IAM (Identity and Access Management)

What is IAM?

- AWS Identity and Access Management (IAM) is a web service that helps you securely control access to AWS resources.
- With IAM, you can manage permissions that control which AWS resources users can access.
- You use IAM to control who is authenticated (signed in) and authorized (has permissions) to use resources such as ec2 instances, s3 bucket, and DBs.
- IAM provides the infrastructure necessary to control authentication and authorization for your AWS accounts.
- IAM enables you to create and manage users, groups, and roles, each with their own permission and access policies.

Identities:

- 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.
- Use IAM to set up other identities in addition to your root user, such as administrators, analysts, and developers, and grant them access to the resources they need to succeed in their tasks.
- Authentication is the process of verifying your identity. You need to provide your login credentials to get authenticated to AWS console.

Access management:

- After a user is set up in IAM, they use their sign-in credentials to authenticate with AWS.
- Once you are authenticated, authorization determines what actions you're allowed to perform or what resources you're allowed to access after your identity has been verified.

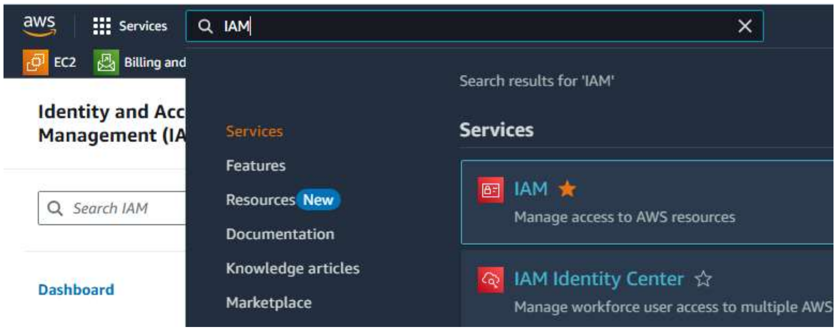
Why use IAM?

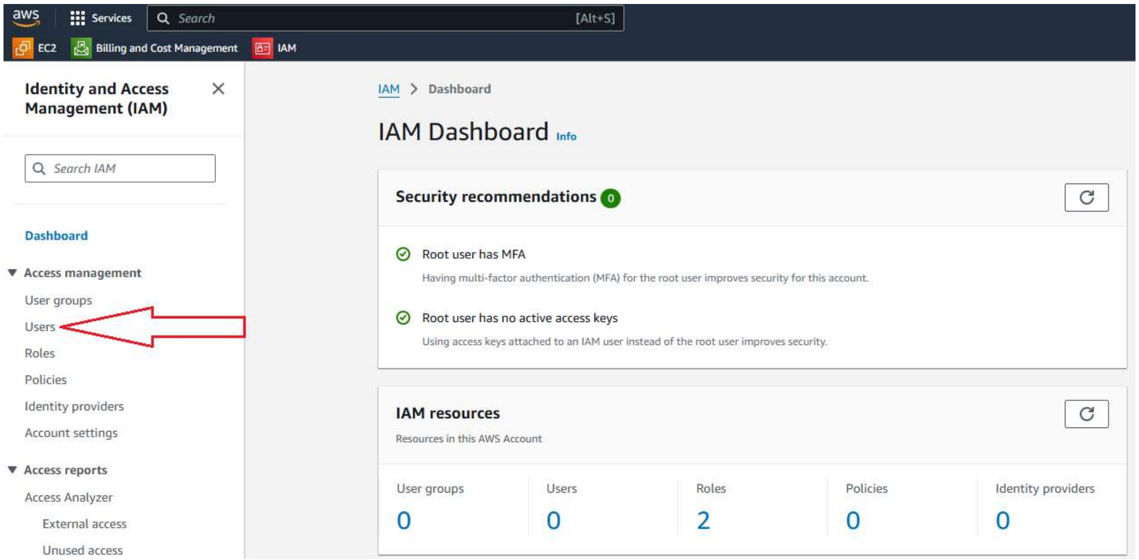
- Use AWS Identity and Access Management (IAM) to manage and scale workload and workforce access securely supporting your agility and innovation in AWS.
- It is not a good practice to perform all the activities by using root user. So, we should create IAM user.

IAM Groups:

In AWS IAM, "groups" are collections of IAM users. Groups allow you to manage permissions for multiple users collectively, rather than individually assigning permissions for each individual user.

Login to AWS console as root user. Click on Services, and search for IAM. Click on IAM.





IAM Dashboard Info

Security recommendations 0

✓ Root user has MFA

Having multi-factor authentication (MFA) for the root user improves security for this account.

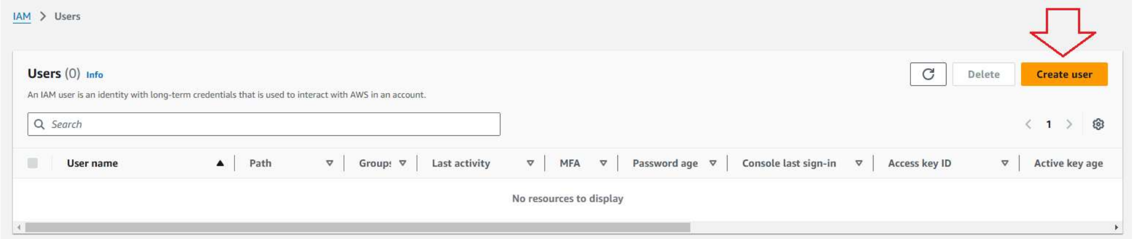
✓ Root user has no active access keys

Using access keys attached to an IAM user instead of the root user improves security.

IAM resources

Resources in this AWS Account

User groups	Users	Roles	Policies	Identity providers
0	0	2	0	0

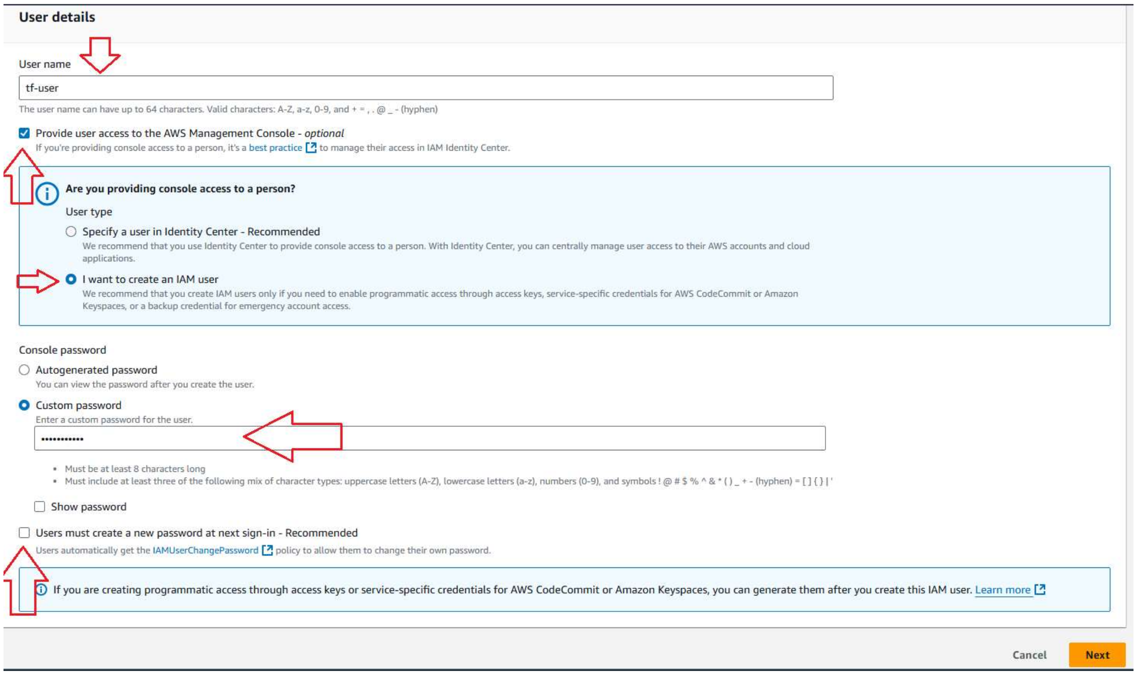


Users (0) Info

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

User name	Path	Groups	Last activity	MFA	Password age	Console last sign-in	Access key ID	Active key age

No resources to display


User details

User name

tf-user

The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = , . @ _ - (hyphen)

☑ 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.

•••••••••••

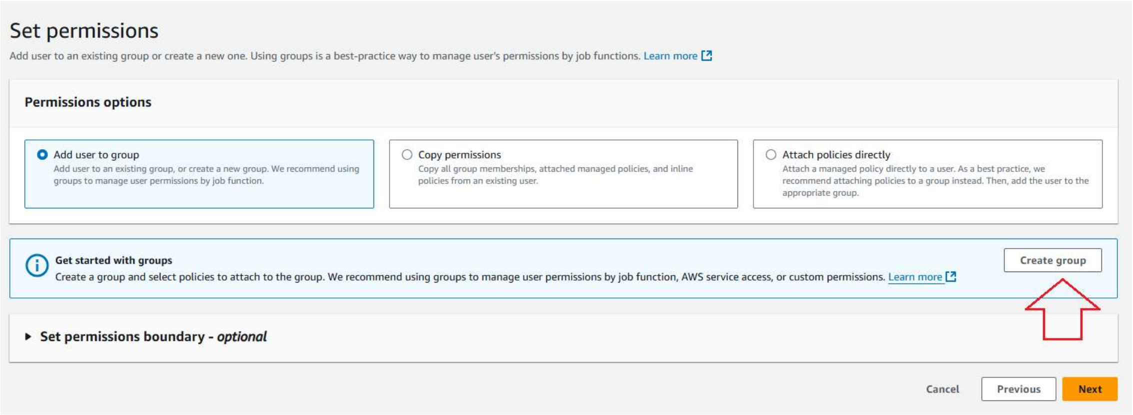
- Must be at least 8 characters long
- Must include at least three of the following mix of character types: uppercase letters (A-Z), lowercase letters (a-z), numbers (0-9), and symbols ! @ # $ % ^ & * ( ) _ + - (hyphen) = [ ] { } | '

□ 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



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

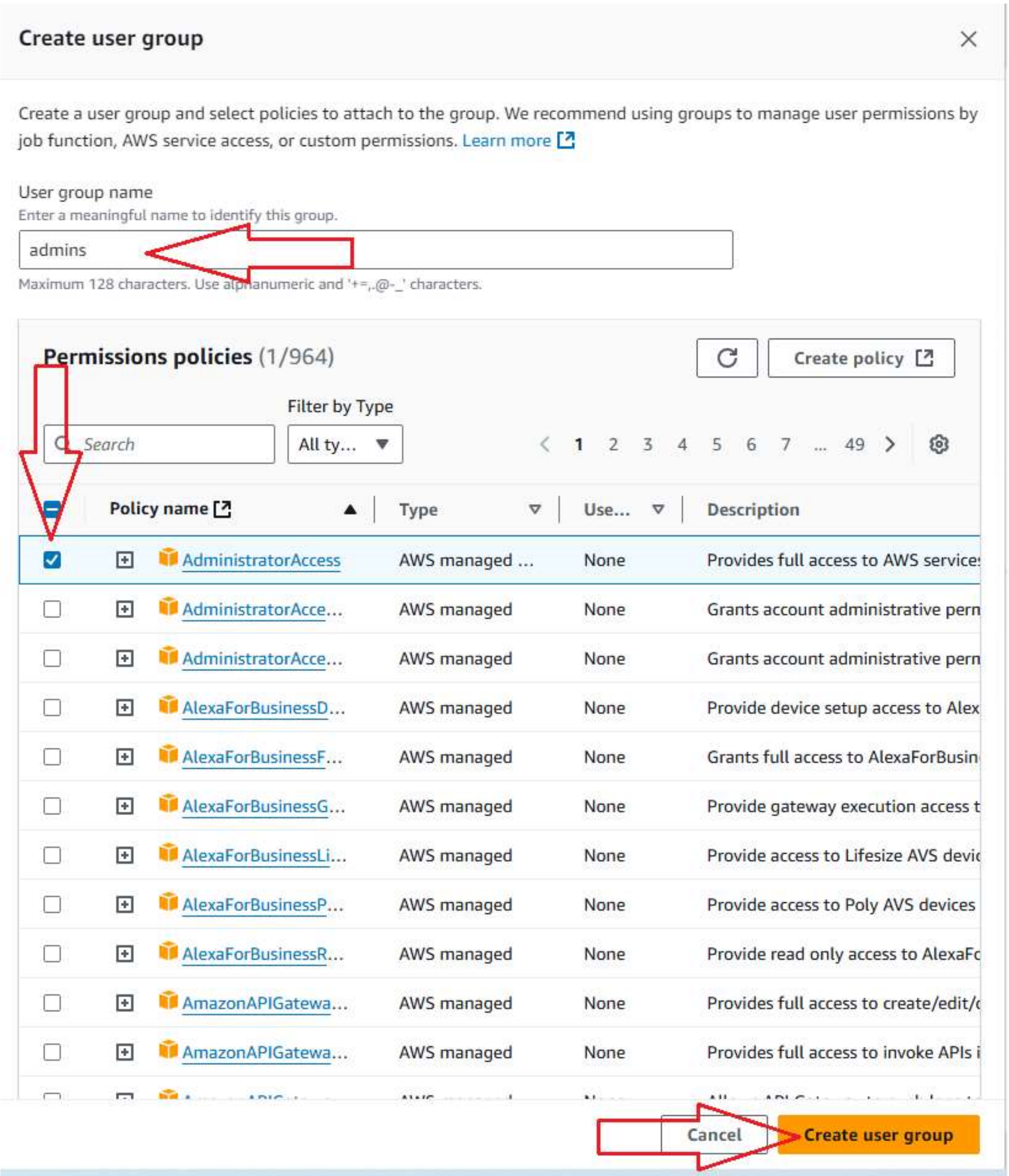
Permissions options

● 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.

Get started with groups
Create a group and select policies to attach to the group. We recommend using groups to manage user permissions by job function, AWS service access, or custom permissions. Learn more



Create user group

Create a user group and select policies to attach to the group. We recommend using groups to manage user permissions by job function, AWS service access, or custom permissions. Learn more

User group name
Enter a meaningful name to identify this group.

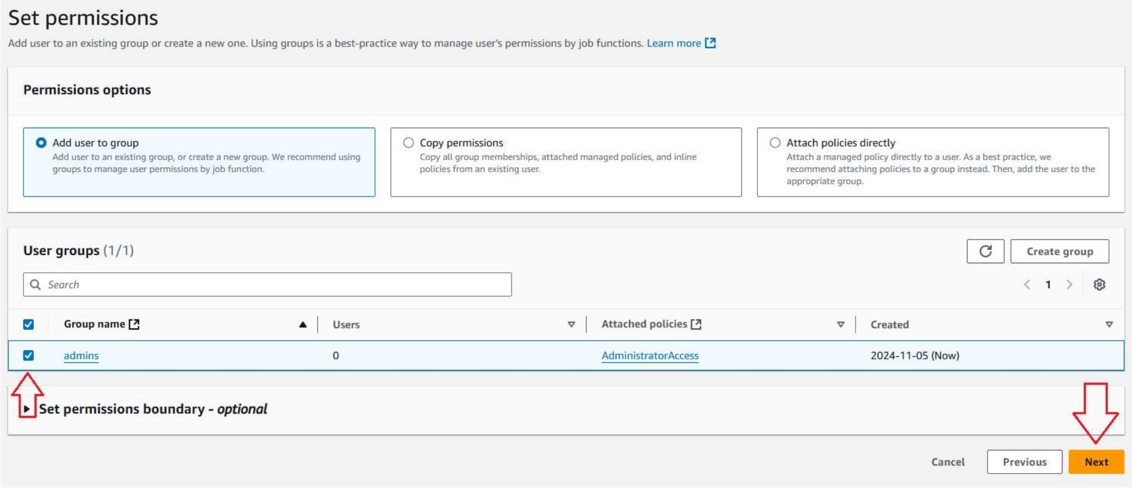
admins

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

Permissions policies (1/964)

Filter by Type: All ty...

		Policy name	Type	Use...	Description
☑	⊞	AdministratorAccess	AWS managed ...	None	Provides full access to AWS service
□	⊞	AdministratorAcce...	AWS managed	None	Grants account administrative pern
□	⊞	AdministratorAcce...	AWS managed	None	Grants account administrative pern
□	⊞	AlexaForBusinessD...	AWS managed	None	Provide device setup access to Alex
□	⊞	AlexaForBusinessF...	AWS managed	None	Grants full access to AlexaForBusin
□	⊞	AlexaForBusinessG...	AWS managed	None	Provide gateway execution access t
□	⊞	AlexaForBusinessLi...	AWS managed	None	Provide access to Lifesize AVS devic
□	⊞	AlexaForBusinessP...	AWS managed	None	Provide access to Poly AVS devices
□	⊞	AlexaForBusinessR...	AWS managed	None	Provide read only access to AlexaFc
□	⊞	AmazonAPIGatewa...	AWS managed	None	Provides full access to create/edit/c
□	⊞	AmazonAPIGatewa...	AWS managed	None	Provides full access to invoke APIs i



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.

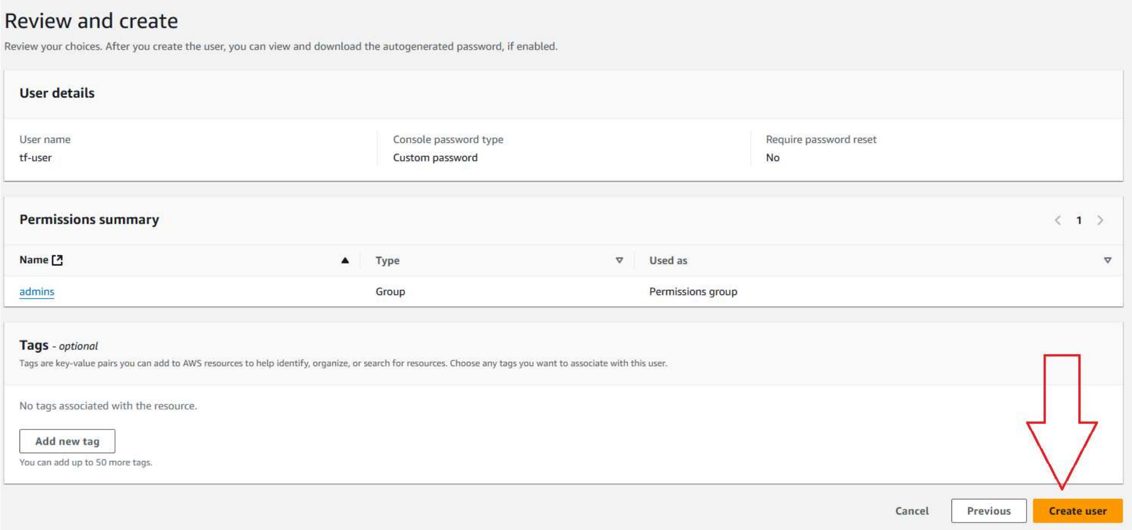
○ Copy permissions
Copy all group memberships, attached managed policies, and inline policies from an existing user.

○ 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)

☑	Group name	Users	Attached policies	Created
☑	admins	0	AdministratorAccess	2024-11-05 (Now)

▸ **Set permissions boundary - *optional***



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	Console password type	Require password reset
tf-user	Custom password	No

Permissions summary

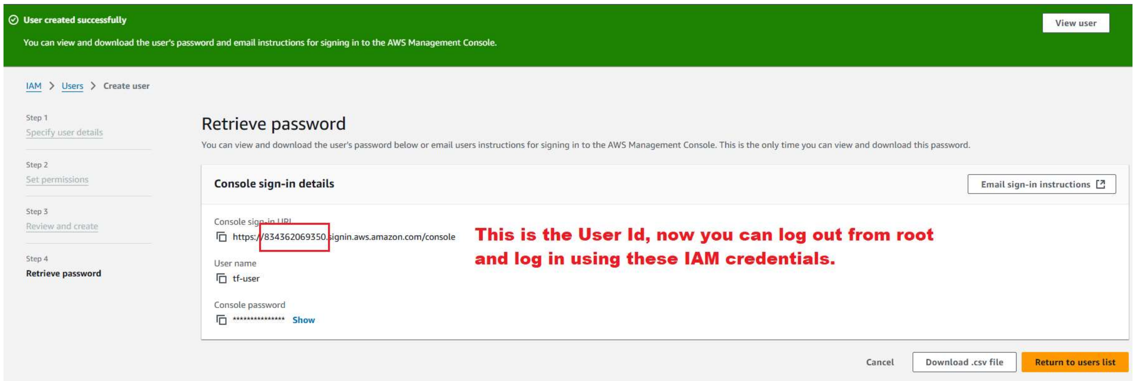
Name	Type	Used as
admins	Group	Permissions group

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.

You can add up to 50 more tags.



User created successfully

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

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.

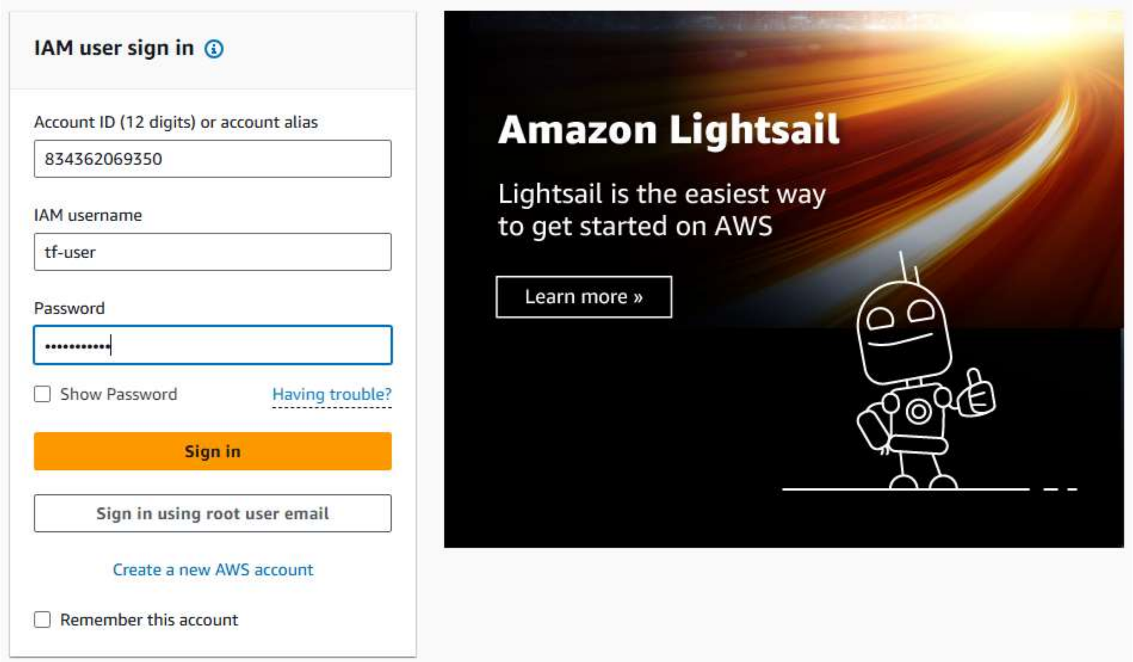
Console sign-in details

Console sign-in URL
https://834362069350.signin.aws.amazon.com/console

User name
tf-user

Console password
*************** Show

This is the User Id, now you can log out from root and log in using these IAM credentials.



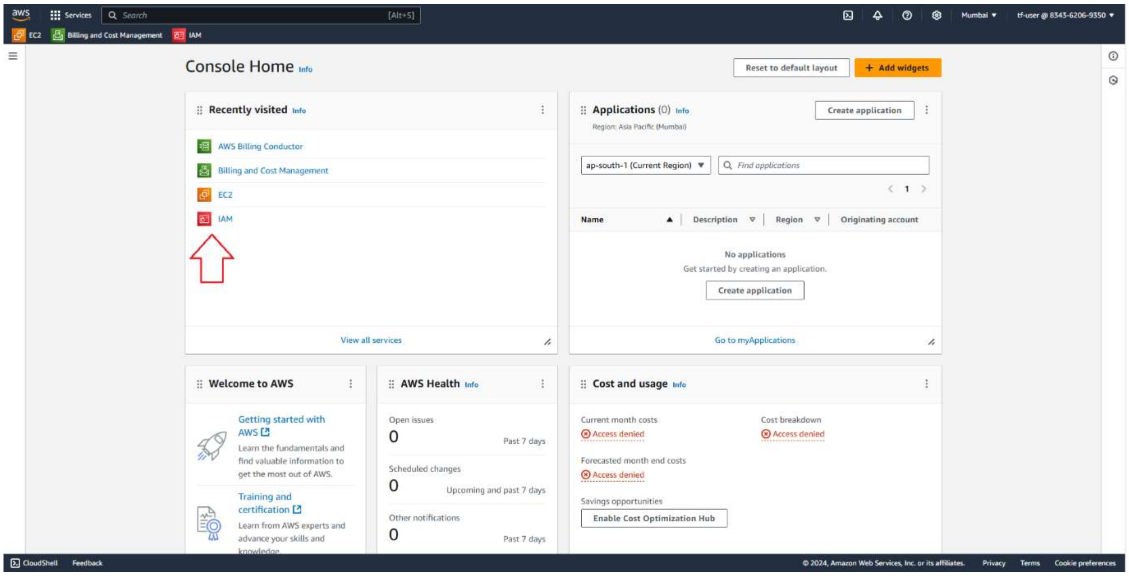
IAM user sign in

Account ID (12 digits) or account alias: 834362069350

IAM username: tf-user

Password: ••••••••••••

□ Show Password



Console Home Info

Recently visited Info

- AWS Billing Conductor
- Billing and Cost Management
- EC2
- IAM

Applications (0) Info

Region: Asia Pacific (Mumbai)

ap-south-1 (Current Region) ▾ | Find applications

Name ▲	Description ▽	Region ▽	Originating account

No applications

Get started by creating an application.

Welcome to AWS

Getting started with AWS
Learn the fundamentals and find valuable information to get the most out of AWS.

Training and certification
Learn from AWS experts and advance your skills and knowledge.

AWS Health Info

Open issues
0 Past 7 days

Scheduled changes
0 Upcoming and past 7 days

Other notifications
0 Past 7 days

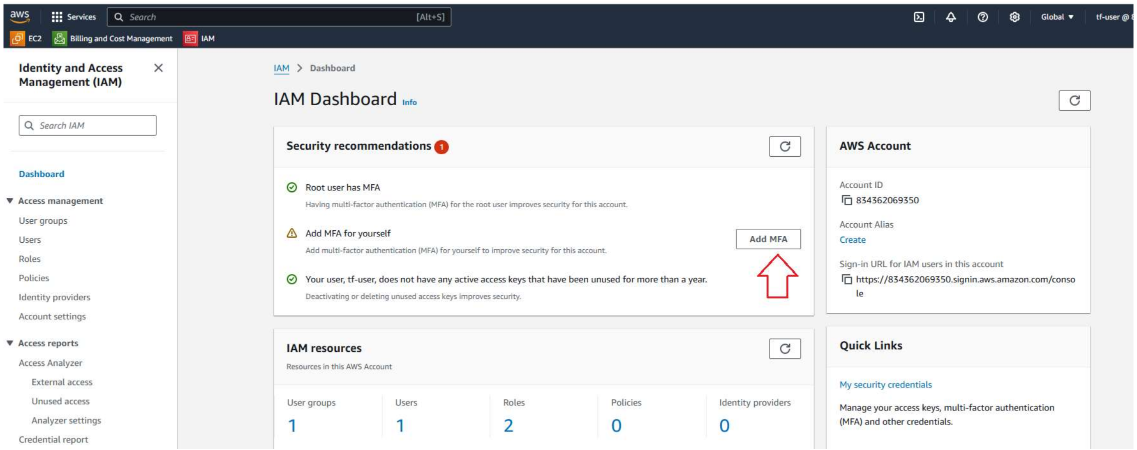
Cost and usage Info

Current month costs: Access denied

Cost breakdown: Access denied

Forecasted month end costs: Access denied

Savings opportunities



IAM > Dashboard

IAM Dashboard Info

Security recommendations 1

Root user has MFA
Having multi-factor authentication (MFA) for the root user improves security for this account.

Add MFA for yourself
Add multi-factor authentication (MFA) for yourself to improve security for this account.

Your user, tf-user, does not have any active access keys that have been unused for more than a year.
Deactivating or deleting unused access keys improves security.

AWS Account

Account ID: 834362069350

Account Alias: Create

Sign-in URL for IAM users in this account: https://834362069350.signin.aws.amazon.com/console

IAM resources

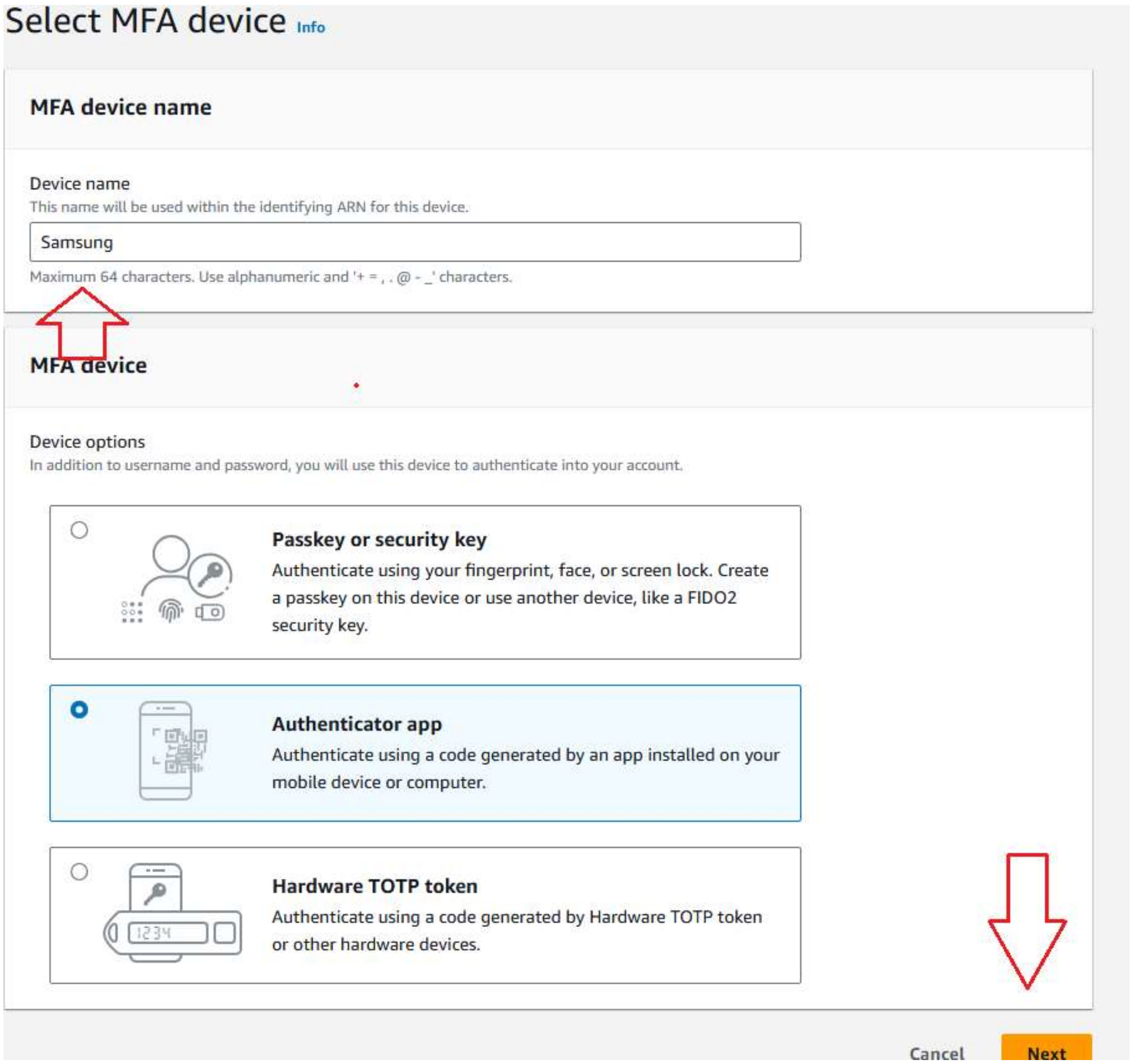
Resources in this AWS Account

User groups	Users	Roles	Policies	Identity providers
1	1	2	0	0

Quick Links

My security credentials

Manage your access keys, multi-factor authentication (MFA) and other credentials.



Select MFA device

MFA device name

Device name

This name will be used within the identifying ARN for this device.

Samsung

Maximum 64 characters. Use alphanumeric and '+ = , . @ - _' characters.

MFA device

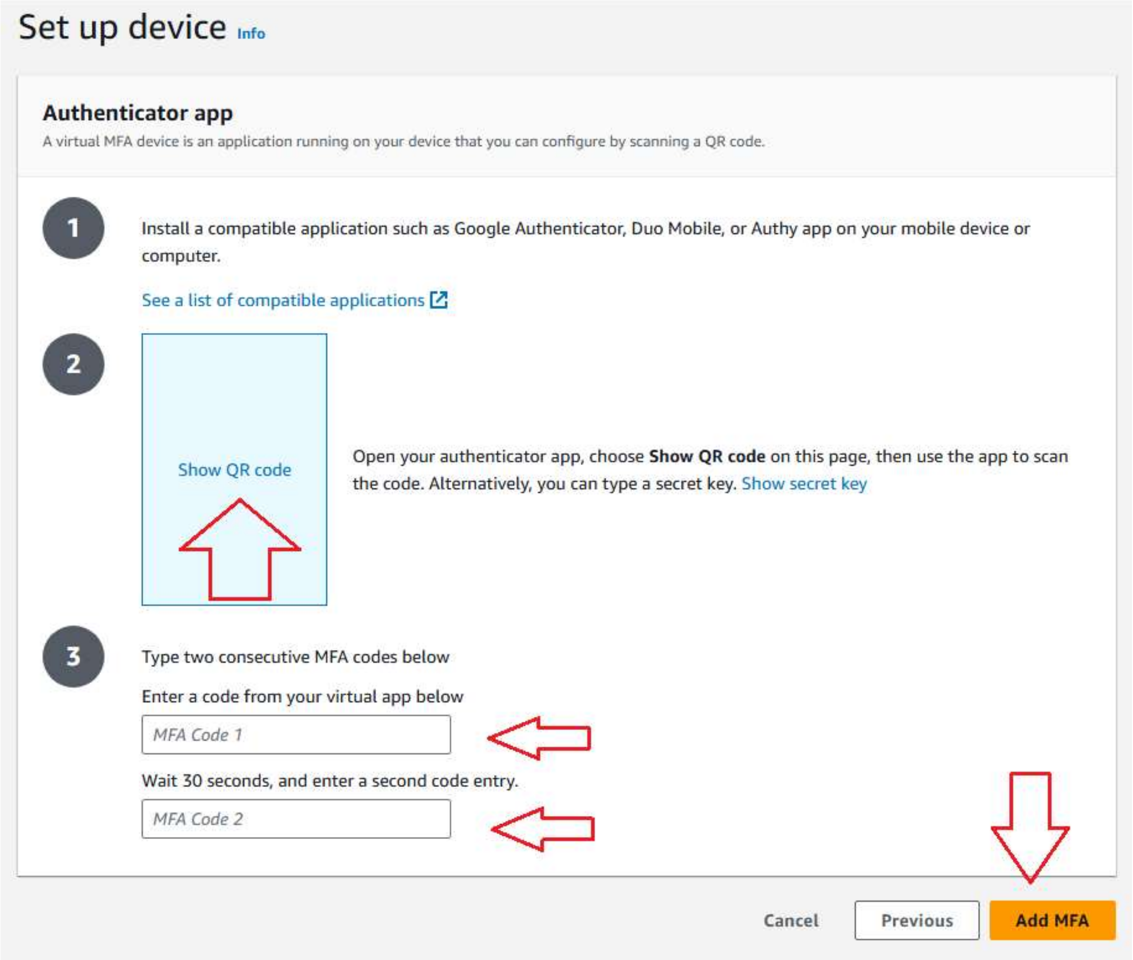
Device options

In addition to username and password, you will use this device to authenticate into your account.

○ **Passkey or security key**
Authenticate using your fingerprint, face, or screen lock. Create a passkey on this device or use another device, like a FIDO2 security key.

● **Authenticator app**
Authenticate using a code generated by an app installed on your mobile device or computer.

○ **Hardware TOTP token**
Authenticate using a code generated by Hardware TOTP token or other hardware devices.



Set up device Info

Authenticator app

A virtual MFA device is an application running on your device that you can configure by scanning a QR code.

1 Install a compatible application such as Google Authenticator, Duo Mobile, or Authy app on your mobile device or computer.

See a list of compatible applications

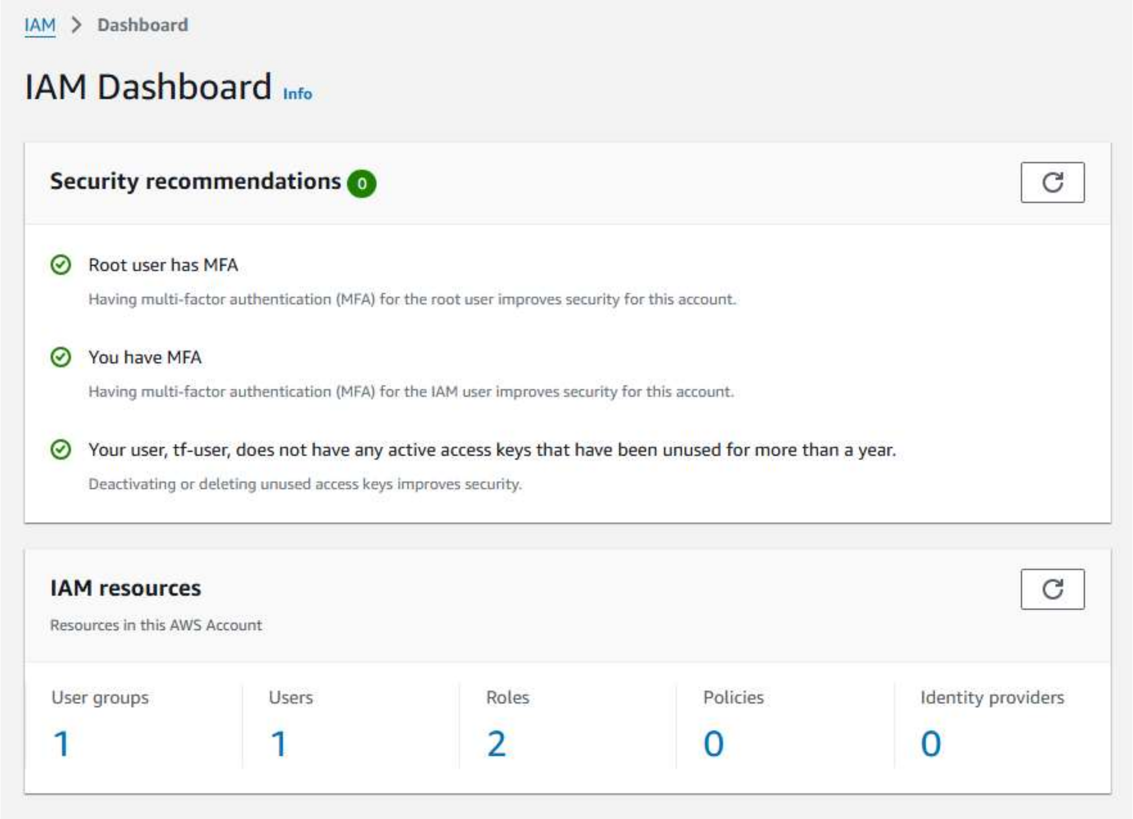
2 Open your authenticator app, choose **Show QR code** on this page, then use the app to scan the code. Alternatively, you can type a secret key. Show secret key

3 Type two consecutive MFA codes below

Enter a code from your virtual app below

Wait 30 seconds, and enter a second code entry.

Now come to IAM Dashboard, it should look like:



IAM > Dashboard

IAM Dashboard Info

Security recommendations 0

✓ Root user has MFA

Having multi-factor authentication (MFA) for the root user improves security for this account.

✓ You have MFA

Having multi-factor authentication (MFA) for the IAM user improves security for this account.

✓ Your user, tf-user, does not have any active access keys that have been unused for more than a year.

Deactivating or deleting unused access keys improves security.

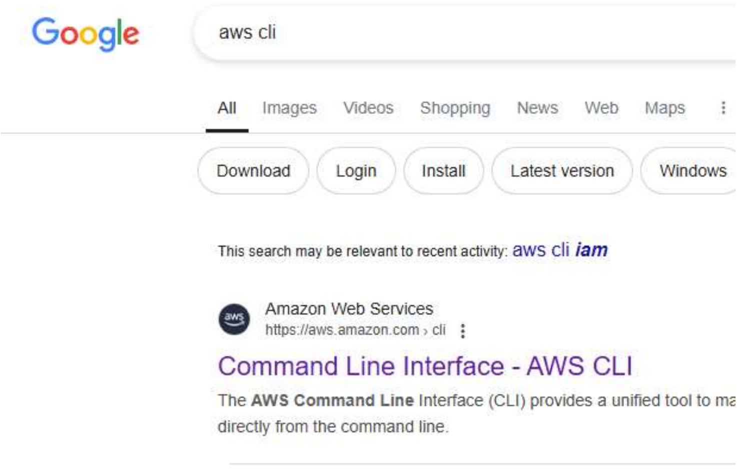
IAM resources

Resources in this AWS Account

User groups	Users	Roles	Policies	Identity providers
1	1	2	0	0

AWS CLI (AWS Command Line Interface)

- The AWS Command Line Interface (AWS CLI) is a unified tool to manage your AWS services. With just one tool to download and configure, you can control multiple AWS services from the command line and automate them through scripts.
- The AWS CLI v2 offers several new features including improved installers, new configuration options such as AWS IAM Identity Center (successor to AWS SSO), and various interactive features.
- Download AWS CLI and install by following these steps:

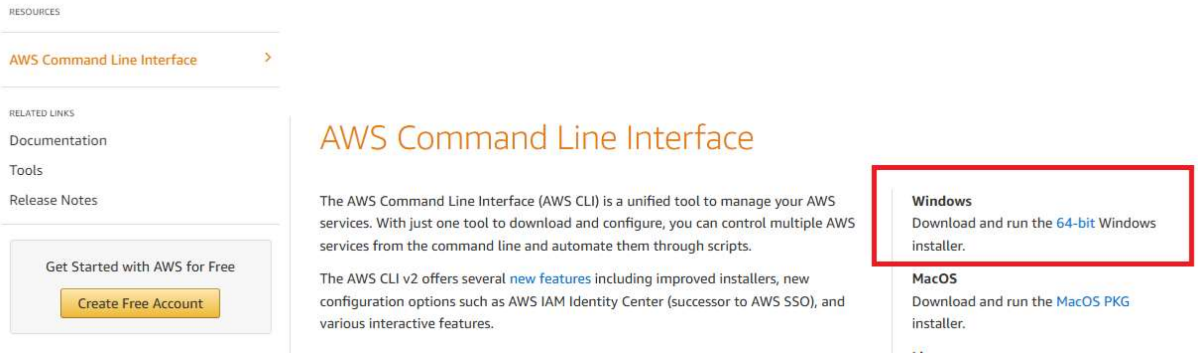


This search may be relevant to recent activity: aws cli *iam*

Amazon Web Services
https://aws.amazon.com › cli

Command Line Interface - AWS CLI

The **AWS Command Line** Interface (CLI) provides a unified tool to ma
directly from the command line.



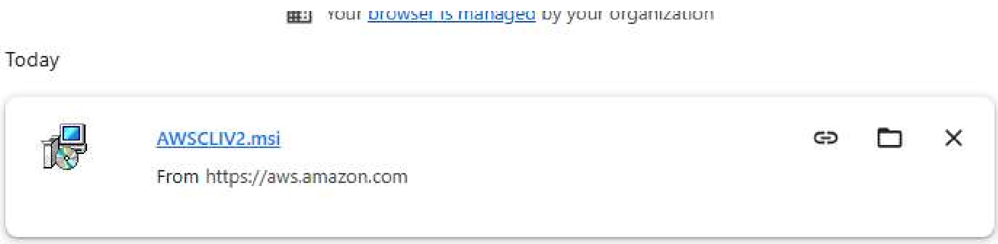
AWS Command Line Interface

The AWS Command Line Interface (AWS CLI) is a unified tool to manage your AWS services. With just one tool to download and configure, you can control multiple AWS services from the command line and automate them through scripts.

The AWS CLI v2 offers several new features including improved installers, new configuration options such as AWS IAM Identity Center (successor to AWS SSO), and various interactive features.

Windows
Download and run the 64-bit Windows installer.

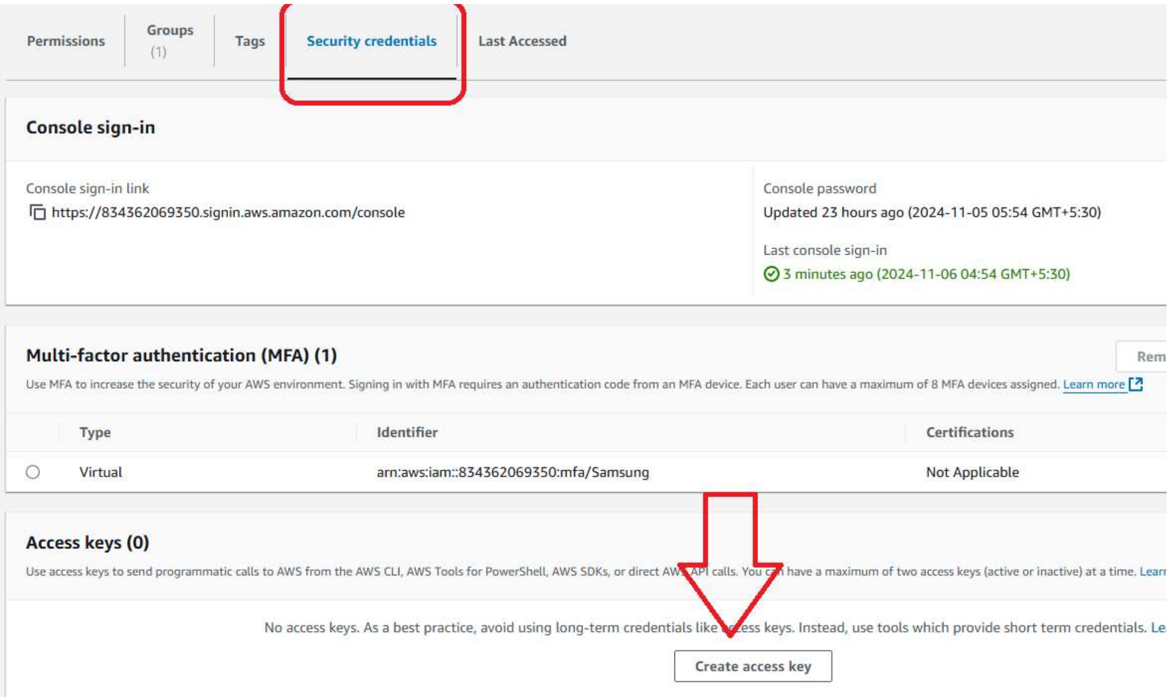
MacOS
Download and run the MacOS PKG installer.



AWSCLIV2.msi
From https://aws.amazon.com

AWS Access Key

- Do the setup for VS Code.
- Login to AWS console with the IAM user created.
- Go to IAM Dashboard.
- Go to Users.
- Click on tf-user which we created previously.
- Click on Security-Credentials.
- Scroll down and click on Create Access Key.



Console sign-in

Console sign-in link

https://834362069350.signin.aws.amazon.com/console

Console password

Updated 23 hours ago (2024-11-05 05:54 GMT+5:30)

Last console sign-in

✓ 3 minutes ago (2024-11-06 04:54 GMT+5:30)

Multi-factor authentication (MFA) (1)

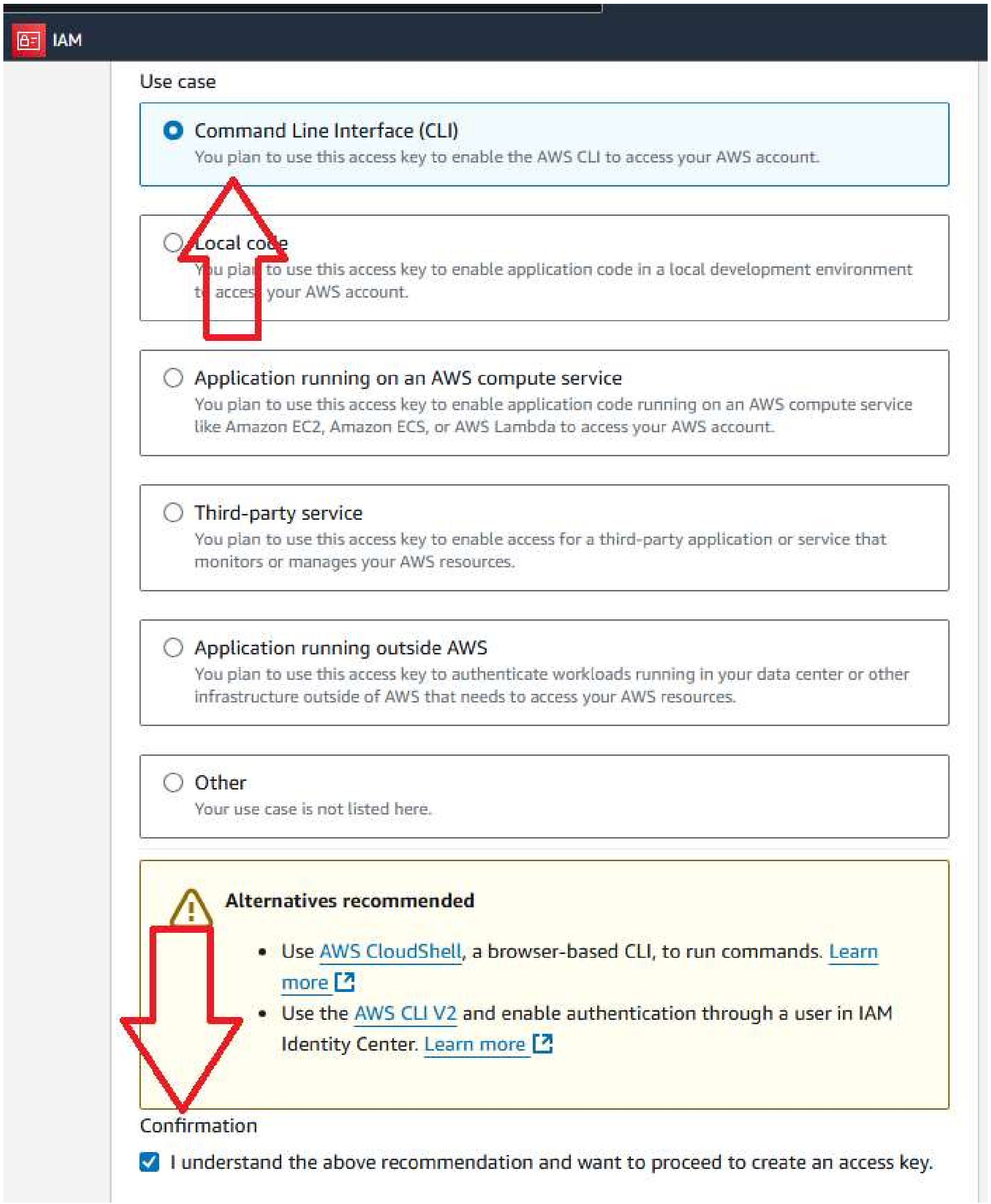
Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. Learn more

	Type	Identifier	Certifications
○	Virtual	arn:aws:iam::834362069350:mfa/Samsung	Not Applicable

Access keys (0)

Use access keys to send programmatic calls to AWS from the AWS CLI, AWS Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time. Learn

No access keys. As a best practice, avoid using long-term credentials like access keys. Instead, use tools which provide short term credentials. Le



Use case

● Command Line Interface (CLI)
You plan to use this access key to enable the AWS CLI to access your AWS account.

○ Local code
You plan to use this access key to enable application code in a local development environment to access your AWS account.

○ Application running on an AWS compute service
You plan to use this access key to enable application code running on an AWS compute service like Amazon EC2, Amazon ECS, or AWS Lambda to access your AWS account.

○ Third-party service
You plan to use this access key to enable access for a third-party application or service that monitors or manages your AWS resources.

○ Application running outside AWS
You plan to use this access key to authenticate workloads running in your data center or other infrastructure outside of AWS that needs to access your AWS resources.

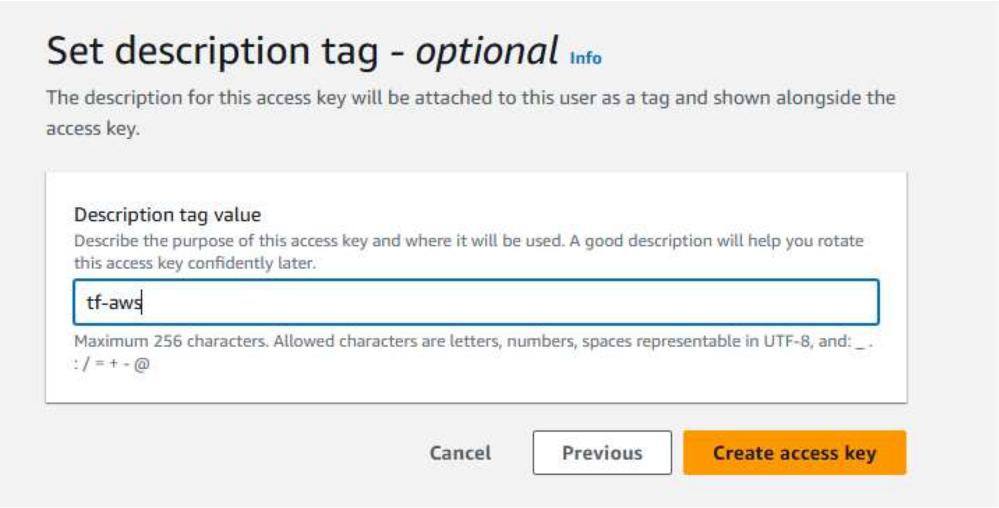
○ Other
Your use case is not listed here.

Alternatives recommended

- Use AWS CloudShell, a browser-based CLI, to run commands. Learn more
- Use the AWS CLI V2 and enable authentication through a user in IAM Identity Center. Learn more

Confirmation

☑ I understand the above recommendation and want to proceed to create an access key.



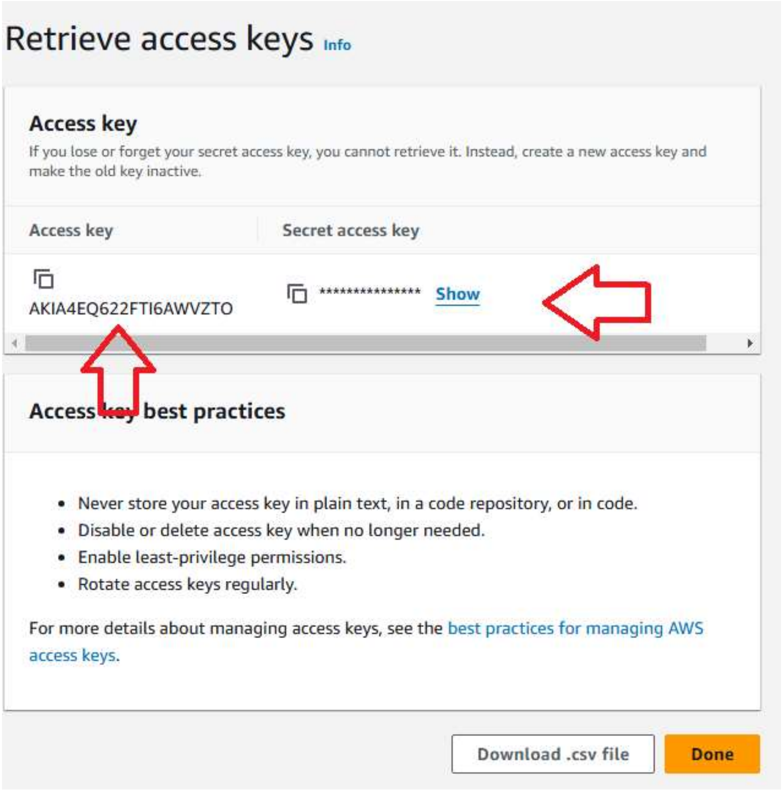
Set description tag - *optional* Info

The description for this access key will be attached to this user as a tag and shown alongside the access key.

Description tag value
Describe the purpose of this access key and where it will be used. A good description will help you rotate this access key confidently later.

tf-aws

Maximum 256 characters. Allowed characters are letters, numbers, spaces representable in UTF-8, and: _ . : / = + - @



Retrieve access keys Info

Access key

If you lose or forget your secret access key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.

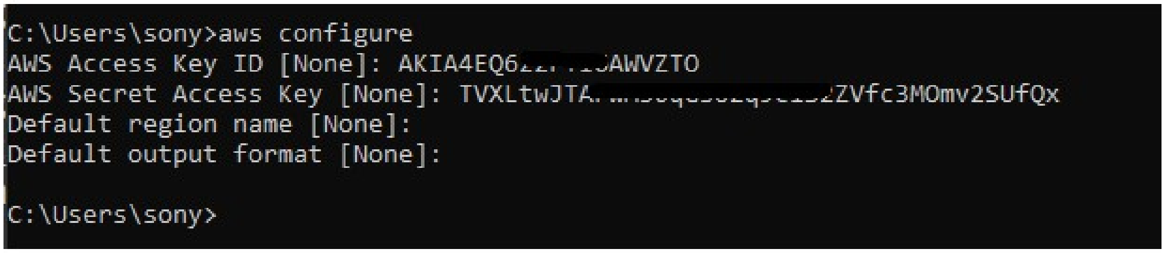
Access key	Secret access key
AKIA4EQ622FTI6AWVZTO	*************** Show

Access key best practices

- Never store your access key in plain text, in a code repository, or in code.
- Disable or delete access key when no longer needed.
- Enable least-privilege permissions.
- Rotate access keys regularly.

For more details about managing access keys, see the best practices for managing AWS access keys.

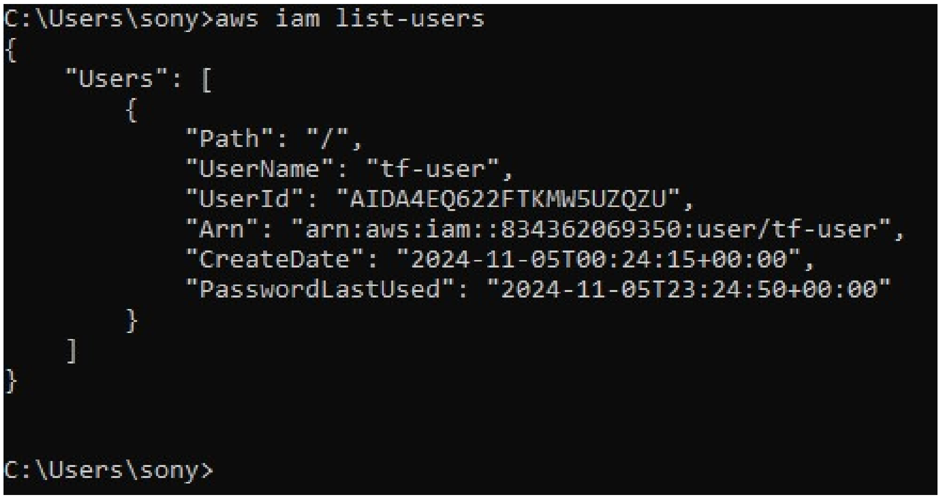
Open a command prompt and enter **aws configure**



```
C:\Users\sony>aws configure
AWS Access Key ID [None]: AKIA4EQ6[redacted]AWVZTO
AWS Secret Access Key [None]: TVXLtwJTA[redacted]ZVfc3MOmv2SUfQx
Default region name [None]:
Default output format [None]:

C:\Users\sony>
```

Close the Command Prompt, open a new one and type: **aws iam list-users**. You can get following output.



```
C:\Users\sony>aws iam list-users
{
    "Users": [
        {
            "Path": "/",
            "UserName": "tf-user",
            "UserId": "AIDA4EQ622FTKMW5UZQZU",
            "Arn": "arn:aws:iam::834362069350:user/tf-user",
            "CreateDate": "2024-11-05T00:24:15+00:00",
            "PasswordLastUsed": "2024-11-05T23:24:50+00:00"
        }
    ]
}


C:\Users\sony>
```