Published in Cloudnloud Tech Community

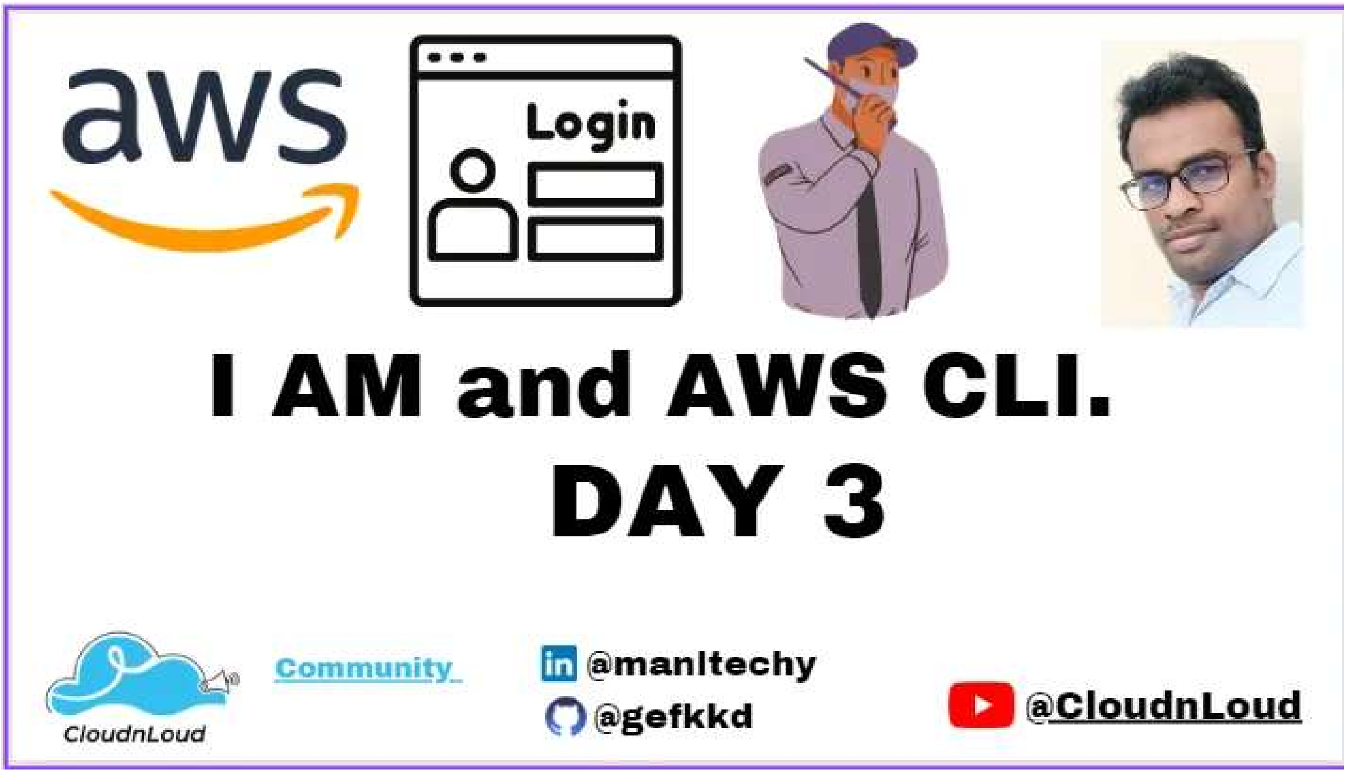
Manikanta Suru

Mar 24 · 6 min read · Listen

Save

📍 “From Zero to Cloud Hero: Day 3 of the 100-Day Journey to Mastering Cloud Computing”

I AM and AWS CLI.



IAM Section :

Users & Groups

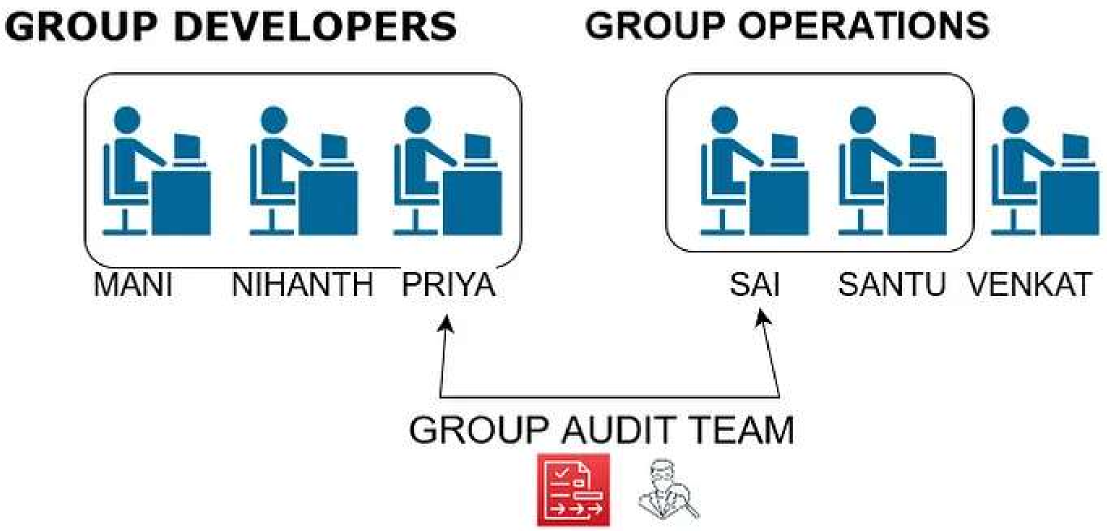
- IAM = Identity and Access Management, Global service

• Root account created by default, shouldn't be used or shared

• Users are people within your organization and can be grouped

• Groups only contain users, not other groups

- Users don't have to belong to a group, and users can belong to multiple groups.

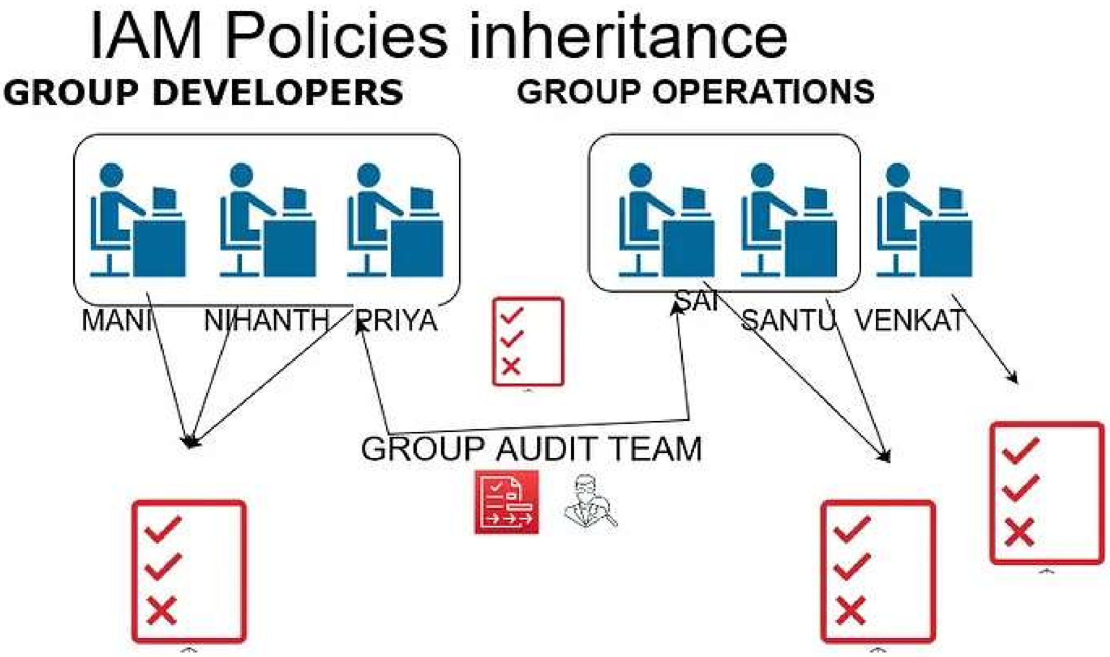


IAM: Permissions :

Users or Groups can be assigned JSON documents called policies
• These policies define the permissions of the users
• In AWS you apply the least privilege principle: don't give more permissions than a user needs

IAM Policies inheritance :

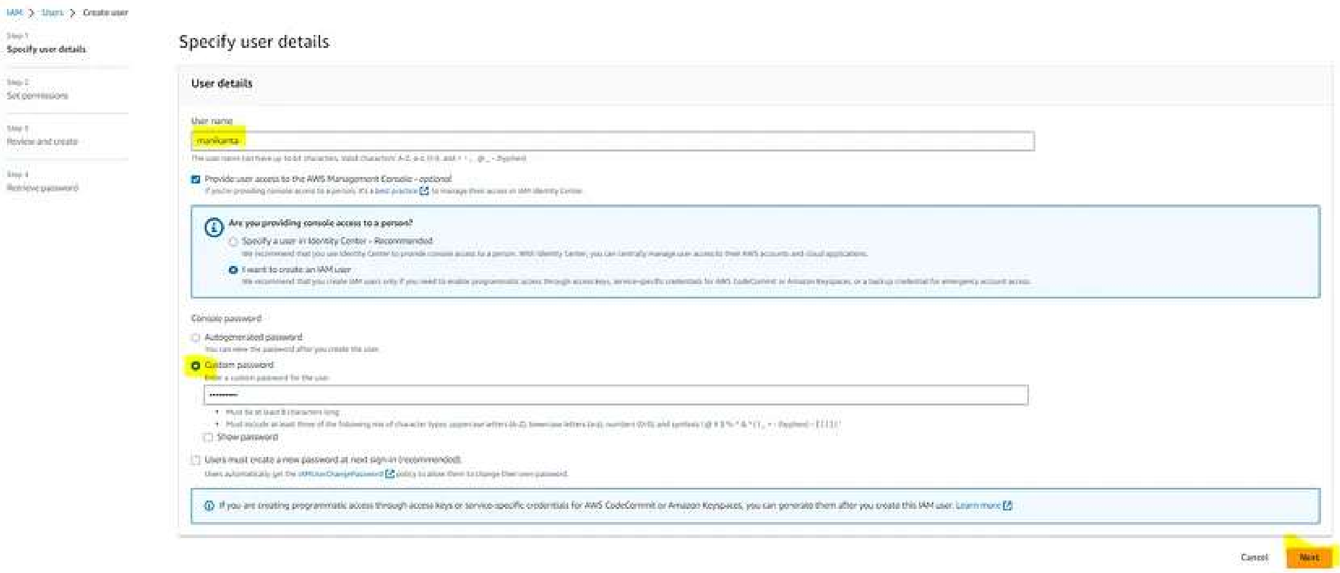
• Users or Groups can be assigned JSON documents called policies
• These policies define the permissions of the users
• In AWS you apply the least privilege principle: don't give more permissions than a user needs



IAM Users & Groups Hands-On :

Go to https://us-east-1.console.aws.amazon.com/iamv2/home#/home

Add new I am a user



Specify user details

User details

User name

manikanta

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?

○ 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.

••••••••••

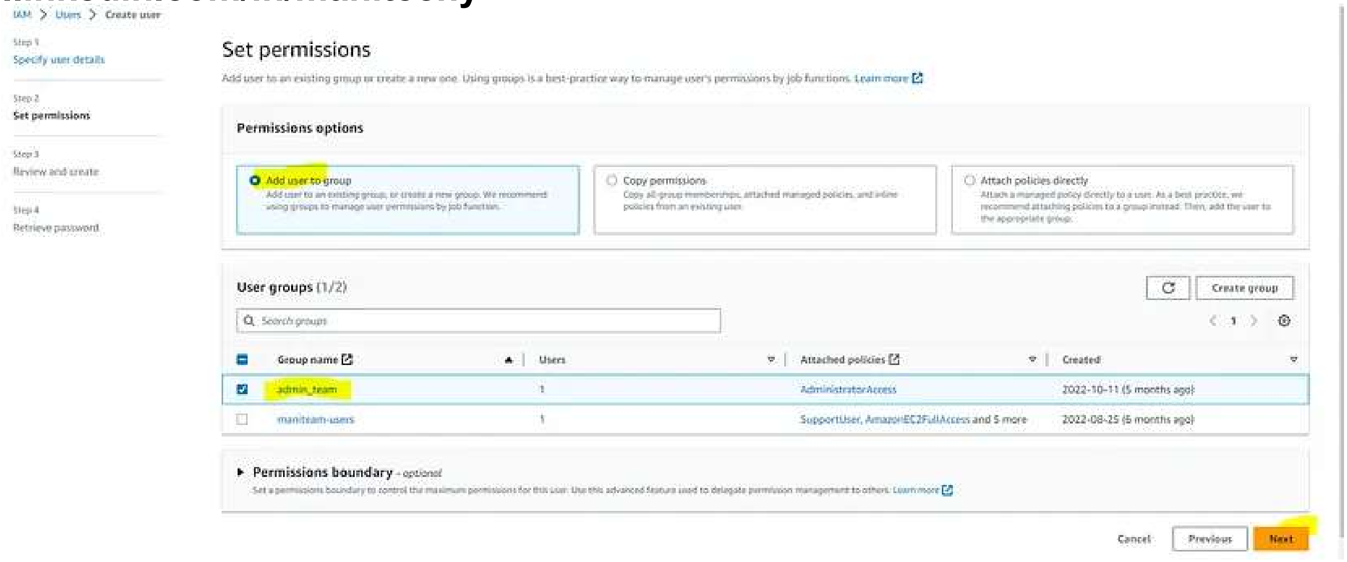
- Must be at least 8 characters long
- Must include at least three of the following mix of character types: uppercase letters (A-Z), lowercase letters (a-z), numbers (0-9), and symbols ! @ # $ % ^ & * ( ) _ + - (hyphen) = [ ] { } | '

□ 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



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.

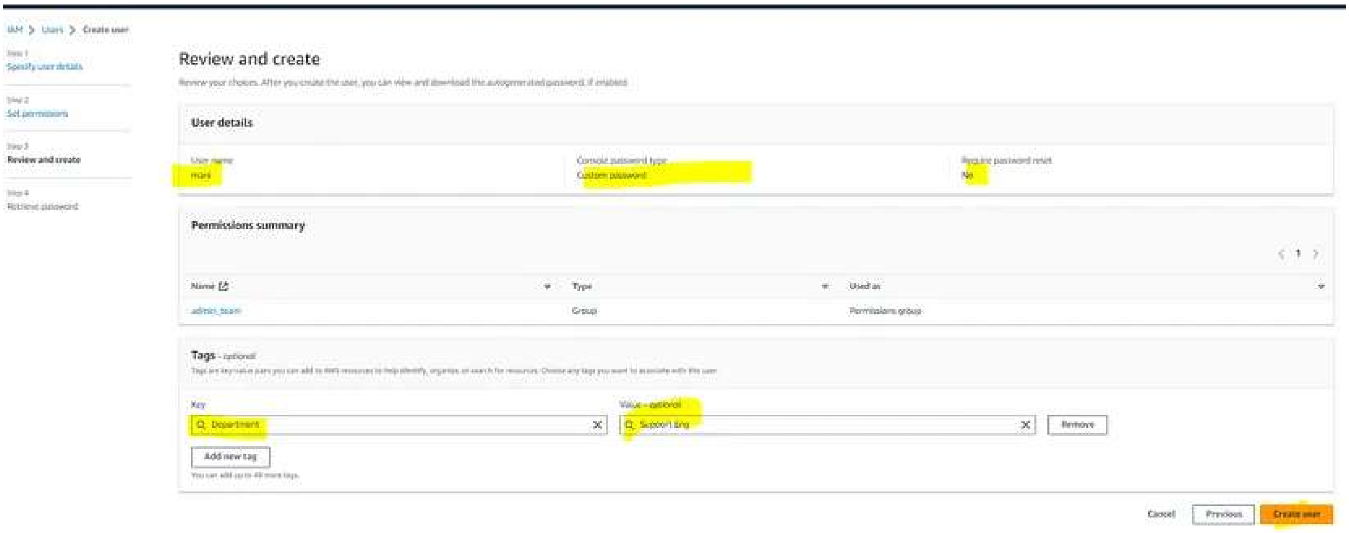
○ Attach policies directly
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

User groups (1/2)

Search groups

	Group name	Users	Attached policies	Created
☑	admin_team	1	AdministratorAccess	2022-10-11 (5 months ago)
□	maniteam-users	1	SupportUser, AmazonEC2FullAccess and 5 more	2022-08-25 (6 months ago)

▸ **Permissions boundary** - optional
Set a permissions boundary to control the maximum permissions for this user. Use this advanced feature used to delegate permission management to others. Learn more



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	Console password type	Require password reset
mani	Custom password	No

Permissions summary

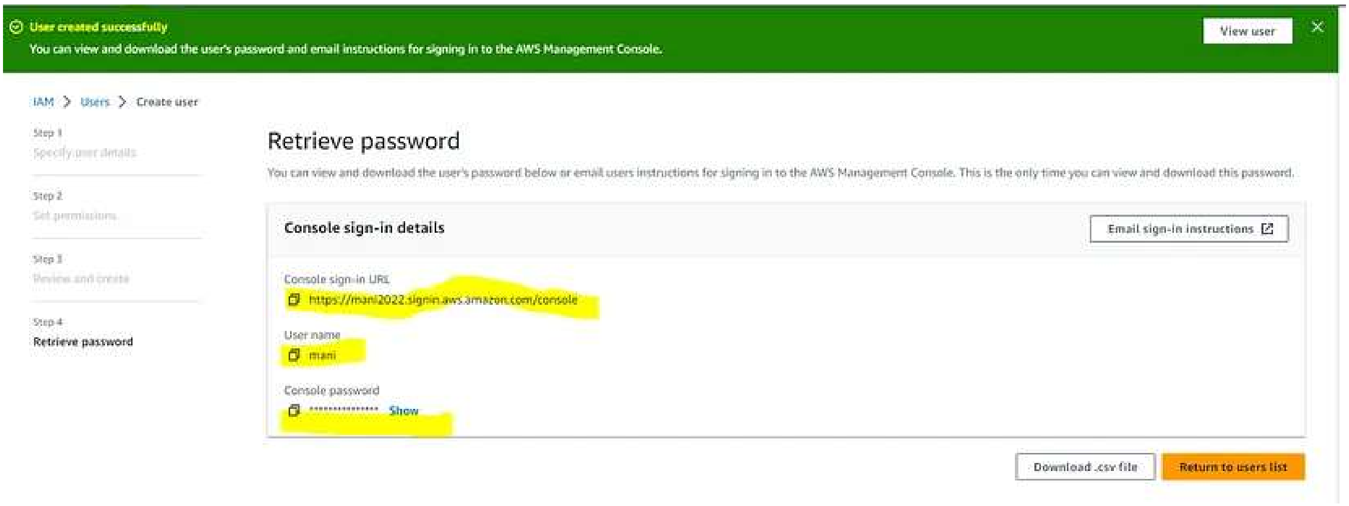
Name	Type	Used as
admin_team	Group	Permissions group

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.

Key	Value - optional
Department	Support Eng

You can add up to 49 more tags.

Click Create user



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

IAM > Users > Create user

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.

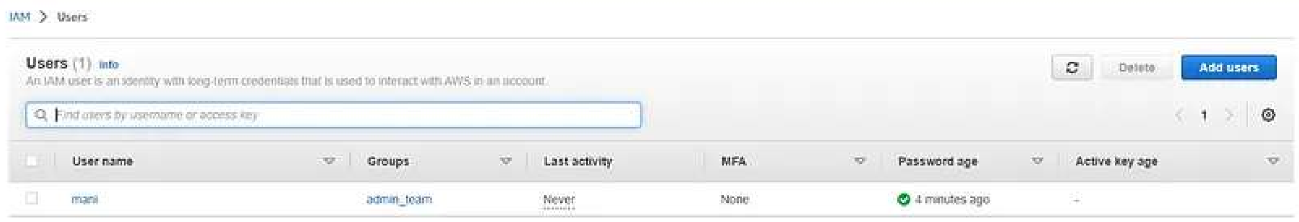
Console sign-in details

Console sign-in URL
https://mani2022.signin.aws.amazon.com/console

User name
mani

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

Click the user name will see permission policies added :

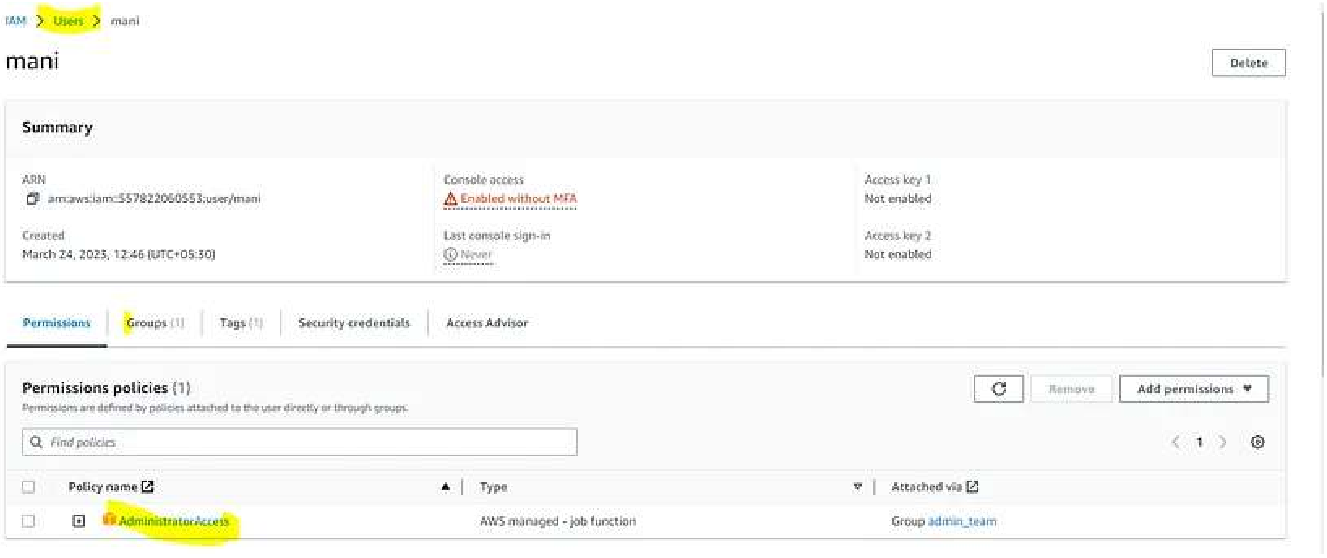


IAM > Users

Users (1) Info

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

User name	Groups	Last activity	MFA	Password age	Active key age
mani	admin_team	Never	None	4 minutes ago	-



IAM > Users > mani

mani

Summary

ARN: arn:aws:iam::557822060553:user/mani

Created: March 24, 2023, 12:46 (UTC+05:30)

Console access: Enabled without MFA

Last console sign-in: Never

Access key 1: Not enabled

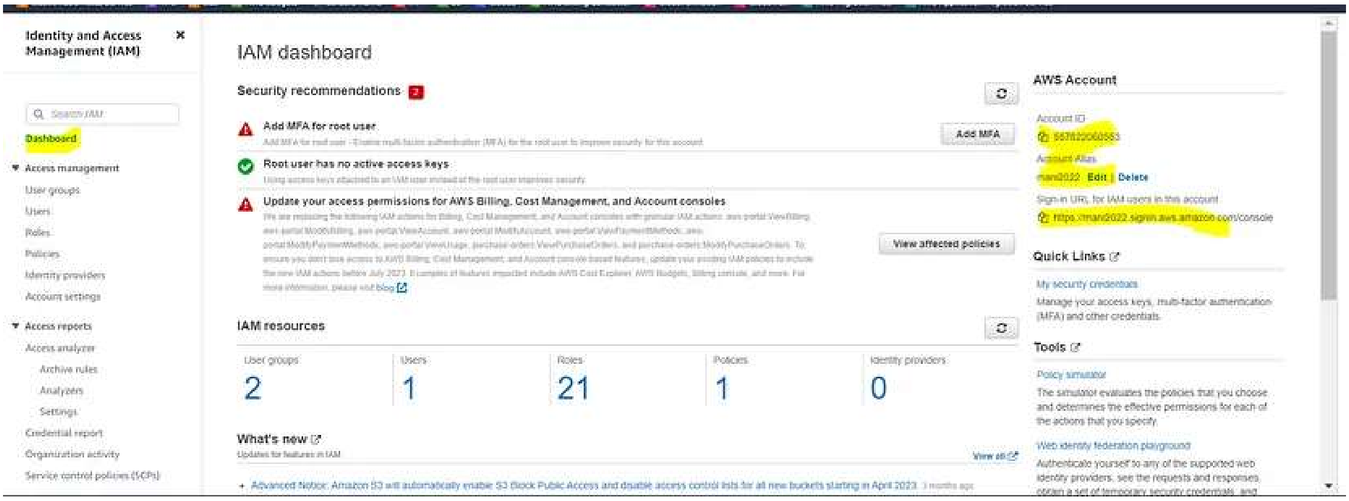
Access key 2: Not enabled

Permissions policies (1)

Permissions are defined by policies attached to the user directly or through groups.

Policy name	Type	Attached via
AdministratorAccess	AWS managed - job function	Group admin_team

Now we can log in with the newly created mani user



IAM dashboard

Security recommendations 2

Add MFA for root user
Add MFA for root user - Enable multi-factor authentication (MFA) for the root user to improve 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.

Update your access permissions for AWS Billing, Cost Management, and Account consoles
We are replacing the following IAM actions for Billing, Cost Management, and Account consoles with granular IAM actions: aws-portal:ViewBilling, aws-portal:ModifyBilling, aws-portal:ViewAccount, aws-portal:ModifyAccount, aws-portal:ViewPaymentMethods, aws-portal:ModifyPaymentMethods, aws-portal:ViewUsage, purchase-orders:ViewPurchaseOrders, and purchase-orders:ModifyPurchaseOrders. To ensure you don't lose access to AWS Billing, Cost Management, and Account console based features, update your existing IAM policies to include the new IAM actions before July 2023. Examples of features impacted include AWS Cost Explorer, AWS Budgets, Billing console, and more. For more information, please visit blog

IAM resources

User groups	Users	Roles	Policies	Identity providers
2	1	21	1	0

What's new
Updates for features in IAM

- Advanced Notice: Amazon S3 will automatically enable S3 Block Public Access and disable access control lists for all new buckets starting in April 2023 3 months ago

AWS Account

Account ID: 557822060553

Account Alias: mani2022 Edit | Delete

Sign-in URL for IAM users in this account: https://mani2022.signin.aws.amazon.com/console

Quick Links

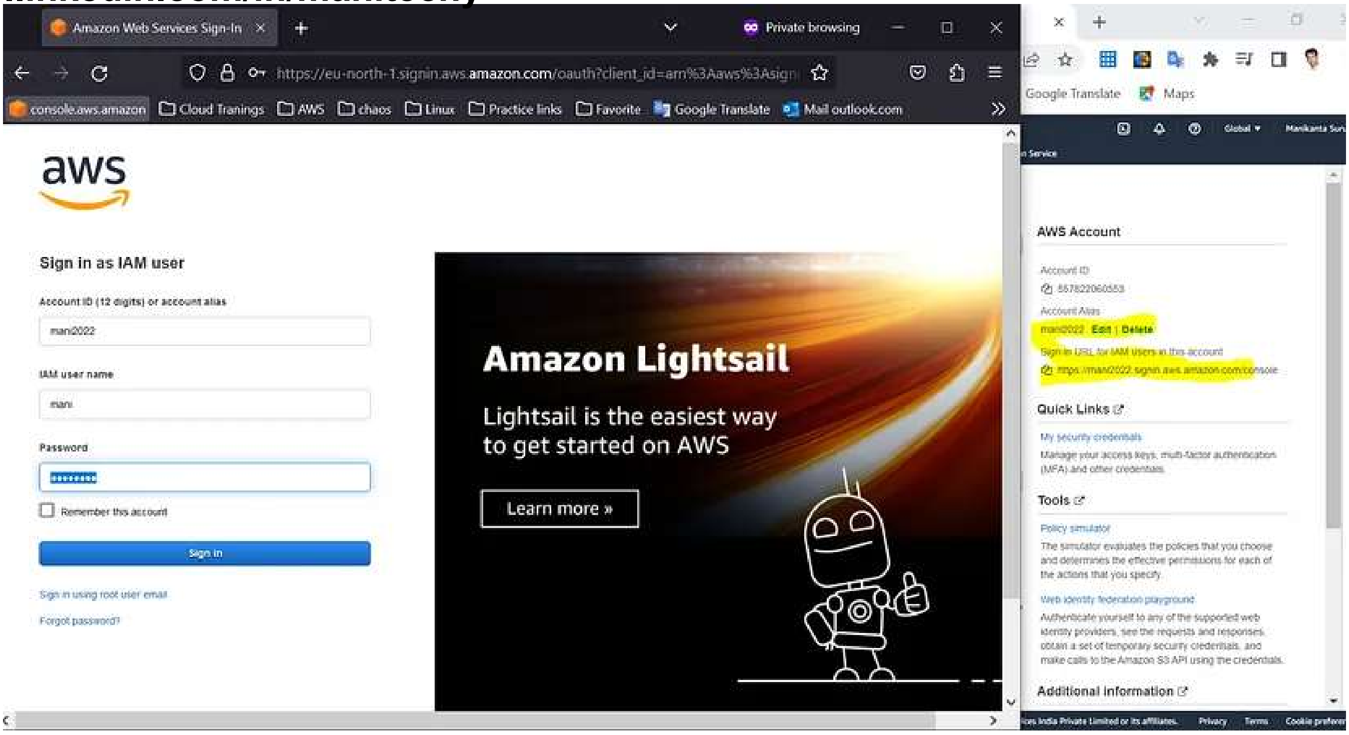
My security credentials
Manage your access keys, multi-factor authentication (MFA) and other credentials.

Tools

Policy simulator
The simulator evaluates the policies that you choose and determines the effective permissions for each of the actions that you specify.

Web identity federation playground
Authenticate yourself to any of the supported web identity providers, see the requests and responses, obtain a set of temporary security credentials, and

Go with https://mani2022.signin.aws.amazon.com/console



Sign in as IAM user

Account ID (12 digits) or account alias: mani2022

IAM user name: mani

Password: ●●●●●●●●●

□ Remember this account

Sign in

Sign in using root user email

Forgot password?

AWS Account

Account ID: 557822060553

Account Alias: mani2022 Edit | Delete

Sign-in URL for IAM users in this account: https://mani2022.signin.aws.amazon.com/console

Quick Links

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

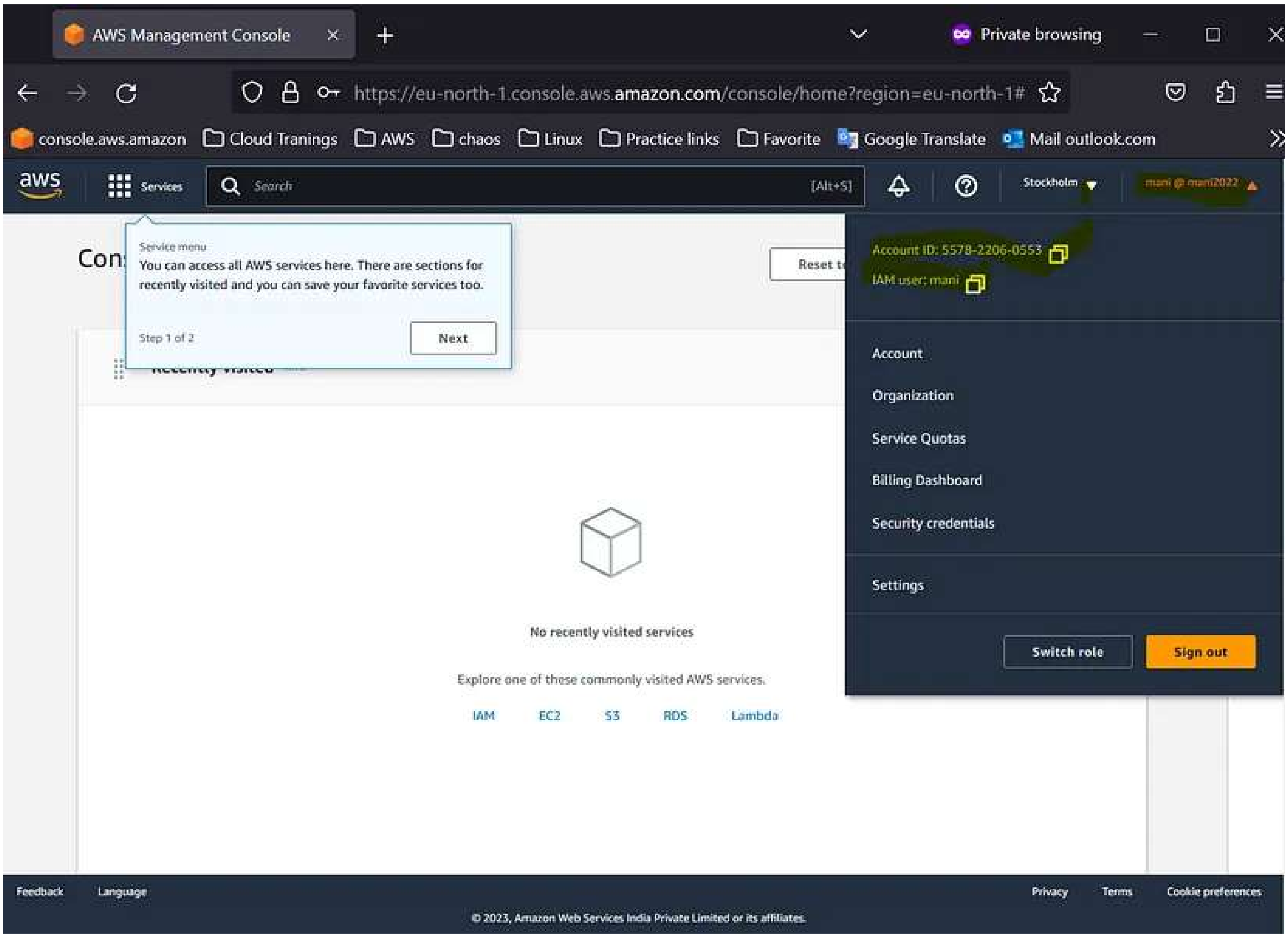
Tools

Policy simulator: The simulator evaluates the policies that you choose and determines the effective permissions for each of the actions that you specify.

Web identity federation playground: Authenticate yourself to any of the supported web identity providers, see the requests and responses, obtain a set of temporary security credentials, and make calls to the Amazon S3 API using the credentials.

Additional information

Login successfully with https://mani2022.signin.aws.amazon.com/console



Service menu

You can access all AWS services here. There are sections for recently visited and you can save your favorite services too.

Step 1 of 2 — Next

Account ID: 5578-2206-0553

IAM user: mani

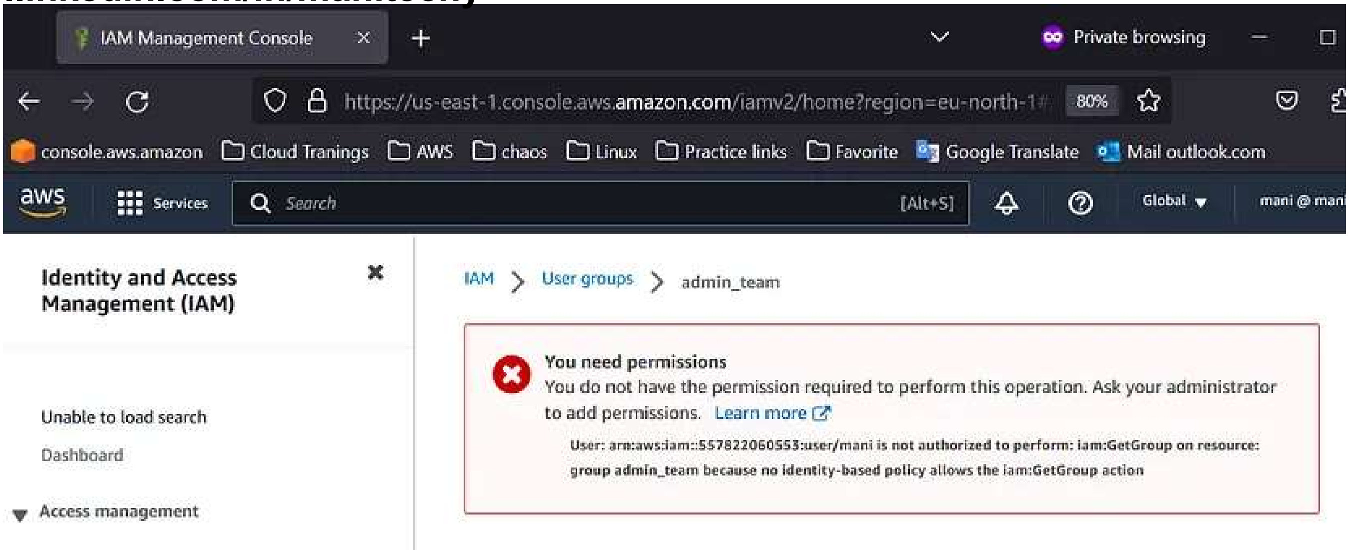
No recently visited services

Explore one of these commonly visited AWS services.

© 2023, Amazon Web Services India Private Limited or its affiliates.

These policies define the permissions of the users :

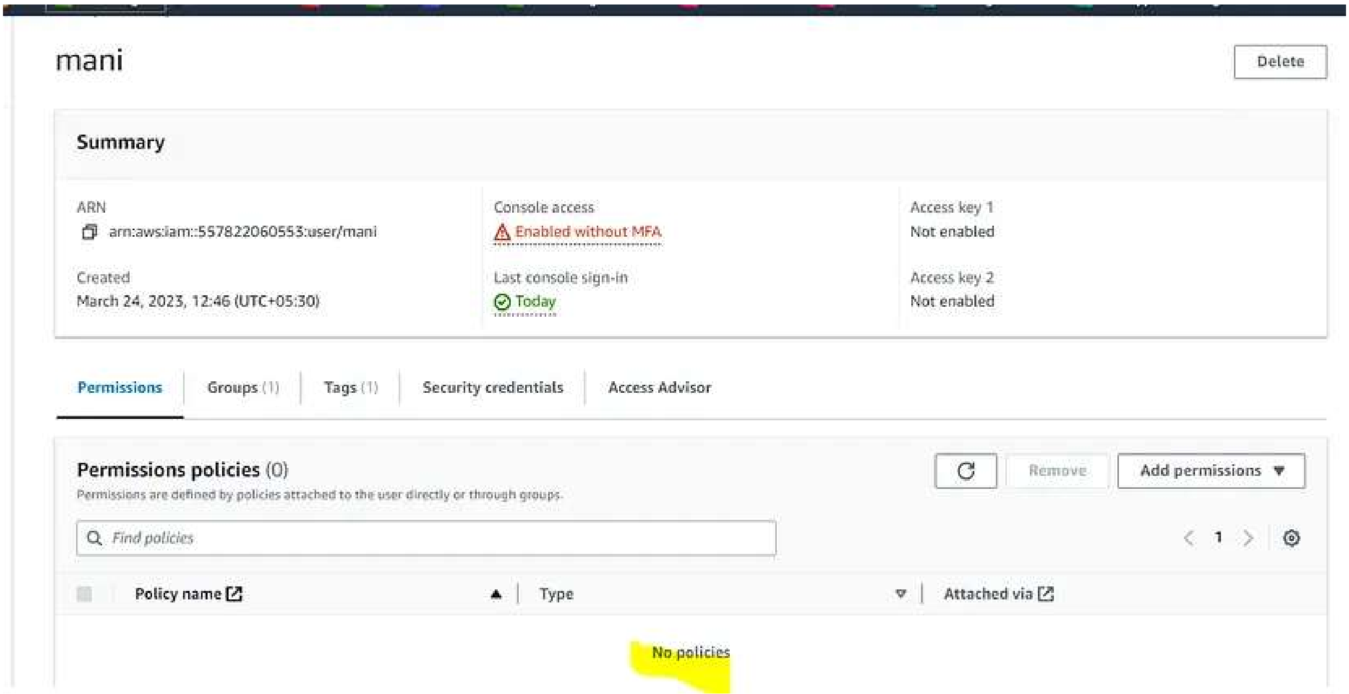
Example :



You need permissions
You do not have the permission required to perform this operation. Ask your administrator to add permissions. Learn more

User: arn:aws:iam::557822060553:user/mani is not authorized to perform: iam:GetGroup on resource: group admin_team because no identity-based policy allows the iam:GetGroup action

users don’t have permission and please find below checking with a root user account.



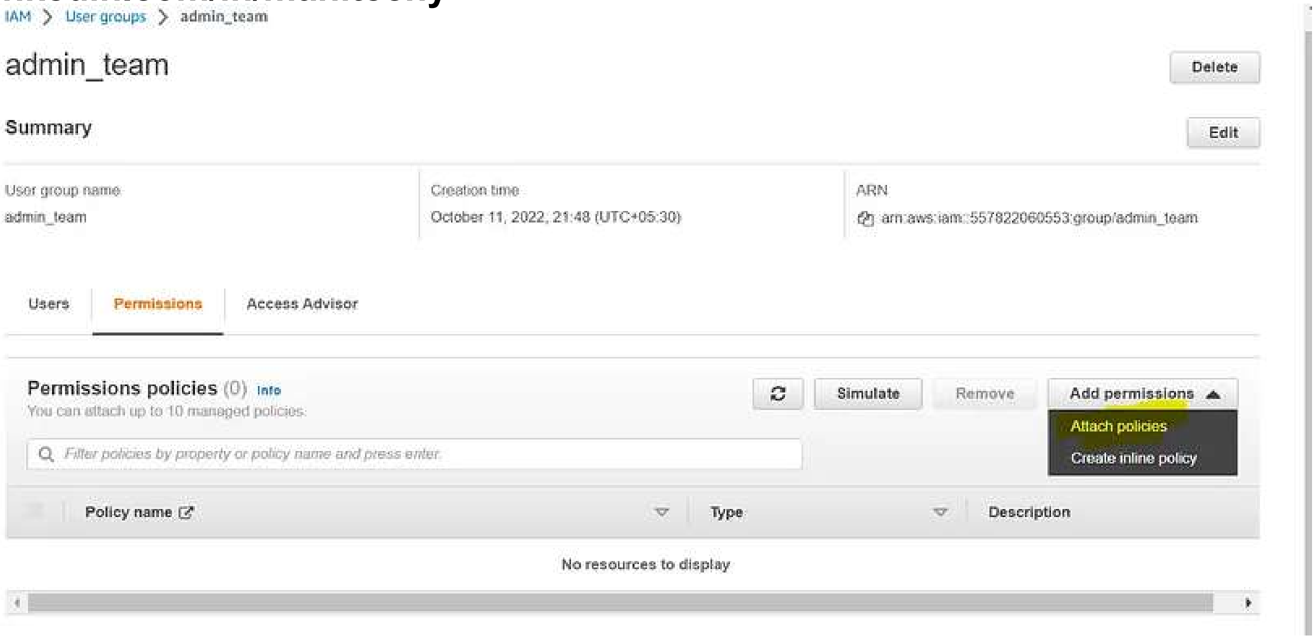
Summary

ARN: arn:aws:iam::557822060553:user/mani
Created: March 24, 2023, 12:46 (UTC+05:30)
Console access: Enabled without MFA
Last console sign-in: Today
Access key 1: Not enabled
Access key 2: Not enabled

Permissions policies (0)
Permissions are defined by policies attached to the user directly or through groups.

Policy name	Type	Attached via
No policies		

Now we need to attach the policy or create inline policy permissions



IAM > User groups > admin_team

admin_team

Summary

User group name	Creation time	ARN
admin_team	October 11, 2022, 21:48 (UTC+05:30)	arn:aws:iam::557822060553:group/admin_team

Permissions policies (0) Info

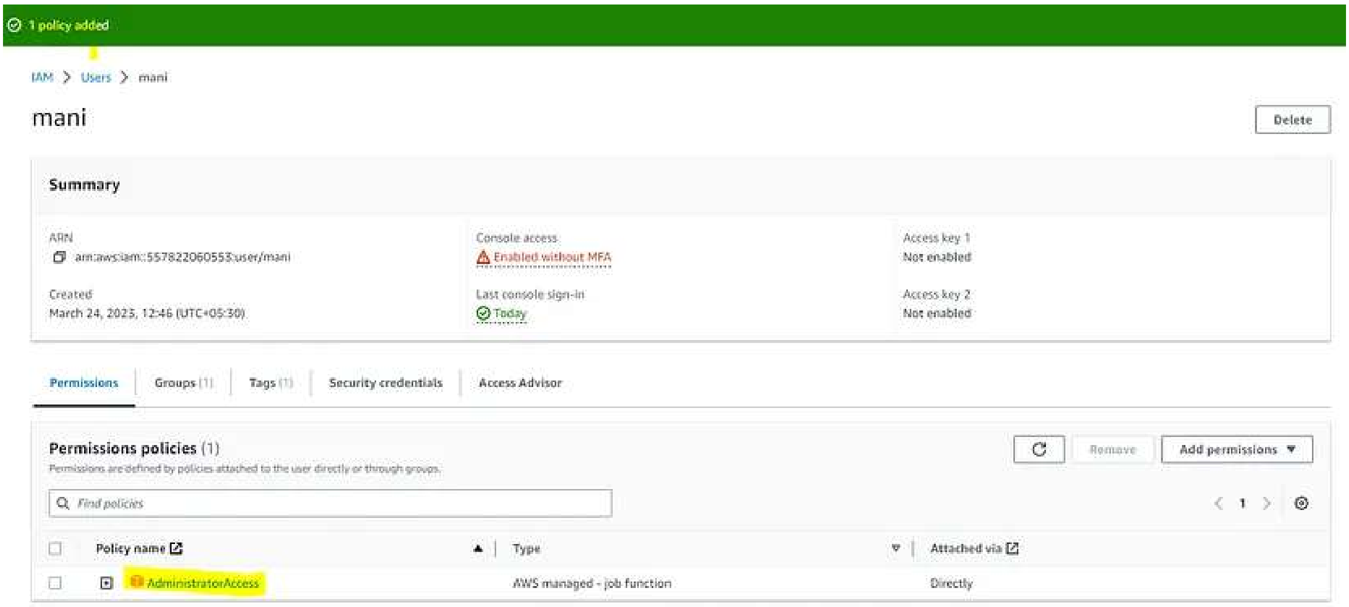
You can attach up to 10 managed policies.

Filter policies by property or policy name and press enter.

Policy name	Type	Description

No resources to display

one policy was added to the user-added group with AdministratorAccess



1 policy added

IAM > Users > mani

mani

Summary

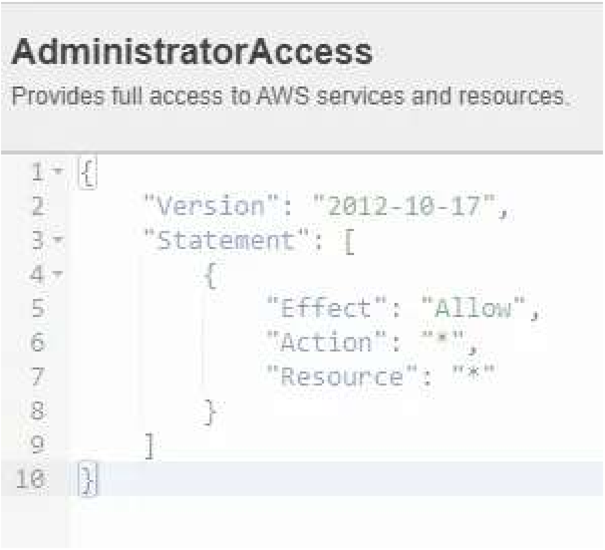
ARN	Console access	Access key 1
arn:aws:iam::557822060553:user/mani	Enabled without MFA	Not enabled
Created	**Last console sign-in**	**Access key 2**
March 24, 2023, 12:46 (UTC+05:30)	Today	Not enabled

Permissions policies (1)

Permissions are defined by policies attached to the user directly or through groups.

Find policies

□	Policy name	Type	Attached via
□	AdministratorAccess	AWS managed - job function	Directly

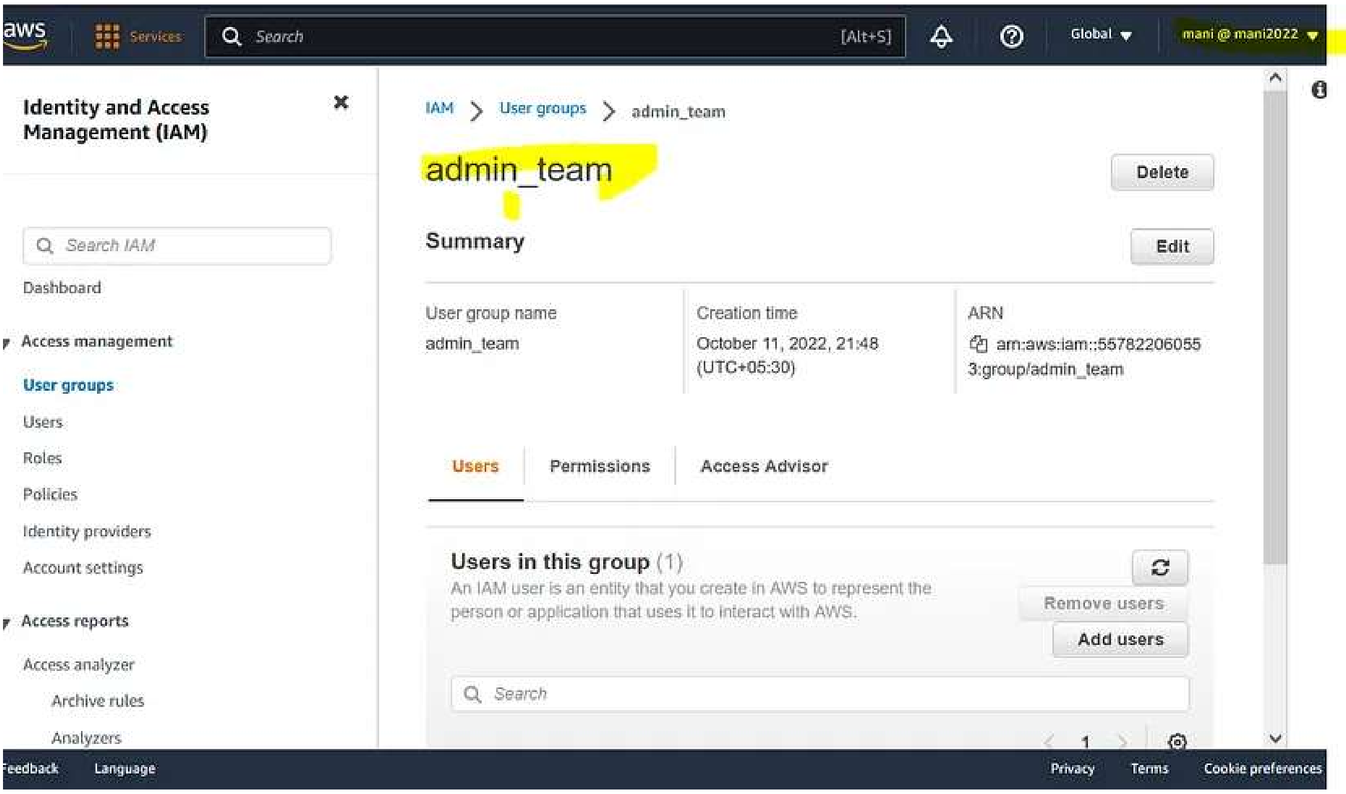


AdministratorAccess

Provides full access to AWS services and resources.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```

Login with mani I am a user Reference the page able to see the page without permission issue



admin_team

Summary

User group name
admin_team

Creation time
October 11, 2022, 21:48 (UTC+05:30)

ARN
arn:aws:iam::557822060553:group/admin_team

Users in this group (1)

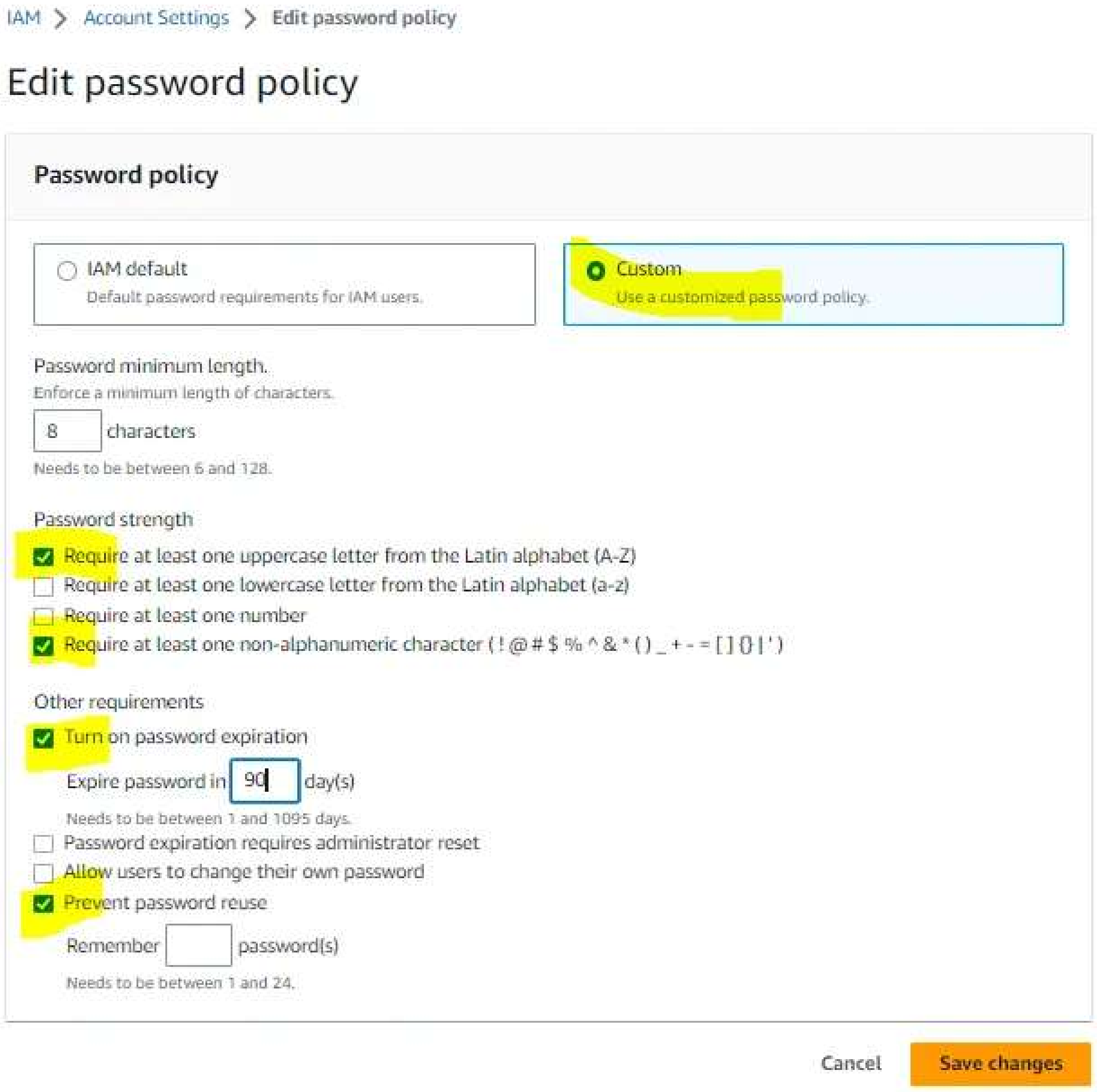
An IAM user is an entity that you create in AWS to represent the person or application that uses it to interact with AWS.

IAM — Password Policy

- Strong passwords = higher security for your account
- In AWS, you can setup a password policy:
- Set a minimum password length
- Require specific character types:
- including uppercase letters
- lowercase letters
- numbers
- non-alphanumeric characters
- Allow all IAM users to change their own passwords
- Require users to change their password after some time (password expiration)

- Prevent password re-use.

goto IAM >Account Settings> **Edit password policy >**



IAM > Account Settings > Edit password policy

Edit password policy

Password policy

○ IAM default
Default password requirements for IAM users.

● Custom
Use a customized password policy.

Password minimum length.
Enforce a minimum length of characters.
8 characters
Needs to be between 6 and 128.

Password strength
☑ Require at least one uppercase letter from the Latin alphabet (A-Z)
□ Require at least one lowercase letter from the Latin alphabet (a-z)
□ Require at least one number
☑ Require at least one non-alphanumeric character (! @ # $ % ^ & * ( ) _ + - = [ ] { } | ')

Other requirements
☑ Turn on password expiration
Expire password in 90 day(s)
Needs to be between 1 and 1095 days.
□ Password expiration requires administrator reset
□ Allow users to change their own password
☑ Prevent password reuse
Remember password(s)
Needs to be between 1 and 24.

Multi-Factor Authentication — MFA

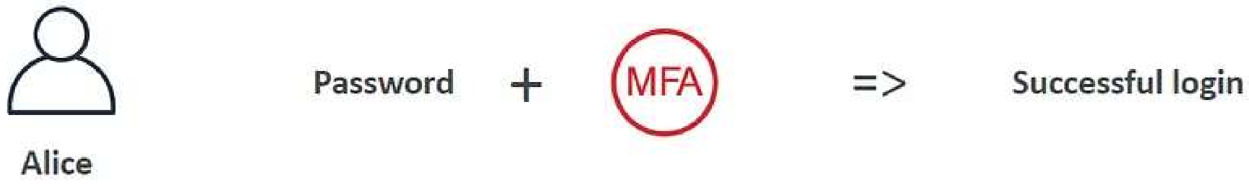
• Users have access to your account and can possibly change

configurations or delete resources in your AWS account

• You want to protect your Root Accounts and IAM users

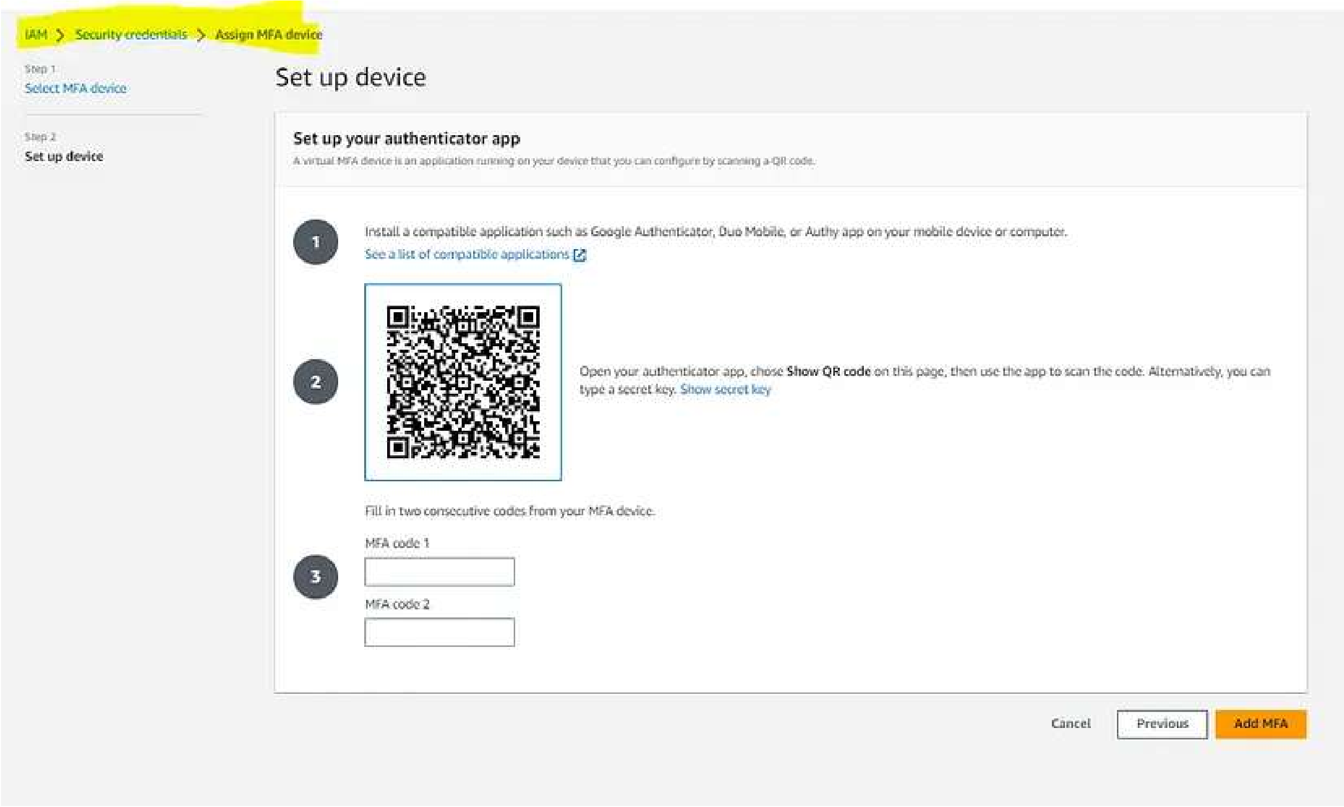
• MFA = password you know + security device you own

- The main benefit of MFA:



if a password is stolen or hacked, the account is not compromised

goto I am **Dashboard** **> Add MFA>select assign MFA >**



Step 1
Select MFA device

Step 2
Set up device

Set up device

Set up your 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, chose **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 Fill in two consecutive codes from your MFA device.

MFA code 1

MFA code 2

How can users access AWS?

• To access AWS, you have three options:

• AWS Management Console (protected by password + MFA)

• AWS Command Line Interface (CLI): protected by access keys

• AWS Software Developer Kit (SDK) — for code: protected by access keys

• Access Keys are generated through the AWS Console

• Users manage their own access keys

• Access Keys are secret, just like a password. Don't share them

• Access Key ID ~= username

• Secret Access Key ~= password

What's the AWS CLI?

• A tool that enables you to interact with AWS services using commands in

your command-line shell

• Direct access to the public APIs of AWS services

• You can develop scripts to manage your resources

• It's open-source https://github.com/aws/aws-cli

- Alternative to using AWS Management Console

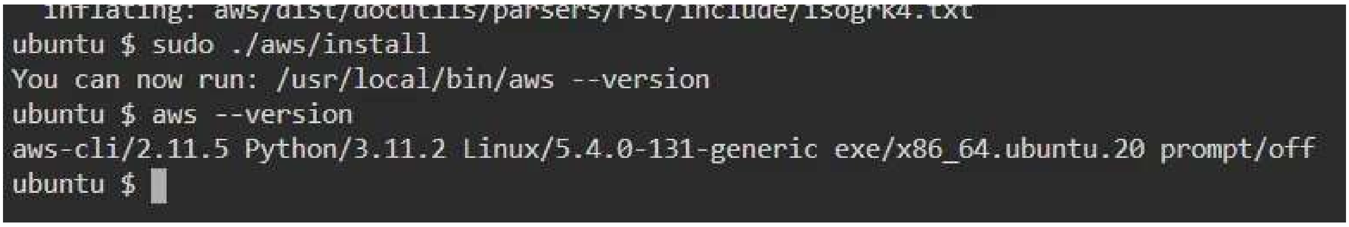
Installing or updating the latest version of the AWS CLI

This topic describes how to install or update the latest release of the AWS Command Line Interface (AWS CLI) on...

docs.aws.amazon.com

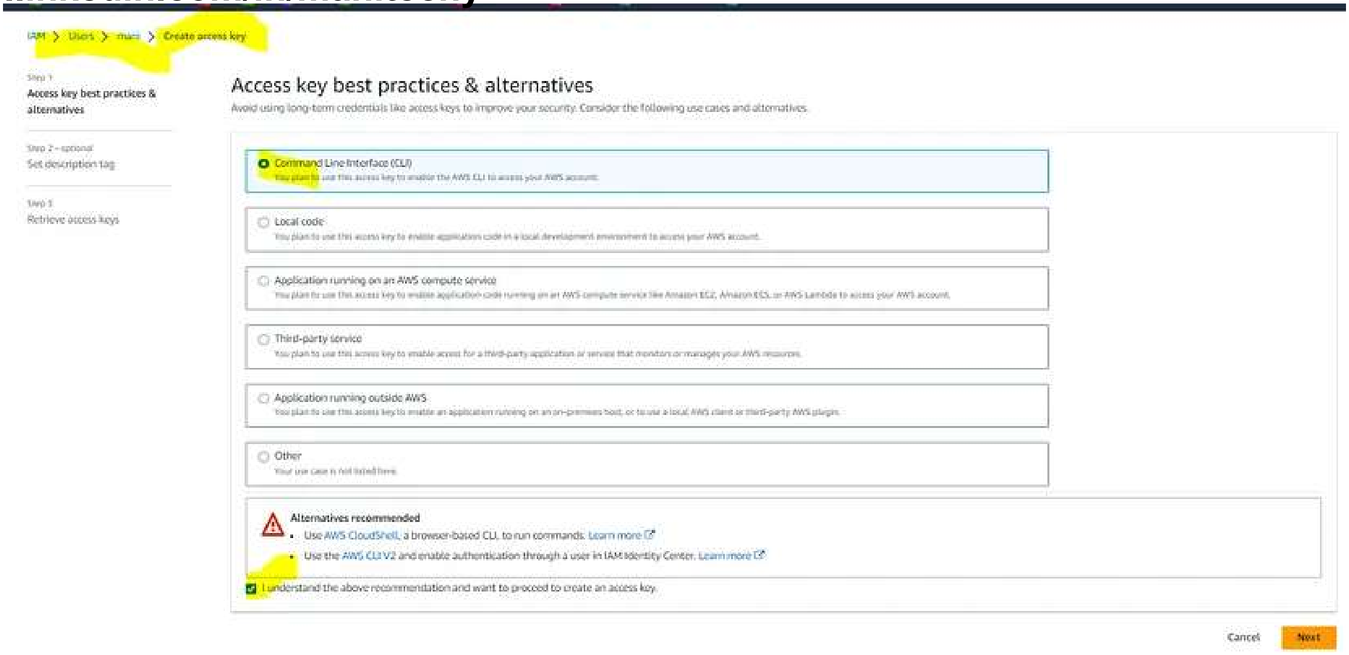
AWS CLI Hands-on and please follow the above link to which type of os use like windows or linux.

i am using Linux vm



```
  inflating: aws/dist/docutils/parsers/rst/include/isogrk4.txt
ubuntu $ sudo ./aws/install
You can now run: /usr/local/bin/aws --version
ubuntu $ aws --version
aws-cli/2.11.5 Python/3.11.2 Linux/5.4.0-131-generic exe/x86_64.ubuntu.20 prompt/off
ubuntu $ 
```

IAM > Users **>ani >Access keys >Create access key**



IAM > Users > mani > Create access key

Step 1
Access key best practices & alternatives

Step 2 - optional
Set description tag

Step 3
Retrieve access keys

Access key best practices & alternatives

Avoid using long-term credentials like access keys to improve your security. Consider the following use cases and alternatives.

● 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.

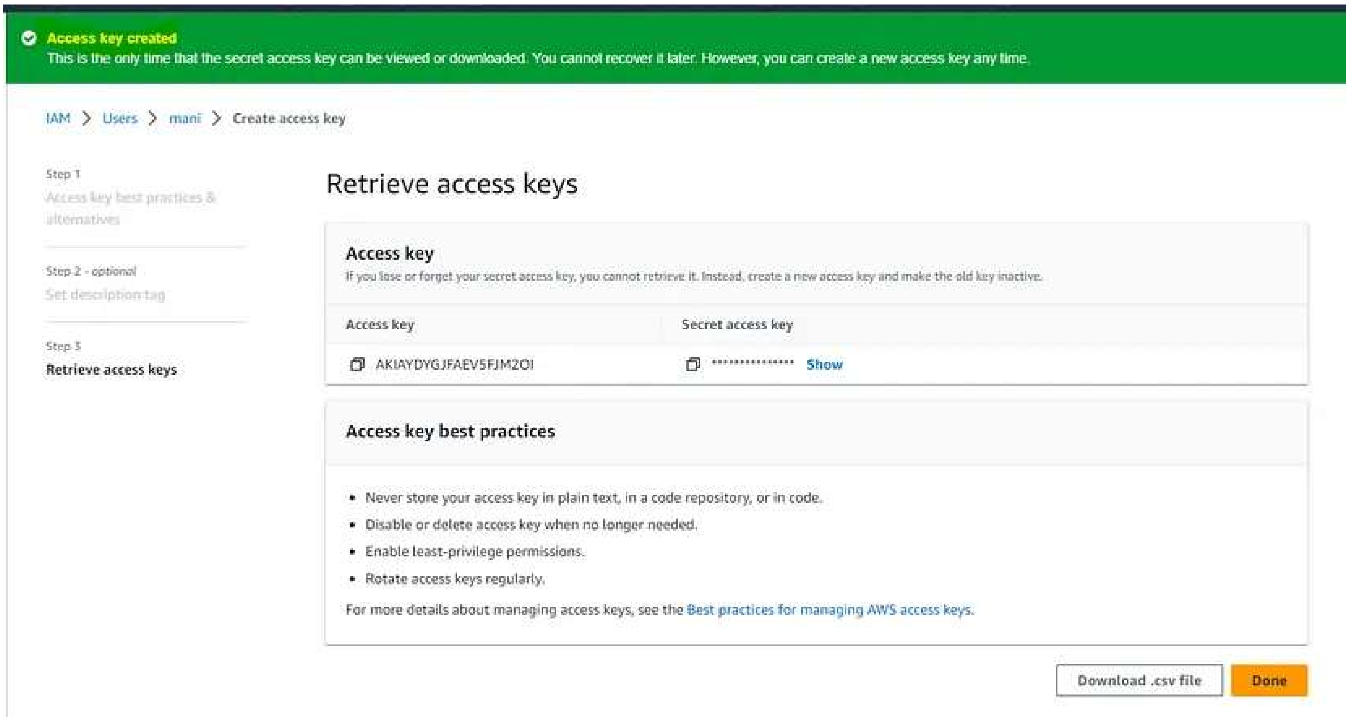
○ Application running outside AWS
You plan to use this access key to enable an application running on an on-premises host, or to use a local AWS client or third-party AWS plugin.

○ 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

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

Access key created and download



Access key created
This is the only time that the secret access key can be viewed or downloaded. You cannot recover it later. However, you can create a new access key any time.

IAM > Users > mani > Create access key

Step 1
Access key best practices & alternatives

Step 2 - optional
Set description tag

Step 3
Retrieve access keys

Retrieve access keys

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
AKIAYDYGJFAEV5FJM2OI	*************** Show

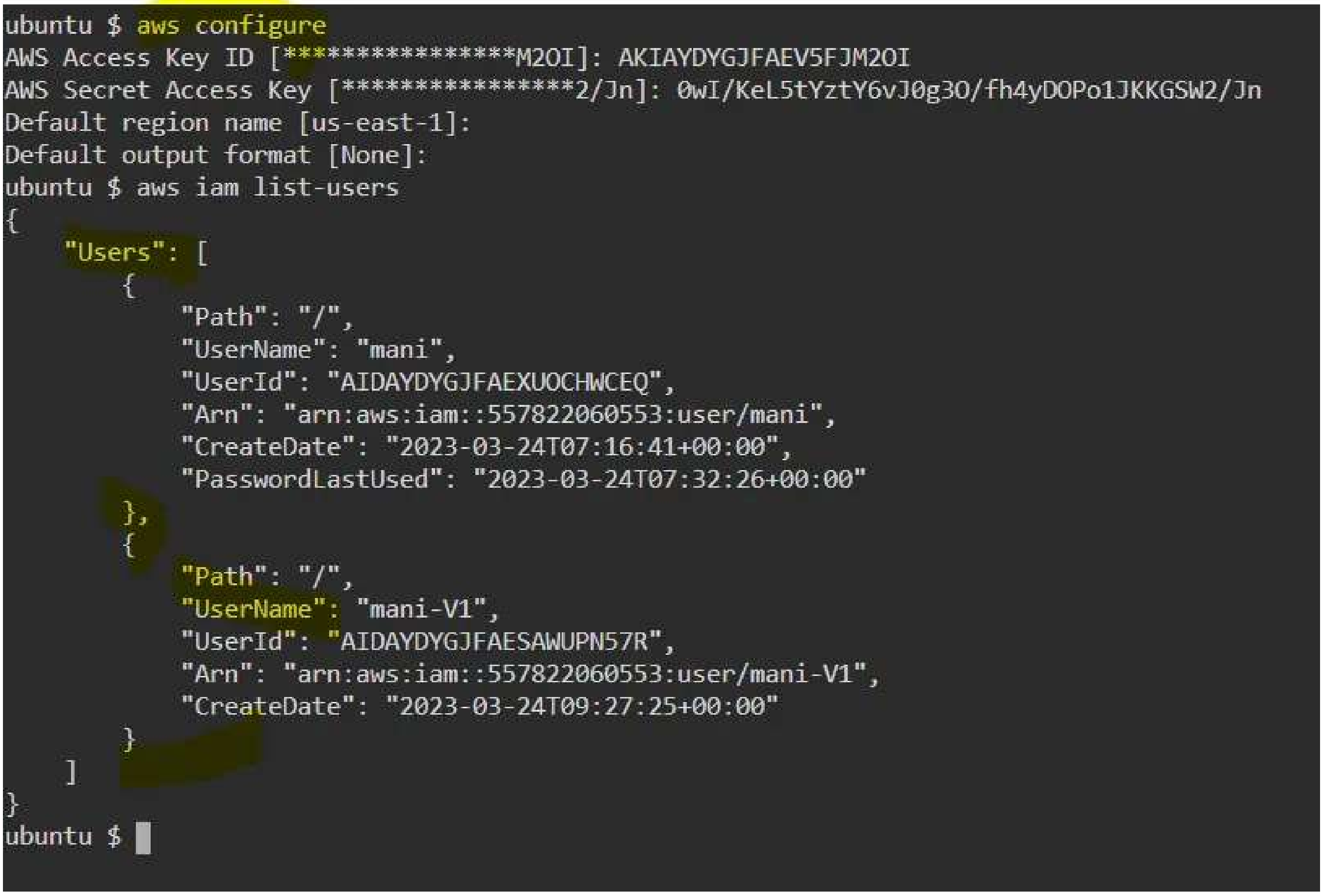
Access key best practices

- 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.

So now my AWS CLI is configured so we can have a look at how it works.

We can do aws iam list-users and press Enter and this will list all the users in my accounts. And as we can see.



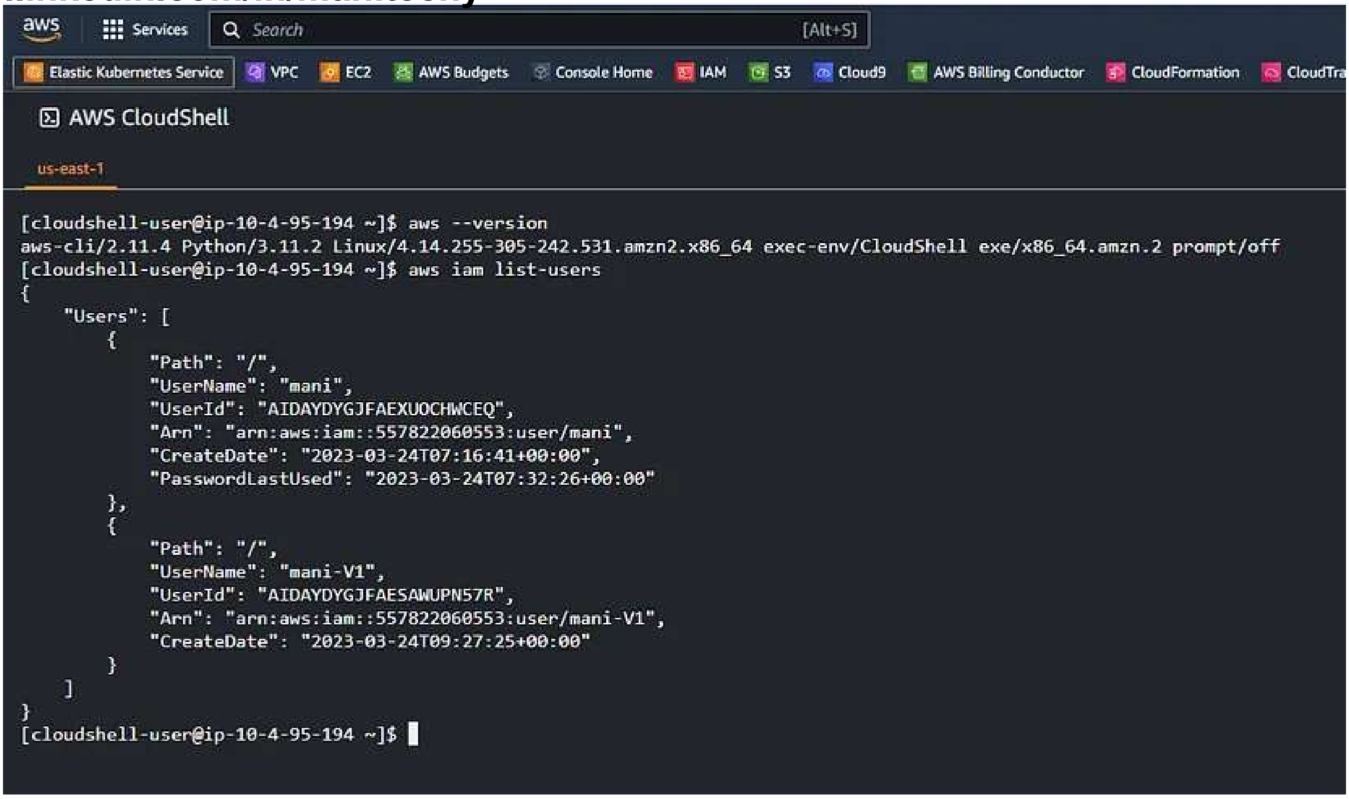
```
ubuntu $ aws configure
AWS Access Key ID [****************M2OI]: AKIAYDYGJFAEV5FJM2OI
AWS Secret Access Key [****************2/Jn]: 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
Default region name [us-east-1]:
Default output format [None]:
ubuntu $ aws iam list-users
{
    "Users": [
        {
            "Path": "/",
            "UserName": "mani",
            "UserId": "AIDAYDYGJFAEXUOCHWCEQ",
            "Arn": "arn:aws:iam::557822060553:user/mani",
            "CreateDate": "2023-03-24T07:16:41+00:00",
            "PasswordLastUsed": "2023-03-24T07:32:26+00:00"
        },
        {
            "Path": "/",
            "UserName": "mani-V1",
            "UserId": "AIDAYDYGJFAESAWUPN57R",
            "Arn": "arn:aws:iam::557822060553:user/mani-V1",
            "CreateDate": "2023-03-24T09:27:25+00:00"
        }
    ]
}
ubuntu $ 
```

AWS CloudShell :

AWS CloudShell is a fully-managed command-line interface (CLI) that allows you to access and manage AWS resources directly from your web browser. It eliminates the need for you to install or configure any additional software on your local computer to use AWS command-line tools. With AWS CloudShell, you can run Linux-based tools and scripts, and access AWS resources from anywhere with an internet connection.

Some use cases for AWS CloudShell include:

- Managing and monitoring AWS resources from anywhere with an internet connection
- Developing and testing AWS Lambda functions, EC2 instances, and other resources using command-line tools and scripts
- Accessing AWS services that are not available through the AWS Management Console
- Collaborating with team members by sharing CloudShell sessions and scripts



```
[cloudshell-user@ip-10-4-95-194 ~]$ aws --version
aws-cli/2.11.4 Python/3.11.2 Linux/4.14.255-305-242.531.amzn2.x86_64 exec-env/CloudShell exe/x86_64.amzn.2 prompt/off
[cloudshell-user@ip-10-4-95-194 ~]$ aws iam list-users
{
    "Users": [
        {
            "Path": "/",
            "UserName": "mani",
            "UserId": "AIDAYDYGJFAEXUOCHWCEQ",
            "Arn": "arn:aws:iam::557822060553:user/mani",
            "CreateDate": "2023-03-24T07:16:41+00:00",
            "PasswordLastUsed": "2023-03-24T07:32:26+00:00"
        },
        {
            "Path": "/",
            "UserName": "mani-V1",
            "UserId": "AIDAYDYGJFAESAWUPN57R",
            "Arn": "arn:aws:iam::557822060553:user/mani-V1",
            "CreateDate": "2023-03-24T09:27:25+00:00"
        }
    ]
}
[cloudshell-user@ip-10-4-95-194 ~]$
```

IAM Guidelines & Best Practices

• Don’t use the root account except for the AWS account setup

• One physical user = One AWS user

• Assign users to groups and assign permissions to groups

• Create a strong password policy

• Use and enforce the use of Multi-Factor Authentication (MFA)

• Create and use Roles for giving permissions to AWS services

• Use Access Keys for Programmatic Access (CLI / SDK)

• Audit permissions of your account with the IAM Credentials Report

- Never share IAM users & Access Keys

IAM Section — Summary

• Users: mapped to a physical user, has a password for AWS Console

• Groups: contains users only

- Policies: JSON document that outlines permissions for users or groups

- Roles: for EC2 instances or AWS services

- Security: MFA + Password Policy

- Access Keys: access AWS using the CLI or SDK

- Audit: IAM Credential Reports & IAM Access Advisor

Sources: AWS.

That’s it, thank you for reading.

https://github.com/gefkkd/AWS_100-Days_Challenge.git

👉 In case you would like to continue the discussion, you can always reach out to me on Twitter or on LinkedIn for professional networking, if you feel like following me on GitHub you can also do that.

👉 Follow Cloudnloud Tech Community for more insightful knowledge & resources & CloudnLoud YouTube channel.

AWS Cloud Computing Learning Security Careers