windows desktops from the AWS Fleet Manager console.

This allows you to securely access your EC2 Windows instances with existing corporate user names, passwords, and MFA devices. You are no longer required to share administrator credentials, access credentials multiple times, or configure remote access client software. You can centrally grant and revoke access to your EC2 Windows instances at scale across multiple AWS accounts. For example, if you remove an employee from your IAM Identity Center integrated identity source, they automatically lose access to all AWS resources, including EC2 Windows instances.

For more information, see How to enable secure seamless single sign-on to Amazon EC2 Windows instances with IAM Identity Center.

For a demonstration of how to configure IAM Identity Center to enable this capability, see Enabling Single Sign-on to EC2 Windows with IAM Identity Center.

Manage your identity source

Your identity source in IAM Identity Center defines where your users and groups are managed. After you configure your identity source, you can look up users or groups to grant them single sign-on access to AWS accounts, cloud applications, or both.

You can have only one identity source per organization in AWS Organizations. You can choose one of the following as your identity source:

- **Identity Center directory –** When you enable IAM Identity Center for the first time, it is automatically configured with an Identity Center directory as your default identity source. This is where you create your users and groups, and assign their level of access to your AWS accounts and applications.
- **Active Directory –** Choose this option if you want to continue managing users in either your AWS Managed Microsoft AD directory using AWS Directory Service or your self-managed directory in Active Directory (AD).
- **External identity provider –** Choose this option if you want to manage users in an external identity provider (IdP) such as Okta or Azure Active Directory.

> **Note**
> IAM Identity Center does not support SAMBA4-based Simple AD as an identity source.

Topics

Considerations for changing your identity source

Although you can change your identity source at any time, we recommend that you consider how this change might affect your current deployment. If you're already managing users and groups in one identity source, changing to a different identity source might remove all user and group assignments that you configured in IAM Identity Center. If this occurs, all users, including the administrative user in IAM Identity Center, will lose single sign-on access to their AWS accounts and applications.

Before you change the identity source for IAM Identity Center, review the following considerations before you proceed. If you want to proceed with changing your identity source, see Change your identity source (p. 23) for more information.

Changing between IAM Identity Center and Active Directory

If you're already managing users and groups in Active Directory, we recommend that you consider connecting your directory when you enable IAM Identity Center and choose your identity source. This should be done before you create any users and groups in the default Identity Center directory and make any assignments. If you're already managing users and groups in the default Identity Center directory, changing your identity source to Active Directory deletes your users and groups from the Identity Center directory. This change also removes your assignments. In this case, after you change to Active Directory, you must synchronize your users and groups from Active Directory into the Identity Center directory, and then reapply their assignments. If you choose to not use Active Directory, you must create your users and groups in the Identity Center directory, and then make assignments.

> **Note**
> Changing your identity source between IAM Identity Center and Active Directory also changes the URL for the AWS access portal.

For information about how IAM Identity Center provisions users and groups, see Connect to a Microsoft AD directory (p. 32).

Changing between IAM Identity Center and an external identity provider (IdP)

If you change your identity source from IAM Identity Center to an external IdP, IAM Identity Center preserves all your assignments. However, these assignments will work only if the user names and groups in IAM Identity Center match those in the external IdP. Any user names and groups that don't match are unusable. If you change from an external IdP to IAM Identity Center, IAM Identity Center preserves all users, groups, and assignments. Users who had passwords in IAM Identity Center can continue signing in with their old passwords. For users who were in the external IdP and weren't in IAM Identity Center, you must force a password reset.

For information about how IAM Identity Center provisions users and groups, see Connect to an external identity provider (p. 45).

Changing from one external IdP to an external IdP

If you're already using an external IdP as your identity source for IAM Identity Center and you change to a different external IdP, IAM Identity Center preserves all of your assignments. The user assignments, group assignments, and group memberships will continue to work as long as the new IdP sends the correct assertions (for example, SAML nameIDs). These assertions must match the user names in IAM Identity Center when your users authenticate through the new external IdP. If you are using SCIM for provisioning into IAM Identity Center, we recommend that you review the IdP-specific information in this guide and the documentation provided by the IdP to ensure that the new provider will match users and groups correctly when SCIM is enabled.

For information about how IAM Identity Center provisions users and groups, see Connect to an external identity provider (p. 45).

Changing between Active Directory and an external IdP

If you change your identity source from an external IdP to Active Directory, or from Active Directory to an external IdP, all users, groups, and assignments are deleted from IAM Identity Center. No user or group information is affected in either the external IdP or Active Directory. If you change to an external IdP, you must configure IAM Identity Center to provision your users. Alternatively, you must manually provision the users and groups for the external IdP before you can configure assignments. If you change to Active Directory, you must create assignments with the users and groups that are in your directory in Active Directory.

For information about how IAM Identity Center provisions users and groups, see Connect to a Microsoft AD directory (p. 32).

Change your identity source

The following procedure describes how to change from a directory that IAM Identity Center provides (the default Identity Center directory) to Active Directory or an external identity provider, or the other way around. Before you proceed, review the information in Considerations for changing your identity source (p. 22). Depending on your current deployment, this change might remove any user and group assignments that you configured in IAM Identity Center. If this occurs, all users, including the administrative user in IAM Identity Center, will lose single sign-on access to their AWS accounts and applications.

To change your identity source

1. Open the IAM Identity Center console.
2. Choose **Settings**.

AWS IAM Identity Center (successor
to AWS Single Sign-On) User Guide
Manage sign-in and attribute
use for all identity source types

3. On the **Settings** page, choose the **Identity source** tab. Choose **Actions**, and then choose **Change identity source**.

4. Under **Choose identity source**, select the source that you want to change to, and then choose **Next**.

   If you are changing to Active Directory, choose the available directory from the menu on the next page.

   > **Important**
   > Changing your identity source to or from Active Directory deletes users and groups from the Identity Center directory. This change also removes any assignments that you configured in IAM Identity Center.

   If you are switching to an external identity provider, we recommend that you follow the steps in How to connect to an external identity provider (p. 45).

5. After you read the disclaimer and are ready to proceed, type **ACCEPT**.

6. Choose **Change identity source**. If you are changing your identity source to Active Directory, proceed to the next step.

7. Changing your identity source to Active Directory takes you to the **Settings** page. On the **Settings** page, do either of the following:

   - Choose **Start guided setup**. For information about how to complete the guided setup process, see Guided setup (p. 41).
   - In the **Identity source** section, choose **Actions**, and then choose **Manage sync** to configure your *sync scope*, the list of users and groups to sync.

Manage sign-in and attribute use for all identity source types

IAM Identity Center provides the following set of features that enables admins to control AWS access portal use, to set session durations for users in the AWS access portal and your applications, and to utilize attributes for access control. These features work with an Identity Center directory or external identity provider as your identity source. Session management is currently not supported for an Active Directory identity source.

Topics

- Disable user access (p. 24)
- Manage IAM Identity Center integrated application sessions (p. 25)
- Configure AWS access portal session duration (p. 25)
- Manage AWS access portal sessions (p. 26)
- Supported user and group attributes (p. 27)
- Using predefined attributes from the IAM Identity Center directory for access control in AWS (p. 27)

Disable user access

When you disable user access, you cannot edit their user details, reset their password, add the user to a group, or view their group membership. Use the following procedure to disable user access in your Identity Center directory.

> **Note**
> When you disable user access or delete a user in IAM Identity Center, that user will immediately be prevented from signing in to the AWS access portal and will not be able to create new sign in sessions. For more information, see Authentication sessions (p. 15).

AWS IAM Identity Center (successor
to AWS Single Sign-On) User Guide
Manage sign-in and attribute
use for all identity source types

To disable user access

1. Open the IAM Identity Center console.
2. Choose **Users**.
3. Choose the user whose access you want to disable.
4. By **General information**, choose **Disable user access**.
5. On the **Disable user access** dialog, choose **Disable user access**.

Manage IAM Identity Center integrated application sessions

Administrators can customize session duration for the AWS access portal to define how often users are required to re-authenticate. Administrators can also terminate AWS access portal sessions. The AWS access portal session duration changes the duration of IAM Identity Center integrated application sessions, and AWS access portal session termination also affects these applications. IAM Identity Center integrated applications poll the AWS access portal sessions and terminate when they detect that the AWS access portal session has ended.

For more information about how to configure the length of AWS access portal sessions, see Configure AWS access portal session duration (p. 25). For more information about how to manage and delete user sessions, see Manage AWS access portal sessions (p. 26). Modifying the AWS access portal session duration and terminating AWS access portal sessions has no effect on the AWS Management Console session duration that you define in your permission sets.

Configure AWS access portal session duration

Use the following procedure to configure the AWS access portal session duration for all your users.

If you have an external identity provider (IdP) connected to IAM Identity Center, the AWS access portal session duration will be the lesser of the session duration you set in your IdP or the session duration you set here. For example, if your IdP session duration is 24 hours and you set an 18 hour session duration in IAM Identity Center, the users must re-authenticate in the AWS access portal after 18 hours. If you set a 72 hour session duration here and your IdP has a session duration of 18 hours, your users must re-authenticate after 18 hours.

The access portal session duration also applies to CLI sessions. If you refresh your permission set just before the IAM Identity Center session was set to expire and the session duration is set to 20 hours while permission set duration is set to 12 hours, the CLI session runs for the maximum of 20 hours plus 12 hours for a total of 32 hours. You must update your AWS CLI and SDKs to the minimum supported version to enable IAM Identity Center session management for AWS CLI and SDKs. To learn the minimum supported version for each SDK, see Minimum supported versions for CLI and SDKs needed to configure IAM Identity Center session duration (p. 26). For more information about the IAM Identity Center CLI, see AWS CLI Command Reference. For more information on how to install or update the latest CLI version, see Installing or updating the latest version of the AWS CLI.

To configure an AWS access portal session duration

1. Open the IAM Identity Center console.
2. Choose **Settings**.
3. On the **Settings** page, choose the **Authentication** tab.
4. Under **Authentication**, next to **Session settings**, choose **Configure**. A **Configure session settings** dialog box appears.
5. In the **Configure session settings** dialog box, choose the maximum session duration in minutes, hours, and days for your users by selecting the drop down arrow. Choose a the length for the session, and then choose **Save**. You return to the **Settings** page.

AWS IAM Identity Center (successor
to AWS Single Sign-On) User Guide
Manage sign-in and attribute
use for all identity source types

Minimum supported versions for CLI and SDKs needed to configure IAM Identity Center session duration

See the following table for the minimum supported version for the AWS CLI and SDKs that you can use to enable IAM Identity Center session management.

SDK	Minimum version
Python	1.26.10
PHP	3.245.0
Ruby	aws-sdk-core 3.167.0
CLI V1	1.27.10
CLI V2	2.9.0
Java V2	AWS SDK for Java v2 (2.18.13)
Go V2	Whole SDK: release-2022-11-11 and specific Go modules: credentials/v1.13.0, config/v1.18.0
JS V2	2.1253.0
JS V3	v3.210.0
C++	1.9.372
.NET	v3.7.400.0

Manage AWS access portal sessions

Use the following procedure to manage sessions for a user in your IAM Identity Center store.

To manage an AWS access portal session

1. Open the IAM Identity Center console.
2. Choose **Users**.
3. On the **Users** page, choose the username of the user whose sessions you want to manage. This takes you to a page with the user's information.
4. On the user's page, choose the **Active sessions** tab. The number in parentheses next to **Active sessions** indicates the number of current active sessions for this user.
5. Select the check boxes beside the sessions that you want to delete, and then choose **Delete session**. A dialog box appears that confirms you're deleting active sessions for this user. Read the information in the dialog box, and if you want to continue, choose **Delete session**.

   Note
   Deleting a session ends the AWS access portal session, but it does not affect the user's active AWS Management Console sessions that they created by choosing a permission set from the AWS access portal. Those sessions will continue until the permission set session duration elapses or the user signs out of the session. This also does not affect any active SAML application sessions. Those sessions continue until the user signs out of the application or the application ends its session with the user. For IAM Identity Center integrated applications such as Amazon SageMaker Studio or Amazon Monitron, those sessions end the next time the application checks to see if the AWS access portal session is still active for up to 2 hours, or if the user signs out of the application.

AWS IAM Identity Center (successor
to AWS Single Sign-On) User Guide
Manage sign-in and attribute
use for all identity source types

> **Note**
> When you delete a session, any running AWS CLI sessions are also revoked. Revoking these sessions does not happen immediately and can take up to an hour.

6. You return to the user's page. A green flash bar appears to indicate that the selected sessions were successfully deleted.

Supported user and group attributes

Attributes are pieces of information that help you define and identify individual user or group objects, such as `name`, `email`, or `members`. IAM Identity Center supports most commonly used attributes regardless if they are entered manually during user creation or when automatically provisioned using a synchronization engine such as defined in the System for Cross-Domain Identity Management (SCIM) specification. For more information about this specification, see https://tools.ietf.org/html/rfc7642. For more information about manual and automatic provisioning, see Provisioning when users come from an external IdP (p. 45).

Because IAM Identity Center supports SCIM for automatic provisioning use cases, the Identity Center directory supports all of the same user and group attributes that are listed in the SCIM specification, with a few exceptions. The following sections describe which attributes are not supported by IAM Identity Center.

User objects

All attributes from the SCIM user schema (https://tools.ietf.org/html/rfc7643#section-8.3) are supported in the IAM Identity Center identity store, except for the following:

- `password`
- `ims`
- `photos`
- `entitlements`
- `x509Certificates`

All sub-attributes for users are supported, except for the following:

- `'display'` sub-attribute of any multi-valued attribute (For example, `emails` or `phoneNumbers`)
- `'version'` sub-attribute of `'meta'` attribute

Group objects

All attributes from the SCIM group schema (https://tools.ietf.org/html/rfc7643#section-8.4) are supported.

All sub-attributes for groups are supported, except for the following:

- `'display'` sub-attribute of any multi-valued attribute (For example, `members`).

Using predefined attributes from the IAM Identity Center directory for access control in AWS

Each user in the IAM Identity Center directory is assigned a unique `UserId`. You can view the `UserId` for your users by using the IAM Identity Center console and navigating to each user or by using

the DescribeUser API action. IAM Identity Center allows you to use this `UserId` in permissions sets or resource-based policies for making access control decisions in AWS. For example, the following permissions policy allows a user with <UserId> to call get operations against all of their Amazon S3 resources. This permission allows the user to read the object data from *mybucket*.

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Effect":"Allow",
            "Principal":"*",
            "Action":[
                "s3:GetObject",
                "s3:GetObjectVersion"
            ],
            "Resource":[
                "arn:aws:s3:::mybucket/*"
            ],
            "Condition":{
                "StringEquals":{
                    "identitystore:UserId":[
                        "<UserId>"
                    ]
                }
            }
        }
    ]
}
```

If you want to confirm that the caller is an IAM Identity Center user, you can specify a `Null` clause as a condition, as follows:

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Condition":{
                "Null":{
                    "identitystore:UserId":"false"
                }
            }
        }
    ]
}
```

Manage identities in IAM Identity Center

- Create your users and groups.
- Add your users as members to the groups.
- Assign the groups with the desired level of access to your AWS accounts and applications.

If you prefer to manage users in AWS Managed Microsoft AD, you can stop using your Identity Center directory at any time and instead connect IAM Identity Center to your directory in Microsoft AD by using AWS Directory Service. For more information, see Connect to a Microsoft AD directory (p. 32).

If you prefer to manage users in an external identity provider (IdP), you can connect IAM Identity Center to your IdP and enable automatic provisioning. For more information, see Connect to an external identity provider (p. 45).

Note

When identities are deleted in the Identity Center directory, corresponding assignments also get deleted in IAM Identity Center. However in Microsoft AD, when identities are deleted (either in AD or the synced in identities), corresponding assignments are not deleted.

To manage users and groups in the IAM Identity Center store, AWS supports the API operations listed in Identity Center Actions.

Note

If you use an external identity provider or Active Directory as your identity source, we recommend that you use Create, Update, and Delete APIs with caution. IAM Identity Center does not support outbound synchronization, so your identity source does not automatically update with the changes that you make to users or groups using these APIs.

Provisioning when users are in IAM Identity Center

When you create users and groups directly in IAM Identity Center, provisioning is automatic. These identities are immediately available for use in making assignments and for use by Identity Center enabled applications. For more information, see User and group provisioning (p. 13).

Topics

- Add users (p. 29)
- Add groups (p. 30)
- Add users to groups (p. 30)
- Edit user properties (p. 31)
- Reset a user password (p. 31)
- Send email OTP for users created from API (p. 32)
- Password requirements when managing identities in IAM Identity Center (p. 32)

Add users

Users and groups that you create in your Identity Center directory are available in IAM Identity Center only. Use the following procedure to add users to your Identity Center directory. Alternatively, you can call the AWS API operation CreateUser to add users.

To add a user

1. Open the IAM Identity Center console.
2. Choose **Users**.
3. Choose **Add user** and provide the following required information:

   a. **Username** – This user name is equired to sign in to the AWS access portal and can't be changed later. It must be between 1 and 100 characters.

   b. **Password** – You can either send an email with the password setup instructions (this is the default option) or generate a one-time password. If you are creating an administrative user and you choose to send an email, make sure that you specify an email address that you can access.

      i. **Send an email to this user with password setup instructions.** – This option automatically sends the user an email addressed from Amazon Web Services, with the subject line **Invitation to join AWS Single Sign-On**. The email invites the user on behalf of your company to access the IAM Identity Center AWS access portal.

         Note

         All emails sent by the IAM Identity Center service will come from either the address no-reply@signin.aws or no-reply@login.awsapps.com. We recommend that you

configure your email system so that it accepts emails from these sender email addresses and does not handle them as junk or spam.

   ii. **Generate a one-time password that you can share with this user.** – This option provides you with the AWS access portal URL and password details that you can manually send to the user from your email address.

   c. **Email address** – The email address must be unique.

   d. **Confirm email address**

   e. **First name** – You must enter a name here for automatic provisioning to work. For more information, see Automatic provisioning (p. 47).

   f. **Last name** – You must enter a name here for automatic provisioning to work.

   g. **Display name**

      Note
      (Optional) If applicable, you can specify values for additional attributes such as the user's **Microsoft 365 immutable ID** to help provide the user with single sign-on access to certain business applications.

4. Choose **Next**.

5. If applicable, select one or more groups to which you want to add the user, and choose **Next**.

6. Review the information that you specified for **Step 1: Specify user details** and **Step 2: Add user to groups - optional**. Choose **Edit** by either step to make any changes. After you confirm that the correct information is specified for both steps, choose **Add user**.

   Note
   If you are adding your first administrative user in IAM Identity Center by following the steps to create a user (p. 8), return to that procedure and follow the steps to finish activating the user account.

Add groups

Use the following procedure to add groups to your Identity Center directory. Alternatively, you can call the AWS API operation CreateGroup to add groups.

To add a group

1. Open the IAM Identity Center console.
2. Choose **Groups**.
3. Choose **Create group**.
4. Enter a **Group name** and **Description - *optional***. The description should provide details on what permissions have been or will be assigned to the group. Under **Add users to group - *optional***, locate the users you want to add as members. Then select the check box next to each of them.
5. Choose **Create group**.

After you add this group to your Identity Center directory, you can assign single sign-on access to this group. For more information, see Assign user access to AWS accounts (p. 95).

Add users to groups

Use the following procedure to add users as members of a group that you previously created in your Identity Center directory. Alternatively, you can call the AWS API operation CreateGroupMembership to add a user as a member of a group.

To add a user as a member of a group

1. Open the IAM Identity Center console.

2. Choose **Groups**.

3. Choose the **group name** that you want to update.

4. On the group details page, under **Users in this group**, choose **Add users to group**.

5. On the **Add users to group** page, under **Other users**, locate the users you want to add as members. Then, select the check box next to each of them.

6. Choose **Add users**.

Edit user properties

Use the following procedure to edit the properties of a user in your Identity Center directory. Alternatively, you can call the AWS API operation UpdateUser to update user properties.

To edit user properties

1. Open the IAM Identity Center console.

2. Choose **Users**.

3. Choose the user that you want to edit.

4. On the user **Profile** page, next to **Profile details**, choose **Edit**.

5. On the **Edit profile details** page, update the properties as needed. Then, choose **Save changes**.

   Note
   (Optional) You can modify additional attributes such as **Employee number** and **Office 365 Immutable ID** to help map the user's identity in IAM Identity Center with certain business applications that users need to use.

   Note
   The **Email address** attribute is an editable field and the value you provide must be unique.

Reset a user password

Use the following procedure to reset the password for a user in your Identity Center directory.

To reset a user password

1. Open the IAM Identity Center console.

2. Choose **Users**.

3. Choose the username whose password you want to reset.

4. On the user details page, choose **Reset password**.

5. In the **Reset password** dialog box, select one of the following choices, and then choose **Reset password**:

   a. **Send an email to the user with instructions to reset the password** – This option automatically sends the user an email addressed from Amazon Web Services that walks them through how to reset their password.

      Warning
      As a security best practice, verify that the email address for this user is correct prior to selecting this option. If this password reset email were to be sent to an incorrect or misconfigured email address, a malicious recipient could use it to gain unauthorized access to your AWS environment.

   b. **Generate a one-time password and share the password with the user** – This option provides you with the password details that you can manually send to the user from your email address.

Send email OTP for users created from API

When you create users with the CreateUser API operation, they do not have passwords. You can change this by electing to send users an email one-time password (OTP) when they're created with the API. Users receive the email OTP when they first attempt to sign in. After receiving the email OTP, when a user signs in, they must set a new password. If you don't enable this setting, then you must generate and share OTP with the users that you create using the **CreateUser** API.

To send email OTP to users created with the CreateUser API

1. Open the IAM Identity Center console.
2. Choose **Settings**.
3. On the **Settings** page, choose the **Authentication** tab.
4. In the **Standard authentication** section, choose **Configure**.
5. A dialog box appears. Check the box next to **Send email OTP**. Then, choose **Save**. The status updates from **Disabled** to **Enabled**.

Password requirements when managing identities in IAM Identity Center

> **Note**
> These requirements apply only to users created in the Identity Center directory. If you have configured an identity source other than IAM Identity Center for authentication, such as Active Directory or an external identity provider, the password policies for your users are defined and enforced in those systems, not in IAM Identity Center.

When you use IAM Identity Center as your identity source, users must adhere to the following password requirements to set or change their password:

- Passwords are case-sensitive.
- Passwords must be between 8 and 64 characters in length.
- Passwords must contain at least one character from each of the following four categories:
  - Lowercase letters (a-z)
  - Uppercase letters (A-Z)
  - Numbers (0-9)
  - Non-alphanumeric characters (~!@#$%^&*_-+=`|\(){}[]:;"'<>,.?/)
- The last three passwords cannot be reused.

Connect to a Microsoft AD directory

With AWS IAM Identity Center (successor to AWS Single Sign-On), you can connect a self-managed directory in Active Directory (AD) or a directory in AWS Managed Microsoft AD by using AWS Directory Service. This Microsoft AD directory defines the pool of identities that administrators can pull from when using the IAM Identity Center console to assign single sign-on access. After connecting your corporate directory to IAM Identity Center, you can then grant your AD users or groups access to AWS accounts, cloud applications, or both.

AWS Directory Service helps you to set up and run a standalone AWS Managed Microsoft AD directory hosted in the AWS Cloud. You can also use AWS Directory Service to connect your AWS resources with an existing self-managed AD. To configure AWS Directory Service to work with your self-managed AD, you must first set up trust relationships to extend authentication to the cloud.

IAM Identity Center uses the connection provided by AWS Directory Service to perform pass-through authentication to the source AD instance. When you use AWS Managed Microsoft AD as your identity source, IAM Identity Center can work with users from AWS Managed Microsoft AD or from any domain connected through an AD trust. If you want to locate your users in four or more domains, users must use the `DOMAIN\user` syntax as their user name when performing sign-ins to IAM Identity Center.

Notes

- As a prerequisite step, make sure your AD Connector or directory in AWS Managed Microsoft AD in AWS Directory Service resides within your AWS Organizations management account. For more information, see Active Directory or an external IdP (p. 4).

- IAM Identity Center does not support SAMBA 4-based Simple AD as a connected directory.

Provisioning when users come from Active Directory

IAM Identity Center uses the connection provided by the AWS Directory Service to synchronize user, group, and membership information from your source directory in Active Directory to the IAM Identity Center identity store. No password information is synchronized to IAM Identity Center, since user authentication takes place directly from the source directory in Active Directory. This identity data is used by IAM Identity Center enabled applications to facilitate in-app lookup, authorization, and collaboration scenarios without passing LDAP activity back to the source directory in Active Directory.

For more information above provisioning, see User and group provisioning (p. 13).

Topics

- Connect a directory in AWS Managed Microsoft AD to IAM Identity Center (p. 33)
- Connect a self-managed directory in Active Directory to IAM Identity Center (p. 34)
- Attribute mappings (p. 34)
- Provision users and groups from Active Directory (p. 38)

Connect a directory in AWS Managed Microsoft AD to IAM Identity Center

Use the following procedure to connect a directory in AWS Managed Microsoft AD that is managed by AWS Directory Service to IAM Identity Center.

To connect AWS Managed Microsoft AD to IAM Identity Center

1. Open the IAM Identity Center console.

    Note
    Make sure that the IAM Identity Center console is using one of the Regions where your AWS Managed Microsoft AD directory is located before you move to the next step.

2. Choose **Settings**.

3. On the **Settings** page, choose the **Identity source** tab, and then choose **Actions > Change identity source**.

4. Under **Choose identity source**, select **Active Directory**, and then choose **Next**.

5. Under **Connect active directory**, choose a directory in AWS Managed Microsoft AD from the list, and then choose **Next**.

6. Under **Confirm change**, review the information and when ready type **ACCEPT**, and then choose **Change identity source**.

> **Important**
> To specify a user in Active Directory as an administrative user in IAM Identity Center, you
> must first synchronize the user to whom you want to grant administrative permissions
> from Active Directory into IAM Identity Center. To do so, follow the steps in Synchronize an
> administrative user into IAM Identity Center (p. 8).

Connect a self-managed directory in Active Directory to IAM Identity Center

Users in your self-managed directory in Active Directory (AD) can also have single sign-on access to AWS accounts and cloud applications in the AWS access portal. To configure single sign-on access for these users, you can do either of the following:

- **Create a two-way trust relationship** – When two-way trust relationships are created between AWS Managed Microsoft AD and a self-managed directory in AD, users in your self-managed directory in AD can sign in with their corporate credentials to various AWS services and business applications. One-way trusts do not work with IAM Identity Center.

  AWS IAM Identity Center (successor to AWS Single Sign-On) requires a two-way trust so that it has permissions to read user and group information from your domain to synchronize user and group metadata. IAM Identity Center uses this metadata when assigning access to permission sets or applications. User and group metadata is also used by applications for collaboration, like when you share a dashboard with another user or group. The trust from AWS Directory Service for Microsoft Active Directory to your domain permits IAM Identity Center to trust your domain for authentication. The trust in the opposite direction grants AWS permissions to read user and group metadata.

  For more information about setting up a two-way trust, see When to Create a Trust Relationship in the *AWS Directory Service Administration Guide*.

- **Create an AD Connector** – AD Connector is a directory gateway that can redirect directory requests to your self-managed AD without caching any information in the cloud. For more information, see Connect to a Directory in the *AWS Directory Service Administration Guide*.

  > **Note**
  > If you are connecting IAM Identity Center to an AD Connector directory, any future user
  > password resets must be done from within AD. This means that users will not be able to reset
  > their passwords from the AWS access portal.
  > If you use AD Connector to connect your Active Directory Domain Service to IAM Identity
  > Center, IAM Identity Center only has access to the users and groups of the single domain to
  > which AD Connector attaches. If you need to support multiple domains or forests, use AWS
  > Directory Service for Microsoft Active Directory.

  > **Note**
  > IAM Identity Center does not work with SAMBA4-based Simple AD directories.

Attribute mappings

Attribute mappings are used to map attribute types that exist in IAM Identity Center with like attributes in an AWS Managed Microsoft AD directory. IAM Identity Center retrieves user attributes from your Microsoft AD directory and maps them to IAM Identity Center user attributes. These IAM Identity Center user attribute mappings are also used for generating SAML assertions for your cloud applications. Each cloud application determines the list of SAML attributes it needs for successful single sign-on.

IAM Identity Center prefills a set of attributes for you under the **Attribute mappings** tab found on your application's configuration page. IAM Identity Center uses these user attributes to populate SAML assertions (as SAML attributes) that are sent to the cloud application. These user attributes are in turn

retrieved from your Microsoft AD directory. For more information, see Map attributes in your application to IAM Identity Center attributes (p. 122).

IAM Identity Center also manages a set of attributes for you under the **Attribute mappings** section of your directory configuration page. For more information, see Map attributes in IAM Identity Center to attributes in your AWS Managed Microsoft AD directory (p. 38).

Supported directory attributes

The following table lists all AWS Managed Microsoft AD directory attributes that are supported and that can be mapped to user attributes in IAM Identity Center.

Supported attributes in your Microsoft AD directory
`${dir:email}`
`${dir:displayname}`
`${dir:distinguishedName}`
`${dir:firstname}`
`${dir:guid}`
`${dir:initials}`
`${dir:lastname}`
`${dir:proxyAddresses}`
`${dir:proxyAddresses:smtp}`
`${dir:proxyAddresses:SMTP}`
`${dir:windowsUpn}`

You can specify any combination of supported Microsoft AD directory attributes to map to a single mutable attribute in IAM Identity Center. For example, you can choose the `subject` attribute under the **User attribute in IAM Identity Center** column. Then map it to either `${dir:displayname}` or `${dir:lastname}${dir:firstname }` or any single supported attribute or any arbitrary combination of supported attributes. For a list of the default mappings for user attributes in IAM Identity Center, see Default mappings (p. 37).

> **Note**
> Certain IAM Identity Center attributes can't be modified because they are immutable and mapped by default to specific Microsoft AD directory attributes.

If you use the ListUsers or ListGroups API actions or the list-users and list-groups AWS CLI commands to assign users and groups access to AWS accounts and to applications, you must specify the value for `AttributeValue` as an FQDN. This value must be in the following format: user@example.com. In the following example, `AttributeValue` is set to `janedoe@example.com`.

```
aws identitystore list-users --identity-store-id d-12345a678b --filters
 AttributePath=UserName,AttributeValue=janedoe@example.com
```

Supported IAM Identity Center attributes

The following table lists all IAM Identity Center attributes that are supported and that can be mapped to user attributes in your AWS Managed Microsoft AD directory. After you set up your application attribute

mappings, you can use these same IAM Identity Center attributes to map to actual attributes used by that application.

Supported attributes in IAM Identity Center
`${user:AD_GUID}`
`${user:email}`
`${user:familyName}`
`${user:givenName}`
`${user:middleName}`
`${user:name}`
`${user:preferredUsername}`
`${user:subject}`

Supported external identity provider attributes

The following table lists all external identity provider (IdP) attributes that are supported and that can be mapped to attributes you can use when configuring Attributes for access control (p. 106) in IAM Identity Center. When using SAML assertions, you can use whichever attributes your IdP supports.

Supported attributes in your IdP
`${path:userName}`
`${path:name.familyName}`
`${path:name.givenName}`
`${path:displayName}`
`${path:nickName}`
`${path:emails[primary eq true].value}`
`${path:addresses[type eq "work"].streetAddress}`
`${path:addresses[type eq "work"].locality}`
`${path:addresses[type eq "work"].region}`
`${path:addresses[type eq "work"].postalCode}`
`${path:addresses[type eq "work"].country}`
`${path:addresses[type eq "work"].formatted}`
`${path:phoneNumbers[type eq "work"].value}`
`${path:userType}`
`${path:title}`
`${path:locale}`

Supported attributes in your IdP
`${path:timezone}`
`${path:enterprise.employeeNumber}`
`${path:enterprise.costCenter}`
`${path:enterprise.organization}`
`${path:enterprise.division}`
`${path:enterprise.department}`
`${path:enterprise.manager.value}`

Default mappings

The following table lists the default mappings for user attributes in IAM Identity Center to the user attributes in your AWS Managed Microsoft AD directory. IAM Identity Center only supports the list of attributes in the **User attribute in IAM Identity Center** column.

> **Note**
> If you don't have any assignments for your users and groups in IAM Identity Center when you enable configurable AD sync, the default mappings in the following table are used. For information about how to customize these mappings, see Configure attribute mappings for your sync (p. 43).

User attribute in IAM Identity Center	Maps to this attribute in your Microsoft AD directory
`AD_GUID`	`${dir:guid}`
`email *`	`${dir:windowsUpn}`
`familyName`	`${dir:lastname}`
`givenName`	`${dir:firstname}`
`middleName`	`${dir:initials}`
`name`	`${dir:displayname}`
`preferredUsername`	`${dir:displayname}`
`subject`	`${dir:windowsUpn}`

* The email attribute in IAM Identity Center must be unique within the directory. Otherwise, the JIT login process could fail.

You can change the default mappings or add more attributes to the SAML assertion based on your requirements. For example, assume that your cloud application requires the users email in the `User.Email` SAML attribute. In addition, assume that email addresses are stored in the `windowsUpn` attribute in your Microsoft AD directory. To achieve this mapping, you must make changes in the following two places in the IAM Identity Center console:

1. On the **Directory** page, under the **Attribute mappings** section, you would need to map the user attribute `email` to the `${dir:windowsUpn}` attribute (in the **Maps to this attribute in your directory** column)

2. On the **Applications** page, choose the application from the table. Choose the **Attribute mappings** tab. Then map the `User.Email` attribute to the `${user:email}` attribute (in the **Maps to this string value or user attribute in IAM Identity Center** column).

Note that you must supply each directory attribute in the form ${dir:**AttributeName**}. For example, the `firstname` attribute in your Microsoft AD directory becomes `${dir:firstname}`. It is important that every directory attribute have an actual value assigned. Attributes missing a value after `${dir:` will cause user sign-in issues.

Map attributes in IAM Identity Center to attributes in your AWS Managed Microsoft AD directory

You can use the following procedure to specify how your user attributes in IAM Identity Center should map to corresponding attributes in your Microsoft AD directory.

To map attributes in IAM Identity Center to attributes in your directory

1. Open the IAM Identity Center console.

2. Choose **Settings**.

3. On the **Settings** page, choose the **Identity source** tab, and then choose **Actions > Manage attribute mappings**.

4. On the **Manage attribute mappings** page, find the attribute in IAM Identity Center that you want to map and then type a value in the text box. For example, you might want to map the IAM Identity Center user attribute **email** to the Microsoft AD directory attribute `${dir:windowsUpn}`.

5. Choose **Save changes**.

Provision users and groups from Active Directory

IAM Identity Center provides the following two ways to provision users and groups from Active Directory.

- IAM Identity Center configurable Active Directory (AD) sync (recommended) (p. 38) — With this sync method, you can do the following:
  - Control data boundaries by explicitly defining the users and groups in Microsoft Active Directory that are automatically synchronized into IAM Identity Center. You can add users and groups (p. 42) or remove users and groups (p. 42) to change the scope of the sync at any time.
  - Assign synchronized users and groups single sign-on access to AWS accounts (p. 95) or access to applications (p. 121). The applications can be Identity Center enabled applications, cloud applications, or custom Security Assertion Markup Language (SAML 2.0) applications.
  - Control the synchronization process by pausing and resuming the sync (p. 43) as needed. This helps you regulate the load on production systems.
- IAM Identity Center AD sync (p. 44) — With this sync method, you use IAM Identity Center to assign users and groups in Active Directory access to AWS accounts and to applications. All identities with assignments are automatically synced into IAM Identity Center.

IAM Identity Center configurable AD sync

IAM Identity Center configurable Active Directory (AD) sync enables you to explicitly configure the identities in Microsoft Active Directory that are automatically synchronized into IAM Identity Center and control the synchronization process.

The following topics provide information to enable you to configure and administer configurable AD sync.

Topics

Prerequisites and considerations

Before you use configurable AD sync, be aware of the following prerequisites and considerations:

- **Specifying users and groups in Active Directory to sync**

  Before you can use IAM Identity Center to assign new users and groups access to AWS accounts and to applications [Identity Center enabled applications, cloud applications, or custom Security Assertion Markup Language (SAML 2.0) applications], you must specify the users and groups in Active Directory to sync, and then sync them into IAM Identity Center.

  - **AD sync** – When you make assignments for new users and groups by using the IAM Identity Center console or related assignment API actions, IAM Identity Center searches the domain controller directly for the specified users or groups, completes the assignment, and then periodically syncs the user or group metadata into IAM Identity Center.

  - **Configurable AD sync** – IAM Identity Center doesn't search your domain controller directly for users and groups. Instead, you must first specify the list of users and groups to sync. You can configure this list, also known as the *sync scope*, in one of the following ways, depending on whether you have users and groups that are already synced into IAM Identity Center, or you have new users and groups that you are syncing for the first time by using configurable AD sync.

    - Existing users and groups: If you have users and groups that are already synced into IAM Identity Center, the sync scope in configurable AD sync is prepopulated with a list of those users and groups. To assign new users or groups, you must specifically add them to the sync scope. For more information, see Add users and groups to your sync scope (p. 42).

    - New users and groups: If you want to assign new users and groups access to AWS accounts and to applications, you must specify which users and groups to add to the sync scope in configurable AD sync before you can use IAM Identity Center to make the assignment. For more information, see Add users and groups to your sync scope (p. 42).

- **Making assignments to nested groups in Active Directory**

  Using configurable AD sync to make assignments to a group in Active Directory that contains other groups (nested groups) might increase the scope of users who have access to AWS accounts or to applications.

  - **AD sync** – When you make assignments to a group in Active Directory that contains other groups (nested groups), only the direct members of the group can access the account. For example, if you assign access to Group A, and Group B is a member of Group A, only the direct members of Group A can access the account. No members of Group B inherit the access.

  - **Configurable AD sync** – When you make assignments to a group in Active Directory that contains nested groups, the assignment applies to all users, including those in nested groups. For example, if you assign access to Group A, and Group B is a member of Group A, members of Group B also inherit this access.

- **Updating automated workflows**

  If you have automated workflows that use the IAM Identity Center identity store API actions and IAM Identity Center assignment API actions to assign new users and groups access to accounts and to applications, and to sync them into IAM Identity Center, you must adjust those workflows by April 15,

2022 so that they function as expected with configurable AD sync. Configurable AD sync changes the order in which user and group assignment and provisioning occur, and the way in which queries are performed.

- **AD sync** – The process of assignments occurs first. You assign users and groups access to AWS accounts and to applications. After the users and groups are assigned access, they are automatically provisioned (synced into IAM Identity Center). If you have an automated workflow, this means that when you add a new user to Active Directory, your automated workflow can query Active Directory for the user by using the identity store `ListUser` API action, and then assign the user access by using the IAM Identity Center assignment API actions. Because the user has an assignment, that user is automatically provisioned into IAM Identity Center.

- **Configurable AD sync** – Provisioning occurs first, and it is not automatically performed. Instead, you must first explicitly add users and groups to the identity store by adding them to your sync scope. For information about the recommended steps for automating your sync configuration for configurable AD sync, see .

How configurable AD sync works

IAM Identity Center refreshes the AD-based identity data in the identity store by using the following process.

Creation

After you connect your self-managed directory in Active Directory or your AWS Managed Microsoft AD directory that is managed by AWS Directory Service to IAM Identity Center, you can explicitly configure the Active Directory users and groups that you want to sync into the IAM Identity Center identity store. The identities that you choose will be synchronized every three hours or so into the IAM Identity Center identity store. Depending on the size of your directory, the sync process might take longer.

Groups that are members of other groups (called nested groups or child groups) are also written to the identity store. The nested groups are *flattened*, that is, users in the nested groups are added to the parent group in the IAM Identity Center identity store. This allows you to use the parent group for authorization. Any access that you assign to the parent group applies to all users in the parent group and the users in nested (child) groups.

You can only assign access to new users or groups after they are synchronized into the IAM Identity Center identity store.

Update

The identity data in the IAM Identity Center identity store stays fresh by periodically reading data from the source directory in Active Directory. Identity data changed in AD usually appears in the AWS identity store within four hours, but might take longer based on the amount of data being synchronized.

User and group objects that are in the sync scope and their memberships are created or updated in IAM Identity Center to map to the corresponding objects in the source directory in Active Directory. For user attributes, only the subset of attributes listed in the **Attribute Mapping** section of the IAM Identity Center console are updated in IAM Identity Center.

You can also update the subset of users and groups that you synchronize into the IAM Identity Center identity store. You can choose to add new users or groups to this subset, or remove them. Any identities that you add are synchronized at the next scheduled sync. Identities that you remove from the subset will stop being updated in the IAM Identity Center identity store. Any user who isn't synchronized for more than 28 days will be disabled in the IAM Identity Center identity store. The corresponding user objects will be automatically disabled in the IAM Identity Center identity store during the next sync cycle, unless they are part of another group that is still part of the sync scope.

Deletion

Users and groups are deleted from the IAM Identity Center identity store when the corresponding user or group objects are deleted from the source directory in Active Directory. Alternatively, you can explicitly delete user objects from the IAM Identity Center identity store by using the IAM Identity Center console. If you use the IAM Identity Center console, you must also remove the users from the sync scope to ensure that they aren't re-synced back into IAM Identity Center during the next sync cycle.

You can also pause and restart synchronization at any time. If you pause synchronization for more than 28 days, all your users will be disabled.

Configure and manage your sync scope

You can configure your sync scope in either of the following ways:

- Guided setup: If you are synchronizing your users and groups from Active Directory into IAM Identity Center for the first time, follow the steps in Guided setup (p. 41) to configure your sync scope. After you complete the guided setup, you can modify your sync scope at any time by following the other procedures in this section.

- If you already have users and groups that are synchronized into IAM Identity Center or you don't want to follow the guided setup, choose **Manage sync**. Skip the guided setup procedure and follow the other procedures in this section as required to configure and manage your sync scope.

Procedures

Guided setup

1. Open the IAM Identity Center console.

   Note
   Make sure that the IAM Identity Center console is using one of the AWS Regions where your AWS Managed Microsoft AD directory is located before you move to the next step.

2. Choose **Settings**.

3. At the top of the page, in the notification message, choose **Start guided setup**.

4. In **Step 1 – *optional*: Configure attribute mappings**, review the default user and group attribute mappings. If no changes are required, choose **Next**. If changes are required, make the changes, and then choose **Save changes**.

5. In **Step 2 – *optional*: Configure sync scope**, choose the **Users** tab. Then, enter the exact username of the user that you want to add to your sync scope and choose **Add**. Next, choose the **Groups** tab. Enter the exact group name of the group that you want to add to your sync scope and choose **Add**. Then, choose **Next**. If you want to add users and groups to your sync scope later, make no changes and choose **Next**.

6. In **Step 3: Review and save configuration**, confirm your **Attribute mappings** in **Step 1: Attribute mappings** and your **Users and groups** in **Step 2: Sync scope**. Choose **Save configuration**. This takes you to the **Manage Sync** page.

Add users and groups to your sync scope

To add users

1. Open the IAM Identity Center console.

2. Choose **Settings**.

3. On the **Settings** page, choose the **Identity source** tab, choose **Actions**, and then choose **Manage Sync**.

4. On the **Manage Sync** page, choose the **Users** tab, and then choose **Add users and groups**.

5. On the **Users** tab, under **User**, enter the exact user name and choose **Add**.

6. Under **Added Users and Groups**, review the user that you want to add.

7. Choose **Submit**.

8. In the navigation pane, choose **Users**.

9. On the **Users** page, it might take some time for the user that you specified to appear in the list. Choose the refresh icon to update the list of users.

To add groups

1. Open the IAM Identity Center console.

2. Choose **Settings**.

3. On the **Settings** page, choose the **Identity source** tab, choose **Actions**, and then choose **Manage Sync**.

4. On the **Manage Sync** page, choose the **Groups** tab, and then choose **Add users and groups**.

5. Choose the **Groups** tab. Under **Group**, enter the exact group name and choose **Add**.

6. Under **Added Users and Groups**, review the group that you want to add.

7. Choose **Submit**.

8. In the navigation pane, choose **Groups**.

9. On the **Groups** page, it might take some time for the group that you specified to appear in the list. Choose the refresh icon to update the list of groups.

Remove users and groups from your sync scope

For more information about what happens when you remove users and groups from your sync scope, see How configurable AD sync works (p. 40).

To remove users

1. Open the IAM Identity Center console.

2. Choose **Settings**.

3. On the **Settings** page, choose the **Identity source** tab, choose **Actions**, and then choose **Manage Sync**.

4. Choose the **Users** tab.

5. Under **Users in sync scope**, select the check box beside the user that you want to delete. To delete all users, select the check box beside **Username**.

6. Choose **Remove**.

To remove groups

1. Open the IAM Identity Center console.

2. Choose **Settings**.

3. On the **Settings** page, choose the **Identity source** tab, choose **Actions**, and then choose **Manage Sync**.

4. Choose the **Groups** tab.

5. Under **Groups in sync scope**, select the check box beside the user that you want to delete. To delete all groups, select the check box beside **Group name**.

6. Choose **Remove**.

Pause and resume your sync

Pausing your sync pauses all future sync cycles and prevents any changes that you make to users and groups in Active Directory from being reflected in IAM Identity Center. After you resume the sync, the sync cycle picks up these changes from the next scheduled sync.

To pause your sync

1. Open the IAM Identity Center console.

2. Choose **Settings**.

3. On the **Settings** page, choose the **Identity source** tab, choose **Actions**, and then choose **Manage Sync**.

4. Under **Manage Sync**, choose **Pause sync**.

To resume your sync

1. Open the IAM Identity Center console.

2. Choose **Settings**.

3. On the **Settings** page, choose the **Identity source** tab, choose **Actions**, and then choose **Manage Sync**.

4. Under **Manage Sync**, choose **Resume sync**.

> **Note**
> If you see **Pause sync** instead of **Resume sync**, the sync from Active Directory to IAM Identity Center has already resumed.

Configure attribute mappings for your sync

For more information about available attributes, see Attribute mappings (p. 34).

To configure attribute mappings in IAM Identity Center to your directory

1. Open the IAM Identity Center console.

2. Choose **Settings**.

3. On the **Settings** page, choose the **Identity source** tab, choose **Actions**, and then choose **Manage Sync**.

4. Under **Manage Sync**, choose **View attribute mapping**.

5. Under **Active Directory user attributes**, configure **IAM Identity Center identity store attributes** and **Active Directory user attributes**. For example, you might want to map the IAM Identity Center identity store attribute `email` to the Active Directory user directory attribute `${objectguid}`.

> **Note**
> Under **Group attributes**, **IAM Identity Center identity store attributes** and **Active Directory group attributes** can't be changed.

6. Choose **Save changes**. This returns you to the **Manage Sync** page.

Automate your sync configuration for configurable AD sync

To ensure that your automated workflow works as expected with configurable AD sync, we recommend that you perform the following steps to automate your sync configuration.

To automate your sync configuration for configurable AD sync

1. In Active Directory, create a *parent sync group* to contain all users and groups that you want to sync into IAM Identity Center. For example, you can name the group *IAMIdentityCenterAllUsersAndGroups*.
2. In IAM Identity Center, add the parent sync group to your configurable sync list. IAM Identity Center will synchronize all users, groups, sub-groups, and members of all groups contained within the parent sync group.
3. Use the Active Directory user and group management API actions provided by Microsoft to add or remove users and groups from the parent sync group.

IAM Identity Center AD sync

With IAM Identity Center AD sync, you use IAM Identity Center to assign users and groups in Active Directory access to AWS accounts and to applications AWS accounts and applications [Identity Center enabled applications, cloud applications, or custom Security Assertion Markup Language (SAML 2.0) applications]. All identities with assignments are automatically synced into IAM Identity Center.

How IAM Identity Center AD sync works

IAM Identity Center refreshes the AD-based identity data in the identity store using the following process.

Creation

When you assign users or groups to AWS accounts or applications by using the AWS console or the assignment API calls, information about the users, groups, and membership is periodically synchronized into the IAM Identity Center identity store. Users or groups that are added to IAM Identity Center assignments usually appear in the AWS identity store within two hours. Depending on the amount of data being synchronized, this process might take longer. Only users and groups that are directly assigned access, or are members of a group that is assigned access, are synchronized.

Groups that are members of other groups (called nested groups) are also written to the identity store. The nested groups are "flattened," that is, users in the nested groups are added to the parent group in the IAM Identity Center identity store. This allows you to use only the parent group for authorization.

If a user accesses IAM Identity Center before their user object has been synchronized for the first time, that user's identity store object is created on demand using just-in-time (JIT) provisioning. Users created by JIT provisioning are not synchronized unless they have directly assigned or group-based IAM Identity Center entitlements. Group memberships for JIT-provisioned users are unavailable until after synchronization.

Update

The identity data in the IAM Identity Center identity store stays fresh by periodically reading data from the source directory in Active Directory. Identity data that is changed in Active Directory will usually appear in the AWS identity store within four hours. Depending on the amount of data being synchronized, this process might take longer.

User and group objects and their memberships are created or updated in IAM Identity Center to map to the corresponding objects in the source directory in Active Directory. For user attributes, only the subset

of attributes listed in the **Attribute Mapping** section of the IAM Identity Center console is updated in
IAM Identity Center. In addition, user attributes are updated with each user authentication event.

Deletion

Users and groups are deleted from the IAM Identity Center identity store when the corresponding user or
group objects are deleted from the source directory in Active Directory.

Connect to an external identity provider

If you're using a self-managed directory in Active Directory or an AWS Managed Microsoft AD, see
Connect to a Microsoft AD directory (p. 32). For other external identity providers (IdPs), you can use
AWS IAM Identity Center (successor to AWS Single Sign-On) to authenticate identities from the IdPs
through the Security Assertion Markup Language (SAML) 2.0 standard. This enables your users to sign
in to the AWS access portal with their corporate credentials. They can then navigate to their assigned
accounts, roles, and applications hosted in external IdPs.

For example, you can connect an external IdP such as Okta or Azure Active Directory (AD), to IAM
Identity Center. Your users can then sign in to the AWS access portal with their existing Okta or Azure
credentials. In addition, you can assign access permissions centrally for your users across all the accounts
and applications in your AWS organization. In addition, developers can simply sign in to the AWS
Command Line Interface (AWS CLI) using their existing credentials, and benefit from automatic short-
term credential generation and rotation.

The SAML protocol does not provide a way to query the IdP to learn about users and groups. Therefore,
you must make IAM Identity Center aware of those users and groups by provisioning them into IAM
Identity Center.

Provisioning when users come from an external IdP

When using an external IdP, you must provision all users and groups into IAM Identity Center before
you can make any assignments to AWS accounts or applications. In this case, you can: You can configure
Automatic provisioning (p. 47), or you can configure Manual provisioning (p. 50) of your users
and groups. Regardless of how you provision users, IAM Identity Center redirects the AWS Management
Console, command line interface, and application authentication to your external IdP. IAM Identity
Center then grants access to those resources based on policies you create in IAM Identity Center. For
more information about provisioning, see User and group provisioning (p. 13).

How to connect to an external identity provider

Use the following procedure to connect to an external identity provider from the IAM Identity Center
console.

To connect to an external identity provider

1. Open the IAM Identity Center console.
2. Choose **Settings**.
3. On the **Settings** page, choose the **Identity source** tab, and then choose **Actions > Change identity
    source**.
4. Under **Choose identity source**, select **External identity provider**, and then choose **Next**.
5. Under **Configure external identity provider**, do the following:

    a. Under **Service provider metadata**, choose **Download metadata file** to download the metadata
        file and save it on your system. The IAM Identity Center SAML metadata file is required by your
        external identity provider.

    b.   Under **Identity provider metadata**, choose **Choose file**, and locate the metadata file that you downloaded from your external identity provider. Then upload the file. This metadata file contains the necessary public x509 certificate used to trust messages that are sent from the IdP.

    c.   Choose **Next**.

> **Important**
> Changing your source to or from Active Directory removes all existing user and group assignments. You must manually reapply assignments after you have successfully changed your source.

6.   After you read the disclaimer and are ready to proceed, enter **ACCEPT**.

7.   Choose **Change identity source**.

Topics

SCIM profile and SAML 2.0 implementation

Both SCIM and SAML are important considerations for configuring IAM Identity Center.

SAML 2.0 implementation

IAM Identity Center supports identity federation with SAML (Security Assertion Markup Language) 2.0. This allows IAM Identity Center to authenticate identities from external identity providers (IdPs). SAML 2.0 is an open standard used for securely exchanging SAML assertions. SAML 2.0 passes information about a user between a SAML authority (called an identity provider or IdP), and a SAML consumer (called a service provider or SP). The IAM Identity Center service uses this information to provide federated single sign-on. Single sign-on allows users to access AWS accounts and configured applications based on their existing identity provider credentials (such as a user name and password).

IAM Identity Center adds SAML IdP capabilities to your IAM Identity Center store, AWS Managed Microsoft AD, or to an external identity provider. Users can then single sign-on into services that support SAML, including the AWS Management Console and third-party applications such as Microsoft 365, Concur, and Salesforce.

The SAML protocol however does not provide a way to query the IdP to learn about users and groups. Therefore, you must make IAM Identity Center aware of those users and groups by provisioning them into IAM Identity Center.

SCIM profile

IAM Identity Center provides support for the System for Cross-domain Identity Management (SCIM) v2.0 standard. SCIM keeps your IAM Identity Center identities in sync with identities from your IdP. This includes any provisioning, updates, and deprovisioning of users between your IdP and IAM Identity Center.

For more information about how to implement SCIM, see Automatic provisioning (p. 47). For additional details about IAM Identity Center's SCIM implementation, see the IAM Identity Center SCIM Implementation Developer Guide.

Topics

- Manual provisioning (p. 50)
- Manage SAML 2.0 certificates (p. 50)

Automatic provisioning

IAM Identity Center supports automatic provisioning (synchronization) of user and group information from your identity provider (IdP) into IAM Identity Center using the System for Cross-domain Identity Management (SCIM) v2.0 protocol. When you configure SCIM synchronization, you create a mapping of your identity provider (IdP) user attributes to the named attributes in IAM Identity Center. This causes the expected attributes to match between IAM Identity Center and your IdP. You configure this connection in your IdP using your SCIM endpoint for IAM Identity Center and a bearer token that you create in IAM Identity Center.

Topics

- Considerations for using automatic provisioning (p. 47)
- How to monitor access token expiry (p. 48)
- How to enable automatic provisioning (p. 48)
- How to disable automatic provisioning (p. 49)
- How to generate a new access token (p. 49)
- How to delete an access token (p. 49)
- How to rotate an access token (p. 49)

Considerations for using automatic provisioning

Before you begin deploying SCIM, we recommend that you first review the following important considerations about how it works with IAM Identity Center. For additional provisioning considerations applicable to your IdP, see Supported identity providers (p. 52).

- If you are provisioning a primary email address, this attribute value must be unique for each user. In some IdPs, the primary email address might not be a real email address. For example, it might be a Universal Principal Name (UPN) that only looks like an email. These IdPs may have a secondary or "other" email address that contains the user's real email address. You must configure SCIM in your IdP to map the non-Null unique email address to the IAM Identity Center primary email address attribute. And you must map the users non-Null unique sign-in identifier to the IAM Identity Center user name attribute. Check to see whether your IdP has a single value that is both the sign-in identifier and the user's email name. If so, you can map that IdP field to both the IAM Identity Center primary email and the IAM Identity Center user name.
- For SCIM synchronization to work, every user must have a **First name**, **Last name**, **Username** and **Display name** value specified. If any of these values are missing from a user, that user will not be provisioned.
- The following special characters are invalid when used in attributes that are synchronized with SCIM: <>;:%
- If you need to use third-party applications, you will first need to map the outbound SAML subject attribute to the user name attribute. If the third-party application needs a routable email address, you must provide the email attribute to your IdP.
- SCIM provisioning and update intervals are controlled by your identity provider. Changes to users and groups in your identity provider are only reflected in IAM Identity Center after your identity provider sends those changes to IAM Identity Center. Check with your identity provider for details on the frequency of user and group updates.
- Currently, multivalue attributes (such as multiple emails or phone numbers for a given user) are not provisioned with SCIM. Attempts to synchronize multivalue attributes into IAM Identity Center with SCIM will fail. To avoid failures, ensure that only a single value is passed for each attribute. If you have

users with multivalue attributes, remove or modify the duplicate attribute mappings in SCIM at your IdP for the connection to IAM Identity Center.

- Verify that the `externalId` SCIM mapping at your IdP corresponds to a value that is unique, always present, and least likely to change for your users. For example, your IdP might provide a guaranteed `objectId` or other identifier that's not affected by changes to user attributes like name and email. If so, you can map that value to the SCIM `externalId` field. This ensures that your users won't lose AWS entitlements, assignments, or permissions if you need to change their name or email.

- Users who have not yet been assigned to an application or AWS account cannot be provisioned into IAM Identity Center. To synchronize users and groups, make sure that they are assigned to the application or other setup that represents your IdP's connection to IAM Identity Center.

For more information about IAM Identity Center's SCIM implementation, see the IAM Identity Center SCIM Implementation Developer Guide.

How to monitor access token expiry

SCIM access tokens are generated with a validity of one year. When your SCIM access token is set to expire in 90 days or less, AWS sends you reminders in the IAM Identity Center console and over the AWS Health Dashboard to help you rotate the token. By rotating the SCIM access token before it expires, you continually secure automatic provisioning of user and group information. If the SCIM access token expires, the synchronization of user and group information from your identity provider into IAM Identity Center stops, so automatic provisioning can no longer make updates or create and delete information. Disruption to automatic provisioning may impose increased security risks and impact access to your services.

The Identity Center console reminders persist until you rotate the SCIM access token. The AWS Health Dashboard events are renewed weekly between 90 to 60 days, twice per week from 60 to 30 days, three times per week from 30 to 15 days, and daily from 15 days until the SCIM access tokens expires.

How to enable automatic provisioning

Use the following procedure to enable automatic provisioning of users and groups from your IdP to IAM Identity Center using the SCIM protocol.

> **Note**
> Before you begin this procedure, we recommend that you first review provisioning considerations that are applicable to your IdP. For more information, see Supported identity providers (p. 52).

To enable automatic provisioning in IAM Identity Center

1. After you have completed the prerequisites, open the IAM Identity Center console.
2. Choose **Settings** in the left navigation pane.
3. On the **Settings** page, locate the **Automatic provisioning** information box, and then choose **Enable**. This immediately enables automatic provisioning in IAM Identity Center and displays the necessary SCIM endpoint and access token information.
4. In the **Inbound automatic provisioning** dialog box, copy each of the values for the following options. You will need to paste these in later when you configure provisioning in your IdP.

   a. **SCIM endpoint**
   b. **Access token**
5. Choose **Close**.

After you complete this procedure, you must configure automatic provisioning in your IdP. For more information, see Supported identity providers (p. 52).

How to disable automatic provisioning

Use the following procedure to disable automatic provisioning in the IAM Identity Center console.

> **Important**
> You must delete the access token before you start this procedure. For more information, see
> How to delete an access token (p. 49).

To disable automatic provisioning in the IAM Identity Center console

1. In the IAM Identity Center console, choose **Settings** in the left navigation pane.
2. On the **Settings** page, choose the **Identity source** tab, and then choose **Actions > Manage provisioning**.
3. On the **Automatic provisioning** page, choose **Disable**.
4. In the **Disable automatic provisioning** dialog box, review the information, type **DISABLE**, and then choose **Disable automatic provisioning**.

How to generate a new access token

Use the following procedure to generate a new access token in the IAM Identity Center console.

> **Note**
> This procedure requires that you have previously enabled automatic provisioning. For more
> information, see How to enable automatic provisioning (p. 48).

To generate a new access token

1. In the IAM Identity Center console, choose **Settings** in the left navigation pane.
2. On the **Settings** page, choose the **Identity source** tab, and then choose **Actions > Manage provisioning**.
3. On the **Automatic provisioning** page, under **Access tokens**, choose **Generate token**.
4. In the **Generate new access token** dialog box, copy the new access token and save it in a safe place.
5. Choose **Close**.

How to delete an access token

Use the following procedure to delete an existing access token in the IAM Identity Center console.

To to delete an existing access token

1. In the IAM Identity Center console, choose **Settings** in the left navigation pane.
2. On the **Settings** page, choose the **Identity source** tab, and then choose **Actions > Manage provisioning**.
3. On the **Automatic provisioning** page, under **Access tokens**, select the access token you want to delete, and then choose **Delete**.
4. In the **Delete access token** dialog box, review the information, type **DELETE**, and then choose **Delete access token**.

How to rotate an access token

If your SCIM access token is close to expiring, you can use the following procedure to rotate an existing access token in the IAM Identity Center console.

To rotate an access token

1. In the IAM Identity Center console, choose **Settings** in the left navigation pane.
2. On the **Settings** page, choose the **Identity source** tab, and then choose **Actions > Manage provisioning**.
3. On the **Automatic provisioning** page, under **Access tokens**, make a note of the token ID of the token you want to rotate.
4. Follow the steps in How to generate a new access token (p. 49) to create a new token. If you have already created the maximum number of SCIM access tokens, you will first need to delete one of the existing tokens.
5. Go to your identity provider's website and configure the new access token for SCIM provisioning, and then test connectivity to IAM Identity Center using the new SCIM access token. Once you've confirmed that provisioning is working successfully using the new token, continue to the next step in this procedure.
6. Follow the steps in How to delete an access token (p. 49) to delete the old access token you noted earlier. You can also use the token's creation date as a hint for which token to remove.

Manual provisioning

Some IdPs do not have System for Cross-domain Identity Management (SCIM) support or have an incompatible SCIM implementation. In those cases, you can manually provision users through the IAM Identity Center console. When you add users to IAM Identity Center, ensure that you set the user name to be identical to the user name that you have in your IdP. At a minimum, you must have a unique email address and user name. For more information, see User name and email address uniqueness (p. 13).

You must also manage all groups manually in IAM Identity Center. To do this, you create the groups and add them using the IAM Identity Center console. These groups do not need to match what exists in your IdP. For more information, see Groups (p. 13).

Manage SAML 2.0 certificates

IAM Identity Center uses certificates to set up a SAML trust relationship between IAM Identity Center and your external identity provider (IdP). When you add an external IdP in IAM Identity Center, you must also obtain at least one public SAML 2.0 X.509 certificate from the external IdP. That certificate is usually installed automatically during the IdP SAML metadata exchange during trust creation.

As an IAM Identity Center administrator, you'll occasionally need to replace older IdP certificates with newer ones. For example, you might need to replace an IdP certificate when the expiration date on the certificate approaches. The process of replacing an older certificate with a newer one is referred to as certificate rotation.

Topics
- Rotate a SAML 2.0 certificate (p. 50)
- Certificate expiration status indicators (p. 52)

Rotate a SAML 2.0 certificate

You may need to import certificates periodically in order to rotate invalid or expired certificates issued by your identity provider. This helps to prevent authentication disruption or downtime. All imported certificates are automatically active. Certificates should only be deleted after ensuring that they are no longer in use with the associated identity provider.

You should also consider that some IdPs might not support multiple certificates. In this case, the act of rotating certificates with these IdPs might mean a temporary service disruption for your users. Service is

restored when the trust with that IdP has been successfully reestablished. Plan this operation carefully during off peak hours if possible.

> **Note**
> As a security best practice, upon any signs of compromise or mishandling of an existing SAML certificate, you should immediately remove and rotate the certificate.

Rotating an IAM Identity Center certificate is a multistep process that involves the following:

- Obtaining a new certificate from the IdP
- Importing the new certificate into IAM Identity Center
- Activating the new certificate in the IdP
- Deleting the older certificate

Use all of the following procedures to complete the certificate rotation process while avoiding any authentication downtime.

Step 1: Obtain a new certificate from the IdP

Go to the IdP website and download their SAML 2.0 certificate. Make sure that the certificate file is downloaded in PEM encoded format. Most providers allow you to create multiple SAML 2.0 certificates in the IdP. It is likely that these will be marked as disabled or inactive.

Step 2: Import the new certificate into IAM Identity Center

Use the following procedure to import the new certificate using the IAM Identity Center console.

1. In the IAM Identity Center console, choose **Settings**.
2. On the **Settings** page, choose the **Identity source** tab, and then choose **Actions > Manage authentication**.
3. On the **Manage SAML 2.0 certificates** page, choose **Import certificate**.
4. On the **Import SAML 2.0 certificate** dialog, choose **Choose file**, navigate to your certificate file and select it, and then choose **Import certificate**.

At this point, IAM Identity Center will trust all incoming SAML messages signed from both of the certificates that you have imported.

Step 3: Activate the new certificate in the IdP

Go back to the IdP website and mark the new certificate that you created earlier as primary or active. At this point all SAML messages signed by the IdP should be using the new certificate.

Step 4: Delete the old certificate

Use the following procedure to complete the certificate rotation process for your IdP. There must always be at least one valid certificate listed, and it cannot be removed.

> **Note**
> Make sure that your identity provider is no longer signing SAML responses with this certificate before deleting it.

1. On the **Manage SAML 2.0 certificates** page, choose the certificate that you want to delete. Choose **Delete**.
2. In the **Delete SAML 2.0 certificate** dialog box, type **DELETE** to confirm, and then choose **Delete**.
3. Return to the IdP's website and perform the necessary steps to remove the older inactive certificate.

Certificate expiration status indicators

While on the **Manage SAML 2.0 certificates** page, you might notice colored status indicator icons. These icons appear in the **Expires on** column next to each certificate in the list. The following describes the criteria that IAM Identity Center uses to determine which icon is displayed for each certificate.

- **Red** – Indicates that a certificate is currently expired.
- **Yellow** – Indicates that a certificate will expire in 90 days or less.
- **Green** – Indicates that a certificate is currently valid and will remain valid for at least 90 more days.

To check the current status of a certificate

1. In the IAM Identity Center console, choose **Settings**.
2. On the **Settings** page, choose the **Identity source** tab, and then choose **Actions > Manage authentication**.
3. On the **Manage SAML 2.0 authentication** page, under **Manage SAML 2.0 certificates**, review the status of the certificates in the list as indicated in the **Expires on** column.

Supported identity providers

The following external identity providers have been tested with the IAM Identity Center SCIM implementation.

Topics
- Azure AD (p. 52)
- CyberArk (p. 55)
- JumpCloud (p. 58)
- Okta (p. 60)
- OneLogin (p. 65)
- Ping Identity (p. 68)

Azure AD

IAM Identity Center supports automatic provisioning (synchronization) of user and group information from Azure AD into IAM Identity Center using the System for Cross-domain Identity Management (SCIM) v2.0 protocol. You configure this connection in Azure AD using your SCIM endpoint for IAM Identity Center and a bearer token that is created automatically by IAM Identity Center. When you configure SCIM synchronization, you create a mapping of your user attributes in Azure AD to the named attributes in IAM Identity Center. This causes the expected attributes to match between IAM Identity Center and your IdP.

The following steps walk you through how to enable automatic provisioning of users and groups from Azure AD to IAM Identity Center using the IAM Identity Center app in the Azure AD Application Gallery and the SCIM protocol.

> **Note**
> Before you begin deploying SCIM, we recommend that you first review Considerations for using automatic provisioning (p. 47), and then continue reviewing Prerequisites (p. 52) and Additional considerations (p. 53) in the next sections.

Prerequisites

You will need the following before you can get started:

- An Azure AD tenant. For more information, see Quickstart: Set up a tenant on Microsoft's website.

- An IAM Identity Center-enabled account (free). For more information, see Enable IAM Identity Center.
- A SAML connection from your Azure AD account to IAM Identity Center, as described in Tutorial: Azure Active Directory single sign-on (SSO) integration with IAM Identity Center on Microsoft's website.

  > **Important**
  > Make sure that all users in Azure AD have filled out **First name**, **Last name**, and **Display name** values in their user properties. Otherwise, automatic provisioning won't work with Azure AD.

Additional considerations

Attributes for access control are used in permission policies that determine who in your identity source can access your AWS resources. If an attribute is removed from a user in Azure AD, that attribute will not be removed from the corresponding user in IAM Identity Center. This is a known limitation in Azure AD. If an attribute is changed to a different (non-empty) value on a user, that change will be synchronized to IAM Identity Center.

Step 1: Set up IAM Identity Center and configure automatic provisioning

To get started, you'll need to first follow the instructions in Tutorial: Configure AWS Single Sign-On for automatic user provisioning. These instructions walk you through the following:

- Enable IAM Identity Center.
- Install the IAM Identity Center app from the Azure AD Application Gallery.
- Configure automatic provisioning (SCIM) within the Azure portal.

Step 2: (Optional) Configure attribute-based access control

Now that you have configured Azure AD to work with IAM Identity Center, you can optionally choose to configure attribute-based access control (ABAC). ABAC is an authorization strategy that defines permissions based on attributes.

With Azure AD, you have two different ways to configure ABAC for use with IAM Identity Center. Choose either of the following methods.

Method 1: Configure ABAC using Azure AD

This method can be used when you need to define which attributes in Azure AD can be used by IAM Identity Center to manage access to your AWS resources. Once defined, Azure AD sends these attributes to IAM Identity Center through SAML assertions. You will then need to Create a permission set (p. 98) in IAM Identity Center to manage access based on the attributes you passed from Azure AD.

Before you begin this procedure, you first need to enable the Attributes for access control (p. 106) feature. For more information about how to do this, see Enable and configure attributes for access control (p. 107).

To configure user attributes in Azure AD for access control in IAM Identity Center

1. While signed into the Azure portal, navigate to **Azure Active Directory**, **Enterprise applications**. Search for the name of the application that you created previously to form your SAML connection. Then choose the application.
2. Choose **Single sign-on**.
3. In the **User Attributes & Claims** section, choose **Edit**.
4. On the **User Attributes & Claims** page, do the following:

   a. Choose **Add new claim**

b. For **Name**, enter `AccessControl:`**`AttributeName`**. Replace **`AttributeName`** with the name of the attribute you are expecting in IAM Identity Center. For example, `AccessControl:`**`Department`**.

c. For **Namespace**, enter **https://aws.amazon.com/SAML/Attributes**.

d. For **Source**, choose **Attribute**.

e. For **Source attribute**, use the drop-down list to choose the Azure AD user attributes. For example, `user.`**`department`**.

5. Repeat the previous step for each attribute you need to send to IAM Identity Center in the SAML assertion.

6. Choose **Save**.

Method 2: Configure ABAC using IAM Identity Center

With this method, you use the Attributes for access control (p. 106) feature in IAM Identity Center to pass an `Attribute` element with the `Name` attribute set to `https://aws.amazon.com/SAML/Attributes/AccessControl:`**`{TagKey}`**. You can use this element to pass attributes as session tags in the SAML assertion. For more information about session tags, see Passing session tags in AWS STS in the *IAM User Guide*.

To pass attributes as session tags, include the `AttributeValue` element that specifies the value of the tag. For example, to pass the tag key-value pair `CostCenter = blue`, use the following attribute:

```
<saml:AttributeStatement>
<saml:Attribute Name="https://aws.amazon.com/SAML/Attributes/AccessControl:CostCenter">
<saml:AttributeValue>blue
</saml:AttributeValue>
</saml:Attribute>
</saml:AttributeStatement>
```

If you need to add multiple attributes, include a separate `Attribute` element for each tag.

Troubleshooting

The following can help you troubleshoot some common issues you might encounter while setting up automatic provisioning with Azure AD.

Azure AD users are not synchronizing to IAM Identity Center

This might be due to a syntax issue that IAM Identity Center has flagged when a new user is being added to IAM Identity Center. You can confirm this by checking the Azure audit logs for failed events, such as an `'Export'`. The **Status Reason** for this event will state:

```
{"schema":["urn:ietf:params:scim:api:messages:2.0:Error"],"detail":"Request is unparsable,
 syntactically incorrect, or violates schema.","status":"400"}
```

You can also check AWS CloudTrail for the failed event. This can be done by searching in the **Event History** console of CloudTrail using the following filter:

```
"eventName":"CreateUser"
```

The error in the CloudTrail event will state the following:

```
"errorCode": "ValidationException",
        "errorMessage": "Currently list attributes only allow single item"
```

Ultimately, this exception means that one of the values passed from Azure contained more values than anticipated. The solution here is to review the attributes of the user in Azure AD, ensuring that none contain duplicate values. One common example of duplicate values is having multiple values present for contact numbers such as **mobile**, **work**, and **fax**. Although separate values, they are all passed to IAM Identity Center under the single parent attribute **phoneNumbers**.

CyberArk

IAM Identity Center supports automatic provisioning (synchronization) of user information from CyberArk Directory Platform into IAM Identity Center. This provisioning uses the System for Cross-domain Identity Management (SCIM) v2.0 protocol. You configure this connection in CyberArk using your IAM Identity Center SCIM endpoint and access token. When you configure SCIM synchronization, you create a mapping of your user attributes in CyberArk to the named attributes in IAM Identity Center. This causes the expected attributes to match between IAM Identity Center and CyberArk.

This guide is based on CyberArk as of August 2021. Steps for newer versions may vary. This guide contains a few notes regarding configuration of user authentication through SAML.

> **Note**
> Before you begin deploying SCIM, we recommend that you first review the Considerations for using automatic provisioning (p. 47). Then continue reviewing additional considerations in the next section.

Topics

Prerequisites

You will need the following before you can get started:

- CyberArk subscription or free trial. To sign up for a free trial visit CyberArk.
- An IAM Identity Center enabled account (free). For more information, see Enable IAM Identity Center.
- A SAML connection from your CyberArk account to IAM Identity Center, as described in CyberArk documentation for IAM Identity Center.
- Associate the IAM Identity Center connector with the roles, users and organizations you want to allow access to AWS accounts.

SCIM considerations

The following are considerations when using CyberArk federation for IAM Identity Center:

- Only roles mapped in the application Provisioning section will be synchronized to IAM Identity Center.
- The provisioning script is supported only in its default state, once changed the SCIM provisioning might fail.
  - Only one phone number attribute can be synchronized and the default is "work phone".
- If the role mapping in CyberArk IAM Identity Center application is changed, the below behavior is expected:

- If the role names are changed - no changes to the group names in IAM Identity Center.

- If the group names are changed - new groups will be created in IAM Identity Center, old groups will remain but will have no members.

- User synchronization and de-provisioning behavior can be set up from the CyberArk IAM Identity Center application, make sure you set up the right behavior for your organization. These are the options you have:

  - Overwrite (or not) users in Identity Center directory with the same principal name.

  - De-provision users from IAM Identity Center when the user is removed from the CyberArk role.

  - De-provision user behavior - disable or delete.

Step 1: Enable provisioning in IAM Identity Center

In this first step, you use the IAM Identity Center console to enable automatic provisioning.

To enable automatic provisioning in IAM Identity Center

1. After you have completed the prerequisites, open the IAM Identity Center console.
2. Choose **Settings** in the left navigation pane.
3. On the **Settings** page, locate the **Automatic provisioning** information box, and then choose **Enable**. This immediately enables automatic provisioning in IAM Identity Center and displays the necessary SCIM endpoint and access token information.
4. In the **Inbound automatic provisioning** dialog box, copy each of the values for the following options. You will need to paste these in later when you configure provisioning in your IdP.

   a. **SCIM endpoint**
   b. **Access token**
5. Choose **Close**.

Now that you have set up provisioning in the IAM Identity Center console, you need to complete the remaining tasks using the CyberArk IAM Identity Center application. These steps are described in the following procedure.

Step 2: Configure provisioning in CyberArk

Use the following procedure in the CyberArk IAM Identity Center application to enable provisioning with IAM Identity Center. This procedure assumes that you have already added the CyberArk IAM Identity Center application to your CyberArk admin console under **Web Apps**. If you have not yet done so, refer to the Prerequisites (p. 55), and then complete this procedure to configure SCIM provisioning.

To configure provisioning in CyberArk

1. Open the CyberArk IAM Identity Center application that you added as part of configuring SAML for CyberArk (**Apps > Web App**). See Prerequisites (p. 55).
2. Choose the **IAM Identity Center** application and go to the **Provisioning** section.
3. Check the box for **Enable provisioning for this application** and choose **Live Mode**.
4. In the previous procedure, you copied the **SCIM endpoint** value from IAM Identity Center. Paste that value into the **SCIM Service URL** field, in the CyberArk's IAM Identity Center's application set the **Authorization Type** to be **Authorization Header**. Make sure that you remove the trailing forward slash at the end of the URL.
5. Set the **Header Type** to **Bearer Token**.
6. From the previous procedure you copied the **Access token** value in IAM Identity Center. Paste that value into the **Bearer Token** field in the CyberArk IAM Identity Center application.

7. Click **Verify** to test and apply the configuration.

8. Under the **Sync Options**, choose the right behavior for which you want the outbound provisioning from CyberArk to work. You can choose to overwrite (or not) existing IAM Identity Center users with similar principal name, and the de-provisioning behavior.

9. Under **Role Mapping** set up the mapping from CyberArk roles, under the **Name** field to the IAM Identity Center group, under the **Destination Group**.

10. Click **Save** at the bottom once you are done.

11. To verify that users have been successfully synchronized to IAM Identity Center, return to the IAM Identity Center console and choose **Users**. Synchronized users from CyberArk will appear on the **Users** page. These users can now be assigned to accounts and can connect within IAM Identity Center.

(Optional) Step 3: Configure user attributes in CyberArk for access control (ABAC) in IAM Identity Center

This is an optional procedure for CyberArk should you choose to configure attributes for IAM Identity Center to manage access to your AWS resources. The attributes that you define in CyberArk will be passed in a SAML assertion to IAM Identity Center. You then create a permission set in IAM Identity Center to manage access based on the attributes you passed from CyberArk.

Before you begin this procedure, you must first enable the Attributes for access control (p. 106) feature. For more information about how to do this, see Enable and configure attributes for access control (p. 107).

To configure user attributes in CyberArk for access control in IAM Identity Center

1. Open the CyberArk IAM Identity Center application that you installed as part of configuring SAML for CyberArk (**Apps > Web Apps**).

2. Go to the **SAML Response** option.

3. Under **Attributes**, add the relevant attributes to the table following the below logic:

   a. **Attribute Name** is the original attribute name from CyberArk.

   b. **Attribute Value** is the attribute name sent in the SAML assertion to IAM Identity Center.

4. Choose **Save**.

(Optional) Passing attributes for access control

You can optionally use the Attributes for access control (p. 106) feature in IAM Identity Center to pass an `Attribute` element with the `Name` attribute set to `https://aws.amazon.com/SAML/Attributes/AccessControl:{TagKey}`. This element allows you to pass attributes as session tags in the SAML assertion. For more information about session tags, see Passing session tags in AWS STS in the *IAM User Guide*.

To pass attributes as session tags, include the `AttributeValue` element that specifies the value of the tag. For example, to pass the tag key-value pair `CostCenter = blue`, use the following attribute.

```
<saml:AttributeStatement>
<saml:Attribute Name="https://aws.amazon.com/SAML/Attributes/AccessControl:CostCenter">
<saml:AttributeValue>blue
</saml:AttributeValue>
</saml:Attribute>
</saml:AttributeStatement>
```

If you need to add multiple attributes, include a separate `Attribute` element for each tag.

JumpCloud

IAM Identity Center supports automatic provisioning (synchronization) of user information from JumpCloud Directory Platform into IAM Identity Center. This provisioning uses the System for Cross-domain Identity Management (SCIM) v2.0 protocol. You configure this connection in JumpCloud using your IAM Identity Center SCIM endpoint and access token. When you configure SCIM synchronization, you create a mapping of your user attributes in JumpCloud to the named attributes in IAM Identity Center. This causes the expected attributes to match between IAM Identity Center and JumpCloud.

This guide is based on JumpCloud as of June 2021. Steps for newer versions may vary. This guide contains a few notes regarding configuration of user authentication through SAML.

The following steps walk you through how to enable automatic provisioning of users and groups from JumpCloud to IAM Identity Center using the SCIM protocol.

> **Note**
> Before you begin deploying SCIM, we recommend that you first review the Considerations for using automatic provisioning (p. 47). Then continue reviewing additional considerations in the next section.

Topics

Prerequisites

You will need the following before you can get started:

- JumpCloud subscription or free trial. To sign up for a free trial visit JumpCloud.
- An IAM Identity Center enabled account (free). For more information, see Enable IAM Identity Center.
- A SAML connection from your JumpCloud account to IAM Identity Center, as described in JumpCloud documentation for IAM Identity Center .
- Associate the IAM Identity Center connector with the groups you want to allow access to AWS accounts.

SCIM considerations

The following are considerations when using JumpCloud federation for IAM Identity Center.

- Only groups associated with the AWS Single Sign-On connector in JumpCloud will be synchronized via SCIM.
- Only one phone number attribute can be synchronized and the default is "work phone."
- Users in JumpCloud directory must have first and last names configured to be synchronized to IAM Identity Center via SCIM.
- Attributes are still synchronized if the user is disabled in IAM Identity Center but still activate in JumpCloud.
- You can choose to enable SCIM sync for only user information by unchecking the "Enable management of User Groups and Group membership" in the connector.

- If there is an existing user in Identity Center directory with the same username and email, the user will be overwritten and synchronized via SCIM from JumpCloud.

Step 1: Enable provisioning in IAM Identity Center

In this first step, you use the IAM Identity Center console to enable automatic provisioning.

To enable automatic provisioning in IAM Identity Center

1. After you have completed the prerequisites, open the IAM Identity Center console.
2. Choose **Settings** in the left navigation pane.
3. On the **Settings** page, locate the **Automatic provisioning** information box, and then choose **Enable**. This immediately enables automatic provisioning in IAM Identity Center and displays the necessary SCIM endpoint and access token information.
4. In the **Inbound automatic provisioning** dialog box, copy each of the values for the following options. You will need to paste these in later when you configure provisioning in your IdP.

   a. **SCIM endpoint**
   b. **Access token**
5. Choose **Close**.

Now that you have set up provisioning in the IAM Identity Center console, you need to complete the remaining tasks using the JumpCloud IAM Identity Center connector. These steps are described in the following procedure.

Step 2: Configure provisioning in JumpCloud

Use the following procedure in the JumpCloud IAM Identity Center connector to enable provisioning with IAM Identity Center. This procedure assumes that you have already added the JumpCloud IAM Identity Center connector to your JumpCloud admin portal and groups. If you have not yet done so, refer to Prerequisites (p. 58), and then complete this procedure to configure SCIM provisioning.

To configure provisioning in JumpCloud

1. Open the JumpCloud IAM Identity Center connector that you installed as part of configuring SAML for JumpCloud (**User Authentication** > **IAM Identity Center**). See Prerequisites (p. 58).
2. Choose the **IAM Identity Center** connector, and then choose the third tab **Identity Management**.
3. Check the box for **Enable management of User Groups and Group membership in this application** if you want groups to SCIM sync.
4. Click on **Configure**.
5. In the previous procedure, you copied the **SCIM endpoint** value in IAM Identity Center. Paste that value into the **Base URL** field in the JumpCloud IAM Identity Center connector. Make sure that you remove the trailing forward slash at the end of the URL.
6. From the previous procedure you copied the **Access token** value in IAM Identity Center. Paste that value into the **Token Key** field in the JumpCloud IAM Identity Center connector.
7. Click **Activate** to apply the configuration.
8. Make sure you have a green indicator next to **Single Sign-On activated**.
9. Move to the fourth tab **User Groups** and check the groups you want to be provisioned via SCIM.
10. Click **Save** at the bottom once you are done.
11. To verify that users have been successfully synchronized to IAM Identity Center, return to the IAM Identity Center console and choose **Users**. Synchronized users from JumpCloud will appear on the **Users** page. These users can now be assigned to accounts within IAM Identity Center.

(Optional) Step 3: Configure user attributes in JumpCloud for access control in IAM Identity Center

This is an optional procedure for JumpCloud should you choose to configure attributes for IAM Identity Center to manage access to your AWS resources. The attributes that you define in JumpCloud will be passed in a SAML assertion to IAM Identity Center. You then create a permission set in IAM Identity Center to manage access based on the attributes you passed from JumpCloud.

Before you begin this procedure, you must first enable the Attributes for access control feature. For more information about how to do this, see Enable and configure attributes for access control.

To configure user attributes in JumpCloud for access control in IAM Identity Center

1. Open the JumpCloud IAM Identity Center connector that you installed as part of configuring SAML for JumpCloud (**User Authentication** > **IAM Identity Center**).
2. Choose the **IAM Identity Center** connector. Then, choose the second tab **IAM Identity Center**.
3. At the bottom of this tab you have **User Attribute Mapping**, choose **Add new attribute**, and then do the following: You must perform these steps for each attribute you will add for use in IAM Identity Center for access control.

    a. In the **Service Provide Attribute Name** field, enter `https://aws.amazon.com/SAML/Attributes/AccessControl:`**`AttributeName`**. Replace **`AttributeName`** with the name of the attribute you are expecting in IAM Identity Center. For example, `https://aws.amazon.com/SAML/Attributes/AccessControl:`**`Email`**.

    b. In the **JumpCloud Attribute Name** field, choose user attributes from your JumpCloud directory. For example, **Email (Work)**.
4. Choose **Save**.

(Optional) Passing attributes for access control

You can optionally use the Attributes for access control (p. 106) feature in IAM Identity Center to pass an `Attribute` element with the `Name` attribute set to `https://aws.amazon.com/SAML/Attributes/AccessControl:`**`{TagKey}`**. This element allows you to pass attributes as session tags in the SAML assertion. For more information about session tags, see Passing session tags in AWS STS in the *IAM User Guide*.

To pass attributes as session tags, include the `AttributeValue` element that specifies the value of the tag. For example, to pass the tag key-value pair `CostCenter = blue`, use the following attribute.

```
<saml:AttributeStatement>
<saml:Attribute Name="https://aws.amazon.com/SAML/Attributes/AccessControl:CostCenter">
<saml:AttributeValue>blue
</saml:AttributeValue>
</saml:Attribute>
</saml:AttributeStatement>
```

If you need to add multiple attributes, include a separate `Attribute` element for each tag.

Okta

IAM Identity Center supports automatic provisioning (synchronization) of user and group information from Okta into IAM Identity Center using the System for Cross-domain Identity Management (SCIM) v2.0 protocol. To configure this connection in Okta, you use your SCIM endpoint for IAM Identity Center and a bearer token that is created automatically by IAM Identity Center. When you configure SCIM synchronization, you create a mapping of your user attributes in Okta to the named attributes in IAM Identity Center. This causes the expected attributes to match between IAM Identity Center and your IdP.

Okta supports the following provisioning features when connected to IAM Identity Center through SCIM:

- Create users – Users assigned to the IAM Identity Center application in Okta will be provisioned in IAM Identity Center.
- Update user attributes – Attribute changes for users who are assigned to the IAM Identity Center application in Okta will be updated in IAM Identity Center.
- Deactivate users – Users who are unassigned from the IAM Identity Center application in Okta will be disabled in IAM Identity Center.
- Group push – Groups (and their members) in Okta are synchronized to IAM Identity Center.

The following steps walk you through how to enable automatic provisioning of users and groups from Okta to IAM Identity Center using the SCIM protocol.

> **Note**
> Before you begin deploying SCIM, we recommend that you first review the Considerations for using automatic provisioning (p. 47). Then continue reviewing additional considerations in the next section.

Topics

Additional considerations

The following are important considerations about Okta that can affect how you implement provisioning with IAM Identity Center.

- Using the same Okta group for both assignments and group push is not currently supported. To maintain consistent group memberships between Okta and IAM Identity Center, you need to create a separate group and configure it to push groups to IAM Identity Center.
- If you update a user's address you must have **streetAddress**, **city**, **state**, **zipCode** and the **countryCode** value specified. If any of these values are not specified for the Okta user at the time of synchronization, the user or changes to the user will not be provisioned.
- Entitlements and role attributes are not supported and cannot be synced to IAM Identity Center.

Prerequisites

You will need the following before you can get started:

- An Okta account (free trial) with Okta's IAM Identity Center application installed. Note also that for paid Okta products, you might need to confirm that your Okta license supports "lifecycle management" or similar capabilities that enable outbound provisioning. These features might be necessary to configure SCIM from Okta to IAM Identity Center.
- A SAML connection from your Okta account to IAM Identity Center, as described in How to Configure SAML 2.0 for IAM Identity Center.

- An IAM Identity Center-enabled account (free). For more information, see Enable IAM Identity Center.

Step 1: Enable provisioning in IAM Identity Center

In this first step, you use the IAM Identity Center console to enable automatic provisioning.

To enable automatic provisioning in IAM Identity Center

1. After you have completed the prerequisites, open the IAM Identity Center console.
2. Choose **Settings** in the left navigation pane.
3. On the **Settings** page, locate the **Automatic provisioning** information box, and then choose **Enable**. This immediately enables automatic provisioning in IAM Identity Center and displays the necessary SCIM endpoint and access token information.
4. In the **Inbound automatic provisioning** dialog box, copy each of the values for the following options. You will need to paste these in later when you configure provisioning in your IdP.

   a. **SCIM endpoint**
   b. **Access token**
5. Choose **Close**.

You have set up provisioning in the IAM Identity Center console. Now you need to do the remaining tasks using the Okta user interface as described in the following procedures.

Step 2: Configure provisioning in Okta

Use the following procedure in the Okta admin portal to enable integration between IAM Identity Center and the IAM Identity Center app.

To configure provisioning in Okta

1. In a separate browser window, log in to the Okta admin portal and navigate to the IAM Identity Center app.
2. On the **IAM Identity Center app** page, choose the **Provisioning** tab, and then choose **Integration**.
3. Choose **Configure API Integration**, and then select the check box next to **Enable API integration** to enable provisioning.
4. In the previous procedure you copied the **SCIM endpoint** value in IAM Identity Center. Paste that value into the **Base URL** field in Okta. Make sure that you remove the trailing forward slash at the end of the URL. Also, in the previous procedure you copied the **Access token** value in IAM Identity Center. Paste that value into the **API Token** field in Okta.
5. Choose **Test API Credentials** to verify the credentials entered are valid.
6. Choose **Save**.
7. Under **Settings**, choose **To App**, choose **Edit**, and then select the **Enable** check box for each of the **Provisioning Features** you want to enable.
8. Choose **Save**.

By default, no users or groups are assigned to your Okta IAM Identity Center app. Therefore you must complete the next procedure to begin synchronizing users and groups to IAM Identity Center.

Step 3: Assign access for users and groups in Okta

Use the following procedures in Okta to assign access to your users and groups. Okta users who belong to groups that you assign here are synchronized automatically to IAM Identity Center. To minimize administrative overhead in both Okta and IAM Identity Center, we recommend that you assign and *push* groups instead of individual users.

After you complete this step and the first synchronization with SCIM is completed, the users and groups that you have assigned appear in IAM Identity Center. Those users are able to access the AWS access portal using their Okta credentials.

To assign access for users in Okta

1. In the **IAM Identity Center app** page, choose the **Assignments** tab.
2. In the **Assignments** page, choose **Assign**, and then choose **Assign to People**.
3. Choose the Okta user or users whom you want to assign access to the IAM Identity Center app. Choose **Assign**, choose **Save and Go Back**, and then choose **Done**. This starts the process of provisioning the user or users into IAM Identity Center.

To assign access for groups in Okta

1. On the **IAM Identity Center app** page, choose the **Assignments** tab.
2. In the **Assignments** page, choose **Assign**, and then choose **Assign to Groups**.
3. Choose the Okta group or groups that you want to assign access to the IAM Identity Center app. Choose **Assign**, choose **Save and Go Back**, and then choose **Done**. This starts the process of provisioning the users in the group into IAM Identity Center.
4. Choose the **Push Groups** tab. Choose the Okta group or groups that you chose in the previous step.

    Note
    These chosen groups must be different from those assigned to the application. To maintain consistent group memberships between Okta and IAM Identity Center, you need to create a separate group and configure it to push groups to IAM Identity Center.

    Then choose **Save**. The group status changes to **Active** after the group and its members have successfully been pushed to IAM Identity Center.

To grant your Okta users access to AWS accounts and cloud applications, complete the following applicable procedures from the IAM Identity Center console:

- To grant access to AWS accounts, see Assign user access to AWS accounts (p. 95).
- To grant access to cloud applications, see Assign user access to applications (p. 121).

(Optional) Step 4: Configure user attributes in Okta for access control in IAM Identity Center

This is an optional procedure for Okta should you choose to configure attributes you will use in IAM Identity Center to manage access to your AWS resources. The attributes you define in Okta will be passed in a SAML assertion to IAM Identity Center, you will then create a permission set in IAM Identity Center to manage access based on the attributes you passed from Okta.

Before you begin this procedure, you first need to enable the Attributes for access control (p. 106) feature. For more information about how to do this, see Enable and configure attributes for access control (p. 107).

To configure user attributes in Okta for access control in IAM Identity Center

1. In a separate browser window, log in to the Okta admin portal and navigate to the IAM Identity Center app.
2. On the **IAM Identity Center app** page, choose the **Sign On** tab, and then choose **Edit**.
3. In the **SAML** section, expand **Attributes (Optional)**.
4. In the **Attribute Statements (optional)** section, do the following for each attribute where you will use IAM Identity Center for access control:

a. In the **Name** field, enter `https://aws.amazon.com/SAML/Attributes/AccessControl:`**`AttributeName`**, and replace **AttributeName** with the name of the attribute you are expecting in IAM Identity Center. For example, `https://aws.amazon.com/SAML/Attributes/AccessControl:`**`Department`**.

b. In the **Name Format** field, choose **URI reference**.

c. In the **Value** field, enter `user.`**`AttributeName`**, replace **AttributeName** with the Okta default user profile variable name. For example, `user.`**`department`**. To find your Okta default user profile variable name, see View the Okta default user profile on the Okta website.

5. (Optional) Choose **Preview SAML**, to review a sample SAML assertion that includes the new attributes.

6. Choose **Save**.

(Optional) Passing attributes for access control

You can optionally use the Attributes for access control (p. 106) feature in IAM Identity Center to pass an `Attribute` element with the `Name` attribute set to `https://aws.amazon.com/SAML/Attributes/AccessControl:`**`{TagKey}`**. This element allows you to pass attributes as session tags in the SAML assertion. For more information about session tags, see Passing session tags in AWS STS in the *IAM User Guide*.

To pass attributes as session tags, include the `AttributeValue` element that specifies the value of the tag. For example, to pass the tag key-value pair `CostCenter = blue`, use the following attribute.

```
<saml:AttributeStatement>
<saml:Attribute Name="https://aws.amazon.com/SAML/Attributes/AccessControl:CostCenter">
<saml:AttributeValue>blue
</saml:AttributeValue>
</saml:Attribute>
</saml:AttributeStatement>
```

If you need to add multiple attributes, include a separate `Attribute` element for each tag.

Troubleshooting

The following can help you troubleshoot some common issues you might encounter while setting up automatic provisioning with Okta.

Base URL: Does not match required pattern

The SCIM endpoint URL that you pasted into **Base URL** likely contains a trailing forward slash (/). Remove the forward slash from the SCIM endpoint URL before pasting into **Base URL**. For example, https://scim.us-east-2.amazonaws.com/*xxxxxxxx-xxxx-xxxxx-xxxxxx-xxxx*/scim/v2.

Error during synchronization

After you have started synchronization, you might see the following error:

```
Automatic profile push of <user> to app IAM Identity Center failed: Error while trying to
 push profile update for <user>@Corp.Example.com: Bad Request. Errors reported by remote
 server: Request is unparsable, syntactically incorrect, or violates schema.
```

For SCIM synchronization to work:

- Every user must have a **First name**, **Last name**, **Username**, and **Display name** value specified. If any of these values are missing from a user, that user will not be provisioned.

- Usernames should be mapped to attributes that are unique within your directory in Okta.

- The following special characters must not be used in attributes that are synchronized with SCIM: `<>; :`
  `%`
- If you update a user's address you must have **streetAddress**, **city**, **state**, **zipCode** and the
  countryCode value specified. If any of these values are not specified for the Okta user at the time of
  synchronization, the user or changes to the user will not be provisioned.

OneLogin

IAM Identity Center supports automatic provisioning (synchronization) of user and group information
from OneLogin into IAM Identity Center using the System for Cross-domain Identity Management (SCIM)
v2.0 protocol. You configure this connection in OneLogin, using your SCIM endpoint for IAM Identity
Center and a bearer token that is created automatically by IAM Identity Center. When you configure
SCIM synchronization, you create a mapping of your user attributes in OneLogin to the named attributes
in IAM Identity Center. This causes the expected attributes to match between IAM Identity Center and
OneLogin.

The following steps walk you through how to enable automatic provisioning of users and groups from
OneLogin to IAM Identity Center using the SCIM protocol.

> **Note**
> Before you begin deploying SCIM, we recommend that you first review the Considerations for
> using automatic provisioning (p. 47).

Topics

Prerequisites

You will need the following before you can get started:

- A OneLogin account. If you do not have an existing account, you may be able to obtain a free trial or
  developer account from the OneLogin website.
- An IAM Identity Center-enabled account (free). For more information, see Enable IAM Identity Center.
- A SAML connection from your OneLogin account to IAM Identity Center. For more information, see
  Enabling Single Sign-On Between OneLogin and AWS on the AWS Partner Network Blog.

Step 1: Enable provisioning in IAM Identity Center

In this first step, you use the IAM Identity Center console to enable automatic provisioning.

To enable automatic provisioning in IAM Identity Center

1. After you have completed the prerequisites, open the IAM Identity Center console.
2. Choose **Settings** in the left navigation pane.
3. On the **Settings** page, locate the **Automatic provisioning** information box, and then choose **Enable**.
   This immediately enables automatic provisioning in IAM Identity Center and displays the necessary
   SCIM endpoint and access token information.

4. In the **Inbound automatic provisioning** dialog box, copy each of the values for the following options. You will need to paste these in later when you configure provisioning in your IdP.

    a. **SCIM endpoint**
    b. **Access token**

5. Choose **Close**.

You have now set up provisioning in the IAM Identity Center console. Now you need to do the remaining tasks using the OneLogin admin console as described in the following procedure.

Step 2: Configure provisioning in OneLogin

Use the following procedure in the OneLogin admin console to enable integration between IAM Identity Center and the IAM Identity Center app. This procedure assumes you have already configured the AWS Single Sign-On application in OneLogin for SAML authentication. If you have not yet created this SAML connection, please do so before proceeding and then return here to complete the SCIM provisioning process. For more information about configuring SAML with OneLogin, see Enabling Single Sign-On Between OneLogin and AWS on the AWS Partner Network Blog.

To configure provisioning in OneLogin

1. Sign in to OneLogin, and then navigate to **Applications > Applications**.
2. On the **Applications** page, search for the application you created previously to form your SAML connection with IAM Identity Center. Choose it and then choose **Configuration** from the left navigation bar.
3. In the previous procedure, you copied the **SCIM endpoint** value in IAM Identity Center. Paste that value into the **SCIM Base URL** field in OneLogin. Make sure that you remove the trailing forward slash at the end of the URL. Also, in the previous procedure you copied the **Access token** value in IAM Identity Center. Paste that value into the **SCIM Bearer Token** field in OneLogin.
4. Next to **API Connection**, click **Enable**, and then click **Save** to complete the configuration.
5. In the left navigation bar, choose **Provisioning**.
6. Select the check boxes for **Enable provisioning**, **Create user**, **Delete user**, and **Update user**, and then choose **Save**.
7. In the left navigation bar, choose **Users**.
8. Click **More Actions** and choose **Sync logins**. You should receive the message *Synchronizing users with AWS Single Sign-on*.
9. Click **More Actions** again, and then choose **Reapply entitlement mappings**. You should receive the message *Mappings are being reapplied*.
10. At this point, the provisioning process should begin. To confirm this, navigate to **Activity > Events**, and monitor the progress. Successful provisioning events, as well as errors, should appear in the event stream.
11. To verify that your users and groups have all been successfully synchronized to IAM Identity Center, return to the IAM Identity Center console and choose **Users**. Your synchronized users from OneLogin will appear on the **Users** page. You can also view your synchronized groups on the **Groups** page.
12. To synchronize user changes automatically to IAM Identity Center, navigate to the **Provisioning** page, locate the **Require admin approval before this action is performed** section, de-select **Create User**, **Delete User**, and/or **Update User**, and click **Save**.

(Optional) Step 3: Configure user attributes in OneLogin for access control in IAM Identity Center

This is an optional procedure for OneLogin should you choose to configure attributes you will use in IAM Identity Center to manage access to your AWS resources. The attributes that you define in OneLogin will be passed in a SAML assertion to IAM Identity Center. You will then create a permission set in IAM Identity Center to manage access based on the attributes you passed from OneLogin.

Before you begin this procedure, you must first enable the Attributes for access control (p. 106) feature. For more information about how to do this, see Enable and configure attributes for access control (p. 107).

To configure user attributes in OneLogin for access control in IAM Identity Center

1. Sign in to OneLogin, and then navigate to **Applications > Applications**.

2. On the **Applications** page, search for the application you created previously to form your SAML connection with IAM Identity Center. Choose it and then choose **Parameters** from the left navigation bar.

3. In the **Required Parameters** section, do the following for each attribute you want to use in IAM Identity Center:

   a. Choose **+**.

   b. In **Field name**, enter `https://aws.amazon.com/SAML/Attributes/AccessControl:`**`AttributeName`**, and replace **AttributeName** with the name of the attribute you are expecting in IAM Identity Center. For example, `https://aws.amazon.com/SAML/Attributes/AccessControl:`**`Department`**.

   c. Under **Flags**, check the box next to **Include in SAML assertion**, and choose **Save**.

   d. In the **Value** field, use the drop-down list to choose the OneLogin user attributes. For example, **Department**.

4. Choose **Save**.

(Optional) Passing attributes for access control

You can optionally use the Attributes for access control (p. 106) feature in IAM Identity Center to pass an `Attribute` element with the `Name` attribute set to `https://aws.amazon.com/SAML/Attributes/AccessControl:`**`{TagKey}`**. This element allows you to pass attributes as session tags in the SAML assertion. For more information about session tags, see Passing session tags in AWS STS in the *IAM User Guide*.

To pass attributes as session tags, include the `AttributeValue` element that specifies the value of the tag. For example, to pass the tag key-value pair `CostCenter = blue`, use the following attribute.

```
<saml:AttributeStatement>
<saml:Attribute Name="https://aws.amazon.com/SAML/Attributes/AccessControl:CostCenter">
<saml:AttributeValue>blue
</saml:AttributeValue>
</saml:Attribute>
</saml:AttributeStatement>
```

If you need to add multiple attributes, include a separate `Attribute` element for each tag.

Troubleshooting

The following can help you troubleshoot some common issues you might encounter while setting up automatic provisioning with OneLogin.

Groups are not provisioned to IAM Identity Center

By default, groups may not be provisioned from OneLogin to IAM Identity Center. Ensure that you've enabled group provisioning for your IAM Identity Center application in OneLogin. To do this, sign in to the OneLogin admin console, and check to make sure that the **Include in User Provisioning** option is selected under the properties of the IAM Identity Center application (**IAM Identity Center application > Parameters > Groups**). For more details on how to create groups in OneLogin, including how to synchronize OneLogin roles as groups in SCIM, please see the OneLogin website.

Nothing is synchronized from OneLogin to IAM Identity Center, despite all settings being correct

In addition to the note above regarding admin approval, you will need to **Reapply entitlement mappings** for many configuration changes to take effect. This can be found in **Applications > Applications > IAM Identity Center application > More Actions**. You can see details and logs for most actions in OneLogin, including synchronization events, under **Activity > Events**.

I've deleted or disabled a group in OneLogin, but it still appears in IAM Identity Center

OneLogin currently does not support the SCIM DELETE operation for groups, which means that the group will continue to exist in IAM Identity Center. You must therefore remove the group from IAM Identity Center directly to ensure that any corresponding permissions in IAM Identity Center for that group are removed.

I deleted a group in IAM Identity Center without first deleting it from OneLogin and now I'm having user/group sync issues

To remedy this situation, first ensure that you do not have any redundant group provisioning rules or configurations in OneLogin. For example, a group directly assigned to an application along with a rule that publishes to the same group. Next, delete any undesirable groups in IAM Identity Center. Finally, in OneLogin, **Refresh** the entitlements (**IAM Identity Center App > Provisioning > Entitlements**), and then **Reapply entitlement mappings (IAM Identity Center App > More Actions)**. To avoid this issue in the future, first make the change to stop provisioning the group in OneLogin, then delete the group from IAM Identity Center.

Ping Identity

The following Ping Identity products have been tested with IAM Identity Center.

Topics

PingFederate

IAM Identity Center supports automatic provisioning (synchronization) of user and group information from the PingFederate product by Ping Identity (hereafter "Ping") into IAM Identity Center. This provisioning uses the System for Cross-domain Identity Management (SCIM) v2.0 protocol. You configure this connection in PingFederate using your IAM Identity Center SCIM endpoint and access token. When you configure SCIM synchronization, you create a mapping of your user attributes in PingFederate to the named attributes in IAM Identity Center. This causes the expected attributes to match between IAM Identity Center and PingFederate.

This guide is based on PingFederate version 10.2. Steps for other versions may vary. Contact Ping for more information about how to configure provisioning to IAM Identity Center for other versions of PingFederate.

The following steps walk you through how to enable automatic provisioning of users and groups from PingFederate to IAM Identity Center using the SCIM protocol.

> **Note**
> Before you begin deploying SCIM, we recommend that you first review the Considerations for using automatic provisioning (p. 47). Then continue reviewing additional considerations in the next section.

Topics

- Step 2: Configure provisioning in PingFederate (p. 70)
- (Optional) Step 3: Configure user attributes in PingFederate for access control in IAM Identity Center (p. 71)
- (Optional) Passing attributes for access control (p. 72)

Prerequisites

You will need the following before you can get started:

- A working PingFederate server. If you do not have an existing PingFederate server, you might be able to obtain a free trial or developer account from the Ping Identity website. The trial includes licenses and software downloads and associated documentation.
- A copy of the PingFederate IAM Identity Center Connector software installed on your PingFederate server. For more information about how to obtain this software, see IAM Identity Center Connector on the Ping Identity website.
- An IAM Identity Center-enabled account (free). For more information, see Enable IAM Identity Center.
- A SAML connection from your PingFederate instance to IAM Identity Center. For instructions on how to configure this connection, see the PingFederate documentation. In summary, the recommended path is to use the IAM Identity Center Connector to configure "Browser SSO" in PingFederate, using the "download" and "import" metadata features on both ends to exchange SAML metadata between PingFederate and IAM Identity Center.

Additional considerations

The following are important considerations about PingFederate that can affect how you implement provisioning with IAM Identity Center.

- If an attribute (such as a phone number) is removed from a user in the data store configured in PingFederate, that attribute will not be removed from the corresponding user in IAM Identity Center. This is a known limitation in PingFederate's provisioner implementation. If an attribute is changed to a different (non-empty) value on a user, that change will be synchronized to IAM Identity Center.

Step 1: Enable provisioning in IAM Identity Center

In this first step, you use the IAM Identity Center console to enable automatic provisioning.

To enable automatic provisioning in IAM Identity Center

1. After you have completed the prerequisites, open the IAM Identity Center console.
2. Choose **Settings** in the left navigation pane.
3. On the **Settings** page, locate the **Automatic provisioning** information box, and then choose **Enable**. This immediately enables automatic provisioning in IAM Identity Center and displays the necessary SCIM endpoint and access token information.
4. In the **Inbound automatic provisioning** dialog box, copy each of the values for the following options. You will need to paste these in later when you configure provisioning in your IdP.

    a. **SCIM endpoint**
    b. **Access token**
5. Choose **Close**.

Now that you have set up provisioning in the IAM Identity Center console, you must complete the remaining tasks using the PingFederate administrative console., The steps are described in the following procedure.

Step 2: Configure provisioning in PingFederate

Use the following procedure in the PingFederate administrative console to enable integration between IAM Identity Center and the IAM Identity Center Connector. This procedure assumes that you have already installed the IAM Identity Center Connector software. If you have not yet done so, refer to Prerequisites (p. 69), and then complete this procedure to configure SCIM provisioning.

> **Important**
> If your PingFederate server has not been previously configured for outbound SCIM provisioning, you may need to make a configuration file change to enable provisioning. For more information, see Ping documentation. In summary, you must modify the `pf.provisioner.mode` setting in the **pingfederate-<version>/pingfederate/bin/run.properties** file to a value other than `OFF` (which is the default), and restart the server if currently running. For example, you may choose to use `STANDALONE` if you don't currently have a high-availability configuration with PingFederate.

To configure provisioning in PingFederate

1. Sign on to the PingFederate administrative console.

2. Select **Applications** from the top of the page, then click **SP Connections**.

3. Locate the application you created previously to form your SAML connection with IAM Identity Center, and click on the connection name.

4. Select **Connection Type** from the dark navigation headings near the top of the page. You should see **Browser SSO** already selected from your previous configuration of SAML. If not, you must complete those steps first before you can continue.

5. Select the **Outbound Provisioning** check box, choose **IAM Identity Center Cloud Connector** as the type, and click **Save**. If **IAM Identity Center Cloud Connector** does not appear as an option, ensure that you have installed the IAM Identity Center Connector and have restarted your PingFederate server.

6. Click **Next** repeatedly until you arrive on the **Outbound Provisioning** page, and then click the **Configure Provisioning** button.

7. In the previous procedure, you copied the **SCIM endpoint** value in IAM Identity Center. Paste that value into the **SCIM URL** field in the PingFederate console. Make sure that you remove the trailing forward slash at the end of the URL. Also, in the previous procedure you copied the **Access token** value in IAM Identity Center. Paste that value into the **Access Token** field in the PingFederate console. Click **Save**.

8. On the **Channel Configuration (Configure Channels)** page, click **Create**.

9. Enter a **Channel Name** for this new provisioning channel (such as **AWSIAMIdentityCenterchannel**), and click **Next**.

10. On the **Source** page, choose the **Active Data Store** you want to use for your connection to IAM Identity Center, and click **Next**.

    > **Note**
    > If you have not yet configured a data source, you must do so now. See the Ping product documentation for information on how to choose and configure a data source in PingFederate.

11. On the **Source Settings** page, confirm all values are correct for your installation, then click **Next**.

12. On the **Source Location** page, enter settings appropriate to your data source, and then click **Next**. For example, if using Active Directory as an LDAP directory:

    a. Enter the **Base DN** of your AD forest (such as **DC=myforest,DC=mydomain,DC=com**).

    b. In **Users > Group DN**, specify a single group that contains all of the users that you want to provision to IAM Identity Center. If no such single group exists, create that group in AD, return to this setting, and then enter the corresponding DN.

    c. Specify whether to search subgroups (**Nested Search**), and any required LDAP **Filter**.

      d.   In **Groups > Group DN**, specify a single group that contains all of the groups that you want to provision to IAM Identity Center. In many cases, this may be the same DN as you specified in the **Users** section. Enter **Nested Search** and **Filter** values as required.

13. On the **Attribute Mapping** page, ensure the following, and then click **Next**:

      a.   The **userName** field must be mapped to an **Attribute** that is formatted as an email (user@domain.com). It must also match the value that the user will use to log in to Ping. This value in turn is populated in the SAML `nameId` claim during federated authentication and used for matching to the user in IAM Identity Center. For example, when using Active Directory, you may choose to specify the `UserPrincipalName` as the **userName**.

      b.   Other fields suffixed with a ***** must be mapped to attributes that are non-null for your users.

14. On the **Activation & Summary** page, set the **Channel Status** to **Active** to cause the synchronization to start immediately after the configuration is saved.

15. Confirm that all configuration values on the page are correct, and click **Done**.

16. On the **Manage Channels** page, click **Save**.

17. At this point, provisioning will start. To confirm activity, you can view the **provisioner.log** file, located by default in the **pingfederate-<version>/pingfederate/log** directory on your PingFederate server.

18. To verify that users and groups have been successfully synchronized to IAM Identity Center, return to the IAM Identity Center Console and choose **Users**. Synchronized users from PingFederate will appear on the **Users** page. You can also view synchronized groups on the **Groups** page.

(Optional) Step 3: Configure user attributes in PingFederate for access control in IAM Identity Center

This is an optional procedure for PingFederate should you choose to configure attributes you will use in IAM Identity Center to manage access to your AWS resources. The attributes that you define in PingFederate will be passed in a SAML assertion to IAM Identity Center. You will then create a permission set in IAM Identity Center to manage access based on the attributes you passed from PingFederate.

Before you begin this procedure, you must first enable the Attributes for access control (p. 106) feature. For more information about how to do this, see Enable and configure attributes for access control (p. 107).

To configure user attributes in PingFederate for access control in IAM Identity Center

1. Sign on to the PingFederate administrative console.

2. Choose **Applications** from the top of the page, then click **SP Connections**.

3. Locate the application you created previously to form your SAML connection with IAM Identity Center, and click on the connection name.

4. Choose **Browser SSO** from the dark navigation headings near the top of the page. Then click on **Configure Browser SSO**.

5. On the **Configure Browser SSO** page, choose **Assertion Creation**, and then click on **Configure Assertion Creation**.

6. On the **Configure Assertion Creation** page, choose **Attribute Contract**.

7. On the **Attribute Contract** page, under **Extend the Contract** section, add a new attribute by performing the following steps:

      a.   In the text box, enter `https://aws.amazon.com/SAML/Attributes/ AccessControl:`**`AttributeName`**, replace **`AttributeName`** with the name of the attribute you are expecting in IAM Identity Center. For example, `https://aws.amazon.com/SAML/ Attributes/AccessControl:`**`Department`**.

    b.   For **Attribute Name Format**, choose **urn:oasis:names:tc:SAML:2.0:attrname-format:uri**.

    c.   Choose **Add**, and then choose **Next**.

8. On the **Authentication Source Mapping** page, choose the Adapter Instance configured with your application.

9. On the **Attribute Contract Fulfillment** page, choose the **Source** (*data store*) and **Value** (*data store attribute*) for the **Attribute Contract** `https://aws.amazon.com/SAML/Attributes/AccessControl:`**`Department`**.

> **Note**
> If you have not yet configured a data source, you will need to do so now. See the Ping product documentation for information on how to choose and configure a data source in PingFederate.

10. Click **Next** repeatedly until you arrive on the **Activation & Summary** page, and then click **Save**.

(Optional) Passing attributes for access control

You can optionally use the Attributes for access control (p. 106) feature in IAM Identity Center to pass an `Attribute` element with the `Name` attribute set to `https://aws.amazon.com/SAML/Attributes/AccessControl:`**`{TagKey}`**. This element allows you to pass attributes as session tags in the SAML assertion. For more information about session tags, see Passing session tags in AWS STS in the *IAM User Guide*.

To pass attributes as session tags, include the `AttributeValue` element that specifies the value of the tag. For example, to pass the tag key-value pair `CostCenter = blue`, use the following attribute.

```
<saml:AttributeStatement>
<saml:Attribute Name="https://aws.amazon.com/SAML/Attributes/AccessControl:CostCenter">
<saml:AttributeValue>blue
</saml:AttributeValue>
</saml:Attribute>
</saml:AttributeStatement>
```

If you need to add multiple attributes, include a separate `Attribute` element for each tag.

PingOne

IAM Identity Center supports automatic provisioning (synchronization) of user information from the PingOne product by Ping Identity (hereafter "Ping") into IAM Identity Center. This provisioning uses the System for Cross-domain Identity Management (SCIM) v2.0 protocol. You configure this connection in PingOne using your IAM Identity Center SCIM endpoint and access token. When you configure SCIM synchronization, you create a mapping of your user attributes in PingOne to the named attributes in IAM Identity Center. This causes the expected attributes to match between IAM Identity Center and PingOne.

This guide is based on PingOne as of October 2020. Steps for newer versions may vary. Contact Ping for more information about how to configure provisioning to IAM Identity Center for other versions of PingOne. This guide also contains a few notes regarding configuration of user authentication through SAML.

The following steps walk you through how to enable automatic provisioning of users from PingOne to IAM Identity Center using the SCIM protocol.

> **Note**
> Before you begin deploying SCIM, we recommend that you first review the Considerations for using automatic provisioning (p. 47). Then continue reviewing additional considerations in the next section.

Topics

Prerequisites

You will need the following before you can get started:

- A PingOne subscription or free trial, with both federated authentication and provisioning capabilities. For more information about how to obtain a free trial, see the Ping Identity website.
- An IAM Identity Center-enabled account (free). For more information, see Enable IAM Identity Center.
- The PingOne IAM Identity Center application added to your PingOne admin portal. You can obtain the PingOne IAM Identity Center application from the PingOne Application Catalog. For general information, see Add an application from the Application Catalog on the Ping Identity website.
- A SAML connection from your PingOne instance to IAM Identity Center. After the PingOne IAM Identity Center application has been added to your PingOne admin portal, you must use it to configure a SAML connection from your PingOne instance to IAM Identity Center. Use the "download" and "import" metadata feature on both ends to exchange SAML metadata between PingOne and IAM Identity Center. For instructions on how to configure this connection, see the PingOne documentation.

Additional considerations

The following are important considerations about PingOne that can affect how you implement provisioning with IAM Identity Center.

- As of October 2020, PingOne does not support provisioning of groups through SCIM. Please contact Ping for the latest information on group support in SCIM for PingOne.
- Users may continue to be provisioned from PingOne after disabling provisioning in the PingOne admin portal. If you need to terminate provisioning immediately, delete the relevant SCIM bearer token, and/or disable Automatic provisioning (p. 47) in IAM Identity Center.
- If an attribute for a user is removed from the data store configured in PingOne, that attribute will not be removed from the corresponding user in IAM Identity Center. This is a known limitation in PingOne's provisioner implementation. If an attribute is modified, the change will be synchronized to IAM Identity Center.
- The following are important notes regarding your SAML configuration in PingOne:
  - IAM Identity Center supports only `emailaddress` as a `NameId` format. This means you need to choose a user attribute that is unique within your directory in PingOne, non-null, and formatted as an email/UPN (for example, user@domain.com) for your **SAML_SUBJECT** mapping in PingOne. **Email (Work)** is a reasonable value to use for test configurations with the PingOne built-in directory.
  - Users in PingOne with an email address containing a **+** character may be unable to sign in to IAM Identity Center, with errors such as `'SAML_215'` or `'Invalid input'`. To fix this, in PingOne, choose the **Advanced** option for the **SAML_SUBJECT** mapping in **Attribute Mappings**. Then set **Name ID Format to send to SP:** to **urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress** in the drop-down menu.

Step 1: Enable provisioning in IAM Identity Center

In this first step, you use the IAM Identity Center console to enable automatic provisioning.

To enable automatic provisioning in IAM Identity Center

1. After you have completed the prerequisites, open the IAM Identity Center console.
2. Choose **Settings** in the left navigation pane.
3. On the **Settings** page, locate the **Automatic provisioning** information box, and then choose **Enable**. This immediately enables automatic provisioning in IAM Identity Center and displays the necessary SCIM endpoint and access token information.
4. In the **Inbound automatic provisioning** dialog box, copy each of the values for the following options. You will need to paste these in later when you configure provisioning in your IdP.

   a. **SCIM endpoint**
   b. **Access token**
5. Choose **Close**.

Now that you have set up provisioning in the IAM Identity Center console, you need to complete the remaining tasks using the PingOne IAM Identity Center application. These steps are described in the following procedure.

Step 2: Configure provisioning in PingOne

Use the following procedure in the PingOne IAM Identity Center application to enable provisioning with IAM Identity Center. This procedure assumes that you have already added the PingOne IAM Identity Center application to your PingOne admin portal. If you have not yet done so, refer to Prerequisites (p. 73), and then complete this procedure to configure SCIM provisioning.

To configure provisioning in PingOne

1. Open the PingOne IAM Identity Center application that you installed as part of configuring SAML for PingOne (**Applications** > **My Applications**). See Prerequisites (p. 73).
2. Scroll to the bottom of the page. Under **User Provisioning**, choose the **complete** link to navigate to the user provisioning configuration of your connection.
3. On the **Provisioning Instructions** page, choose **Continue to Next Step**.
4. In the previous procedure, you copied the **SCIM endpoint** value in IAM Identity Center. Paste that value into the **SCIM URL** field in the PingOne IAM Identity Center application. Make sure that you remove the trailing forward slash at the end of the URL. Also, in the previous procedure you copied the **Access token** value in IAM Identity Center. Paste that value into the **ACCESS_TOKEN** field in the PingOne IAM Identity Center application.
5. For **REMOVE_ACTION**, choose either **Disabled** or **Deleted** (see the description text on the page for more details).
6. On the **Attribute Mapping** page, choose a value to use for the **SAML_SUBJECT** (NameId) assertion, following guidance from Additional considerations (p. 73) earlier on this page. Then choose **Continue to Next Step**.
7. On the **PingOne App Customization - IAM Identity Center** page, make any desired customization changes (optional), and click **Continue to Next Step**.
8. On the **Group Access** page, choose the groups containing the users you would like to enable for provisioning and single sign-on to IAM Identity Center. Choose **Continue to Next Step**.
9. Scroll to the bottom of the page, and choose **Finish** to start provisioning.
10. To verify that users have been successfully synchronized to IAM Identity Center, return to the IAM Identity Center console and choose **Users**. Synchronized users from PingOne will appear on the **Users** page. These users can now be assigned to accounts and applications within IAM Identity Center.

    Remember that PingOne does not support provisioning of groups or group memberships through SCIM. Contact Ping for more information.

(Optional) Step 3: Configure user attributes in PingOne for access control in IAM Identity Center

This is an optional procedure for PingOne should you choose to configure attributes for IAM Identity Center to manage access to your AWS resources. The attributes that you define in PingOne will be passed in a SAML assertion to IAM Identity Center. You then create a permission set in IAM Identity Center to manage access based on the attributes you passed from PingOne.

Before you begin this procedure, you must first enable the Attributes for access control (p. 106) feature. For more information about how to do this, see Enable and configure attributes for access control (p. 107).

To configure user attributes in PingOne for access control in IAM Identity Center

1. Open the PingOne IAM Identity Center application that you installed as part of configuring SAML for PingOne (**Applications > My Applications**).
2. Choose **Edit**, and then choose **Continue to Next Step** until you get to the **Attribute Mappings** page.
3. On the **Attribute Mappings** page, choose **Add new attribute**, and then do the following. You must perform these steps for each attribute you will add for use in IAM Identity Center for access control.

   a. In the **Application Attribute** field, enter `https://aws.amazon.com/SAML/Attributes/AccessControl:`*`AttributeName`*. Replace *`AttributeName`* with the name of the attribute you are expecting in IAM Identity Center. For example, `https://aws.amazon.com/SAML/Attributes/AccessControl:`**`Email`**.

   b. In the **Identity Bridge Attribute or Literal Value** field, choose user attributes from your PingOne directory. For example, **Email (Work)**.

4. Choose **Next** a few times, and then choose **Finish**.

(Optional) Passing attributes for access control

You can optionally use the Attributes for access control (p. 106) feature in IAM Identity Center to pass an `Attribute` element with the `Name` attribute set to `https://aws.amazon.com/SAML/Attributes/AccessControl:`**`{TagKey}`**. This element allows you to pass attributes as session tags in the SAML assertion. For more information about session tags, see Passing session tags in AWS STS in the *IAM User Guide*.

To pass attributes as session tags, include the `AttributeValue` element that specifies the value of the tag. For example, to pass the tag key-value pair `CostCenter = blue`, use the following attribute.

```
<saml:AttributeStatement>
<saml:Attribute Name="https://aws.amazon.com/SAML/Attributes/AccessControl:CostCenter">
<saml:AttributeValue>blue
</saml:AttributeValue>
</saml:Attribute>
</saml:AttributeStatement>
```

If you need to add multiple attributes, include a separate `Attribute` element for each tag.

Other identity providers

IAM Identity Center implements the following standards-based protocols for identity federation:

- SAML 2.0 for user authentication
- SCIM for provisioning

Any identity provider (IdP) that implements these standard protocols is expected to interoperate successfully with IAM Identity Center, with the following special considerations:

- **SAML**
  - IAM Identity Center requires a SAML nameID format of email address (that is, `urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress`).
  - The value of the nameID field in assertions must be an RFC 2822 (https://tools.ietf.org/html/rfc2822) addr-spec compliant ("name@domain.com") string (https://tools.ietf.org/html/rfc2822#section-3.4.1).
  - The metadata file cannot be over 75000 characters.
  - The metadata must contain an entityId, X509 certificate, and SingleSignOnService as part of the sign-in URL.
  - An encryption key is not supported.

- **SCIM**
  - The IAM Identity Center SCIM implementation is based on SCIM RFCs 7642 (https://tools.ietf.org/html/rfc7642), 7643 (https://tools.ietf.org/html/rfc7643), and 7644 (https://tools.ietf.org/html/rfc7644), and the interoperability requirements laid out in the March 2020 draft of the FastFed Basic SCIM Profile 1.0 (https://openid.net/specs/fastfed-scim-1_0-02.html#rfc.section.4). Any differences between these documents and the current implementation in IAM Identity Center are described in the Supported API operations section of the *IAM Identity Center SCIM Implementation Developer Guide*.

IdPs that do not conform to the standards and considerations mentioned above are not supported. Please contact your IdP for questions or clarifications regarding the conformance of their products to these standards and considerations.

If you have any issues connecting your IdP to IAM Identity Center, we recommend that you check:

- AWS CloudTrail logs by filtering on the event name **ExternalIdPDirectoryLogin**
- IdP-specific logs and/or debug logs
- Troubleshooting IAM Identity Center issues (p. 186)

> **Note**
> Some IdPs, including the list of Supported identity providers (p. 52), offer a simplified configuration experience for IAM Identity Center in the form of an "application" or "connector" built specifically for IAM Identity Center. If your IdP provides this option, we recommend that you use it, being careful to choose the item that's built specifically for IAM Identity Center. Other items called "AWS", "AWS federation", or similar generic "AWS" names may use other federation approaches and/or endpoints, and may not work as expected with IAM Identity Center.

Using the AWS access portal

Your AWS access portal provides you with single sign-on access to all your AWS accounts and most commonly used cloud applications such as Office 365, Concur, Salesforce, and many more. From here you can quickly launch multiple applications simply by choosing the AWS account or application icon in the portal. The presence of icons in your portal means that an administrator or designated help desk employee from your company has granted you access to those AWS accounts or applications. It also means that you can access all these accounts or applications from the portal without additional sign-in prompts.

Contact your administrator or help desk to request additional access in the following situations:

- You don't see an AWS account or application that you need access to.
- The access that you have to a given account or application is not what you expected.

Topics

Tips for using the portal

Like any business tool or application that you use on a daily basis, the AWS access portal might not work as you expected. If that happens, try these tips:

- Occasionally, you may need to sign out and sign back in to the AWS access portal. This might be necessary to access new applications that your administrator recently assigned to you. This is not required, however, because all new applications are refreshed every hour.

- When you sign in to the AWS access portal, you can open any of the applications listed in the portal by choosing the application's icon. After you are done using the application, you can either close the application or sign out of the AWS access portal. Closing the application signs you out of that application only. Any other applications that you have opened from the AWS access portal remain open and running.

- Before you can sign in as a different user, you must first sign out of the AWS access portal. Signing out from the portal completely removes your credentials from the browser session.

- Once you sign in to the AWS access portal, you can switch to a role. This temporarily sets aside your original user permissions and instead gives you the permissions assigned to the role. For more information, see Switching to a role (console).

Accepting the invitation to join IAM Identity Center

If this is your first time signing into the AWS access portal, check your email for instructions on how to activate your account.

To activate your account

1. Depending on the email you received from your company, choose one of the following methods to activate your account so that you can start using the AWS access portal.

   a. If you received an email with the subject **Invitation to join AWS Single Sign-On**, open it and choose **Accept invitation**. On the **New user sign up** page, enter and confirm a password, and then choose **Set new password**. You'll use that password each time you sign in to the portal.

   b. If you were sent an email from your company's IT support or IT administrator, follow the instructions they provided to activate your account.

2. After you activate your account by providing a new password, the AWS access portal signs you in automatically. If this doesn't occur, you can manually sign in to the AWS access portal by using the instructions provided in the next section.

Signing up in the AWS access portal

When you sign up for an AWS account, you create a personal or business account. This helps us determine what information we need to collect during sign-up.

To sign up for AWS

1. Under **Root user email address**, enter your email address. For information about security best practices, see Getting Started: Follow Security Best Practices as You Configure Your AWS Resources .

2. **AWS account name** auto-fills with your email domain.

3. Choose **Verify email address**. We then send a verification code to your email address.

   > **Note**
   > Email from the IAM Identity Center service comes from the address no-reply@signin.aws.
   > We recommend that you configure your mail system so that it accepts email from this
   > address and does not treat it as junk or spam.
   > You might receive a different email from us if there is a problem creating your account.
   > Review this email and follow its instructions to correct the problem.

4. Under **Verification code**, enter the code that you received. Then choose **Verify**.

5. Under **Root user password**, enter a password containing 8-20 characters, numbers, and both uppercase and lowercase letters. Under **Confirm root user password**, enter your password again. Choose **Continue**.

6. Under **How do you plan to use AWS?**, choose **Business - for your work, school, or organization** or **Personal - for your own projects**, based on your account needs.

7. Under **Contact information**, provide the required information. Choose the **I have read and agree to the terms of the AWS Customer Agreement** check box. Then choose **Continue**.

8. Under **Billing information**, provide the required information. Choose **Verify**.

9. Under **Identity Verification**, choose either **Phone call** or **SMS** and enter your phone number.

   a. If you selected **Phone call**, you receive a verification code on your screen and a phone call. Enter this code on the phone's keypad.

   b. If you selected **SMS**, you receive a text with a verification code. Enter this code in the field provided.

10. Under **Select a support plan**, choose the support plan that best meets your personal or business needs. Choose **Complete sign up**. You're taken to a page that confirms we are activating your account. Choose **Let's go!**

Signing in to the AWS access portal

By this time, you should have been provided a specific sign-in URL to the AWS access portal by an administrator or help desk employee. Once you have this, you can proceed with the following steps to sign in to the portal.

> **Note**
> Once you have been signed-in, your AWS access portal session will be valid for 8 hours.

To sign in to the AWS access portal

1. In your browser window, paste in the sign-in URL that you were provided and press **Enter**. The URL looks like d-xxxxxxxxxx.awsapps.com/start or *your_subdomain*.awsapps.com/start. We recommend that you bookmark this link to the portal now so that you can quickly access it later.

2. Sign in using your standard company user name and password.

Note

If your administrator sent you an email one-time password (OTP) and this is your first time signing in, enter that password. After you're signed in, you must create a new password for future sign-ins.

If you are prompted for a **Verification code**, check your email and then copy and paste the code into the sign-in page.

Note

Verification codes are typically sent through email, but the delivery method can vary. Check with your administrator for details.

3. Once signed in, you can access any AWS account and application that appears in the portal. Simply choose an icon.

Trusted devices

After you choose the option **This is a trusted device** from the sign-in page, IAM Identity Center will consider all future sign-ins from that device as authorized. This means that IAM Identity Center will not present an option to enter in an MFA code as long as you are using that trusted device. However, there are some exceptions. These include signing in from a new browser or when your device has been issued an unknown IP address.

Signing out of the AWS access portal

When you sign out from the portal, your credentials are completely removed from the browser session.

Note

If you want to sign in as a different user, you must first sign out of the AWS access portal.

To sign out of the AWS access portal

- In the AWS access portal, choose **Sign out** from the upper right corner of the portal.

Searching for an AWS account or application

If your list of applications or AWS accounts is too large to find what you need, you can use the **Search** box.

To search for an AWS account or application in the AWS access portal

1. While signed into the portal, choose the **Search** box.
2. Enter the name of the application. Then press **Enter**.

Resetting your password

From time to time you may need to reset your password, depending on your company policies.

To reset your password

1. Open a browser and go to the sign-in page for your AWS access portal.
2. Under the **Sign In** button, choose **Forgot Password?**.
3. Provide your **Username** and enter the characters for the provided image to confirm that you are not a robot. Then choose **Recover Password**. This sends an email to you with the subject **AWS Directory Service Reset Password Request**.

4. Once you receive the email, choose **Reset Password**.

5. On the **Single Sign-On** page, need to specify a new password for the portal. Once you have provided a password and have confirmed it, choose **Reset Password**.

Getting IAM role credentials for CLI access

Use this procedure in the AWS access portal when you need temporary security credentials for short-term access to resources in an AWS account using the AWS CLI. The AWS access portal makes it easy for you to quickly select an AWS account and get the temporary credentials for a given IAM role. You can then copy the necessary CLI syntax (including all necessary credentials) and paste them into your AWS CLI command prompt.

By default, credentials retrieved through the AWS access portal are valid for 1 hour. You can change this value up to 12 hours. Once you have completed this procedure, you can run any AWS CLI command (that your administrator has given you access to) until those temporary credentials have expired.

> **Note**
> Before you get started with the steps in this procedure, you must first install the AWS CLI. For instructions, see Installing the AWS Command Line Interface.

To get temporary credentials of an IAM role for CLI access to an AWS account

1. While signed into the portal, choose the **AWS Accounts** icon ● to expand the list of accounts.

2. Choose the AWS account from which you want to retrieve access credentials. Then, next to the IAM role name (for example **Administrator**), choose **Command line or programmatic access**.

3. In the **Get credentials** dialog box, choose either **MacOS and Linux** or **Windows**, depending on the operating system where you plan to use the CLI command prompt.

4. Depending on how you want to use the temporary credentials, choose one or more of the following options:

   - If you need to run commands from the AWS CLI in the selected AWS account, under **Option 1: Set AWS environment variables**, pause on the commands. Then choose **Copy**. Paste the commands into the CLI terminal window and press **Enter** to set the necessary environment variables.

   - If you need to run commands from multiple command prompts in the same AWS account, under **Option 2: Add a profile to your AWS credentials file**, pause on the commands. Then choose **Copy**. Paste the commands into your AWS credentials file to set up a newly named profile. For more information, see Configuration and Credential Files in the *AWS CLI User Guide*. Modifying the credential files in this way enables the `--profile` option in your AWS CLI command so that you can use this credential. This affects all command prompts that use the same credential file.

   - If you need to access AWS resources from an AWS service client, under **Option 3: Use individual values in your AWS service client**, choose **Copy** next to the commands you need to use. For more information, see Configuration and Credential File Settings in the *AWS CLI User Guide* or see Tools for Amazon Web Services.

5. Continue using the AWS CLI as necessary for your AWS account until the credentials have expired.

Bookmarking an IAM role

To make it easier for you to access frequently used IAM roles from the AWS access portal, you can create a bookmark for a given role associated with a specific AWS account.

To bookmark an IAM role for a specific AWS account

1. While signed into the portal, choose the **AWS Accounts** icon ● to expand the list of accounts.

2. Choose the AWS account where you want to create the bookmark.

3. Right-click the **Management console** link, copy the link address, and then use that URL to create your bookmark.

Registering a device for MFA

Use the following procedure within the AWS access portal to register your new device for multi-factor authentication (MFA).

> **Note**
> We recommend that you first download the appropriate Authenticator app onto your device before starting the steps in this procedure. For a list of apps that you can use for MFA devices, see Authenticator apps (p. 85).

To register your device for use with MFA

1. Sign in to your AWS access portal. For more information, see Signing in to the AWS access portal (p. 78).

2. Near the top-right of the page, choose **MFA devices**.

3. On the **Multi-factor authentication (MFA) devices** page, choose **Register device**.

   > **Note**
   > If the **Register MFA device** option is grayed out, you will need to contact your administrator for assistance with registering your device.

4. On the **Register MFA device** page, select one of the following MFA device types, and follow the instructions:

   - **Authenticator app**

     1. On the **Set up the authenticator app** page, you might notice configuration information for the new MFA device, including a QR code graphic. The graphic is a representation of the secret key that is available for manual entry on devices that do not support QR codes.

     2. Using the physical MFA device, do the following:

        a. Open a compatible MFA authenticator app. For a list of tested apps that you can use with MFA devices, see Authenticator apps (p. 85). If the MFA app supports multiple accounts (multiple MFA devices), choose the option to create a new account (a new MFA device).

        b. Determine whether the MFA app supports QR codes, and then do one of the following on the **Set up the authenticator app** page:

           i. Choose **Show QR code**, and then use the app to scan the QR code. For example, you might choose the camera icon or choose an option similar to **Scan code**. Then use the device's camera to scan the code.

           ii. Choose **show secret key**, and then enter that secret key into your MFA app.

              > **Important**
              > When you configure an MFA device for IAM Identity Center, we recommend that you save a copy of the QR code or secret key *in a secure place*. This can help if you lose the phone or have to reinstall the MFA authenticator app. If either of those things happen, you can quickly reconfigure the app to use the same MFA configuration.

     3. On the **Set up the authenticator app** page, under **Authenticator code**, enter the one-time password that currently appears on the physical MFA device.

        > **Important**
        > Submit your request immediately after generating the code. If you generate the code and then wait too long to submit the request, the MFA device is successfully associated with your user account. But the MFA device is out of sync. This happens because time-

based one-time passwords (TOTP) expire after a short period of time. If this happens, you can sync the device again.

4. Choose **Assign MFA**. The MFA device can now start generating one-time passwords and is now ready for use with AWS.

- **Security key** or **Built-in authenticator**

  1. On the **Register your user's security key** page, follow the instructions given to you by your browser or platform.

     Note
     The experience here will vary based on the different operating systems and browsers, so follow the instructions displayed by your browser or platform. After your device has been successfully registered, you will be given the option to associate a friendly display name to your newly enrolled device. If you want to change this, choose **Rename**, enter the new name, and then choose **Save**.

Customizing the AWS access portal URL

By default, your AWS access portal is accessible through a URL of the form `d-xxxxxxxxxx.awsapps.com/start`. However, you can customize the URL to `your_subdomain.awsapps.com/start`. This change to the AWS access portal URL is permanent, and you will not be able to edit it later.

To customize your URL

1. Sign in to your AWS access portal. For more information, see How to sign in to the AWS access portal.

2. In the IAM Identity Center console, under Dashboard, go to the **AWS access portal** section at the bottom of the page.

3. Choose **Customize**.

4. Enter your desired domain name and choose **Save**.

You can now sign in to the AWS Console through your AWS access portal with your customized `awsapps.com/start` URL.

Multi-factor authentication

When you enable multi-factor authentication (MFA), users must sign in to the AWS access portal with their user name and password. This is the first factor, something they know. Users must also sign in with either a code or security key. This is the second factor, something they have or something they are. The second factor could be either an authentication code generated from their mobile device or alternatively by tapping on a security key connected to their computer. Taken together, these multiple factors provide increased security by preventing unauthorized access to your AWS resources unless a valid MFA challenge has been successfully completed.

Important
As a security best practice, we strongly recommend that you enable multi-factor authentication. MFA provides a simple and secure way to add an extra layer of protection on top of the default authentication mechanism of user name and password.

Topics
- Getting started (p. 83)

Getting started

You can enforce secure access to the AWS access portal, IAM Identity Center integrated apps, and the AWS CLI by enabling multi-factor authentication (MFA). Review the following topics to get started.

Topics

Considerations before using MFA in IAM Identity Center

Before you enable MFA, consider the following information:

- Users are encouraged to register multiple backup authenticators for all enabled MFA types. This practice can prevent the user's losing access in case of a broken or misplaced MFA device.

- Do not use the option **Require Them to Provide a One-Time Password Sent by Email** if your users must sign in to the user portal to access their email. For example, your users might use Office 365 on the AWS access portal to read their email. In this case, users would not be able to retrieve the verification code and would be unable to sign in to the AWS access portal. For more information, see Configure MFA device enforcement (p. 86).

- If you are already using RADIUS MFA that you configured with AWS Directory Service, then you do not need to enable MFA within IAM Identity Center. MFA in IAM Identity Center is an alternative to RADIUS MFA for Microsoft Active Directory users of IAM Identity Center. For more information, see RADIUS MFA (p. 85).

- You can use IAM Identity Center's multi-factor authentication capabilities when your identity source is configured with IAM Identity Center's identity store, AWS Managed Microsoft AD, or AD Connector. MFA in IAM Identity Center is currently not supported for use by external identity providers.

Enable MFA

Use the following steps to enable MFA using the IAM Identity Center console. Before you enable MFA, we recommend that you first review details about MFA types (p. 84).

> **Note**
> If you're using an external IdP, you will not see the **Multi-factor authentication** section. The external IdP manages MFA settings rather than IAM Identity Center managing them.

To enable MFA

1. Open the IAM Identity Center console.

2. In the left navigation pane, choose **Settings**.

3. On the **Settings** page, choose the **Authentication** tab.

4. In the **Multi-factor authentication** section, choose **Configure**.

5. On the **Configure multi-factor authentication** page, choose one of the following authentication modes based on the level of security that your business needs:

   - **Only when their sign-in context changes (context-aware)**

In this mode (the default), IAM Identity Center provides users the option to trust their device during sign-in. After a user indicates that they want to trust a device, IAM Identity Center prompts the user for MFA once and analyzes the sign-in context (such as device, browser, and location) for the user's subsequent sign-ins. For subsequent sign-ins, IAM Identity Center determines if the user is signing in with a previously trusted context. If the user's sign-in context changes, IAM Identity Center prompts the user for MFA in addition to their email address and password credentials.

This mode provides ease of use for users who frequently sign in from their workplace, so they don't need to complete MFA on every sign-in. They are only prompted for MFA if their sign-in context changes.

- **Every time they sign in (always-on)**

  In this mode, IAM Identity Center requires that users with a registered MFA device will be prompted every time they sign in. You should use this mode if you have organizational or compliance policies that require your users to complete MFA every time they sign in to the user portal. For example, PCI DSS strongly recommends MFA during every sign-in to access applications that support high-risk payment transactions.

- **Never (disabled)**

  While in this mode, all users will sign in with their standard user name and password only. Choosing this option disables IAM Identity Center MFA.

  > **Note**
  > If you are already using RADIUS MFA with AWS Directory Service, and want to continue using it as your default MFA type, then you can leave the authentication mode as disabled to bypass IAM Identity Center MFA's capabilities. Changing from **Disabled** mode to **Context-aware** or **Always-on** mode will override the existing RADIUS MFA settings. For more information, see RADIUS MFA (p. 85).

6. Choose **Save changes**.

   Related Topics

   - Configure MFA types (p. 86)
   - Configure MFA device enforcement (p. 86)
   - Allow users to register their own MFA devices (p. 89)

MFA types

MFA types represent the different mechanisms that users will be able to register and provide as secondary means of verifying their identity when challenged. All MFA types are supported for both browser-based console access as well as using the AWS CLI v2 with IAM Identity Center.

IAM Identity Center MFA provides support to enable one or both of the following types of client-side authentication types.

- Authenticator apps
- Security keys and built-in authenticators

Alternatively, you can also use your own RADIUS implementation connected through AWS Managed Microsoft AD. For more information, see RADIUS MFA (p. 85).

For more information, see Configure MFA types (p. 86).

Topics
- Authenticator apps (p. 85)

- Security keys and built-in authenticators (p. 85)
- RADIUS MFA (p. 85)

Authenticator apps

Authenticator apps are essentially one-time password (OTP)–based third party-authenticators. Users can use an authenticator application installed on their mobile device or tablet as an authorized MFA device. The third-party authenticator application must be compliant with RFC 6238, which is a standards-based TOTP (time-based one-time password) algorithm capable of generating six-digit authentication codes.

When prompted for MFA, users must enter a valid code from their authenticator app within the input box presented. Each MFA device assigned to a user must be unique. Two authenticator apps can be registered for any given user.

Tested authenticator apps

Although any TOTP-compliant application will work with IAM Identity Center MFA, the following table lists well-known third-party authenticator apps to choose from.

Operating system	Tested authenticator app
Android	Authy, Duo Mobile, LastPass Authenticator, Microsoft Authenticator, Google Authenticator
iOS	Authy, Duo Mobile, LastPass Authenticator, Microsoft Authenticator, Google Authenticator

Security keys and built-in authenticators

Web authentication (WebAuthn), enables strong cryptographic authentication using a variety of interoperable authenticators. WebAuthn is a core component of FIDO Alliance's FIDO2 set of specifications. WebAuthn enables the use of any FIDO2 compliant device as well as backward-compatible support for U2F devices.

- **Security keys –** Users can be authenticated using an external USB/BLE/NFC-connected security key such as YubiKey or Fetian devices. Authentication involves simply tapping on a key's sensor when prompted for MFA.
- **Built-In Authenticators –** Some FIDO2-enabled authenticators are built into devices. Examples include TouchID on MacBook or a Windows Hello compatible camera. Users with such a built-in authenticator, can simply provide their fingerprint or facial recognition as a suitable second factor.

FIDO2 and WebAuthn ensures privacy, as cryptographic data generated on devices is unique across sites and biometric data never leaves the device when used. FIDO devices are more secure than traditional forms of MFA, as they are strongly attestable and phishing resistant. For more information about WebAuthn and FIDO2, see FIDO2: Web Authentication (WebAuthn).

Not all operating systems and browser versions support WebAuthn, so you may have compatibility issues with some U2F devices. If you experience a compatibility issue, check with your U2F device provider to learn if you're on an unsupported browser.

RADIUS MFA

Remote Authentication Dial-In User Service (RADIUS) is an industry-standard client-server protocol that provides authentication, authorization, and accounting management so users can connect to network

services. AWS Directory Service includes a RADIUS client that connects to the RADIUS server upon which you have implemented your MFA solution. For more information, see Enable Multi-Factor Authentication for AWS Managed Microsoft AD.

You can use either RADIUS MFA or MFA in IAM Identity Center for user sign-ins to the user portal, but not both. MFA in IAM Identity Center is an alternative to RADIUS MFA in cases where you want AWS native two-factor authentication for access to the portal.

When you enable MFA in IAM Identity Center, your users need an MFA device to sign in to the AWS access portal. If you had previously used RADIUS MFA, enabling MFA in IAM Identity Center effectively overrides RADIUS MFA for users who sign in to the AWS access portal. However, RADIUS MFA continues to challenge users when they sign in to all other applications that work with AWS Directory Service, such as Amazon WorkDocs.

If your MFA is **Disabled** on the IAM Identity Center console and you have configured RADIUS MFA with AWS Directory Service, RADIUS MFA governs AWS access portal sign-in. This means that IAM Identity Center falls back to RADIUS MFA configuration if MFA is disabled.

How to manage MFA for IAM Identity Center

The following topics provide instructions for managing MFA in IAM Identity Center.

Topics
- Configure MFA types (p. 86)
- Configure MFA device enforcement (p. 86)
- Register an MFA device (p. 87)
- Manage a user's MFA device (p. 89)
- Allow users to register their own MFA devices (p. 89)
- Disable MFA (p. 89)

Configure MFA types

Use the following procedure to configure which device types your users can use when prompted for MFA in the AWS access portal.

To configure MFA types for your users

1. Open the IAM Identity Center console.
2. In the left navigation pane, choose **Settings**.
3. On the **Settings** page, choose the **Authentication** tab.
4. In the **Multi-factor authentication** section, choose **Configure**.
5. On the **Configure multi-factor authentication** page, under **Users can authenticate with these MFA types** choose one of the following MFA types based on your business needs. For more information, see MFA types (p. 84).

   - **Security keys and built-in authenticators**
   - **Authenticator apps**
6. Choose **Save changes**.

Configure MFA device enforcement

Use the following procedure to determine whether your users must have a registered MFA device when signing in to the AWS access portal.

To configure MFA device enforcement for your users

1. Open the IAM Identity Center console.
2. In the left navigation pane, choose **Settings**.
3. On the **Settings** page, choose the **Authentication** tab.
4. In the **Multi-factor authentication** section, choose **Configure**.
5. On the **Configure multi-factor authentication** page, under **If a user does not yet have a registered MFA device** choose one of the following choices based on your business needs:

   - **Require them to register an MFA device at sign in**

     Use this option when you want to require users who do not yet have a registered MFA device, to self-enroll a device during sign-in following a successful password authentication. This allows you to secure your organization's AWS environments with MFA without having to individually enroll and distribute authentication devices to your users. During self-enrollment, your users can register any device from the available MFA types (p. 84) you've previously enabled. After completing registration, users have the option to give their newly enrolled MFA device a friendly name, after which IAM Identity Center redirects the user to their original destination. If the user's device is lost or stolen, you can simply remove that device from their account, and IAM Identity Center will require them to self-enroll a new device during their next sign-in.

   - **Require them to provide a one-time password sent by email to sign in**

     Use this option when you want to have verification codes sent to users by email. Because email is not bound to a specific device, this option does not meet the bar for industry-standard multi-factor authentication. But it does improve security over having a password alone. Email verification will only be requested if a user has not registered an MFA device. If the **Context-aware** authentication method has been enabled, the user will have the opportunity to mark the device on which they receive the email as trusted. Afterward they will not be required to verify an email code on future logins from that device, browser, and IP address combination.

     > **Note**
     > If you are using Active Directory as your IAM Identity Center enabled identity source, the email address will always be based on the Active Directory `email` attribute. Custom Active Directory attribute mappings will not override this behavior.

   - **Block their sign-in**

     Use the **Block Their Sign-In** option when you want to enforce MFA use by every user before they can sign in to AWS.

     > **Important**
     > If your authentication method is set to **Context-aware** a user might select the **This is a trusted device** check box on the sign-in page. In that case, that user will not be prompted for MFA even if you have the **Block their sign in** setting enabled. If you want these users to be prompted, change your authentication method to **Always On**.

   - **Allow them to sign in**

     The default setting when you first configure IAM Identity Center MFA. Use this option to indicate that MFA devices are not required in order for your users to sign in to the AWS access portal. Users who chose to register MFA devices will still be prompted for MFA.

6. Choose **Save changes**.

Register an MFA device

Use the following procedure to set up a new MFA device for access by a specific user in the IAM Identity Center console. You must have physical access to the user's MFA device in order to register it. For example, you might configure MFA for a user who will use an MFA device running on a smartphone. In

that case, you must have the smartphone available in order to finish the wizard. For this reason, you might want to let users configure and manage their own MFA devices. For details on how to set this up, see Allow users to register their own MFA devices (p. 89).

To register an MFA device

1. Open the IAM Identity Center console.

2. In the left navigation pane, choose **Users**. Choose a user in the list. Don't select the checkbox next to the user for this step.

3. On the user details page, choose the **MFA devices** tab, and then choose **Register MFA device**.

4. On the **Register MFA device** page, select one of the following MFA device types, and follow the instructions:

   - **Authenticator app**

     1. On the **Set up the authenticator app** page, IAM Identity Center displays configuration information for the new MFA device, including a QR code graphic. The graphic is a representation of the secret key that is available for manual entry on devices that do not support QR codes.

     2. Using the physical MFA device, do the following:

        a. Open a compatible MFA authenticator app. For a list of tested apps that you can use with MFA devices, see Authenticator apps (p. 85). If the MFA app supports multiple accounts (multiple MFA devices), choose the option to create a new account (a new MFA device).

        b. Determine whether the MFA app supports QR codes, and then do one of the following on the **Set up the authenticator app** page:

           i. Choose **Show QR code**, and then use the app to scan the QR code. For example, you might choose the camera icon or choose an option similar to **Scan code**. Then use the device's camera to scan the code.

           ii. Choose **show secret key**, and then type that secret key into your MFA app.

              > **Important**
              > When you configure an MFA device for IAM Identity Center, we recommend that you save a copy of the QR code or secret key *in a secure place*. This can help if the assigned user loses the phone or has to reinstall the MFA authenticator app. If either of those things happen, you can quickly reconfigure the app to use the same MFA configuration. This avoids the need to create a new MFA device in IAM Identity Center for the user.

     3. On the **Set up the authenticator app** page, under **Authenticator code**, type the one-time password that currently appears on the physical MFA device.

        > **Important**
        > Submit your request immediately after generating the code. If you generate the code and then wait too long to submit the request, the MFA device is successfully associated with the user. But the MFA device is out of sync. This happens because time-based one-time passwords (TOTP) expire after a short period of time. If this happens, you can resync the device.

     4. Choose **Assign MFA**. The MFA device can now start generating one-time passwords and is now ready for use with AWS.

   - **Security key**

     1. On the **Register your user's security key** page, follow the instructions given to you by your browser or platform.

        > **Note**
        > The experience here varies based on the different operating systems and browsers, so please follow the instructions displayed by your browser or platform. After your user's

device has been successfully registered, you will be given the option to associate a
friendly display name to your user's newly enrolled device. If you want to change this,
choose **Rename**, enter the new name, and then choose **Save**. If you have enabled the
option to allow users to manage their own devices, the user will see this friendly name
in the AWS access portal.

Manage a user's MFA device

Use the following procedures when you need to rename or delete a user's MFA device.

To rename an MFA device

1. Open the IAM Identity Center console.
2. In the left navigation pane, choose **Users**. Choose the user in the list. Don't select the checkbox next to the user for this step.
3. On the user details page, choose the **MFA devices** tab, select the device, and then choose **Rename**.
4. When prompted, enter the new name and then choose **Rename**.

To delete an MFA device

1. Open the IAM Identity Center console.
2. In the left navigation pane, choose **Users**. Choose the user in the list.
3. On the user details page, choose the **MFA devices** tab, select the device, and then choose **Delete**.
4. To confirm, type **DELETE**, and then choose **Delete**.

Allow users to register their own MFA devices

Use the following procedure to allow your users to self-register their own MFA devices.

To allow users to register their own MFA devices

1. Open the IAM Identity Center console.
2. In the left navigation pane, choose **Settings**.
3. On the **Settings** page, choose the **Authentication** tab.
4. In the **Multi-factor authentication** section, choose **Configure**.
5. On the **Configure multi-factor authentication** page, under **Who can manage MFA devices**, choose **Users can add and manage their own MFA devices**.
6. Choose **Save changes**.

> **Note**
> After you set up self-registration for your users, you might want to send them a link to the
> procedure Registering a device for MFA (p. 81). This topic provides instructions on how to set
> up their own MFA devices.

Disable MFA

Use the following procedure to disable MFA in the IAM Identity Center console.

To disable MFA

1. Open the IAM Identity Center console.

2. In the left navigation pane, choose **Settings**.

3. On the **Settings** page, choose the **Authentication** tab.

4. In the **Multi-factor authentication** section, choose **Configure**.

5. On the **Confgure multi-factor authentication** page, choose the **Never (disabled)** radio button.

6. Choose **Save changes**.

Multi-account permissions

AWS IAM Identity Center (successor to AWS Single Sign-On) is integrated with AWS Organizations so that administrators can pick multiple AWS accounts whose users need single sign-on (SSO) access to the AWS Management Console. These AWS accounts can be either the management account of the AWS Organizations or a member account. A management account is the AWS account that is used to create the organization. The rest of the accounts that belong to an organization are called member accounts. For more information about the different account types, see AWS Organizations Terminology and Concepts in the *AWS Organizations User Guide*.

After you assign access from the IAM Identity Center console, you can use permission sets to further refine what users can do in the AWS Management Console. For more information about permission sets, see Create and manage permission sets (p. 98).

Users follow a simple sign-in process:

1. Users use their directory credentials to sign in to the AWS access portal.
2. Users then choose the AWS account name that will give them federated access to the AWS Management Console for that account.
3. Users who are assigned multiple permission sets choose which IAM role to use.

Permission sets are a way to define permissions centrally in IAM Identity Center so that they can be applied to all of your AWS accounts. These permission sets are provisioned to each AWS account as an IAM role. The AWS access portal gives users the ability to retrieve temporary credentials for the IAM role of a given AWS account so they can use it for short-term access to the AWS CLI. For more information, see Getting IAM role credentials for CLI access (p. 80).

To use IAM Identity Center with AWS Organizations, you must first enable IAM Identity Center, which grants IAM Identity Center the capability to create Service-linked roles (p. 110) in each account in your organization. These roles are not created until after you Assign user access to AWS accounts (p. 95) for a given account.

You can also connect an AWS account that is not part of your organization by setting up the account as a custom SAML 2.0 application in IAM Identity Center. In this scenario, you provision and manage the IAM roles and trust relationships that are required to enable SSO access. For more information on how to do this, see Add and configure a custom SAML 2.0 application (p. 116).

Topics

Delegated administration

Delegated administration provides a convenient way for assigned users in a registered member account to perform most IAM Identity Center administrative tasks. When you enable IAM Identity Center, your IAM Identity Center instance is created in the management account in AWS Organizations by default.

AWS IAM Identity Center (successor
to AWS Single Sign-On) User Guide
What tasks can be performed in the
delegated administrator account

This was originally designed this way so that IAM Identity Center can provision, de-provision, and update roles across all your organization's member accounts. Even though your IAM Identity Center instance must always reside in the management account, you can choose to delegate administration of IAM Identity Center to a member account in AWS Organizations, thereby extending the ability to manage IAM Identity Center from outside the management account.

Enabling delegated administration provides the following benefits:

- Minimizes the number of people who require access to the management account to help mitigate security concerns
- Allows select administrators to assign users and groups to applications and to your organization's member accounts

For more information about how IAM Identity Center works with AWS Organizations, see Multi-account permissions (p. 91). For additional information and to review an example company scenario showing how to configure delegated administration, see Getting started with IAM Identity Center delegated administration in the *AWS Security Blog*.

What tasks can be performed in the delegated administrator account

To configure delegated administration you must first register a member account in your AWS organization as a delegated administrator, users in that member account who have sufficient permissions will have administrative access to IAM Identity Center. After a member account has been successfully registered for delegated administration, it can then be referred to as the delegated administrator account.

The following table describes the administrative tasks available for a delegated administrator account compared to a management account.

IAM Identity Center administrative tasks	Delegated administrator account	Management account
Add, edit, or delete users or groups	X	X
Enable or disable user access	X	X
Enable, disable, or manage incoming attributes	X	X
Change or manage identity sources	X	X
Create, edit, or delete applications	X	X
Configure MFA	X	X
Manage permission sets not provisioned in the management account	X	X
Manage permission sets provisioned in the management account		X

IAM Identity Center administrative tasks	Delegated administrator account	Management account
Enable IAM Identity Center		X
Delete IAM Identity Center configuration		X
Enable or disable user access in the management account		X
Register or deregister a member account as a delegated administrator		X

Best practices

Here are some best practices to consider before you configure delegated administration.

- **Grant least privilege to the management account** – Knowing that the management account is a highly privileged account and to adhere to the principal of least privilege, we highly recommend that you restrict access to the management account to as few people as possible. The delegated administrator feature is intended to minimize the number of people who require access to the management account.

- **Create permission sets for use only in the management account** – This makes it easier to administer permission sets tailored just for users accessing your management account and helps to differentiate them from permission sets managed by your delegated administrator account.

- **Consider your Active Directory location** – If you plan on using Active Directory as your IAM Identity Center identity source, locate the directory in the member account where you have enabled the IAM Identity Center delegated administrator feature. If you decide to change the IAM Identity Center identity source from any other source to Active Directory, or change it from Active Directory to any other source, the directory must reside in (be owned by) the IAM Identity Center delegated administrator member account if one exists; otherwise, it must be in the management account.

Prerequisites

Before you can register an account as a delegated administrator you must first have the following environment deployed:

- AWS Organizations must be enabled and configured with at least one member account in addition to your default management account.

- If your identity source is set to Active Directory, the IAM Identity Center configurable AD sync (p. 38) feature must be enabled.

  Note
  If your IAM Identity Center configuration was enabled prior to November 2019 and you have not yet enabled access to other accounts in AWS Organizations, you might see an info alert in the console to enable access to IAM Identity Center for a member account in your organization. To provide a registered delegated administrator access to your users and groups and to setup entitlements you have to Allow Identity Center enabled applications in AWS accounts (p. 15). If your IAM Identity Center configuration was enabled after November 2019, there is nothing more for you to do because access to accounts in your organization were automatically enabled by AWS.

Register a member account

IAM Identity Center only supports registering one member account as a delegated administrator at a time. You can only register a member account while signed in with credentials from the management account.

Use the following procedure to grant administrative access to IAM Identity Center by registering a specific member account in your AWS organization as a delegated administrator.

> **Important**
> This operation delegates IAM Identity Center administrative access to admin users in this member account. All users who have sufficient permissions to this delegated administrator account can perform all IAM Identity Center administrative tasks from the account, except for: enabling IAM Identity Center, deleting IAM Identity Center configurations, managing permission sets provisioned in the management account, registering or deregistering other member accounts as delegated administrators, or enabling or disabling user access in the management account.

To register a member account

1. Sign in to the AWS Management Console using the credentials of your management account in AWS Organizations. Management account credentials are required to run the RegisterDelegatedAdministrator API.
2. Select the Region where IAM Identity Center is enabled, and then open the IAM Identity Center console.
3. Choose **Settings**, and then select the **Management** tab.
4. In the **Delegated administrator** section, choose **Register account**.
5. On the **Register delegated administrator** page, select the AWS account you want to register, and then choose **Register account**.

Deregister a member account

You can only deregister a member account while signed in with credentials from the management account.

Use the following procedure to remove administrative access from IAM Identity Center by deregistering a member account in your AWS organization that had previously been designated as a delegated administrator.

> **Important**
> When you deregister an account, you effectively remove the ability for all admin users to manage IAM Identity Center from that account. As a result, they can no longer administer IAM Identity Center identities, access management, authentication, or application access from this account. This operation will not affect any permissions or assignments configured in IAM Identity Center and therefore will have no impact on your end users as they will continue to have access to their apps and AWS accounts from within the AWS access portal.

To deregister a member account

1. Sign in to the AWS Management Console using the credentials of your management account in AWS Organizations. Management account credentials are required to run the DeregisterDelegatedAdministrator API.
2. Select the Region where IAM Identity Center is enabled, and then open the IAM Identity Center console.
3. Choose **Settings**, and then select the **Management** tab.

AWS IAM Identity Center (successor
to AWS Single Sign-On) User Guide
View which member account has been
registered as the delegated administrator

4. In the **Delegated administrator** section, choose **Deregister account**.

5. In the **Deregister account** dialog box, review the security implications, and then enter the name of the member account to confirm that you understand.

6. Choose **Deregister account**.

View which member account has been registered as the delegated administrator

Use the following procedure to find which member account in your AWS Organizations has been configured as the delegated administrator for IAM Identity Center.

To view your registered member account

1. Open the IAM Identity Center console.

2. Choose **Settings**.

3. In the **Details** section, locate the registered account name under **Delegated administrator**. You can also locate this information by selecting the **Management** tab, and viewing it under the **Delegated administrator** section.

Single sign-on access to AWS accounts

You can assign users in your connected directory permissions to the management account or member accounts in your AWS Organizations organization based on common job functions. Or you can use custom permissions to meet your specific security requirements. For example, you can grant database administrators broad permissions to Amazon RDS in development accounts but limit their permissions in production accounts. IAM Identity Center configures all the necessary user permissions in your AWS accounts automatically.

Note
You might need to grant users or groups permissions to operate in the AWS Organizations management account. Because it is a highly privileged account, additional security restrictions require you to have the IAMFullAccess policy or equivalent permissions before you can set this up. These additional security restrictions are not required for any of the member accounts in your AWS organization.

Assign user access to AWS accounts

Use the following procedure to assign single sign-on access to users and groups in your connected directory and use permission sets to determine their level of access.

Note
To simplify administration of access permissions, we recommended that you assign access directly to groups rather than to individual users. With groups you can grant or deny permissions to groups of users rather than having to apply those permissions to each individual. If a user moves to a different organization, you simply move that user to a different group and they automatically receive the permissions that are needed for the new organization.

To assign user or group access to AWS accounts

1. Open the IAM Identity Center console.

   Note
   Make sure that the IAM Identity Center console is using the Region where your AWS Managed Microsoft AD directory is located before you move to the next step.

2. In the navigation pane, under **Multi-account permissions**, choose **AWS accounts**.

3. On the **AWS accounts** page, a tree view list of your organization appears. Select the check box next to one or more AWS accounts to which you want to assign single sign-on access.

    > **Note**
    > You can select up to 10 AWS accounts at a time per permission set when you assign single sign-on access to users and groups. To assign more than 10 AWS accounts to the same set of users and groups, repeat this procedure as required for the additional accounts. When prompted, select the same users, groups, and permission set.

4. Choose **Assign users or groups**.

5. For **Step 1: Select users and groups**, on the **Assign users and groups to "*AWS-account-name*"** page, do the following:

    1. On the **Users** tab, select one or more users to whom to grant single sign-on access.

        To filter the results, start typing the name of the user that you want in the search box.

    2. On the **Groups** tab, select one or more groups to which to grant single sign-on access.

        To filter the results, start typing the name of the group that you want in the search box.

    3. To display the users and groups that you selected, choose the sideways triangle next to **Selected users and groups**.

    4. After you confirm that the correct users and groups are selected, choose **Next**.

6. For **Step 2: Select permission sets**, on the **Assign permission sets to "*AWS-account-name*"** page, do the following:

    1. Select one or more permission sets. If required, you can create and select new permission sets.

        *   To select one or more existing permission sets, under **Permission sets**, select the permission sets that you want to apply to the users and groups that you selected in the previous step.

        *   To create one or more new permission sets, choose **Create permission set**, and follow the steps in Create a permission set (p. 98). After you create the permission sets that you want to apply, in the IAM Identity Center console, return to **AWS accounts** and follow the instructions until you reach **Step 2: Select permission sets**. When you reach this step, select the new permission sets that you created, and proceed to the next step in this procedure.

    2. After you confirm that the correct permission sets are selected, choose **Next**.

7. For **Step 3: Review and Submit**, on the **Review and submit assignments to "*AWS-account-name*"** page, do the following:

    1. Review the selected users, groups, and permission sets.

    2. After you confirm that the correct users, groups, and permission sets are selected, choose **Submit**.

        > **Important**
        > The user and group assignment process might take a few minutes to complete. Leave this page open until the process successfully completes.

        > **Note**
        > You might need to grant users or groups permissions to operate in the AWS Organizations management account. Because it is a highly privileged account, additional security restrictions require you to have the IAMFullAccess policy or equivalent permissions before you can set this up. These additional security restrictions are not required for any of the member accounts in your AWS organization.

Remove user and group access

Use this procedure to remove single sign-on access to an AWS account for one or more users and groups in your connected directory.

AWS IAM Identity Center (successor
to AWS Single Sign-On) User Guide
Delegate who can assign single sign-on access
to users and groups in the management account

To remove user and group access to an AWS account

1. Open the IAM Identity Center console.

2. In the navigation pane, under **Multi-account permissions**, choose **AWS accounts**.

3. On the **AWS accounts** page, a tree view list of your organization appears. Select the name of the AWS account that contains the users and groups for whom you want to remove single sign-on access.

4. On the **Overview** page for the AWS account, under **Assigned users and groups**, select the name of one or more users or groups, and choose **Remove access**.

5. In the **Remove access** dialog box, confirm that the names of the users or groups are correct, and choose **Remove access**.

Delegate who can assign single sign-on access to users and groups in the management account

Assigning single sign-on access to the management account using the IAM Identity Center console is a privileged action. By default, only an AWS account root user, or a user who has the **AWSSSOMasterAccountAdministrator** and **IAMFullAccess** AWS managed policies attached, can assign single sign-on access to the management account. The **AWSSSOMasterAccountAdministrator** and **IAMFullAccess** policies manage SSO access to the management account within an AWS Organizations organization.

Use the following steps to delegate permissions to manage single sign-on access to users and groups in your directory.

To grant permissions to manage single sign-on access to users and groups in your directory

1. Sign in to the IAM Identity Center console as a root user of the management account or with another IAM user who has IAM administrator permissions to the management account.

2. Follow the steps in Create a permission set (p. 98) to create a permission set, and then do the following:

   1. On the **Create new permission set** page, select the **Create a custom permission set** check box, and then choose **Next: Details**.

   2. On the **Create new permission set page**, specify a name for the custom permission set and optionally, a description. If required, modify the session duration and specify a relay state URL.

   3. Under **What policies do you want to include in your permission set?**, select the **Attach AWS managed policies** check box.

   4. In the list of IAM policies, choose both the **AWSSSOMasterAccountAdministrator** and **IAMFullAccess** AWS managed policies. These policies grant permissions to any user and groups who are assigned access to this permission set in the future.

   5. Choose **Next: Tags**.

   6. Under **Add tags (optional)**, specify values for **Key** and **Value (optional)**, and then choose **Next: Review**. For more information about tags, see Tagging AWS IAM Identity Center (successor to AWS Single Sign-On) resources (p. 177).

   7. Review the selections you made, and then choose **Create**.

3. Follow the steps in Assign user access to AWS accounts (p. 95) to assign the appropriate users and groups to the permission set that you just created.

4. Communicate the following to the assigned users: When they sign in to the AWS access portal and select the **AWS Account** icon, they must choose the appropriate role name to be authenticated with the permissions that you just delegated.

Create and manage permission sets

Permission sets define the level of access that users and groups have to an AWS account. Permission sets are stored in IAM Identity Center and can be provisioned to one or more AWS accounts. You can assign more than one permission set to a user. For more information, see Permission sets (p. 16).

Create a permission set

Use this procedure to create a predefined permission set that uses a single AWS managed policy, or a custom permission set that uses up to 10 AWS managed or customer managed policies and an inline policy. You can request an adjustment to the maximum number of 10 policies in the Service Quotas console for IAM.

You can create a permission set in the AWS Management Console.

Use this procedure to create a predefined permission set that uses a single AWS managed policy, or a custom permission set that uses up to 10 AWS managed or customer managed policies and an inline policy.

To create a permission set

1. Open the IAM Identity Center console.
2. Under **Multi-account permissions**, choose **Permission sets**.
3. Choose **Create permission set**.
4. On the **Select permission set type** page, under **Permission set type**, select a permission set type.
5. Choose one or more policies that you want to use for the permission set, based on the permission set type:

   - **Predefined permission set**

     1. Choose **Next**.

     2. Under **Predefined policy**, select one of the IAM **Job function policies** or **Common permission policies** in the list, and then choose **Next**. For more information, see AWS managed policies for job functions and AWS managed policies in the *AWS Identity and Access Management User Guide*.

     3. At the **Review and create** screen, review the selections you made, and then choose **Create**.

   - **Custom permission set**

     1. Choose **Next**.

     2. On the **Specify policies** page, choose the types of IAM policies that you want to apply to your new permission set. By default, you can add any combination of up to 10 **AWS managed policies** and **Customer managed policies** to your permission set. This quota is set by IAM. To raise it, request an increase to the IAM quota *Managed policies attached to an IAM role* in the Service Quotas console in each AWS account where you want to assign the permission set.

        - Expand **AWS managed policies** to add policies from IAM that AWS builds and maintains. For more information, see AWS managed policies (p. 18).

          a. Search for and choose **AWS managed policies** that you want to apply to your users in the permission set.

          b. If you want to add another type of policy, choose its container and make your selection. Choose **Next** when you've chosen all the policies that you want to apply.

        - Expand **Customer managed policies** to add policies from IAM that you build and maintain. For more information, see Customer managed policies (p. 18).

          a. Choose **Attach policies** and enter the name of a policy that you want to add to your permission set. In each account where you want to assign the permission set, create a

      policy with the name you entered. As a best practice, assign the same permissions to the
      policy in each account.

    b. Choose **Attach more** to add another policy.

    c. If you want to add another type of policy, choose its container and make your selection.
      Choose **Next** when you've chosen all the policies that you want to apply.

- Expand **Custom inline policy** to add custom JSON-formatted policy text. Inline policies don't
  correspond to existing IAM resources. To create an inline policy, enter custom policy language
  in the provided form. IAM Identity Center adds the policy to the IAM resources that it creates
  in your member accounts. For more information, see Inline policies (p. 18).

    a. Choose **Design** to use an interactive editor to choose permissions that you want to include
      in your inline policy. Choose **Code** to paste in preformatted policy JSON.

    b. If you want to add another type of policy, choose its container and make your selection.
      Choose **Next** when you've chosen all the policies that you want to apply.

- Expand **Permissions boundary** to add an AWS managed or customer managed IAM policy as
  the maximum permissions that your other policies in the permission set can assign. For more
  information, see Permissions boundaries (p. 19).

    a. Choose **Use a permissions boundary to control the maximum permissions**.

    b. Choose **AWS managed policy** to set a policy from IAM that *AWS* builds and maintains as
      your permissions boundary. Chose **Customer managed policies** to set a policy from IAM
      that *you* build and maintain as your permissions boundary.

    c. If you want to add another type of policy, choose its container and make your selection.
      Choose **Next** when you've chosen all the policies that you want to apply.

6. On the **Specify permission set details** page, do the following:

   1. Under **Permission set name**, type a name to identify this permission set in IAM Identity Center.
      The name that you specify for this permission set appears in the AWS access portal as an available
      role. Users sign into the AWS access portal, choose an AWS account, and then choose the role.

   2. (Optional) You can also type a description. The description appears in the IAM Identity Center
      console only, not the AWS access portal.

   3. (Optional) Specify the value for **Session duration**. This value determines the length of time that
      a user can be logged on before the console logs them out of their session. For more information,
      see Set session duration (p. 101).

   4. (Optional) Specify the value for **Relay state**. This value is used in the federation process to
      redirect users within the account. For more information, see Set relay state (p. 102).

   5. Expand **Tags (optional)**, choose **Add tag**, and then specify values for **Key** and **Value (optional)**.

      For information about tags, see Tagging AWS IAM Identity Center (successor to AWS Single Sign-
      On) resources (p. 177).

   6. Choose **Next**.

7. On the **Review and create** page, review the selections that you made, and then choose **Create**.

8. By default, when you create a permission set, the permission set isn't provisioned (used in any AWS
   accounts). To provision a permission set in an AWS account, you must assign IAM Identity Center
   access to users and groups in the account, and then apply the permission set to those users and
   groups. For more information, see Single sign-on access to AWS accounts (p. 95).

Delegate permission set administration

IAM Identity Center enables you to delegate management of permission sets and assignments in
accounts by creating IAM policies that reference the Amazon Resource Names (ARNs) of IAM Identity
Center resources. For example, you can create policies that enable different administrators to manage
assignments in specified accounts for permission sets with specific tags.

You can use any of the following three methods to create these kinds of policies.

- (Recommended) Create permission sets in IAM Identity Center, each with a different policy, and assign the permission sets to different users or groups. This enables you to manage administrative permissions for users that sign in using your chosen IAM Identity Center identity source.

- Create custom policies in IAM, and then attach them to IAM roles that your administrators assume. This enables IAM users or IAM federated users to assume the role to get their assigned IAM Identity Center administrative permissions.

- Create custom policies in IAM, and then attach them to IAM users that you use for IAM Identity Center administrator purposes. This enables you to give individual IAM users specific IAM Identity Center administrative permissions.

> **Important**
> IAM Identity Center resource ARNs are case sensitive.

The following shows the proper case for referencing the IAM Identity Center permission set and account resource types.

Resource Types	ARN	Context Keys
PermissionSet	arn:${Partition}:sso:::permissionSet/${InstanceId}/${PermissionSetId}	aws:ResourceTag/${TagKey}
Account	arn:${Partition}:sso:::account/${AccountId}	Not Applicable

Use IAM policies in permission sets

In Create a permission set (p. 98), you learned how to add policies, including customer managed policies and permissions boundaries, to a permission set. When you add customer managed policies and permissions to a permission set, IAM Identity Center doesn't create a policy in any AWS accounts. You must instead create those policies in advance in each account where you want to assign your permission set, and match them to the name and path specifications of your permission set. When you assign a permission set to an AWS account in your organization, IAM Identity Center creates an AWS Identity and Access Management (IAM) role and attaches your IAM policies to that role.

> **Note**
> Before you assign your permission set with IAM policies, you must prepare your member account. The name of an IAM policy in your member account must be a case-sensitive match to name of the policy in your management account. IAM Identity Center fails to assign the permission set if the policy doesn't exist in your member account.
> The permissions that the policy grants don't have to be an exact match between accounts.

To assign an IAM policy to a permission set

1. Create an IAM policy in each of the AWS accounts where you want to assign the permission set.

2. Assign permissions to the IAM policy. You can assign different permissions in different accounts. For a consistent experience, configure and maintain identical permissions in each policy. You can use automation resources like AWS CloudFormation StackSets to create copies of an IAM policy with the same name and permissions in each member account. For more information about CloudFormation StackSets, see Working with AWS CloudFormation StackSets in the *AWS CloudFormation User guide*.

3. Create a permission set in your management account and add your IAM policy under **Customer managed policies** or **Permissions boundary**. For more details about how to create a permission set, See Create a permission set (p. 98).

4. Add any inline policies, AWS managed policies, or additional IAM policies that you have prepared.

5. Create and assign your permission set.

Configure permission set properties

In IAM Identity Center you can customize the user experience by configuring the following permission set properties.

Topics

Set session duration

For each permission set, you can specify a session duration to control the length of time that a user can be signed in to an AWS account. When the specified duration elapses, AWS signs the user out of the session.

When you create a new permission set, the session duration is set to 1 hour (in seconds) by default. The minimum session duration is 1 hour, and can be set to a maximum of 12 hours. IAM Identity Center automatically creates IAM roles in each assigned account for each permission set, and configures these roles with a maximum session duration of 12 hours.

When users federate into their AWS account console or when the AWS Command Line Interface (AWS CLI) is used, IAM Identity Center uses the session duration setting on the permission set to control the duration of the session. By default, IAM roles generated by IAM Identity Center for permission sets can only be assumed by IAM Identity Center users, which ensures that the session duration specified in the IAM Identity Center permission set is enforced.

> **Important**
> As a security best practice, we recommend that you do not set the session duration length longer than is needed to perform the role.

After you create a permission set, you can update it to apply a new session duration. Use the following procedure to modify the session duration length for a permission set.

To set the session duration

1. Open the IAM Identity Center console.

2. Under **Multi-account permissions**, choose **Permission sets**.

3. Choose the name of the permission set for which you want to change the session duration.

4. On the details page for the permission set, to the right of the **General settings** section heading, choose **Edit**.

5. On the **Edit general permission set settings** page, choose a new value for **Session duration**.

6. If the permission set is provisioned in any AWS accounts, the names of the accounts appear under **AWS accounts to reprovision automatically**. After the session duration value for the permission set is updated, all AWS accounts that use the permission set are reprovisioned. This means that the new value for this setting is applied to all AWS accounts that use the permission set.

7. Choose **Save changes**.

8. At the top of the **AWS accounts** page, a notification appears.

   - If the permission set is provisioned in one or more AWS accounts, the notification confirms that the AWS accounts were reprovisioned successfully, and the updated permission set was applied to the accounts.
   - If the permission set isn't provisioned in an AWS account, the notification confirms that the settings for the permission set were updated.

Set relay state

By default, when a user signs into the AWS access portal, chooses an account, and then chooses the role that AWS creates from the assigned permission set, IAM Identity Center redirects the user's browser to the AWS Management Console. You can change this behavior by setting the relay state to a different console URL. Setting the relay state enables you to provide the user with quick access to the console that is most appropriate for their role. For example, you can set the relay state to the Amazon EC2 console URL (`https://console.aws.amazon.com/ec2/`) to redirect the user to that console when they choose the Amazon EC2 administrator role. During the redirection to the default URL or relay state URL, IAM Identity Center routes the user's browser to the console endpoint in the last AWS Region used by the user. For example, if a user ended their last console session in the Europe (Stockholm) Region (eu-north-1), the user is redirected to the Amazon EC2 console in that Region.

To configure IAM Identity Center to redirect the user to a console in a specific AWS Region, include the Region specification as part of the URL. For example, to redirect the user to the Amazon EC2 console in the US East (Ohio) Region (us-east-2), specify the URL for the Amazon EC2 console in that Region (`https://us-east-2.console.aws.amazon.com/ec2/`). If you enabled IAM Identity Center in the US West (Oregon) Region (us-west-2) Region and you want to direct the user to that Region, specify `https://us-west-2.console.aws.amazon.com`.

Use the following procedure to configure the relay state URL for a permission set.

To configure the relay state

1. Open the IAM Identity Center console.
2. Under **Multi-account permissions**, choose **Permission sets**.
3. Choose the name of the permission set for which you want to set the new relay state URL.
4. On the details page for the permission set, to the right of the **General settings** section heading, choose **Edit**.
5. On the **Edit general permission set settings** page, under **Relay state**, type a URL for any of the AWS services.
6. If the permission set is provisioned in any AWS accounts, the names of the accounts appear under **AWS accounts to reprovision automatically**. After the relay state URL for the permission set is updated, all AWS accounts that use the permission set are reprovisioned. This means that the new value for this setting is applied to all AWS accounts that use the permission set.
7. Choose **Save changes**.
8. At the top of the **AWS Organization** page, a notification appears.

   - If the permission set is provisioned in one or more AWS accounts, the notification confirms that the AWS accounts were reprovisioned successfully, and the updated permission set was applied to the accounts.

- If the permission set isn't provisioned in an AWS account, the notification confirms that the settings for the permission set were updated.

Note
You can automate this process by using the AWS API, an AWS SDK, or the AWS Command Line Interface(AWS CLI). For more information, see:

- The `CreatePermissionSet` or `UpdatePermissionSet` actions in the IAM Identity Center API Reference
- The `create-permission-set` or `update-permission-set` commands in the sso-admin section of the *AWS CLI Command Reference*.

Delete permission sets

Before you can delete a permission set from IAM Identity Center, you must remove it from all AWS accounts that use the permission set.

To remove a permission set from an AWS account

1. Open the IAM Identity Center console.
2. Under **Multi-account permissions**, choose **AWS accounts**.
3. On the **AWS accounts** page, a tree view list of your organization appears. Select the name of the AWS account from which you want to remove the permission set.
4. On the **Overview** page for the AWS account, choose the **Permission sets** tab.
5. Select the check box next to the permission set that you want to remove, and then choose **Remove**.
6. In the **Remove permission set** dialog box, confirm that the correct permission set is selected, type **Delete** to confirm removal, and then choose **Remove access**.

Use the following procedure to delete one or more permission sets so that they can no longer be used by any AWS account in the organization.

> **Note**
> All users and groups that have been assigned this permission set, regardless of what AWS account is using it, will no longer be able to sign in.

To delete a permission set from an AWS account
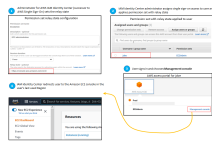
1. Open the IAM Identity Center console.
2. Under **Multi-account permissions**, choose **Permission sets**.
3. Select the permission set that you want to delete, and then choose **Delete**.
4. In the **Delete permission set** dialog box, type the name of the permission set to confirm deletion, and then choose **Delete**. The name is case-sensitive.

Attribute-based access control

Attribute-based access control (ABAC) is an authorization strategy that defines permissions based on attributes. You can use IAM Identity Center to manage access to your AWS resources across multiple AWS accounts using user attributes that come from any IAM Identity Center identity source. In AWS, these attributes are called tags. Using user attributes as tags in AWS helps you simplify the process of creating fine-grained permissions in AWS and ensures that your workforce gets access only to the AWS resources with matching tags.

For example, you can assign developers Bob and Sally, who are from two different teams, to the same permission set in IAM Identity Center and then select the team name attribute for access control. When Bob and Sally sign in to their AWS accounts, IAM Identity Center sends their team name attribute in the AWS session so Bob and Sally can access AWS project resources only if their team name attribute matches the team name tag on the project resource. If Bob moves to Sally's team in the future, you can modify his access by simply updating his team name attribute in the corporate directory. When Bob signs in next time, he will automatically get access to the project resources of his new team without requiring any permissions updates in AWS.

This approach also helps in reducing the number of distinct permissions you need to create and manage in IAM Identity Center as users associated with the same permission sets can now have unique permissions based on their attributes. You can use these user attributes in IAM Identity Center permission sets and resource-based policies to implement ABAC to AWS resources and simplify permissions management at scale.

Benefits

The following are additional benefits of using ABAC in IAM Identity Center.

- **ABAC requires fewer permission sets** – Because you don't have to create different policies for different job functions, you create fewer permission sets. This reduces your permissions management complexity.
- **Using ABAC, teams can change and grow quickly** – Permissions for new resources are automatically granted based on attributes when resources are appropriately tagged upon creation.
- **Use employee attributes from your corporate directory with ABAC** – You can use existing employee attributes from any identity source configured in IAM Identity Center to make access control decisions in AWS.
- **Track who is accessing resources** – Security administrators can easily determine the identity of a session by looking at the user attributes in AWS CloudTrail to track user activity in AWS.

For information about how to configure ABAC using the IAM Identity Center console, see Attributes for access control (p. 106). For information about how to enable and configure ABAC using the IAM Identity Center APIs, see CreateInstanceAccessControlAttributeConfiguration in the *IAM Identity Center API Reference Guide*.

Topics

Checklist: Configuring ABAC in AWS using IAM Identity Center

This checklist includes the configuration tasks that are necessary to prepare your AWS resources and to set up IAM Identity Center for ABAC access. Complete the tasks in this checklist in order. When a reference link takes you to a topic, return back to this topic so that you can proceed with the remaining tasks in this checklist.

Step	Task	Reference
1	Review how to add tags to all your AWS resources. To implement ABAC in IAM Identity Center, you'll first need	• Tagging AWS resources

Step	Task	Reference
	to add tags to all your AWS resources that you want to implement ABAC for.	
2	Review how to configure your identity source in IAM Identity Center with the associated user identities and attributes in your identity store. IAM Identity Center lets you use user attributes from any supported IAM Identity Center identity source for ABAC in AWS.	• Manage your identity source (p. 21)
3	Based on the following criteria, determine which attributes you want to use for making access control decisions in AWS and send them to IAM Identity Center.	• Getting started (p. 106)
	• If you are using an external identity provider (IdP), decide whether you want to use attributes passed from the IdP or select attributes from within IAM Identity Center.	• Choosing attributes when using an external identity provider as your identity source (p. 106)
	• If you choose to have your IdP send attributes, configure your IdP to transmit the attributes in SAML assertions. See the `Optional` sections in the procedure for your specific IdP.	• Supported identity providers (p. 52)
	• If you use an IdP as your identity source and choose to select attributes in IAM Identity Center, investigate how to configure SCIM so that the attribute values come from your IdP. If you cannot use SCIM with your IdP, add the users and their attributes using the IAM Identity Center console **User** page.	• Automatic provisioning (p. 47) • Supported external identity provider attributes (p. 36)
	• If you use Active Directory or IAM Identity Center as your identity source, or you use an IdP and choose to select attributes in IAM Identity Center, review the available attributes that you can configure. Then jump immediately to step 4 to start configuring your ABAC attributes using the IAM Identity Center console.	• Choosing attributes when using IAM Identity Center as your identity source (p. 106) • Choosing attributes when using AWS Managed Microsoft AD as your identity source (p. 106) • Default mappings (p. 37)
4	Select the attributes to use for ABAC using the **Attributes for access control** page in the IAM Identity Center console. From this page you can select attributes for access control from the identity source that you configured in step 2. After your identities and their attributes are in IAM Identity Center, you must create key-value pairs (mappings) which will be passed to your AWS accounts for use in access control decisions.	• Enable and configure attributes for access control (p. 107)
5	Create custom permissions policies within your permission set and use access control attributes to create ABAC rules so that users can only access resources with matching tags. User attributes that you configured in step 4 are used as tags in AWS for access control decisions. You can refer to the access control attributes in the permissions policy using the `aws:PrincipalTag/key` condition.	• Create permission policies for ABAC in IAM Identity Center (p. 109)

Step	Task	Reference
6	In your various AWS accounts, assign users to permissions sets you created in step 5. Doing so ensures that when they federate into their accounts and access AWS resources, they only get access based on matching tags.	• Assign user access to AWS accounts (p. 95)

Once you have completed this checklist, users that federate into an AWS account using single sign-on will get access to their AWS resources based on matching attributes.

Attributes for access control

Attributes for access control is the name of the page in the IAM Identity Center console where you select user attributes that you want to use in policies to control access to resources. Administrators can assign users to workloads in AWS based on existing attributes in the users' identity source.

For example, suppose you want to assign access to S3 buckets based on department names. While on the **Attributes for access control** page, you select the **Department** user attribute for use with attribute-based access control (ABAC). In the IAM Identity Center permission set, you then write a policy that grants users access only when the **Department** attribute matches the department tag that you assigned to your S3 buckets. IAM Identity Center passes the user's department attribute to the account being accessed. The attribute is then used to determine access based on the policy. For more information about ABAC, see Attribute-based access control (p. 103).

Getting started

How you get started configuring attributes for access control depends on which identity source you are using. Regardless of the identity source you choose, after you have selected your attributes you need to create or edit permission set policies. These policies must grant user identities access to AWS resources.

Choosing attributes when using IAM Identity Center as your identity source

When you configure IAM Identity Center as the identity source, you first add users and configure their attributes. Next, navigate to the **Attributes for access control** page and select the attributes you want to use in policies. Finally, navigate to the **AWS accounts** page to create or edit permission sets to use the attributes for ABAC.

Choosing attributes when using AWS Managed Microsoft AD as your identity source

When you configure IAM Identity Center with AWS Managed Microsoft AD as your identity source, you first map a set of attributes from Active Directory to user attributes in IAM Identity Center. Next, navigate to the **Attributes for access control** page. Then choose which attributes to use in your ABAC configuration based on the existing set of SSO attributes mapped from Active Directory. Finally, author ABAC rules using the access control attributes in permission sets to grant user identities access to AWS resources. For a list of the default mappings for user attributes in IAM Identity Center to the user attributes in your AWS Managed Microsoft AD directory, see Default mappings (p. 37).

Choosing attributes when using an external identity provider as your identity source

When you configure IAM Identity Center with an external identity provider (IdP) as your identity source, there are two ways to use attributes for ABAC.

• You can configure your IdP to send the attributes through SAML assertions. In this case, IAM Identity Center passes the attribute name and value from the IdP through for policy evaluation.

Note

Attributes in SAML assertions will not be visible to you on the **Attributes for access control** page. You will have to know these attributes in advance and add them to access control rules when you author policies. If you decide to trust your external IdPs for attributes, then these attributes will always be passed when users federate into AWS accounts. In scenarios where the same attributes are coming to IAM Identity Center through SAML and SCIM, the SAML attributes value take precedence in access control decisions.

- You can configure which attributes you use from the **Attributes for access control** page in the IAM Identity Center console. The attributes values that you choose here replace the values for any matching attributes that come from an IdP through an assertion. Depending on whether you are using SCIM, consider the following:

  - If using SCIM, the IdP automatically synchronizes the attribute values into IAM Identity Center. Additional attributes that are required for access control might not be present in the list of SCIM attributes. In that case, consider collaborating with the IT admin in your IdP to send such attributes to IAM Identity Center via SAML assertions using the required `https://aws.amazon.com/SAML/Attributes/AccessControl:` prefix. For information about how to configure user attributes for access control in your IdP to send through SAML assertions, see Supported identity providers (p. 52).

  - If you are not using SCIM, you must manually add the users and set their attributes just as if you were using IAM Identity Center as an identity source. Next, navigate to the **Attributes for access control** page and choose the attributes you want to use in policies.

For a complete list of supported attributes for user attributes in IAM Identity Center to the user attributes in your external IdPs, see Supported external identity provider attributes (p. 36).

To get started with ABAC in IAM Identity Center, see the following topics.

Topics

Enable and configure attributes for access control

To use ABAC in all cases, you must first enable ABAC using the IAM Identity Center console or the IAM Identity Center API. If you choose to use IAM Identity Center to select attributes, you use the **Attributes for access control** page in the IAM Identity Center console or the IAM Identity Center API. If you use an external identity provider (IdP) as an identity source and choose to send attributes through the SAML assertions, you configure your IdP to pass the attributes. If a SAML assertion passes any of these attributes, IAM Identity Center will replace the attribute value with the value from the IAM Identity Center identity store. Only attributes configured in IAM Identity Center will be sent over for making access control decisions when users federate into their accounts.

Note

You cannot view attributes configured and sent by an external IdP from the **Attributes for access control** page in the IAM Identity Center console. If you are passing access control attributes in the SAML assertions from your external IdP, then those attributes are directly sent to the AWS account when users federate in. The attributes won't be available in IAM Identity Center for mapping.

Enable attributes for access control

Use the following procedure to enable the attributes for access (ABAC) control feature using the IAM Identity Center console.

Note

If you have existing permission sets and you plan to enable ABAC in your IAM Identity Center instance, additional security restrictions require you to first have the

`iam:UpdateAssumeRolePolicy` policy. These additional security restrictions are not required if you do not have any permission sets created in your account.

To enable Attributes for access control

1. Open the IAM Identity Center console.
2. Choose **Settings**
3. On the **Settings** page, locate the **Attributes for access control** information box, and then choose **Enable**. Continue to the next procedure to configure it.

Select your attributes

Use the following procedure to set up attributes for your ABAC configuration.

To select your attributes using the IAM Identity Center console

1. Open the IAM Identity Center console.
2. Choose **Settings**
3. On the **Settings** page, choose the **Attributes for access control** tab, and then choose **Manage attributes**.
4. On the **Attributes for access control** page, choose **Add attribute** and enter the **Key** and **Value** details. This is where you will be mapping the attribute coming from your identity source to an attribute that IAM Identity Center passes as a session tag.

Key ⓘ	Value (optional) ⓘ	Remove
Department	${path:enterprise.department}	✖
CostCenter	${path:enterprise.costCenter}	✖
Add new key	Add new value	

Key represents the name you are giving to the attribute for use in policies. This can be any arbitrary name, but you need to specify that exact name in the policies you author for access control. For example, lets say that you are using Okta (an external IdP) as your identity source and need to pass your organization's cost center data along as session tags. In **Key**, you would enter a similarly matched name like **CostCenter** as your key name. It's important to note that whichever name you choose here, it must also be named exactly the same in your aws:PrincipalTag condition key (p. 109) (that is, "ec2:ResourceTag/CostCenter": "${aws:PrincipalTag/CostCenter}").

Value represents the content of the attribute coming from your configured identity source. Here you can enter any value from the appropriate identity source table listed in Attribute mappings (p. 34). For example, using the context provided in the above mentioned example, you would review the list of supported IdP attributes and determine that the closest match of a supported attribute would be **${path:enterprise.costCenter}** and you would then enter it in the **Value** field. See the screenshot provided above for reference. Note, that you can't use external IdP attribute values outside of this list for ABAC unless you use the option of passing attributes through the SAML assertion.

5. Choose **Save changes**.

Now that you have configured mapping your access control attributes, you need to complete the ABAC configuration process. To do this, author your ABAC rules and add them to your permission sets and/or resource-based policies. This is required so that you can grant user identities access to AWS resources. For more information, see Create permission policies for ABAC in IAM Identity Center (p. 109).

Disable attributes for access control

Use the following procedure to disable the ABAC feature and delete all of the attribute mappings that have been configured.

To disable Attributes for access control

1. Open the IAM Identity Center console.
2. Choose **Settings**
3. On the **Settings** page, choose the **Attributes for access control** tab, and then choose **Disable**.
4. In the **Disable attributes for access control** dialog, review the information and when ready enter DELETE, and then choose **Confirm**.

   > **Important**
   > This step deletes all attributes that have been configured. Once deleted, any attributes that are received from an identity source and any custom attributes you have previously configured will not be passed.

Create permission policies for ABAC in IAM Identity Center

You can create permissions policies that determine who can access your AWS resources based on the configured attribute value. When you enable ABAC and specify attributes, IAM Identity Center passes the attribute value of the authenticated user into IAM for use in policy evaluation.

aws:PrincipalTag condition key

You can use access control attributes in your permission sets using the `aws:PrincipalTag` condition key for creating access control rules. For example, in the following trust policy you can tag all the resources in your organization with their respective cost centers. You can also use a single permission set that grants developers access to their cost center resources. Now, whenever developers federate into the account using single sign-on and their cost center attribute, they only get access to the resources in their respective cost centers. As the team adds more developers and resources to their project, you only have to tag resources with the correct cost center. Then you pass cost center information in the AWS session when developers federate into AWS accounts. As a result, as the organization adds new resources and developers to the cost center, developers can manage resources aligned to their cost centers without needing any permission updates.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:StartInstances",
                "ec2:StopInstances"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "ec2:ResourceTag/CostCenter": "${aws:PrincipalTag/CostCenter}"
                }
```

```
                }
            }
        ]
}
```

For more information, see aws:PrincipalTag and EC2: Start or stop instances based on matching principal and resource tags in the *IAM User Guide*.

If policies contain invalid attributes in their conditions, then the policy condition will fail and access will be denied. For more information, see Error 'An unexpected error has occurred' when a user tries to sign in using an external identity provider (p. 189).

IAM identity provider

When you add single sign-on access to an AWS account, IAM Identity Center creates an IAM identity provider in each AWS account. An IAM identity provider helps keep your AWS account secure because you don't have to distribute or embed long-term security credentials, such as IAM access keys, in your application.

Repair the IAM identity provider

If you accidentally delete or modify your identity provider, you must manually reapply your user and group assignments. Reapplying your user and group assignments recreates the identity provider. For more information, see:

- Multi-account permissions (p. 91)
- Application assignments (p. 111)

Service-linked roles

Service-linked roles are predefined IAM permissions that allow IAM Identity Center to delegate and enforce which users have single sign-on access to specific AWS accounts in your organization in AWS Organizations. The service enables this functionality by provisioning a service-linked role in every AWS account within its organization. The service then allows other AWS services like IAM Identity Center to leverage those roles to perform service-related tasks. For more information, see AWS Organizations and service-linked roles.

When you enable IAM Identity Center, IAM Identity Center creates a service-linked role in all accounts within the organization in AWS Organizations. IAM Identity Center also creates the same service-linked role in every account that is subsequently added to your organization. This role allows IAM Identity Center to access each account's resources on your behalf. For more information, see Multi-account permissions (p. 91).

Service-linked roles that are created in each AWS account are named `AWSServiceRoleForSSO`. For more information, see Using service-linked roles for IAM Identity Center (p. 151).

Application assignments

With AWS IAM Identity Center (successor to AWS Single Sign-On), you can easily control who can have single sign-on access to your cloud applications. Users get one-click access to these applications after they use their directory credentials to sign in to their AWS access portal.

IAM Identity Center securely communicates with these applications through a trusted relationship between IAM Identity Center and the application's service provider. This trust is created when you add the application from the IAM Identity Center console and configure it with the appropriate metadata for both IAM Identity Center and the service provider.

After the application has been successfully added to the IAM Identity Center console, you can manage which users or groups need permissions to the application. By default, when you add an application, no users are assigned to the application. In other words, newly added applications in the IAM Identity Center console are inaccessible until you assign users to them. IAM Identity Center supports the following applications types:

- Identity Center enabled applications
- Cloud applications
- Custom Security Assertion Markup Language (SAML 2.0) applications

You can also grant your employees access to the AWS Management Console for a given AWS account in your organization. For more information on how to do this, see Multi-account permissions (p. 91).

The following sections explain how to configure user access to your AWS applications and third-party software as a service (SaaS) applications. You can also configure any custom applications that support identity federation with SAML 2.0.

Topics

Identity Center enabled applications

With Identity Center enabled applications (p. 14), AWS enterprise applications such as Amazon SageMaker or AWS IoT SiteWise, can exist in a child account in your organization but still use your IAM Identity Center identities. This provides your application end users with an easy sign-in experience and allows for delegation of administrator of your applications to operators in a child account.

Constraining Identity Center enabled application use in AWS accounts

If you want to constrain which of your AWS Organizations accounts that an Identity Center enabled application can be used, you can do so using service control policies (SCPs). You can use SCPs to block

access to the IAM Identity Center user and group information and to prevent the application from being started except in designated accounts.

Add and configure an Identity Center enabled application

To use Identity Center enabled applications, you must first enable IAM Identity Center to allow them access. For more information, see Identity Center enabled applications (p. 14).

After they are enabled, Identity Center enabled applications can access user and group information directly from IAM Identity Center. As a result, you won't have to manage access in both IAM Identity Center and then again inside the application. Instead, IAM Identity Center delegates application access to the application administrator. To add users to Identity Center enabled applications, use the console of the application where you created the application.

To add and configure your application

1. Open the IAM Identity Center console.
2. Choose **Applications**.
3. Choose **Add application**.
4. Under **Applications**, search for an application, and select the application from the list. Choose **Next**.
5. Under **Configure application**, the **Display name** and **Description** pre-populates with the application you chose. You can edit these. Under **IAM Identity Center metadata**, download or copy any certificate you might need. Under **Application properties**, optionally fill out the fields. Under **Application metadata**, fill out all the fields. Then, choose **Submit**. You're taken to the details page of the application that you just added.

Remove an Identity Center enabled application

To remove an Identity Center enabled application, you can remove the application from the IAM Identity Center console. This action is irreversible and you might not be able to recover data from the application.

> **Warning**
> Removing an application deletes all user permissions to this application, disconnects the application from IAM Identity Center, and renders the application inaccessible.

To remove an AWS application

1. Open the IAM Identity Center console.
2. Choose **Applications**.
3. On the **Applications** page, under **Configured applications**, choose the application that you want to remove.
4. With the application selected, choose **Actions** and in the dropdown, choose **Remove**.
5. A **Remove application** dialog box appears. Follow the prompt to type and confirm the application that you want to remove. Choose **Remove application**.

Cloud applications

You can use the IAM Identity Center application configuration wizard to include built-in SAML integrations to many popular cloud applications. Examples include Salesforce, Box, and Office 365. For a complete list of applications that you can add from the wizard, see Supported applications (p. 113).

Most cloud applications come with detailed instructions on how to set up the trust between IAM Identity Center and the application's service provider. You can find these instructions on the cloud applications configuration page during the setup process and after the application has been set up. After the application has been configured, you can assign access to the groups or users that require it.

Supported applications

IAM Identity Center has built-in support for the following commonly used cloud applications.

> **Note**
> AWS Support engineers can assist customers who have Business and Enterprise support plans with some integration tasks that involve third-party software. For a current list of supported platforms and applications, see Third-Party Software Support on the *AWS Support Features* page.

10000ft	Cybozu Mailwise	Heap	Pivotal Tracker	Squadcast
4me	Cybozu Office	HelloSign	PlanMyLeave	Stackify
7Geese	Cybozu.com	Helpdocs.io	PolicyIQ	Status Hero
Abstract	Dashlane	HelpScout	ProcessPlan	StatusCast
Accredible	Databricks	HighGear	ProdPad	StatusDashboard
Adobe Connect	Datadog	Hightail	Proto.io	StatusHub
Adobe Creative Cloud	Declaree	Honey	Proxyclick	Statuspage
Adobe Sign	Deputy	Honeycomb.io	PurelyHR	StoriesOnBoard
Aha	DeskPro	HostedGraphite	Quip	Stormboard
Airbrake	Deskradar	HubSpot	Rapid7 Insight products	SugarCRM
AlertOps	Detectify	Humanity	Recognize	SumoLogic
AlertSite	Digicert	IdeaScale	Redash.io	SurveyGizmo
Amazon Business	Dmarcian	Igloo	Redlock	SurveyMonkey
Amazon WorkLink	Docebo	ImageRelay	RescueAssist	Syncplicity
Andfrankly	DocuSign	iSpring	RingCentral	Tableau
AnswerHub	Dome9	IT Glue	Roadmunk	Tableau Server
AppDynamics	Domo	JamaSoftware	Robin	TalentLMS
AppFollow	Drift	Jamf	Rollbar	TargetProcess
Area 1	Dropbox	Jenkins	Room Booking System	TeamSupport
Asana	DruvaInSync	JFrog Artifactory	Salesforce	Tenable.io
Assembla	Duo	Jira	Salesforce Service Cloud	TextMagic

Atlassian	Dynatrace	Jitbit	Samanage	ThousandEyes
Automox	EduBrite	Jive	SAP BW	TinfoilSecurity
BambooHR	Egnyte	join.me	SAP Cloud Platform	TitanFile
BenSelect	eLeaP	Kanbanize	SAP CRM ABAP	TOPdesk Operator
BitaBIZ	Engagedly	Keeper Security	SAP CRM Java	TOPdesk Self Service Desk
Bitglass	Envoy	Kentik	SAP Enterprise Portal Java	Trakdesk
BlueJeans	Evernote	Kintone	SAP ERP ABAP	Trello
BMCRemedyforce	ExpenseIn	Klipfolio	SAP EWM ABAP	Trend Micro Deep Security
Bonusly	Expensify	KnowledgeOwl	SAP Fiori ABAP	Uptime.com
Box	Expiration Reminder	Kudos	SAP GRC Access Control ABAP	Uptrends
Brandfolder	External AWS Account	LiquidFiles	SAP LMS	UserEcho
Breezy HR	EZOfficeInventory	LiquidPlanner	SAP Netweaver ABAP	UserVoice
Buddy Punch	EZRentOut	Litmos	SAP Netweaver Java	Velpic
Bugsee	Fastly	LiveChat	SAP S4 ABAP	Veracode
BugSnag	Federated Directory	LogMeInRescue	SAP Solution Manager ABAP	VictorOps
Buildkite	FileCloud	Lucidchart	SAP Solution Manager Java	vtiger
Bynder	FireHydrant	ManageEngine	SAP SRM ABAP	WayWeDo
CakeHR	Fivetran	MangoApps	SAP xMII Java	WeekDone
Canvas	Flock	Marketo	ScaleFT	WhosOnLocation
Chartio	FogBugz	Metricly	Scalyr	Wordbee
Chatwork	Formstack	Miro	ScreenSteps	Workable
Circonus	Fossa	MockFlow	Seeit	Workfront
Cisco Webex	Freedcamp	Mode Analytics	Sentry.io	Workplace by Facebook
CiscoMeraki	Freshdesk	MongoDB	ServiceNow	Workstars
CiscoUmbrella	FreshService	Moodle	SimpleMDM	Wrike
CitrixShareFile	Front	MuleSoft Anypoint	Skeddly	xMatters

Clari	G Suite	MyWebTimeSheets	Skilljar	XperienceHR
Clarizen	Genesys Cloud	N2F Expense Reports	Slack	Yodeck
ClickTime	GitBook	NewRelic	Slemma	Zendesk
Cloud CMS	Github	Nuclino	Sli.do	Zephyr
Cloud Conformity	GitLab	Office365	Small Improvements	Ziflow
CloudAMQP	Glasscubes	OnDMARC	Smartsheet	Zillable
CloudCheckr	GlassFrog	OpenVoice	SnapEngage	Zoho
CloudEndure	GorillaStack	OpsGenie	Snowflake	Zoho One
CloudHealth	GoToAssist	Pacific Timesheet	SonarQube	Zoom
CloudPassage	GoToMeeting	PagerDuty	SparkPost	
CMNTY	GoToTraining	Panopta	Spinnaker	
CoderPad	GoToWebinar	Panorama9	Split.io	
Confluence	Grovo	ParkMyCloud	Splunk Cloud	
Convo	HackerOne	Peakon	Splunk Enterprise	
Coralogix	HackerRank	PhraseApp	Spotinst	
Cybozu Garoon	HappyFox	PipeDrive	SproutVideo	

Add and configure a cloud application

Use this procedure when you need to set up a SAML trust relationship between IAM Identity Center and your cloud application's service provider. Before you begin this procedure, make sure you have the service provider's metadata exchange file so that you can more efficiently set up the trust. If you do not have this file, you can still use this procedure to configure it manually.

To add and configure a cloud application

1. Open the IAM Identity Center console.
2. Choose **Applications**.
3. Choose **Add application**.
4. Under **Applications**, search for an application, and select the application from the list. Choose **Next**.
5. Under **Configure application**, the **Display name** and **Description** pre-populates with the application you chose. You can edit these.
6. Under **IAM Identity Center metadata**, do the following:

   a. Under **IAM Identity Center SAML metadata file**, choose **Download** to download the identity provider metadata.

   b. Under **IAM Identity Center certificate**, choose **Download certificate** to download the identity provider certificate.

Note
You will need these files later when you set up the cloud application from the service provider's website. Follow the instructions from that provider.

7. (Optional) Under **Application properties**, you can specify additional properties for the **Application start URL**, **Relay state**, and **Session duration**. For more information, see Application properties (p. 119).

8. Under **Application metadata**, do one of the following:

    a. Choose **Upload application SAML metadata file**. Then, select **Choose file** to find and select the metadata file.

    b. If you do not have a metadata file, choose **Manually type your metadata values**, and then provide the **Application ACS URL** and **Application SAML audience** values.

9. Choose **Submit**. You're taken to the details page of the application that you just added.

Custom SAML 2.0 applications

You can use the IAM Identity Center application configuration wizard to add support for applications that allow identity federation using Security Assertion Markup Language (SAML) 2.0. In the console, you set these up by choosing **Custom SAML 2.0 application** from the application selector. Most of the steps for configuring a custom SAML application are the same as configuring a cloud application.

However, you also need to provide additional SAML attribute mappings for a custom SAML application. These mappings tell IAM Identity Center how to populate the SAML assertion correctly for your application. You can provide this additional SAML attribute mapping when you set up the application for the first time. You can also provide SAML attribute mappings on the application detail page that is accessible from the IAM Identity Center console.

Add and configure a custom SAML 2.0 application

Use this procedure when you need to set up a SAML trust relationship between IAM Identity Center and your custom application's service provider. Before you begin this procedure, make sure that you have the service provider's certificate and metadata exchange files so that you can finish setting up the trust.

To add and configure a custom SAML application

1. Open the IAM Identity Center console.
2. Choose **Applications**.
3. Choose **Add application**.
4. On the **Select an application** page, choose **Add custom SAML 2.0 application**. Then, choose **Next**.
5. On the **Configure application** page, under **Configure application**, enter a **Display name** for the application, such as **MyApp**. Then, enter a **Description**.
6. Under **IAM Identity Center metadata**, do the following:

    a. Under **IAM Identity Center SAML metadata file**, choose **Download** to download the identity provider metadata.

    b. Under **IAM Identity Center certificate**, choose **Download certificate** to download the identity provider certificate.

    Note
    You will need these files later when you set up the custom application from the service provider's website.

7. (Optional) Under **Application properties**, you can specify additional properties for the **Application start URL**, **Relay State**, and **Session Duration**. For more information, see Application properties (p. 119).

8. Under **Application metadata**, choose **Manually type your metadata values**. Then, provide the **Application ACS URL** and **Application SAML audience** values.

9. Choose **Submit**. You're taken to the details page of the application that you just added.

Manage IAM Identity Center certificates

IAM Identity Center uses certificates to set up a SAML trust relationship between IAM Identity Center and your cloud application's service provider. When you add an application in IAM Identity Center, an IAM Identity Center certificate is automatically created for use with that application during the setup process. By default, this autogenerated IAM Identity Center certificate is valid for a period of five years.

As an IAM Identity Center administrator, you'll occasionally need to replace older certificates with newer ones for a given application. For example, you might need to replace a certificate when the expiration date on the certificate approaches. The process of replacing an older certificate with a newer one is referred to as *certificate rotation*.

Topics

- Considerations before rotating a certificate (p. 117)
- Rotate an IAM Identity Center certificate (p. 117)
- Certificate expiration status indicators (p. 119)

Considerations before rotating a certificate

Before you start the process of rotating a certificate in IAM Identity Center, consider the following:

- The certification rotation process requires that you reestablish the trust between IAM Identity Center and the service provider. To reestablish the trust, use the procedures provided in Rotate an IAM Identity Center certificate (p. 117).

- Updating the certificate with the service provider may cause a temporary service disruption for your users until the trust has been successfully reestablished. Plan this operation carefully during off peak hours if possible.

Rotate an IAM Identity Center certificate

Rotating an IAM Identity Center certificate is a multistep process that involves the following:

- Generating a new certificate
- Adding the new certificate to the service provider's website
- Setting the new certificate to active
- Deleting the inactive certificate

Use all of the following procedures in the following order to complete the certificate rotation process for a given application.

Step 1: Generate a new certificate.

New IAM Identity Center certificates that you generate can be configured to use the following properties:

- **Validity period** – Specifies the time allotted (in months) before a new IAM Identity Center certificate expires.

- **Key size** – Determines the number of bits that a key must use with its cryptographic algorithm. You can set this value to either 1024-bit RSA or 2048-bit RSA. For general information about how key sizes work in cryptography, see Key size.

- **Algorithm** – Specifies the algorithm that IAM Identity Center uses when signing the SAML assertion/ response. You can set this value to either SHA-1 or SHA-256. AWS recommends using SHA-256 when possible, unless your service provider requires SHA-1. For general information about how cryptography algorithms work, see Public-key cryptography.

1. Open the IAM Identity Center console.

2. Choose **Applications**.

3. In the list of applications, choose the application that you want to generate a new certificate for.

4. On the application details page, choose the **Configuration** tab. Under **IAM Identity Center metadata**, choose **Manage certificate**. If you do not have a **Configuration** tab or the configuration setting is not available, you do not need to rotate the certificate for this application.

5. On the **IAM Identity Center certificate** page, choose **Generate new certificate**.

6. In the **Generate new IAM Identity Center certificate** dialog box, specify the appropriate values for **Validity period**, **Algorithm**, and **Key size**. Then choose **Generate**.

Step 2: Update the service provider's website.

Use the following procedure to reestablish the trust with the application's service provider.

> **Important**
> When you upload the new certificate to the service provider, your users might not be able to get authenticated. To correct this situation, set the new certificate as active as described in the next step.

1. In the IAM Identity Center console, choose the application that you just generated a new certificate for.

2. On the application details page, choose **Edit configuration**.

3. Choose **View instructions**, and then follow the instructions for your specific application service provider's website to add the newly generated certificate.

Step 3: Set the new certificate to active.

An application can have up to two certificates assigned to it. Whichever certificate is set as active, IAM Identity Center will use it to sign all SAML assertions.

1. Open the IAM Identity Center console.

2. Choose **Applications**.

3. In the list of applications, choose your application.

4. On the application details page, choose the **Configuration** tab. Under **IAM Identity Center metadata**, choose **Manage certificate**.

5. On the **IAM Identity Center certificate** page, select the certificate you want to set to active, choose **Actions**, and then choose **Set as active**.

6. In the **Set the selected certificate as active** dialog, confirm that you understand that setting a certificate to active may require you to re-establish the trust, and then choose **Make active**.

Step 4: Delete the old certificate.

Use the following procedure to complete the certificate rotation process for your application. You can only delete a certificate that is in an **Inactive** state.

1. Open the IAM Identity Center console.

2. Choose **Applications**.

3. In the list of applications, choose your application.

4. On the application details page, select the **Configuration** tab. Under **IAM Identity Center metadata**, choose **Manage certificate**.

5. On the **IAM Identity Center certificate** page, select the certificate you want to delete. Choose **Actions** and then choose **Delete**.

6. In the **Delete certificate** dialog box, choose **Delete**.

Certificate expiration status indicators

While on the **Applications** page in the properties of an application, you may notice colored status indicator icons. These icons appear in the **Expires on** column next to each certificate in the list. The following describes the criteria that IAM Identity Center uses to determine which icon is displayed for each certificate.

- **Red** – Indicates that a certificate is currently expired.

- **Yellow** – Indicates that a certificate will expire in 90 days or less.

- **Green** – Indicates that a certificate is currently valid and will remain valid for at least 90 more days.

To check the current status of a certificate

1. Open the IAM Identity Center console.

2. Choose **Applications**.

3. In the list of applications, review the status of the certificates in the list as indicated in the **Expires on** column.

Application properties

In IAM Identity Center you can customize the user experience by configuring the following additional application properties.

Application start URL

You use an application start URL to start the federation process with your application. The typical use is for an application that supports only service provider (SP)-initiated binding.

The following steps and diagram illustrate the application start URL authentication workflow when a user chooses an application in the AWS access portal:

1. The user's browser redirects the authentication request using the value for the application start URL (in this case https://example.com).

2. The application sends an HTML POST with a SAMLRequest to IAM Identity Center.

3. IAM Identity Center then sends an HTML POST with a SAMLResponse back to the application.
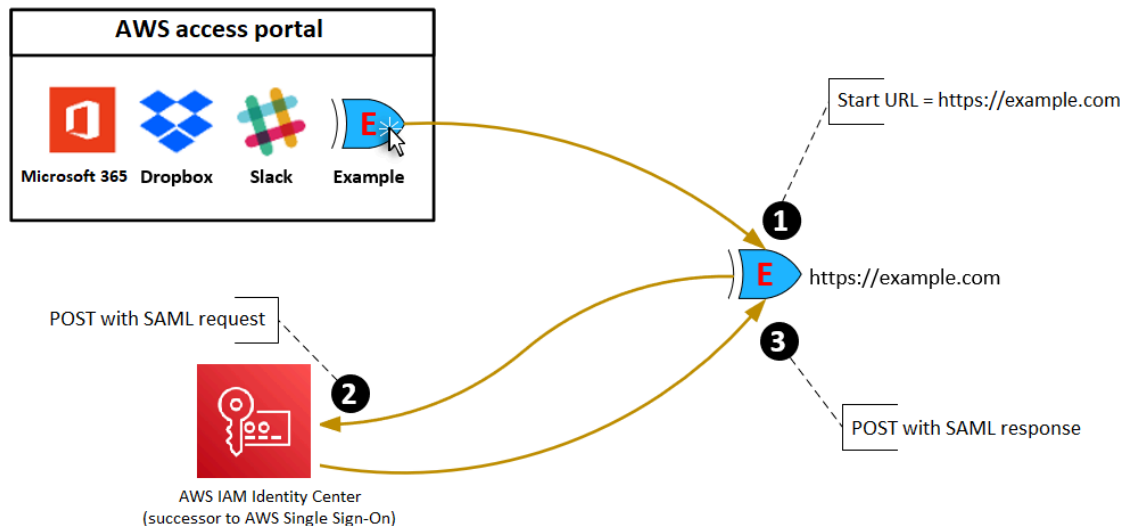
Relay state

During the federation authentication process, the relay state redirects users within the application. For SAML 2.0, this value is passed, unmodified, to the application. After the application properties are configured, IAM Identity Center sends the relay state value along with a SAML response to the application.
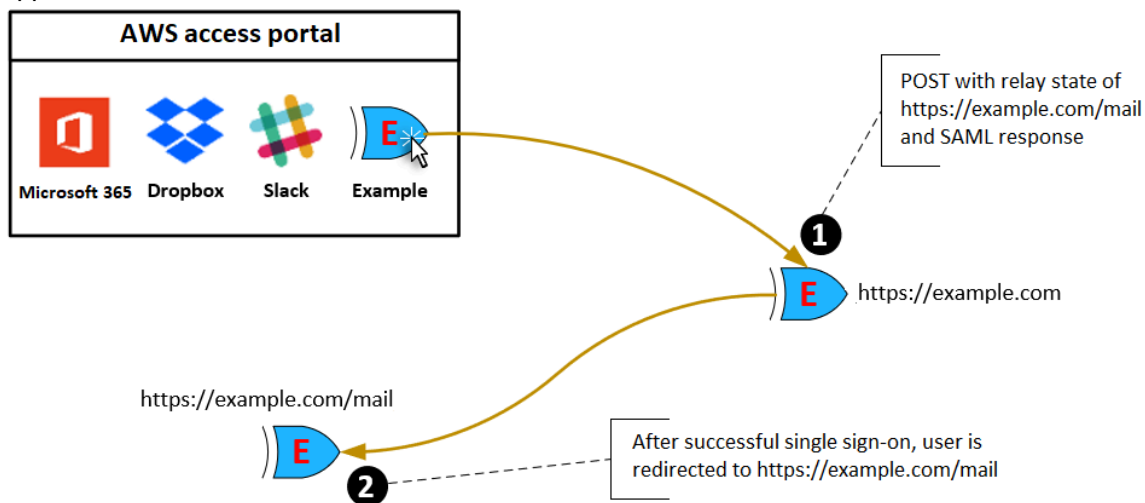
Session duration

Session duration is the length of time that the application user sessions are valid for. For SAML 2.0, this is used to set the `NotOnOrAfter` date of the SAML assertion's elements; `saml2:SubjectConfirmationData` and `saml2:Conditions`.

Session duration can be interpreted by applications in any of the following ways:

- Applications can use it to determine how long the SAML assertion is valid. Applications do not consider session duration when deciding the time allowed for the user.
- Applications can use it to determine the maximum time that is allowed for the user's session. Applications might generate a user session with a shorter duration. This can happen when the

application only supports user sessions with a duration that is shorter than the configured session length.

- Applications can use it as the exact duration and might not allow administrators to configure the value. This can happen when the application only supports a specific session length.

For more information about how session duration is used, see your specific application's documentation.

Assign user access to applications

Use the following procedure to assign users single sign-on access to cloud applications or custom SAML 2.0 applications.

Note

- To help simplify administration of access permissions, we recommend that you assign access directly to groups rather than to individual users. With groups you can grant or deny permissions to groups of users, rather than having to apply those permissions to each individual. If a user moves to a different organization, you simply move that user to a different group. The user then automatically receives the permissions that are needed for the new organization.
- When assigning user access to applications, IAM Identity Center does not currently support users being added to nested groups. If a user is added to a nested group, they may receive a "You do not have any applications" message during sign-in. Assignments must be made against the immediate group the user is a member of.

To assign user or group access to applications

1. Open the IAM Identity Center console.

   Note
   Make sure that the IAM Identity Center console is using the Region where your AWS Managed Microsoft AD directory is located before taking the next step.

2. Choose **Applications**.
3. In the list of applications, choose an application to which you want to assign access.
4. On the application details page, choose the **Assigned users** tab. Then choose **Assign users**.
5. In the **Assign users** dialog box, enter a user or group name. Then choose **Search connected directory**. You can specify multiple users or groups by selecting the applicable accounts as they appear in search results.
6. Choose **Assign users**.

Remove user access

Use this procedure to remove user access to cloud applications or custom SAML 2.0 applications.

To remove user access from an application

1. Open the IAM Identity Center console.
2. Choose **Applications**.
3. In the list of applications, choose an application whose access you want to remove.
4. On the application details page, choose the **Assigned users** tab. Select the user or group that you want to remove and then choose **Remove**.

AWS IAM Identity Center (successor
to AWS Single Sign-On) User Guide
Map attributes in your application
to IAM Identity Center attributes

5. In the **Remove access** dialog box, verify the user or group name. Then choose **Remove access**.

Map attributes in your application to IAM Identity Center attributes

Some service providers require custom SAML assertions to pass additional data about your user sign-ins. In that case, use the following procedure to specify how your applications user attributes should map to corresponding attributes in IAM Identity Center.

To map application attributes to attributes in IAM Identity Center

1. Open the IAM Identity Center console.
2. Choose **Applications**.
3. In the list of applications, choose the application where you want to map attributes.
4. On the application details page, choose the **Attribute mappings** tab.
5. Choose **Add new attribute mapping**
6. In the first text box, enter the application attribute.
7. In the second text box, enter the attribute in IAM Identity Center that you want to map to the application attribute. For example, you might want to map the application attribute `Username` to the IAM Identity Center user attribute `email`. To see the list of allowed user attributes in IAM Identity Center, see the table in Attribute mappings (p. 34).
8. In the third column of the table, choose the appropriate format for the attribute from the menu.
9. Choose **Save changes**.

Resiliency design and Regional behavior

The IAM Identity Center service is fully managed and uses highly available and durable AWS services, such as Amazon S3 and Amazon EC2. To ensure availability in the event of an availability zone disruption, IAM Identity Center operates across multiple availability zones.

You enable IAM Identity Center in your AWS Organizations management account. This is required so that IAM Identity Center can provision, de-provision, and update roles across all your AWS accounts. When you enable IAM Identity Center, it is deployed to the AWS Region that is currently selected. If you want to deploy to a specific AWS Region, change the region selection before enabling IAM Identity Center.

> **Note**
> IAM Identity Center controls access to its permission sets and applications from its primary Region only. We recommend that you consider the risks associated with access control when IAM Identity Center operates in a single Region.

Although IAM Identity Center determines access from the Region in which you enable the service, AWS accounts are global. This means that after users sign in to IAM Identity Center, they can operate in any Region when they access AWS accounts through IAM Identity Center. Most Identity Center enabled applications such as Amazon SageMaker, however, must be installed in the same Region as IAM Identity Center for users to authenticate and assign access to these applications. For information about Regional constraints when using an application with IAM Identity Center, see the documentation for the application.

You can also use IAM Identity Center to authenticate and authorize access to SAML-based applications that are reachable through a public URL, regardless of the platform or cloud on which the application is built.

Set up emergency access to the AWS Management Console

IAM Identity Center is built from highly available AWS infrastructure and uses an Availability Zone architecture to eliminate single points of failure. For an extra layer of protection in the unlikely event of an IAM Identity Center or AWS Region disruption, we recommend that you set up a configuration that you can use to provide temporary access to the AWS Management Console.

Contents

Overview

AWS enables you to:

- Connect your third-party IdP to IAM Identity Center.
- Connect your third-party IdP to individual AWS accounts by using SAML 2.0-based federation.

If you use IAM Identity Center, you can use these capabilities to create the emergency access configuration described in the following sections. This configuration enables you to use IAM Identity Center as the mechanism for AWS account access. If IAM Identity Center is disrupted, your emergency operations users can sign in to the AWS Management Console through direct federation, by using the same credentials that they use to access their accounts. This configuration works when IAM Identity Center is unavailable, but the IAM data plane and your external identity provider (IdP) are available.

> **Important**
> We recommend that you deploy this configuration before a disruption occurs because you can't create the configuration if your access to create the required IAM roles is also disrupted. Also, test this configuration periodically to ensure that your team understands what to do if IAM Identity Center is disrupted.

Summary of emergency access configuration

To configure emergency access, you must complete the following tasks:

1. Create an emergency operations account in your organization in AWS Organizations.
2. Connect your IdP to the emergency operations account by using SAML 2.0-based federation.
3. In the emergency operations account, create a role for third-party identity provider federation. Also, create an emergency operations role in each of your workload accounts, with your required permissions.
4. Delegate access to your workload accounts for the IAM role that you created in the emergency operations account. To authorize access to your emergency operations account, create an emergency operations group in your IdP, with no members.
5. Enable the emergency operations group in your IdP to use the emergency operations role by creating a rule in your IdP that enables SAML 2.0 federated access to the AWS Management Console.

During normal operations, no one has access to the emergency operations account because the emergency operations group in your IdP has no members. In the event of an IAM Identity Center disruption, use your IdP to add trusted users to the emergency operations group in your IdP. These users can then sign in to your IdP, navigate to the AWS Management Console, and assume the emergency operations role in the emergency operations account. From there, these users can switch roles to the emergency access role in your workload accounts where they need to perform operations work.

How to design your critical operations roles

With this design, you configure a single AWS account in which you federate through IAM, so that users can assume critical operations roles. The critical operations roles have a trust policy that enables users to assume a corresponding role in your workload accounts. The roles in the workload accounts provide the permissions that users require to perform essential work. The following diagram provides a design overview.
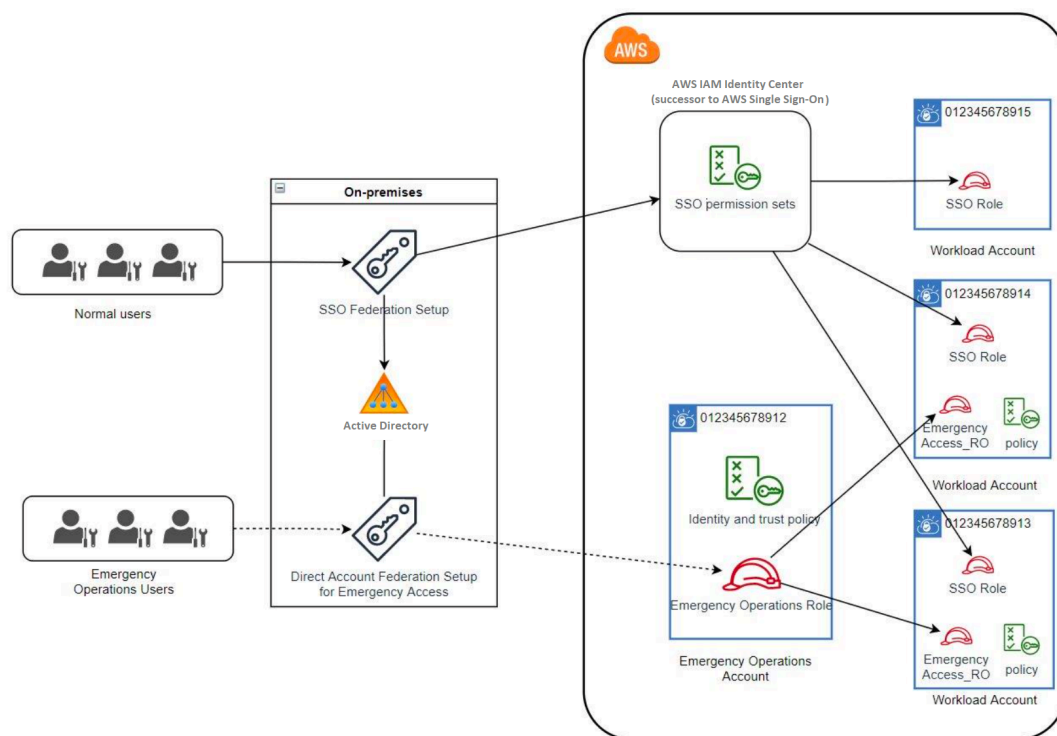
How to plan your access model

Before you configure emergency access, create a plan for how the access model will work. Use the following process to create this plan.

1. Identify the AWS accounts where emergency operator access is essential during a disruption to IAM Identity Center. For example, your production accounts are probably essential, but your development and test accounts might not be.

2. For that collection of accounts, identify the specific critical roles that you need in your accounts. Across these accounts, be consistent in defining what the roles can do. This simplifies work in your emergency access account where you create cross-account roles. We recommend that you start with two distinct roles in these accounts: Read Only (RO) and Operations (Ops). If required, you can create more roles and map these roles to a more distinct group of emergency access users in your setup.

3. Identify and create emergency access groups in your IdP. The group members are the users to whom you are delegating access to emergency access roles.

4. Define which roles these groups can assume in the emergency access account. To do this, define rules in your IdP that generate claims that list which roles the group can access. These groups can then assume your Read Only or Operations roles in emergency access account. From those roles, they can assume corresponding roles in your workload accounts.

How to design emergency role, account, and group mapping

The following diagram shows how to map your emergency access groups to roles in your emergency access account. The diagram also shows the cross-account role trust relationships that enable emergency

access account roles to access corresponding roles in your workload accounts. We recommend that your
emergency plan design use these mappings as a starting point.
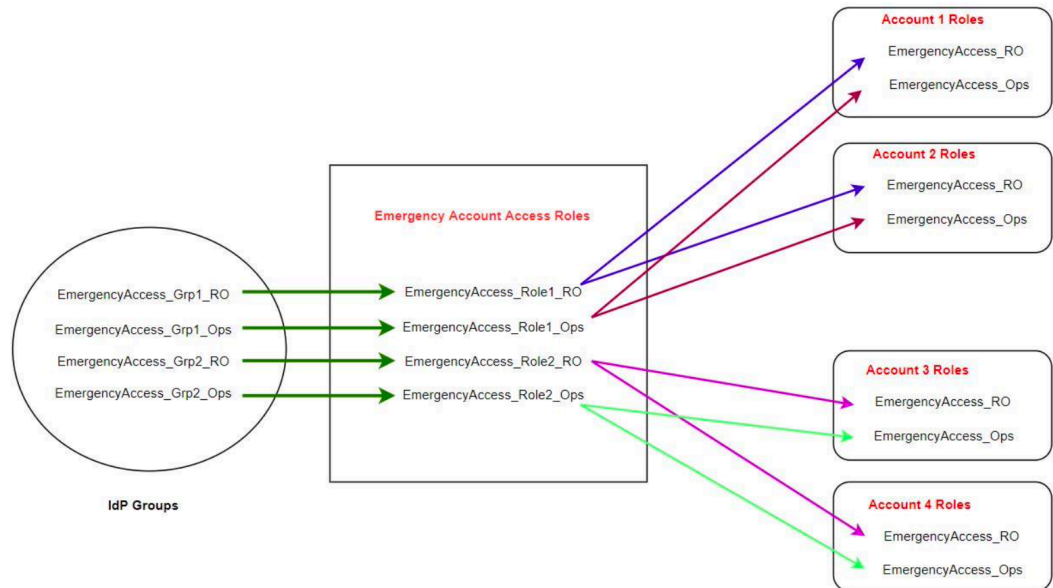
How to create your emergency access configuration

Use the following mapping table to create your emergency access configuration. This table reflects
a plan that includes two roles in the workload accounts: Read Only (RO) and Operations (Ops) , with
corresponding trust policies and permissions policies. The trust policies enable the emergency access
account roles to access the individual workload account roles. The individual workload account roles also
have permissions policies for what the role can do in the account. The permissions policies can be AWS
managed policies or customer managed policies.

Account	Roles to create	Trust policy	Permissions policy
Account 1	EmergencyAccess_RO	EmergencyAccess_Role1_RO	arn:aws:iam::aws:policy/ ReadOnlyAccess
Account 1	EmergencyAccess_Ops	EmergencyAccess_Role1_Ops	arn:aws:iam::aws:policy/ job-function/ SystemAdministrator
Account 2	EmergencyAccess_RO	EmergencyAccess_Role2_RO	arn:aws:iam::aws:policy/ ReadOnlyAccess
Account 2	EmergencyAccess_Ops	EmergencyAccess_Role2_Ops	arn:aws:iam::aws:policy/ job-function/ SystemAdministrator
Emergency access account	EmergencyAccess_Role1_RO EmergencyAccess_Role1_Ops EmergencyAccess_Role2_RO EmergencyAccess_Role2_Ops	N/A	AssumeRole for role resource in account

In this mapping plan, the emergency access account contains two read-only roles and two operations roles. These roles trust your IdP to authenticate and authorize your selected groups to access the roles by passing the names of the roles in assertions. There are corresponding read-only and operations roles in workload Account 1 and Account 2. For workload Account 1, the `EmergencyAccess_RO` role trusts the `EmergencyAccess_Role1_RO` role that resides in the emergency access account. The table specifies similar trust patterns between the workload account read-only and operations roles and the corresponding emergency access roles.

Emergency preparation tasks

To prepare your emergency access configuration, we recommend that you perform the following tasks before an emergency occurs.

1. Set up a direct IAM federation application in your IdP. For more information, see One-time setup of a direct IAM federation application in Okta (p. 128).
2. Create an IdP connection in the emergency access account that can be accessed during the event.
3. Create emergency access roles in the emergency access accounts as described in the mapping table above.
4. Create temporary operations roles with trust and permission policies in each of the workload accounts.
5. Create temporary operations groups in your IdP. The group names will depend on the names of the temporary operations roles.
6. Test direct IAM federation.
7. Disable the IdP federation application in your IdP to prevent regular usage.

Emergency failover process

When an IAM Identity Center instance isn't available and you determine that you must provide emergency access to the AWS Management Console, we recommend the following failover process.

1. The IdP administrator enables the direct IAM federation application in your IdP.
2. Users request access to the temporary operations group through your existing mechanism, such as an email request, Slack channel, or other form of communication.
3. Users that you add to your emergency access groups sign in to the IdP, select the emergency access account, and, users choose a role to use in the emergency access account. From these roles, they can assume roles in corresponding workload accounts that have cross-account trust with the emergency account role.

Return to normal operations

Check the AWS Health Dashboardto confirm when the health of the IAM Identity Center service is restored. To return to normal operations, perform the following steps.

1. After the status icon for the IAM Identity Center service indicates that the service is healthy, sign in to IAM Identity Center.
2. If you can sign in to IAM Identity Center successfully, communicate to emergency access users that IAM Identity Center is available. Instruct these users to sign out and use the AWS access portal to sign back in to IAM Identity Center.
3. After all emergency access users sign out, in the IdP, disable the IdP federation application. We recommend that you perform this task after working hours.
4. Remove all users from the emergency access group in the IdP.

AWS IAM Identity Center (successor
to AWS Single Sign-On) User Guide
One-time setup of a direct IAM
federation application in Okta

Your emergency access role infrastructure remains in place as a backup access plan, but it is now disabled.

One-time setup of a direct IAM federation application in Okta

1. Sign in to your Okta account as a user with administrative permissions.

2. In the Okta Admin Console, under **Applications**, choose **Applications.**

3. Choose **Browse App Catalog**. Search for and choose **AWS Account Federation**. Then choose **Add integration**.

4. Set up direct IAM federation with AWS by following the steps in How to Configure SAML 2.0 for AWS Account Federation.

5. On the **Sign-On Options** tab, select SAML 2.0 and enter **Group Filter** and **Role Value Pattern** settings. The name of the group for the user directory depends on the filter that you configure.

Group Filter	^aws\#\S+\#(?{{role}}[\w\-]+)\#(?{{accountid}}\d+)$
Role Value Pattern	arn:aws:iam::${accountid}:saml-provider/Okta,arn:aws:iam::${accountid}:role/${role}

   In the figure above, the `role` variable is for the emergency operations role in your emergency access account. For example, if you create the `EmergencyAccess_Role1_RO` role (as described in the mapping table) in AWS account 123456789012, and if your group filter setting is configured as shown in the figure above, your group name should be `aws#EmergencyAccess_Role1_RO#123456789012`.

6. In your directory (for example, your directory in Active Directory), create the emergency access group and specify a name for the directory (for example, `aws#EmergencyAccess_Role1_RO#123456789012`). Assign your users to this group by using your existing provisioning mechanism.

7. In the emergency access account, configure a custom trust policy that provides the permissions required for the emergency access role to be assumed during a disruption. Following is an example statement for a custom **trust policy** that is attached to the `EmergencyAccess_Role1_RO` role. For an illustration, see the emergency account in the diagram under How to design emergency role, account, and group mapping (p. 125).

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect":"Allow",
        "Principal":{
           "Federated":"arn:aws:iam::123456789012:saml-provider/Okta"
        },
        "Action":[
           "sts:AssumeRoleWithSAML",
           "sts:SetSourceIdentity",
           "sts:TagSession"
        ],
        "Condition":{
           "StringEquals":{
              "SAML:aud":"https:~/~/signin.aws.amazon.com/saml"
           }
        }
      }
```

AWS IAM Identity Center (successor
to AWS Single Sign-On) User Guide
One-time setup of a direct IAM
federation application in Okta

```
    ]
}
```

8. The following is an example statement for a **permissions policy** that is attached to the
   `EmergencyAccess_Role1_RO` role. For an illustration, see the emergency account in the diagram
   under How to design emergency role, account, and group mapping (p. 125).

```
{
    "Version": "2012-10-17",
    "Statement":[
      {
         "Effect":"Allow",
         "Action":"sts:AssumeRole",
         "Resource":[
            "arn:aws:iam::<account 1>:role/EmergencyAccess_RO",
            "arn:aws:iam::<account 2>:role/EmergencyAccess_RO"
         ]
      }
    ]
}
```

9. On the workload accounts, configure a custom trust policy. Following is an example statement
   for a **trust policy** that is attached to the `EmergencyAccess_RO` role. In this example, account
   123456789012 is the emergency access account. For an illustration, see workload account in the
   diagram under How to design emergency role, account, and group mapping (p. 125).

```
{
    "Version": "2012-10-17",
    "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "AWS":"arn:aws:iam::123456789012:root"
         },
         "Action":"sts:AssumeRole"
      }
    ]
}
```

Note
Most IdPs enable you to keep an application integration deactivated until required. We
recommend that you keep the direct IAM federation application deactivated in your IdP
until required for emergency access.

Security in AWS IAM Identity Center (successor to AWS Single Sign-On)

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from a data center and network architecture that is built to meet the requirements of the most security-sensitive organizations.

Security is a shared responsibility between AWS and you. 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 IAM Identity Center (successor to AWS Single Sign-On), see AWS Services in Scope by Compliance Program.

- **Security in the cloud** – Your responsibility is determined by the AWS service that you use. You are also responsible for other factors including the sensitivity of your data, your company's requirements, and applicable laws and regulations.

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

Topics

- Identity and access management for IAM Identity Center (p. 130)
- IAM Identity Center console and API authorization (p. 156)
- Logging and monitoring in IAM Identity Center (p. 157)
- Compliance validation for IAM Identity Center (p. 173)
- Resilience in IAM Identity Center (p. 175)
- Infrastructure security in IAM Identity Center (p. 176)

Identity and access management for IAM Identity Center

Access to IAM Identity Center requires credentials that AWS can use to authenticate your requests. Those credentials must have permissions to access AWS resources, such as an IAM Identity Center application.

Authentication to the AWS access portal is controlled by the directory that you have connected to IAM Identity Center. However, authorization to the AWS accounts that are available to end users from within the AWS access portal is determined by two factors:

1. Who has been assigned access to those AWS accounts in the IAM Identity Center console. For more information, see Single sign-on access to AWS accounts (p. 95).

2. What level of permissions have been granted to the end users in the IAM Identity Center console
   to allow them the appropriate access to those AWS accounts. For more information, see Create and
   manage permission sets (p. 98).

The following sections explain how you as an administrator can control access to the IAM Identity Center
console or can delegate administrative access for day-to-day tasks from the IAM Identity Center console.

- Authentication (p. 131)
- Access control (p. 132)

Authentication

You can access AWS as any of the following types of identities:

- **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 do not 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 *AWS General Reference*.

- **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 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)
  in the *IAM User Guide*.

- **IAM role**

  An IAM role is an IAM identity that you can create in your account that has specific permissions. An IAM
  role is similar to an IAM user in that it is an AWS identity with permissions policies that determine what
  the identity can and cannot do in AWS. However, instead of being uniquely associated with one person,
  a role is intended to be assumable by anyone who needs it. Also, a role does not have standard long-
  term credentials such as a password or access keys associated with it. Instead, when you assume a role,
  it provides you with temporary security credentials for your role session. 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 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 permissions sets, see Permission sets in the *AWS IAM Identity Center (successor to AWS Single Sign-On) User Guide*.

- **AWS service access** – 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*.

- **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*.

Access control

You can have valid credentials to authenticate your requests, but unless you have permissions you cannot create or access IAM Identity Center resources. For example, you must have permissions to create an IAM Identity Center connected directory.

The following sections describe how to manage permissions for IAM Identity Center. We recommend that you read the overview first.

- Overview of managing access permissions to your IAM Identity Center resources (p. 132)
- Identity-based policy examples for IAM Identity Center (p. 135)
- Using service-linked roles for IAM Identity Center (p. 151)

Overview of managing access permissions to your IAM Identity Center resources

Every AWS resource is owned by an AWS account, and permissions to create or access the resources are governed by permissions policies. An account administrator can attach permissions policies to IAM identities (that is, users, groups, and roles). Some services (such as AWS Lambda) also support attaching permissions policies to resources.

> **Note**
> An *account administrator* (or administrator user) is a user with administrator privileges. For more information, see IAM best practices in the *IAM User Guide*.

When granting permissions, you decide who is getting the permissions, the resources they get permissions for, and the specific actions that you want to allow on those resources.

Topics

- IAM Identity Center resources and operations (p. 133)
- Understanding resource ownership (p. 133)
- Managing access to resources (p. 133)
- Specifying policy elements: actions, effects, resources, and principals (p. 134)
- Specifying conditions in a policy (p. 135)

IAM Identity Center resources and operations

In IAM Identity Center, the primary resources are application instances, profiles, and permission sets.

Understanding resource ownership

A *resource owner* is the AWS account that created a resource. That is, the resource owner is the AWS account of the *principal entity* (the account, an IAM user, or an IAM role) that authenticates the request that creates the resource. The following examples illustrate how this works:

- If the AWS account root user creates an IAM Identity Center resource, such as an application instance or permission set, your AWS account is the owner of that resource.
- If you create an IAM user in your AWS account and grant that user permissions to create IAM Identity Center resources, the user can then create IAM Identity Center resources. However, your AWS account, to which the user belongs, owns the resources.
- If you create an IAM role in your AWS account with permissions to create IAM Identity Center resources, anyone who can assume the role can create IAM Identity Center resources. Your AWS account, to which the role belongs, owns the IAM Identity Center resources.

Managing access to resources

A *permissions policy* describes who has access to what. The following section explains the available options for creating permissions policies.

> **Note**
> This section discusses using IAM in the context of IAM Identity Center. It doesn't provide detailed information about the IAM service. For complete IAM documentation, see What is IAM? in the *IAM User Guide*. For information about IAM policy syntax and descriptions, see AWS IAM policy reference in the *IAM User Guide*.

Policies that are attached to an IAM identity are referred to as *identity-based* policies (IAM policies). Policies that are attached to a resource are referred to as *resource-based* policies. IAM Identity Center supports only identity-based policies (IAM policies).

Topics
- Identity-based policies (IAM policies) (p. 133)
- Resource-based policies (p. 134)

Identity-based policies (IAM policies)

You can attach policies to IAM identities. For example, you can do the following:

- **Attach a permissions policy to a user or a group in your account** – An account administrator can use a permissions policy that is associated with a particular user to grant permissions for that user to add an IAM Identity Center resource, such as a new application.
- **Attach a permissions policy to a role (grant cross-account permissions)** – You can attach an identity-based permissions policy to an IAM role to grant cross-account permissions. For example, the administrator in Account A can create a role to grant cross-account permissions to another AWS account (for example, Account B) or an AWS service as follows:
  1. Account A administrator creates an IAM role and attaches a permissions policy to the role that grants permissions to resources in Account A.
  2. Account A administrator attaches a trust policy to the role identifying Account B as the principal who can assume the role.

3. Account B administrator can then delegate permissions to assume the role to any users in Account B. Doing this allows users in Account B to create or access resources in Account A. The principal in the trust policy can also be an AWS service principal if you want to grant an AWS service permissions to assume the role.

For more information about using IAM to delegate permissions, see Access management in the *IAM User Guide*.

The following permissions policy grants permissions to a user to run all of the actions that begin with `List`. These actions show information about an IAM Identity Center resource, such as an application instance or permissions set. Note that the wildcard character (*) in the `Resource` element indicates that the actions are allowed for all IAM Identity Center resources that are owned by the account.

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

For more information about using identity-based policies with IAM Identity Center, see Identity-based policy examples for IAM Identity Center (p. 135). For more information about users, groups, roles, and permissions, see Identities (users, groups, and roles) in the *IAM User Guide*.

Resource-based policies

Other services, such as Amazon S3, also support resource-based permissions policies. For example, you can attach a policy to an S3 bucket to manage access permissions to that bucket. IAM Identity Center doesn't support resource-based policies.

Specifying policy elements: actions, effects, resources, and principals

For each IAM Identity Center resource (see IAM Identity Center resources and operations (p. 133)), the service defines a set of API operations. To grant permissions for these API operations, IAM Identity Center defines a set of actions that you can specify in a policy. Note that performing an API operation can require permissions for more than one action.

The following are the basic policy elements:

- **Resource** – In a policy, you use an Amazon Resource Name (ARN) to identify the resource to which the policy applies. For IAM Identity Center resources, you always use the wildcard character (*) in IAM policies. For more information, see IAM Identity Center resources and operations (p. 133).
- **Action** – You use action keywords to identify resource operations that you want to allow or deny. For example, the `sso:DescribePermissionsPolicies` permission allows the user permissions to perform the IAM Identity Center `DescribePermissionsPolicies` operation.
- **Effect** – You specify the effect when the user requests the specific action—this can be either allow or deny. If you don't explicitly grant access to (allow) a resource, access is implicitly denied. You can also explicitly deny access to a resource, which you might do to make sure that a user cannot access it, even if a different policy grants access.
- **Principal** – In identity-based policies (IAM policies), the user that the policy is attached to is the implicit principal. For resource-based policies, you specify the user, account, service, or other entity

that you want to receive permissions (applies to resource-based policies only). IAM Identity Center
doesn't support resource-based policies.

To learn more about IAM policy syntax and descriptions, see AWS IAM policy reference in the *IAM User
Guide*.

Specifying conditions in a policy

When you grant permissions, you can use the access policy language to specify the conditions that are
required for a policy to take effect. For example, you might want a policy to be applied only after a
specific date. For more information about specifying conditions in a policy language, see Condition in the
IAM User Guide.

To express conditions, you use predefined condition keys. There are no condition keys specific to IAM
Identity Center. However, there are AWS condition keys that you can use as appropriate. For a complete
list of AWS keys, see Available global condition keys in the *IAM User Guide*.

Identity-based policy examples for IAM Identity Center

This topic provides examples of permissions policies that you can attach to AWS identities, including
IAM users, groups, and roles, and IAM Identity Center users (as part of a custom permissions policy), for
administration of IAM Identity Center.

> **Important**
> We recommend that you first review the introductory topics that explain the basic concepts
> and options available for you to manage access to your IAM Identity Center resources. For
> more information, see Overview of managing access permissions to your IAM Identity Center
> resources (p. 132).

The sections in this topic cover the following:

- Custom policy examples (p. 135)
- Permissions required to use the IAM Identity Center console (p. 140)

Custom policy examples

This section provides examples of common use cases that require a custom IAM policy. These example
policies are identity-based policies, which do not specify the Principal element. This is because with an
identity-based policy, you don't specify the principal who gets the permission. Instead, you attach the
policy to the principal. When you attach an identity-based permission policy to an IAM role, the principal
identified in the role's trust policy gets the permissions. You can create identity-based policies in IAM and
attach them to users, groups, and/or roles. You can also apply these policies to IAM Identity Center users
when you create a permission set in IAM Identity Center.

> **Note**
> Use these examples when you create policies for your environment and make sure to test for
> both positive ("access granted") and negative ("access denied") test cases before you deploy
> these policies in your production environment. For more information about testing IAM policies,
> see Testing IAM policies with the IAM policy simulator in the *IAM User Guide*.

Topics

- Example 1: Allow a user to view IAM Identity Center (p. 136)
- Example 2: Allow a user to manage permissions to AWS accounts in IAM Identity Center (p. 136)

Example 1: Allow a user to view IAM Identity Center

The following permissions policy grants read-only permissions to a user so they can view all the settings and directory information configured in IAM Identity Center.

> **Note**
> This policy is provided for example purposes only. In a production environment, we recommend that you use the `ViewOnlyAccess` AWS managed policy for IAM Identity Center.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "ds:DescribeDirectories",
                "ds:DescribeTrusts",
                "iam:ListPolicies",
                "organizations:DescribeOrganization",
                "organizations:DescribeAccount",
                "organizations:ListParents",
                "organizations:ListChildren",
                "organizations:ListAccounts",
                "organizations:ListRoots",
                "organizations:ListAccountsForParent",
                "organizations:ListOrganizationalUnitsForParent",
                "sso:ListManagedPoliciesInPermissionSet",
                "sso:ListPermissionSetsProvisionedToAccount",
                "sso:ListAccountAssignments",
                "sso:ListAccountsForProvisionedPermissionSet",
                "sso:ListPermissionSetsProvisionedToAccount",
                "sso:ListPermissionSets",
                "sso:DescribePermissionSet",
                "sso:GetInlinePolicyForPermissionSet",
                "sso-directory:DescribeDirectory",
                "sso-directory:SearchUsers",
                "sso-directory:SearchGroups"
            ],
            "Resource": "*"
        }
    ]
}
```

Example 2: Allow a user to manage permissions to AWS accounts in IAM Identity Center

The following permissions policy grants permissions to allow a user to create, manage, and deploy permission sets for your AWS accounts.

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

```
            "sso:AttachManagedPolicyToPermissionSet",
            "sso:CreateAccountAssignment",
            "sso:CreatePermissionSet",
            "sso:DeleteAccountAssignment",
            "sso:DeleteInlinePolicyFromPermissionSet",
            "sso:DeletePermissionSet",
            "sso:DetachManagedPolicyFromPermissionSet",
            "sso:ProvisionPermissionSet",
            "sso:PutInlinePolicyToPermissionSet",
            "sso:UpdatePermissionSet"
        ],
        "Resource": "*"
    },
    {
        "Sid": "IAMListPermissions",
        "Effect": "Allow",
        "Action": [
            "iam:ListRoles",
            "iam:ListPolicies"
        ],
        "Resource": "*"
    },
    {
        "Sid": "AccessToSSOProvisionedRoles",
        "Effect": "Allow",
        "Action": [
            "iam:AttachRolePolicy",
            "iam:CreateRole",
            "iam:DeleteRole",
            "iam:DeleteRolePolicy",
            "iam:DetachRolePolicy",
            "iam:GetRole",
            "iam:ListAttachedRolePolicies",
            "iam:ListRolePolicies",
            "iam:PutRolePolicy",
            "iam:UpdateRole",
            "iam:UpdateRoleDescription"
        ],
        "Resource": "arn:aws:iam::*:role/aws-reserved/sso.amazonaws.com/*"
    },
    {
        "Effect": "Allow",
        "Action": [
            "iam:GetSAMLProvider"
        ],
        "Resource": "arn:aws:iam::*:saml-provider/AWSSSO_*_DO_NOT_DELETE"
    }
  ]
}
```

Note

The additional permissions listed under the "Sid": "IAMListPermissions", and "Sid":
"AccessToSSOProvisiondRoles" sections are required only to enable the user to create
assignments in the AWS Organizations management account.

Example 3: Allow a user to manage applications in IAM Identity Center

The following permissions policy grants permissions to allow a user to view and configure applications
in IAM Identity Center, including pre-integrated SaaS applications from within the IAM Identity Center
catalog.

Note

The sso:AssociateProfile operation used in the following policy example is required for
management of user and group assignments to applications. It also allows a user to assign

users and groups to AWS accounts by using existing permission sets. If a user must manage
AWS account access within IAM Identity Center, and requires permissions necessary to manage
permission sets, see Example 2: Allow a user to manage permissions to AWS accounts in IAM
Identity Center (p. 136).

As of October 2020, many of these operations are available only through the AWS console. This example
policy includes "read" actions such as list, get, and search, which are relevant to the error-free operation
of the console for this case.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sso:AssociateProfile",
                "sso:CreateApplicationInstance",
                "sso:ImportApplicationInstanceServiceProviderMetadata",
                "sso:DeleteApplicationInstance",
                "sso:DeleteProfile",
                "sso:DisassociateProfile",
                "sso:GetApplicationTemplate",
                "sso:UpdateApplicationInstanceServiceProviderConfiguration",
                "sso:UpdateApplicationInstanceDisplayData",
                "sso:DeleteManagedApplicationInstance",
                "sso:UpdateApplicationInstanceStatus",
                "sso:GetManagedApplicationInstance",
                "sso:UpdateManagedApplicationInstanceStatus",
                "sso:CreateManagedApplicationInstance",
                "sso:UpdateApplicationInstanceSecurityConfiguration",
                "sso:UpdateApplicationInstanceResponseConfiguration",
                "sso:GetApplicationInstance",
                "sso:CreateApplicationInstanceCertificate",
                "sso:UpdateApplicationInstanceResponseSchemaConfiguration",
                "sso:UpdateApplicationInstanceActiveCertificate",
                "sso:DeleteApplicationInstanceCertificate",
                "sso:ListApplicationInstanceCertificates",
                "sso:ListApplicationTemplates",
                "sso:ListApplications",
                "sso:ListApplicationInstances",
                "sso:ListDirectoryAssociations",
                "sso:ListProfiles",
                "sso:ListProfileAssociations",
                "sso:ListInstances",
                "sso:GetProfile",
                "sso:GetSSOStatus",
                "sso:GetSsoConfiguration",
                "sso-directory:DescribeDirectory",
                "sso-directory:DescribeUsers",
                "sso-directory:ListMembersInGroup",
                "sso-directory:SearchGroups",
                "sso-directory:SearchUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Example 4: Allow a user to manage users and groups in your Identity Center directory

The following permissions policy grants permissions to allow a user to create, view, modify, and delete
users and groups in IAM Identity Center.

In some cases, direct modifications to users and groups in IAM Identity Center are restricted. For example, when Active Directory, or an external identity provider with Automatic Provisioning enabled, is selected as the identity source.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sso-directory:ListGroupsForUser",
                "sso-directory:DisableUser",
                "sso-directory:EnableUser",
                "sso-directory:SearchGroups",
                "sso-directory:DeleteGroup",
                "sso-directory:AddMemberToGroup",
                "sso-directory:DescribeDirectory",
                "sso-directory:UpdateUser",
                "sso-directory:ListMembersInGroup",
                "sso-directory:CreateUser",
                "sso-directory:DescribeGroups",
                "sso-directory:SearchUsers",
                "sso:ListDirectoryAssociations",
                "sso-directory:RemoveMemberFromGroup",
                "sso-directory:DeleteUser",
                "sso-directory:DescribeUsers",
                "sso-directory:UpdateGroup",
                "sso-directory:CreateGroup"
            ],
            "Resource": "*"
        }
    ]
}
```

Example 5: Allow a user to administer IAM Identity Center for specific accounts

As your team grows, consider implementing a delegation model that enables AWS account administrators to manage their users' single sign-on access to their resources. By using this model and the following example policy, you can delegate permissions for administering AWS accounts. After you create this policy, you can select the same policy any time you need to delegate the same permissions to other administrative users.

Account-based delegation

The following policy grants permissions that allow you to delegate access for each of the AWS accounts under `Resource`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid":"DelegateAccountsAdminAccess",
            "Effect":"Allow",
            "Action":[
                "sso:ProvisionPermissionSet",
                "sso:CreateAccountAssignment",
                "sso:DeleteInlinePolicyFromPermissionSet",
                "sso:UpdateInstanceAccessControlAttributeConfiguration",
                "sso:PutInlinePolicyToPermissionSet",
                "sso:DeleteAccountAssignment",
                "sso:DetachManagedPolicyFromPermissionSet",
                "sso:DeletePermissionSet",
                "sso:AttachManagedPolicyToPermissionSet",
```

```
                "sso:CreatePermissionSet",
                "sso:UpdatePermissionSet",
                "sso:CreateInstanceAccessControlAttributeConfiguration",
                "sso:DeleteInstanceAccessControlAttributeConfiguration"
            ],
            "Resource":[
                "arn:aws:sso:::account/112233445566",
                "arn:aws:sso:::account/223344556677",
                "arn:aws:sso:::account/334455667788",
                "arn:aws:sso:::permissionSet/*/*",
                "arn:aws:sso:::instance/ssoins-xxxxxxxxx"
            ]
        }
    ]
}
```

More information

For an example walkthrough that covers how to delegate administration of user identities in an IAM environment, see How to delegate management of identity in AWS IAM Identity Center (successor to AWS Single Sign-On) on the AWS Security Blog.

Permissions required to use the IAM Identity Center console

For a user to work with the IAM Identity Center console without errors, additional permissions are required. If an IAM policy has been created that is more restrictive than the minimum required permissions, the console won't function as intended for users with that policy. The following example lists the set of permissions that might be needed to ensure error-free operation within the IAM Identity Center console.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sso:DescribeAccountAssignmentCreationStatus",
                "sso:DescribeAccountAssignmentDeletionStatus",
                "sso:DescribePermissionSet",
                "sso:DescribePermissionSetProvisioningStatus",
                "sso:DescribePermissionsPolicies",
                "sso:DescribeRegisteredRegions",
                "sso:GetApplicationInstance",
                "sso:GetApplicationTemplate",
                "sso:GetInlinePolicyForPermissionSet",
                "sso:GetManagedApplicationInstance",
                "sso:GetMfaDeviceManagementForDirectory",
                "sso:GetPermissionSet",
                "sso:GetPermissionsPolicy",
                "sso:GetProfile",
                "sso:GetSharedSsoConfiguration",
                "sso:GetSsoConfiguration",
                "sso:GetSSOStatus",
                "sso:GetTrust",
                "sso:ListAccountAssignmentCreationStatus",
                "sso:ListAccountAssignmentDeletionStatus",
                "sso:ListAccountAssignments",
                "sso:ListAccountsForProvisionedPermissionSet",
                "sso:ListApplicationInstanceCertificates",
                "sso:ListApplicationInstances",
                "sso:ListApplications",
                "sso:ListApplicationTemplates",
                "sso:ListDirectoryAssociations",
```

```
                "sso:ListInstances",
                "sso:ListManagedPoliciesInPermissionSet",
                "sso:ListPermissionSetProvisioningStatus",
                "sso:ListPermissionSets",
                "sso:ListPermissionSetsProvisionedToAccount",
                "sso:ListProfileAssociations",
                "sso:ListProfiles",
                "sso:ListTagsForResource",
                "sso-directory:DescribeDirectory",
                "sso-directory:DescribeGroups",
                "sso-directory:DescribeUsers",
                "sso-directory:ListGroupsForUser",
                "sso-directory:ListMembersInGroup",
                "sso-directory:SearchGroups",
                "sso-directory:SearchUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS managed policies for IAM Identity Center

To create IAM customer managed policies that provide your team with only the permissions they need takes time and expertise. To get started quickly, you can use AWS managed policies. These policies cover common use cases and are available in your AWS account. For more information about AWS managed policies, see AWS managed policies in the *IAM User Guide*.

AWS services maintain and update AWS managed policies. You can't change the permissions in AWS managed policies. Services occasionally add additional permissions to an AWS managed policy to support new features. This type of update affects all identities (users, groups, and roles) where the policy is attached. Services are most likely to update an AWS managed policy when a new feature is launched or when new operations become available. Services do not remove permissions from an AWS managed policy, so policy updates won't break your existing permissions.

Additionally, AWS supports managed policies for job functions that span multiple services. For example, the **ReadOnlyAccess** AWS managed policy provides read-only access to all AWS services and resources. When a service launches a new feature, AWS adds read-only permissions for new operations and resources. For a list and descriptions of job function policies, see AWS managed policies for job functions in the *IAM User Guide*.

New actions that allow you to list and delete user sessions are available under the new namespace `identitystore-auth`. Any additional permissions for actions in this namespace will be updated on this page. When creating your custom IAM policies, avoid using `*` after `identitystore-auth` because this applies to all actions that exist in the namespace today or in the future.

AWS managed policy: AWSSSOMasterAccountAdministrator

The `AWSSSOMasterAccountAdministrator` policy provides required administrative actions to principals. The policy is intended for principals who perform the job role of an AWS IAM Identity Center (successor to AWS Single Sign-On) administrator. Over time the list of actions provided will be updated to match the existing functionality of IAM Identity Center and the actions that are required as an administrator.

You can attach the `AWSSSOMasterAccountAdministrator` policy to your IAM identities. When you attach the `AWSSSOMasterAccountAdministrator` policy to an identity, you grant administrative AWS IAM Identity Center (successor to AWS Single Sign-On) permissions. Principals with this policy can access IAM Identity Center within the AWS Organizations management account and all member accounts. This principal can fully manage all IAM Identity Center operations, including the ability to create an IAM Identity Center instance, users, permission sets, and assignments. The principal can also instantiate those

assignments throughout the AWS organization member accounts and establish connections between AWS Directory Service managed directories and IAM Identity Center. As new administrative features are released, the account administrator will be granted these permissions automatically.

Permissions groupings

This policy is grouped into statements based on the set of permissions provided.

- `AWSSSOMasterAccountAdministrator` – Allows IAM Identity Center to pass the service role named AWSServiceRoleforSSO to IAM Identity Center so that it can later assume the role and perform actions on their behalf. This is necessary when the person or application attempts to enable IAM Identity Center. For more information, see Multi-account permissions (p. 91).
- `AWSSSOMemberAccountAdministrator` – Allows IAM Identity Center to perform account administrator actions in a multi-account AWS environment. For more information, see AWS managed policy: AWSSSOMemberAccountAdministrator (p. 144).
- `AWSSSOManageDelegatedAdministrator` – Allows IAM Identity Center to register and deregister a delegated administrator for your organization.

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Sid": "AWSSSOCreateSLR",
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "arn:aws:iam::*:role/aws-service-role/sso.amazonaws.com/
AWSServiceRoleForSSO",
            "Condition": {
                "StringLike": {
                    "iam:AWSServiceName": "sso.amazonaws.com"
                }
            }
        },
        {
            "Sid":"AWSSSOMasterAccountAdministrator",
            "Effect":"Allow",
            "Action":"iam:PassRole",
            "Resource":"arn:aws:iam::*:role/aws-service-role/sso.amazonaws.com/
AWSServiceRoleForSSO",
            "Condition":{
                "StringLike":{
                    "iam:PassedToService":"sso.amazonaws.com"
                }
            }
        },
        {
            "Sid":"AWSSSOMemberAccountAdministrator",
            "Effect":"Allow",
            "Action":[
                "ds:DescribeTrusts",
                "ds:UnauthorizeApplication",
                "ds:DescribeDirectories",
                "ds:AuthorizeApplication",
                "iam:ListPolicies",
                "organizations:EnableAWSServiceAccess",
                "organizations:ListRoots",
                "organizations:ListAccounts",
                "organizations:ListOrganizationalUnitsForParent",
                "organizations:ListAccountsForParent",
                "organizations:DescribeOrganization",
                "organizations:ListChildren",
                "organizations:DescribeAccount",
```

```
                "organizations:ListParents",
                "organizations:ListDelegatedAdministrators",
                "sso:*",
                "sso-directory:*",
                "identitystore:*",
                "identitystore-auth:*",
                "ds:CreateAlias",
                "access-analyzer:ValidatePolicy"
            ],
            "Resource":"*"
        },
        {
            "Sid": "AWSSSOManageDelegatedAdministrator",
            "Effect": "Allow",
            "Action": [
                "organizations:RegisterDelegatedAdministrator",
                "organizations:DeregisterDelegatedAdministrator"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "organizations:ServicePrincipal": "sso.amazonaws.com"
                }
            }
        }

    ]
}
```

Additional information about this policy

When IAM Identity Center is enabled for the first time, the IAM Identity Center service creates
a service linked role in the AWS Organizations management account (formerly master account)
so that IAM Identity Center can manage the resources in your account. The actions required are
iam:CreateServiceLinkedRole and iam:PassRole, which are shown in the following snippets.

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Sid": "AWSSSOCreateSLR",
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "arn:aws:iam::*:role/aws-service-role/sso.amazonaws.com/
AWSServiceRoleForSSO",
            "Condition": {
                "StringLike": {
                    "iam:AWSServiceName": "sso.amazonaws.com"
                }
            }
        },
        {
            "Sid":"AWSSSOMasterAccountAdministrator",
            "Effect":"Allow",
            "Action":"iam:PassRole",
            "Resource":"arn:aws:iam::*:role/aws-service-role/sso.amazonaws.com/
AWSServiceRoleForSSO",
            "Condition":{
                "StringLike":{
                    "iam:PassedToService":"sso.amazonaws.com"
                }
            }
        },
    ]
```

```
}
```

AWS managed policy: AWSSSOMemberAccountAdministrator

The `AWSSSOMemberAccountAdministrator` policy provides required administrative actions to principals. The policy is intended for principals who perform the job role of an IAM Identity Center administrator. Over time the list of actions provided will be updated to match the existing functionality of IAM Identity Center and the actions that are required as an administrator.

You can attach the `AWSSSOMemberAccountAdministrator` policy to your IAM identities. When you attach the `AWSSSOMemberAccountAdministrator` policy to an identity, you grant administrative AWS IAM Identity Center (successor to AWS Single Sign-On) permissions. Principals with this policy can access IAM Identity Center within the AWS Organizations management account and all member accounts. This principal can fully manage all IAM Identity Center operations, including the ability to create users, permission sets, and assignments. The principal can also instantiate those assignments throughout the AWS organization member accounts and establish connections between AWS Directory Service managed directories and IAM Identity Center. As new administrative features are released, the account administrator is granted these permissions automatically.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AWSSSOMemberAccountAdministrator",
      "Effect": "Allow",
      "Action": [
        "ds:DescribeDirectories",
        "ds:AuthorizeApplication",
        "ds:UnauthorizeApplication",
        "ds:DescribeTrusts",
        "iam:ListPolicies",
        "organizations:EnableAWSServiceAccess",
        "organizations:DescribeOrganization",
        "organizations:DescribeAccount",
        "organizations:ListRoots",
        "organizations:ListAccounts",
        "organizations:ListAccountsForParent",
        "organizations:ListParents",
        "organizations:ListChildren",
        "organizations:ListOrganizationalUnitsForParent",
        "organizations:ListDelegatedAdministrators",
        "sso:*",
        "sso-directory:*",
        "identitystore:*",
        "identitystore-auth:*",
        "ds:CreateAlias",
        "access-analyzer:ValidatePolicy"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AWSSSOManageDelegatedAdministrator",
      "Effect": "Allow",
      "Action": [
        "organizations:RegisterDelegatedAdministrator",
        "organizations:DeregisterDelegatedAdministrator"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "organizations:ServicePrincipal": "sso.amazonaws.com"
        }
```

```
                }
            }
        ]
}
```

Additional information about this policy

IAM Identity Center administrators manage users, groups, and passwords in their Identity Center directory store (sso-directory). The account admin role includes permissions for the following actions:

- `"sso:*"`
- `"sso-directory:*"`

IAM Identity Center administrators need limited permissions to the following AWS Directory Service actions to perform daily tasks.

- `"ds:DescribeTrusts"`
- `"ds:UnauthorizeApplication"`
- `"ds:DescribeDirectories"`
- `"ds:AuthorizeApplication"`
- `"ds:CreateAlias"`

These permissions allow IAM Identity Center administrators to identify existing directories and manage applications so that they can be configured for use with IAM Identity Center. For more information about each of these actions, see AWS Directory Service API permissions: Actions, resources, and conditions reference.

IAM Identity Center uses IAM policies to grant permissions to IAM Identity Center users. IAM Identity Center administrators create permission sets and attach polices to them. The IAM Identity Center administrator must have the permissions to list the existing policies so that they can choose which polices to use with the permission set they are creating or updating. To set secure and functional permissions, the IAM Identity Center administrator must have permissions to run the IAM Access Analyzer policy validation.

- `"iam:ListPolicies"`
- `"access-analyzer:ValidatePolicy"`

IAM Identity Center administrators need limited access to the following AWS Organizations actions to perform daily tasks:

- `"organizations:EnableAWSServiceAccess"`
- `"organizations:ListRoots"`
- `"organizations:ListAccounts"`
- `"organizations:ListOrganizationalUnitsForParent"`
- `"organizations:ListAccountsForParent"`
- `"organizations:DescribeOrganization"`
- `"organizations:ListChildren"`
- `"organizations:DescribeAccount"`
- `"organizations:ListParents"`
- `"organizations:ListDelegatedAdministrators"`
- `"organizations:RegisterDelegatedAdministrator"`
- `"organizations:DeregisterDelegatedAdministrator"`

These permissions allow IAM Identity Center administrators the ability to work with organization resources (accounts) for basic IAM Identity Center administrative tasks such as the following:

- Identifying the management account that belongs to the organization
- Identifying the member accounts that belong to the organization
- Enabling AWS service access for accounts
- Setting up and managing a delegated administrator

For more information about using a delegated administrator with IAM Identity Center, see Delegated administration (p. 91). For more information about how these permissions are used with AWS Organizations, see Using AWS Organizations with other AWS services.

AWS managed policy: AWSSSODirectoryAdministrator

You can attach the `AWSSSODirectoryAdministrator` policy to your IAM identities.

This policy grants administrative permissions over IAM Identity Center users and groups. Principals with this policy attached can make any updates to IAM Identity Center users and groups. The content of this policy statement is shown in the following snippet.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AWSSSODirectoryAdministrator",
            "Effect": "Allow",
            "Action": [
                "sso-directory:*",
                "identitystore:*",
                "identitystore-auth:*",
                "sso:ListDirectoryAssociations"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS managed policy: AWSSSOReadOnly

You can attach the `AWSSSOReadOnly` policy to your IAM identities.

This policy grants read-only permissions that allow users to view information in IAM Identity Center. Principals with this policy attached cannot view the IAM Identity Center users or groups directly. Principals with this policy attached cannot make any updates in IAM Identity Center. For example, principals with these permissions can view IAM Identity Center settings, but cannot change any of the setting values. The content of this policy statement is shown in the following snippet.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AWSSSOReadOnly",
            "Effect": "Allow",
            "Action": [
                "ds:DescribeDirectories",
                "ds:DescribeTrusts",
                "iam:ListPolicies",
                "organizations:DescribeOrganization",
```

```
            "organizations:DescribeAccount",
            "organizations:ListParents",
            "organizations:ListChildren",
            "organizations:ListAccounts",
            "organizations:ListRoots",
            "organizations:ListAccountsForParent",
            "organizations:ListOrganizationalUnitsForParent",
            "organizations:ListDelegatedAdministrators",
            "sso:Describe*",
            "sso:Get*",
            "sso:List*",
            "sso:Search*",
            "sso-directory:DescribeDirectory",
            "access-analyzer:ValidatePolicy"
        ],
            "Resource": "*"
        }
    ]
}
```

AWS managed policy: AWSSSODirectoryReadOnly

You can attach the `AWSSSODirectoryReadOnly` policy to your IAM identities.

This policy grants read-only permissions that allow users to view users and groups in IAM Identity Center. Principals with this policy attached cannot view IAM Identity Center assignments, permission sets, applications, or settings. Principals with this policy attached can't make any updates in IAM Identity Center. For example, principals with these permissions can view IAM Identity Center users, but they can't change any user attributes or assign MFA devices. The content of this policy statement is shown in the following snippet.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AWSSSODirectoryReadOnly",
            "Effect": "Allow",
            "Action": [
                "sso-directory:Search*",
                "sso-directory:Describe*",
                "sso-directory:List*",
                "sso-directory:Get*",
                "identitystore:Describe*",
                "identitystore:List*"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS managed policy: AWSIdentitySyncFullAccess

You can attach the `AWSIdentitySyncFullAccess` policy to your IAM identities.

Principals with this policy attached have full access permissions to create and delete sync profiles, associate or update a sync profile with a sync target, create, list and delete sync filters, and start or stop synchronization.

Permission details

This policy includes the following permissions when accessing Active Directory.

- `ds:AuthorizeApplication` – Allows identity-sync to grant access to the application during the sync profile creation process.
- `ds:UnauthorizeApplication` – Allows identity-sync to remove access to the application during the sync profile deletion process.

The content of this policy statement is shown in the following snippet.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ds:AuthorizeApplication",
                "ds:UnauthorizeApplication"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "identity-sync:DeleteSyncProfile",
                "identity-sync:CreateSyncProfile",
                "identity-sync:GetSyncProfile",
                "identity-sync:StartSync",
                "identity-sync:StopSync",
                "identity-sync:CreateSyncFilter",
                "identity-sync:DeleteSyncFilter",
                "identity-sync:ListSyncFilters",
                "identity-sync:CreateSyncTarget",
                "identity-sync:DeleteSyncTarget",
                "identity-sync:GetSyncTarget",
                "identity-sync:UpdateSyncTarget"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS managed policy: AWSIdentitySyncReadOnlyAccess

You can attach the `AWSIdentitySyncReadOnlyAccess` policy to your IAM identities.

This policy grants read-only permissions that allow users to view information about the identity synchronization profile, filters, and target settings. Principals with this policy attached can't make any updates to synchronization settings. For example, principals with these permissions can view identity synchronization settings, but can't change any of the profile or filter values. The content of this policy statement is shown in the following snippet.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "identity-sync:GetSyncProfile",
                "identity-sync:ListSyncFilters",
                "identity-sync:GetSyncTarget",
            ],
            "Resource": "*"
        }
```

```
    ]
}
```

AWS managed policy: AWSSSOServiceRolePolicy

You can't attach the `AWSSSOServiceRolePolicy` policy to your IAM identities.

This policy is attached to a service-linked role that allows IAM Identity Center to delegate and enforce which users have single sign-on access to specific AWS accounts in AWS Organizations. When you enable IAM, a service-linked role is created in all of the AWS accounts within your organization. IAM Identity Center also creates the same service-linked role in every account that is subsequently added to your organization. This role allows IAM Identity Center to access each account's resources on your behalf. Service-linked roles that are created in each AWS account are named `AWSServiceRoleForSSO`. For more information, see Using service-linked roles for IAM Identity Center (p. 151).

IAM Identity Center updates to AWS managed policies

The following table describes the updates to AWS managed policies for IAM Identity Center since this service began tracking these changes. For automatic alerts about changes to this page, subscribe to the RSS feed on the IAM Identity Center Document history page.

Change	Description	Date
AWSSSOServiceRolePolicy (p. 149)	This policy now allows the `UpdateSAMLProvider` action to be taken on the management account.	October 20, 2022
AWSSSOMasterAccountAdministrator (p. 140)	This policy now includes the new namespace `identitystore-auth` with new permissions to allow the admin to list and delete sessions for a user.	October 20, 2022
AWSSSOMemberAccountAdministrator (p. 144)	This policy now includes the new namespace `identitystore-auth` with new permissions to allow the admin to list and delete sessions for a user.	October 20, 2022
AWSSSODirectoryAdministrator (p. 146)	This policy now includes the new namespace `identitystore-auth` with new permissions to allow the admin to list and delete sessions for a user.	October 20, 2022
AWSSSOMasterAccountAdministrator (p. 140)	This policy now includes new permissions to call `ListDelegatedAdministrators` in AWS Organizations. This policy also now includes a subset of permissions AWSSSOManageDelegatedAdministrator that includes permissions to call `RegisterDelegatedAdministrator` and `DeregisterDelegatedAdministrator`.	August 16, 2022

Change	Description	Date
AWSSSOMemberAccountAdministration (p. 144)	This policy now includes new permissions to call `ListDelegatedAdministrators` in AWS Organizations. This policy also now includes a subset of permissions AWSSSOManageDelegatedAdministrator that includes permissions to call `RegisterDelegatedAdministrator` and `DeregisterDelegatedAdministrator`.	August 16, 2022
AWSSSOReadOnly (p. 146)	This policy now includes new permissions to call `ListDelegatedAdministrators` in AWS Organizations.	August 11, 2022
AWSSSOServiceRolePolicy (p. 149)	This policy now includes new permissions to call `DeleteRolePermissionsBoundary` and `PutRolePermisionsBoundary`.	July 14, 2022
AWSSSOServiceRolePolicy (p. 149)	This policy now includes new permissions that allow calls to `ListAWSServiceAccessForOrganization` and `ListDelegatedAdministrators` in AWS Organizations.	May 11, 2022
AWSSSOMasterAccountAdministrator (p. 141) AWSSSOMemberAccountAdministration (p. 144) AWSSSOReadOnly (p. 146)	Add IAM Access Analyzer permissions that allow a principal to use the policy checks for validation.	April 28, 2022
AWSSSOMasterAccountAdministrator (p. 141)	This policy now allows all IAM Identity Center Identity Store service actions. For information about the actions available in the IAM Identity Center Identity Store service, see the IAM Identity Center Identity Store API Reference.	March 29, 2022
AWSSSOMemberAccountAdministration (p. 144)	This policy now allows all IAM Identity Center Identity Store service actions.	March 29, 2022
AWSSSODirectoryAdministrator (p. 146)	This policy now allows all IAM Identity Center Identity Store service actions.	March 29, 2022

Change	Description	Date
AWSSSODirectoryReadOnly (p. 147)	This policy now grants access to the IAM Identity Center Identity Store service read actions. This access is required to retrieve user and group information from the IAM Identity Center Identity Store service.	March 29, 2022
AWSIdentitySyncFullAccess (p. 147)	This policy allows full access to identity-sync permissions.	March 3, 2022
AWSIdentitySyncReadOnlyAccess (p. 148)	This policy grants read-only permissions that allow a principal to view identity-sync settings.	March 3, 2022
AWSSSOReadOnly (p. 146)	This policy grants read-only permissions that allow a principal to view IAM Identity Center configuration settings.	August 4, 2021
IAM Identity Center started tracking changes	IAM Identity Center started tracking changes for AWS managed policies.	August 4, 2021

Using service-linked roles for IAM Identity Center

AWS IAM Identity Center (successor to AWS Single Sign-On) uses AWS Identity and Access Management (IAM) service-linked roles. A service-linked role is a unique type of IAM role that is linked directly to IAM Identity Center. It is predefined by IAM Identity Center and includes all the permissions that the service requires to call other AWS services on your behalf. For more information, see Service-linked roles (p. 110).

A service-linked role makes setting up IAM Identity Center easier because you don't have to manually add the necessary permissions. IAM Identity Center defines the permissions of its service-linked role, and unless defined otherwise, only IAM Identity Center can assume its role. The defined permissions include the trust policy and the permissions policy, and that permissions policy cannot be attached to any other IAM entity.

For information about other services that support service-linked roles, see AWS Services That Work with IAM and look for the services that have **Yes** in the **Service-Linked Role** column. Choose a **Yes** with a link to view the service-linked role documentation for that service.

Service-linked role permissions for IAM Identity Center

IAM Identity Center uses the service-linked role named **AWSServiceRoleForSSO** to grant IAM Identity Center permissions to manage AWS resources, including IAM roles, policies, and SAML IdP on your behalf.

The AWSServiceRoleForSSO service-linked role trusts the following services to assume the role:

- `IAM Identity Center`

The AWSServiceRoleForSSO service-linked role permissions policy allows IAM Identity Center to complete the following on roles on the path "/aws-reserved/sso.amazonaws.com/" and with the name prefix "AWSReservedSSO_":

- `iam:AttachRolePolicy`
- `iam:CreateRole`
- `iam:DeleteRole`
- `iam:DeleteRolePermissionsBoundary`
- `iam:DeleteRolePolicy`
- `iam:DetachRolePolicy`
- `iam:GetRole`
- `iam:ListRolePolicies`
- `iam:PutRolePolicy`
- `iam:PutRolePermissionsBoundary`
- `iam:ListAttachedRolePolicies`

The AWSServiceRoleForSSO service-linked role permissions policy allows IAM Identity Center to complete the following on SAML providers with name prefix as "AWSSSO_":

- `iam:CreateSAMLProvider`
- `iam:GetSAMLProvider`
- `iam:UpdateSAMLProvider`
- `iam:DeleteSAMLProvider`

The AWSServiceRoleForSSO service-linked role permissions policy allows IAM Identity Center to complete the following on all organizations:

- `organizations:DescribeAccount`
- `organizations:DescribeOrganization`
- `organizations:ListAccounts`
- `organizations:ListAWSServiceAccessForOrganization`
- `organizations:ListDelegatedAdministrators`

The AWSServiceRoleForSSO service-linked role permissions policy allows IAM Identity Center to complete the following on all IAM roles (*):

- `iam:listRoles`

The AWSServiceRoleForSSO service-linked role permissions policy allows IAM Identity Center to complete the following on "arn:aws:iam::*:role/aws-service-role/sso.amazonaws.com/AWSServiceRoleForSSO":

- `iam:GetServiceLinkedRoleDeletionStatus`
- `iam:DeleteServiceLinkedRole`

The role permissions policy allows IAM Identity Center to complete the following actions on resources.

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Sid":"IAMRoleProvisioningActions",
            "Effect":"Allow",
            "Action":[
```

```
            "iam:AttachRolePolicy",
            "iam:CreateRole",
            "iam:DeleteRolePermissionsBoundary",
            "iam:PutRolePermissionsBoundary",
            "iam:PutRolePolicy",
            "iam:UpdateRole",
            "iam:UpdateRoleDescription",
            "iam:UpdateAssumeRolePolicy"
        ],
        "Resource":[
            "arn:aws:iam::*:role/aws-reserved/sso.amazonaws.com/*"
        ],
        "Condition":{
            "StringNotEquals":{
                "aws:PrincipalOrgMasterAccountId":"${aws:PrincipalAccount}"
            }
        }
    },
    {
        "Sid":"IAMRoleReadActions",
        "Effect":"Allow",
        "Action":[
            "iam:GetRole",
            "iam:ListRoles"
        ],
        "Resource":[
            "*"
        ]
    },
    {
        "Sid":"IAMRoleCleanupActions",
        "Effect":"Allow",
        "Action":[
            "iam:DeleteRole",
            "iam:DeleteRolePolicy",
            "iam:DetachRolePolicy",
            "iam:ListRolePolicies",
            "iam:ListAttachedRolePolicies"
        ],
        "Resource":[
            "arn:aws:iam::*:role/aws-reserved/sso.amazonaws.com/*"
        ]
    },
    {
        "Sid":"IAMSLRCleanupActions",
        "Effect":"Allow",
        "Action":[
            "iam:DeleteServiceLinkedRole",
            "iam:GetServiceLinkedRoleDeletionStatus",
            "iam:DeleteRole",
            "iam:GetRole"
        ],
        "Resource":[
            "arn:aws:iam::*:role/aws-service-role/sso.amazonaws.com/AWSServiceRoleForSSO"
        ]
    },
    {
      "Sid": "IAMSAMLProviderCreationAction",
      "Effect": "Allow",
      "Action": [
        "iam:CreateSAMLProvider"
    ],
    "Resource": [
      "arn:aws:iam::*:saml-provider/AWSSSO_*"
     ],
    "Condition": {
```

```json
      "StringNotEquals": {
        "aws:PrincipalOrgMasterAccountId": "${aws:PrincipalAccount}"
      }
    }
  },
  {
    "Sid": "IAMSAMLProviderUpdateAction",
    "Effect": "Allow",
    "Action": [
      "iam:UpdateSAMLProvider"
    ],
    "Resource": [
      "arn:aws:iam::*:saml-provider/AWSSSO_*"
    ]
  },
  {
    "Sid":"IAMSAMLProviderCleanupActions",
    "Effect":"Allow",
    "Action":[
      "iam:DeleteSAMLProvider",
      "iam:GetSAMLProvider"
    ],
    "Resource":[
      "arn:aws:iam::*:saml-provider/AWSSSO_*"
    ]
  },
  {
    "Effect":"Allow",
    "Action":[
      "organizations:DescribeAccount",
      "organizations:DescribeOrganization",
      "organizations:ListAccounts",
      "organizations:ListAWSServiceAccessForOrganization",
      "organizations:ListDelegatedAdministrators"
    ],
    "Resource":[
      "*"
    ]
  },
  {
    "Sid":"AllowUnauthAppForDirectory",
    "Effect":"Allow",
    "Action":[
      "ds:UnauthorizeApplication"
    ],
    "Resource":[
      "*"
    ]
  },
  {
    "Sid":"AllowDescribeForDirectory",
    "Effect":"Allow",
    "Action":[
      "ds:DescribeDirectories",
      "ds:DescribeTrusts"
    ],
    "Resource":[
      "*"
    ]
  },
  {
    "Sid":"AllowDescribeAndListOperationsOnIdentitySource",
    "Effect":"Allow",
    "Action":[
      "identitystore:DescribeUser",
      "identitystore:DescribeGroup",
```

```
                "identitystore:ListGroups",
                "identitystore:ListUsers"
            ],
            "Resource":[
                "*"
            ]
        }
    ]
}
```

You must configure permissions to allow an IAM entity (such as a user, group, or role) to create, edit, or delete a service-linked role. For more information, see Service-linked role permissions in the *IAM User Guide*.

Creating a service-linked role for IAM Identity Center

You don't need to manually create a service-linked role. Once enabled, IAM Identity Center creates a service-linked role in all accounts within the organization in AWS Organizations. IAM Identity Center also creates the same service-linked role in every account that is subsequently added to your organization. This role allows IAM Identity Center to access each account's resources on your behalf.

> **Note**
> If you're signed in to the management account, it uses your currently signed-in role and not the the service-linked role. This prevents the escalation of privileges.

> **Important**
> If you were using the IAM Identity Center service before December 7, 2017, when it began supporting service-linked roles, then IAM Identity Center created the AWSServiceRoleForSSO role in your account. To learn more, see A New Role Appeared in My IAM Account.

If you delete this service-link role and then need to create it again, you can use the same process to recreate the role in your account.

Editing a service-linked role for IAM Identity Center

IAM Identity Center does not allow you to edit the AWSServiceRoleForSSO service-linked role. After you create a service-linked role, you cannot change the name of the role because various entities might reference the role. However, you can edit the description of the role using IAM. For more information, see Editing a service-linked role in the *IAM User Guide*.

Deleting a service-linked role for IAM Identity Center

You don't need to manually delete the AWSServiceRoleForSSO role. When an AWS account is removed from an AWS organization, IAM Identity Center automatically cleans up the resources and deletes the service-linked role from that AWS account.

You can also use the IAM console, the IAM CLI, or the IAM API to manually delete the service-linked role. To do this, you must first manually clean up the resources for your service-linked role and then you can manually delete it.

> **Note**
> If the IAM Identity Center service is using the role when you try to delete the resources, then the deletion might fail. If that happens, wait for a few minutes and try the operation again.

To delete IAM Identity Center resources used by the AWSServiceRoleForSSO

1. Remove user and group access (p. 96) for all users and groups that have access to the AWS account.
2. Delete permission sets (p. 103) that you have associated with the AWS account.

To manually delete the service-linked role using IAM

Use the IAM console, the IAM CLI, or the IAM API to delete the AWSServiceRoleForSSO service-linked role. For more information, see Deleting a Service-Linked Role in the *IAM User Guide*.

IAM Identity Center console and API authorization

Existing IAM Identity Center console APIs support dual authorization. If you have existing IAM Identity Center instances that were created prior to October 15th 2020, you can use the following table to determine which API operations now map to newer API operations that were released after that date.

IAM Identity Center instances created before October 15th 2020 honor both old and new API actions as long as there is no explicit deny on any of the actions. Instances created after October 15th 2020 use the newer API actions for authorization in the IAM Identity Center console.

Operation name	API actions used before October 15th, 2020	API actions used after October 15th, 2020
AssociateProfile	AssociateProfile	CreateAccountAssignment
AttachManagedPolicy	PutPermissionsPolicy	AttachManagedPolicyToPermissionSet
CreatePermissionSet	CreatePermissionSet	CreatePermissionSet
DeleteApplicationInstanceForAWSAccount	DeleteApplicationInstance | DeleteTrust	DeleteAccountAssignment
DeleteApplicationProfileForAwsAccount	DeleteProfile	DeleteAccountAssignment
DeletePermissionsPolicy	DeletePermissionsPolicy	DeleteInlinePolicyFromPermissionSet
DeletePermissionSet	DeletePermissionSet	DeletePermissionSet
DescribePermissionsPolicies	DescribePermissionsPolicies	ListManagedPoliciesInPermissionSet
DetachManagedPolicy	DeletePermissionsPolicy	DetachManagedPolicyFromPermissionSet
DisassociateProfile	DisassociateProfile	DeleteAccountAssignment
GetApplicationInstanceForAWSAccount	GetApplicationInstance	ListAccountAssignments
GetAWSAccountProfileStatus	GetProfile	ListPermissionSetsProvisionedToAccount
GetPermissionSet	GetPermissionSet	DescribePermissionSet
GetPermissionsPolicy	GetPermissionsPolicy	GetInlinePolicyForPermissionSet
ListAccountsWithProvisionedPermissionSet	ListApplicationInstances | GetApplicationInstance	ListAccountsForProvisionedPermissionSet
ListAWSAccountProfiles	ListProfiles | GetProfile	ListPermissionSetsProvisionedToAccount
ListPermissionSets	ListPermissionSets	ListPermissionSets
ListProfileAssociations	ListProfileAssociations	ListAccountAssignments
ProvisionApplicationInstanceForAWSAccount	SetApplicationInstance | CreateApplicationInstance	CreateAccountAssignment

Operation name	API actions used before October 15th, 2020	API actions used after October 15th, 2020
ProvisionApplicationProfileForAWSAccountInstance	GetProfile	CreateProfile	UpdateProfile	CreateAccountAssignment
ProvisionSAMLProvider	GetTrust	CreateTrust	UpdateTrust	CreateAccountAssignment
PutPermissionsPolicy	PutPermissionsPolicy	PutInlinePolicyToPermissionSet
UpdatePermissionSet	UpdatePermissionSet	UpdatePermissionSet

Logging and monitoring in IAM Identity Center

As a best practice, you should monitor your organization to ensure that changes are logged. This helps you to ensure that any unexpected change can be investigated and unwanted changes can be rolled back. AWS IAM Identity Center (successor to AWS Single Sign-On) currently supports two AWS services that help you monitor your organization and the activity that happens within it.

Topics

Logging IAM Identity Center API calls with AWS CloudTrail

IAM Identity Center is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in IAM Identity Center. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, Amazon CloudWatch Logs, and Amazon CloudWatch Events. Using the information collected by CloudTrail, you can determine the request that was made to IAM Identity Center, the IP address from which the request was made, who made the request, when it was made, and additional details.

To learn more about CloudTrail, see the AWS CloudTrail User Guide.

Topics

IAM Identity Center information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in IAM Identity Center, that activity is recorded in a CloudTrail event along with other AWS service events in **Event history**. You can view, search, and download recent events in your AWS account. For more information, see Viewing events with CloudTrail event history.

For an ongoing record of events in your AWS account, including events for IAM Identity Center, create a trail. A *trail* enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all AWS Regions. The trail logs events from all Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can

configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see the following:

- Overview for creating a trail
- CloudTrail supported services and integrations
- Configuring Amazon SNS notifications for CloudTrail
- Receiving CloudTrail log files from multiple Regions and Receiving CloudTrail log files from multiple accounts

When CloudTrail logging is enabled in your AWS account, API calls made to IAM Identity Center actions are tracked in log files. IAM Identity Center records are written together with other AWS service records in a log file. CloudTrail determines when to create and write to a new file based on a time period and file size.

The following IAM Identity Center CloudTrail operations are supported:

Console API operations	Public API operations
AssociateDirectory	AttachManagedPolicyToPermissionSet
AssociateProfile	CreateAccountAssignment
BatchDeleteSession	CreateInstanceAccessControlAttributeConfiguration
BatchGetSession	CreatePermissionSet
CreateApplicationInstance	DeleteAccountAssignment
CreateApplicationInstanceCertificate	DeleteInlinePolicyFromPermissionSet
CreatePermissionSet	DeleteInstanceAccessControlAttributeConfiguration
CreateProfile	DeletePermissionSet
DeleteApplicationInstance	DescribeAccountAssignmentCreationStatus
DeleteApplicationInstanceCertificate	DescribeAccountAssignmentDeletionStatus
DeletePermissionsPolicy	DescribeInstanceAccessControlAttributeConfiguration
DeletePermissionSet	DescribePermissionSet
DeleteProfile	DescribePermissionSetProvisioningStatus
DescribePermissionsPolicies	DetachManagedPolicyFromPermissionSet
DisassociateDirectory	GetInlinePolicyForPermissionSet
DisassociateProfile	ListAccountAssignmentCreationStatus
GetApplicationInstance	ListAccountAssignmentDeletionStatus
GetApplicationTemplate	ListAccountAssignments
GetMfaDeviceManagementForDirectory	ListAccountsForProvisionedPermissionSet
GetPermissionSet	ListInstances
GetSSOStatus	ListManagedPoliciesInPermissionSet

Console API operations	Public API operations
`ImportApplicationInstanceServiceProviderMetadata`	`ListPermissionSetProvisioningStatus`
`ListApplicationInstances`	`ListPermissionSets`
`ListApplicationInstanceCertificates`	`ListPermissionSetsProvisionedToAccount`
`ListApplicationTemplates`	`ListTagsForResource`
`ListDirectoryAssociations`	`ProvisionPermissionSet`
`ListPermissionSets`	`PutInlinePolicyToPermissionSet`
`ListProfileAssociations`	`TagResource`
`ListProfiles`	`UntagResource`
`ListSessions`	`UpdateInstanceAccessControlAttributeConfiguration`
`PutMfaDeviceManagementForDirectory`	`UpdatePermissionSet`
`PutPermissionsPolicy`	
`StartSSO`	
`UpdateApplicationInstanceActiveCertificate`	
`UpdateApplicationInstanceDisplayData`	
`UpdateApplicationInstanceServiceProviderConfiguration`	
`UpdateApplicationInstanceStatus`	
`UpdateApplicationInstanceResponseConfiguration`	
`UpdateApplicationInstanceResponseSchemaConfiguration`	
`UpdateApplicationInstanceSecurityConfiguration`	
`UpdateDirectoryAssociation`	
`UpdateProfile`	

For more information about IAM Identity Center's public API operations, see the IAM Identity Center API Reference Guide.

The following IAM Identity Center identity store CloudTrail operations are supported:

- `AddMemberToGroup`
- `CompleteVirtualMfaDeviceRegistration`
- `CompleteWebAuthnDeviceRegistration`
- `CreateAlias`
- `CreateExternalIdPConfigurationForDirectory`
- `CreateGroup`
- `CreateUser`
- `DeleteExternalIdPConfigurationForDirectory`
- `DeleteGroup`

- DeleteMfaDeviceForUser
- DeleteUser
- DescribeDirectory
- DescribeGroups
- DescribeUsers
- DisableExternalIdPConfigurationForDirectory
- DisableUser
- EnableExternalIdPConfigurationForDirectory
- EnableUser
- GetAWSSPConfigurationForDirectory
- ListExternalIdPConfigurationsForDirectory
- ListGroupsForUser
- ListMembersInGroup
- ListMfaDevicesForUser
- PutMfaDeviceManagementForDirectory
- RemoveMemberFromGroup
- SearchGroups
- SearchUsers
- StartVirtualMfaDeviceRegistration
- StartWebAuthnDeviceRegistration
- UpdateExternalIdPConfigurationForDirectory
- UpdateGroup
- UpdateMfaDeviceForUser
- UpdatePassword
- UpdateUser
- VerifyEmail

The following IAM Identity Center OIDC CloudTrail actions are supported:

- CreateToken
- RegisterClient
- StartDeviceAuthorization

The following IAM Identity Center Portal CloudTrail actions are supported:

- Authenticate
- Federate

Every log entry contains information about who generated the request. The identity information in the log helps you determine whether the request was made by an AWS account root user or with IAM user credentials. You can also learn whether the request was made with temporary security credentials for a role or federated user or by another AWS service. For more information, see the CloudTrail userIdentity Element.

You can create a trail and store your log files in your Amazon S3 bucket for as long as you want. You can also define Amazon S3 lifecycle rules to archive or delete log files automatically. By default, your log files are encrypted with Amazon S3 server-side encryption (SSE).

To be notified of log file delivery, configure CloudTrail to publish Amazon SNS notifications when new log files are delivered. For more information, see Configuring Amazon SNS notifications for CloudTrail.

You can also aggregate IAM Identity Center log files from multiple AWS Regions and multiple AWS accounts into a single Amazon S3 bucket. For more information, see Receiving CloudTrail log files from multiple Regions and Receiving CloudTrail log files from multiple accounts.

Understanding IAM Identity Center log file entries

A trail is a configuration that enables delivery of events as log files to an Amazon S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single request from any source and includes information about the requested action, the date and time of the action, request parameters, and so on. CloudTrail log files are not an ordered stack trace of the public API calls, so they do not appear in any specific order.

The following example shows a CloudTrail log entry for an administrator (samadams@example.com) that took place in the IAM Identity Center console:

```
{
    "Records":[
        {
            "eventVersion":"1.05",
            "userIdentity":{
                "type":"IAMUser",
                "principalId":"AIDAJAIENLMexample",
                "arn":"arn:aws:iam::08966example:user/samadams",
                "accountId":"08966example",
                "accessKeyId":"AKIAIIJM2K4example",
                "userName":"samadams"
            },
            "eventTime":"2017-11-29T22:39:43Z",
            "eventSource":"sso.amazonaws.com",
            "eventName":"DescribePermissionsPolicies",
            "awsRegion":"us-east-1",
            "sourceIPAddress":"203.0.113.0",
            "userAgent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_6) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/62.0.3202.94 Safari/537.36",
            "requestParameters":{
                "permissionSetId":"ps-79a0dde74b95ed05"
            },
            "responseElements":null,
            "requestID":"319ac6a1-d556-11e7-a34f-69a333106015",
            "eventID":"a93a952b-13dd-4ae5-a156-d3ad6220b071",
            "readOnly":true,
            "resources":[

            ],
            "eventType":"AwsApiCall",
            "recipientAccountId":"08966example"
        }
    ]
}
```

The following example shows a CloudTrail log entry for an end-user (bobsmith@example.com) action that took place in the AWS access portal:

```
{
    "Records":[
        {
            "eventVersion":"1.05",
            "userIdentity":{
                "type":"Unknown",
```

```
            "principalId":"example.com//S-1-5-21-1122334455-3652759393-4233131409-1126",
            "accountId":"08966example",
            "userName":"bobsmith@example.com"
          },
          "eventTime":"2017-11-29T18:48:28Z",
          "eventSource":"sso.amazonaws.com",
          "eventName":"https://portal.sso.us-east-1.amazonaws.com/instance/appinstances",
          "awsRegion":"us-east-1",
          "sourceIPAddress":"203.0.113.0",
          "userAgent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_6) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/62.0.3202.94 Safari/537.36",
          "requestParameters":null,
          "responseElements":null,
          "requestID":"de6c0435-ce4b-49c7-9bcc-bc5ed631ce04",
          "eventID":"e6e1f3df-9528-4c6d-a877-6b2b895d1f91",
          "eventType":"AwsApiCall",
          "recipientAccountId":"08966example"
      }
    ]
}
```

The following example shows a CloudTrail log entry for an end-user (bobsmith@example.com) action
that took place in IAM Identity Center OIDC:

```
{
      "eventVersion": "1.05",
      "userIdentity": {
        "type": "Unknown",
        "principalId": "example.com//S-1-5-21-1122334455-3652759393-4233131409-1126",
        "accountId": "08966example",
        "userName": "bobsmith@example.com"
      },
      "eventTime": "2020-06-16T01:31:15Z",
      "eventSource": "sso.amazonaws.com",
      "eventName": "CreateToken",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "203.0.113.0",
      "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_6) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/62.0.3202.94 Safari/537.36",
      "requestParameters": {
        "clientId": "clientid1234example",
        "clientSecret": "HIDDEN_DUE_TO_SECURITY_REASONS",
        "grantType": "urn:ietf:params:oauth:grant-type:device_code",
        "deviceCode": "devicecode1234example"
      },
      "responseElements": {
        "accessToken": "HIDDEN_DUE_TO_SECURITY_REASONS",
        "tokenType": "Bearer",
        "expiresIn": 28800,
        "refreshToken": "HIDDEN_DUE_TO_SECURITY_REASONS",
        "idToken": "HIDDEN_DUE_TO_SECURITY_REASONS"
      },
      "eventID": "09a6e1a9-50e5-45c0-9f08-e6ef5089b262",
      "readOnly": false,
      "resources": [
        {
          "accountId": "08966example",
          "type": "IdentityStoreId",
          "ARN": "d-1234example"
        }
      ],
      "eventType": "AwsApiCall",
      "recipientAccountId": "08966example"
}
```

Understanding IAM Identity Center sign-in events

AWS CloudTrail logs successful and unsuccessful sign-in events for all AWS IAM Identity Center (successor to AWS Single Sign-On) identity sources. Native SSO and Active Directory (AD Connector and AWS Managed Microsoft AD) sourced identities will include additional sign-in events that are captured each time a user is prompted to solve a specific credential challenge or factor, as well as the status of that particular credential verification request. Only after a user has completed all required credential challenges will the user be signed in, which will result in a `UserAuthentication` event being logged.

The following table captures each of the IAM Identity Center sign-in CloudTrail event names, their purpose, and applicability to different identity sources.

Event name	Event purpose	Identity source applicability
`CredentialChallenge`	Used to notify that IAM Identity Center has requested the user to solve a specific credential challenge and specifies the `CredentialType` that was required (For example, PASSWORD or TOTP).	Native IAM Identity Center users, AD Connector, and AWS Managed Microsoft AD
`CredentialVerification`	Used to notify that the user has attempted to solve a specific `CredentialChallenge` request and specifies whether that credential succeeded or failed.	Native IAM Identity Center users, AD Connector, and AWS Managed Microsoft AD
`UserAuthentication`	Used to notify that all authentication requirements the user was challenged with have been successfully completed and that the user was successfully signed in. *Users failing to successfully complete the required credential challenges will result in no `UserAuthentication` event being logged.*	All identity sources

The following table captures additional useful event data fields contained within specific sign-in CloudTrail events.

Event name	Event purpose	Sign-in event applicability	Example values
`AuthWorkflowID`	Used to correlate all events emitted across an entire sign-in sequence. For each user sign-in, multiple events may be emitted by IAM Identity Center.	`CredentialChallenge`, `CredentialVerification`, `UserAuthentication`	"AuthWorkflowID": "9de74b32-8362-4a01-a524-de21df59fd83"

Event name	Event purpose	Sign-in event applicability	Example values
CredentialType	Used to specify the credential or factor that was challenged. UserAuthentication events will include all of the CredentialType values that were successfully verified across the user's sign-in sequence.	CredentialChallenge, CredentialVerification, UserAuthentication	CredentialType": "PASSWORD" or "CredentialType": "PASSWORD,TOTP" (possible values include: PASSWORD, TOTP, WEBAUTHN, EXTERNAL_IDP, RESYNC_TOTP)
DeviceEnrollmentRequired	Used to specify that the user was required to register an MFA device during sign-in, and that the user successfully completed that request.	UserAuthentication	"DeviceEnrollmentRequired": "true"
LoginTo	Used to specify the redirect location following a successful sign-in sequence.	UserAuthentication	"LoginTo": "https:// mydirectory.awsapps.com/ start/....."

Example events for IAM Identity Center sign-in scenarios

The following examples show the expected sequence of CloudTrail events for different sign-in scenarios.

Topics

Successful sign-in when authenticating with only a password

The following sequence of events captures an example of a successful password only sign-in.

CredentialChallenge (Password)

```
{
    "eventVersion":"1.08",
    "userIdentity":{
        "type":"Unknown",
        "principalId":"111122223333",
        "arn":"",
        "accountId":"111122223333",
        "accessKeyId":"",
        "userName":"user1"
    },
    "eventTime":"2020-12-07T20:33:58Z",
    "eventSource":"signin.amazonaws.com",
```

```
    "eventName":"CredentialChallenge",
    "awsRegion":"us-east-1",
    "sourceIPAddress":"203.0.113.0",
    "userAgent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/87.0.4280.66 Safari/537.36",
    "requestParameters":null,
    "responseElements":null,
    "additionalEventData":{
        "AuthWorkflowID":"9de74b32-8362-4a01-a524-de21df59fd83",
        "CredentialType":"PASSWORD"
    },
    "requestID":"5be44ffb-6946-4f47-acaf-1adebd4afead",
    "eventID":"27ea7725-c1fd-4355-bdba-d0e628e0e604",
    "readOnly":false,
    "eventType":"AwsServiceEvent",
    "managementEvent":true,
    "eventCategory":"Management",
    "recipientAccountId":"111122223333",
    "serviceEventDetails":{
        "CredentialChallenge":"Success"
    }
}
```

Successful CredentialVerification (Password)

```
{
    "eventVersion":"1.08",
    "userIdentity":{
        "type":"Unknown",
        "principalId":"111122223333",
        "arn":"",
        "accountId":"111122223333",
        "accessKeyId":"",
        "userName":"user1"
    },
    "eventTime":"2020-12-07T20:34:09Z",
    "eventSource":"signin.amazonaws.com",
    "eventName":"CredentialVerification",
    "awsRegion":"us-east-1",
    "sourceIPAddress":"203.0.113.0",
    "userAgent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/87.0.4280.66 Safari/537.36",
    "requestParameters":null,
    "responseElements":null,
    "additionalEventData":{
        "AuthWorkflowID":"9de74b32-8362-4a01-a524-de21df59fd83",
        "CredentialType":"PASSWORD"
    },
    "requestID":"f3cf52ad-fd3d-4889-8c15-f18d1a7c7393",
    "eventID":"c49640f6-0c8a-43d3-a6e0-900e3bb188d4",
    "readOnly":false,
    "eventType":"AwsServiceEvent",
    "managementEvent":true,
    "eventCategory":"Management",
    "recipientAccountId":"111122223333",
    "serviceEventDetails":{
        "CredentialVerification":"Success"
    }
}
```

Successful UserAuthentication (Password Only)

```
{
    "eventVersion":"1.08",
```

```
    "userIdentity":{
        "type":"Unknown",
        "principalId":"111122223333",
        "arn":"",
        "accountId":"111122223333",
        "accessKeyId":"",
        "userName":"user1"
    },
    "eventTime":"2020-12-07T20:34:09Z",
    "eventSource":"signin.amazonaws.com",
    "eventName":"UserAuthentication",
    "awsRegion":"us-east-1",
    "sourceIPAddress":"203.0.113.0",
    "userAgent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/87.0.4280.66 Safari/537.36",
    "requestParameters":null,
    "responseElements":null,
    "additionalEventData":{
        "AuthWorkflowID":"9de74b32-8362-4a01-a524-de21df59fd83",
        "LoginTo":"https://d-1234567890.awsapps.com/start/?
state=QVlBQmVGMHFiS0wzWlp1SFgrR25BRnFobU5nQUlnQUJBQk5FWVhSaFVHeGQGibVZUZEdGMFpWQmhjbUZ0QUFsUVpYSmxaM0pwwY
BshlIc5OBAA6ftz73M6LsfLWDlfOxviO2K3wet946lC30f_iWdilx-
zv__4pSHf7mcUIs&wdc_csrf_token=srAzW1jK4GPYYoR452ruZ38DxEsDY9x81q1tVRSnno5pUjISvP7TqziOLiBLBUSxEjOmQk2X
east-1",
        "CredentialType":"PASSWORD"
    },
    "requestID":"f3cf52ad-fd3d-4889-8c15-f18d1a7c7393",
    "eventID":"e959a95a-2b33-478d-906c-4fe303e8a9f1",
    "readOnly":false,
    "eventType":"AwsServiceEvent",
    "managementEvent":true,
    "eventCategory":"Management",
    "recipientAccountId":"111122223333",
    "serviceEventDetails":{
        "UserAuthentication":"Success"
    }
}
```

Successful sign-in when authenticating with an external identity provider

The following sequence of events captures an example of a successful sign-in when authenticated through the SAML protocol using an external identity provider.

Successful UserAuthentication (External Identity Provider)

```
{
    "eventVersion":"1.08",
    "userIdentity":{
        "type":"Unknown",
        "principalId":"111122223333",
        "arn":"",
        "accountId":"111122223333",
        "accessKeyId":"",
        "userName":"user1"
    },
    "eventTime":"2020-12-07T20:34:09Z",
    "eventSource":"signin.amazonaws.com",
    "eventName":"UserAuthentication",
    "awsRegion":"us-east-1",
    "sourceIPAddress":"203.0.113.0",
    "userAgent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/87.0.4280.66 Safari/537.36",
    "requestParameters":null,
    "responseElements":null,
```

```
    "additionalEventData":{
        "AuthWorkflowID":"9de74b32-8362-4a01-a524-de21df59fd83",
        "LoginTo":"https://d-1234567890.awsapps.com/start/?
state=QVlBQmVGMHFiS0wzWlp1SFgrR25BRnFobU5nQUlnQUJBQk5FWVhSaFVHeGhibVZUZUdGFpWQmhjbUZ0QUFsUVpYSmxaM0pwY
BshlIc5OBAA6ftz73M6LsfLWDlfOxviO2K3wet946lC30f_iWdilx-
zv__4pSHf7mcUIs&wdc_csrf_token=srAzW1jK4GPYYoR452ruZ38DxEsDY9x81q1tVRSnno5pUjISvP7TqziOLiBLBUSxEjOmQk2X
east-1",
        "CredentialType":"EXTERNAL_IDP"
    },
    "requestID":"f3cf52ad-fd3d-4889-8c15-f18d1a7c7393",
    "eventID":"e959a95a-2b33-478d-906c-4fe303e8a9f1",
    "readOnly":false,
    "eventType":"AwsServiceEvent",
    "managementEvent":true,
    "eventCategory":"Management",
    "recipientAccountId":"111122223333",
    "serviceEventDetails":{
        "UserAuthentication":"Success"
    }
}
```

Successful sign-in when authenticating with a password and a TOTP authenticator app

The following sequence of events captures an example where multi-factor authentication was required during sign-in and the user successfully signed in using a password and a TOTP authenticator app.

CredentialChallenge (Password)

```
{
    "eventVersion":"1.08",
    "userIdentity":{
        "type":"Unknown",
        "principalId":"111122223333",
        "arn":"",
        "accountId":"111122223333",
        "accessKeyId":"",
        "userName":"user1"
    },
    "eventTime":"2020-12-08T20:40:13Z",
    "eventSource":"signin.amazonaws.com",
    "eventName":"CredentialChallenge",
    "awsRegion":"us-east-1",
    "sourceIPAddress":"203.0.113.0",
    "userAgent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/87.0.4280.66 Safari/537.36",
    "requestParameters":null,
    "responseElements":null,
    "additionalEventData":{
        "AuthWorkflowID":"303486b5-fce1-4d59-ba1d-eb3acb790729",
        "CredentialType":"PASSWORD"
    },
    "requestID":"e454ea66-1027-4d00-9912-09c0589649e1",
    "eventID":"d89cc0b5-a23a-4b88-843a-89329aeaef2e",
    "readOnly":false,
    "eventType":"AwsServiceEvent",
    "managementEvent":true,
    "eventCategory":"Management",
    "recipientAccountId":"111122223333",
    "serviceEventDetails":{
        "CredentialChallenge":"Success"
    }
}
```

Successful CredentialVerification (Password)

```
{
    "eventVersion":"1.08",
    "userIdentity":{
        "type":"Unknown",
        "principalId":"111122223333",
        "arn":"",
        "accountId":"111122223333",
        "accessKeyId":"",
        "userName":"user1"
    },
    "eventTime":"2020-12-08T20:40:20Z",
    "eventSource":"signin.amazonaws.com",
    "eventName":"CredentialVerification",
    "awsRegion":"us-east-1",
    "sourceIPAddress":"203.0.113.0",
    "userAgent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/87.0.4280.66 Safari/537.36",
    "requestParameters":null,
    "responseElements":null,
    "additionalEventData":{
        "AuthWorkflowID":"303486b5-fce1-4d59-ba1d-eb3acb790729",
        "CredentialType":"PASSWORD"
    },
    "requestID":"92c4ac90-0d9b-452d-95d5-728487612f5e",
    "eventID":"4533fd49-6669-4d0b-b272-a0b2139309a8",
    "readOnly":false,
    "eventType":"AwsServiceEvent",
    "managementEvent":true,
    "eventCategory":"Management",
    "recipientAccountId":"111122223333",
    "serviceEventDetails":{
        "CredentialVerification":"Success"
    }
}
```

CredentialChallenge (TOTP)

```
{
    "eventVersion":"1.08",
    "userIdentity":{
        "type":"Unknown",
        "principalId":"111122223333",
        "arn":"",
        "accountId":"111122223333",
        "accessKeyId":"",
        "userName":"user1"
    },
    "eventTime":"2020-12-08T20:40:20Z",
    "eventSource":"signin.amazonaws.com",
    "eventName":"CredentialChallenge",
    "awsRegion":"us-east-1",
    "sourceIPAddress":"203.0.113.0",
    "userAgent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/87.0.4280.66 Safari/537.36",
    "requestParameters":null,
    "responseElements":null,
    "additionalEventData":{
        "AuthWorkflowID":"303486b5-fce1-4d59-ba1d-eb3acb790729",
        "CredentialType":"TOTP"
    },
    "requestID":"92c4ac90-0d9b-452d-95d5-728487612f5e",
    "eventID":"29202f08-f240-40cc-b789-c0cea8a27847",
    "readOnly":false,
    "eventType":"AwsServiceEvent",
```

```
    "managementEvent":true,
    "eventCategory":"Management",
    "recipientAccountId":"111122223333",
    "serviceEventDetails":{
        "CredentialChallenge":"Success"
    }
}
```

Successful CredentialVerification (TOTP)

```
{
    "eventVersion":"1.08",
    "userIdentity":{
        "type":"Unknown",
        "principalId":"111122223333",
        "arn":"",
        "accountId":"111122223333",
        "accessKeyId":"",
        "userName":"user1"
    },
    "eventTime":"2020-12-08T20:40:27Z",
    "eventSource":"signin.amazonaws.com",
    "eventName":"CredentialVerification",
    "awsRegion":"us-east-1",
    "sourceIPAddress":"203.0.113.0",
    "userAgent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/87.0.4280.66 Safari/537.36",
    "requestParameters":null,
    "responseElements":null,
    "additionalEventData":{
        "AuthWorkflowID":"303486b5-fce1-4d59-ba1d-eb3acb790729",
        "CredentialType":"TOTP"
    },
    "requestID":"c40a691f-eeb1-4352-b286-5e909f96f318",
    "eventID":"e889ff1d-fcaf-454f-805d-7132cf2362a4",
    "readOnly":false,
    "eventType":"AwsServiceEvent",
    "managementEvent":true,
    "eventCategory":"Management",
    "recipientAccountId":"111122223333",
    "serviceEventDetails":{
        "CredentialVerification":"Success"
    }
}
```

Successful UserAuthentication (Password + TOTP)

```
{
    "eventVersion":"1.08",
    "userIdentity":{
        "type":"Unknown",
        "principalId":"111122223333",
        "arn":"",
        "accountId":"111122223333",
        "accessKeyId":"",
        "userName":"user1"
    },
    "eventTime":"2020-12-08T20:40:27Z",
    "eventSource":"signin.amazonaws.com",
    "eventName":"UserAuthentication",
    "awsRegion":"us-east-1",
    "sourceIPAddress":"203.0.113.0",
    "userAgent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/87.0.4280.66 Safari/537.36",
```

```
    "requestParameters":null,
    "responseElements":null,
    "additionalEventData":{
        "AuthWorkflowID":"303486b5-fce1-4d59-ba1d-eb3acb790729",
        "LoginTo":"https://d-1234567890.awsapps.com/start/?state
\u003dQVlBQmVLeFhWeDRmZFJmMmxHcWYwdzhZck5RQUlnQUJBQk5FWVhSaFVHeGhibVZUZUdGMFpWQmhjbUZZ0QUFsUVpYSmxaM0pwV\
\u0026auth_code
\u003d11Fir1mCVJ-4Y5UY6RI10UCXvRePCHd6195xvYg1rwo1Pj7B-7UGIGlYUUVe31Nkzd7ihxKn6DMdnFfO01O8qc3RFR8FUd1w8
Sx-pjBXKG_jUcvBk_UILdGytV4o1u97h42B-
TA_6uwdmJiw1dcCz_Rv44d_BS0PkulW-5LVJy1oeP1H0FPPMeheyuk5Uy48d5of9-c\u0026wdc_csrf_token
\u003dNMlui44guoVnxRd0qu2tYJIdyyFPX6SDRNTspIScfMM0AgFbho1nvvCaxPTghHbgHCRIXdffFtzH0sL1ow419BobnmqBsnJNx
\u0026organization\u003dd-9067230c03\u0026region\u003dus-east-1",
        "CredentialType":"PASSWORD,TOTP"
    },
    "requestID":"c40a691f-eeb1-4352-b286-5e909f96f318",
    "eventID":"7a8c8725-db2f-488d-a43e-788dc6c73a4a",
    "readOnly":false,
    "eventType":"AwsServiceEvent",
    "managementEvent":true,
    "eventCategory":"Management",
    "recipientAccountId":"111122223333",
    "serviceEventDetails":{
        "UserAuthentication":"Success"
    }
}
```

Successful sign-in when authenticating with a password and forced MFA registration is required

The following sequence of events captures an example of a successful password sign in, but the user was required and successfully completed registering an MFA device before completing their sign-in.

CredentialChallenge (Password)

```
{
    "eventVersion":"1.08",
    "userIdentity":{
        "type":"Unknown",
        "principalId":"111122223333",
        "arn":"",
        "accountId":"111122223333",
        "accessKeyId":"",
        "userName":"user1"
    },
    "eventTime":"2020-12-09T01:24:02Z",
    "eventSource":"signin.amazonaws.com",
    "eventName":"CredentialChallenge",
    "awsRegion":"us-east-1",
    "sourceIPAddress":"203.0.113.0",
    "userAgent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/87.0.4280.66 Safari/537.36",
    "requestParameters":null,
    "responseElements":null,
    "additionalEventData":{
        "AuthWorkflowID":"76d8a26d-ad9c-41a4-90c3-d607cdd7155c",
        "CredentialType":"PASSWORD"
    },
    "requestID":"321f4b13-42b5-4005-a0f7-826cad26d159",
    "eventID":"8c707b0f-e45a-4a9c-bee2-ff68638d2f1b",
    "readOnly":false,
    "eventType":"AwsServiceEvent",
    "managementEvent":true,
    "eventCategory":"Management",
    "recipientAccountId":"111122223333",
    "serviceEventDetails":{
```

```
        "CredentialChallenge":"Success"
    }
}
```

Successful CredentialVerification (Password)

```
{
    "eventVersion":"1.08",
    "userIdentity":{
        "type":"Unknown",
        "principalId":"111122223333",
        "arn":"",
        "accountId":"111122223333",
        "accessKeyId":"",
        "userName":"user1"
    },
    "eventTime":"2020-12-09T01:24:09Z",
    "eventSource":"signin.amazonaws.com",
    "eventName":"CredentialVerification",
    "awsRegion":"us-east-1",
    "sourceIPAddress":"203.0.113.0",
    "userAgent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/87.0.4280.66 Safari/537.36",
    "requestParameters":null,
    "responseElements":null,
    "additionalEventData":{
        "AuthWorkflowID":"76d8a26d-ad9c-41a4-90c3-d607cdd7155c",
        "CredentialType":"PASSWORD"
    },
    "requestID":"12b57efa-0a92-4479-91a3-5b6641817c21",
    "eventID":"783b0c89-7142-4942-8b84-6ee0de1b992e",
    "readOnly":false,
    "eventType":"AwsServiceEvent",
    "managementEvent":true,
    "eventCategory":"Management",
    "recipientAccountId":"111122223333",
    "serviceEventDetails":{
        "CredentialVerification":"Success"
    }
}
```

Successful UserAuthentication (Password + MFA Registration Required)

```
{
    "eventVersion":"1.08",
    "userIdentity":{
        "type":"Unknown",
        "principalId":"111122223333",
        "arn":"",
        "accountId":"111122223333",
        "accessKeyId":"",
        "userName":"user1"
    },
    "eventTime":"2020-12-09T01:24:14Z",
    "eventSource":"signin.amazonaws.com",
    "eventName":"UserAuthentication",
    "awsRegion":"us-east-1",
    "sourceIPAddress":"203.0.113.0",
    "userAgent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/87.0.4280.66 Safari/537.36",
    "requestParameters":null,
    "responseElements":null,
    "additionalEventData":{
        "AuthWorkflowID":"76d8a26d-ad9c-41a4-90c3-d607cdd7155c",
```

```
        "LoginTo":"https://d-1234567890.awsapps.com/start/?state
\u003dQVlBQmVGQ3VqdHF5aW9CUDdrNXRTVTJUaWNnQUlnQUJBQk5FWVhSaFVHeGhibVZUZUZdGMFpWQmhjbUZ0QUFsUVpYSmxaM0pwwY
\u0026auth_code
\u003d11eZ80S_maUsZ7ABETjeQhyWfvIHYz52rgR28sYAKN1oEk2G07czrwzXvE9HLlN2K9De8LyBEV83SFeDQfrWpkwXfaBc2kNR1
FJyJqkoGrt_w6rm_MpAn0uyrVq8udY EgU3fhOL3QWvWiquYnDPMyPmmy_qkZgR9rz__BI\u0026wdc_csrf_token
\u003dJih9U62o5LQDtYLNqCK8a6xj0gJg5BRWq2tbl75y8vAmwZhAqrgrgbxXat2M646UZGp93krw7WYQdHIgi5OYI9QSckf4aovhQ
\u003dd-9067230c03\u0026region\u003dus-east-1",
        "CredentialType":"PASSWORD",
        "DeviceEnrollmentRequired":"true"
    },
    "requestID":"74d24604-a365-4237-8c4a-350795494b92",
    "eventID":"a15bf257-7f37-46c0-b67c-fea5fa6166be",
    "readOnly":false,
    "eventType":"AwsServiceEvent",
    "managementEvent":true,
    "eventCategory":"Management",
    "recipientAccountId":"111122223333",
    "serviceEventDetails":{
        "UserAuthentication":"Success"
    }
}
```

Failed sign-in when authenticating with only a password

The following sequence of events captures an example of a failed password only sign-in.

CredentialChallenge (Password)

```
{
    "eventVersion":"1.08",
    "userIdentity":{
        "type":"Unknown",
        "principalId":"111122223333",
        "arn":"",
        "accountId":"111122223333",
        "accessKeyId":"",
        "userName":"user1"
    },
    "eventTime":"2020-12-08T18:56:15Z",
    "eventSource":"signin.amazonaws.com",
    "eventName":"CredentialChallenge",
    "awsRegion":"us-east-1",
    "sourceIPAddress":"203.0.113.0",
    "userAgent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/87.0.4280.66 Safari/537.36",
    "requestParameters":null,
    "responseElements":null,
    "additionalEventData":{
        "AuthWorkflowID":"adbf67c4-8188-4e2b-8527-fe539e328fa7",
        "CredentialType":"PASSWORD"
    },
    "requestID":"f54848ea-b1aa-402f-bf0d-a54561a2ffcc",
    "eventID":"d96f1d6c-dbd9-4a0b-9a45-6a2b66078c78",
    "readOnly":false,
    "eventType":"AwsServiceEvent",
    "managementEvent":true,
    "eventCategory":"Management",
    "recipientAccountId":"111122223333",
    "serviceEventDetails":{
        "CredentialChallenge":"Success"
    }
}
```

Failed CredentialVerification (Password)

```
{
    "eventVersion":"1.08",
    "userIdentity":{
        "type":"Unknown",
        "principalId":"111122223333",
        "arn":"",
        "accountId":"111122223333",
        "accessKeyId":"",
        "userName":"user1"
    },
    "eventTime":"2020-12-08T18:56:21Z",
    "eventSource":"signin.amazonaws.com",
    "eventName":"CredentialVerification",
    "awsRegion":"us-east-1",
    "sourceIPAddress":"203.0.113.0",
    "userAgent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/87.0.4280.66 Safari/537.36",
    "requestParameters":null,
    "responseElements":null,
    "additionalEventData":{
        "AuthWorkflowID":"adbf67c4-8188-4e2b-8527-fe539e328fa7",
        "CredentialType":"PASSWORD"
    },
    "requestID":"04528c82-a678-4a1f-a56d-ea2c6445a72a",
    "eventID":"9160fe06-fc2a-474f-9b78-000ee067a09d",
    "readOnly":false,
    "eventType":"AwsServiceEvent",
    "managementEvent":true,
    "eventCategory":"Management",
    "recipientAccountId":"111122223333",
    "serviceEventDetails":{
        "CredentialVerification":"Failure"
    }
}
```

Amazon CloudWatch Events

IAM Identity Center can work with CloudWatch Events to raise events when administrator-specified actions occur in an organization. For example, because of the sensitivity of such actions, most administrators would want to be warned every time someone creates a new account in the organization or when an administrator of a member account attempts to leave the organization. You can configure CloudWatch Events rules that look for these actions and then send the generated events to administrator-defined targets. Targets can be an Amazon SNS topic that emails or text messages its subscribers. You could also create an AWS Lambda function that logs the details of the action for your later review.

To learn more about CloudWatch Events, including how to configure and enable it, see the *Amazon CloudWatch Events User Guide*.

Compliance validation for IAM Identity Center

Third-party auditors assess the security and compliance of AWS services such as AWS IAM Identity Center (successor to AWS Single Sign-On) as part of multiple AWS compliance programs.

To learn whether an AWS service is within the scope of specific compliance programs, see AWS services in Scope by Compliance Program and choose the compliance program that you are interested in. For general information, see AWS Compliance Programs.

You can download third-party audit reports using AWS Artifact. For more information, see Downloading Reports in AWS Artifact.

Your compliance responsibility when using AWS services 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 for deploying baseline environments on AWS that are security and compliance focused.
- Architecting for HIPAA Security and Compliance on Amazon Web Services – This whitepaper describes how companies can use AWS to create HIPAA-eligible applications.

  > **Note**
  > Not all AWS services are HIPAA eligible. For more information, see the HIPAA Eligible Services Reference.

- 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* – The AWS Config service 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 that helps you check your compliance with security industry standards and best practices.
- AWS Audit Manager – This AWS service helps you continuously audit your AWS usage to simplify how you manage risk and compliance with regulations and industry standards.

Supported compliance standards

IAM Identity Center has undergone auditing for the following standards and is eligible for use as part of solutions for which you need to obtain compliance certification.

AWS has expanded its Health Insurance Portability and Accountability Act (HIPAA) compliance program to include IAM Identity Center as a HIPAA eligible service.

AWS offers a HIPAA-focused whitepaper for customers who want to learn more about how they can use AWS services to process and store health information. For more information, see HIPAA compliance.

	The Information Security Registered Assessors Program (IRAP) enables Australian Government customers to ensure that appropriate compliance controls are in place and determine the appropriate responsibility model for addressing the requirements of the Australian Government Information Security Manual (ISM) produced by the Australian Cyber Security Centre (ACSC). For more information, see IRAP Resources.
	IAM Identity Center has an Attestation of Compliance for Payment Card Industry (PCI) Data Security Standard (DSS) version 3.2 at Service Provider Level 1. Customers who use AWS products and services to store, process, or transmit cardholder data can use the following identity sources in IAM Identity Center to manage their own PCI DSS compliance certification: • Active Directory • External identity provider The IAM Identity Center identity source is currently not compliant with PCI DSS. For more information about PCI DSS, including how to request a copy of the AWS PCI Compliance Package, see PCI DSS level 1.
	System & Organization Control (SOC) Reports are independent, third-party examination reports that demonstrate how IAM Identity Center achieves key compliance controls and objectives. These reports help you and your auditors to understand how controls support operations and compliance. There are three types of SOC reports: • AWS SOC 1 Report - Download with AWS Artifact • AWS SOC 2: Security, Availability, & Confidentiality Report - Download with AWS Artifact • AWS SOC 3: Security, Availability, & Confidentiality Report IAM Identity Center is in scope for AWS SOC 1, SOC 2, and SOC 3 reports. For more information, see SOC Compliance.

Resilience in IAM Identity Center

The AWS global infrastructure is built around AWS Regions and Availability Zones. AWS Regions provide multiple physically separated and isolated Availability Zones, which 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 Availability 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.

To learn more about AWS IAM Identity Center (successor to AWS Single Sign-On) resiliency, see Resiliency design and Regional behavior (p. 123).

Infrastructure security in IAM Identity Center

As a managed service, AWS IAM Identity Center (successor to AWS Single Sign-On) is protected by the AWS global network security procedures that are described in Best Practices for Security, Identity, & Compliance .

You use AWS published API calls to access IAM Identity Center through the network. Clients must support Transport Layer Security (TLS) 1.0 or later. We recommend TLS 1.2 or later. Clients must also support cipher suites with perfect forward secrecy (PFS) such as Ephemeral Diffie-Hellman (DHE) or Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). Most modern systems such as Java 7 and later support these modes.

Additionally, requests must be signed by using an access key ID and a secret access key that is associated with an IAM principal. Or you can use the AWS Security Token Service (AWS STS) to generate temporary security credentials to sign requests.

Tagging AWS IAM Identity Center (successor to AWS Single Sign-On) resources

A *tag* is a metadata label that you assign or that AWS assigns to an AWS resource. Each tag consists of a *key* and a *value*. For tags that you assign, you define the key and value. For example, you might define the key as `stage` and the value for one resource as `test`.

Tags help you do the following:

- Identify and organize your AWS resources. Many AWS services support tagging, so you can assign the same tag to resources from different services to indicate that the resources are related. For example, you can assign the same tag to a specific permission set in your IAM Identity Center instance.
- Track your AWS costs. You activate these tags on the AWS Billing and Cost Management dashboard. AWS uses the tags to categorize your costs and deliver a monthly cost allocation report to you. For more information, see Use cost allocation tags in the *AWS Billing User Guide*.
- Control access to your resources based on the tags that are assigned to the resources. You control access by specifying tag keys and values in the conditions for an AWS Identity and Access Management (IAM) policy. For example, you can allow an IAM user to update an IAM Identity Center permission set, but only if the IAM Identity Center permission set has an `owner` tag with a value of that user's name. For more information, see Controlling access using tags in the *IAM User Guide*.

Currently, you can apply tags to permission sets only. You can't apply tags to the corresponding roles that IAM Identity Center creates in AWS accounts. You can use the IAM Identity Center console, AWS CLI or the IAM Identity Center APIs to add, edit, or delete tags for a permission set.

For tips on using tags, see the AWS tagging strategies post on the *AWS Answers* blog.

The following sections provide more information about tags for IAM Identity Center.

Tag restrictions

The following basic restrictions apply to tags on IAM Identity Center resources:

- The maximum number of tags that you can assign to a resource is 50.
- The maximum key length is 128 Unicode characters.
- The maximum value length is 256 Unicode characters.
- Valid characters for a tag key and value are:

  a-z, A-Z, 0-9, space, and the following characters: _ . : / = + - and @
- Keys and values are case sensitive.
- Don't use `aws:` as a prefix for keys; it's reserved for AWS use

Manage tags by using the IAM Identity Center console

You can use the IAM Identity Center console to add, edit, and remove tags that are associated with your permission sets.

To manage tags for an IAM Identity Center console

1. Open the IAM Identity Center console.
2. Choose **Permission sets**.
3. Choose the name of the permission set that has the tags you want to manage.
4. On the **Permissions** tab, under **Tags**, do one of the following, and then proceed to the next step:

   a. If tags are already assigned for this permission set, choose **Edit tags**.

   b. If no tags are assigned to this permission set, choose **Add tags**.

5. For each new tag, type the values in the **Key** and **Value (optional)** columns. When you are finished, choose **Save changes.**

To remove a tag, choose the **X** in the **Remove** column next to the tag that you want to remove.

AWS CLI examples

The AWS CLI provides commands that you can use to manage the tags that you assign to your permission set.

Assigning tags

Use the following commands to assign tags to your permission set.

Example `tag-resource` Command for a permission set

Assign tags to a permission set by using `tag-resource` within the `sso` set of commands:

```
$ aws sso tag-resource \
> --instance-arn sso-instance-arn \
> --resource-arn sso-resource-arn \
> --tags Stage=Test
```

This command includes the following parameters:

- `instance-arn` – The Amazon Resource Name (ARN) of the IAM Identity Center instance under which the operation will be executed.
- `resource-arn` – The ARN of the resource with the tags to be listed.
- `tags` – The key-value pairs of the tags.

To assign multiple tags at once, specify them in a comma-separated list:

```
$ aws sso tag-resource \
> --instance-arn sso-instance-arn \
> --resource-arn sso-resource-arn \
```

```
> --tags Stage=Test,CostCenter=80432,Owner=SysEng
```

Viewing tags

Use the following commands to view the tags that you have assigned to your permission set.

Example `list-tags-for-resource` Command for a permission set

View the tags that are assigned to a permission set by using `list-tags-for-resource` within the `sso` set of commands:

```
$ aws sso list-tags-for-resource --resource-arn sso-resource-arn
```

Removing tags

Use the following commands to remove tags from a permission set.

Example `untag-resource` Command for a permission set

Remove tags from a permission set by using `untag-resource` within the `sso` set of commands:

```
$ aws sso untag-resource \
> --instance-arn sso-instance-arn \
> --resource-arn sso-resource-arn \
> --tag-keys Stage CostCenter Owner
```

For the `--tag-keys` parameter, specify one or more tag keys, and do not include the tag values.

Applying tags when you create a permission set

Use the following commands to assign tags at the moment you create a permission set.

Example `create-permission-set` Command with tags

When you create a permission set by using the `create-permission-set` command, you can specify tags with the `--tags` parameter:

```
$ aws sso create-permission-set \
> --instance-arn sso-instance-arn \
> --name permission=set-name \
> --tags Stage=Test,CostCenter=80432,Owner=SysEng
```

Manage tags using the IAM Identity Center API

You can use the following actions in the IAM Identity Center API to manage the tags for your permission set.

API actions for IAM Identity Center instance tags

Use the following API actions to assign, view, and remove tags for a permission set.

- TagResource

- ListTagsForResource
- UntagResource
- CreatePermissionSet

Integrating AWS CLI with IAM Identity Center

AWS Command Line Interface (CLI) version 2 integration with IAM Identity Center simplifies the sign-in process. Developers can sign in directly to the AWS CLI using the same Active Directory or IAM Identity Center credentials that they normally use to sign in to IAM Identity Center, and access their assigned accounts and roles. For example, after an administrator configures IAM Identity Center to use Active Directory for authentication, a developer can sign into the AWS CLI directly using their Active Directory credentials.

AWS CLI integration with IAM Identity Center offers the following benefits:

- Enterprises can enable their developers to sign in using credentials from IAM Identity Center or Active Directory by connecting IAM Identity Center to their Active Directory using AWS Directory Service.
- Developers can sign in from the CLI for faster access.
- Developers can list and switch between accounts and roles to which they have assigned access.
- Developers can generate and save named role profiles in their CLI configuration automatically and reference them in the CLI to run commands in desired accounts and roles.
- The CLI manages short-term credentials automatically so developers can start in and stay in the CLI securely without interruption, and run long running scripts.

How to integrate AWS CLI with IAM Identity Center

To use the AWS CLI integration with IAM Identity Center, you need to download, install, and configure AWS Command Line Interface version 2. For detailed steps on how to download and integrate the AWS CLI with IAM Identity Center, see Configuring the AWS CLI to use IAM Identity Center in the *AWS Command Line Interface User Guide*.

AWS IAM Identity Center (successor to AWS Single Sign-On) Region availability

IAM Identity Center is available in several commonly used AWS Regions. This availability makes it easier for you to configure user access to multiple AWS accounts and business applications. When your users sign in to the AWS access portal, they can select the AWS account to which they have permissions, and then access the AWS Management Console. For a full list of the Regions that IAM Identity Center supports, see IAM Identity Center endpoints and quotas.

> **Note**
> When you create a user or when a user verifies their email while operating in the AWS GovCloud (US-East) Region, AWS GovCloud (US-East) makes cross-region API calls to send emails to AWS GovCloud (US-West). In these cross-region calls, user attributes include email, directory ID, user ID, user name, first and last name, and account in AWS Organizations.

IAM Identity Center Region data

When you first enable IAM Identity Center, all the data that you configure in IAM Identity Center is stored in the Region where you configured it. This data includes directory configurations, permission sets, application instances, and user assignments to AWS account applications. If you are using the IAM Identity Center identity store, all users and groups that you create in IAM Identity Center are also stored in the same Region. We recommend that you install IAM Identity Center in a Region that you intend to keep available for users, not a Region that you might need to disable.

AWS Organizations supports only one AWS Region at a time. To enable IAM Identity Center in a different Region, you must first delete your current IAM Identity Center configuration. Switching to a different Region also changes the URL for the AWS access portal, and you must reconfigure all permission sets and assignments.

Managing IAM Identity Center in Regions that must be manually enabled

Most AWS Regions are enabled for operations in all AWS services by default. Those Regions are automatically activated for use with IAM Identity Center. Some Regions, such as the Europe (Milan) Region, must be manually enabled. When you enable IAM Identity Center for a management account in a manually enabled Region, the following IAM Identity Center metadata for any member accounts is stored in the Region.

- Account ID
- Account name
- Account email
- Amazon Resource Names (ARNs) of the IAM roles that IAM Identity Center creates in the member account

If you disable a Region in which IAM Identity Center is installed, IAM Identity Center is also disabled. After IAM Identity Center is disabled in a Region, users in that Region won't have single sign-on access to AWS accounts and applications. AWS retains the data in your IAM Identity Center configuration for at least 10 days. If you re-enable IAM Identity Center within this time frame, your IAM Identity Center configuration data will still be available in the Region.

To re-enable IAM Identity Center in Regions that must be manually enabled, you must re-enable the Region. Because IAM Identity Center must reprocess all paused events again, re-enabling IAM Identity Center might take some time.

> **Note**
> IAM Identity Center can manage access only to the AWS accounts that are enabled for use in a Region. To manage access across all accounts in your organization, enable IAM Identity Center in the management account in a Region that is automatically activated for use with IAM Identity Center.

For more information about enabling and disabling AWS Regions, see Managing AWS Regions in the *AWS General Reference*.

Delete your IAM Identity Center configuration

When an IAM Identity Center configuration is deleted, all the data in that configuration is deleted and can't be recovered. The following table describes what data is deleted based on the directory type that is currently configured in IAM Identity Center.

What data gets deleted	Connected directory (AWS Managed Microsoft AD or AD Connector)	IAM Identity Center identity store
All permission sets you have configured for AWS accounts	X	X
All applications you have configured in IAM Identity Center	X	X
All user assignments you have configured for AWS accounts and applications	X	X
All users and groups in the directory or store	N/A	X

Use the following procedure when you need to delete your current IAM Identity Center configuration.

To delete your IAM Identity Center configuration

1. Open the IAM Identity Center console.
2. In the left navigation pane, choose **Settings**.
3. On the **Settings** page, choose **Delete**.
4. In the **Delete IAM Identity Center configuration** dialog, select each of the check boxes to acknowledge you understand that your data that will be deleted. Type **DELETE** in the text box, and then choose **Confirm**.

AWS IAM Identity Center (successor to AWS Single Sign-On) quotas

The following tables describe quotas within IAM Identity Center. To increase a quota, see Requesting a quota increase.

> **Note**
> We recommend using the AWS CLI and APIs if you have more than 50,000 users, 10,000 groups, or 500 permission sets. For more information about the CLI, see Integrating AWS CLI with IAM Identity Center (p. 181). For more information about APIs, see Welcome to the IAM Identity Center API Reference.

Application quotas

Resource	Default quota	Can be increased
File size of service provider SAML certificates (in PEM format)	2 KB	No
File size limit of the IdP certificate uploaded to SSO	2500 (UTF-8) characters	No

AWS account quotas

Resource	Default quota	Can be increased
Number of permission sets allowed in IAM Identity Center	2000	Yes
Number of permission sets allowed per AWS account	50	Yes
Number of inline policies per permission set	1	No
Number of AWS managed and customer managed policies per permission set	20^1	No
Maximum size of inline policy per permission set	10,240 bytes	No
Number of IAM roles (permission sets) in the AWS account that can be updated at a time	1	No

[1]AWS Identity and Access Management (IAM) sets a quota of 10 managed policies per role. To take advantage of this quota, request an increase to the IAM quota *Managed policies attached to an IAM role* in the Service Quotas console for each AWS account where you want to deploy the permission set.

> **Note**
> Permission sets (p. 16) are provisioned in AWS accounts as IAM roles, or use existing IAM roles in AWS accounts, and therefore follow IAM quotas. For more information about quotas that are associated with IAM roles, see IAM and STS quotas.

Active Directory quotas

Resource	Default quota	Can be increased
Number of connected directories that you can have at a time	1	No

IAM Identity Center identity store quotas

Resource	Default quota	Can be increased
Number of users supported in IAM Identity Center	100000	Yes
Number of groups supported in IAM Identity Center	100000	No

IAM Identity Center throttle limits

Resource	Default quota
IAM Identity Center APIs	IAM Identity Center APIs have a collective throttle maximum of 20 transactions per second (TPS). The CreateAccountAssignment has a maximum rate of 10 outstanding async calls. These quotas cannot be changed.

Additional quotas

Resource	Default quota	Can be increased
Total number of AWS accounts or applications that can be configured *	3000	Yes

* Up to 3000 AWS accounts or applications (total combined) are supported. For example, you might configure 2750 accounts and 250 applications, resulting in a total of 3000 accounts and applications.

AWS IAM Identity Center (successor
to AWS Single Sign-On) User Guide
Issues regarding contents of SAML
assertions created by IAM Identity Center

Troubleshooting IAM Identity Center issues

The following can help you troubleshoot some common issues you might encounter while setting up or using the IAM Identity Center console.

Issues regarding contents of SAML assertions created by IAM Identity Center

IAM Identity Center provides a web-based debug experience for the SAML assertions created and sent by IAM Identity Center, including attributes within these assertions, when accessing AWS accounts and SAML applications from the AWS access portal. To see the details of a SAML assertion that IAM Identity Center generates, use the following steps.

1. Sign in to the AWS access portal.
2. While you are signed into the portal, hold the **Shift** key down, choose the application tile, and then release the **Shift** key.
3. Examine the information on the page titled **You are now in administrator mode**. To keep this information for future reference, choose **Copy XML**, and paste the contents elsewhere.
4. Choose **Send to <application>** to continue. This option sends the assertion to the service provider.

> **Note**
> Some browser configurations and operating systems may not support this procedure. This procedure has been tested on Windows 10 using Firefox, Chrome, and Edge browsers.

Specific users fail to synchronize into IAM Identity Center from an external SCIM provider

If SCIM synchronization succeeds for a subset of users configured in your IdP for provisioning into IAM Identity Center but fails for others, you might see an error similar to `'Request is unparsable, syntactically incorrect, or violates schema'` from your identity provider. You may also see detailed provisioning failure messages in AWS CloudTrail.

This issue often indicates that the user in your IdP is configured in a way that IAM Identity Center does not support. Full details of the IAM Identity Center SCIM implementation, including the specifications of required, optional, and forbidden parameters and operations for user objects, can be found in the IAM Identity Center SCIM Implementation Developer Guide. The *SCIM Developer Guide* should be considered authoritative for information around SCIM requirements. However, the following are a couple of common reasons for this error:

1. The user object in the IdP lacks a first (given) name, a last (family) name, and/or a display name.

    *   **Solution:** Add a first (given), last (family), and display name for the user object. In addition, ensure that the SCIM provisioning mappings for user objects at your IdP are configured to send nonempty values for all of these attributes.

2. More than one value for a single attribute is being sent for the user (also known as "multi-value attributes"). For example, the user may have both a work and a home phone number specified in the IdP, or multiple emails or physical addresses, and your IdP is configured to try to synchronize multiple or all values for that attribute.

- **Solution options:**

  i. Update your SCIM provisioning mappings for user objects at your IdP to send only a single value for a given attribute. For example, configure a mapping that sends only the work phone number for each user.

  ii. If the additional attributes can safely be removed from the user object at the IdP, you can remove the additional values, leaving either one or zero values set for that attribute for the user.

  iii. If the attribute is not needed for any actions in AWS, remove the mapping for that attribute from the SCIM provisioning mappings for user objects at your IdP.

3. Your IdP is trying to match users in the target (IAM Identity Center, in this case) based on multiple attributes. Since user names are guaranteed unique within a given IAM Identity Center instance, you only need to specify `username` as the attribute used for matching.

- **Solution:** Ensure that your SCIM configuration in your IdP is using only a single attribute for matching with users in IAM Identity Center. For example, mapping `username` or `userPrincipalName` in the IdP to the `userName` attribute in SCIM for provisioning to IAM Identity Center will be correct and sufficient for most implementations.

Users can't sign in when their user name is in UPN format

Users might not be able to sign in to the AWS access portal based on the format they use to enter in their user name on the sign in page. For the most part, users can sign in to the user portal using either their plain user name, their down-level logon name (DOMAIN\UserName) or their UPN logon name (UserName@Corp.Example.com). The exception to this is when IAM Identity Center is using a connected directory that has been enabled with MFA and the verification mode has been set to either **Context-aware** or **Always-on**. In this scenario, users must sign in with their down-level logon name (DOMAIN \UserName). For more information, see Multi-factor authentication (p. 82). For general information about user name formats used to sign in to Active Directory, see User Name Formats on the Microsoft documentation website.

I get a 'Cannot perform the operation on the protected role' error when modifying an IAM role

When reviewing IAM Roles in an account, you may notice role names beginning with 'AWSReservedSSO_'. These are the roles which the IAM Identity Center service has created in the account, and they came from assigning a permission set to the account. Attempting to modify these roles from within the IAM console will result in the following error:

```
'Cannot perform the operation on the protected role 'AWSReservedSSO_RoleName_Here' - this
 role is only modifiable by AWS'
```

These roles can only be modified from the IAM Identity Center Administrator console, which is in the management account of AWS Organizations. Once modified, you can then push the changes down to the AWS accounts that it is assigned to.

Directory users cannot reset their password

When a directory user resets their password using the **Forgot Password?** option during sign-in of the AWS access portal, their new password must adhere to the default password policy as described in Password requirements when managing identities in IAM Identity Center (p. 32).

If a user enters a password that adheres to the policy and then receives the error `We couldn't update your password`, check to see if AWS CloudTrail recorded the failure. This can be done by searching in the Event History console of CloudTrail using the following filter:

```
"UpdatePassword"
```

If the message states the following, then you may need to contact support:

```
"errorCode": "InternalFailure",
     "errorMessage": "An unknown error occurred"
```

Another possible cause of this issue is in the naming convention that was applied to the user name value. Naming conventions must follow specific patterns such as 'surname.givenName'. However, some user names can be quite long, or contain special characters, and this can cause characters to be dropped in the API call, thereby resulting in an error. You may want to attempt a password reset with a test user in the same manner to verify if this is the case.

If the issue persists, contact the AWS Support Center.

My user is referenced in a permission set but can't access the assigned accounts or applications

This issue can occur if you're using System for Cross-domain Identity Management (SCIM) for Automatic Provisioning with an external identity provider. Specifically, when a user, or the group the user was a member of, is deleted then re-created using the same user name (for users) or name (for groups) in the identity provider, a new unique internal identifier is created for the new user or group in IAM Identity Center. However, IAM Identity Center still has a reference to the old identifier in its permission database, such that the name of the user or group still appears in the UI, but access fails. This is because the underlying user or group ID to which the UI refers no longer exists.

To restore AWS account access in this case, you can remove access for the old user or group from the AWS account(s) where it was originally assigned, and then reassign access back to the user or group. This updates the permission set with the correct identifier for the new user or group. Similarly, to restore application access, you can remove access for the user or group from the assigned users list for that application, then add the user or group back again.

You can also check to see if AWS CloudTrail recorded the failure by searching your CloudTrail logs for SCIM synchronization events that reference the name of the user or group in question.

I cannot get my cloud application configured correctly

Each service provider of a preintegrated cloud application in IAM Identity Center has its own detailed instruction manual. You can access the manual from the **Configuration** tab for that application in the IAM Identity Center console.

AWS IAM Identity Center (successor
to AWS Single Sign-On) User Guide
Error 'An unexpected error has occurred' when a user
tries to sign in using an external identity provider

If the problem is related to setting up the trust between the service provider's application and IAM Identity Center, make sure to check the instruction manual for troubleshooting steps.

Error 'An unexpected error has occurred' when a user tries to sign in using an external identity provider

This error may occur for multiple reasons, but one common reason is a mis-match between the user information carried in the SAML request, and the information for the user in IAM Identity Center.

In order for an IAM Identity Center user to sign in successfully when using an external IdP as the identity source, the following must be true:

- The SAML nameID format (configured at your identity provider) must be 'email'

- The nameID value must be a properly (RFC2822)-formatted string (user@domain.com)

- The nameID value must exactly match the user name of an existing user in IAM Identity Center (it doesn't matter if the email address in IAM Identity Center matches or not – the inbound match is based on username)

- The IAM Identity Center implementation of SAML 2.0 federation supports only 1 assertion in the SAML response between the identity provider and IAM Identity Center. It does not support encrypted SAML assertions.

- The following statements apply if Attributes for access control (p. 106) is enabled in your IAM Identity Center account:

  - The number of attributes mapped in the SAML request must be 50 or less.

  - The SAML request must not contain multi-valued attributes.

  - The SAML request must not contain multiple attributes with the same name.

  - The attribute must not contain structured XML as the value.

  - The Name format must be a SAML specified format, not generic format.

  Note
  IAM Identity Center does not perform "just in time" creation of users or groups for new users or groups via SAML federation. This means that the user must be pre-created in IAM Identity Center, either manually or via automatic provisioning, in order to sign in to IAM Identity Center.

This error can also occur when the Assertion Consumer Service (ACS) endpoint configured in your identity provider does not match the ACS URL provided by your IAM Identity Center instance. Ensure that these two values match exactly.

Additionally, you can troubleshoot external identity provider sign-in failures further by going to AWS CloudTrail and filtering on the event name **ExternalIdPDirectoryLogin**.

Error 'Attributes for access control failed to enable'

This error may occur if the user enabling ABAC does not have the `iam:UpdateAssumeRolePolicy` permissions required to enable Attributes for access control (p. 106).

AWS IAM Identity Center (successor
to AWS Single Sign-On) User Guide
I get a 'Browser not supported' message
when I attempt to register a device for MFA

I get a 'Browser not supported' message when I attempt to register a device for MFA

WebAuthn is currently supported in Google Chrome, Mozilla Firefox, Microsoft Edge and Apple Safari web browsers, as well as Windows 10 and Android platforms. Some components of WebAuthn support may be varied, such as platform authenticator support across macOS and iOS browsers. If users attempt to register WebAuthn devices on an unsupported browser or platform, they will see certain options greyed out that are not supported, or they will receive an error that all supported methods are not supported. In these cases, please refer to FIDO2: Web Authentication (WebAuthn) for more information about browser/platform support. For more information about WebAuthn in IAM Identity Center, see Security keys and built-in authenticators (p. 85).

Active Directory "Domain Users" group does not properly sync into IAM Identity Center

The Active Directory Domain Users group is the default "primary group" for AD user objects. Active Directory primary groups and their memberships cannot be read by IAM Identity Center. When assigning access to IAM Identity Center resources or applications, use groups other than the Domain Users group (or other groups assigned as primary groups) to have group membership properly reflected in the IAM Identity Center identity store.

Invalid MFA credentials error

This error can occur when a user attempts to sign in to IAM Identity Center using an account from an external identity provider (for example, Okta or Azure AD) before their user account has been fully provisioned to IAM Identity Center using the SCIM protocol. Once the user account has been provisioned to IAM Identity Center, this issue should be resolved. Confirm that the user account has been provisioned to IAM Identity Center and, if not, check the provisioning logs in the external identity provider.

I get a 'An unexpected error has occurred' message when I attempt to register or sign in using an authenticator app

Time-based one-time password (TOTP) systems, such as those used by IAM Identity Center in combination with code-based authenticator apps, rely on time synchronization between the client and the server. Ensure that the device where your authenticator app is installed is correctly synchronized to a reliable time source, or manually set the time on your device to match a reliable source, such as NIST (https://www.time.gov/) or other local/regional equivalents.

My users are not receiving emails from IAM Identity Center

All emails sent by the IAM Identity Center service will come from either the address no-reply@signin.aws or no-reply@login.awsapps.com. Your mail system must be configured so that it accepts emails from these sender email addresses and does not handle them as junk or spam.

Error: You can't delete/modify/remove/assign access to permission sets provisioned in the management account

This message indicates that the Delegated administration (p. 91) feature has been enabled and that the operation you previously attempted can only be successfully performed by someone who has management account privileges in AWS Organizations. To resolve this issue, sign in as a user who has these privileges and try performing the task again or assign this task to someone who has the correct permissions. For more information, see What tasks can be performed in the delegated administrator account (p. 92).

Document history

The following table describes important additions to the AWS IAM Identity Center (successor to AWS Single Sign-On) documentation. We also update the documentation frequently to address the feedback that you send us.

- **Latest major documentation update:** September 23, 2022

Change	Description	Date
Enhanced guidance for getting started with IAM Identity Center	Added new content for getting started with IAM Identity Center and creating an administrative user	September 23, 2022
Updated users and groups in the Identity Center API Reference	This update includes references to the new Create, Update and Delete APIs in the Identity Center API Reference Guide.	August 31, 2022
AWS Single Sign-On (AWS SSO) renamed to AWS IAM Identity Center	AWS introduces AWS IAM Identity Center (successor to AWS Single Sign-On). IAM Identity Center expands the capabilities of AWS Identity and Access Management (IAM) to help you centrally manage account and access to cloud applications for your workforce users. IAM Identity Center features include application assignments, multi-account permissions, and an AWS access portal.	July 26, 2022
Support for permissions boundaries and customer managed policies in permission sets	Added content for using AWS managed and customer managed AWS Identity and Access Management (IAM) policies with permission sets.	July 14, 2022
Support for manually enabled AWS Regions	Added content for using IAM Identity Center in manually enabled Regions.	June 15, 2022
Updates for AWS managed policies	Updated permissions for the `AWSSSOServiceRolePolicy` AWS managed policy.	May 11, 2022
Support for delegated administration	Added content for the delegated administration feature.	May 11, 2022
Updates for AWS managed policies	Updated permissions for the `AWSSSOMasterAccountAdministrator`, `AWSSSOMemberAccountAdministrator`,	April 28, 2022

	and `AWSSSOReadOnly` AWS managed policies.	
Support for configurable AD sync	Added content for the configurable AD sync feature.	April 14, 2022
New AWS managed policy topic	Added details for the `AWSSSOMasterAccountAdministrator` AWS managed policy.	August 4, 2021
Updates for quotas	Adjustments to quota tables.	December 21, 2020
New example policies	Added new customer managed policy examples and updates to the permissions required section.	December 21, 2020
Support for attribute-based access control (ABAC)	Added content for ABAC feature.	November 24, 2020
Support for MFA forced enrollment	Updates to require users to enroll an MFA device at sign-in.	November 23, 2020
Support for WebAuthn	Added content for new WebAuthn feature.	November 20, 2020
Support for Ping Identity	Added content to integrate with Ping Identity products as a supported external identity provider.	October 26, 2020
Support for OneLogin	Added content to integrate with OneLogin as a supported external identity provider.	July 31, 2020
Support for Okta	Added content to integrate with Okta as a supported external identity provider.	May 28, 2020
Support for external identity providers	Changed references from directory to identity source, added content to support external identity providers.	November 26, 2019
New MFA settings	Removed two-step verification topic and added new MFA topic in its place.	October 24, 2019
New setting to add two-step verification	Added content on how to enable two-step verification for users.	January 16, 2019
Support for session duration on AWS accounts	Added content on how to set the session duration for an AWS account.	October 30, 2018
New option to use Identity Center directory	Added content for choosing either Identity Center directory or connecting to an existing directory in Active Directory.	October 17, 2018

Support for relay state and session duration on applications	Added content about relay state and session duration for cloud applications.	October 10, 2018
Additional support for new cloud applications	Added 4me, BambooHR, Bonusly, Citrix ShareFile, ClickTime, Convo, Deputy, Deskpro, Dome9, DruvaInSync, Egnyte, Engagedly, Expensify, Freshdesk, IdeaScale, Igloo, Jitbit, Kudos, LiquidFiles, Lucidchart, PurelyHR, Samanage, ScreenSteps, Sli.do, SmartSheet, Syncplicity, TalentLMS, Trello, UserVoice, Zoho, OpsGenie, DigiCert, WeekDone, ProdPad, and UserEcho to the application catalog.	August 3, 2018
Support for multi-account access to management accounts	Added content about how to delegate multi-account access to users in a management account.	July 9, 2018
Support for new cloud applications	Added DocuSign, Keeper Security, and SugarCRM to the application catalog.	March 16, 2018
Get temporary credentials for CLI access	Added information about how to get temporary credentials to run AWS CLI commands.	February 22, 2018
New guide	This is the first release of the IAM Identity Center User Guide.	December 7, 2017

AWS glossary

For the latest AWS terminology, see the AWS glossary in the *AWS General Reference*.