**AWS CLI (Command Line Interface) Guide**

Connecting from CLI:

* **Access Keys:** Obtain Access Key ID and Secret Access Key from IAM dashboard.
* **Generation of Root Access Keys:**
  + Go to IAM dashboard > Quick Links > My access key > Create new Access key.
  + Get Access Key ID and Secret Access Key

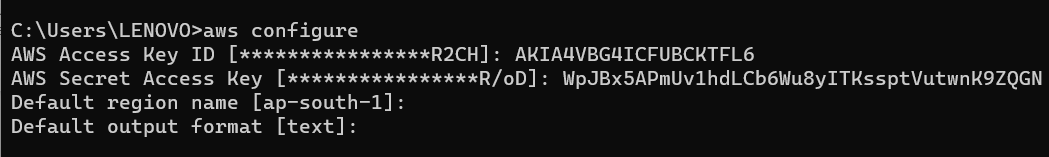
**Installing AWS CLI on Windows:**

* Google "Aws cli tool for windows."
* Visit docs.amazon.com, download and install the AWS CLI MSI installer for windows.

**Configuring AWS CLI:**

* Open Command Prompt and run:
* aws configure

Enter Access Key ID, Secret Access Key, Default region name and Default output format.



Note: ap-south-1 (Is nothing but region name of Mumbai)

We get the prompt. (We have logged in)

Search in google " aws s3 commands"

(Note: we do not need to remember commands)

Under available commands > select ls > under Examples

**Basic AWS CLI Commands:**

* **List S3 buckets:**

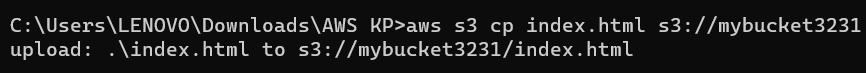
aws s3 ls

* **Create a new S3 bucket:**

aws s3 mb s3://mybucketeee

* **Upload a file to the bucket:**

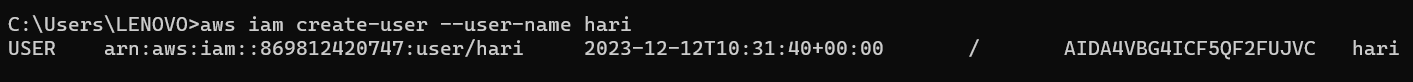
aws s3 cp test.txt s3://mybucketeee



**User Creation from CLI:**

* Google "aws iam commands."
* Create a new IAM user (e.g., "hari"):

aws iam create-user --user-name hari



Now, user is created from the command line.

We can also check from console.

We have 1000's of commands in AWS

Most of the operations we can also do it from CLI

We do not need to remember the commands

We have experienced AWS CLI from windows machine.

**Working with AWS CLI on Linux:**

* Create and connect to an EC2 machine using.

Services >EC2 > Amazon Linux > security group (open SSH and HTTP) -- Launch.

* AWS CLI is pre-installed on EC2; no need to install.
* Configure AWS CLI on Linux:

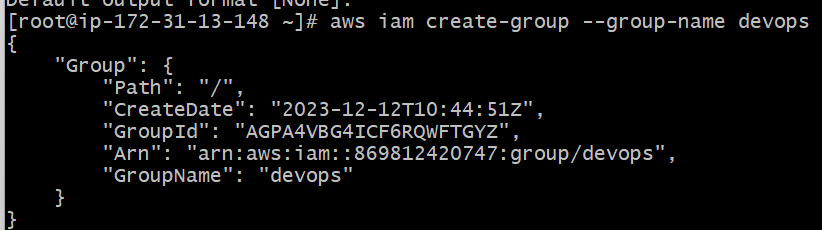
$ sudo su

# yum update -y

# aws configure

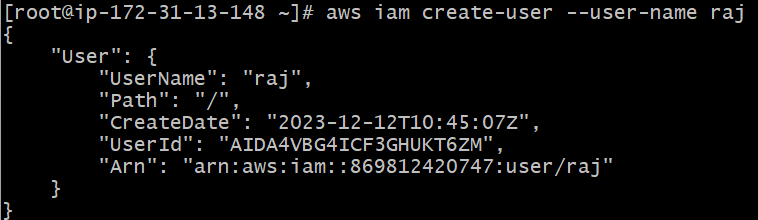
**Create group**

# aws iam create-group --group-name devops



**Create user**

# aws iam create-user --user-name raj

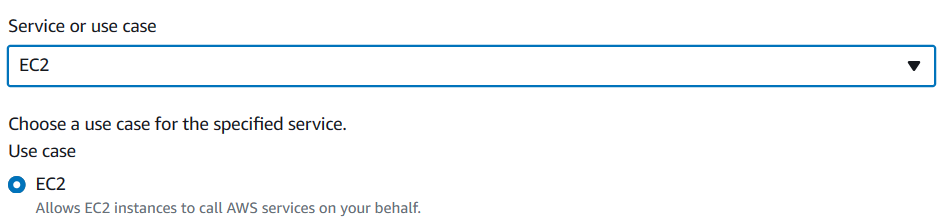


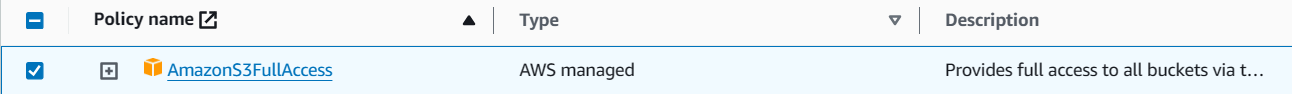
(We can verify users and group from console)

**IAM Roles**

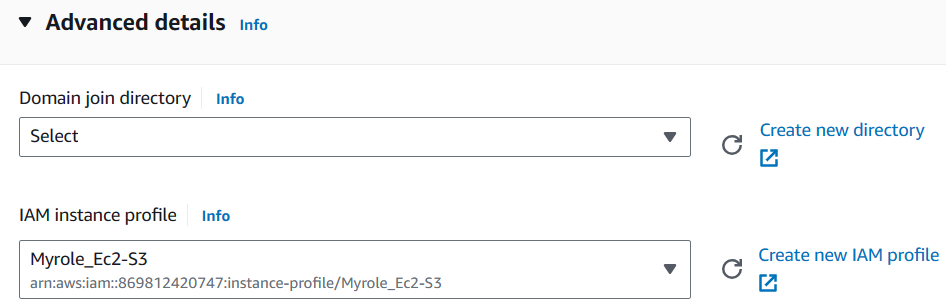
Role is replacement of credentials.

1. **Creating IAM Roles:** Roles are temporary security credentials assigned to EC2 instances. Create a role through the IAM dashboard by selecting EC2 as the source and S3FullAccess as the permission.





**2. Attaching Roles to EC2 Instances:** Launch a new EC2 instance and choose your created role in the IAM Role section. This grants the instance temporary access to specific resources.



As we have attached role to EC2 machine

We do not require aws configure commands, access key, secret access key

$ sudo su

# yum update -y

# aws s3 ls

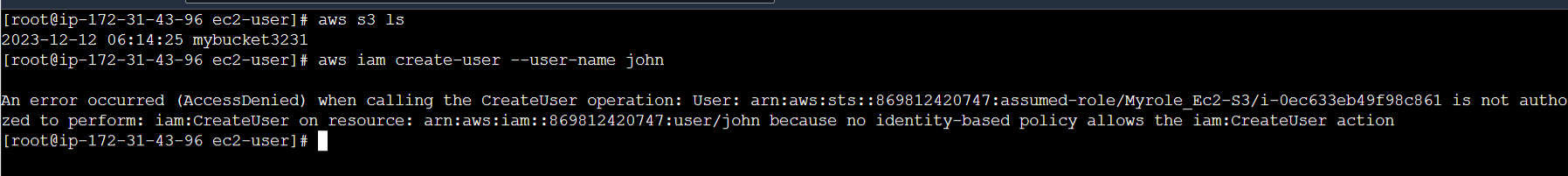
We see the list of all the buckets.

Create user

# aws iam create-user --user-name john

Is it possible?

An error occurred (AccessDenied)

 Now, I want to give IAM Access also

Select role > Attach policies > Select IAMFullAccess > Attach policy

Now, the role is having two policies

So, can we create new users now

# aws iam create-user --user-name test

Yes!!

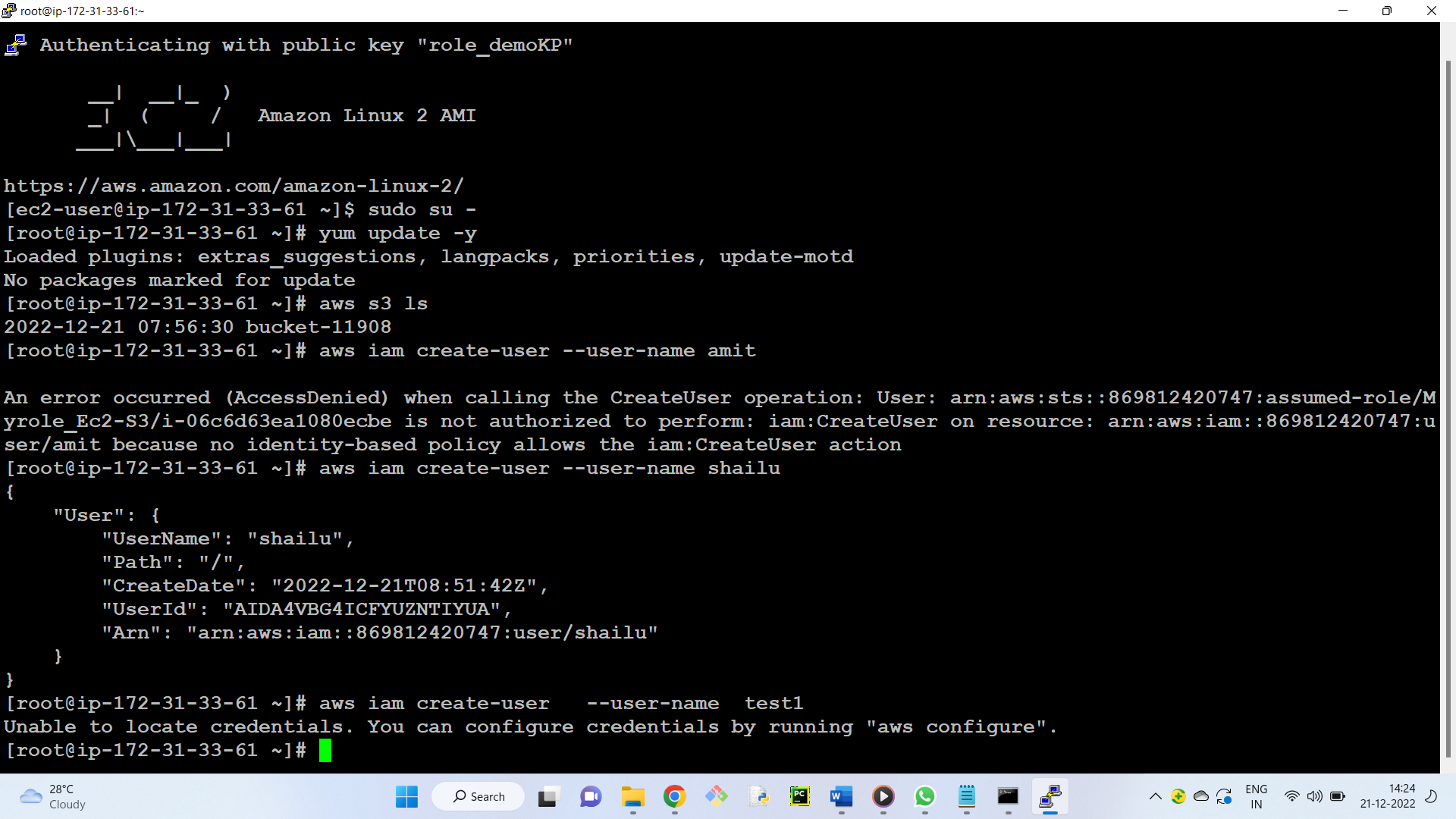
I want to detach the role from EC2 machine

Select the EC2 machine > Actions > security > modify iam role - No Role > Update IAM Role

Now, if you try to create new user

# aws iam create-user --user-name test1

Unable to locate credentials.



Roles are replacement of credentials!!

Note: Roles can be attached to EC2 machines only.

Not to our personal laptops.

**Benefits of Roles:** Roles eliminate the need for managing access keys on individual instances, improving security and simplifying management.

**Deletion steps**

1) Delete root access Key

Manage security credentials --> Access Keys --> Delete -- Yes

2) Delete all groups

3) Delete all users

4) Delete IAM role

5) Delete buckets

5) Delete EC2 machines