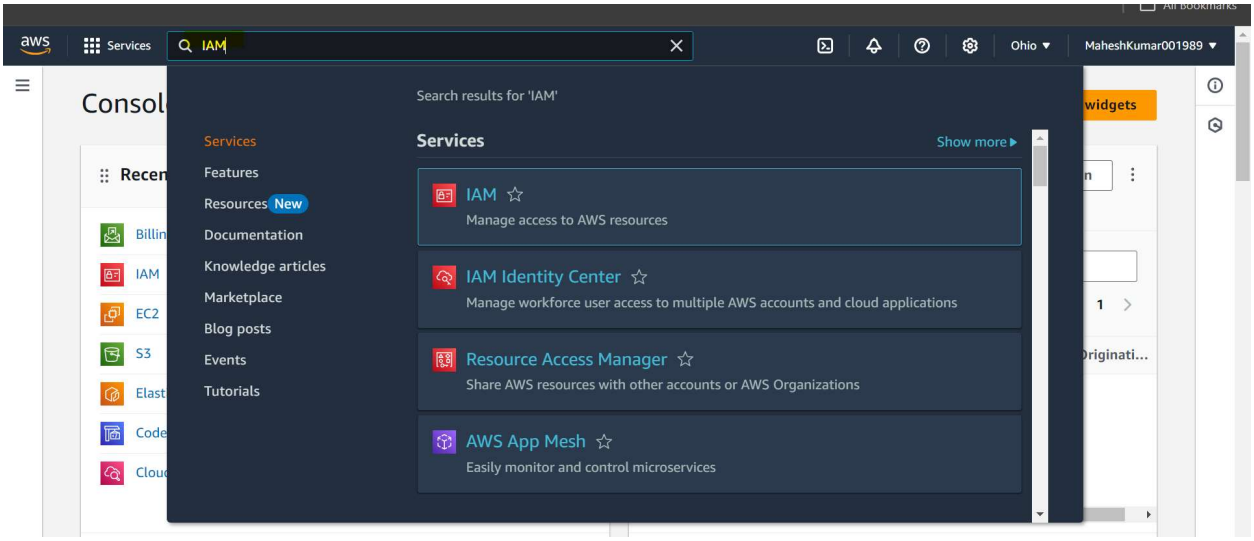
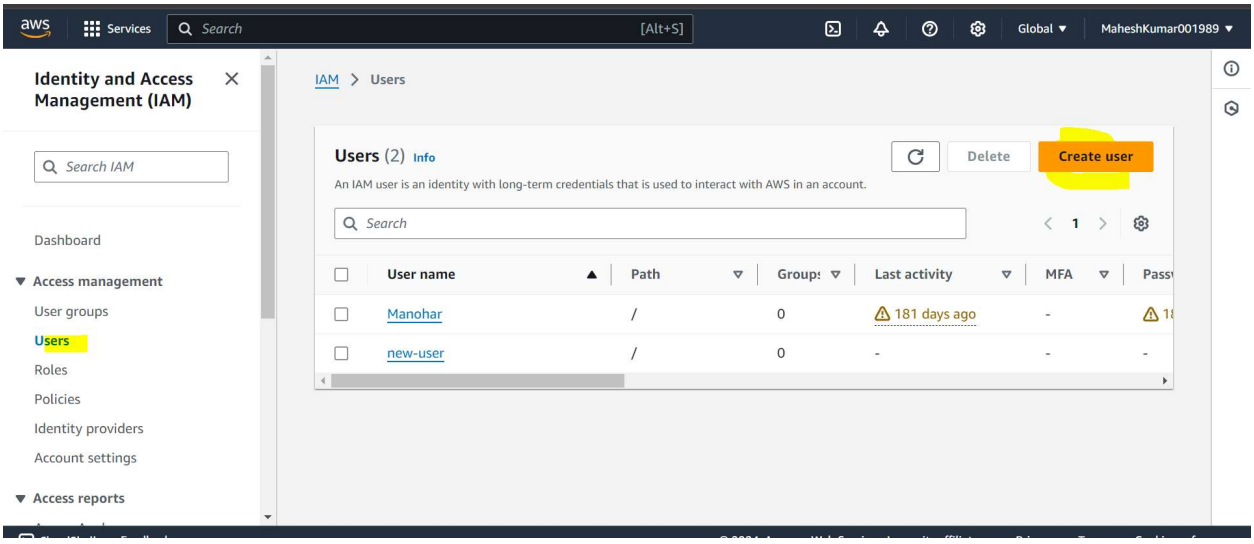
Create a IAM User

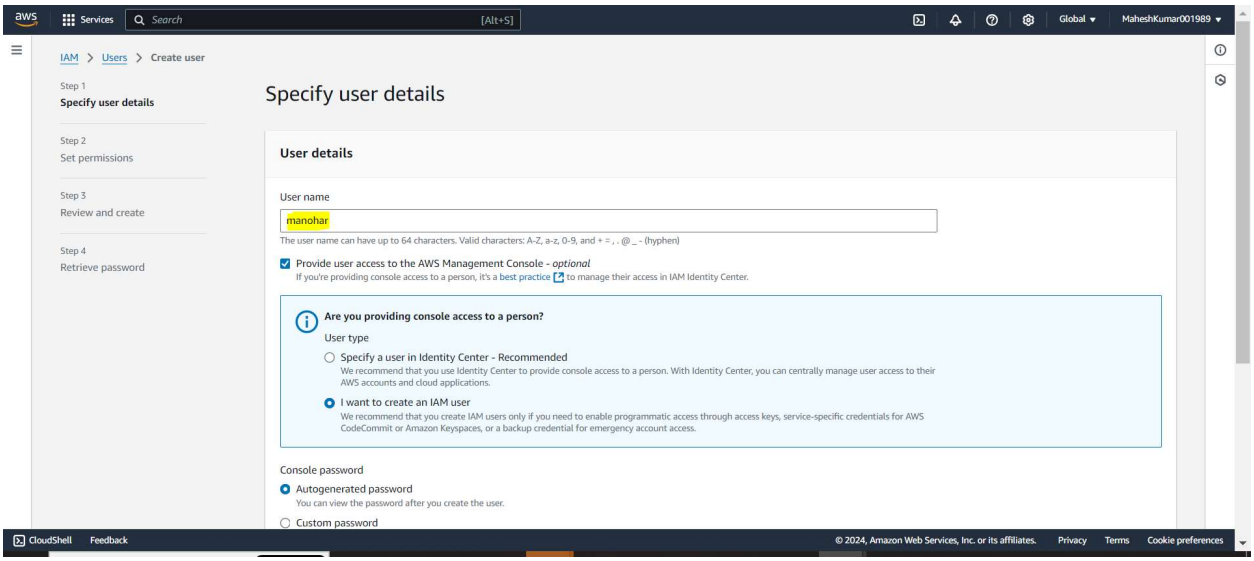
Login to aws console and search for IAM



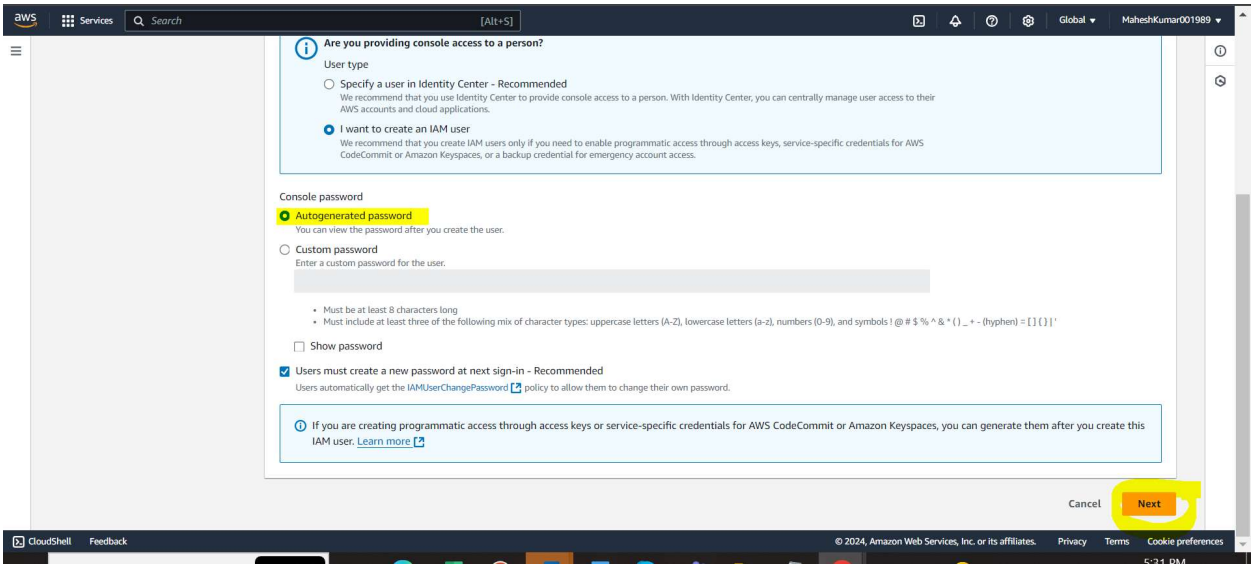
Click on users and click on create a user



Give a name to your IAM user, check the box “provide user access to the aws management console” and select option I want to create an IAM user.

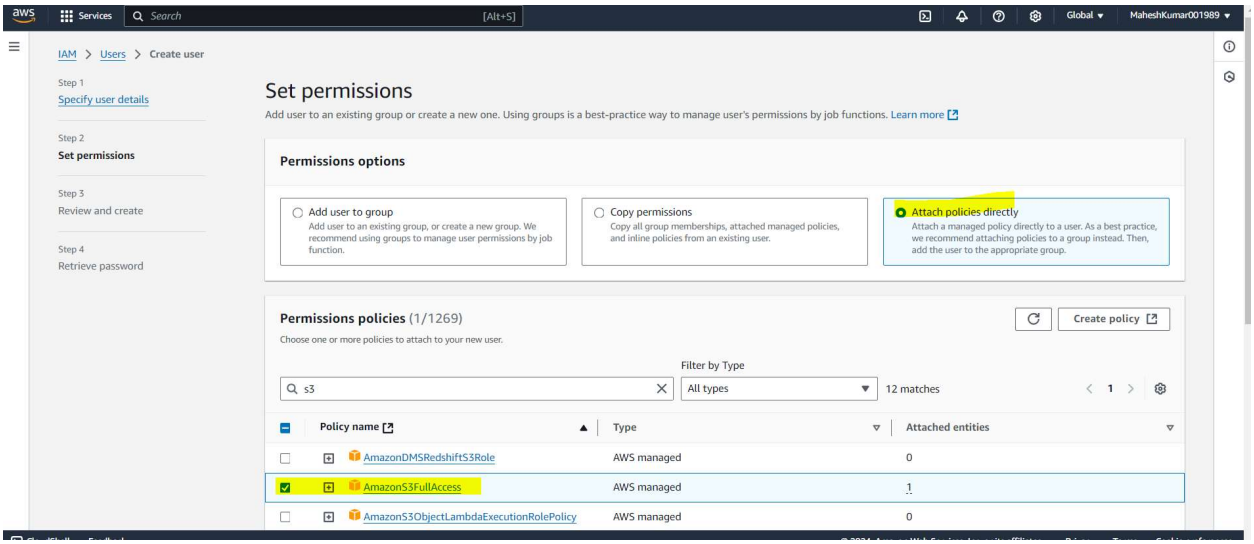


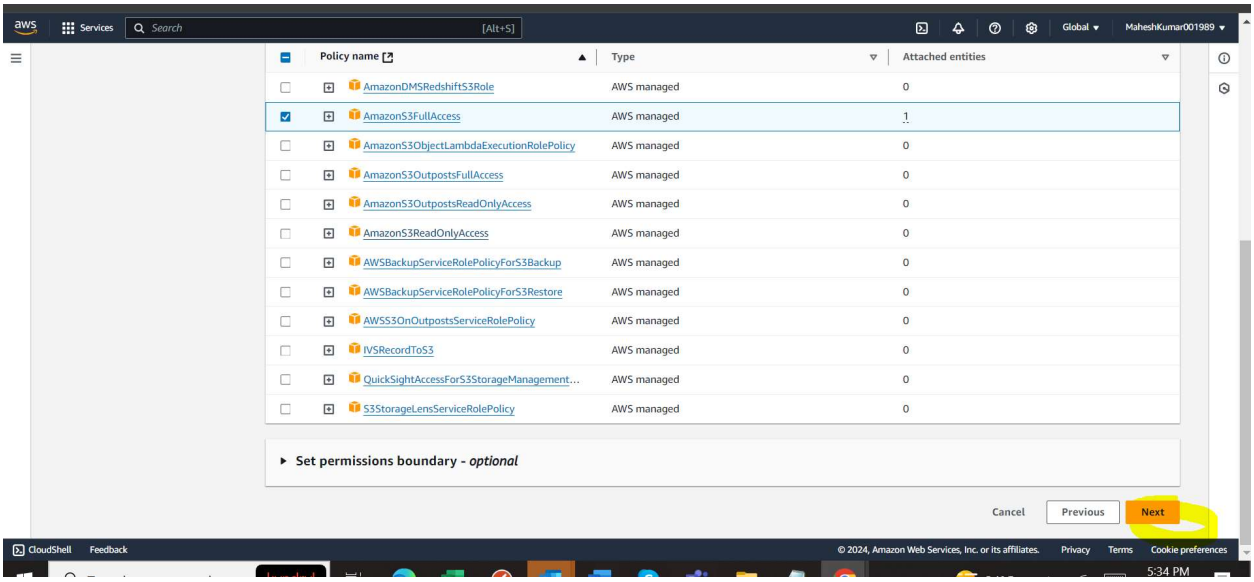
We have option to create password so either you can create a custom password for your IAM user or choose option autogenerated password but for this time I choose option autogenerated password and click on next.



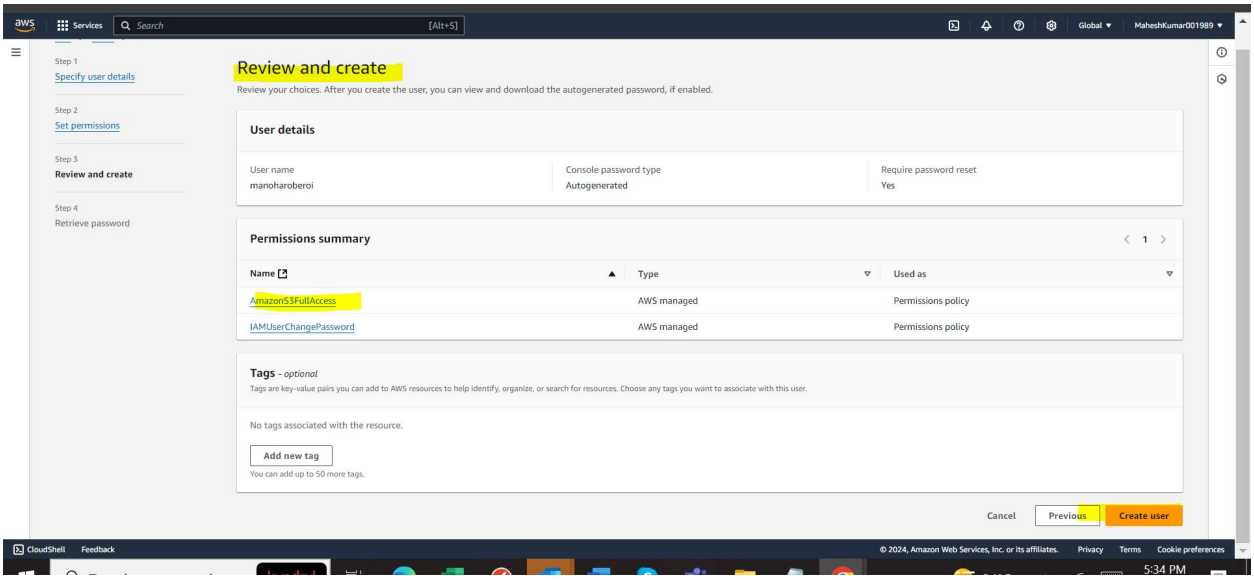
In set permission we can give IAM user full access to the S3

Note: S3 is simple storage.

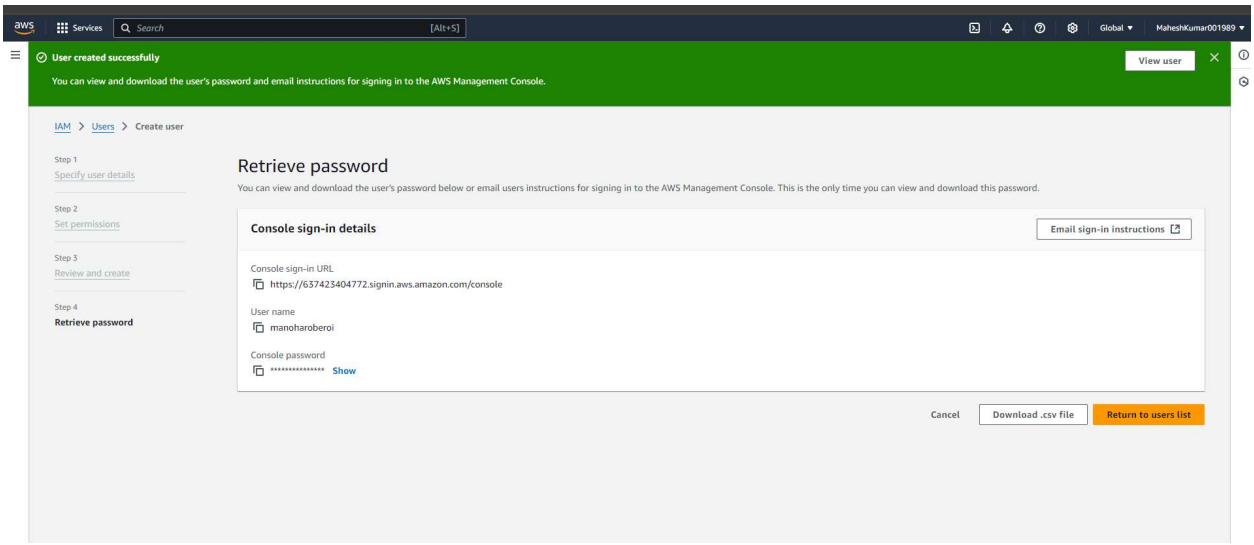




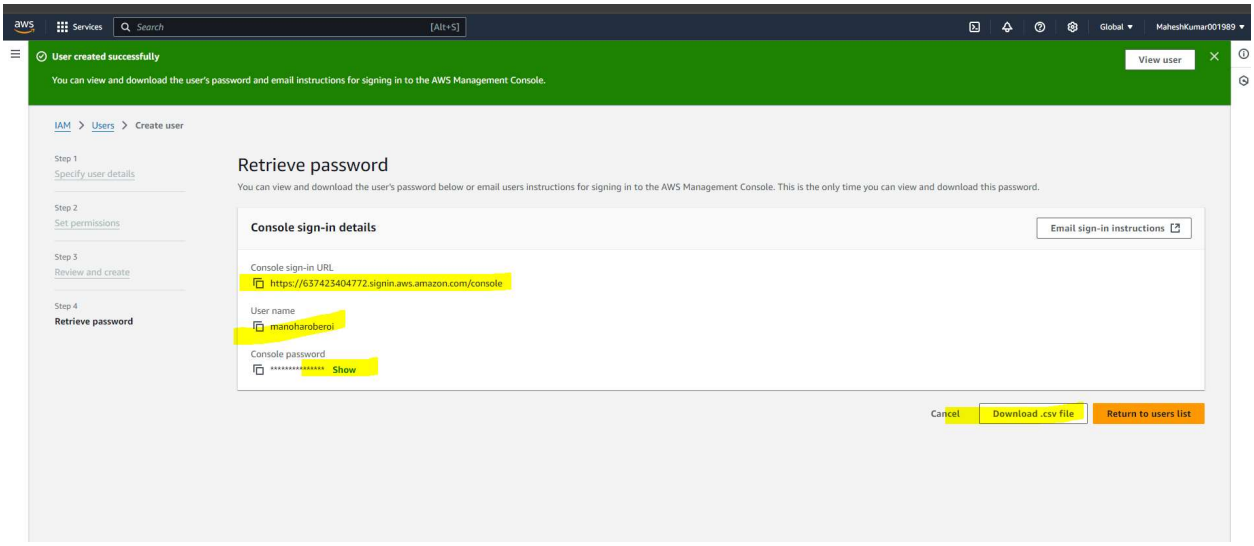
Click on create IAM user



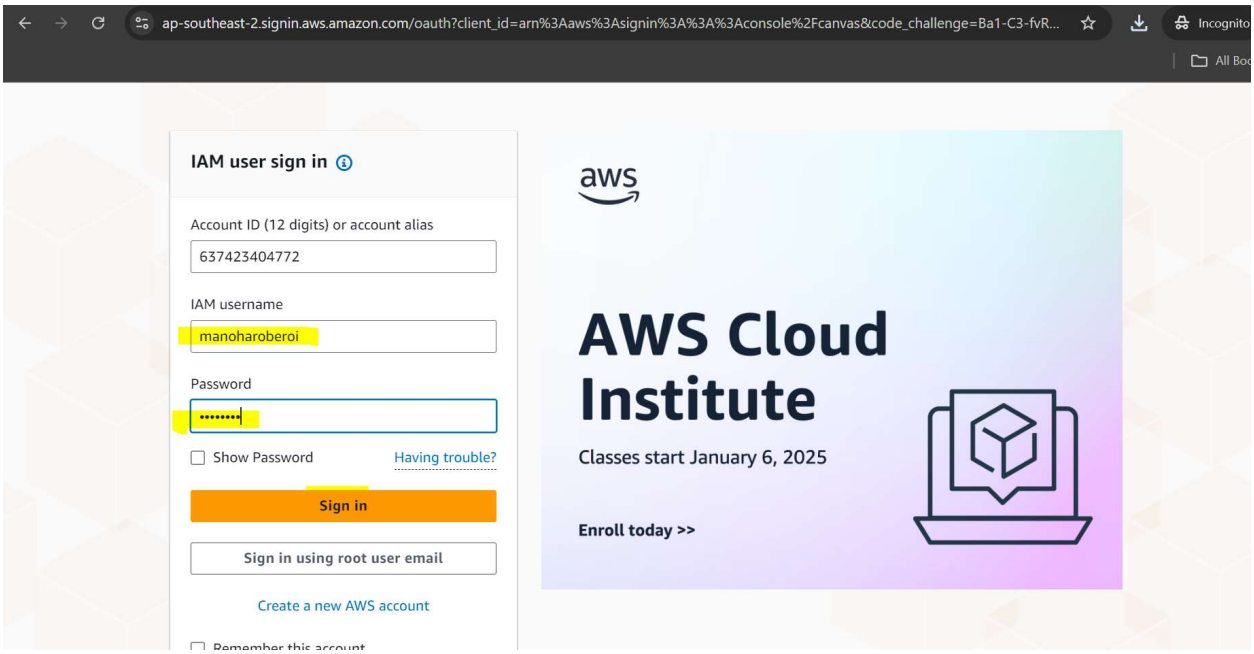
IAM user has been created successfully.



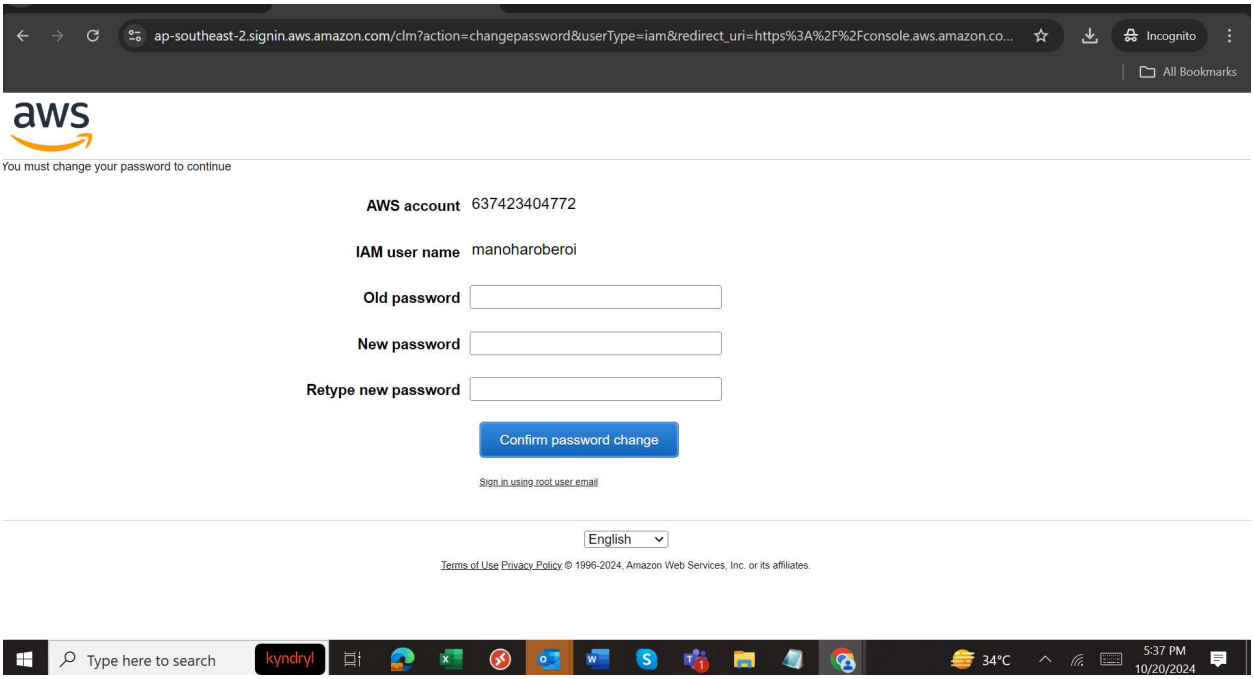
In a notepad you can copy the console URL, username and password or you can download the .csv file

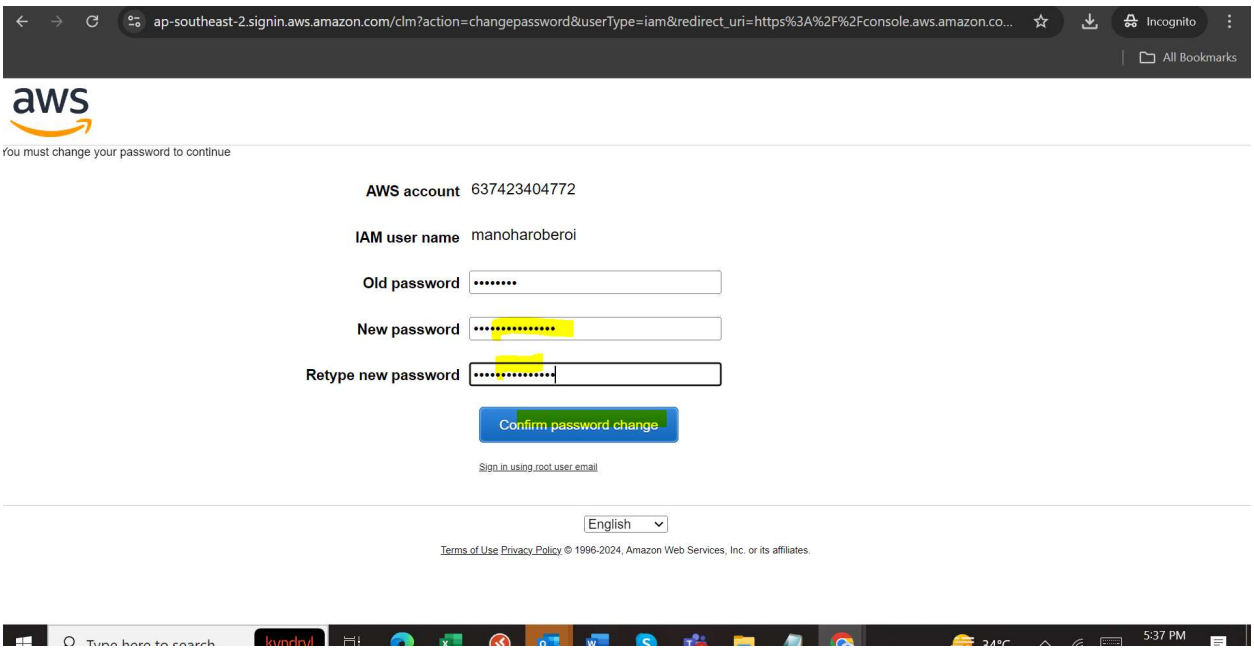


Now we can copy the console URL and paste on the browser and sign in to aws console with a IAM user

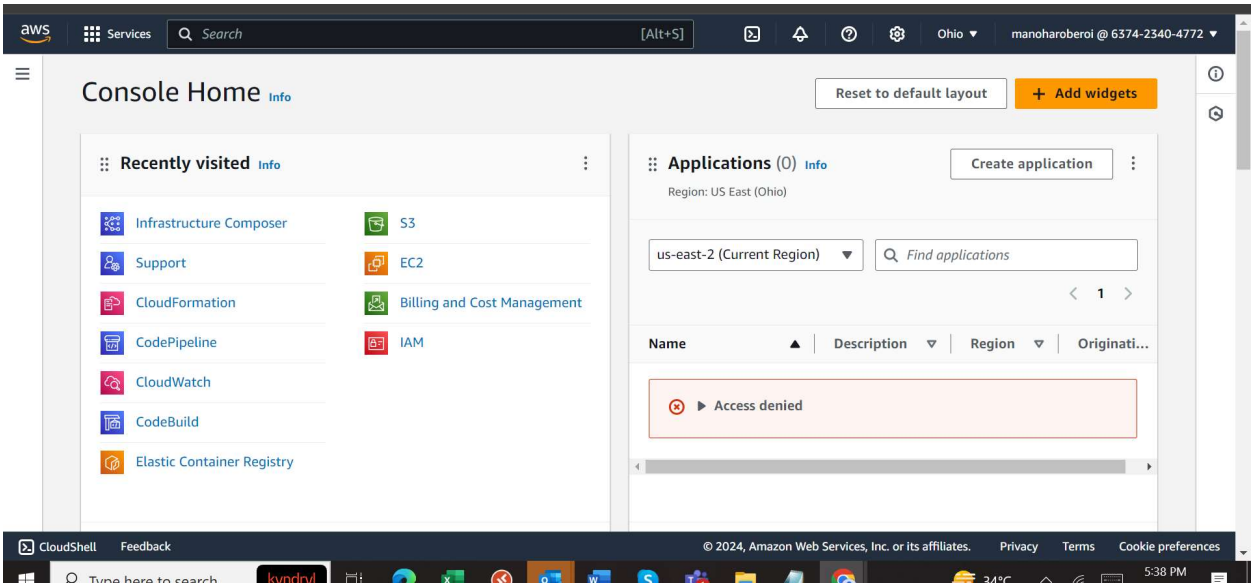


Its asking to change the password so I will create a new password for my IAM user

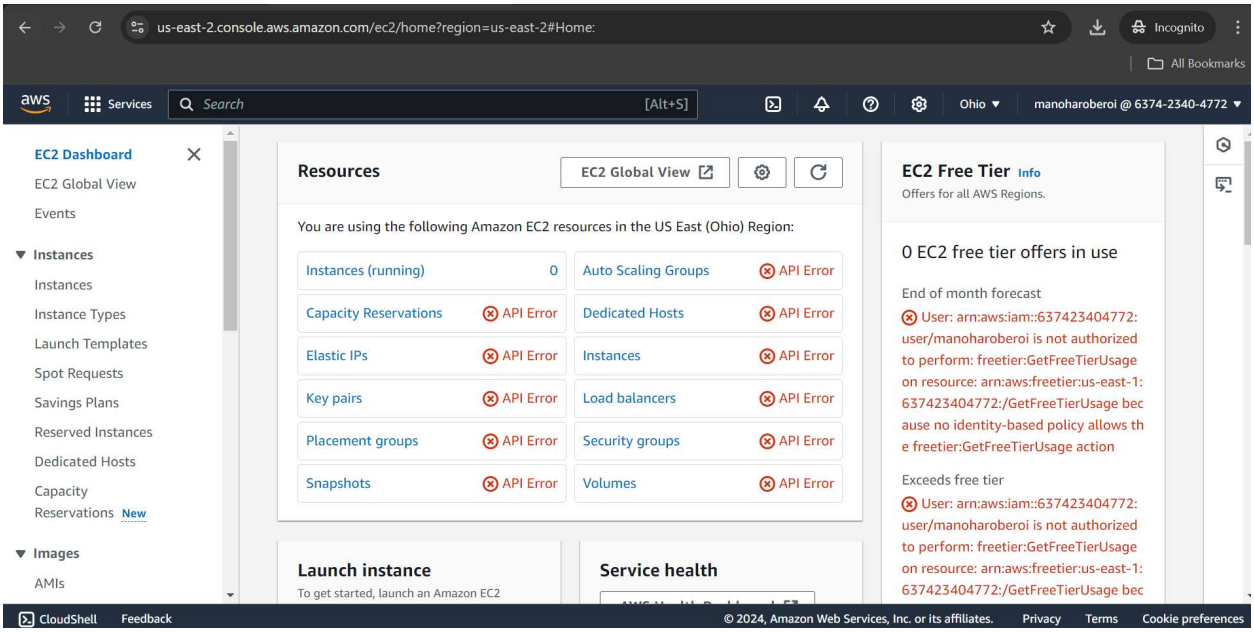




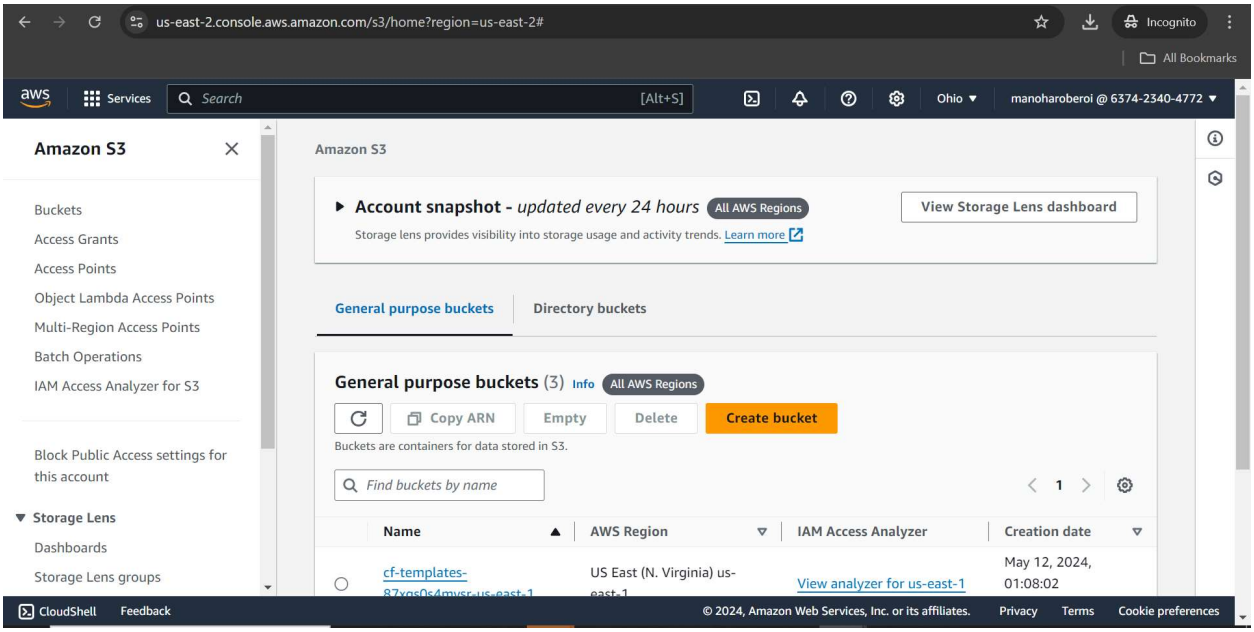
Successfully logged in to aws console with a IAM user



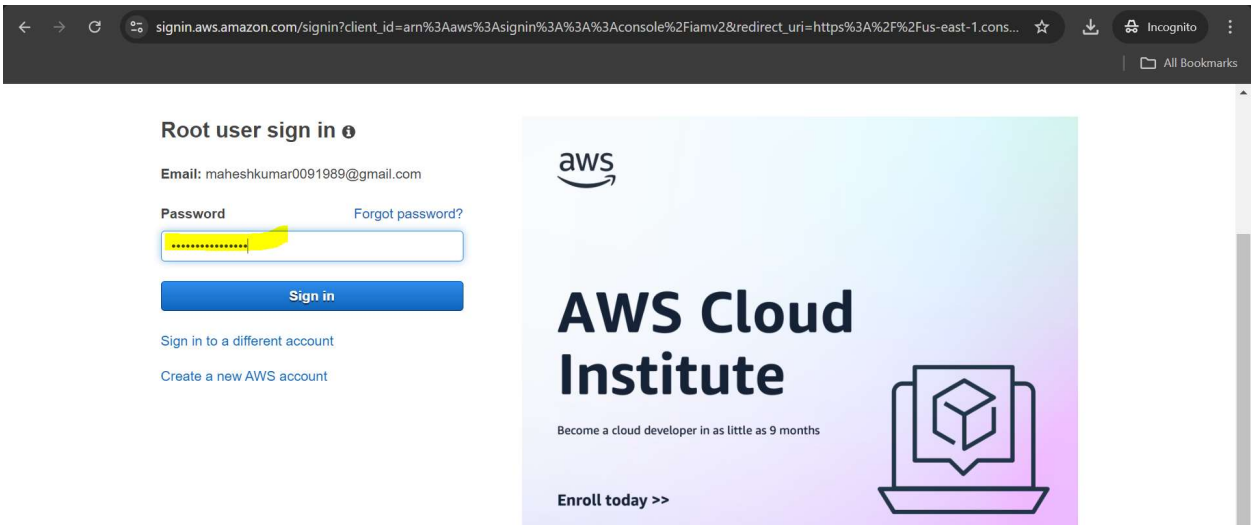
Search for EC2 but I am getting API error becoz to this IAM user I have only given the s3 full access but not the EC2 full access so I am getting this error.

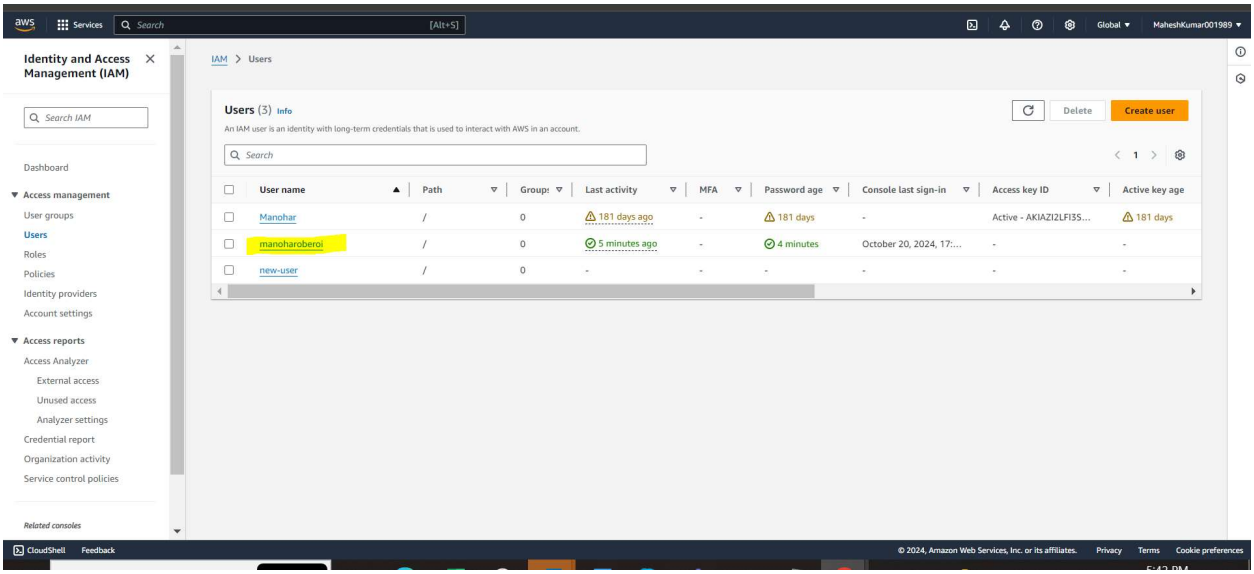


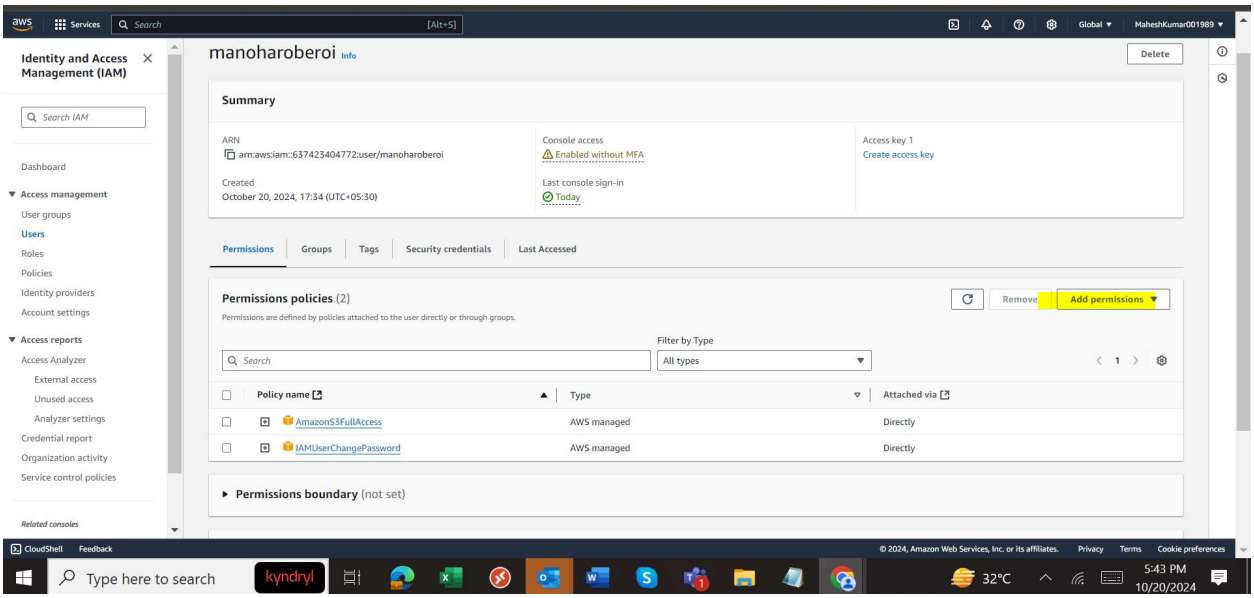
I will search for s3 and I am not getting any api error.

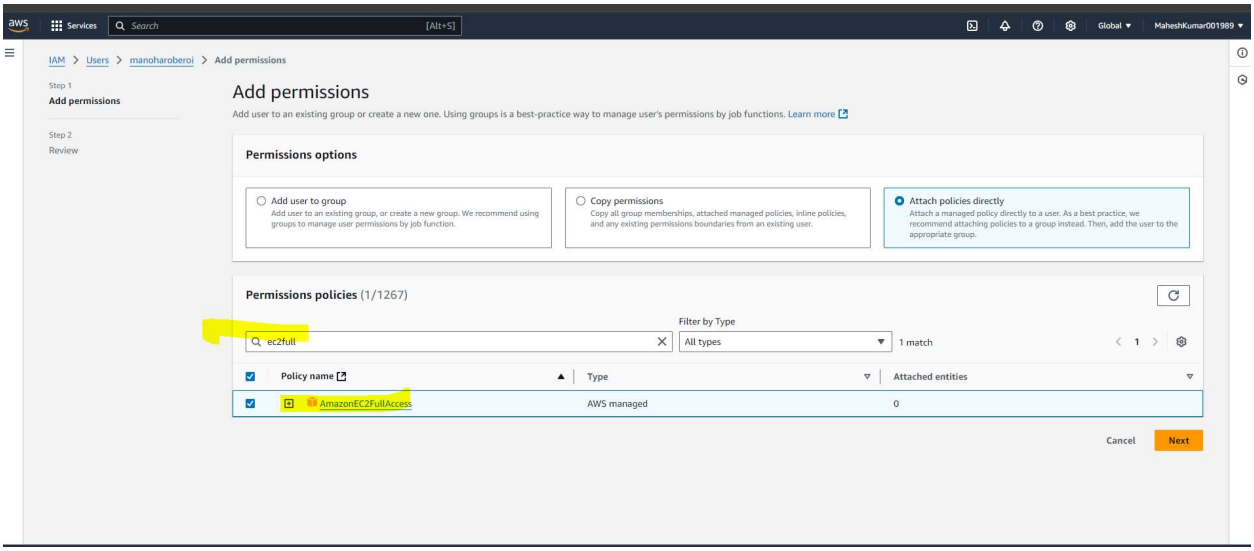


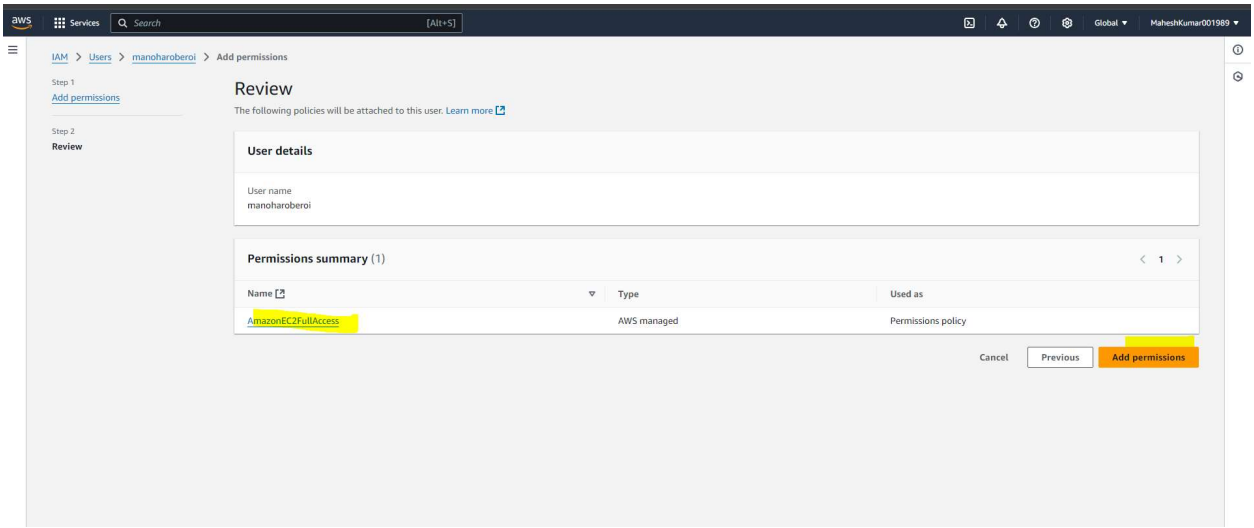
If I want I can give my IAM user ec2 full access. So I will logout from the IAM user and I will login to aws console with my root user account and add ec2 full access permissions to my IAM user



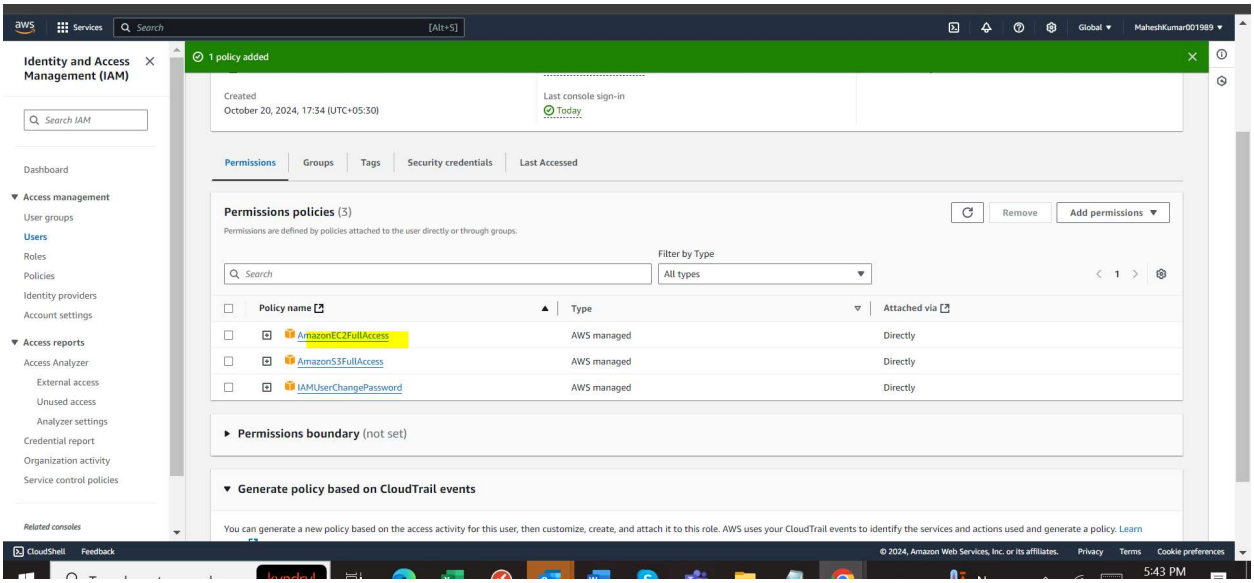

aws
Services
Search
[Alt+S]
Global
MaheshKumar001989
Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access Analyzer
External access
Unused access
Analyzer settings
Credential report
Organization activity
Service control policies
Related consoles
IAM > Users
Users (3) Info
An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.
Delete
Create user
Search
User name
Path
Groups
Last activity
MFA
Password age
Console last sign-in
Access key ID
Active key age
Manohar
/
0
181 days ago
-
181 days
-
Active - AKIAZI2LFI3S...
181 days
manoharoberoi
/
0
5 minutes ago
-
4 minutes
October 20, 2024, 17:...
-
-
new-user
/
0
-
-
-
-
-
-
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
Search
[Alt+S]
Global
MaheshKumar001989
Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access Analyzer
External access
Unused access
Analyzer settings
Credential report
Organization activity
Service control policies
Related consoles
manoharoberoi Info
Delete
Summary
ARN
arn:aws:iam::637423404772:user/manoharoberoi
Console access
Enabled without MFA
Access key 1
Create access key
Created
October 20, 2024, 17:34 (UTC+05:30)
Last console sign-in
Today
Permissions
Groups
Tags
Security credentials
Last Accessed
Permissions policies (2)
Permissions are defined by policies attached to the user directly or through groups.
Remove
Add permissions
Filter by Type
Search
All types
Policy name
Type
Attached via
AmazonS3FullAccess
AWS managed
Directly
IAMUserChangePassword
AWS managed
Directly
Permissions boundary (not set)
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
kyndryl
32°C
5:43 PM
10/20/2024

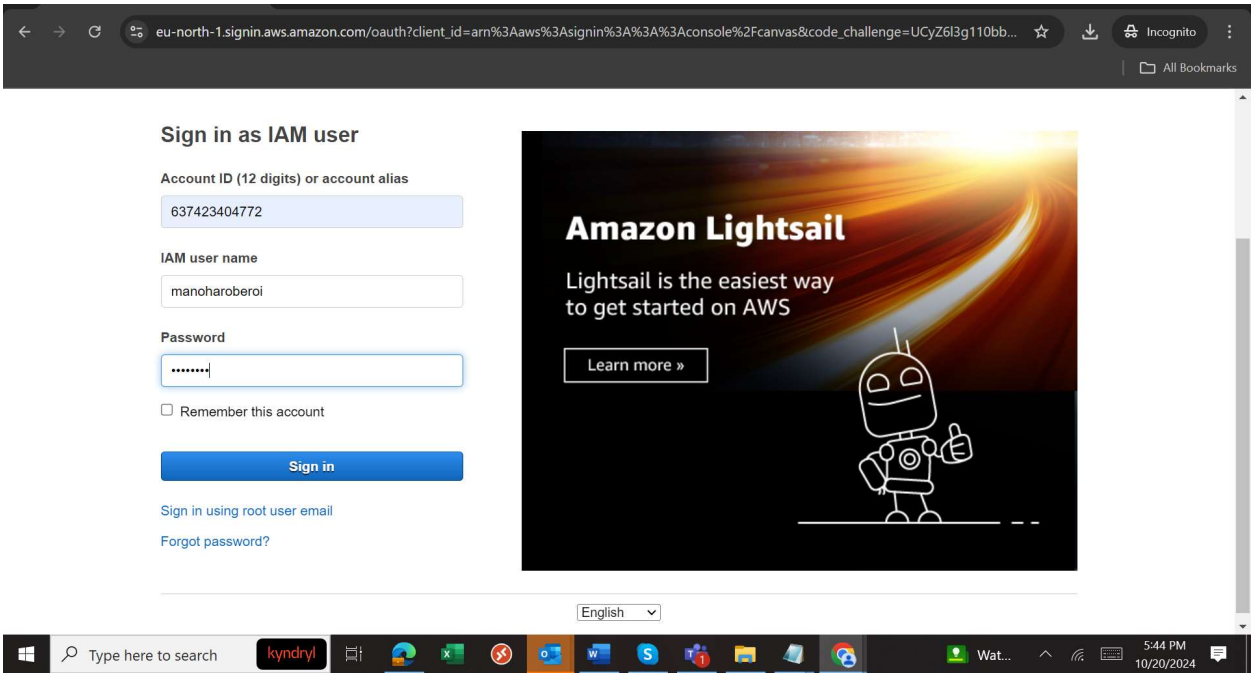




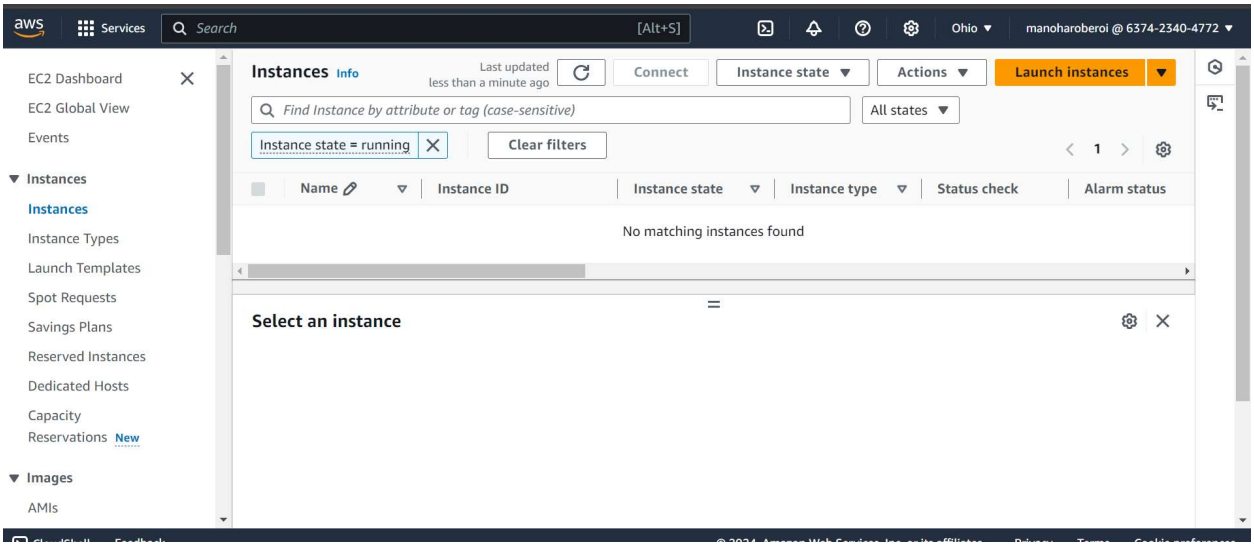
Successfully added ec2 full access permission to my IAM user



I will again login to aws console with my IAM user.



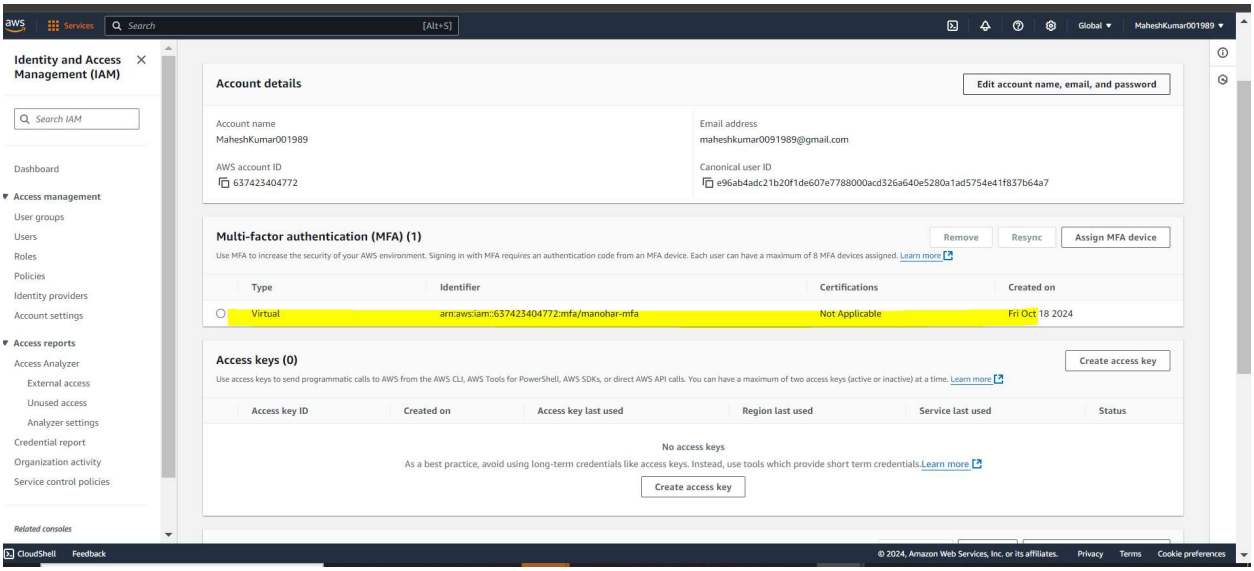
This time when I search for ec2 I am not seeing the API error.



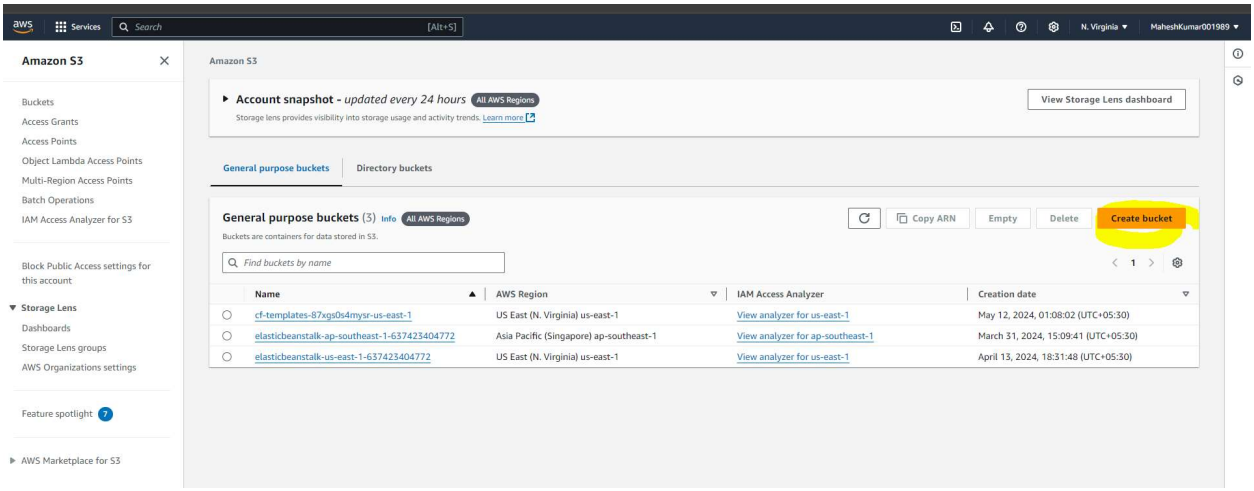
Note: In cloud trial we can check which user has done which changes.

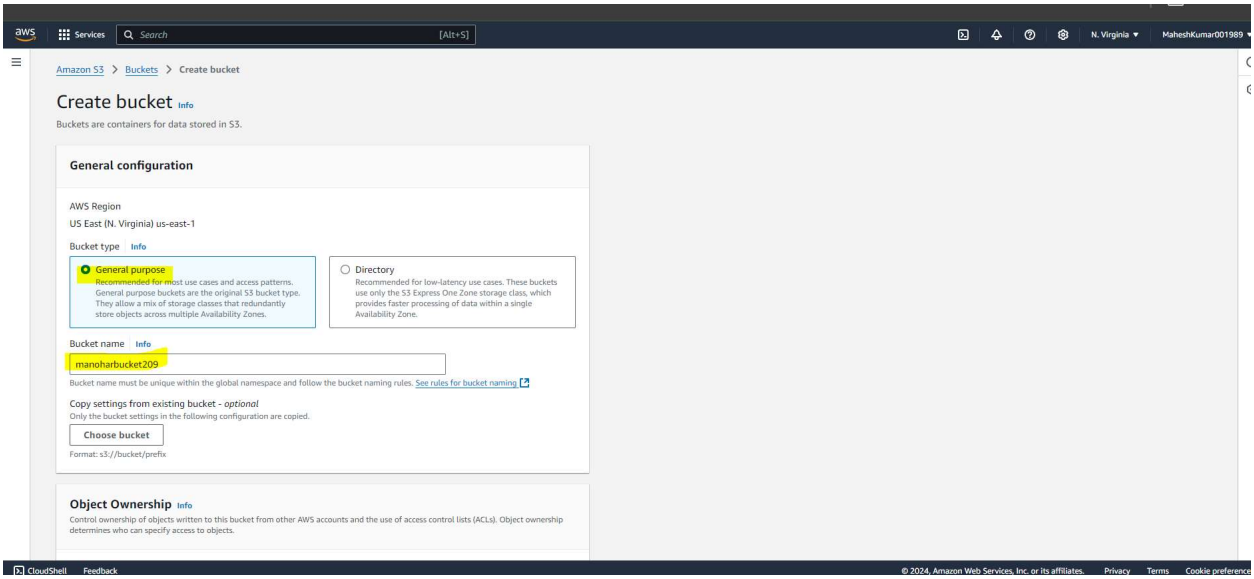
Note: IAM Role we can also create suppose if we want to integerate jenkins with s3 so for s3 we can create a role and give full ec2 permission to our s3 bucket so for that we need to create a role.

MFA already enabled for the root user account.

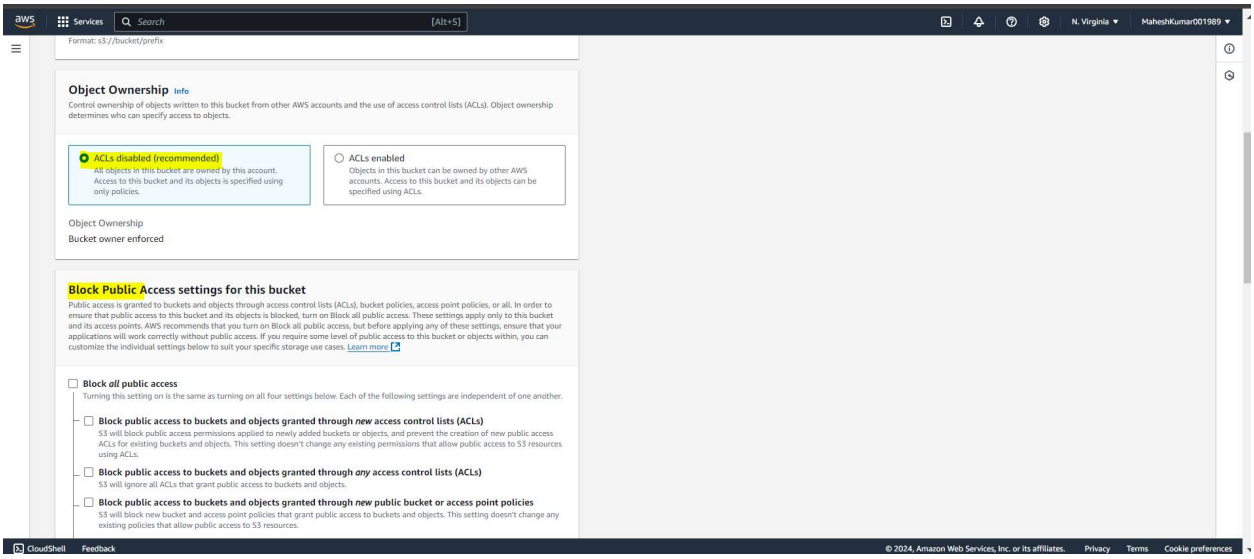


Create a S3 bucket- search for s3 and click on create a bucket

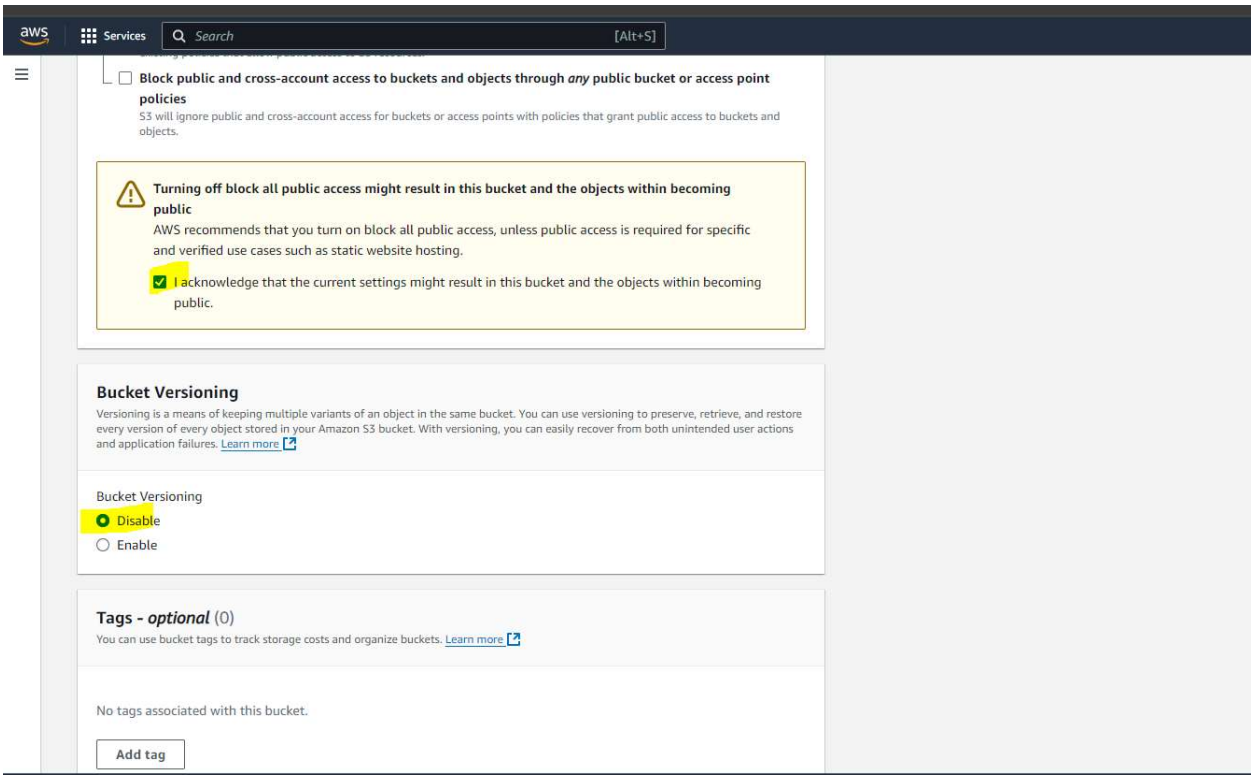




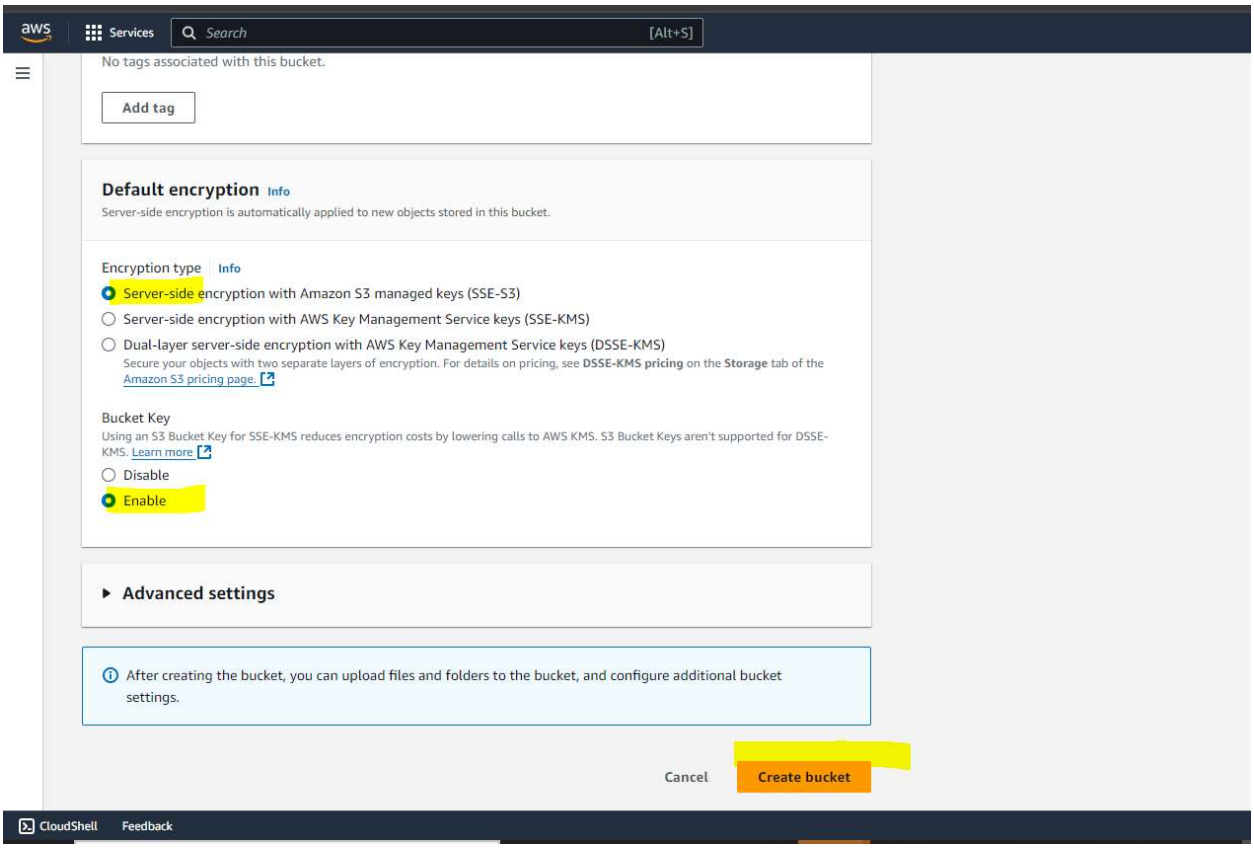
ACL disabled (recommended) an enable public access



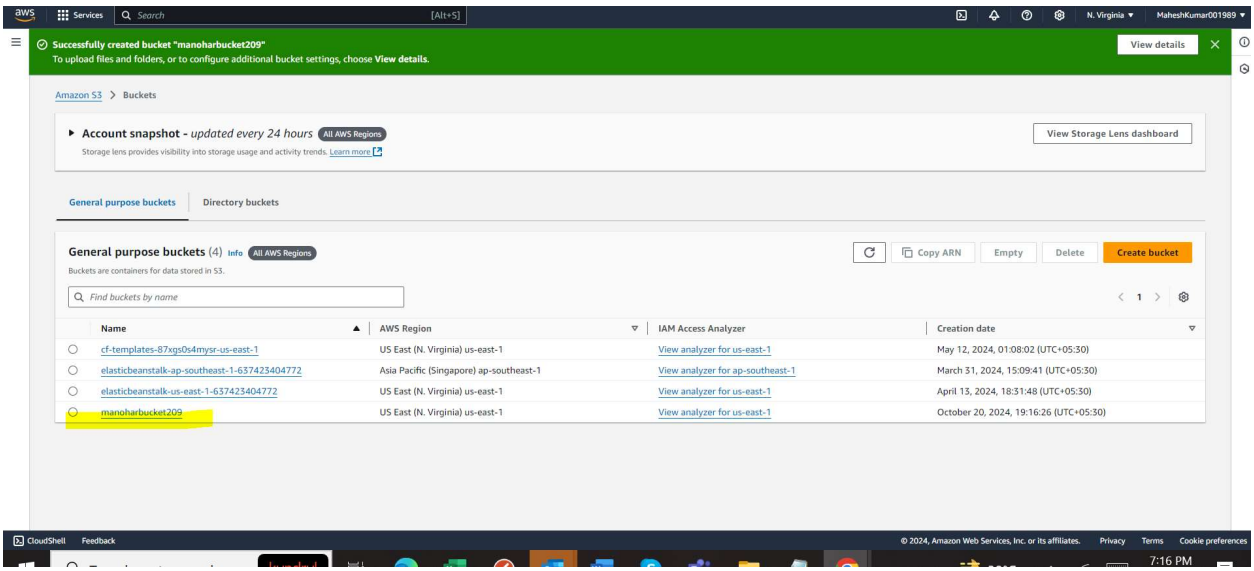
Versioning- disabled



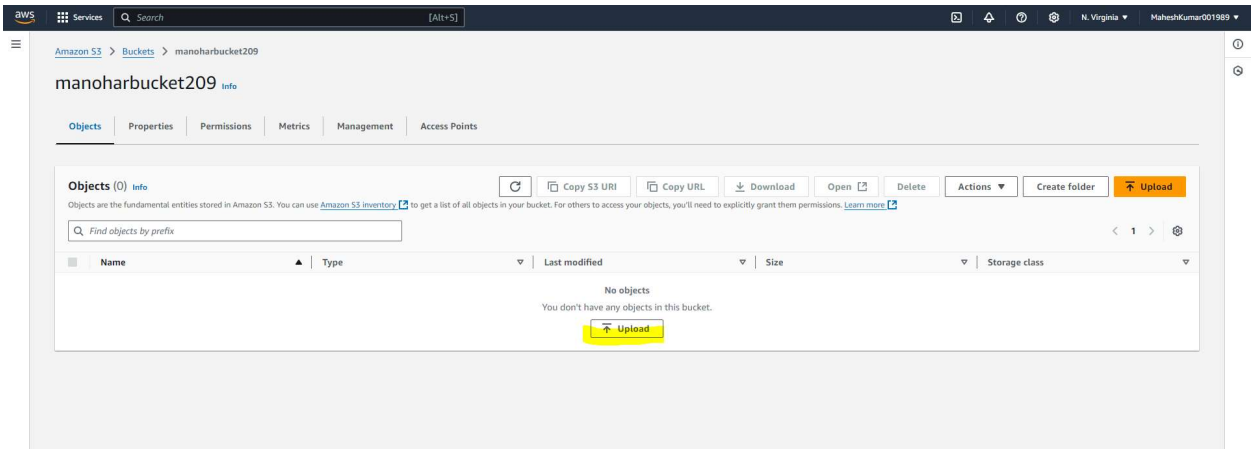
Server side encryption enabled and bucket key enabled and click on create bucket.

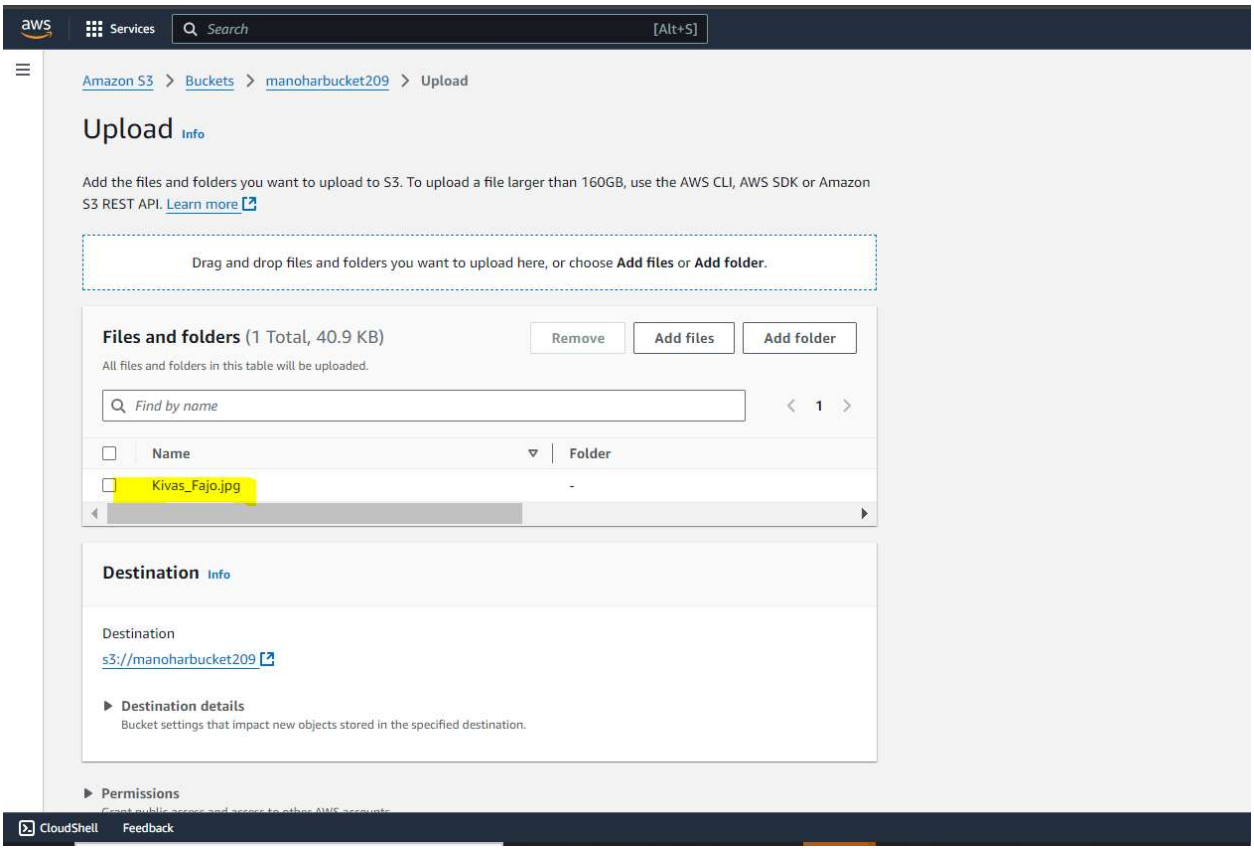


Bucket is created successfully.

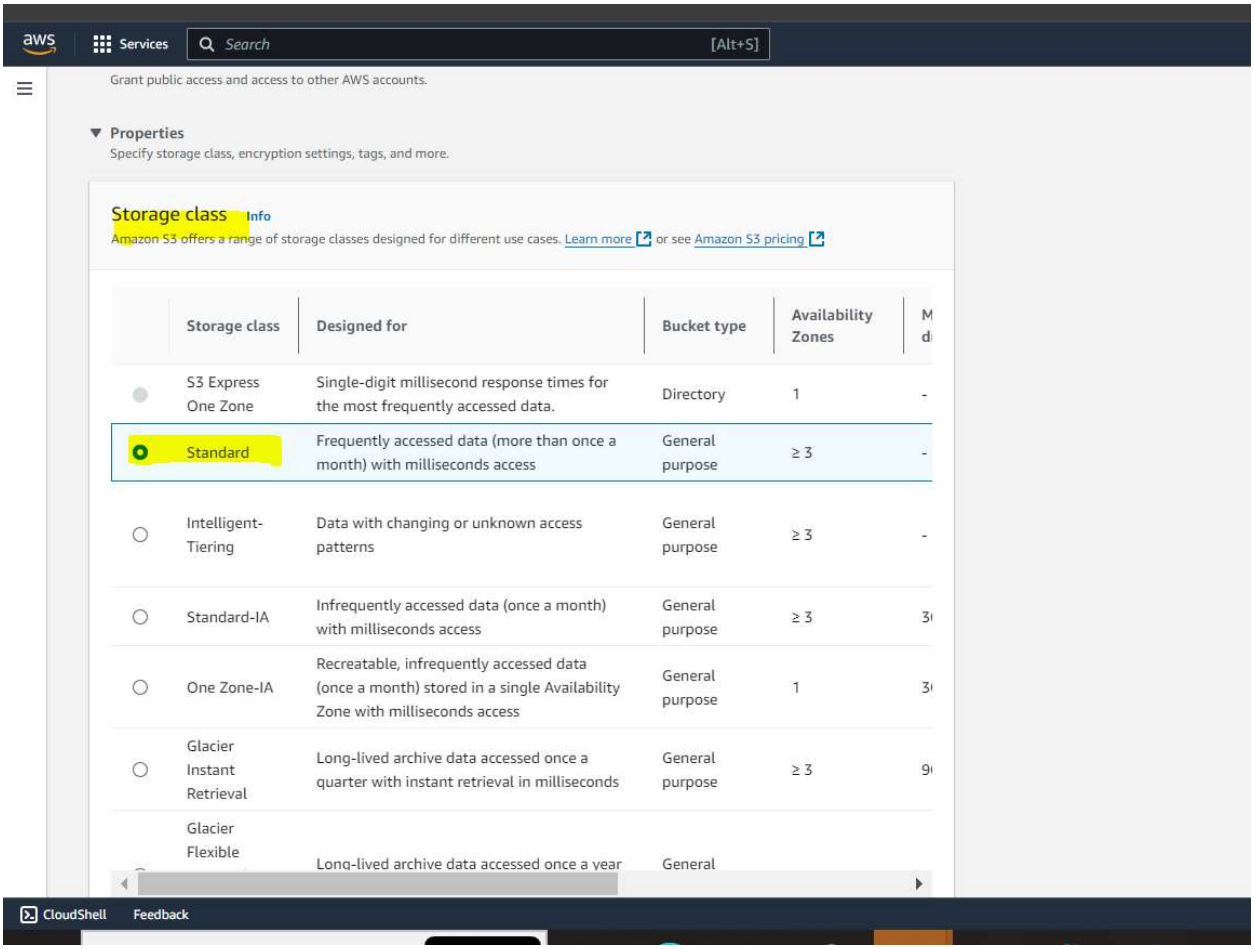


Upload a file in the bucket

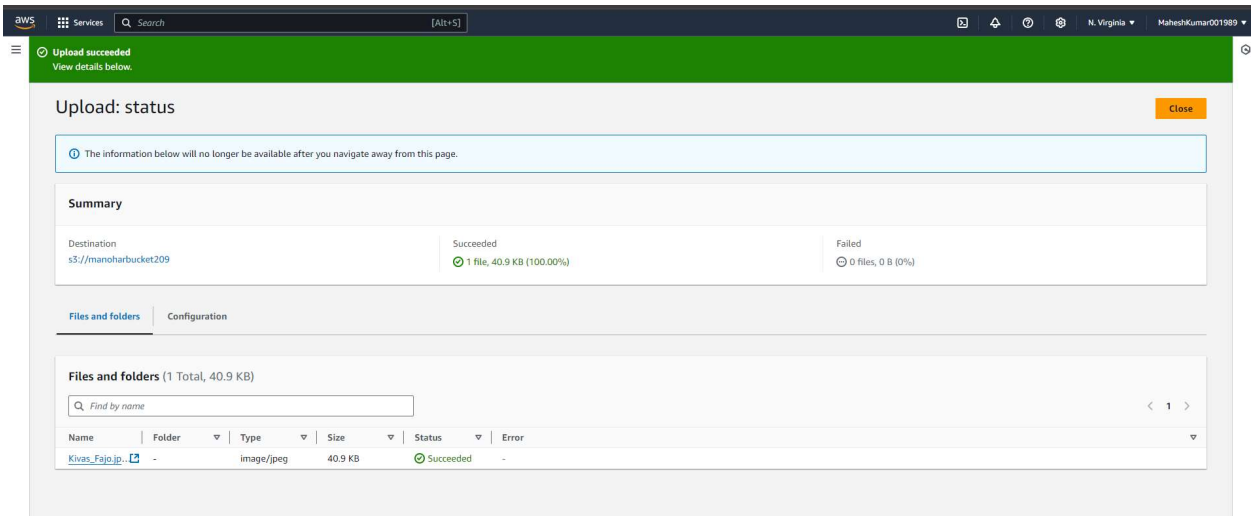




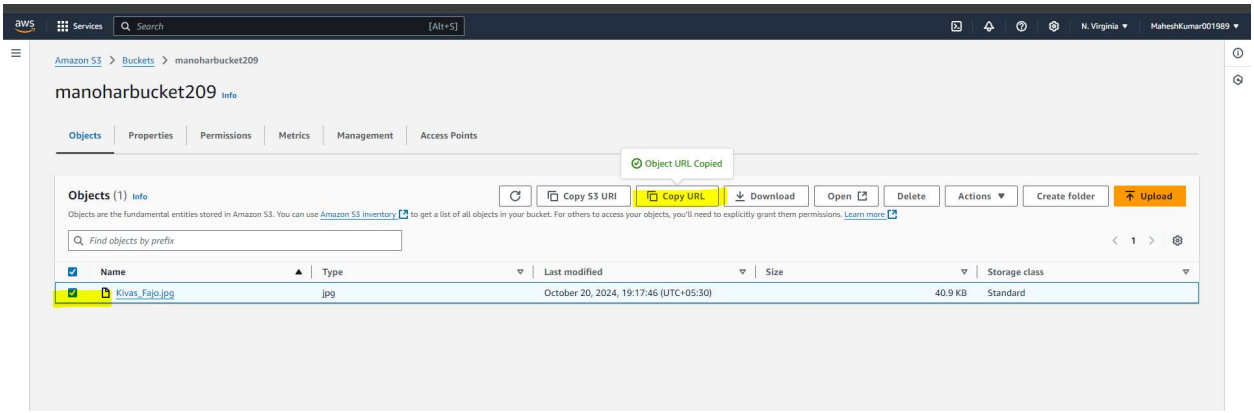
Storage class- standard



File uploaded successfully

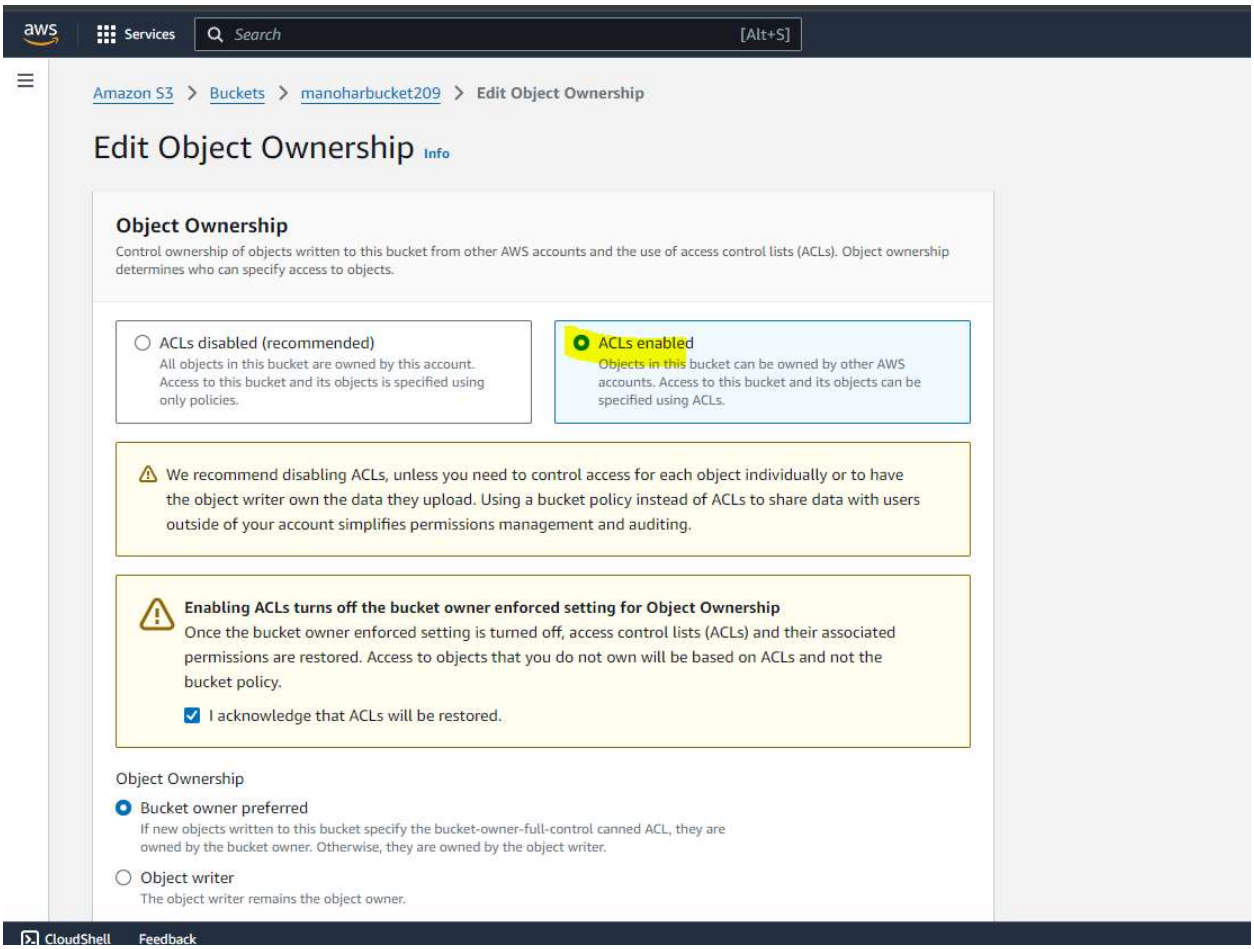


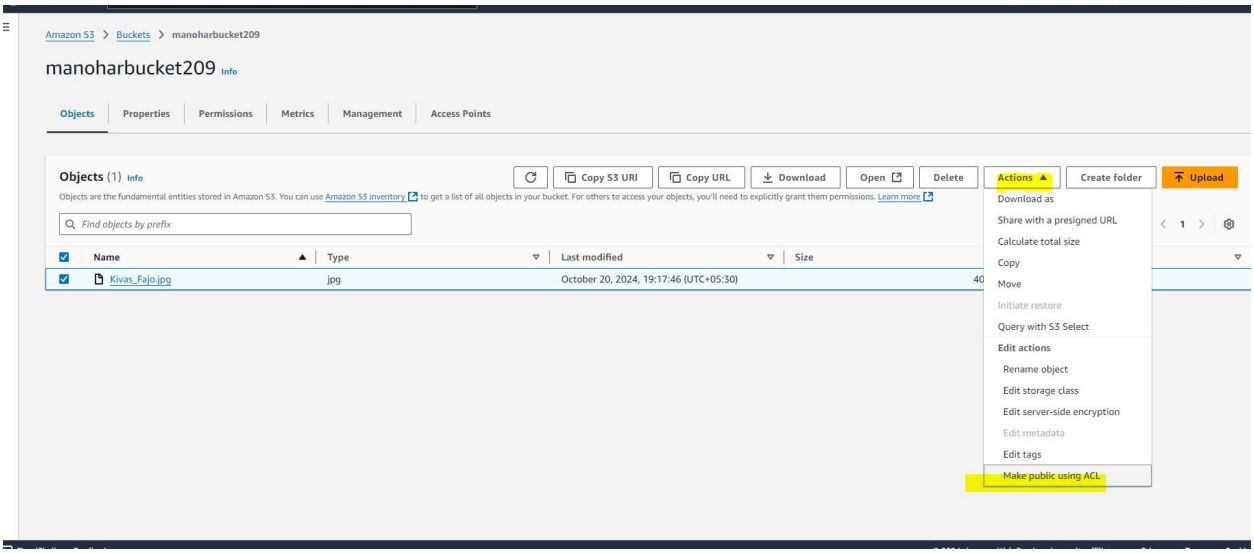
Copy the URL and paste on the browser

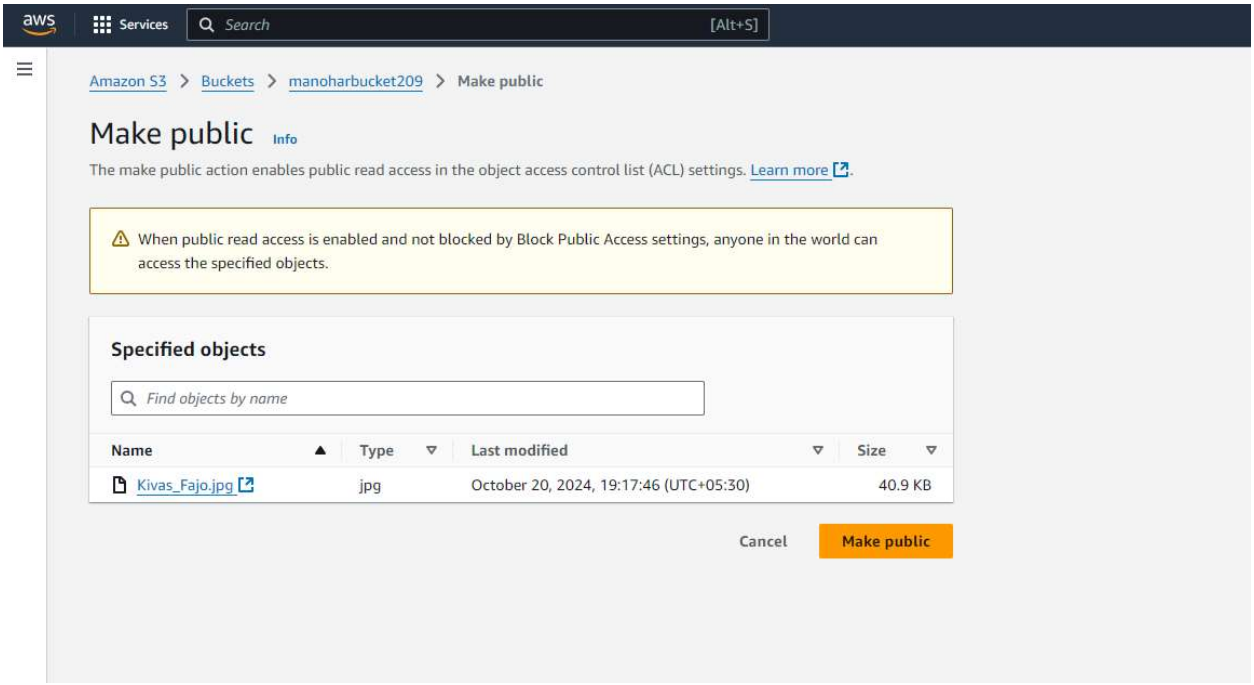


This XML file does not appear to have any style information associated with it. The document tree is shown below.


```
▼<Error>
  <Code>AccessDenied</Code>
  <Message>Access Denied</Message>
  <RequestId>SV4AYX9NSSDHN9X0</RequestId>
  <HostId>kV5VuybDey6ZrTdnWapsMN+O36uamrM4YO5GxvbxU1lvRL/S1OsgK6+XOB9lCJnw66KKVdGVNtM=</HostId>
</Error>
```


aws
Services
Search
[Alt+S]
Amazon S3 > Buckets > manoharbucket209 > Edit Object Ownership
Edit Object Ownership Info
Object Ownership
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.
We recommend disabling ACLs, unless you need to control access for each object individually or to have the object writer own the data they upload. Using a bucket policy instead of ACLs to share data with users outside of your account simplifies permissions management and auditing.
Enabling ACLs turns off the bucket owner enforced setting for Object Ownership
Once the bucket owner enforced setting is turned off, access control lists (ACLs) and their associated permissions are restored. Access to objects that you do not own will be based on ACLs and not the bucket policy.
I acknowledge that ACLs will be restored.
Object Ownership
Bucket owner preferred
If new objects written to this bucket specify the bucket-owner-full-control canned ACL, they are owned by the bucket owner. Otherwise, they are owned by the object writer.
Object writer
The object writer remains the object owner.
CloudShell
Feedback


Amazon S3 > Buckets > manoharbucket209
manoharbucket209 Info
Objects
Properties
Permissions
Metrics
Management
Access Points
Objects (1) Info
Copy S3 URI
Copy URL
Download
Open
Delete
Actions
Create folder
Upload
Find objects by prefix
Name
Type
Last modified
Size
Kivas_Fajo.jpg
jpg
October 20, 2024, 19:17:46 (UTC+05:30)
Download as
Share with a presigned URL
Calculate total size
Copy
Move
Initiate restore
Query with S3 Select
Edit actions
Rename object
Edit storage class
Edit server-side encryption
Edit metadata
Edit tags
Make public using ACL



After enable the ACL and make it public I can access the image which I uploaded in S3 bucket using the S3 url