

Identity and Access Management (IAM)

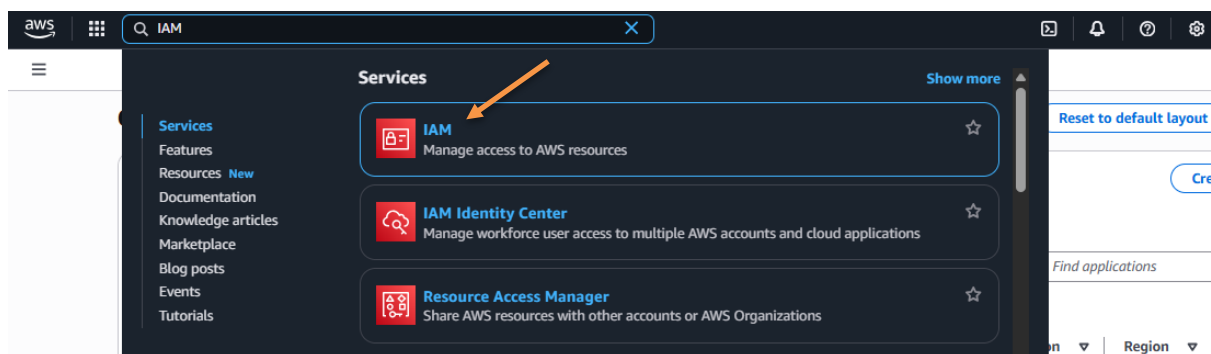
Introduction:

- AWS Identity and Access Management (IAM) is a service that helps securely control access to AWS resources.
- It allows you to create and manage users, groups, and roles, and assign specific permissions to them.
- With IAM, you can define who can access what resources and under what conditions, ensuring strong security and access control in your AWS environment.
- IAM supports fine-grained permission policies and enables multi-factor authentication for enhanced protection.

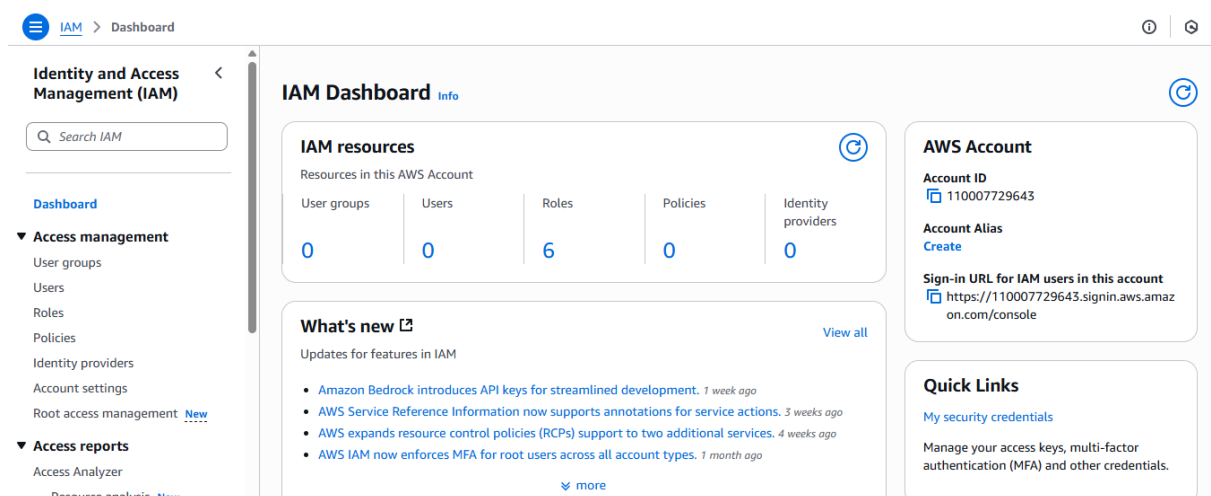
Step by Step Instructions:

Step 1:

- Go to “AWS Management Console” and search “IAM”.



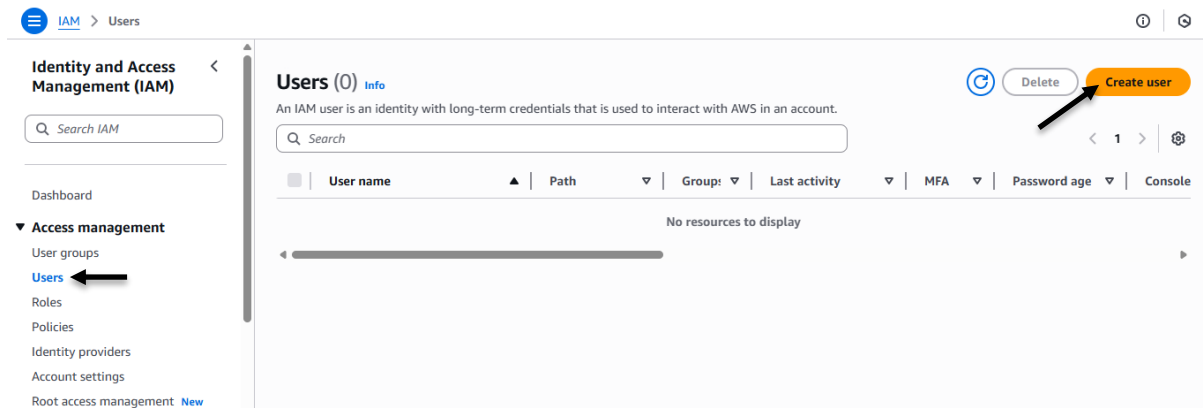
- The “IAM Dashboard” will open.



Creating users with the help of IAM:

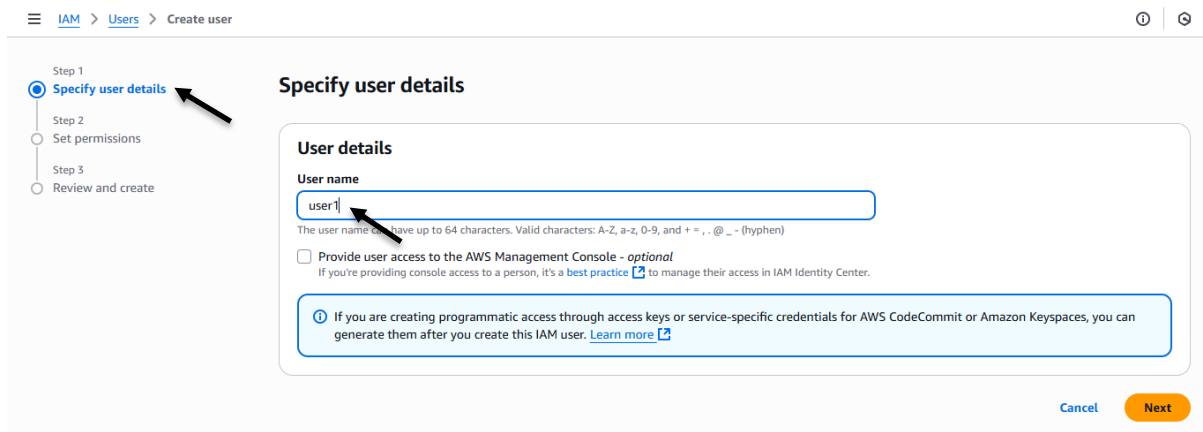
Step 2:

- On the left side under “Access management”, click on “Users”.
- Click on “Create user”.



Step 3:

- In “Specify user details”, write the name of the user.



- Check the “Provide user access to the AWS Management Console” option.
- In “User type”, select “I want to create an IAM user”.
- In “Console password”, select “Custom password” and write a password for user1 (e.g. “User1@123”).

- ☒ Provide user access to the AWS Management Console - *optional*
If you're providing console access to a person, it's a [best practice](#) to manage their access in IAM Identity Center.

Are you providing console access to a person?

User type

- ☐ Specify a user in Identity Center - Recommended

We recommend that you use Identity Center to provide console access to a person. With Identity Center, you can centrally manage user access to their AWS accounts and cloud applications.

- ☒ I want to create an IAM user

We recommend that you create IAM users only if you need to enable programmatic access through access keys, service-specific credentials for AWS CodeCommit or Amazon Keyspaces, or a backup credential for emergency account access.

Console password

- ☐ Autogenerated password

You can view the password after you create the user.

- ☒ Custom password

Enter a custom password for the user.

User1@123

- Must be at least 8 characters long
- Must include at least three of the following mix of character types: uppercase letters (A-Z), lowercase letters (a-z), numbers (0-9), and symbols ! @ # \$ % ^ & * () _ + - (hyphen) = [] { } | ' "

- Uncheck the option “Users must create a new password at next sign-in”.
- Click on “Next”.

- Must be at least 8 characters long
- Must include at least three of the following mix of character types: uppercase letters (A-Z), lowercase letters (a-z), numbers (0-9), and symbols ! @ # \$ % ^ & * () _ + - (hyphen) = [] { } | ' "

- ☐ Show password

- ☐ Users must create a new password at next sign-in - Recommended

Users automatically get the [IAMUserChangePassword](#) policy to allow them to change their own password.

! If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keyspaces, you can generate them after you create this IAM user. [Learn more](#)

Cancel

Next

Step 4:

- In “Set permissions”, select “Attach policies directly” as “Permissions options”.
- Now select the “Permissions policies” that you want to add.

Step 1 Specify user details

Step 2 **Set permissions**

Step 3 Review and create

Step 4 Retrieve password

Set permissions

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Permissions options

☐ Add user to group
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

☐ Copy permissions
Copy all group memberships, attached managed policies, and inline policies from an existing user.

☒ **Attach policies directly**
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

Permissions policies (1378)

Choose one or more policies to attach to your new user.

Filter by Type

Search

All types

< 1 2 3 4 5 6 7 ... 69 >

Permissions policies (1/1378)

Choose one or more policies to attach to your new user.

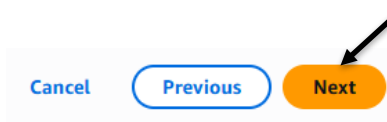
[Create policy](#)

Filter by Type

ec2 All types 44 matches

	Policy name	Type	Attached entities
<input type="checkbox"/>	AmazonEC2ContainerRegistryF...	AWS managed	0
<input type="checkbox"/>	AmazonEC2ContainerRegistryP...	AWS managed	0
<input type="checkbox"/>	AmazonEC2ContainerRegistryP...	AWS managed	0
<input type="checkbox"/>	AmazonEC2ContainerRegistryR...	AWS managed	0
<input type="checkbox"/>	AmazonEC2ContainerServiceA...	AWS managed	0
<input type="checkbox"/>	AmazonEC2ContainerServiceEv...	AWS managed	0
<input type="checkbox"/>	AmazonEC2ContainerServicefo...	AWS managed	0
<input type="checkbox"/>	AmazonEC2ContainerServiceRole	AWS managed	0
<input checked="" type="checkbox"/>	AmazonEC2FullAccess	AWS managed	0
<input type="checkbox"/>	AmazonEC2ReadOnlyAccess	AWS managed	0

- Suppose I want to give EC2 full access to user1 so, I checked the option “AmazonEC2FullAccess”.
- Similarly, you can give other permissions also.
- Click on “Next”.



Step 5:

- Now review all the options and click on “Create user”.

IAM > Users > Create user

Step 1: Specify user details
Step 2: Set permissions
Step 3: Review and create
Step 4: Retrieve password

Review and create

Review your choices. After you create the user, you can view and download the autogenerated password, if enabled.

User details

User name user1	Console password type Custom password	Require password reset No
--------------------	--	------------------------------

Permissions summary

Name	Type	Used as
AmazonEC2FullAccess	AWS managed	Permissions policy

Tags - optional

Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.

No tags associated with the resource.

[Add new tag](#)

You can add up to 50 more tags.

Cancel Previous **Create user**

- User is created.
- You can retrieve the password.

✔ User created successfully
View user ✕

You can view and download the user's password and email instructions for signing in to the AWS Management Console.

Step 1

Specify user details

Step 2

Set permissions

Step 3

Review and create

Step 4

Retrieve password

Retrieve password

You can view and download the user's password below or email users instructions for signing in to the AWS Management Console. This is the only time you can view and download this password.

Console sign-in details

[Email sign-in instructions](#)

Console sign-in URL

<https://110007729643.signin.aws.amazon.com/console>

User name

[user1](#)

Console password

[***** Show](#)

Users (1/1) Info

🔄
Delete
Create user

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

☒ **User name**

☐ Path

☐ Group

☐ Last activity

☐ MFA

☐ Password age

☐ Console

User name	Path	Group	Last activity	MFA	Password age	Console
<input checked="" type="checkbox"/> user1	/	0	-	-	6 minutes	-

Step 6:

- Now go to another browser or incognito tab and open “AWS Management Console”.

- Enter your root account ID in “Account ID or alias” box.

🔔
❓
⚙️
Global ▼
Riddhisha ▲

Account ID

[1100-0772-9643](#)

IAM user sign in ⓘ

Account ID or alias [\(Don't have?\)](#)

110007729643

☐ Remember this account

IAM username

user1

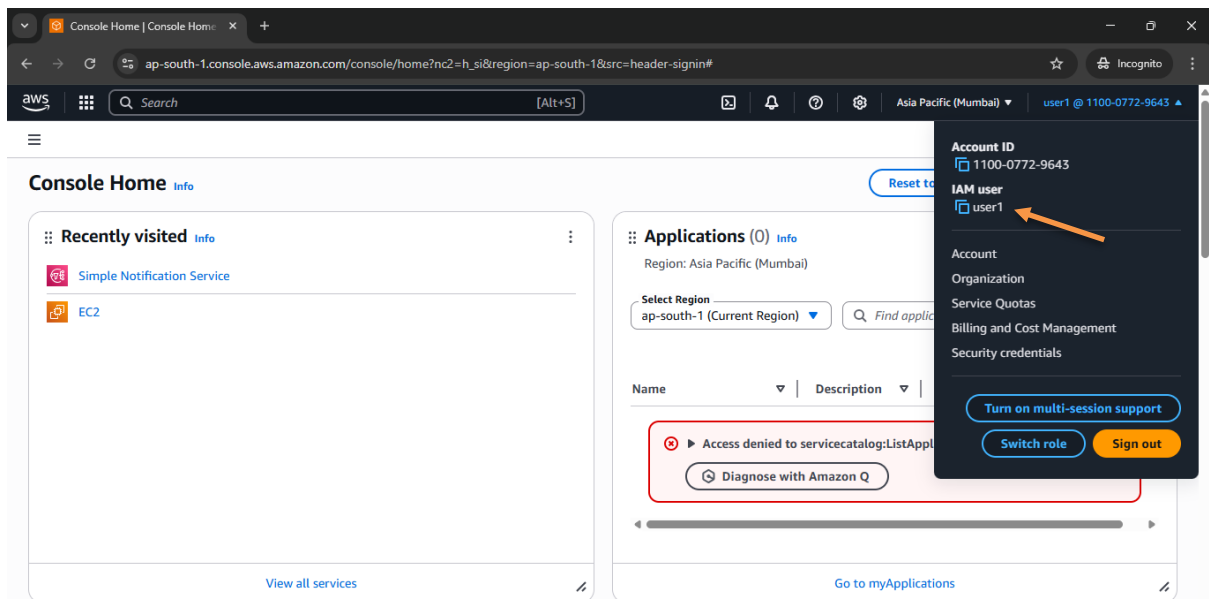
Password

User1@123

☒ Show Password [Having trouble?](#)

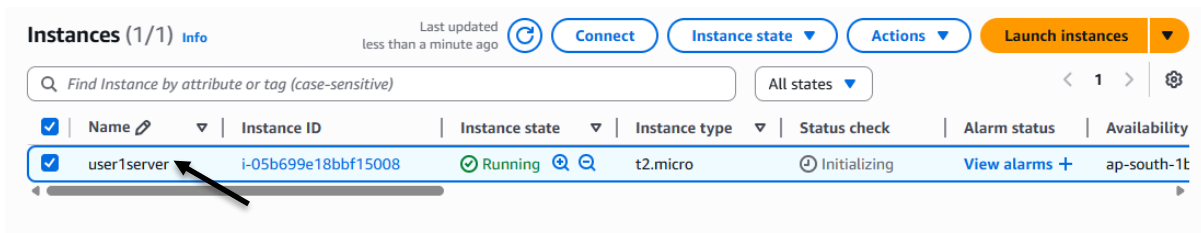
Sign in

- Enter the “IAM username” that you have created i.e. “user1”.
- Enter the password and click on “Sign in”.
- Now “user1” is signed in to the root user account.

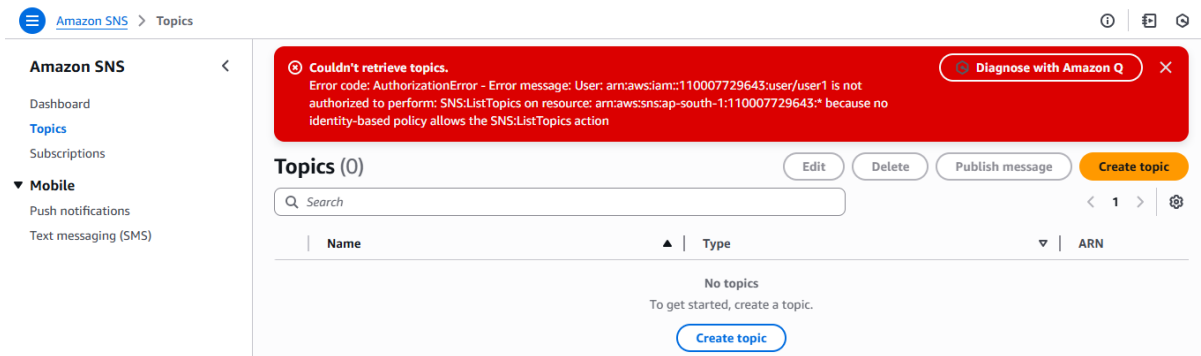


Step 7:

- “user1” can easily access all the properties of EC2.
- But it cannot access other options.
- For example, “user1” can easily launch an instance, create a volume, snapshots, and all other things that comes under EC2.



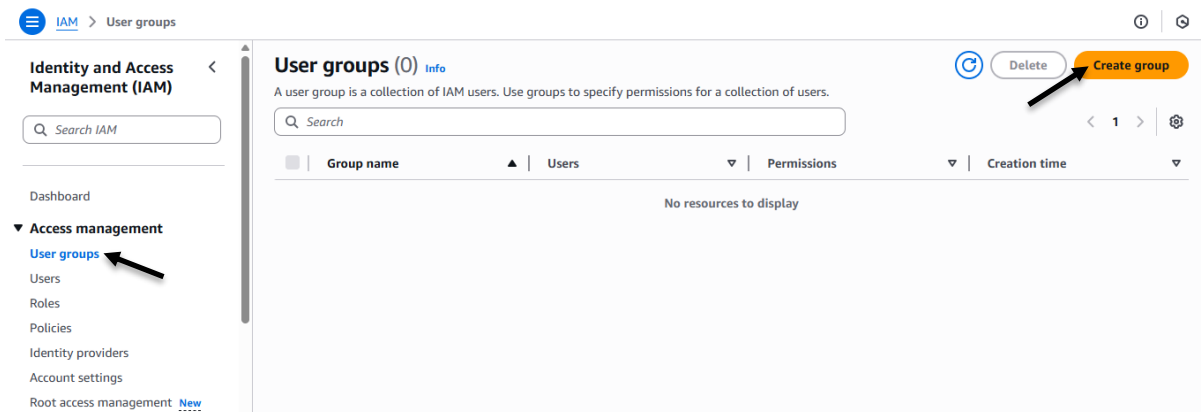
- But if “user1” tries to create a topic in SNS, he/she will be unable to do so.



Creating user groups:

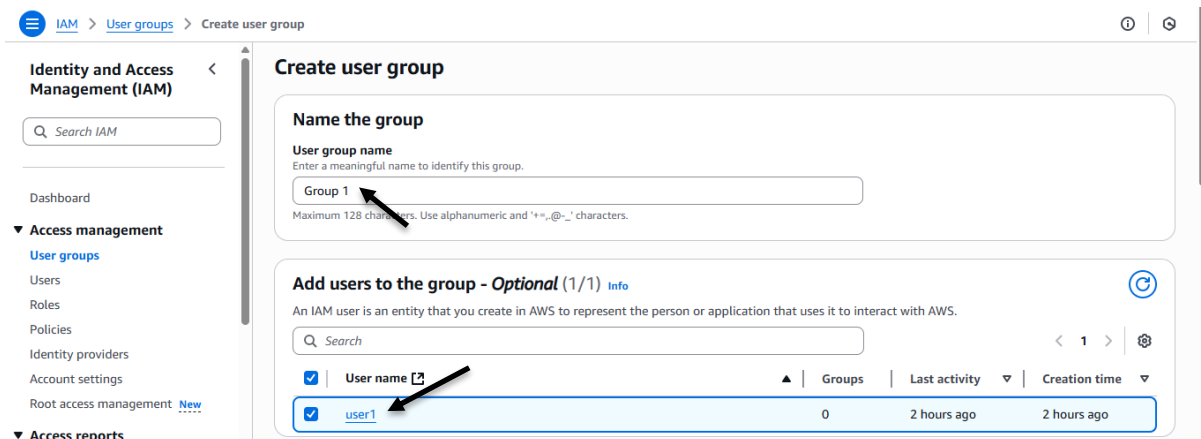
Step 8:

- In “Access management”, go to “User groups” and click on “Create group”.

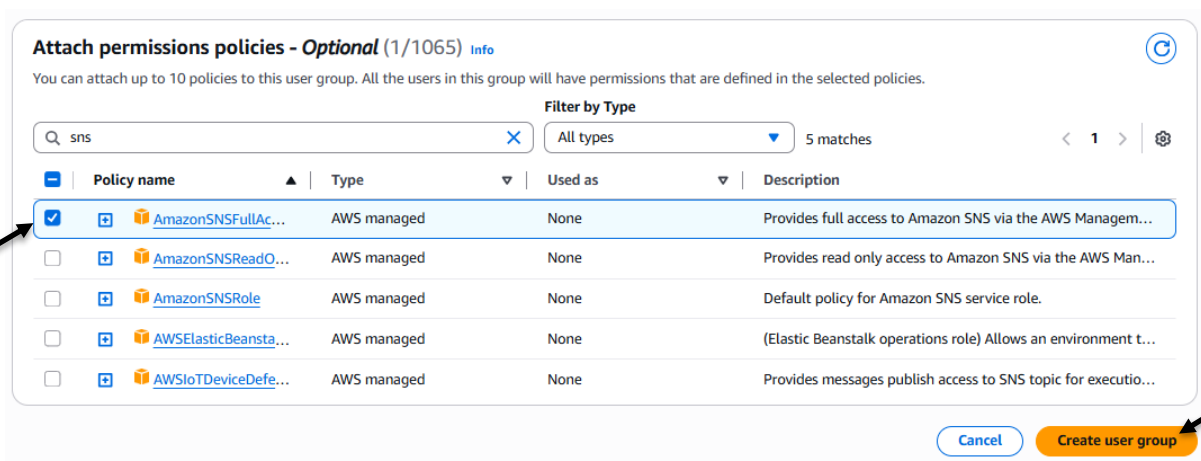


Step 9:

- Write the “User group name”.
- Select the users whom you want to add to the group.
- For now, we will add user1 to “Group 1”.

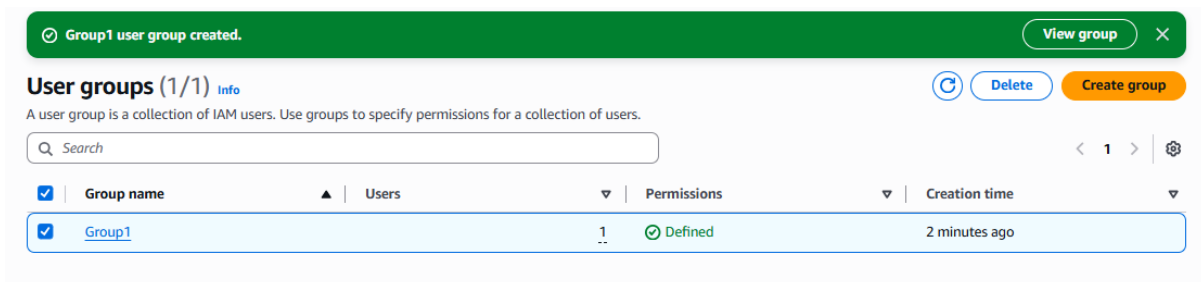


- Attach permission policies to the group as per your choice.
- For example, I have added “AmazonSNSFullAccess”.
- Click on “Create user group”.



Step 10:

- User group “Group1” is created successfully.



Step 11:

- Similarly, if we create another user (e.g. “user2”), under “Set permissions”, we can directly add the user to the group we just created.
- Select “Add user to group”.
- Select “Group1”.

IAM > Users > Create user

Step 1: Specify user details
 Step 2: **Set permissions**
 Step 3: Review and create
 Step 4: Retrieve password

Set permissions

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Permissions options

- ☒ **Add user to group**
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.
- ☐ **Copy permissions**
Copy all group memberships, attached managed policies, and inline policies from an existing user.
- ☐ **Attach policies directly**
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

User groups (1/1)

Search

<input checked="" type="checkbox"/>	Group name	Users	Attached policies	Created
<input checked="" type="checkbox"/>	Group1	1	AmazonSNSFullAccess	2025-07-19 (4 minutes ago)

Create group

- Now we can see that “Group1” has two users.
- We can add more users by clicking on “Add users”.

Group1 Info Delete

Summary Edit

User group name Group1	Creation time July 19, 2025, 23:28 (UTC+05:30)	ARN arn:aws:iam::110007729643:group/Group1
----------------------------------	--	--

Users (2) | Permissions | Access Advisor

Users in this group (2) Remove Add users

An IAM user is an entity that you create in AWS to represent the person or application that uses it to interact with AWS.

Search

<input type="checkbox"/>	User name	Groups	Last activity	Creation time
<input type="checkbox"/>	user1	1	2 hours ago	3 hours ago
<input type="checkbox"/>	user2	1	None	1 minute ago

- The users in Group1 have only one permission – “AmazonSNSFullAccess”.
- They cannot access other features except SNS.

Creating policies:

Step 12:

- In “Access management”, go to “Policies”.
- There are already 1378 policies present.
- We can create and add more policies as per our choice.
- Click on “Create policy”.

Identity and Access Management (IAM)

Search IAM

Dashboard

▼ **Access management**

- User groups
- Users
- Roles
- Policies**
- Identity providers
- Account settings
- Root access management *New*

▼ **Access reports**

- Access Analyzer
- Resource analysis *New*
- Unused access
- Analyzer settings

Policies (1378) *Info*

A policy is an object in AWS that defines permissions.

Filter by Type: All types

Policy name	Type	Used as	Description
AccessAnalyzerServiceRole	AWS managed	None	-
AdministratorAccess	AWS managed - job function	None	Provides full access to AWS services an...
AdministratorAccess-Amp...	AWS managed	None	Grants account administrative permissi...
AdministratorAccess-AWS...	AWS managed	None	Grants account administrative permissi...
AIOpsAssistantPolicy	AWS managed	None	Provides ReadOnly permissions require...
AIOpsConsoleAdminPolicy	AWS managed	None	Grants full access to Amazon AI Opera...
AIOpsOperatorAccess	AWS managed	None	Grants access to the Amazon AI Opera...
AIOpsReadOnlyAccess	AWS managed	None	Grants ReadOnly permissions to the A...
AlexaForBusinessDeviceS...	AWS managed	None	Provide device setup access to AlexaFo...
AlexaForBusinessFullAccess	AWS managed	None	Grants full access to AlexaForBusiness ...

Step 13:

- Now select a service under “Specify permissions”.
- Select “SNS”.

Specify permissions *Info*

Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.

Policy editor *Visual* | *JSON* *Actions* *Visual*

▼ **Select a service**

Specify what actions can be performed on specific resources in a service.

Service

Choose a service

+ Add more permissions

Cancel Next

Filter services

- Auto Scaling
- CloudFront
- EC2
- IAM
- Lambda
- RDS
- S3
- SNS**
- Other services
- Choose a service

- Under “Actions allowed”, select all the permissions that you want to add to your policy.
- Select from “List (7)”, “Read (10)”, “Write (19)”, “Permissions management (3)” and “Tagging (2)”.

▼ SNS

Set permissions for SNS

Specify what actions can be performed on specific resources in SNS.

▼ Actions allowed

Specify actions from the service to be allowed.

Manual actions | [Add actions](#)

☐ All SNS actions (sns:*)

Access level

► List (7)

► Read (10)

► Write (19)

► Permissions management (3)

► Tagging (2)

Effect

☒ Allow
☐ Deny

[Expand all](#) | [Collapse all](#)

▼ List (Selected 3/7)

☐ All list actions

☐ ListEndpointsByPlatformApplication | Info
☒ ListOriginationNumbers | Info
☐ ListPlatformApplications | Info
☐ ListSMSSandboxPhoneNumbers | Info
☒ ListSubscriptions | Info
☐ ListSubscriptionsByTopic | Info
☒ ListTopics | Info

▼ Read (Selected 3/10)

☐ All read actions

☐ CheckIfPhoneNumberIsOptedOut | Info
☒ GetDataProtectionPolicy | Info
☐ GetEndpointAttributes | Info
☐ GetPlatformApplicationAttributes | Info
☒ GetSMSAttributes | Info
☐ GetSMSSandboxAccountStatus | Info
☐ GetSubscriptionAttributes | Info
☒ GetTopicAttributes | Info
☐ ListPhoneNumbersOptedOut | Info
☐ ListTagsForResource | Info

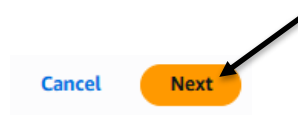
▼ Write (Selected 5/19)

☐ All write actions

☐ ConfirmSubscription | Info
☒ CreateTopic | Info
☐ CreatePlatformApplication | Info
☐ CreatePlatformEndpoint | Info
☐ CreateSMSSandboxPhoneNumber | Info
☐ DeleteEndpoint | Info
☐ DeletePlatformApplication | Info
☒ DeleteTopic | Info
☐ DeleteSMSSandboxPhoneNumber | Info
☒ Publish | Info
☐ PutDataProtectionPolicy | Info
☐ OptInPhoneNumber | Info
☐ SetPlatformApplicationAttributes | Info
☐ SetSMSAttributes | Info
☐ SetEndpointAttributes | Info
☒ Subscribe | Info
☐ SetSubscriptionAttributes | Info
☒ Unsubscribe | Info
☐ VerifySMSSandboxPhoneNumber | Info

► Permissions management (3)

- Click on “Next”.



Step 14:

- In “Review and create”, give a unique name to your policy (e.g. “Policy12”).
- Leave rest other options as it is.
- Click on “Create policy”.

Step 1
Specify permissions

Step 2
Review and create

Review and create [Info](#)

Review the permissions, specify details, and tags.

Policy details

Policy name
Enter a meaningful name to identify this policy.

Policy12

Maximum 128 characters. Use alphanumeric and "+=, @, _" characters.

Description - optional
Add a short explanation for this policy.

Maximum 1,000 characters. Use alphanumeric and "+=, @, _" characters.

Permissions defined in this policy [Info](#)

Permissions defined in this policy document specify which actions are allowed or denied. To define permissions for an IAM identity (user, user group, or role), attach a policy to it

Allow (1 of 446 services) Show remaining 445 services

Service	Access level	Resource	Request condition
SNS	Limited: List, Read, Write	All resources	None

Add tags - optional [Info](#)

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

[Add new tag](#)

You can add up to 50 more tags.

Cancel [Previous](#) **Create policy**

Step 15:

- The policy is created.

Policy Policy12 created. [View policy](#)

Policies (1/1379) [Info](#)

A policy is an object in AWS that defines permissions.

[Filter by Type](#) All types 1 match

Policy name	Type	Used as	Description
Policy12	Customer managed	None	-

- Now we can use this policy the next time we create a new user.

Set permissions

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Permissions options

☐ Add user to group
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

☐ Copy permissions
Copy all group memberships, attached managed policies, and inline policies from an existing user.

☒ **Attach policies directly**
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

Permissions policies (1/1379) [Create policy](#)

Choose one or more policies to attach to your new user.

[Filter by Type](#) All types 1 match

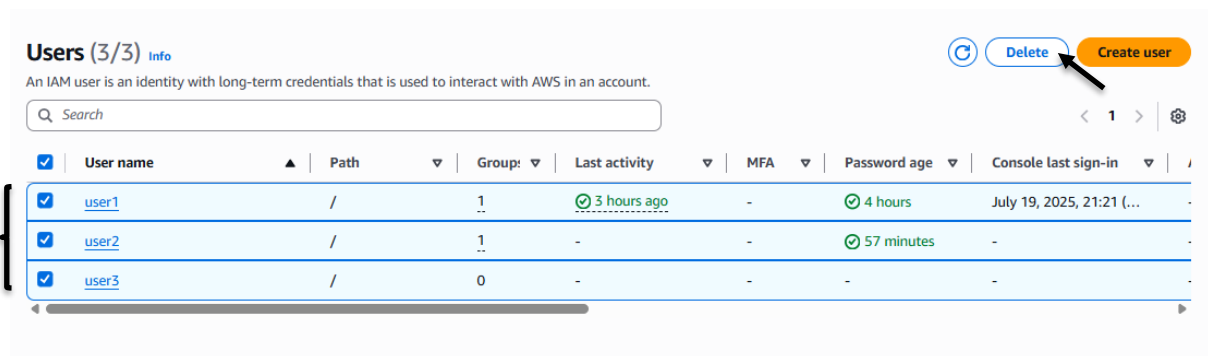
Policy name	Type	Attached entities
<input checked="" type="checkbox"/> Policy12	Customer managed	0

- These policies define what permissions a user have and what actions they can perform.

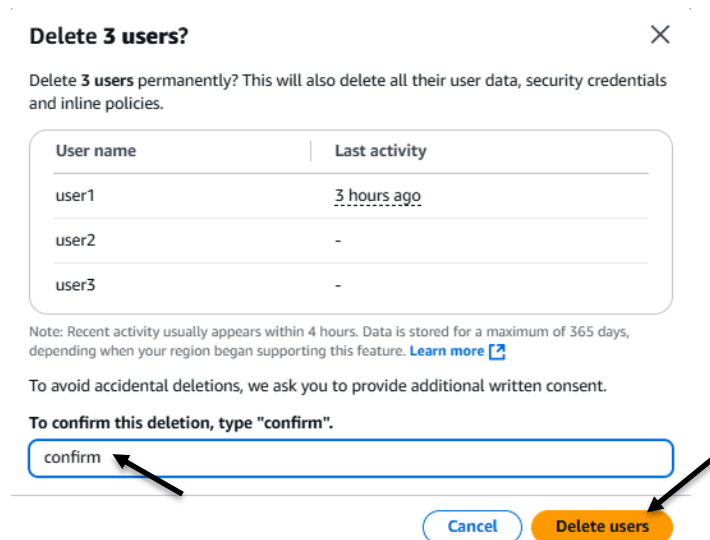
Delete a user:

Step 16:

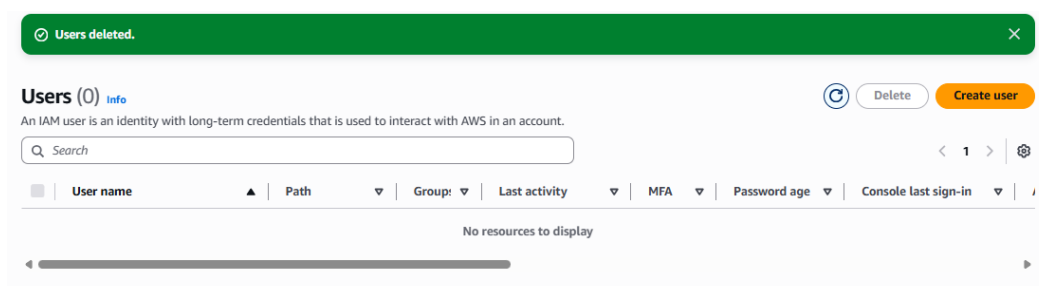
- In “Users”, select all the users that you want to delete.
- Click on “Delete”.



- Type “confirm” and click on “Delete users”.



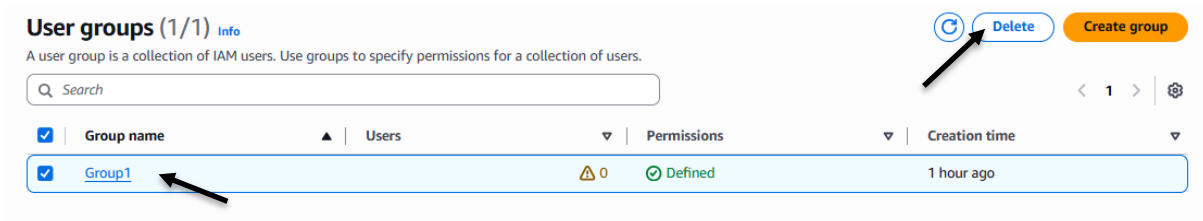
- The users are deleted.



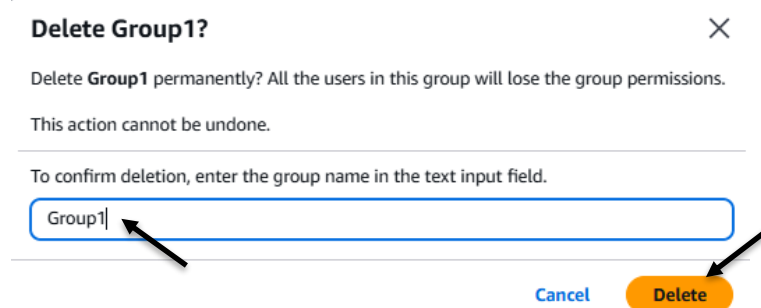
Delete user groups:

Step 17:

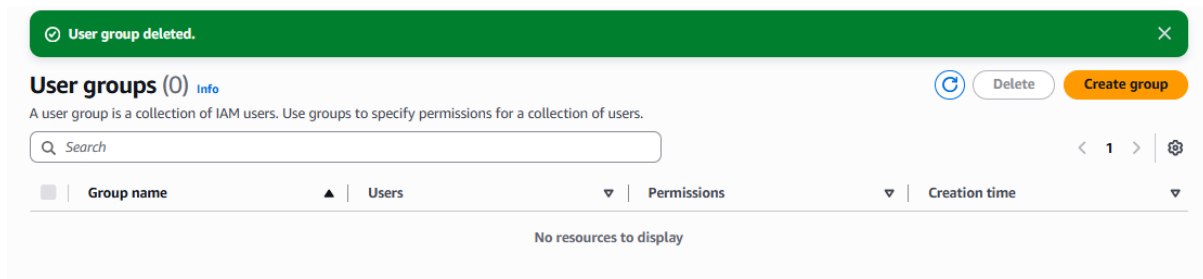
- In “User groups”, select the group that you want to delete.
- Click on “Delete”.



- Type “Group1” and click on “Delete”.



- The user group is also deleted.



Delete a policy:

Step 18:

- Go to “Policies” and search the policy you created.
- Select your policy and click on “Delete”.
- It doesn’t matter even if your policy is not deleted.

Policies (1/1379) [Info](#)

A policy is an object in AWS that defines permissions.

[Actions](#)[Delete](#)[Create policy](#)**Filter by Type**[All types](#)

1 match

< 1 >

Policy name	Type	Used as	Description
Policy12	Customer managed	None	-

- Type "Policy12" and click on "Delete".

Delete Policy12?



Delete **Policy12** policy and all its versions permanently?

This action cannot be undone.

To confirm deletion, enter the policy name in the text input field.

[Cancel](#)[Delete](#)

- The policy is deleted.

✓ Policy deleted.



Policies (1378) [Info](#)

A policy is an object in AWS that defines permissions.

[Actions](#)[Delete](#)[Create policy](#)**Filter by Type**[All types](#)

0 matches

< 1 >

Policy name	Type	Used as	Description
-------------	------	---------	-------------

No matches

[Clear filters](#)