



Integrating Azure AD with AWS

Integrating Azure Active Directory (Azure AD) with AWS allows you to enable single sign-on (SSO) and federated authentication for your AWS accounts. This integration provides a seamless login experience for users and centralized management of access policies.



Use Cases for Integrating Azure AD with AWS

- Centralized User Management:** Organizations can manage user access to AWS resources through Azure AD, leveraging existing user identities and groups. This simplifies user provisioning and de-provisioning, ensuring that only authorized users have access to AWS services.
- Single Sign-On (SSO):** Users can log in to AWS using their Azure AD credentials. This provides a seamless login experience, reducing the need to remember multiple passwords and improving user productivity.
- Federated Authentication:** Integrating Azure AD with AWS allows for federated authentication, enabling users to authenticate against Azure AD and obtain temporary security credentials to access AWS resources.
- Enhanced Security:** By leveraging Azure AD's security features such as Multi-Factor Authentication (MFA), Conditional Access Policies, and Identity Protection, organizations can enhance the security of their AWS environments.
- Compliance and Governance:** Centralizing access management through Azure AD helps organizations maintain compliance with various regulatory requirements by providing consistent access control and detailed audit logs.
- Role-Based Access Control (RBAC):** Azure AD groups can be mapped to AWS IAM roles, allowing for granular access control. This ensures users have the appropriate level of access based on their role within the organization.
- Improved User Experience:** By providing a unified login experience across multiple cloud services, organizations can improve the user experience and reduce login-related issues.



Benefits of Integrating Azure AD with AWS

- Streamlined Access Management:** Centralizing access management through Azure AD simplifies the administration of user identities and access permissions, reducing the administrative overhead associated with managing multiple identity providers.
- Enhanced Security Posture:** Leveraging Azure AD's advanced security features, such as MFA and Conditional Access, helps to protect AWS resources from unauthorized access and potential security threats.
- Operational Efficiency:** Integrating Azure AD with AWS allows for the automation of user provisioning and de-provisioning, ensuring that access is granted and revoked in a timely manner.

timely manner. This reduces the risk of orphaned accounts and potential security vulnerabilities.

4. **Cost Savings:** By consolidating identity management and access control into a single platform, organizations can reduce costs associated with maintaining multiple identity solutions and streamline their IT operations.
5. **Consistent User Experience:** Providing a consistent login experience across different cloud platforms improves user satisfaction and reduces the likelihood of login-related issues.
6. **Compliance and Auditability:** Centralizing access management through Azure AD ensures that access controls are consistently applied and auditable, helping organizations meet compliance requirements and maintain detailed records of user activity.
7. **Scalability:** Azure AD is designed to handle large-scale enterprise environments, making it suitable for organizations of all sizes. This scalability ensures that the identity and access management solution can grow with the organization's needs.
8. **Simplified Onboarding and Offboarding:** With Azure AD integration, onboarding new employees and offboarding departing employees becomes more efficient, as access to AWS resources can be managed through existing identity management workflows.

What are we doing in this lab?

This process outlines the steps to set up Single Sign-On (SSO) and automatic user provisioning between Azure Active Directory (Azure AD) and AWS IAM Identity Center. Here's a summary of what you're doing and the end goal:

1. **Configure SSO Integration:** You're setting up SSO by integrating Azure AD with AWS IAM Identity Center. This involves enabling IAM Identity Center in AWS, configuring SAML-based authentication, and uploading necessary metadata files from both Azure and AWS.
2. **Set Up Automatic Provisioning:** You're configuring automatic user provisioning from Azure AD to AWS. This includes providing the SCIM endpoint and access token in Azure to automate the creation and management of AWS users based on Azure AD.
3. **Add and Manage Users:** You add users to Azure AD and ensure they are automatically provisioned into AWS IAM Identity Center. You then assign permissions to these users within AWS to grant appropriate access.
4. **Verify and Test:** Finally, you test the integration by logging in to the AWS Access Portal using Azure AD credentials to confirm that everything is working correctly and that users have the expected access.

End Goal: In simple terms, the end goal is to make it easier for users to log in to AWS using their existing Azure AD credentials, and to automatically manage their access and permissions in AWS based on their Azure AD account. This means users don't need to remember a new set

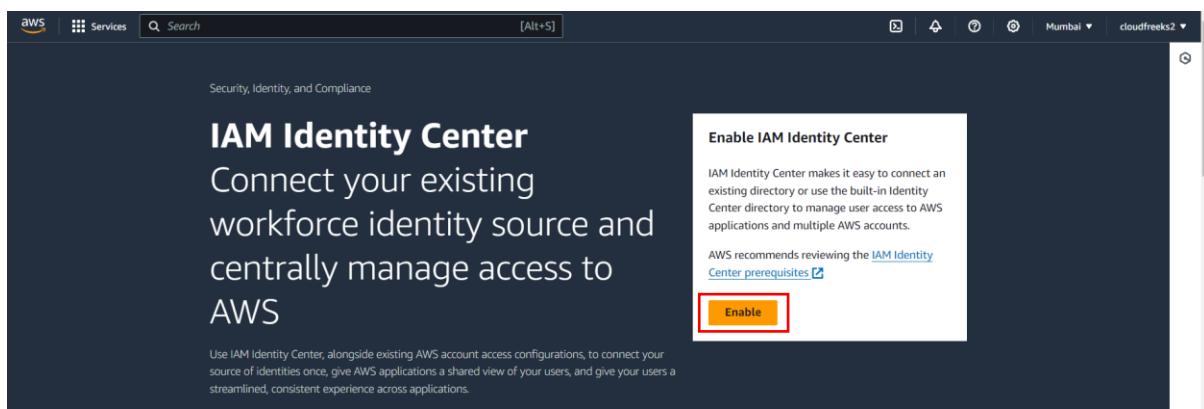
of login details, and their access rights in AWS are automatically kept up-to-date based on what's set in Azure AD.

😊 To begin with the Lab:

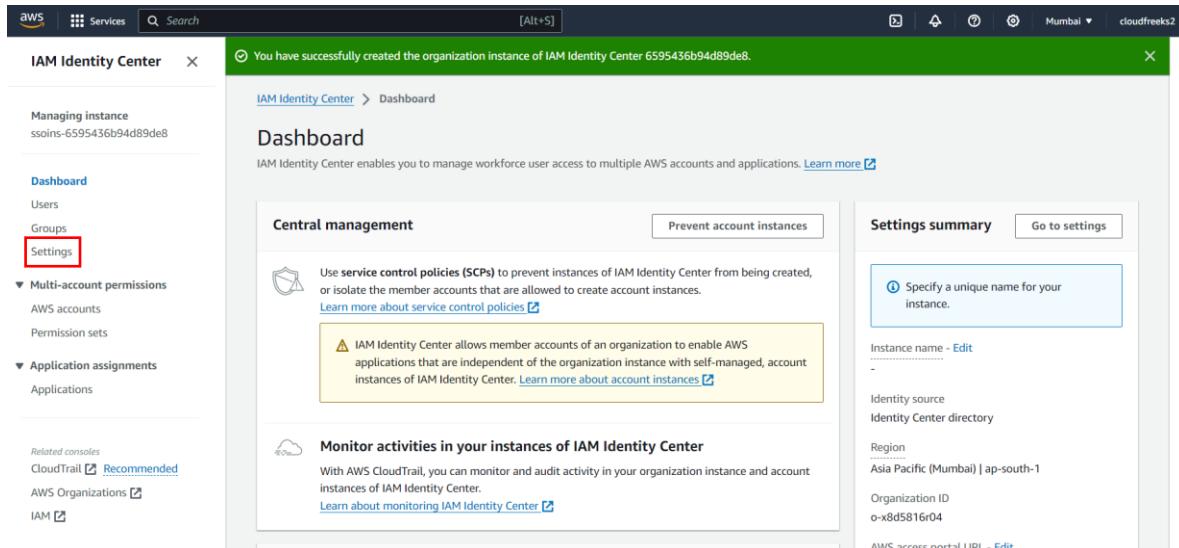
1. Login to AWS Console and navigate to IAM Identity Centre. Choose the service accordingly.



2. Now from its dashboard click on Enable.



3. Once it is enabled you need to click on settings, highlighted below.



4. Then in the settings you need to scroll down and you will see the Identity source. So, click on Actions and choose change identity source.

The screenshot shows the 'Identity source' configuration page in the AWS IAM Identity Center. At the top, there are tabs for 'Identity source', 'Authentication', 'Management', and 'Tags'. The 'Identity source' tab is selected. On the left, there's a sidebar with 'Identity source', 'Authentication method', 'AWS access portal URL', and 'Issuer URL'. On the right, there are fields for 'Provisioning method' (set to 'Direct'), 'Identity store ID' (set to 'd-9f670ee52b'), and a 'Actions' button which is highlighted with a red box. Below the 'Actions' button are two options: 'Customize AWS access portal URL' and 'Change identity source'.

5. Now you need to choose External identity provider and click on next.

The screenshot shows the 'Choose identity source' step in the IAM Identity Center wizard. It has three steps: Step 1 (Choose identity source), Step 2 (Configure external identity provider), and Step 3 (Confirm change). The current step is Step 1. The 'External identity provider' option is selected and highlighted with a blue circle. The other two options, 'Identity Center directory' and 'Active Directory', are not selected. A 'Learn more' link is also present. At the bottom, there are 'Cancel' and 'Next' buttons.

6. Now you have to download the metadata file.

The screenshot shows the 'Configure external identity provider' step in the IAM Identity Center wizard. It has three steps: Step 1 (Choose identity source), Step 2 (Configure external identity provider), and Step 3 (Confirm change). The current step is Step 2. Under 'Service provider metadata', there is a 'Download metadata file' button which is highlighted with a red box. Below it, there are fields for 'AWS access portal sign-in URL' (set to 'https://d-9f670ee52b.awsapps.com/start'), 'IAM Identity Center Assertion Consumer Service (ACS) URL' (set to 'https://ap-south-1.sigin.aws.amazon.com/platform/saml/acs/c45a599c-d538-466c-ba35-7e5d89c07d6a'), and 'IAM Identity Center issuer URL' (set to 'https://ap-south-1.sigin.aws.amazon.com/platform/saml/d-9f670ee52b').

7. Now you are going to open your Azure Portal and here we are going to configure the related parameters required. In your Portal you need to go to Microsoft Entra ID.
8. So, here you need to go to Enterprise Application.

The screenshot shows the Microsoft Azure Cloudfreaks Overview page. On the left, there's a sidebar with various navigation options. The 'Enterprise applications' option is highlighted with a red box. The main content area displays basic information about the tenant, including the name 'Cloudfreaks', tenant ID '429a3ec3-3806-45fc-b987-be94de140efd', primary domain 'Cloudfreaks451.onmicrosoft.com', and license 'Microsoft Entra ID P2'. There are also sections for users, groups, applications, devices, and alerts.

9. Then choose to add a new application.

The screenshot shows the Microsoft Azure Enterprise applications | All applications page. The 'All applications' section is selected in the sidebar. The main area shows a search bar and a table with columns for Name, Object ID, Application ID, Homepage URL, Created on, Certificate Expiry, and Active Certificate. A message indicates 'No results'.

10. Now you need to search for AWS IAM and choose IAM Identity Centre as shown below, then click on create.

The screenshot shows the Microsoft Azure Browse Microsoft Entra Gallery page. A search bar at the top has 'AWS IAM' typed into it. Below the search bar, there are filters for Single Sign-on, User Account Management, and Categories, all set to 'All'. The results section shows one item: 'AWS IAM Identity Center (successor to AWS Single Sign-On)'. This item is highlighted with a red box.

11. Here now you can see the AWS IAM Identity Centre as your enterprise application in Azure.

12. Now you need to go inside of it.

The screenshot shows the Microsoft Entra admin center interface. In the top navigation bar, the path is Home > Cloudfreaks | Enterprise applications > Enterprise applications. The main title is Enterprise applications | All applications. Below the title, there's a search bar and filter options. The left sidebar has sections like Overview, Manage (with All applications selected), Security, Activity, and Troubleshooting + Support. The main content area displays a table of applications. The first row in the table is for the AWS IAM Identity Center, which matches the search criteria 'Enterprise Applications'. The table columns include Name, Object ID, Application ID, Homepage URL, Created on, Certificate Expiry, and Active Certificate. The table shows 1 application found.

13. Here we need to select Single sign-on.

The screenshot shows the AWS IAM Identity Center Overview page. The top navigation bar includes Home, Cloudfreaks, and Enterprise applications. The main title is AWS IAM Identity Center (successor to AWS Single Sign-On) | Overview. On the left, a navigation menu lists Diagnose and solve problems, Manage (Properties, Owners, Roles and administrators, Users and groups, Single sign-on, Provisioning, Self-service, Custom security attributes), and Security/Activity. The 'Single sign-on' item is highlighted with a red box. The right side of the screen shows the Properties section with fields for Name (AWS IAM Identity Center (s...)), Application ID (fbcc05c6-6cc3-443c-866e-d...), and Object ID (ed92ef7d-060b-45a9-b715-...). Below this is the Getting Started section, which includes two cards: '1. Assign users and groups' and '2. Set up single sign on'.

14. Now you are going to choose SAML (Security Assertion Markup Language).

Select a single sign-on method [Help me decide](#)

The screenshot displays three cards:

- Disabled**: Single sign-on is not enabled. The user won't be able to launch the app from My Apps.
- SAML**: Rich and secure authentication to applications using the SAML (Security Assertion Markup Language) protocol. This card is highlighted with a red border.
- Linked**: Link to an application in My Apps and/or Office 365 application launcher.

15. Here we are going to click on Upload metadata file.

AWS IAM Identity Center (successor to AWS Single Sign-On) | SAML-based Sign-on ...

Enterprise Application

The screenshot shows the 'Single sign-on' configuration page for an enterprise application. The left sidebar includes options like Overview, Deployment Plan, Diagnose and solve problems, Manage (Properties, Owners, Roles and administrators, Users and groups), Single sign-on (selected), Provisioning, Self-service, Custom security attributes, and Security.

The main content area is titled 'Set up Single Sign-On with SAML'. It provides instructions for an SSO implementation based on federation protocols and links to a configuration guide. Step 1, 'Basic SAML Configuration', is shown with fields for Identifier (Entity ID), Reply URL (Assertion Consumer Service URL), Sign on URL, Relay State (Optional), and Logout Url (Optional). Step 2, 'Attributes & Claims', is partially visible below. A red box highlights the 'Upload metadata file' button at the top of the page.

16. Then we are going to upload the metadata file which we'd downloaded from AWS and click on Add.

Upload metadata file.

Values for the fields below are provided by AWS IAM Identity Center (successor to AWS Single Sign-On). You may either enter those values manually, or upload a pre-configured SAML metadata file if provided by AWS IAM Identity Center (successor to AWS Single Sign-On).

"2024-7-1_16-044_d-9f670ee52b_sp_saml_metadata.xml"



Add

Cancel

17. Now you can see the basic configuration settings and just click on save no need to add or change anything.

Basic SAML Configuration



[Save](#) | [Got feedback?](#)

Identifier (Entity ID) *

The unique ID that identifies your application to Microsoft Entra ID. This value must be unique across all applications in your Microsoft Entra tenant. The default identifier will be the audience of the SAML response for IDP-initiated SSO.

Default

https://ap-south-1.signin.aws.amazon.com/platform/saml/d-9f670ee52b



Add identifier

Patterns: https://REGION.signin.aws.amazon.com/platform/saml/*

Reply URL (Assertion Consumer Service URL) *

The reply URL is where the application expects to receive the authentication token. This is also referred to as the "Assertion Consumer Service" (ACS) in SAML.

Index	Default
-------	---------

https://ap-south-1.signin.aws.amazon.com/platform/saml/acs/c45a599c-d538-466c...	0
--	---



Add reply URL

Patterns: https://<REGION>.signin.aws.amazon.com/platform/saml/acs/<ID>

18. Once it is done now you are going to scroll down a little and from the SAML Certificates, you need to download the Federation Metadata XML. So, this file will be required in the AWS console.

3 SAML Certificates

Token signing certificate

Status	Active	Edit
Thumbprint	F119E9FFF67CE0D45E688F867F31D516E424DBE1	
Expiration	1/8/2027, 4:57:39 pm	
Notification Email	DemoUser@Cloudfreaks451.onmicrosoft.com	
App Federation Metadata Url	https://login.microsoftonline.com/429a3ec3-3806-483c-93d1-6a0d96c33d22/FederationMetadata/2007-06/FederationMetadata.xml	Download
Certificate (Base64)		Download
Certificate (Raw)		Download
Federation Metadata XML		Download

Verification certificates (optional)

Required	No	Edit
Active	0	
Expired	0	

19. Now comeback to AWS Console on the same page where you have left if you scroll down, you will see this identity provider metadata option. Here in the IdP SAML metadata you need to upload the certificate which you downloaded from Azure Portal. Click on next.

Identity provider metadata

AWS requires specific metadata provided by your identity provider (IdP) to establish trust. You may copy and paste from your IdP, type the metadata in manually, or upload a metadata exchange file that you download from your IdP.

IdP SAML metadata

[Choose file](#)

AWS IAM Identity Center (successor to AWS Single Sign-On).xml [X](#)

14.23 KB
2024-08-01T16:59:41

20. In the end you just need to verify everything and type ACCEPT, then just click on change identity source.

Review and confirm

⚠ Review the following consequences of your requested identity source change:

- You are changing your identity source to use an external identity provider (IdP).
- IAM Identity Center will delete your current multi-factor authentication (MFA) configuration.
- All current permission sets and SAML 2.0 application configurations will be retained.
- IAM Identity Center preserves your current users and groups, and their assignments. However, only users who have usernames that match the usernames in your identity provider (IdP) can authenticate.
- You must complete your identity provider (IdP) SAML configuration for IAM Identity Center so that your users can sign in. Identity Center will use your IdP for all authentications.
- You must manage your multi-factor authentication (MFA) configuration and policies in your identity provider (IdP).
- You must add (provision) all users in your identity provider (IdP) who will use IAM Identity Center before they can sign in. If you enable System for Cross-domain Identity Management (SCIM) to provision users and groups (recommended), your IdP will be the authoritative source of users and groups, and you must add and modify them in your IdP. Without SCIM, you can provision users and manage groups in IAM Identity Center only; all provisioned usernames must match the corresponding usernames in your IdP.
- IAM Identity Center will keep your current configuration of attributes for access control. We recommend that you review your configuration and update it after you complete the identity source change.

Confirm that you want to change your identity source by entering ACCEPT in the field below.

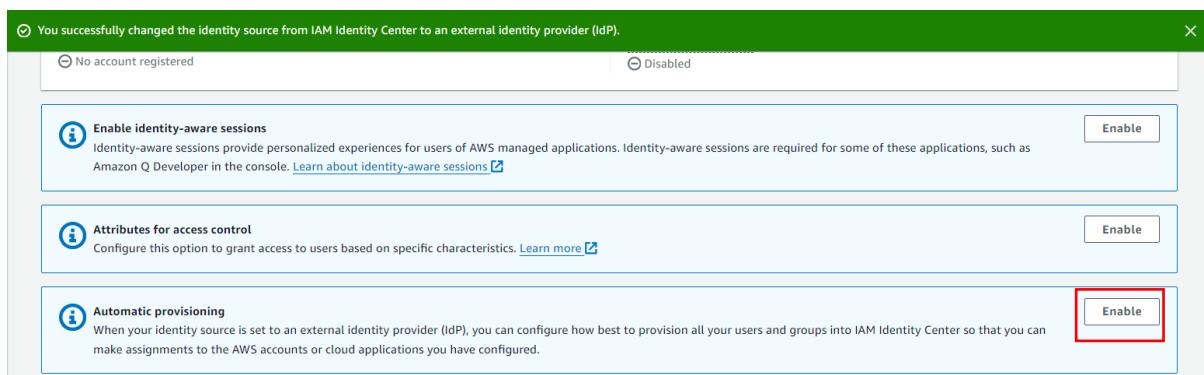
ACCEPT

Cancel

Previous

Change identity source

21. You also get a message that you have successfully changed the identity. Now you are going to enable automation provisioning.



22. So, this automatic provisioning will give you an endpoint and access token. You must copy them both onto your notepad.

Inbound automatic provisioning

X



Automatic provisioning was successfully enabled in your Identity Center directory.

Next you'll need to provide the following information to configure your external provider and create the trust relationship.

Note: Only the top-level groups from your identity provider will be provisioned in your Identity Center directory. [Learn more](#)

Download or copy the access token as this is the only time it will be shown

You cannot recover it later. However, you can generate new tokens at any time. [Learn more](#)

SCIM endpoint

<https://scim.ap-south-1.amazonaws.com/lSwc443cde5-e83e-49cb-a5bc-064578f35ee0/scim/v2>

Access token

[16f30860-c011-4666-a5ef-242a04c88809:2ae71b2e-f653-4e09-8c08-53e94c3517c5:RKU41htKuWXmQhTksdP+s2lemn2f6oqFhBnp8ohcLGWhd595NV5FSc8XTPGA5GN/Z64E/uloVW+AOS4JQ5qwE3L9qqnZV/Ouxhvlu8IDJYNet3/9cnlP3llyyDQAFPMjaZM+UhslegIT4iT0e7RYEy01iaCTZOE0vNY=:AzJ7NsLusP1D3zZC2yhnFMmtfj72vIYu+3SdH7NNh8qy/GS8NFQSuUJOh92FYLKLAGk1y4U+INV6LB3HcHzQGUT3WU+L3JW6HhFQ3c6UXg+fLDLNC2wtLmzLmFoP6jQXhLFofMkqKjvaCim/SQ8uJILJ2xFHbe48fr3bBL1oAxNZ6bqF0pJm+DXcvggR6Vqps7NDPiYRV1iFQoToFqSUDIuso7q74P7e9dgl245DDJmpqy6hd96kj/w22DFRnivbCFFUEOxdhGnO8Ntk/hx03Ny8AiGPvE2i40+MMvLO5cljrv3YNH/NuPVutAiB1FguMeHgVqBHFArTwpdIN5Y1EA==](https://scim.ap-south-1.amazonaws.com/lSwc443cde5-e83e-49cb-a5bc-064578f35ee0/scim/v2)

[Hide token](#)

[Close](#)

23. Once you have the endpoint and token now you are going to Azure Portal, here you need to go to Provisioning.

▽ Manage

Properties

Owners

Roles and administrators

Users and groups

Single sign-on

Provisioning

Self-service

24. Now here you have to go to provisioning and choose the provisioning mode to Automatic. Then you need to paste the SCIM Endpoint and Access Token. After that click on Test connection. If your test is successful then click on save.

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

The screenshot shows the AWS IAM Identity Center Provisioning page. The left sidebar has options like Overview, Provision on demand, Manage, **Provisioning** (which is selected and highlighted with a red box), Monitor, Provisioning logs, Audit logs, Insights, Troubleshoot, and New support request. The main area shows 'Provisioning Mode' set to 'Automatic'. Below it, there's a note about using Microsoft Entra to manage user accounts. Under 'Admin Credentials', there's a 'Tenant URL' input field containing 'https://scim.ap-south-1.amazonaws.com/ISwc443cde5-e83e-49cb-a5bc-064578f35ee0/scim/v2' with a green checkmark icon. A 'Secret Token' input field contains a long string of dots. A red box highlights the 'Test Connection' button. A modal dialog box is displayed at the bottom left, titled 'Testing connection to AWS IAM Identity Center (successor to AWS Single Sign-On)'. It contains the message 'The supplied credentials are authorized to enable provisioning' and has a checkmark icon and an 'X' icon.

25. Now we are going to add the users from Azure to our Enterprise application. By doing so we can also see the users in our AWS Console. So, for that come back to your application in Azure Portal and choose users and groups then click on add user/group.

The screenshot shows the AWS IAM Identity Center interface. The top navigation bar includes the Microsoft Azure logo, a search bar, and a user profile for 'DemoUser@Cloudfree... CLOUDFREEKS (CLOUDFREEKS45...)'. Below the navigation is a breadcrumb trail: Home > Cloudfreeks | Enterprise applications > Enterprise applications > All applications > AWS IAM Identity Center (successor to AWS Single Sign-On). The main title is 'AWS IAM Identity Center (successor to AWS Single Sign-On) | Users and groups'. The left sidebar has a tree view with 'Diagnose and solve problems', 'Manage' (Properties, Owners, Roles and administrators, **Users and groups**), 'Single sign-on', 'Provisioning', 'Self-service', 'Custom security attributes', 'Security', 'Activity', and 'Troubleshooting + Support'. The main content area has a header 'Assign users and groups to app-roles for your application here. To create new app-roles for this application, use the application registration.' Below it is a search bar 'First 200 shown, to search all users & gro...'. A table with columns 'Display Name', 'Object Type', and 'Role assigned' is shown, with a note 'No application assignments found'. At the top of the content area are buttons: '+ Add user/group' (highlighted with a red box), 'Edit assignment', 'Remove', 'Update credentials', 'Columns', and 'Got feedback?'. A tooltip says 'The application will appear for assigned users within My Apps. Set 'visible to users?' to no in properties to prevent this.'

26. Then you need to add a user and click on save. Below you can see that user has been added.

Add Assignment

Cloudfreaks

Users and groups

1 user selected.

Select a role

User

AWS IAM Identity Center (successor to AWS Single Sign-On) | Users and groups

Enterprise Application

Diagnose and solve problems

Manage

Properties

Owners

Roles and administrators

Users and groups

Single sign-on

Provisioning

The application will appear for assigned users within My Apps. Set 'visible to users?' to no in properties to prevent this.

Assign users and groups to app-roles for your application here. To create new app-roles for this application, use the application registration.

First 200 shown, to search all users & gro...

Display Name	Object Type	Role assigned
<input type="checkbox"/> Adminuser	User	User

27. Once again go to provisioning and click on start provisioning.

The screenshot shows the AWS IAM Identity Center Overview page. In the top right corner, there is a modal window titled "Start provisioning" with the message "Provisioning is scheduled to start." Below the modal, the main interface shows the "Current cycle status" as "Incremental cycle stopped." Under "Statistics to date", it shows "0% complete". There are links to "View provisioning details" and "View technical information". A "View provisioning logs" link is also present. On the left sidebar, under the "Overview" section, there are links for "Provision on demand", "Manage" (which includes "Provisioning", "Users and groups", "Expression builder"), "Monitor" (which includes "Provisioning logs", "Audit logs", "Insights"), "Troubleshoot", and "New support request".

28. Now you need to restart the page and you will see that your user has been added.

NOTE: Your Azure User should have a first and last name. It is mandatory or it will not show up.

The screenshot shows the AWS IAM Identity Center Overview page after a provisioning cycle has completed. The "Current cycle status" is now "Incremental cycle completed." with a progress bar at "100% complete". The "Statistics to date" section displays the following details: **Completed:** 1/8/2024, 5:20:21 pm; **Duration:** 2.210 seconds; **Steady state achieved:** 1/8/2024, 5:20:21 pm; **Provisioning interval(fixed):** 40 minutes. The "Users" section shows a count of 1. A "View provisioning logs" link is available. The left sidebar remains the same as in the previous screenshot.

29. Then come back to AWS Console and go to user section in IAM Identity Center. Here you will see the user has been added successfully.

The screenshot shows the AWS IAM Identity Center Users page. The sidebar on the left includes "Managing instance", "Dashboard", "Users" (which is selected), "Groups", "Settings", "Multi-account permissions", "AWS accounts", "Permission sets", and "Application assignments". The main content area shows a message: "Your identity source is currently configured as 'External identity provider'. To add new users or edit their attributes, you must do this using your external identity provider (IdP)." Below this, a table lists a single user: "Adminuser@Cloudfreaks451.onmicrosoft...." with a display name "Adminuser", status "Enabled", and created by "SCIM". There are "Edit users" and "Delete users" buttons at the top of the table.

30. Now we are going to assign this user with some permissions so that it can work. For that in the IAM Identity Center we need to click on AWS Accounts, then choose your account after that you need to choose Assign Users or Groups.

The screenshot shows the IAM Identity Center interface for the 'cloudfreaks2' account. On the left, there's a sidebar with options like Dashboard, Users, Groups, Settings, Multi-account permissions (with 'AWS accounts' selected), Application assignments, and Related consoles (CloudTrail, AWS Organizations, IAM). The main area shows the account overview with details: Account name (cloudfreaks2), Account ID (878895308172), and Email (pulkitkumar2711@gmail.com). Below this, the 'Users and groups' tab is selected, showing the 'Assigned users and groups (0)' section. A large red box highlights the 'Assign users or groups' button at the bottom of this section.

31. Now you need to choose the user and click on next.

The screenshot shows the 'Assign users and groups' step in the IAM Identity Center. It's a three-step process: Step 1 (Select users and groups), Step 2 (Select permission sets), and Step 3 (Review and submit). The current screen is Step 1. It has tabs for 'Users' and 'Groups', with 'Users' selected. A list of users is shown, with one user ('Adminuser@Cloudfreaks451.onmicrosoft.com') selected and highlighted with a blue border. At the bottom, there's a 'Selected users and groups (1)' summary and a 'Remove' button. The 'Next' button is highlighted with a yellow box.

32. Then you have to click on Create Permission set.

IAM Identity Center > AWS Organizations: AWS accounts > cloudfreeks2 > Assign users and groups

Step 1
Select users and groups

Step 2
Select permission sets

Step 3
Review and submit

Assign permission sets to "cloudfreeks2"

Permission sets define the level of access that users and groups in IAM Identity Center have to an AWS account. You can assign more than one permission set to a user. To ensure least privilege access to AWS accounts, users in IAM Identity Center with multiple permission sets on an AWS account must pick a specific permission set when selecting the account and then return to the AWS access portal to pick a different set when necessary [Learn more](#)

Permission sets (0)

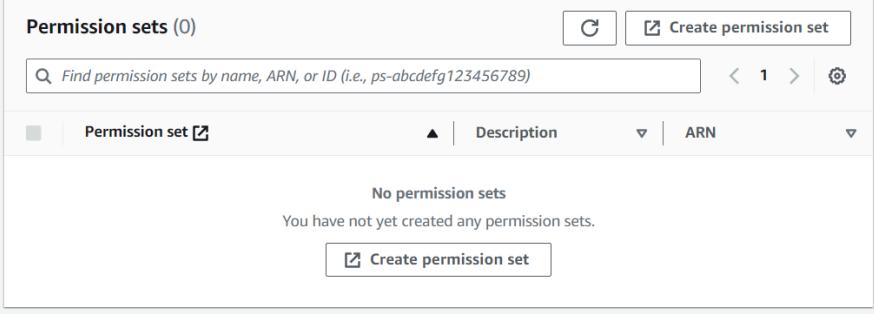
Create permission set

Find permission sets by name, ARN, or ID (i.e., ps-abcdefg123456789)

No permission sets
You have not yet created any permission sets.

Create permission set

Cancel Previous Next



33. Then just choose the predefined permission set and in the choose administrator access and create your permission set.

IAM Identity Center > Permission sets > Create permission set

Step 1
Select permission set type

Step 2
Specify permission set details

Step 3
Review and create

Select permission set type

A permission set contains policies that determine a user's permissions to access an AWS account. When you assign a user or group to a permission set in an AWS account, IAM Identity Center creates an IAM role in the account and attaches the policies specified in the permission set to that role. Select an option to specify the permission set type. [Learn more](#)

Permission set type

Types

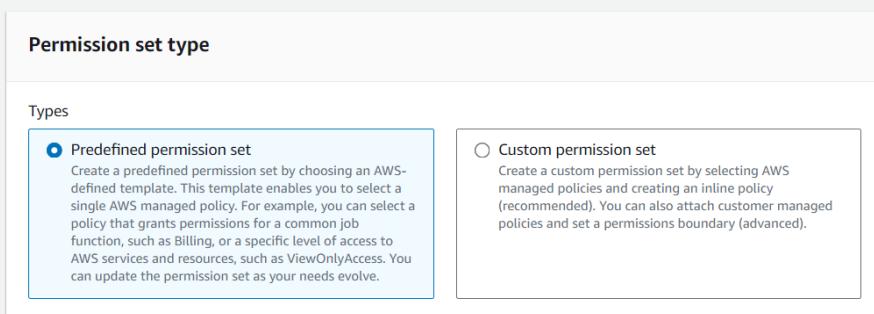
Predefined permission set
Create a predefined permission set by choosing an AWS-defined template. This template enables you to select a single AWS managed policy. For example, you can select a policy that grants permissions for a common job function, such as Billing, or a specific level of access to AWS services and resources, such as ViewOnlyAccess. You can update the permission set as your needs evolve.

Custom permission set
Create a custom permission set by selecting AWS managed policies and creating an inline policy (recommended). You can also attach customer managed policies and set a permissions boundary (advanced).

Policy for predefined permission set

Select an AWS managed policy

AdministratorAccess
Provides full access to AWS services and resources.



34. Once it is created come back and refresh it you will see your Administrator Access Permission set choose it and click on next to go to review page. Then just create it.

IAM Identity Center > AWS Organizations: AWS accounts > cloudfreeks2 > Assign users and groups

Step 1
[Select users and groups](#)

Step 2
Select permission sets

Step 3
Review and submit

Assign permission sets to "cloudfreeks2"

Permission sets define the level of access that users and groups in IAM Identity Center have to an AWS account. You can assign more than one permission set to a user. To ensure least privilege access to AWS accounts, users in IAM Identity Center with multiple permission sets on an AWS account must pick a specific permission set when selecting the account and then return to the AWS access portal to pick a different set when necessary [Learn more](#)

Permission sets (1/1)

Permission set	Description	ARN
<input checked="" type="checkbox"/> AdministratorAccess	-	arn:aws:ss:::permission Set/ssoin- 6595436b94d89de8/ps -dd8eedbebdda6dd9

35. Below you can see that the user now has the administrator access.

cloudfreeks2

Overview

Account name cloudfreeks2	Account ID 878893308172	Email pulkitkumar2711@gmail.com
------------------------------	----------------------------	------------------------------------

Users and groups (1) Permission sets (1)

Assigned users and groups (1)

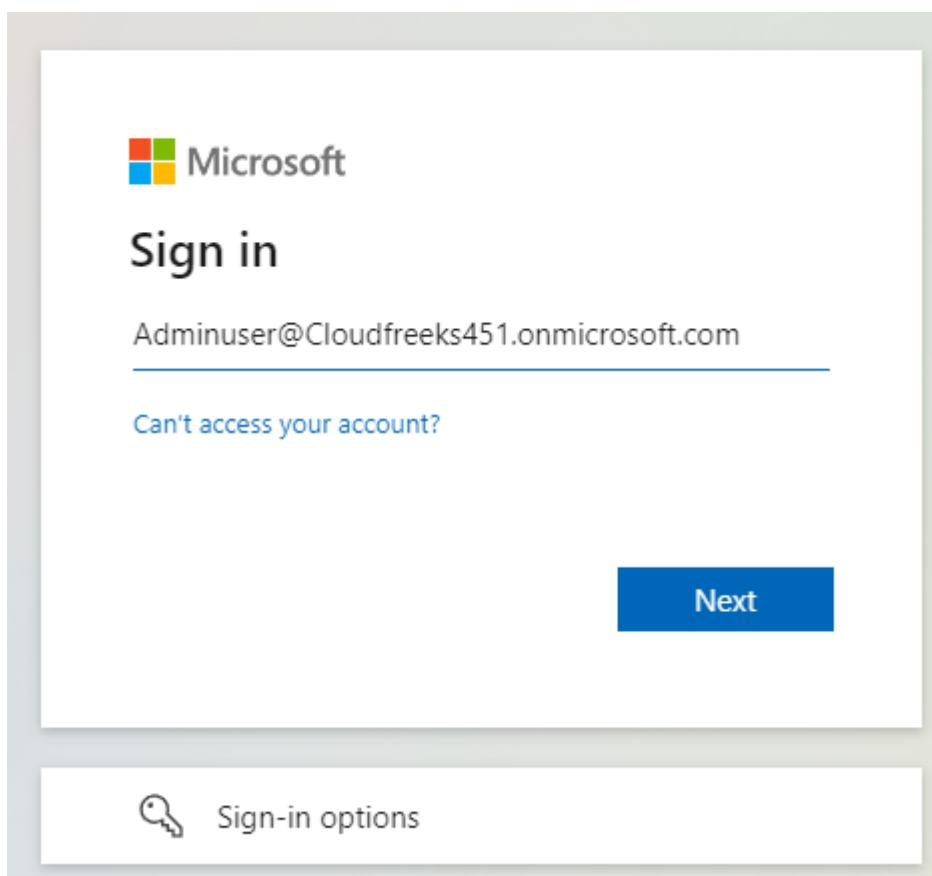
The following users and groups in IAM Identity Center can select this AWS account from within their AWS access portal. [Learn more](#)

Username / group name	Permission sets	Type
<input type="radio"/> Adminuser@Cloudfreeks451.onmicrosoft.com	AdministratorAccess	User

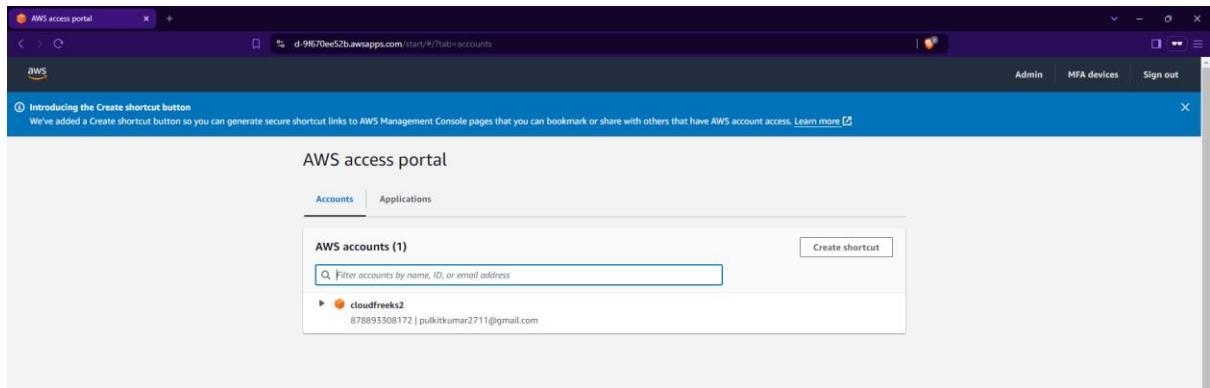
36. Then just go to the Dashboard of IAM Identity Center and copy the AWS Access Portal URL.

The screenshot shows the IAM Identity Center dashboard. The left sidebar has a 'Dashboard' tab selected, along with 'Users', 'Groups', and 'Settings'. Under 'Multi-account permissions', there are 'AWS accounts' and 'Permission sets'. Under 'Application assignments', there is an 'Applications' section. On the right, there's a 'Settings summary' panel with fields for 'Instance name' (set to 'Specify a unique name for your instance'), 'Identity source' (set to 'External identity provider'), 'Region' (set to 'Asia Pacific (Mumbai) | ap-south-1'), and 'Organization ID' (set to 'o-8d5816r04'). A red box highlights the 'AWS access portal URL - Edit' field, which contains the URL <https://d-9f670ee52b.awssapps.com/start>.

37. After that open a new browser and paste the URL and it will ask you to sign in, you can use your Azure User credentials to login. So, give the username and password.



38. Below you can see that you are in the AWS Access Portal now .



39. Then expand the account name and click on Administrator Access.

A screenshot of the AWS access portal, similar to the previous one but with a red box highlighting the "AdministratorAccess" link next to the account name "cloudfreeks2".

40. Now you can see that you are in the AWS account using Azure User Credentials.

A screenshot of the AWS Console Home page. The top navigation bar shows the URL "eu-north-1.console.aws.amazon.com/console/home?region=eu-north-1" and the user "AdministratorAccess/Adminuser@Cloudfreeks451.onmicrosoft.com". The main dashboard includes sections for "Recently visited" services (with a note: "No recently visited services"), "Applications" (with a note: "No applications"), "Welcome to AWS" (with a note: "Getting started with AWS"), and "Cost and usage" (with a note: "Access denied"). The bottom of the page includes links for "CloudShell", "Feedback", and copyright information: "© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences".

Configuring the AWS CLI to use IAM Identity Center

What are we doing in this part?

In this process, you are setting up the AWS Command Line Interface (CLI) to use AWS IAM Identity Center (previously known as AWS Single Sign-On) for authentication. The end goal is to securely manage access to your AWS resources by using centralized authentication through Azure Active Directory.

Summary of Steps:

1. **Install AWS CLI:** Begin by installing the AWS CLI on your computer.
2. **Configure AWS CLI with Identity Center:** Use the `aws configure sso` command to set up Single Sign-On (SSO) authentication. You'll provide the SSO Start URL and other details from the AWS IAM Identity Center.
3. **Authentication Flow:** After entering the configuration details, you are redirected to a web browser to authenticate. You match a code displayed in the command prompt with one in the browser to confirm your identity.
4. **Granting Permissions:** You ensure that your Azure Admin user account has the necessary permissions in the AWS IAM Identity Center and Azure Portal.
5. **Assign Administrator Access:** Assign the appropriate roles and permissions to the user in AWS IAM Identity Center, ensuring they have the required access to AWS resources.
6. **Finalizing Configuration:** Complete the setup by specifying profile details like region and output format. Then, use the `aws sso login --profile demo-admin` command to log in, which authenticates you via the browser.
7. **Access AWS Console:** After successful authentication, you can access the AWS Management Console with the assigned administrator role.

End Goal:

The end goal is to make it easy and secure for you to log in and manage your AWS resources using your existing Azure account. By setting this up, you can use a single sign-on process to access AWS, so you don't have to remember multiple passwords or go through complicated login steps. This setup also ensures that only authorized users can access important AWS tools and services, keeping everything secure.

To begin in this part

1. Now we are going to configure AWS CLI to use Identity Center for that first we need to install AWS CLI into our laptop.
2. Once it is done then we are going to run some commands to configure it. Now enter the command given below into your command prompt.

`aws configure sso`

3. Then you will see that it will ask you the session name, and the SSO Start URL which you get from the AWS IAM Identity Center Dashboard. You can also see that in the snapshot below.
4. After that write your SSO region then leave the SSO registration scopes and hit on Enter.
5. Now you are directed to the browser and there you need to match the code which you can see below in the command prompt.

```
C:\>aws configure sso
SSO session name (Recommended): DemoSession
SSO start URL [None]: https://d-9f670ee52b.awsapps.com/start
SSO region [None]: ap-south-1
SSO registration scopes [sso:account:access]:
Attempting to automatically open the SSO authorization page in your default browser.
If the browser does not open or you wish to use a different device to authorize this request, open the following URL:
https://device.sso.ap-south-1.amazonaws.com/
Then enter the code:
```

CHRR-XJJN

Dashboard

IAM Identity Center enables you to manage workforce user access to multiple AWS accounts and applications. [Learn more](#)

Central management

Use service control policies (SCPs) to prevent instances of IAM Identity Center from being created, or isolate the member accounts that are allowed to create account instances. [Learn more about service control policies](#)

Monitor activities in your instances of IAM Identity Center With AWS CloudTrail, you can monitor and audit activity in your organization instance and account instances of IAM Identity Center. [Learn about monitoring IAM Identity Center](#)

Settings summary

[Go to settings](#)

Specify a unique name for your instance.
Instance name - Edit

Identity source
External identity provider

Region
Asia Pacific (Mumbai) | ap-south-1

Organization ID
o-x8d5816r04

AWS access portal URL - Edit
<https://d-9f670ee52b.awsapps.com/start>

Issuer URL
<https://identitycenter.amazonaws.com/ssoins-6595436b94d89de8>

6. On the browser also you can see the same code. If the code matches, then you need to click on Confirm and Continue.



Authorization requested

An application or service has requested access to your AWS account(s) and resources.



Confirm this code matches the one given to you.

CHRR-XJJN

If you did not initiate this request or your codes do not match, cancel this request.

Confirm and continue

Cancel

NOTE: If you are facing this error shown below after clicking on confirm and continue then it means that you need to add your Azure Admin user account into the Enterprise application which is AWS IAM Identity center application in Azure Portal.

You need to go inside of your Enterprise application and click on Users and Groups then choose to add new user then choose to add your Account Admin user and then wait for it to get provisioned.

Once it has been provisioned then you need to come back to AWS and go to users in IAM Identity Center. There you need to check whether the user has been added or not. If it has been added, then you need to go AWS Accounts and from there give your new user the Administrator access like you had given to the previous user.

Once it is done then you can go back to command prompt and run the aws sso configure command again and you will be able to perform the steps shown below.



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

Sorry, but we're having trouble signing you in.

AADSTS50105: Your administrator has configured the application AWS IAM Identity Center (successor to AWS Single Sign-On) ('fbcc05c6-6cc3-443c-866e-dc2935eb19cb') to block users unless they are specifically granted ('assigned') access to the application. The signed in user 'DemoUser@Cloudfreaks451.onmicrosoft.com' is blocked because they are not a direct member of a group with access, nor had access directly assigned by an administrator. Please contact your administrator to assign access to this application.

Troubleshooting details



If you contact your administrator, send this info to them.

[Copy info to clipboard](#)

Request Id: cd4ad298-1f8d-4906-8001-773c3bbdf400

Correlation Id: bc2570f5-a49d-4223-b2b3-f6d69dfffc3a

Timestamp: 2024-08-02T04:50:44Z

Message: AADSTS50105: Your administrator has configured the application AWS IAM Identity Center (successor to AWS Single Sign-On) ('fbcc05c6-6cc3-443c-866e-dc2935eb19cb') to block users unless they are specifically granted ('assigned') access to the application. The signed in user 'DemoUser@Cloudfreaks451.onmicrosoft.com' is blocked because they are not a direct member of a group with access, nor had access directly assigned by an administrator. Please contact your administrator to assign access to this application.

Flag sign-in errors for review: [Enable flagging](#)

If you plan on getting help for this problem, enable flagging and try to reproduce the error within 20 minutes. Flagged events make diagnostics available and are raised to admin attention.

- After that you will see this screen and you need to click on Allow Access. Then will see a message saying that your request has been approved.



Allow botocore-client-DemoSession to access your data?

By choosing **Allow access**, you agree to allow **botocore-client-DemoSession** to access the following:



Applications and AWS accounts
[Show details](#)

[Deny access](#)

[Allow access](#)



Request approved

botocore-client-DemoSession can now
access your data in Applications and AWS
accounts.

You can close this window.

8. Now you should come back to command prompt and here you will see that it is saying that the only account available is my own account and the role name is Administrator access.
9. Then just leave the default region name and default output format. After that you need to give a profile name. If you want to use, then you can use the default name AWS is suggesting you.

```
Then enter the code:
```

```
PTNM-HHGL
The only AWS account available to you is: 878893308172
Using the account ID 878893308172
The only role available to you is: AdministratorAccess
Using the role name "AdministratorAccess"
CLI default client Region [ap-south-1]:
CLI default output format [none]:
CLI profile name [AdministratorAccess-878893308172]: demo-admin

To use this profile, specify the profile name using --profile, as shown:
aws s3 ls --profile demo-admin
```

10. Once your profile is setup now you are going to login for that you need to write this command in your command prompt. Then it will take you to the browser again to confirm the code. Just click on confirm and continue.

```
aws sso login --profile demo-admin
```



Authorization requested

An application or service has requested access to your AWS account(s) and resources.



Confirm this code matches the one given to you.

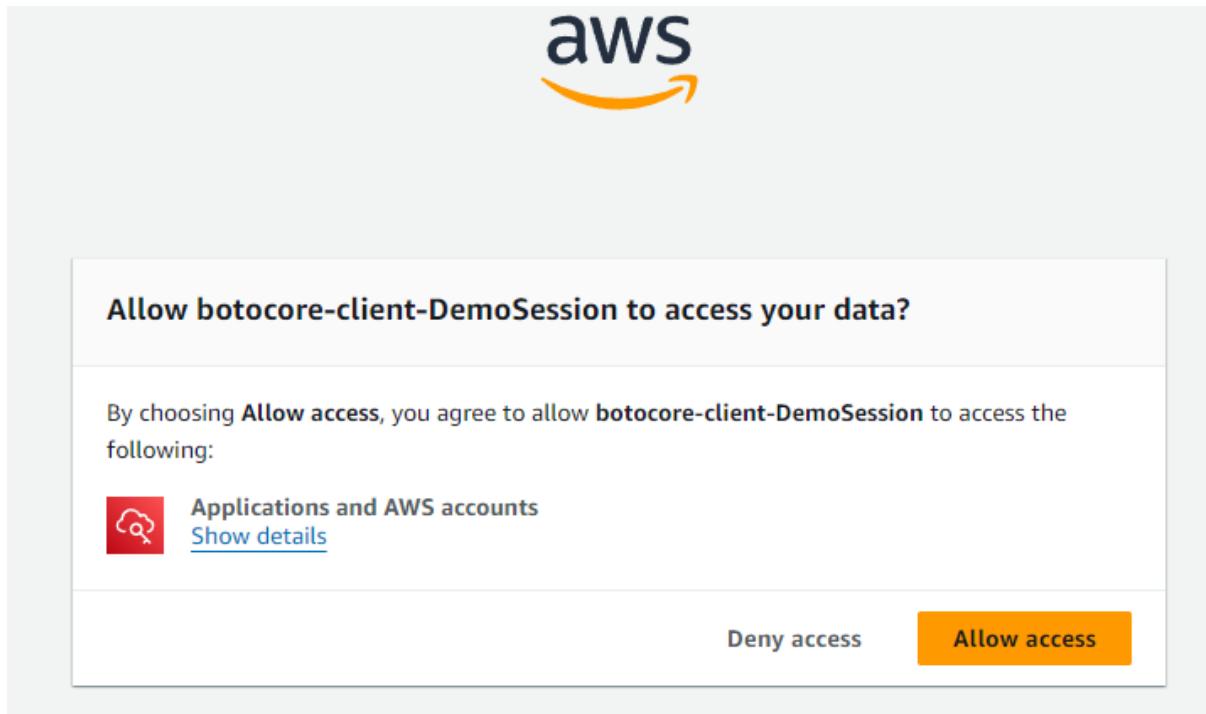
RPZD-PBKG

If you did not initiate this request or your codes do not match, cancel this request.

Confirm and continue

Cancel

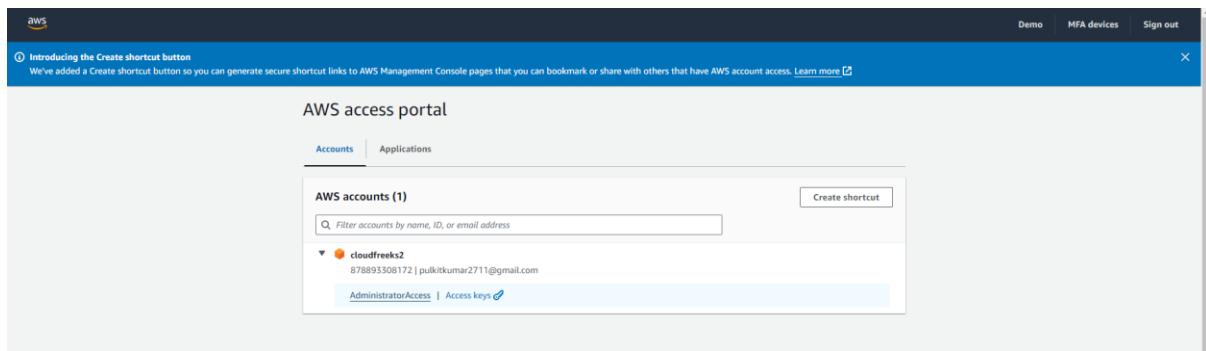
11. Now you need to click on Allow Access and then close the tab.



12. After that come back to the command prompt and here you will see that it has given you the URL. Copy this URL and paste it in a new tab or browser.

```
C:\>aws sso login --profile demo-admin
Attempting to automatically open the SSO authorization page in your default browser.
If the browser does not open or you wish to use a different device to authorize this request, open the following URL:
https://device.sso.ap-south-1.amazonaws.com/
Then enter the code:
RPZD-PBKG
Successfully logged into Start URL: https://d-9f670ee52b.awsapps.com/start
C:\>
```

13. Then you will be on this page and you need to go to console, for that click on administrator access



14. Then you will be in the Console and you can use it.

AWS Services Search [Alt+S] Stockholm Administrator/Access/demoUser@CloudFreaks451.onmicrosoft.com

Console Home [Info](#)

Recently visited [Info](#)

No recently visited services

Explore one of these commonly visited AWS services.

EC2 S3 RDS Lambda

View all services

Welcome to AWS

Getting started with AWS [Info](#)
Learn the fundamentals and find valuable information to get the most out of AWS.

Training and certification [Info](#)
Learn from AWS experts and advance your skills and

AWS Health [Info](#)

Open issues 0 Past 7 days

Scheduled changes 0 Upcoming and past 7 days

Other notifications 0 Past 7 days

Cost and usage [Info](#)

Current month costs [Access denied](#) Cost breakdown [Access denied](#)

Forecasted month end costs [Access denied](#)

Savings opportunities [Enable Cost Optimization Hub](#)

Reset to default layout + Add widgets

Applications (0) [Info](#) Region: Europe (Stockholm)

eu-north-1 (Current Region) [Find applications](#)

Name Description Region Originating account

No applications

Get started by creating an application.

Create application

Go to myApplications

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

This screenshot shows the AWS Console Home page. It features several sections: 'Recently visited' (empty), 'AWS Health' (with 0 open issues, scheduled changes, and other notifications), and 'Cost and usage' (showing current month costs and forecasted month end costs, both with 'Access denied' status). Other sections include 'Welcome to AWS' (with links to getting started and training/certification) and a search bar at the top.