

AWS IAM

AWS Identity and Access Management (IAM) is a web service that enables secure control over access to AWS resources. It allows you to create and manage user authentication, and limit access to specific AWS resources.

1. Creating user with name – Sanjeeva and giving ec2 full access

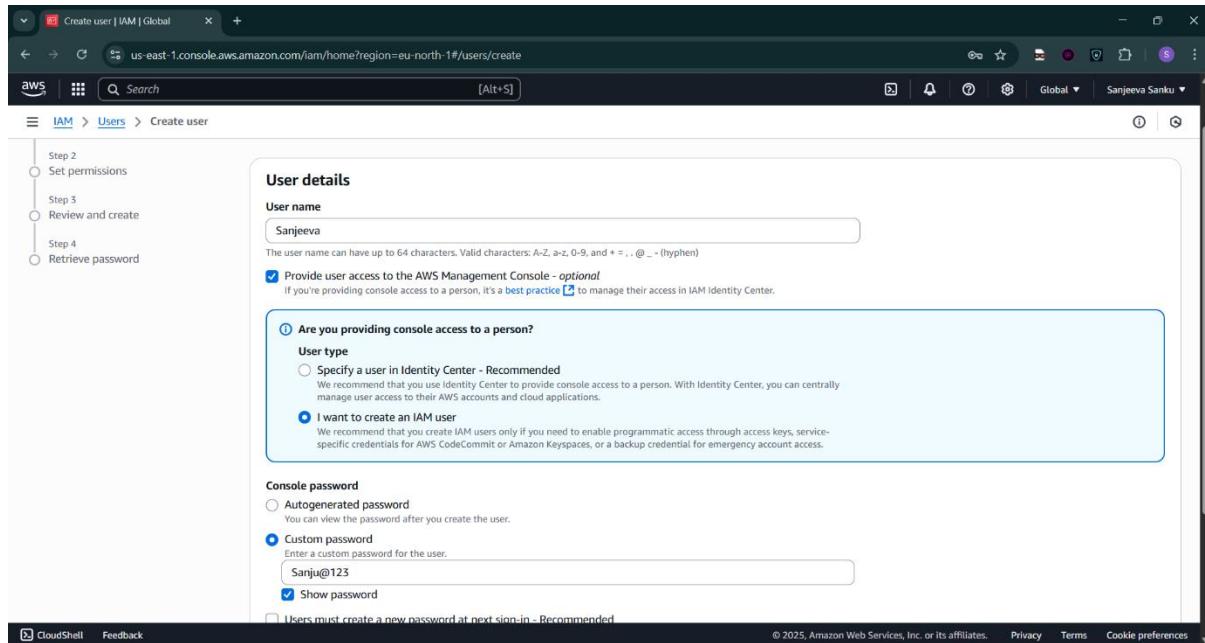
Login to the root user

The screenshot shows the AWS IAM Dashboard. On the left sidebar, under 'Access management', 'Users' is selected. The main content area displays 'Security recommendations' with a warning about adding MFA for the root user and a note that the root user has no active access keys. Below this is the 'IAM resources' section, which shows 0 User groups, 0 Users, 2 Roles, 0 Policies, and 0 Identity providers. A 'What's new' section lists recent announcements from AWS IAM. On the right side, there are three boxes: 'AWS Account' (Account ID: 391984502893, Account Alias: Create, Sign-in URL: https://391984502893.signin.aws.amazon.com/console), 'Quick Links' (My security credentials, Manage your access keys, multi-factor authentication (MFA) and other credentials), and 'Tools' (Policy simulator, The simulator evaluates the policies that you choose and determines the effective permissions for each of).

Click on Users on the side bar and click on create user

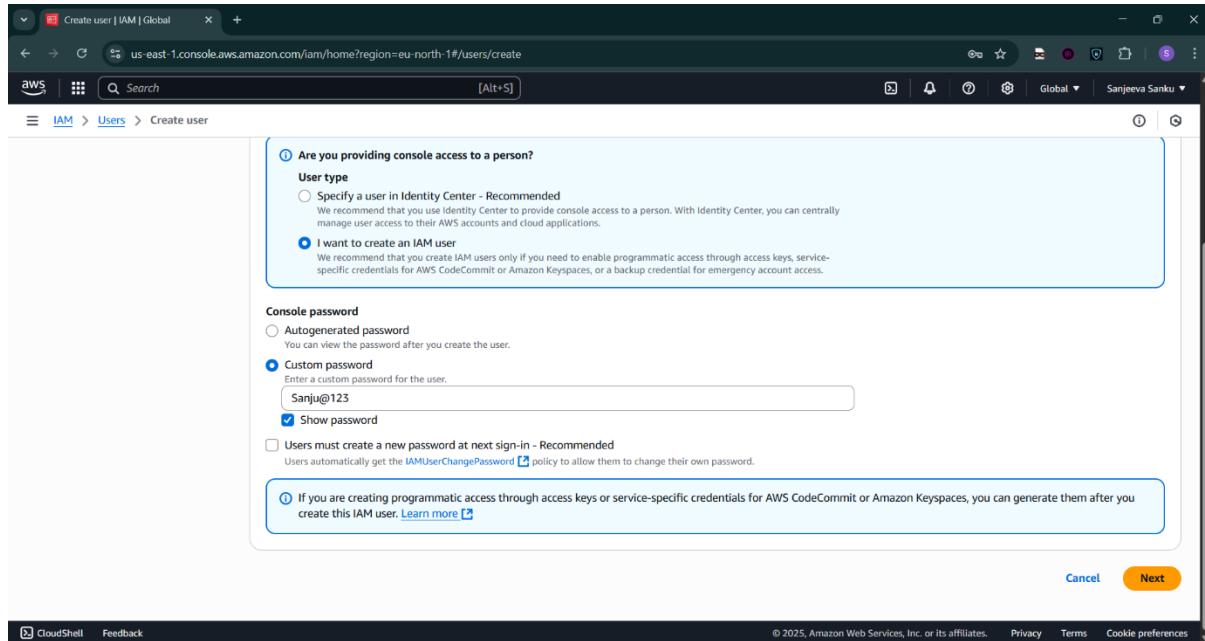
The screenshot shows the 'Users' page in the AWS IAM console. The left sidebar shows 'Users' is selected under 'Access management'. The main content area has a heading 'Users (0) Info' with a note that an IAM user is an identity with long-term credentials used to interact with AWS. A search bar is present above a table. The table has columns for 'User name', 'Path', 'Group', 'Last activity', 'MFA', 'Password age', 'Console last sign-in', and 'Acc'. A message at the bottom of the table says 'No resources to display'. On the right side, there are buttons for 'Delete' and 'Create user'.

Give user name as Sanjeeva, and provide user access to the Aws Management console so that they can view console, and create an IAM user with custom password.



The screenshot shows the 'Create user' wizard in the AWS Management Console. The user name is set to 'Sanjeeva'. The 'Provide user access to the AWS Management Console' checkbox is checked. The 'User type' section shows 'I want to create an IAM user' selected. The 'Console password' section shows 'Custom password' selected with the value 'Sanju@123'.

Uncheck the users much create a new password at next sign in it is not needed.



The screenshot shows the 'Create user' wizard in the AWS Management Console. The user name is set to 'Sanjeeva'. The 'Provide user access to the AWS Management Console' checkbox is unchecked. The 'User type' section shows 'I want to create an IAM user' selected. The 'Console password' section shows 'Custom password' selected with the value 'Sanju@123'.

After clicking on next it will show 3 options Add user to group , copy permissions, Attach policies directly, since we don't have group we will go for attach policies directly

And give AmazonEC2FullAccess

Permissions policies (1/1353)			
Choose one or more policies to attach to your new user.			
Filter by Type			
Policy name	Type	Attached entities	
AmazonEC2ContainerRegistryFullAccess	AWS managed	0	
AmazonEC2ContainerRegistryPowerUser	AWS managed	0	
AmazonEC2ContainerRegistryPullOnly	AWS managed	0	
AmazonEC2ContainerRegistryReadOnly	AWS managed	0	
AmazonEC2ContainerServiceAutoscale...	AWS managed	0	
AmazonEC2ContainerServiceEventsRole	AWS managed	0	
AmazonEC2ContainerServiceforEC2Role	AWS managed	0	
AmazonEC2ContainerServiceRole	AWS managed	0	
AmazonEC2FullAccess	AWS managed	0	
AmazonEC2ReadOnlyAccess	AWS managed	0	
AmazonEC2RoleforAWSCodeDeploy	AWS managed	0	

Click on next to review and create the user

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
Sanjeeda	Custom password	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.

Create user

We can download .csv file for the credentials

The screenshot shows the 'Create user' process in the AWS Management Console. The current step is 'Step 4: Retrieve password'. A green success message at the top states 'User created successfully'. Below it, a note says 'You can view and download the user's password and email instructions for signing in to the AWS Management Console.' A 'View user' button is available. On the left, a vertical navigation bar shows steps: Step 1 (Specify user details), Step 2 (Set permissions), Step 3 (Review and create), and Step 4 (Retrieve password, which is selected). The main content area displays 'Console sign-in details' with a 'Console sign-in URL' (https://391984502893.signin.aws.amazon.com/console) and a 'User name' (Sanjeeva). A 'Console password' field is shown with a 'Show' link. A 'Email sign-in instructions' button is also present. At the bottom are 'Cancel', 'Download .csv file' (which is highlighted in blue), and 'Return to users list' buttons.

Next create a user named Sanjeeva-frontend and five AmazonS3FullAccess

The screenshot shows the 'Create user' process in the AWS Management Console. The current step is 'Step 3: Review and create'. The 'User details' section shows a 'User name' of 'Sanjeeva-frontend'. The 'Console password type' is set to 'Custom password' and 'Require password reset' is set to 'No'. The 'Permissions summary' section shows a single permission entry: 'Name' (AmazonS3FullAccess), 'Type' (AWS managed), and 'Used as' (Permissions policy). The 'Tags - optional' section indicates no tags are associated with the resource. At the bottom are 'Cancel', 'Previous', and 'Create user' buttons.

2.Creating a Group and adding xyz user to it

Now go to User groups and create group and give name sanjeeva-cse and attach permissions policies AmazonEC2FullAccess

The screenshot shows the 'Create user group' interface in the AWS IAM console. The 'User group name' field is filled with 'sanjeeva-cse'. In the 'Add users to the group' section, two users are listed: 'Sanjeeda' and 'Sanjeeda-frontend'. Under 'Attach permissions policies', the 'AmazonEC2FullAccess' policy is selected. The left sidebar shows navigation options like 'Identity and Access Management (IAM)', 'Access management', and 'Access reports'.

Now create user with name xyz and now choose option add user to group.

Choose sanjeeva-cse

The screenshot shows the 'Create user' process at Step 2: Set permissions. The 'Add user to group' option is selected, and 'sanjeeva-cse' is chosen from the user group list. The 'Permissions options' section shows other options like 'Copy permissions' and 'Attach policies directly'. The 'User groups' section lists 'sanjeeva-cse' with 'AmazonEC2FullAccess' attached. Navigation buttons 'Cancel', 'Previous', and 'Next' are visible at the bottom.

Screenshot of the AWS IAM 'Create user' wizard - Step 3: Review and create.

The page shows the following details:

- User details:** User name: xyz, Console password type: Custom password, Require password reset: No
- Permissions summary:** sanjeeva-cse (Group) is assigned to the 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.
- Buttons:** Cancel, Previous, Create user

3. Login all the IAM users and check their access

Sanjeeva has AmazonEC2FullAccess

Screenshot of the AWS IAM user sign-in page.

The sign-in form includes the following fields:

- Account ID or alias (Don't have?): 391984502893
- Remember this account (checkbox)
- IAM username: Sanjeeva
- Password: (redacted)
- Show Password (checkbox)
- Having trouble? (link)
- Sign in (button)
- Sign in using root user email (link)
- Create a new AWS account (link)

To the right of the sign-in form is a promotional banner for Amazon Lightsail:

Amazon Lightsail
Lightsail is the easiest way to get started on AWS
[Learn more »](#)

Try to create EC2 instance

The screenshot shows the 'Launch an instance' wizard on the AWS EC2 console. The current step is 'Name and tags'. A blue callout box highlights the 'Name' field, which contains the value 'first'. Other fields in this section include 'Add additional tags' and a 'Search' bar.

Summary
Number of instances: 1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.7.2... [read more](#)
ami-006b4a3ad5f56fbdf

Virtual server type (instance type)
t3.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2 micro isn't available) when used with free tier

Launch instance [Preview code](#)

The screenshot shows the 'Launch an instance' wizard on the AWS EC2 console. The current step is 'Instance type'. A blue callout box highlights the 't3.micro' instance type selection. Other details shown include 'Family: t3', '2 vCPU', '1 GiB Memory', 'Current generation: true', and various pricing options. A note states 'Additional costs apply for AMIs with pre-installed software'.

Summary
Number of instances: 1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.7.2... [read more](#)
ami-006b4a3ad5f56fbdf

Virtual server type (instance type)
t3.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2 micro isn't available) when used with free tier

Launch instance [Preview code](#)

We can see that it is created.

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with sections like Dashboard, EC2 Global View, Events, Instances (selected), Images, Elastic Block Store, and CloudShell. The main area has a table titled 'Instances (1) Info' with one row. The row details are:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
first	i-00591a309fe06e3ff	Running	t3.micro	Initializing		eu-north-1b	ec2-13-51-1

Below the table, there's a section titled 'Select an instance' with a dropdown menu.

Now let us try to create S3bucket

The screenshot shows the AWS S3 Create Bucket page. The top navigation bar includes links for Create S3 bucket, AWS Lambda, and CloudWatch Metrics. The main content area is titled 'Create bucket' and includes a 'General configuration' section. In this section, the 'Bucket type' dropdown is set to 'General purpose'. Other options shown include 'Directory' and 'Archival'. The 'Bucket name' field is filled with 'mybucket-531'. Below the bucket name, there's a note about naming rules and a 'Copy settings from existing bucket - optional' section with a 'Choose bucket' button. At the bottom of the configuration section, there's an 'Object Ownership' section with a note about controlling object ownership from other accounts via ACLs.

We can see that we don't have permission to create s3 bucket

The screenshot shows the 'Create bucket' page in the AWS S3 console. At the top, there are options for server-side encryption (SSE-KMS or DSSE-KMS) and a 'Bucket Key' section where 'Enable' is selected. Below this is an 'Advanced settings' section. A note at the bottom of the page states: 'After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.' A red box highlights an error message: 'Failed to create bucket' with the subtext 'To create a bucket, the s3:CreateBucket permission is required.' There is a 'Diagnose with Amazon Q' button next to it. At the bottom right are 'Cancel' and 'Create bucket' buttons.

Sanjeeva-frontend has awss3fullaccess

The screenshot shows the 'IAM user sign in' page for the user 'Sanjeeva-frontend'. The form includes fields for 'Account ID or alias', 'Remember this account', 'IAM username' (set to 'Sanjeeva-frontend'), 'Password', 'Show Password' (unchecked), 'Having trouble?' (link), 'Sign in' (button), and 'Sign in using root user email' (link). To the right of the form is a promotional banner for 'Amazon Lightsail' featuring a cartoon robot character. A message at the top of the page says: 'You are currently using the improved sign in experience. The improved sign in experience will launch soon. During this time, you can still change back to legacy sign in using the dropdown in the upper right corner.'

Let us create s3 bucket

The screenshot shows the 'Create bucket' page in the AWS S3 console. In the 'General configuration' section, the 'Bucket name' is set to '53l-bucket-frontend'. Under 'Bucket type', 'General purpose' is selected. In the 'Object Ownership' section, 'ACLS disabled (recommended)' is selected. Other options like 'ACLS enabled' and 'Directory' are also shown.

We can create

The screenshot shows the 'Buckets' page in the AWS S3 console. A green banner at the top indicates that the bucket '53l-bucket-frontend' was successfully created. Below this, the 'General purpose buckets' list shows one item: '53l-bucket-frontend' (Region: Europe (Stockholm) eu-north-1, Creation date: June 5, 2025, 10:29:05 (UTC+05:30)).

Try adding object (write)

The screenshot shows the AWS S3 console interface for uploading objects. The URL in the address bar is `eu-north-1.console.aws.amazon.com/s3/upload/53l-bucket-frontend?region=eu-north-1&bucketType=general`. The page title is "Upload objects - S3 bucket 53l". The main area is titled "Upload" with a sub-section "Info". It instructs users to add files or folders by dragging them or clicking "Add files" or "Add folder". A table titled "Files and folders (1 total, 253.0 B)" lists the uploaded file "kwaishain.txt". The file details are: Name: kwaishain.txt, Type: text/plain, Size: 253.0 B. Below this is a "Destination" section with "Destination": `s3://53l-bucket-frontend` and a "Destination details" subsection. At the bottom, there are "Permissions" settings and navigation links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

try downloading (read) both successful

The screenshot shows the AWS S3 console interface for viewing objects. The URL in the address bar is `eu-north-1.console.aws.amazon.com/s3/buckets/53l-bucket-frontend?region=eu-north-1&tab=objects`. The page title is "53l-bucket-frontend". The main area is titled "Objects (1/1)". It displays a single object "kwaishain.txt" with details: Name: kwaishain.txt, Type: txt, Last modified: June 5, 2025, 10:30:18 (UTC+05:30), Size: 253.0 B, Storage class: Standard. Below the object list is a note about Amazon S3 inventory. A context menu is open over the "kwaishain.txt" object, listing other files: "kwaishain.txt", "sanjeva-iam-pem.pem", "xyc_credentials.csv", "Sanjeva-frontend_credentials.csv", and "Sanjeva_credentials.csv". The menu also includes options for Copy S3 URI, Copy URL, Download, Open, and Delete. At the bottom, there are "CloudShell", "Feedback", "Privacy", "Terms", and "Cookie preferences" links.

Same manner let us create EC2 instance, we can see we not authorized to create instance

The screenshot shows the AWS EC2 Instances page. The left sidebar is collapsed. The main area has a heading 'Instances info' with a search bar and filters for 'Name', 'Instance ID', 'Instance state', 'Instance type', 'Status check', 'Alarm status', 'Availability Zone', and 'Public IPv4'. A red box highlights an error message: 'You are not authorized to perform this operation. User: arn:aws:iam::391984502893:user/Sanjeeva-frontend is not authorized to perform: ec2:DescribeInstances because no identity-based policy allows the ec2:DescribeInstances action'. Below this is a section titled 'Select an instance'.

Now check whether xyz has EC2 full access

Create instance with name xyz

The screenshot shows the AWS IAM user sign-in page. The left sidebar is collapsed. The main area has a heading 'IAM user sign in' with fields for 'Account ID or alias' (set to '391984502893'), 'IAM username' (set to 'xyz'), 'Password' (redacted), and 'Remember this account' (unchecked). There are checkboxes for 'Show Password' and 'Having trouble?'. Below these are 'Sign in' and 'Sign in using root user email' buttons. To the right is a promotional banner for 'Amazon Lightsail' featuring a cartoon robot.

Launch an instance | EC2 | eu-north-1

eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#LaunchInstances:

aws Search [Alt+S] Europe (Stockholm) xyz @ 3919-8450-2893

EC2 Instances Launch an instance

Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name Add additional tags

Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

Search our full catalog including 1000s of application and OS images

Quick Start

Amazon Linux macOS Ubuntu Windows Red Hat SUSE Linux Debian

aws Mac ubuntu Microsoft RedHat SUSE debian

Browse more AMIs Including AMIs from AWS, Marketplace and the Community

Summary

Number of instances Info

1

Software Image (AMI)

Amazon Linux 2023 AMI 2023.7.2... read more
ami-006b4a3ad5f56fbdf

Virtual server type (instance type)

t3.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where +2 micro hours available when used with free tier)

Cancel Launch instance Preview code

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Instances | EC2 | eu-north-1

eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#Instances:

aws Search [Alt+S] Europe (Stockholm) xyz @ 3919-8450-2893

EC2 Instances

Instances (2) Info

Last updated less than a minute ago

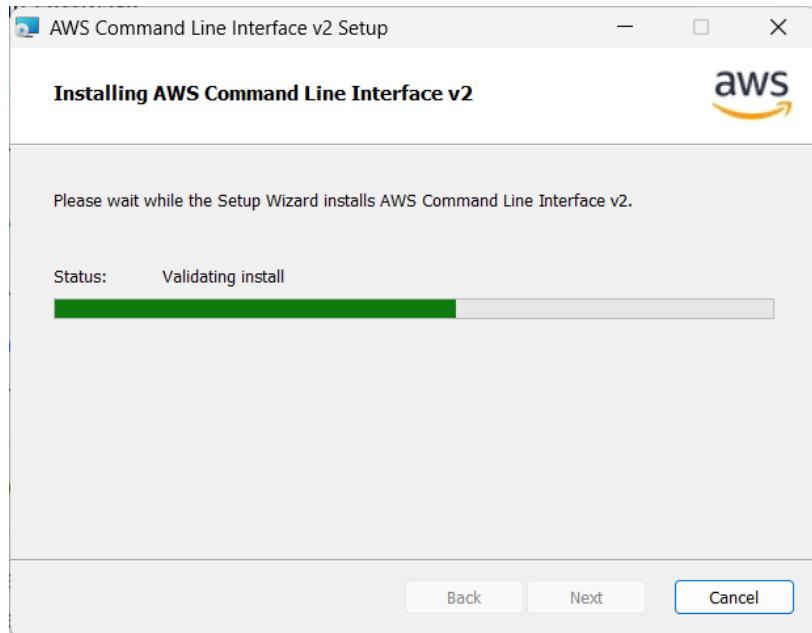
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
xyz	i-0a47ed518703e1570	Running	t3.micro	Initializing	View alarms +	eu-north-1b	ec2-13-51-2
first	i-00591a309fe06e3ff	Terminated	t3.micro	-	View alarms +	eu-north-1b	-

Select an instance

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

3. Access the IAM user with CLI and List IAM Users

Now go to browser type for aws cli tool download ->install the first link provided.



Open the command prompt

Type -> aws configure , it asks for access key and secret access key

Now create access key by moving to existing user> security credentials and choose access key > purpose CLI and click on create

Copy both access key and secret access key

A screenshot of a web browser showing the "Create access key" page in the AWS IAM console. The URL is "us-east-1.console.aws.amazon.com/iam/home?region=eu-north-1#/users/details/Sanjeeva/create-access-key". The page shows a green success message: "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." On the left, a sidebar lists steps: Step 1 (Access key best practices & alternatives), Step 2 (optional Set description tag), Step 3 (Retrieve access keys). The "Retrieve access keys" step is selected. It shows fields for "Access key" (AKIAVWRAY3BW4VDPHWOJ) and "Secret access key" (*****). Below is a section on "Access key best practices" with a bulleted list: "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.", and "Rotate access keys regularly.". At the bottom are "Download .csv file" and "Done" buttons.

```

Command Prompt
Microsoft Windows [Version 10.0.26100.4061]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Sanjeeva Sanku>aws --version
aws-cli/2.27.29 Python/3.13.3 Windows/11 exe/AMD64

C:\Users\Sanjeeva Sanku>aws configure
AWS Access Key ID [None]: AKIAVWRAY3BW4VDPHW0J
AWS Secret Access Key [None]: eQGp8DP86RTqG1kkyg7dDjGmiCM1AUWI4/qAKeMe
Default region name [None]: eu-north-1
Default output format [None]: json

C:\Users\Sanjeeva Sanku>aws iam list-users
An error occurred (AccessDenied) when calling the ListUsers operation: User: arn:aws:iam::391984502893:user/Sanjeeva is
not authorized to perform: iam>ListUsers on resource: arn:aws:iam::391984502893:user/ because no identity-based policy al
lows the iam>ListUsers action

C:\Users\Sanjeeva Sanku>

```

Create a inline policy under permissions of existing user as we can't see output because of no permission of ListUsers

Step 1
 Specify permissions
 Step 2
 Review and create

Specify permissions Info

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

Policy editor

```

1▼ {
2    "Version": "2012-10-17",
3    "Statement": [
4        {
5            "Sid": "Statement1",
6            "Effect": "Allow",
7            "Action": [
8                "iam>ListUsers"
9            ],
10           "Resource": "*"
11       ]
12   ]
13 }

```

Visual **JSON** Actions ▾

Edit statement
Statement1 Remove

Add actions
Choose a service
Filter services

Included
IAM

Available
AI Operations
AMP
API Gateway
API Gateway V2
ARC Zonal Shift
ASC

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Add 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, inline policies, and any existing permissions boundaries 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 (1/1352)

Filter by Type
<input type="text" value="iamre"/> All types 1 match
<input checked="" type="checkbox"/> Policy name <input type="text" value="iamre"/>
Type
Attached entities
<input checked="" type="checkbox"/> IAMReadOnlyAccess AWS managed 0

[Cancel](#) [Next](#)

Add permission of IAMReadOnlyAccess

Identity and Access Management (IAM)

Permissions

Permissions policies (3)

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

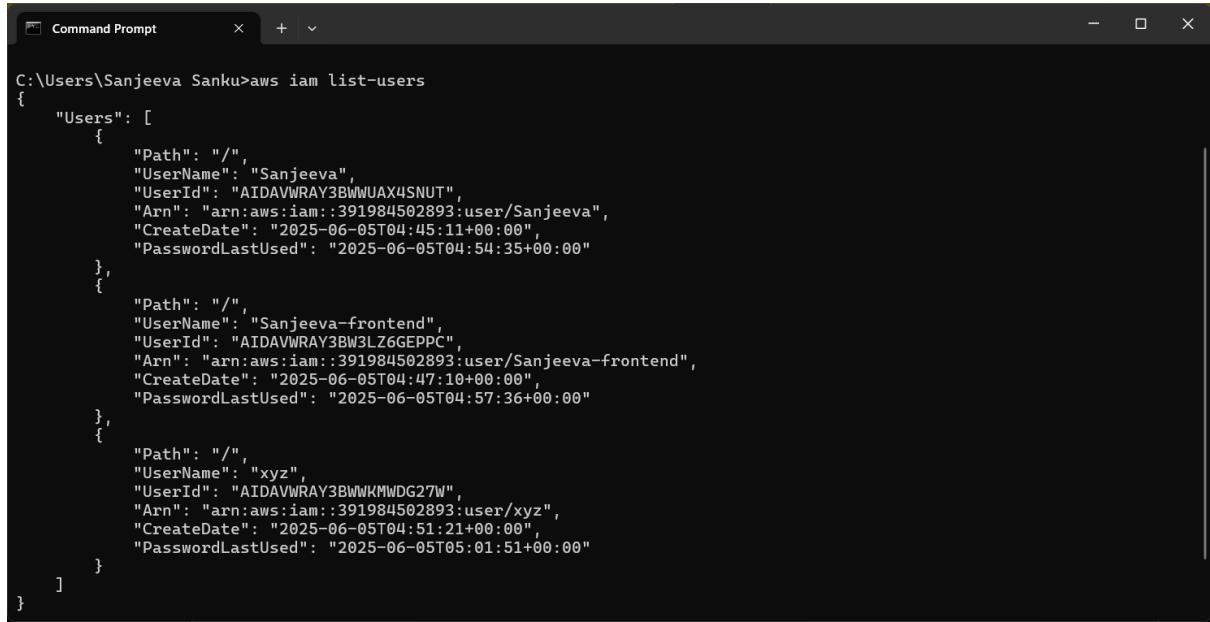
Filter by Type
<input type="text" value="Search"/> All types
<input checked="" type="checkbox"/> Policy name <input type="text" value="iamre"/>
Type
Attached via
<input type="checkbox"/> AmazonEC2FullAccess AWS managed Directly
<input type="checkbox"/> IAMReadOnlyAccess AWS managed Directly
<input type="checkbox"/> listUsers Customer inline Inline

Permissions boundary (not set)

Generate policy based on CloudTrail events

You can generate a new policy based on the access activity for this user, then customize, create, and attach it to this role. AWS uses your CloudTrail events to identify the services and

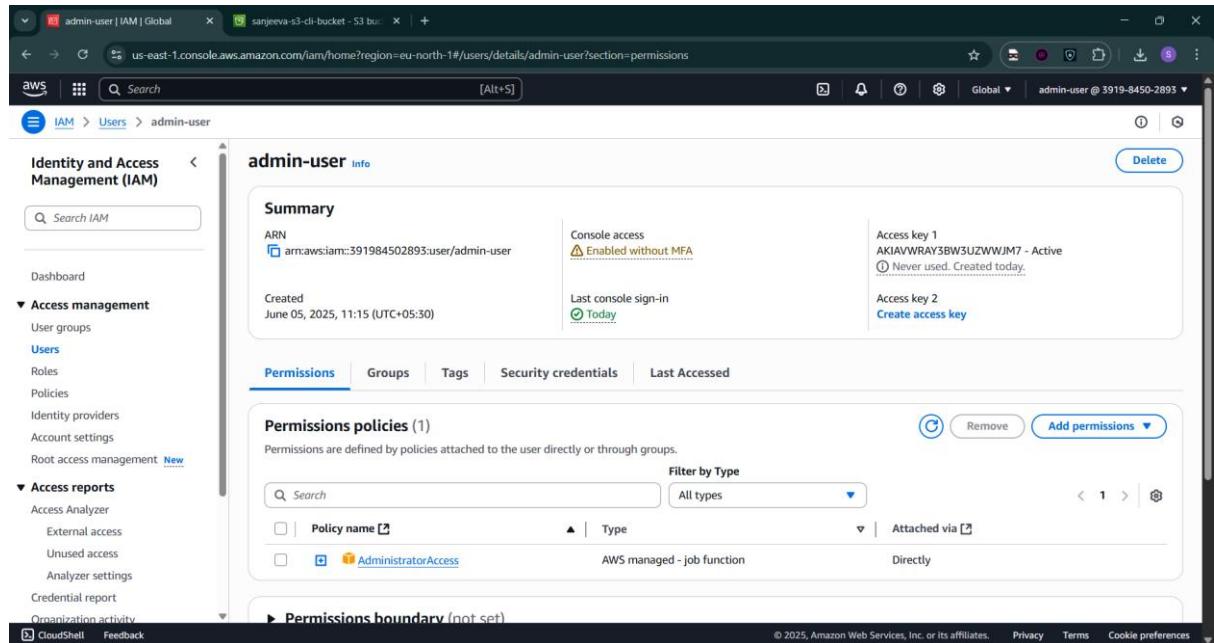
Now we got the list of users



```
C:\Users\Sanjeeva Sanku>aws iam list-users
{
    "Users": [
        {
            "Path": "/",
            "UserName": "Sanjeeva",
            "UserId": "AIDAVWRAY3BWWUAX4SNUT",
            "Arn": "arn:aws:iam:391984502893:user/Sanjeeva",
            "CreateDate": "2025-06-05T04:45:11+00:00",
            "PasswordLastUsed": "2025-06-05T04:54:35+00:00"
        },
        {
            "Path": "/",
            "UserName": "Sanjeeva-frontend",
            "UserId": "AIDAVWRAY3BW3LZ6GEPPC",
            "Arn": "arn:aws:iam:391984502893:user/Sanjeeva-frontend",
            "CreateDate": "2025-06-05T04:47:10+00:00",
            "PasswordLastUsed": "2025-06-05T04:57:36+00:00"
        },
        {
            "Path": "/",
            "UserName": "xyz",
            "UserId": "AIDAVWRAY3BWWKMWDG27W",
            "Arn": "arn:aws:iam:391984502893:user/xyz",
            "CreateDate": "2025-06-05T04:51:21+00:00",
            "PasswordLastUsed": "2025-06-05T05:01:51+00:00"
        }
    ]
}
```

4.Create IAM User Two with CLI

Created user using cli



The screenshot shows the AWS IAM console for the 'admin-user' user. The 'Summary' section displays the ARN (arn:aws:iam:391984502893:user/admin-user), which is highlighted. It also shows 'Console access' is enabled without MFA. There are two access keys listed: 'Access key 1' (Active, created today) and 'Access key 2' (Create access key). The 'Permissions' tab is selected, showing one policy attached: 'AdministratorAccess'. This policy is listed under 'Attached via' and is described as 'AWS managed - job function'.

Created one more user using cli

```
C:\Users\Sanjeeva Sanku>aws iam create-user --user-name admin-sanjeeva2
{
    "User": {
        "Path": "/",
        "UserName": "admin-sanjeeva2",
        "UserId": "AIDAVWRAY3BW4YGD5K7SQ",
        "Arn": "arn:aws:iam::391984502893:user/admin-sanjeeva2",
        "CreateDate": "2025-06-05T06:05:58+00:00"
    }
}

C:\Users\Sanjeeva Sanku>aws iam attach-user-policy --user-name admin-sanjeeva2 --policy-arn arn:aws:iam::aws:policy/AdministratorAccess
C:\Users\Sanjeeva Sanku>
```

The screenshot shows the AWS IAM User Details page for the user 'admin-sanjeeva2'. The 'Permissions' tab is active, displaying the attached policies. One policy, 'AdministratorAccess', is listed under 'Permissions policies (1)'. The policy details show it is an AWS managed - job function policy attached directly.

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

5) Create a IAM user with administrative full access and check whether created IAM user can create other IAM user with S3 full access and another IAM user with administrative full access with both CLI and GUI.

```
Command Prompt
C:\Users\Sanjeeda Sanku>aws iam create-user --user-name admin-sanjeeda2
{
    "User": {
        "Path": "/",
        "UserName": "admin-sanjeeda2",
        "UserId": "AIDAVWRAY3BW4YGD5K7SQ",
        "Arn": "arn:aws:iam::391984502893:user/admin-sanjeeda2",
        "CreateDate": "2025-06-05T06:05:58+00:00"
    }
}

C:\Users\Sanjeeda Sanku>aws iam attach-user-policy --user-name admin-sanjeeda2 --policy-arn arn:aws:iam::aws:policy/AdministratorAccess

C:\Users\Sanjeeda Sanku>aws iam create-user --user-name sanjeeda-s3-2
An error occurred (EntityAlreadyExists) when calling the CreateUser operation: User with name sanjeeda-s3-2 already exists.

C:\Users\Sanjeeda Sanku>aws iam create-user --user-name sanjeeda-s3-2
{
    "User": {
        "Path": "/",
        "UserName": "sanjeeda-s3-2",
        "UserId": "AIDAVWRAY3BWZEU2DALJ5",
        "Arn": "arn:aws:iam::391984502893:user/sanjeeda-s3-2",
        "CreateDate": "2025-06-05T06:09:05+00:00"
    }
}
```

The screenshot shows the AWS IAM User Details page for a user named 'admin-user'. The 'Summary' section displays the ARN (arn:aws:iam::391984502893:user/admin-user), which is highlighted with a red box. It also shows the creation date (June 05, 2025, 11:15 (UTC+05:30)), console access status (Enabled without MFA), and last sign-in information (Today). The 'Permissions' tab is selected, showing one policy attached: 'AdministratorAccess' (AWS managed - job function, Directly). The 'Permissions policies' section lists the attached policy. The 'Permissions boundary' section is noted as 'not set'.

6)Create the bucket in S3 and copy the contents from CLI

```
Command Prompt
    "CreateDate": "2025-06-05T06:09:05+00:00"
}

C:\Users\Sanjeeva Sanku>aws iam attach-user-policy --user-name sanjeeva-s3-2 --policy-arn arn:aws:iam::aws:policy/AmazonS3FullAccess

C:\Users\Sanjeeva Sanku>aws s3 mb s3://sanjeeva-s3-cli-bucket
make_bucket failed: s3://sanjeeva-s3-cli-bucket An error occurred (AccessDenied) when calling the CreateBucket operation
: User: arn:aws:iam::391984502893:user/Sanjeeva is not authorized to perform: s3:CreateBucket on resource: "arn:aws:s3:::sanjeeva-s3-cli-bucket" because no identity-based policy allows the s3:CreateBucket action

C:\Users\Sanjeeva Sanku>aws configure
AWS Access Key ID [*****HWOJ]: AKIAWRAY3BW4WF42ZU
AWS Secret Access Key [*****KeMe]: c+3XPGf7XGUU+sE2/I54uQSJvAGpmXfo2pqzXR/r
Default region name [eu-north-1]:
Default output format [json]

C:\Users\Sanjeeva Sanku>aws s3 mb s3://sanjeeva-s3-cli-bucket
make_bucket: sanjeeva-s3-cli-bucket

C:\Users\Sanjeeva Sanku>aws s3 cp C:\Users\Sanjeeva Sanku\OneDrive\Desktop\kwaishain.txt s3://sanjeeva-s3-cli-bucket/
Unknown options: s3://sanjeeva-s3-cli-bucket/

C:\Users\Sanjeeva Sanku>aws s3 cp "C:\Users\Sanjeeva Sanku\OneDrive\Desktop\kwaishain.txt" s3://sanjeeva-s3-cli-bucket/
upload: OneDrive\Desktop\kwaishain.txt to s3://sanjeeva-s3-cli-bucket/kwaishain.txt

C:\Users\Sanjeeva Sanku>
```

The screenshot shows the AWS S3 console interface. The left sidebar lists various services like General purpose buckets, Storage Lens, and IAM Access Analyzer. The main content area displays the 'sanjeeva-s3-cli-bucket' page. At the top, there are tabs for Objects, Properties, Permissions, Metrics, Management, and Access Points. Below this, there's a toolbar with actions like Copy S3 URI, Copy URL, Download, Open, Delete, Actions (with a dropdown), Create folder, and Upload. A search bar is also present. The main content area shows a table of objects. The table has columns: Name, Type, Last modified, Size, and Storage class. One object is listed: 'kwaishain.txt' (Type: txt, Last modified: June 5, 2025, 11:45:08 (UTC+05:30), Size: 253.0 B, Storage class: Standard).