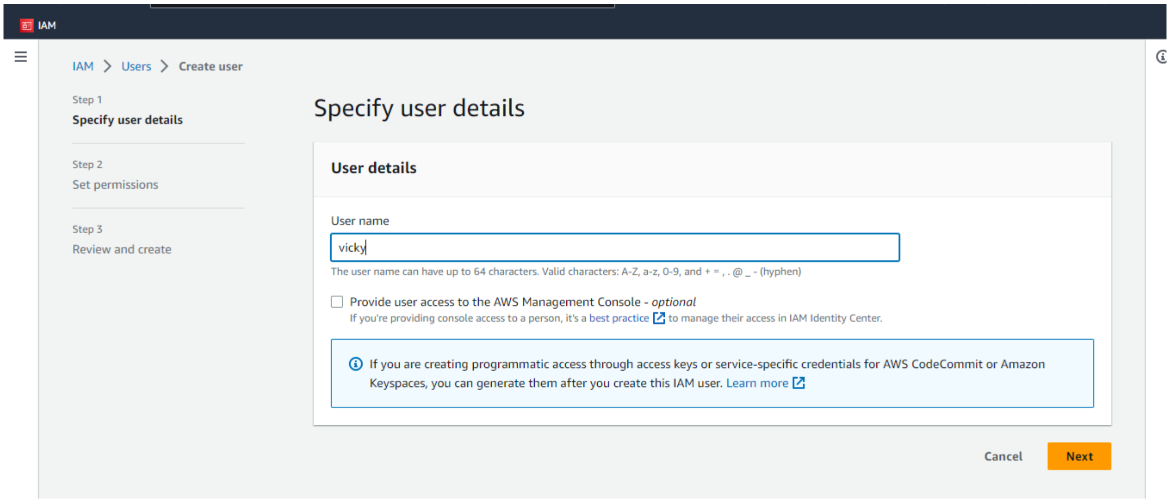
HOW TO UPLOAD A FILE IN S3 BUCKET USING AWS CLI

1. Create a new IAM User.



Step 1
Specify user details

Step 2
Set permissions

Step 3
Review and create

Specify user details

User details

User name
vicky

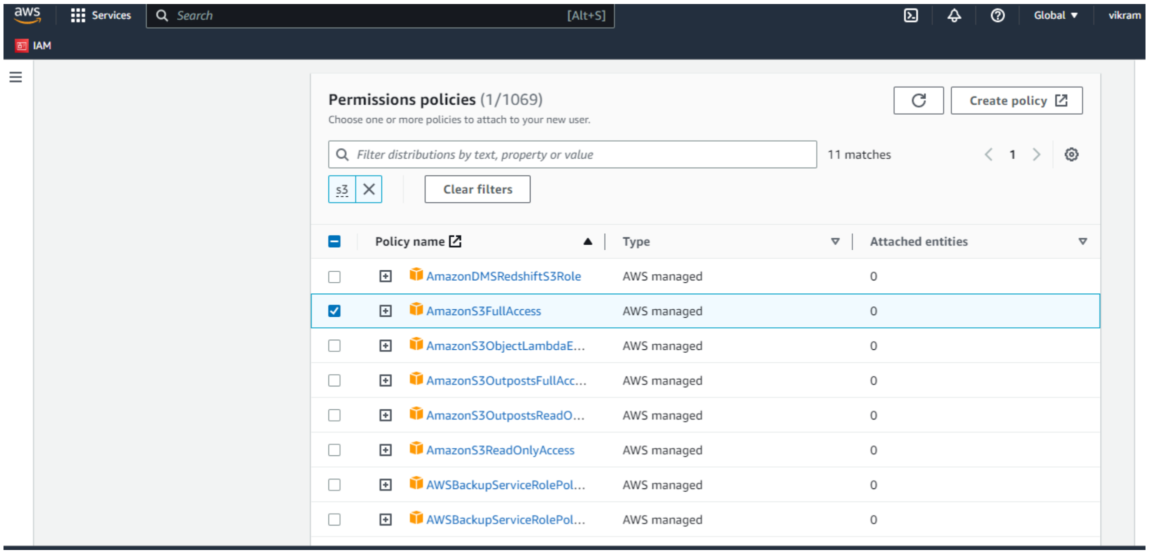
The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = , . @ _ - (hyphen)

□ 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.

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

2. Enter the name and click on “Next”

3. Here, we need to select “Attach Policies directly” and click on “Create Policy”.

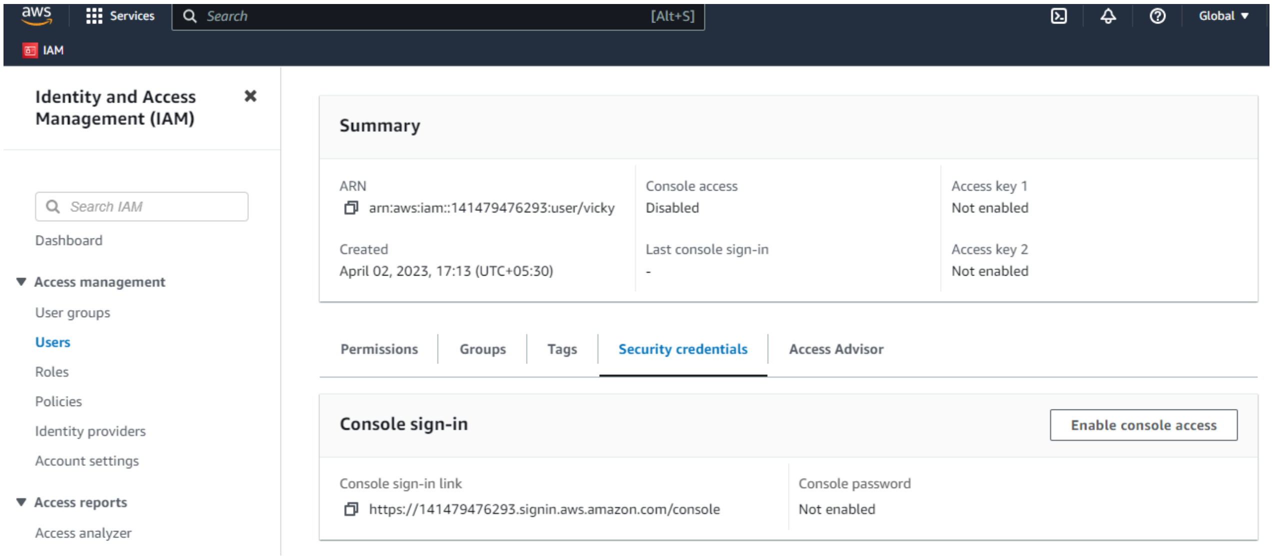


Permissions policies (1/1069)

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

	Policy name	Type	Attached entities
□	AmazonDMSRedshiftS3Role	AWS managed	0
☑	AmazonS3FullAccess	AWS managed	0
□	AmazonS3ObjectLambdaE...	AWS managed	0
□	AmazonS3OutpostsFullAcc...	AWS managed	0
□	AmazonS3OutpostsReadO...	AWS managed	0
□	AmazonS3ReadOnlyAccess	AWS managed	0
□	AWSBackupServiceRolePol...	AWS managed	0
□	AWSBackupServiceRolePol...	AWS managed	0

4.Now, go to “Security Credentials”.



Summary

ARN
arn:aws:iam::141479476293:user/vicky

Console access
Disabled

Access key 1
Not enabled

Created
April 02, 2023, 17:13 (UTC+05:30)

Last console sign-in
-

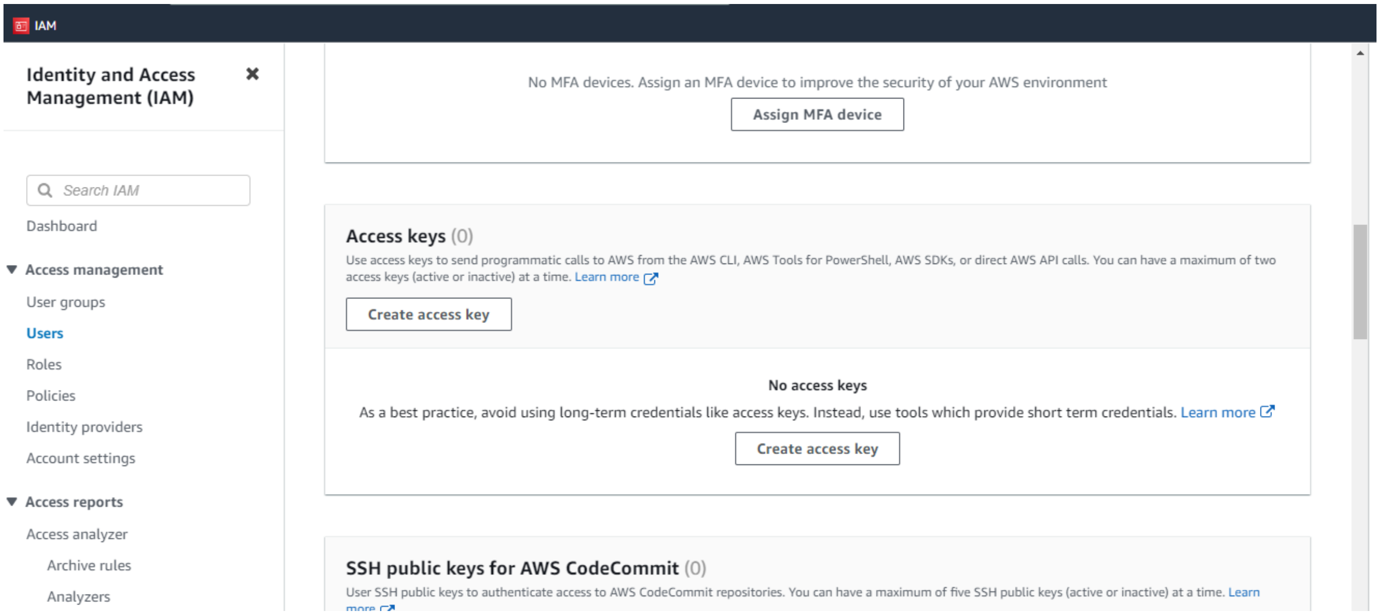
Access key 2
Not enabled

Console sign-in

Console sign-in link
https://141479476293.signin.aws.amazon.com/console

Console password
Not enabled

5.In the “Access Keys” and click on “Create Keys”



No MFA devices. Assign an MFA device to improve the security of your AWS environment

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 more

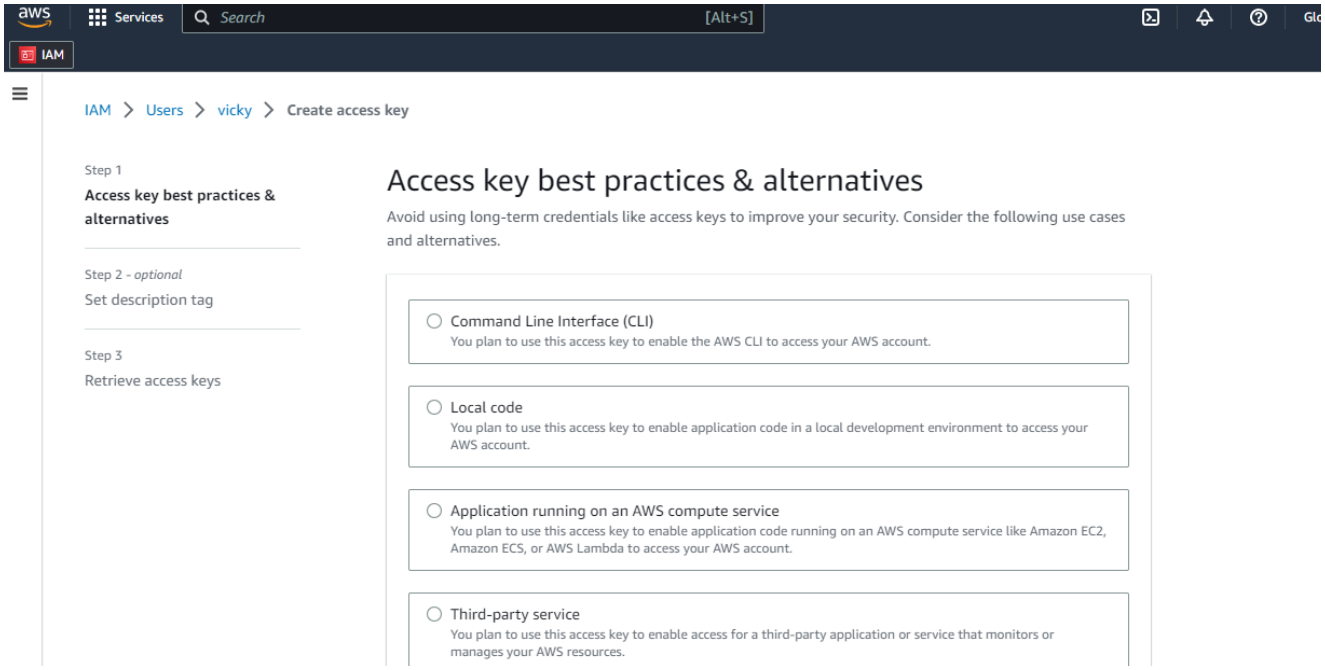
No access keys

As a best practice, avoid using long-term credentials like access keys. Instead, use tools which provide short term credentials. Learn more

SSH public keys for AWS CodeCommit (0)

User SSH public keys to authenticate access to AWS CodeCommit repositories. You can have a maximum of five SSH public keys (active or inactive) at a time. Learn more

6. Here select “Command Line Interface CLI”.



IAM > Users > vicky > 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.

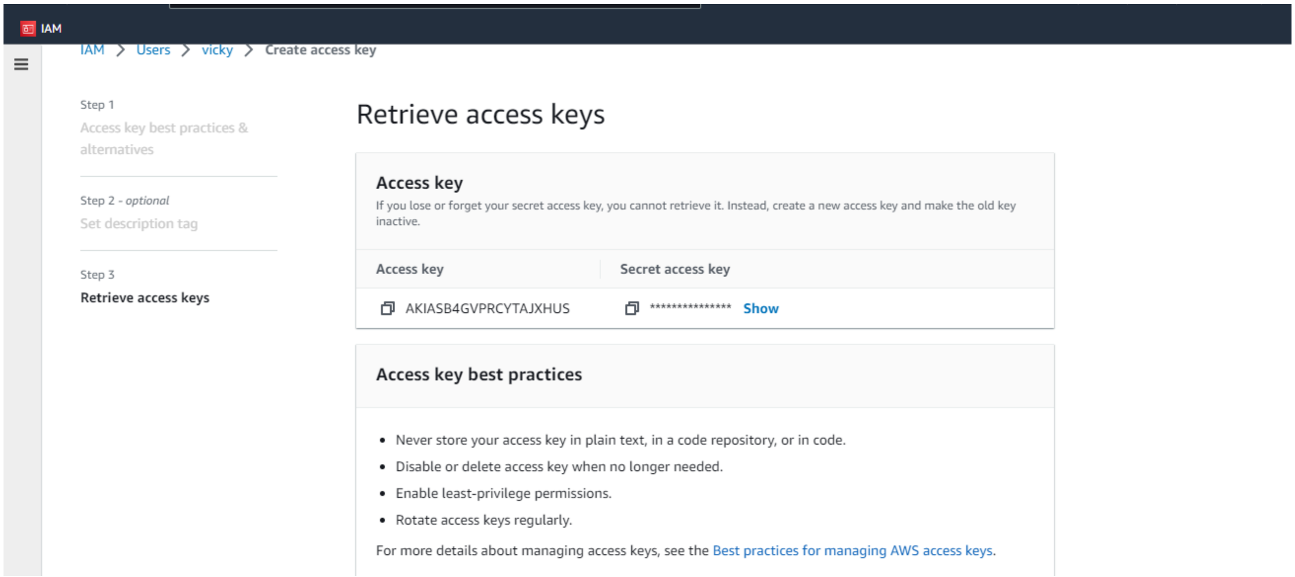
○ Command Line Interface (CLI)
You plan to use this access key to enable the AWS CLI to access your AWS account.

○ 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.

7.You will get the Access Key and Secret.



IAM > Users > vicky > 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.

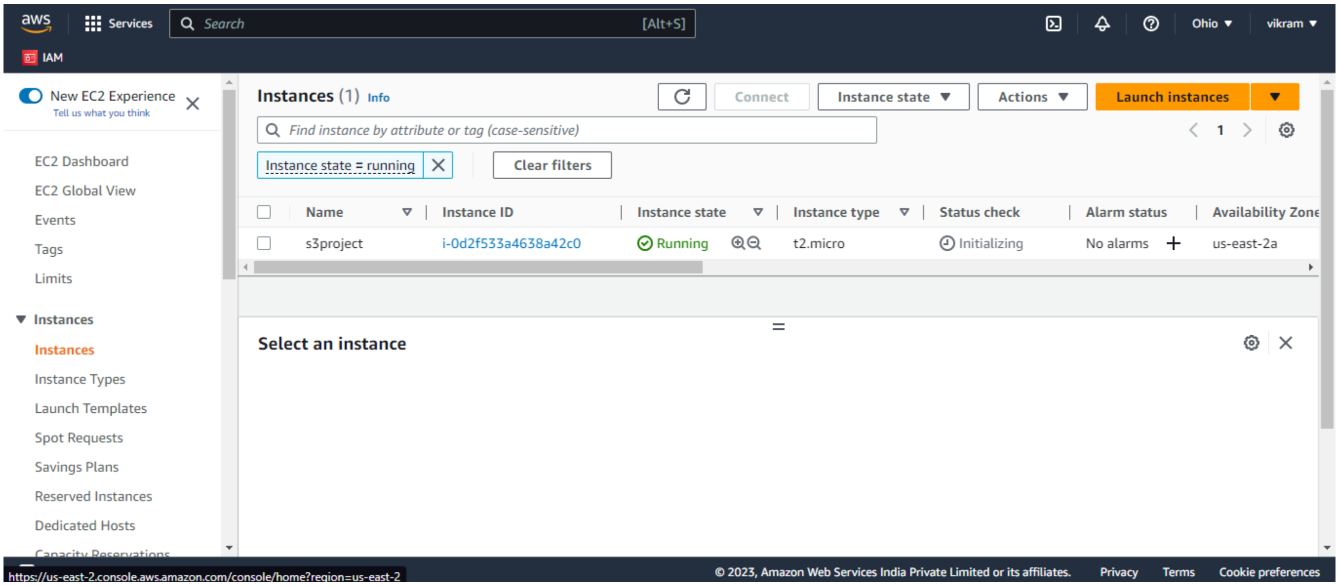
Access key	Secret access key
AKIASB4GVPRCYTAJXHUS	*************** Show

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.

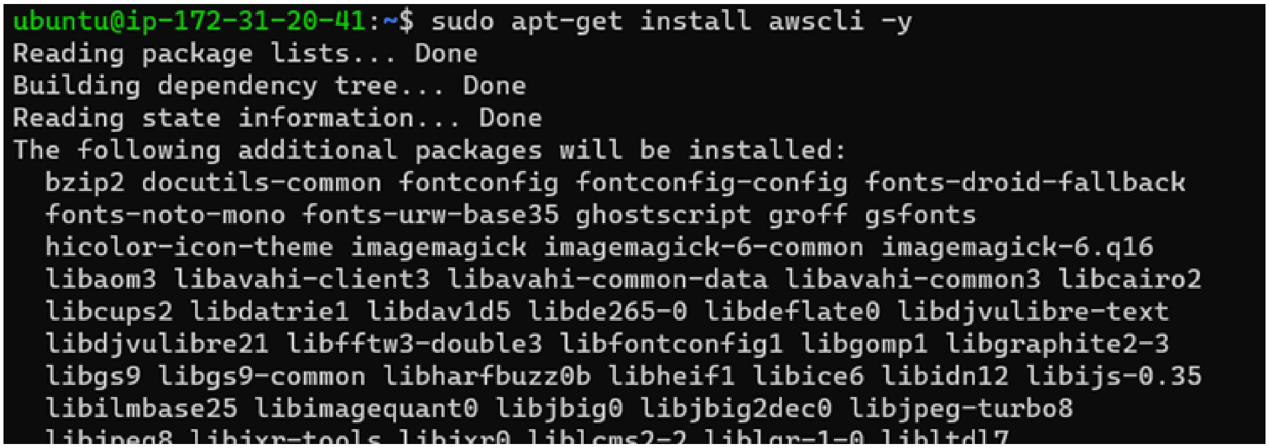
8.Create a new “t2.micro” instance.



Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
s3project	i-0d2f533a4638a42c0	Running	t2.micro	Initializing	No alarms	us-east-2a

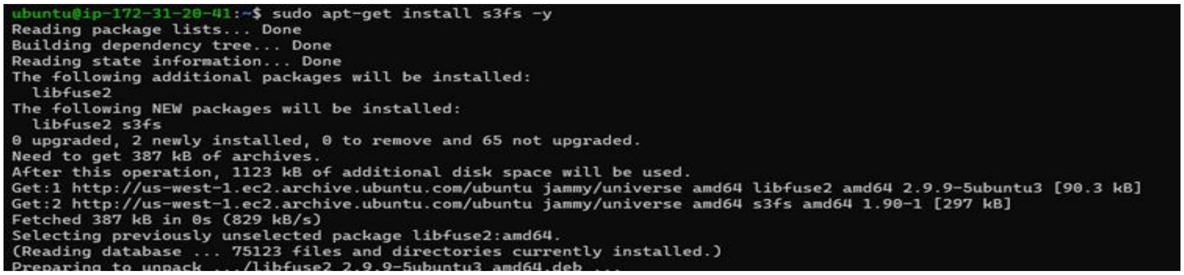
9.SSH to the server.

10.Install AWS CLI.



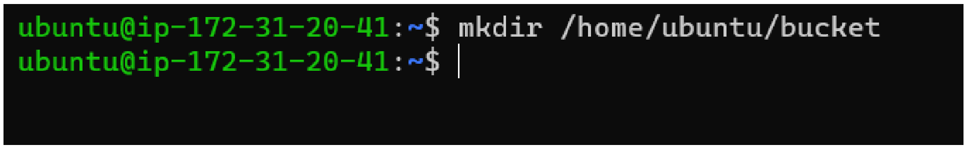
```
ubuntu@ip-172-31-20-41:~$ sudo apt-get install awscli -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  bzip2 docutils-common fontconfig fontconfig-config fonts-droid-fallback
  fonts-noto-mono fonts-urw-base35 ghostscript groff gsfonts
  hicolor-icon-theme imagemagick imagemagick-6-common imagemagick-6.q16
  libaom3 libavahi-client3 libavahi-common-data libavahi-common3 libcairo2
  libcups2 libdatrie1 libdav1d5 libde265-0 libdeflate0 libdjvulibre-text
  libdjvulibre21 libfftw3-double3 libfontconfig1 libgomp1 libgraphite2-3
  libgs9 libgs9-common libharfbuzz0b libheif1 libice6 libidn12 libijs-0.35
  libilmbase25 libimagequant0 libjbig0 libjbig2dec0 libjpeg-turbo8
  libjpeg8 libjxr-tools libjxr0 liblcms2-2 liblqr-1-0 libltdl7
```

11.After installation of AWS CLI, we will now Install the s3fs.



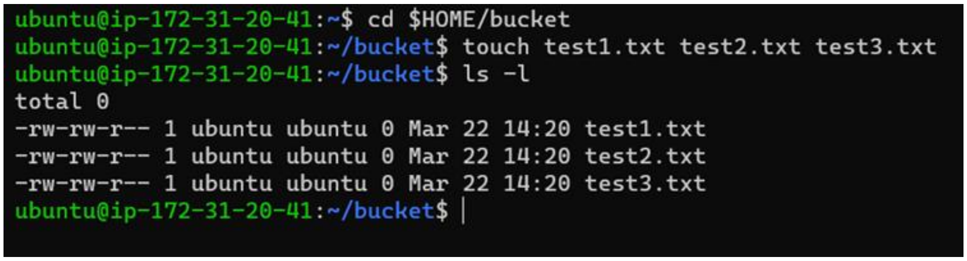
```
ubuntu@ip-172-31-20-41:~$ sudo apt-get install s3fs -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libfuse2
The following NEW packages will be installed:
  libfuse2 s3fs
0 upgraded, 2 newly installed, 0 to remove and 65 not upgraded.
Need to get 387 kB of archives.
After this operation, 1123 kB of additional disk space will be used.
Get:1 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 libfuse2 amd64 2.9.9-5ubuntu3 [90.3 kB]
Get:2 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 s3fs amd64 1.90-1 [297 kB]
Fetched 387 kB in 0s (829 kB/s)
Selecting previously unselected package libfuse2:amd64.
(Reading database ... 75123 files and directories currently installed.)
Preparing to unpack .../libfuse2_2.9.9-5ubuntu3_amd64.deb ...
```

12. Create a folder named “bucket” to the location “/home/ubuntu”



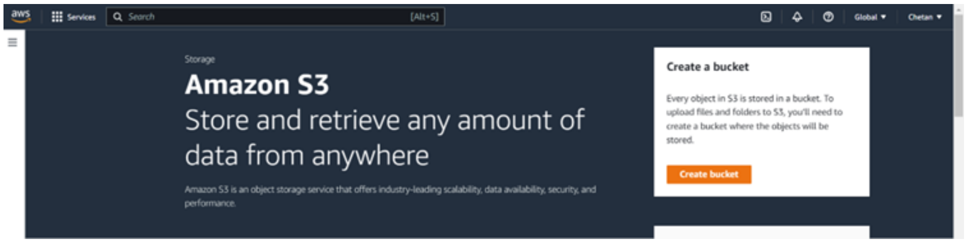
```
ubuntu@ip-172-31-20-41:~$ mkdir /home/ubuntu/bucket
ubuntu@ip-172-31-20-41:~$ 
```

13. Now, we will add 2-3 files in “/home/ubuntu/bucket”.



```
ubuntu@ip-172-31-20-41:~$ cd $HOME/bucket
ubuntu@ip-172-31-20-41:~/bucket$ touch test1.txt test2.txt test3.txt
ubuntu@ip-172-31-20-41:~/bucket$ ls -l
total 0
-rw-rw-r-- 1 ubuntu ubuntu 0 Mar 22 14:20 test1.txt
-rw-rw-r-- 1 ubuntu ubuntu 0 Mar 22 14:20 test2.txt
-rw-rw-r-- 1 ubuntu ubuntu 0 Mar 22 14:20 test3.txt
ubuntu@ip-172-31-20-41:~/bucket$ 
```

14. Now, go to the AWS console and create a bucket.



Storage

Amazon S3

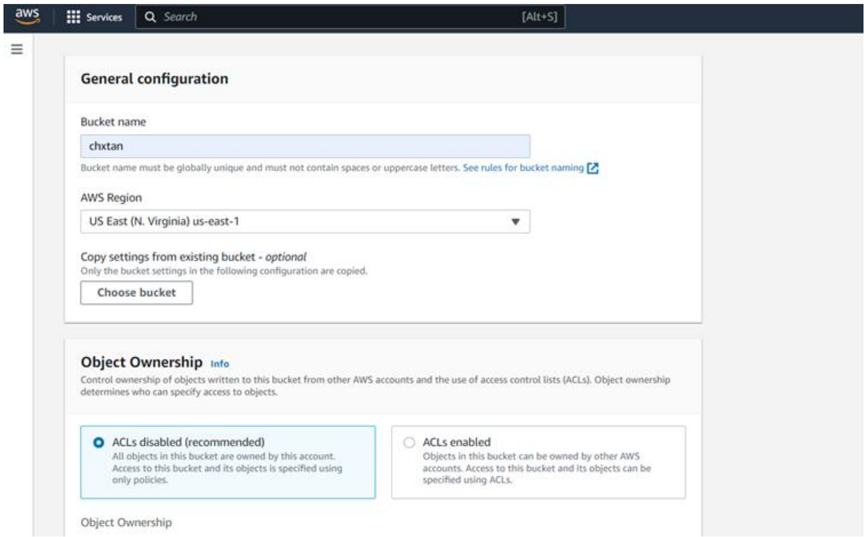
Store and retrieve any amount of data from anywhere

Amazon S3 is an object storage service that offers industry-leading scalability, data availability, security, and performance.

Create a bucket

Every object in S3 is stored in a bucket. To upload files and folders to S3, you'll need to create a bucket where the objects will be stored.

15. Give the bucket a name accordingly.



General configuration

Bucket name

chxtan

Bucket name must be globally unique and must not contain spaces or uppercase letters. See rules for bucket naming

AWS Region

US East (N. Virginia) us-east-1

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

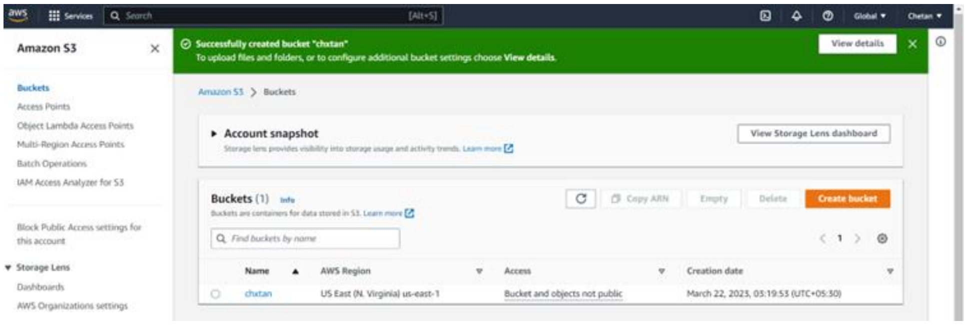
Object Ownership Info

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

● ACLs disabled (recommended)
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

○ ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership



Successfully created bucket "chxtan"
To upload files and folders, or to configure additional bucket settings choose **View details**.

Amazon S3 > Buckets

Account snapshot
Storage lens provides visibility into storage usage and activity trends. Learn more

Buckets (1) Info
Buckets are containers for data stored in S3. Learn more

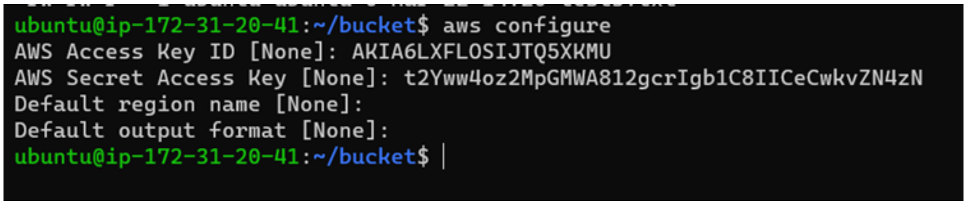
	Name	AWS Region	Access	Creation date
○	chxtan	US East (N. Virginia) us-east-1	Bucket and objects not public	March 22, 2023, 03:19:53 (UTC+05:30)

aws
Services
Search
[Alt+S]
Global
Chetan
Amazon S3
Buckets
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
AWS Organizations settings
View details
View Storage Lens dashboard
Copy ARN
Empty
Delete
Create bucket
Find buckets by name
1

16. Jump back to the VM and configure the AWS CLI, by running the command

“aws configure”

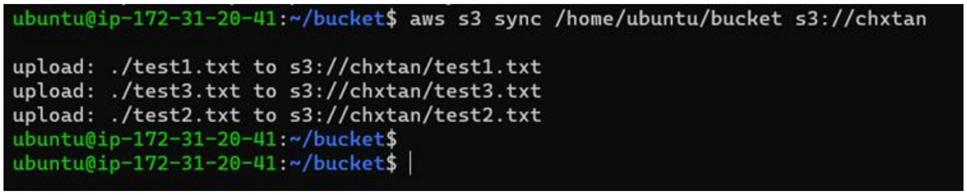
Provide the username and password that we created in Step 7.



```
ubuntu@ip-172-31-20-41:~/bucket$ aws configure
AWS Access Key ID [None]: AKIA6LXFLOSIJTQ5XKMU
AWS Secret Access Key [None]: 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
Default region name [None]:
Default output format [None]:
ubuntu@ip-172-31-20-41:~/bucket$ |
```

17. Now, run the command

“aws s3 sync <location_of_files> <s3://bucket_name>”

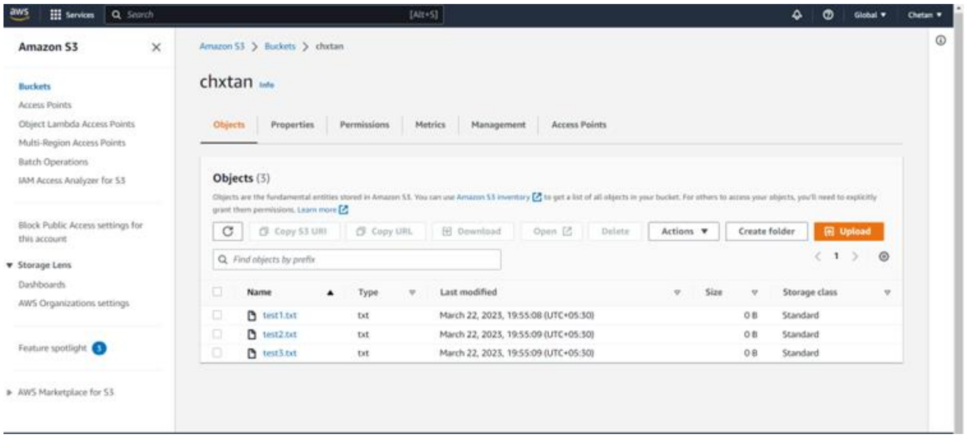


```
ubuntu@ip-172-31-20-41:~/bucket$ aws s3 sync /home/ubuntu/bucket s3://chxtan

upload: ./test1.txt to s3://chxtan/test1.txt
upload: ./test3.txt to s3://chxtan/test3.txt
upload: ./test2.txt to s3://chxtan/test2.txt
ubuntu@ip-172-31-20-41:~/bucket$
ubuntu@ip-172-31-20-41:~/bucket$ |
```

18. All the files present inside the given location in AWS EC2, will be uploaded to the S3 bucket.

19. Now, refresh the objects inside the bucket, now you can see all the files that were in the EC2 will be in the S3 bucket.



Objects (3)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name ▲	Type	Last modified	Size	Storage class
□	test1.txt	txt	March 22, 2023, 19:55:08 (UTC+05:30)	0 B	Standard
□	test2.txt	txt	March 22, 2023, 19:55:09 (UTC+05:30)	0 B	Standard
□	test3.txt	txt	March 22, 2023, 19:55:09 (UTC+05:30)	0 B	Standard