AWS TASK LIST

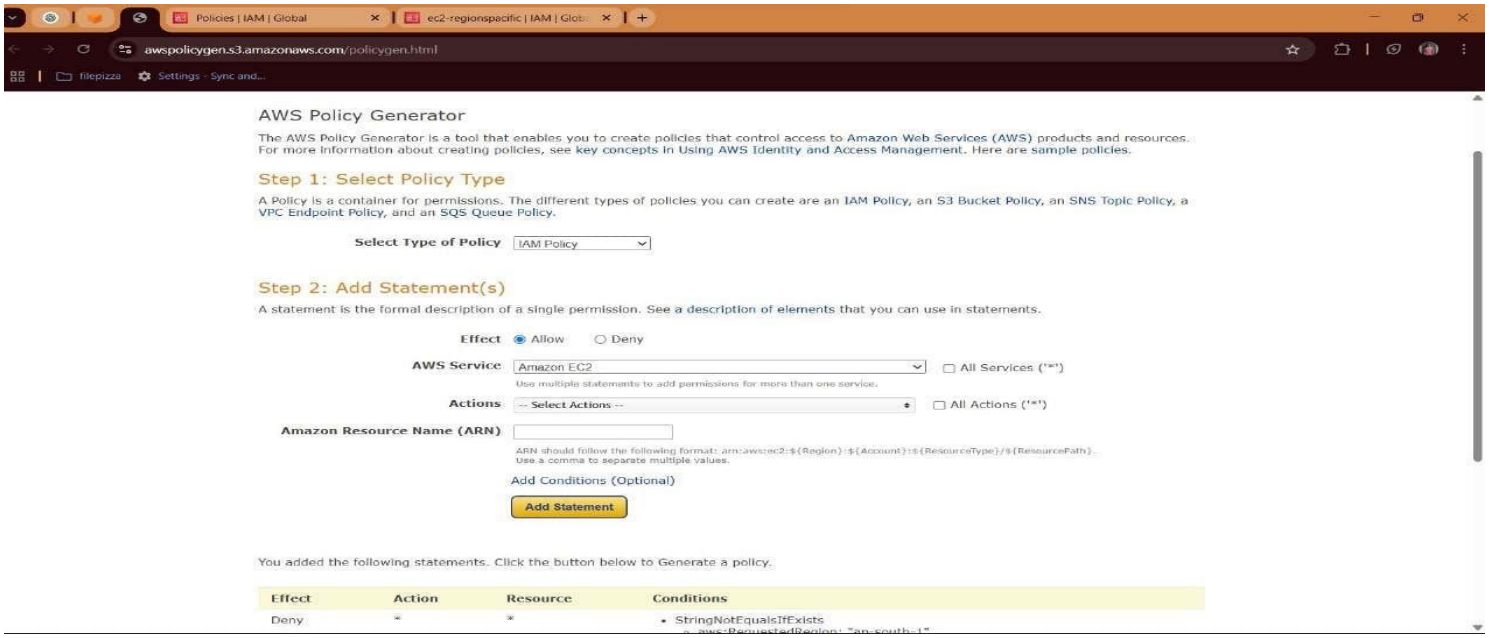
SR/NO	TASK
1	create user assign one policy (ec2 full permission) but any specific region(mumbai)
2	create bucket and apply mfa to the bucket
3	create policy that given root user access to another user for specific time after specific time access get denied if he try to access get access denied error
4	upload object in bucket this object is private but other user can access it only for specific time
5	create policy and give user to access or use only one specific bucket read and write list permission
6	ec2 instance name:- scripted-web-hosting
7	Ec2 Unblock visibilty Public Access for AMIs
8	Share an EC2 AMI with a Friend's AWS Account.
9	ec2 :- EC2 recover loss key-pair .

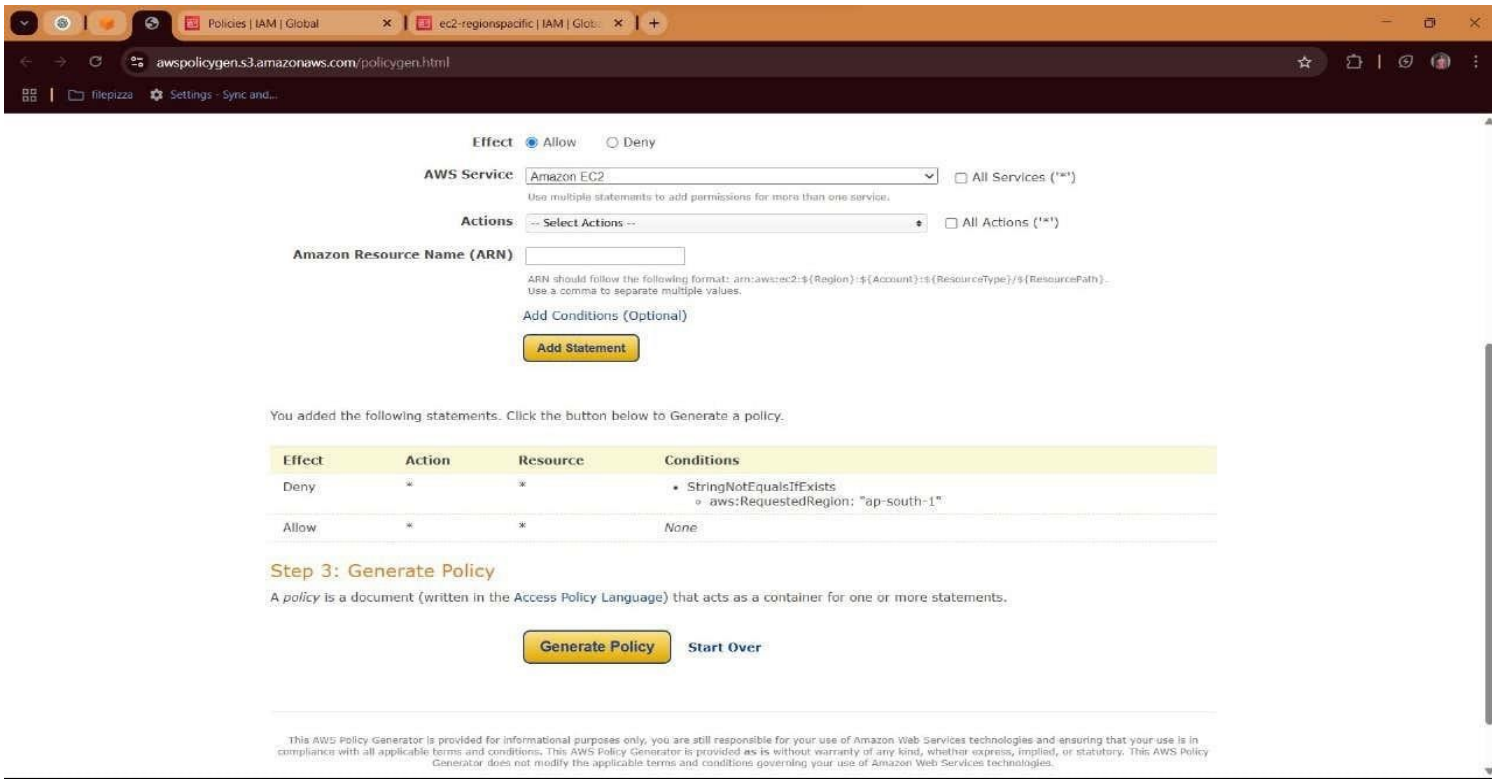
Task No:1

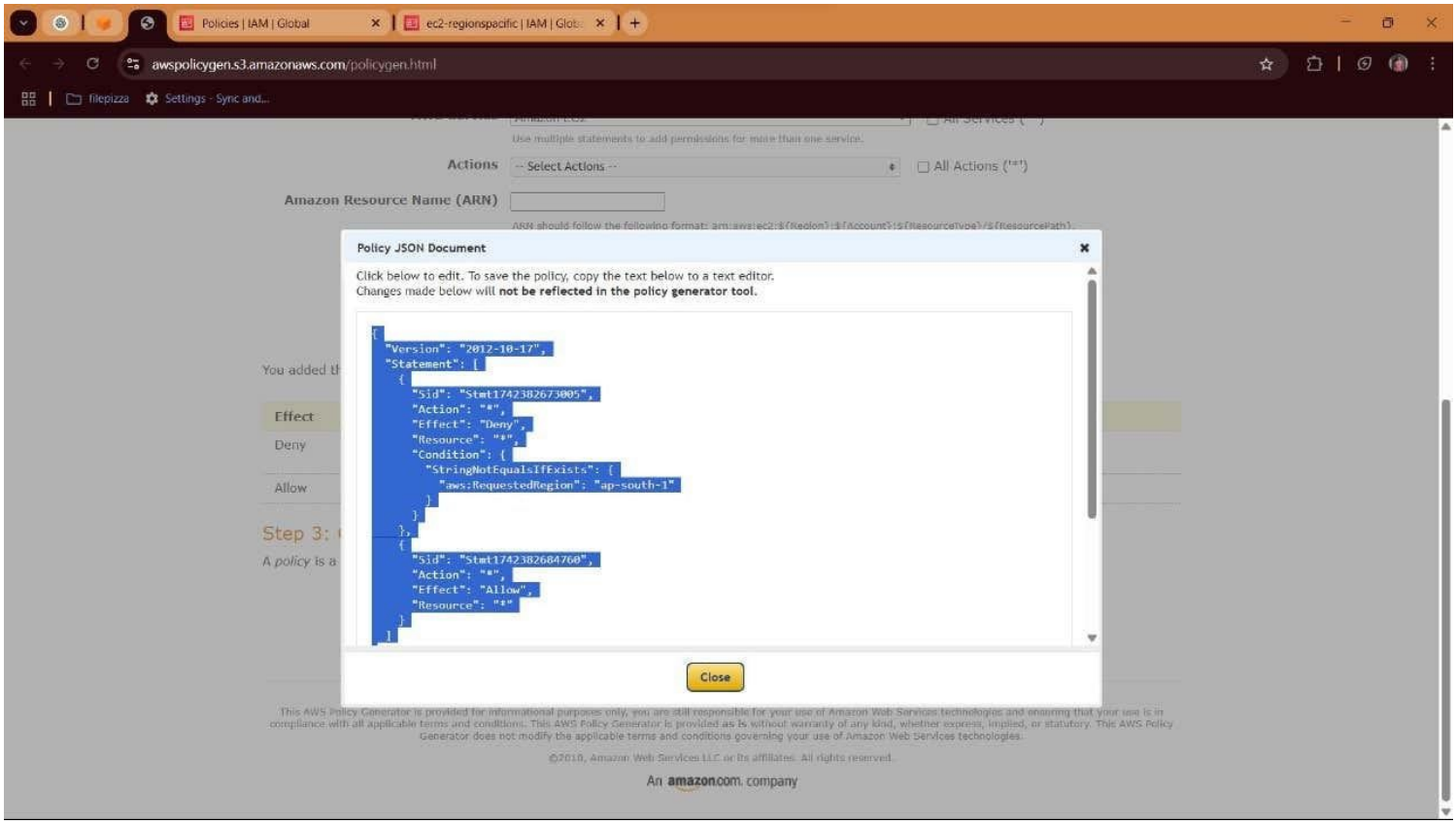
Name: sagar

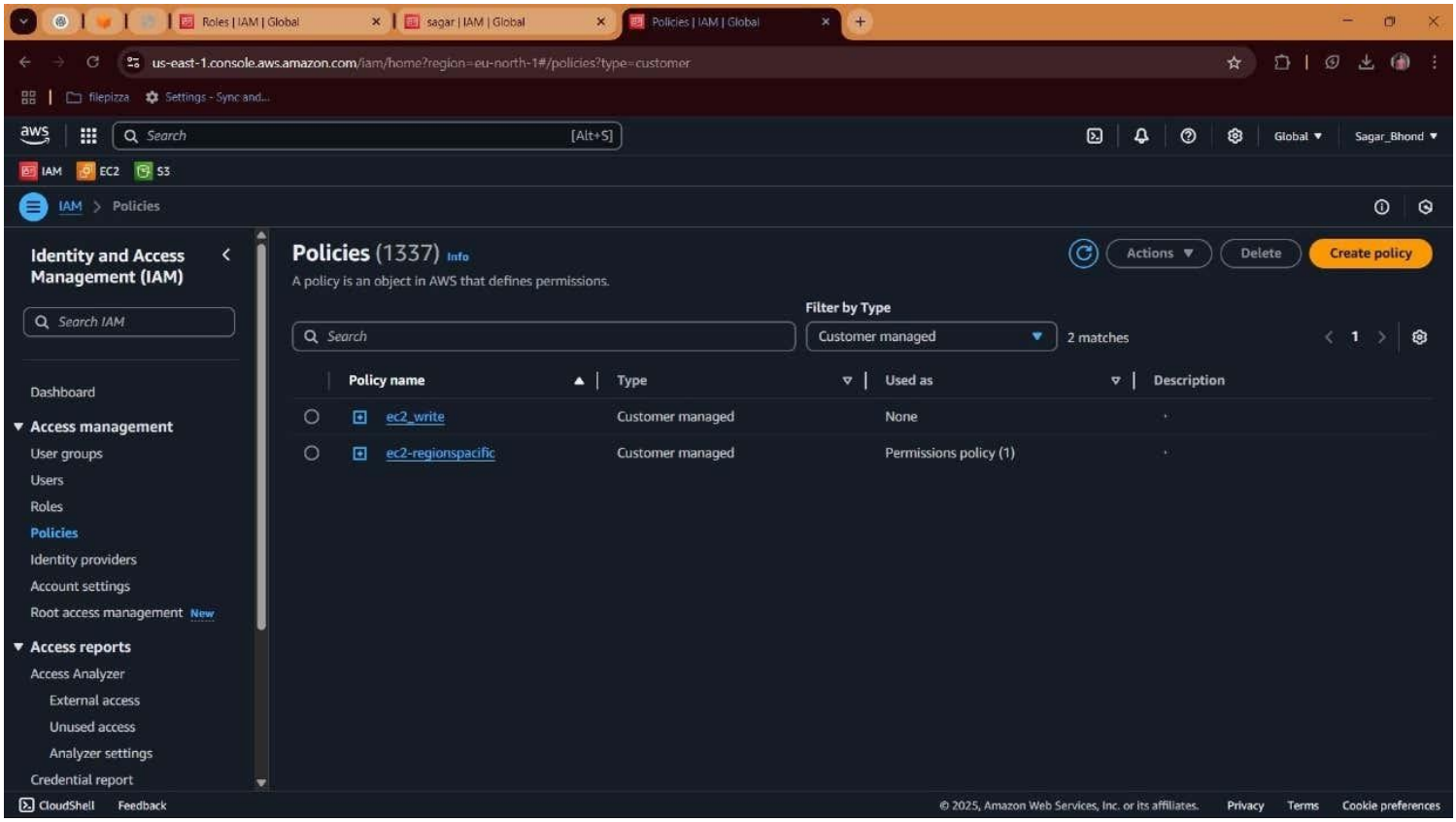
* Ec2-to grant region specific permation allow only ap-south-1a

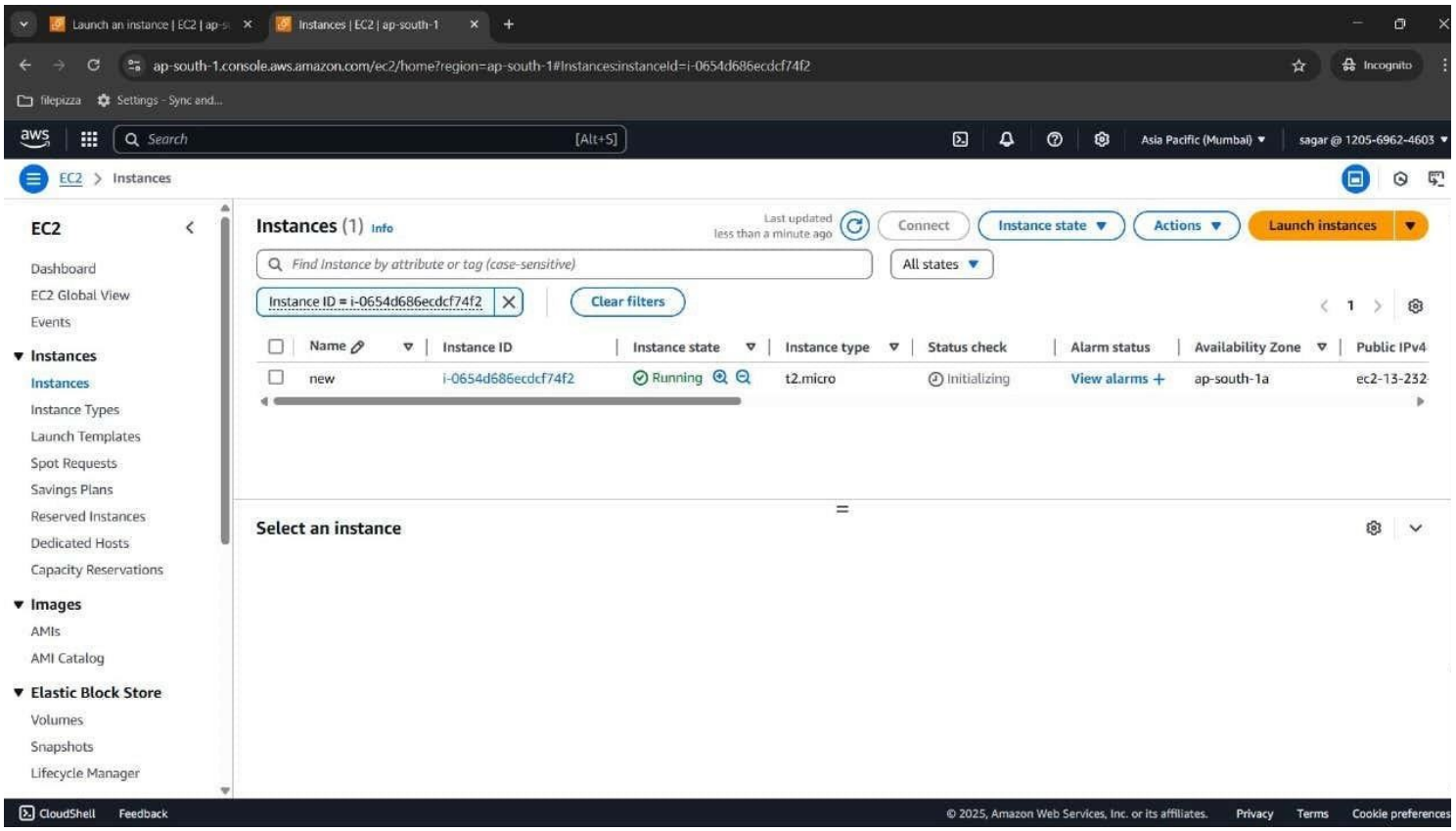
//create policy

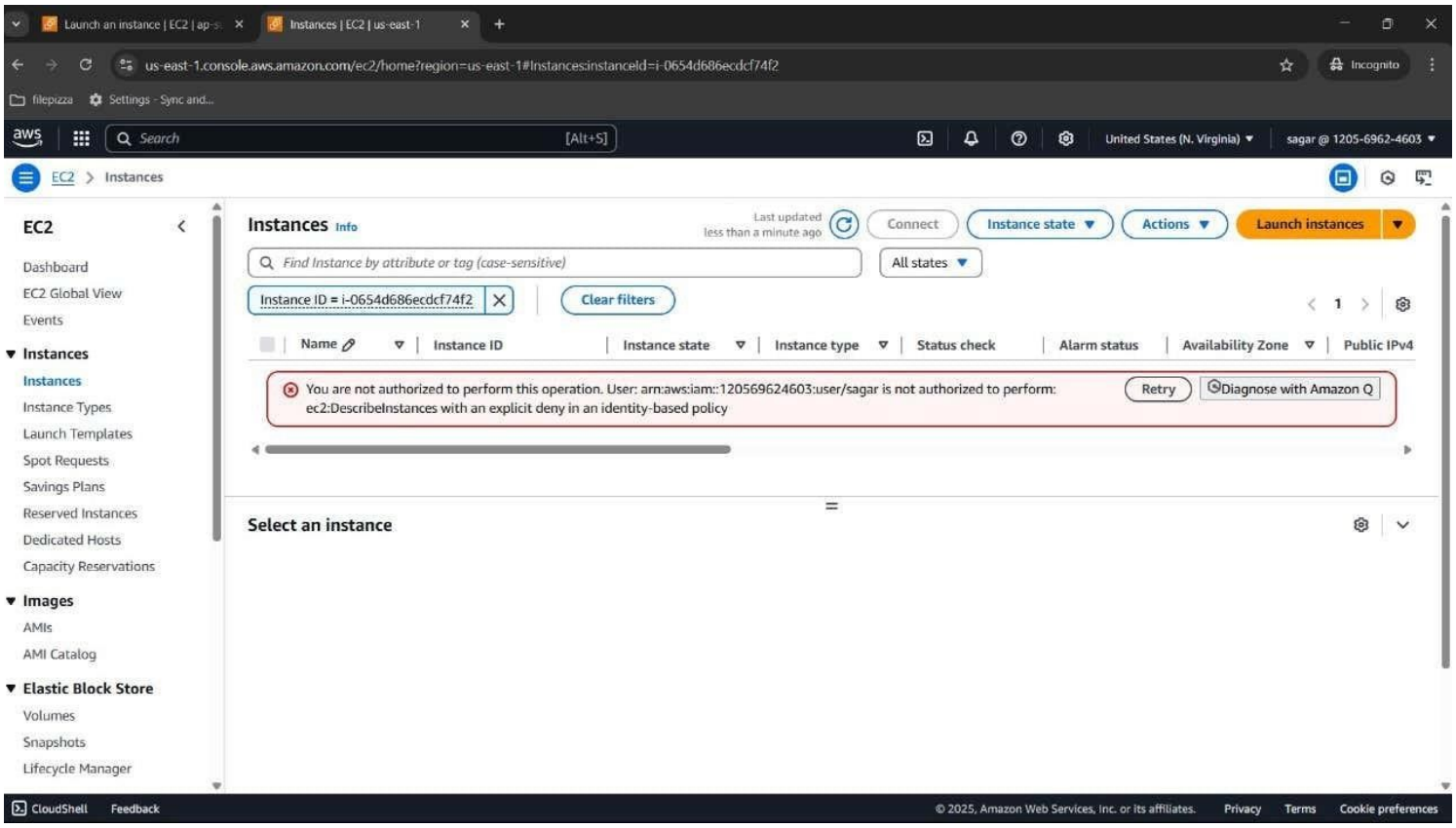





Policies | IAM | Global
ec2-regionspecific | IAM | Glob
awspolicygen.s3.amazonaws.com/policygen.html
filepizza
Settings - Sync and...
Actions
-- Select Actions --
All Actions ('*')
Amazon Resource Name (ARN)
Policy JSON Document
Click below to edit. To save the policy, copy the text below to a text editor.
Changes made below will not be reflected in the policy generator tool.
{
"Version": "2012-10-17",
"Statement": [
{
"Sid": "Stmt1742382673005",
"Action": "*",
"Effect": "Deny",
"Resource": "*",
"Condition": {
"StringNotEqualsIfExists": {
"aws:RequestedRegion": "ap-south-1"
}
}
},
{
"Sid": "Stmt1742382684760",
"Action": "*",
"Effect": "Allow",
"Resource": "*"
}
]
Close
Effect
Deny
Allow
Step 3:
A policy is a
You added th
This AWS Policy Generator is provided for informational purposes only, you are still responsible for your use of Amazon Web Services technologies and ensuring that your use is in compliance with all applicable terms and conditions. This AWS Policy Generator is provided AS IS without warranty of any kind, whether express, implied, or statutory. This AWS Policy Generator does not modify the applicable terms and conditions governing your use of Amazon Web Services technologies.
©2010, Amazon Web Services LLC or its affiliates. All rights reserved.
An amazon.com company


Roles | IAM | Global
sagar | IAM | Global
Policies | IAM | Global
us-east-1.console.aws.amazon.com/iam/home?region=eu-north-1#/policies?type=customer
filepizza
Settings - Sync and...
aws
Search
[Alt+S]
Global
Sagar_Bhond
IAM
EC2
S3
IAM > Policies
Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Root access management New
Access reports
Access Analyzer
External access
Unused access
Analyzer settings
Credential report
Policies (1337) Info
A policy is an object in AWS that defines permissions.
Actions
Delete
Create policy
Filter by Type
Search
Customer managed
2 matches
1
Policy name
Type
Used as
Description
ec2_write
Customer managed
None
ec2-regionspacific
Customer managed
Permissions policy (1)
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Launch an instance | EC2 | ap-s
Instances | EC2 | ap-south-1
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#Instances:instanceId=i-0654d686ecdcf74f2
Incognito
filepizza
Settings - Sync and...
Search
[Alt+S]
Asia Pacific (Mumbai)
sagar @ 1205-6962-4603
EC2 > Instances
EC2
Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Instances (1) Info
Last updated less than a minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Instance ID = i-0654d686ecdcf74f2
Clear filters
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public IPv4
new
i-0654d686ecdcf74f2
Running
t2.micro
Initializing
View alarms +
ap-south-1a
ec2-13-232
Select an instance
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Launch an instance | EC2 | ap-s
Instances | EC2 | us-east-1
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Instances:instanceId=i-0654d686ecdcf74f2
Incognito
filepizza
Settings - Sync and...
Search
[Alt+S]
United States (N. Virginia)
sagar @ 1205-6962-4603
EC2 > Instances
EC2
Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Instances Info
Last updated less than a minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Instance ID = i-0654d686ecdcf74f2
Clear filters
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public IPv4
You are not authorized to perform this operation. User: arn:aws:iam::120569624603:user/sagar is not authorized to perform: ec2:DescribeInstances with an explicit deny in an identity-based policy
Retry
Diagnose with Amazon Q
Select an instance
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Task No:2

Name: sagar

*AWS-MFA on S3 - Notepad

Step:1 $ aws configure

Step:2 $ aws s3api list-buckets

//check bucket list

```
sagar_c7otrfh@SagarBhond MINGW64 ~
$ aws s3api list-buckets
{
    "Buckets": [
        {
            "Name": "sagar-007",
            "CreationDate": "2025-03-23T18:46:36+00:00"
        }
    ],
    "Owner": {
        "ID": "8d1069d9c60f0fd3a07a44000b77dfdc4c68252f1a82a58f66b1eae3da53b79b"
    },
    "Prefix": null
}
```

Step:3 $ aws s3api get-bucket-versioning --bucket sagar-007

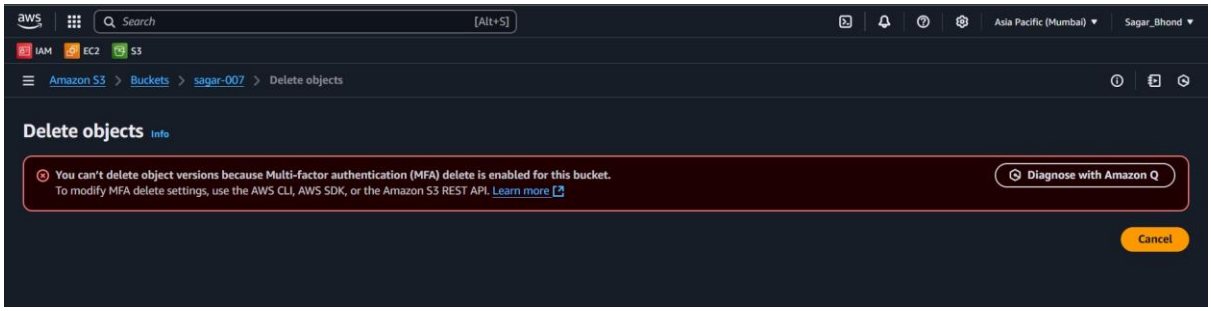
//check versioning is on or off

```
sagar_c7otrfh@SagarBhond MINGW64 ~
$ aws s3api get-bucket-versioning --bucket sagar-007
{
    "Status": "Enabled",
    "MFADelete": "Disabled"
}
```

Step:4 $ aws s3api put-bucket-versioning --bucket sagar-007 --versioning-configuration Status=Enabled,MFADelete=Enabled --mfa "arn:aws:iam::120569624603:mfa/rootuser 270127"

//enable mfa providing root arn and root mfa code

```
C:\Users\sagar_c7otrfh>aws s3api put-bucket-versioning --bucket sagar-007 --versioning-configuration Status=Enabled,MFAD
elete=Enabled --mfa "arn:aws:iam::120569624603:mfa/rootuser 270127"
```



//if you want to delete file

Step:1 $ aws s3api delete-object -- bucket sagar-007 -- key .jpg

(File Was deleted because it protect only Versionong File)

Step:2 $ aws s3api delete-object -- bucket sagar-007 -- key . jpg -- version-id Eqv102ILFiiBJPL2kCWVUBOXPSw>

(now Try to delet VersioniD File It required MFA)

Step:3 $ aws s3api put-bucket-versioning -- bucket sagar-007 -- versioning-configuration Status=Enabled, MFADelete=Disab]

Note: - It is very in-secure methos because your "access key" & "secrate key" store on locally.

Anyone can access your laptop can access your aws keys.

Versioning must be enable on Bucket.

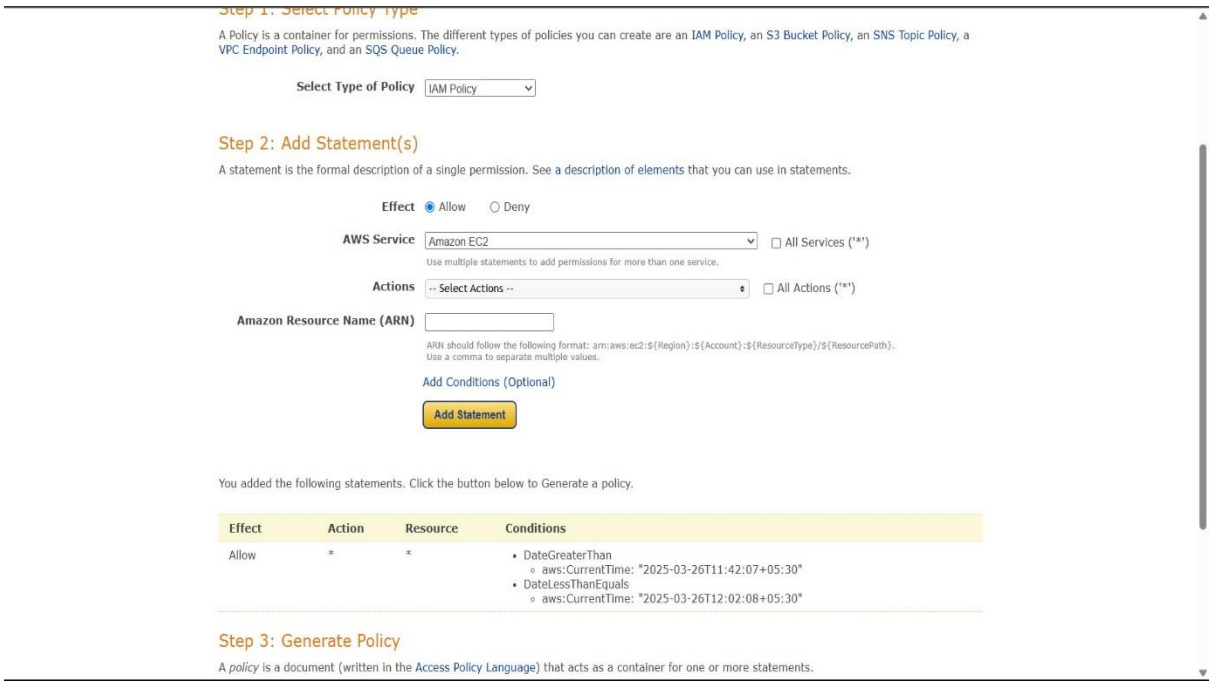
Task No:3

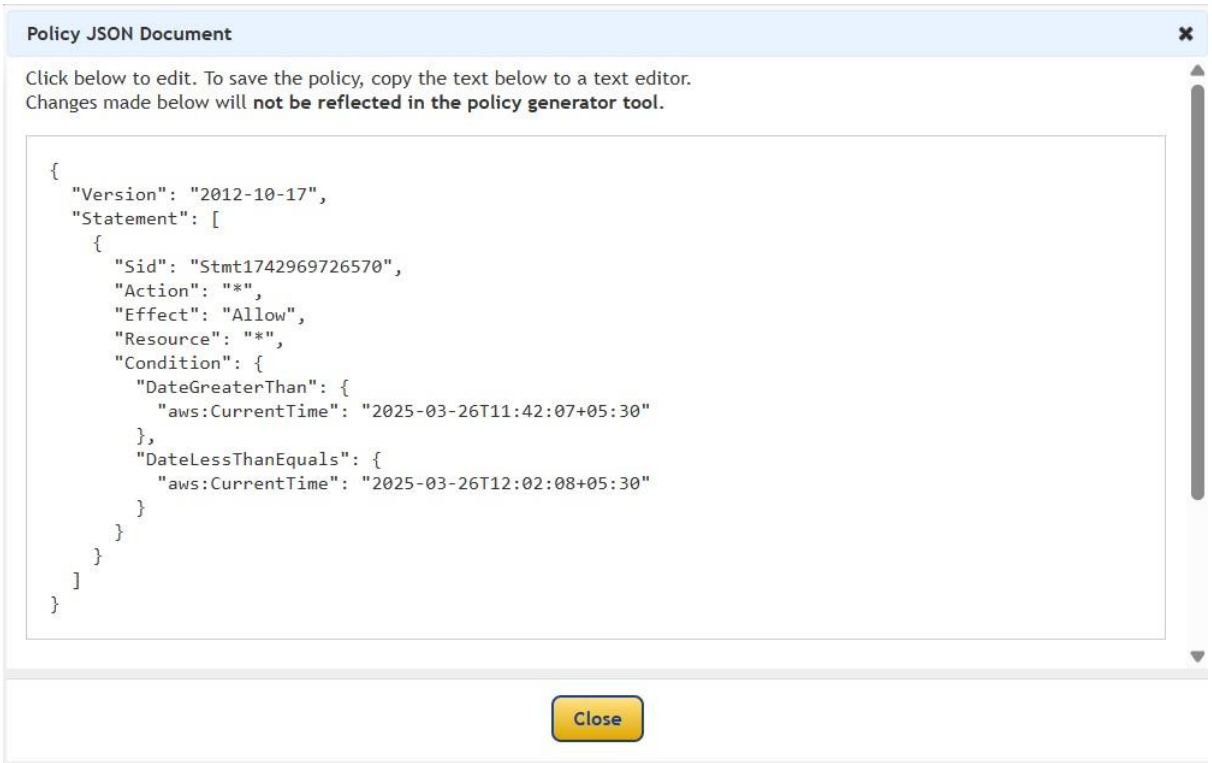
Name: sagar

* IAMAutoStopPolicy- after 10 minute.

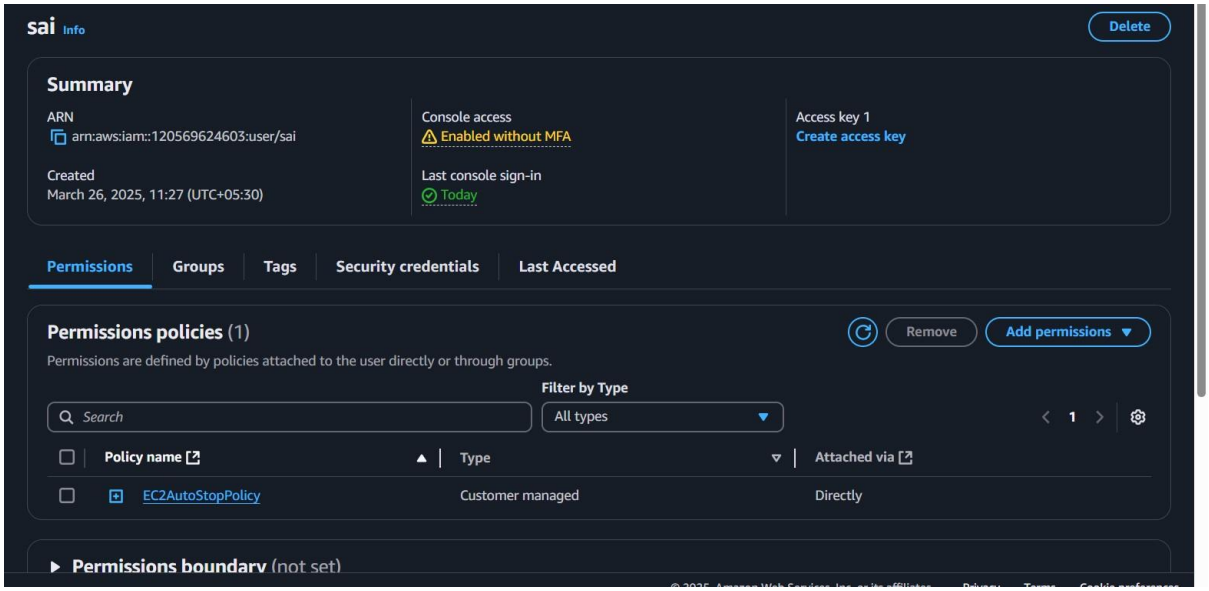
Step:1 create user sai.

Step:2 create policey to logout sai user after 10 minute.



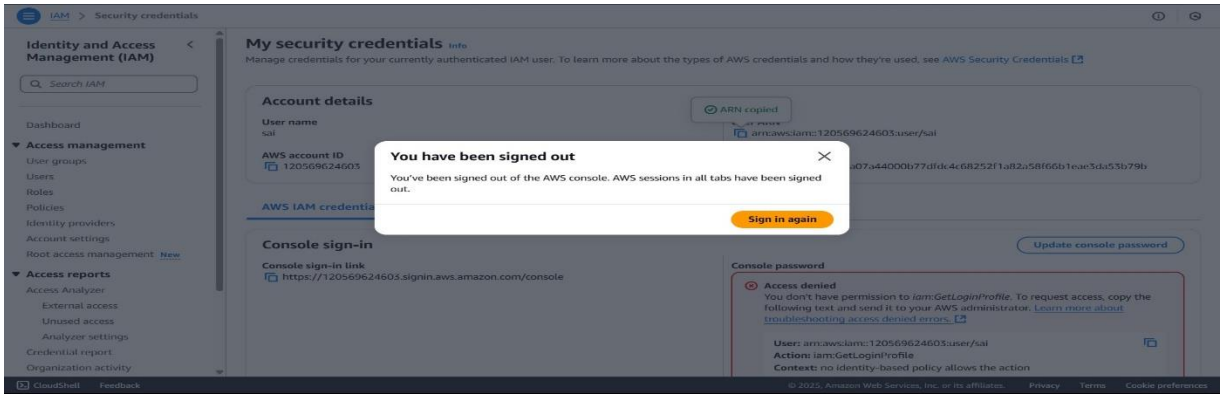


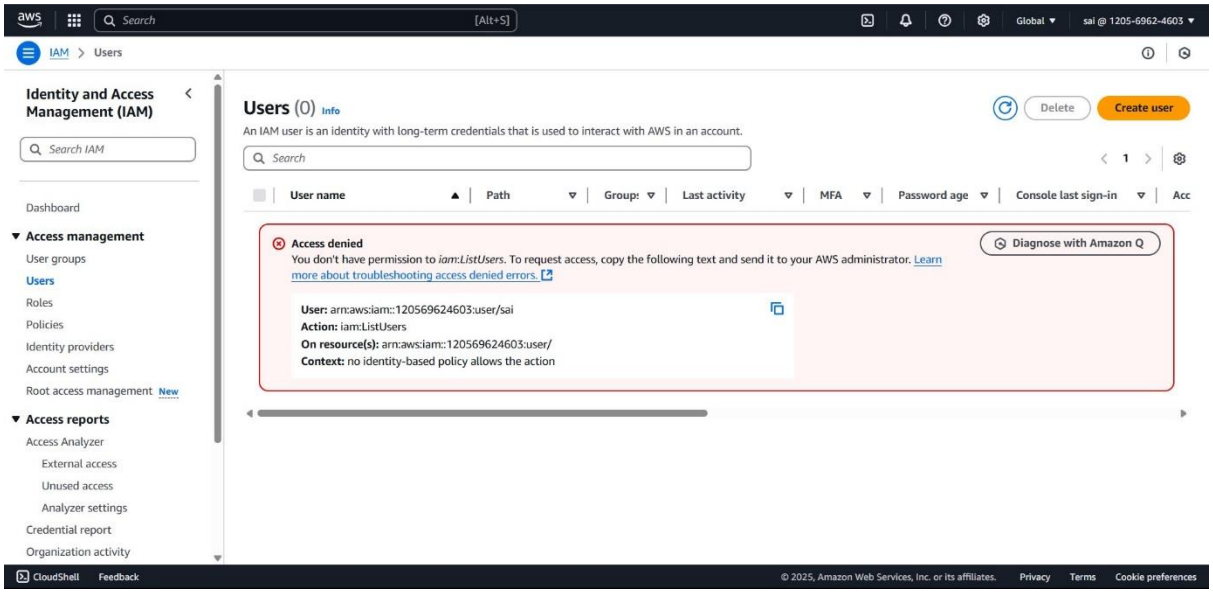
Step:3 attach that policy to sai user



Step:4 login sai user with 1) username = sai 2) password sai@1234

//after 10 minute it automatically logout





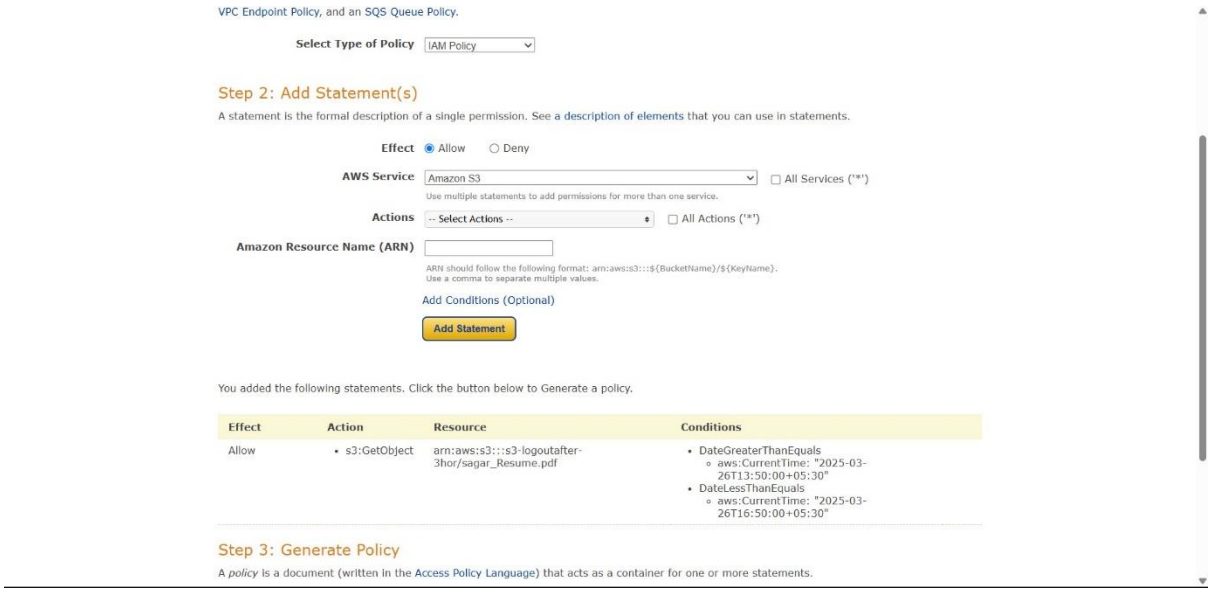
Task No:4

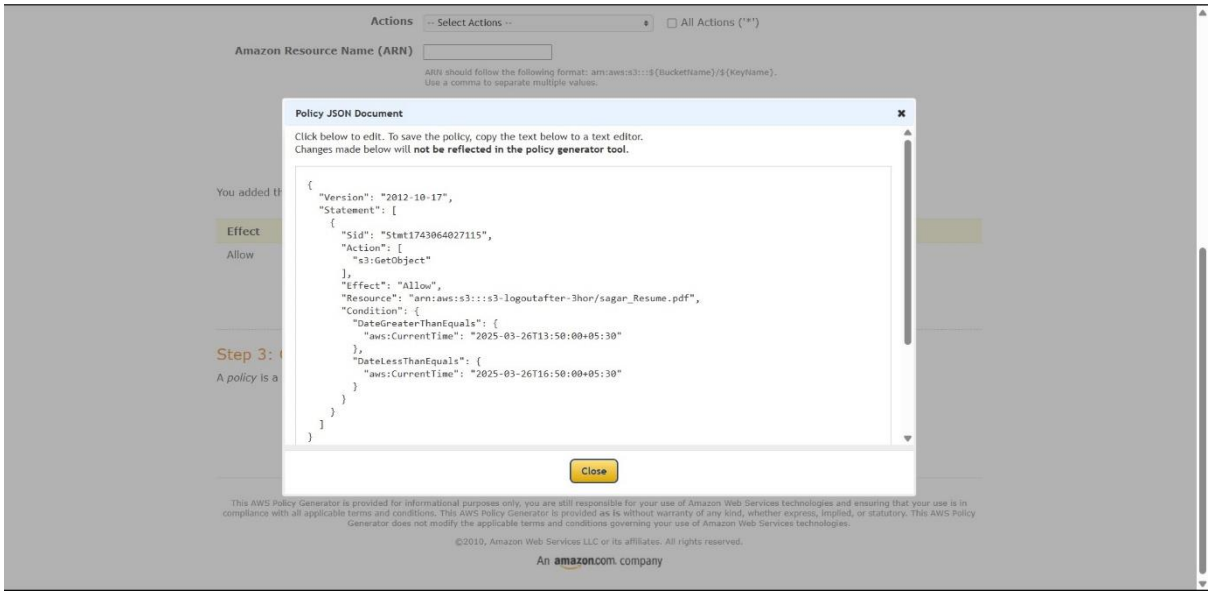
Name: sagar

* Grant Permition Via Json s3AutoStopPolicy- after 3 hr.

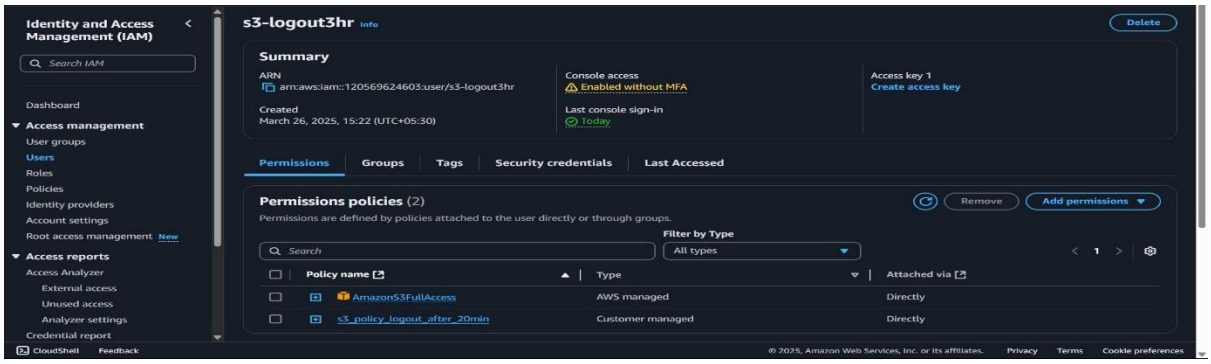
Step:1 create user s3-logout3hr.

Step:2 create policey to Grant Permition through json.





Step:3 attach that policy to **s3-logoutafter-3hor** user

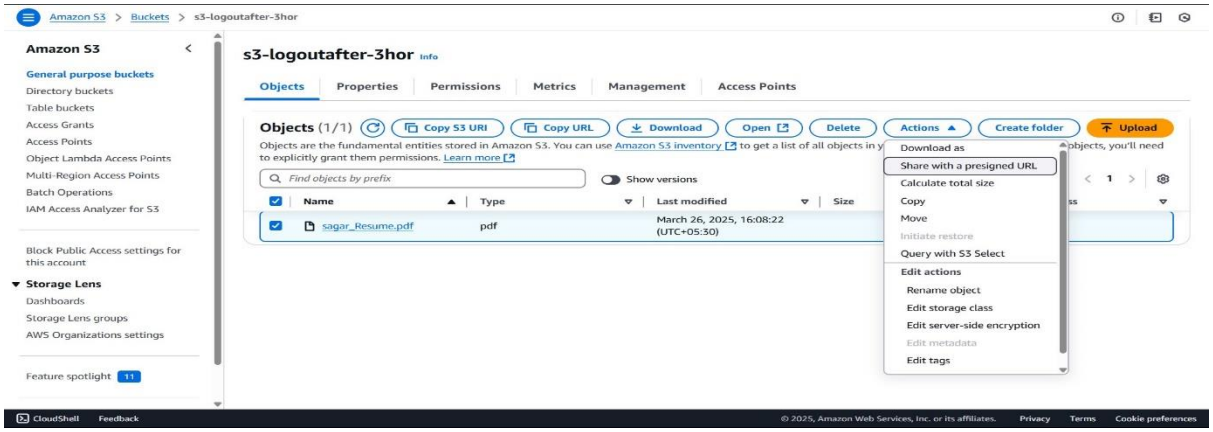


Step:4 login s3-logoutafter-3hor user with 1) username = s3-logoutafter-3hor 2) password **s3-logoutafter-3hor**

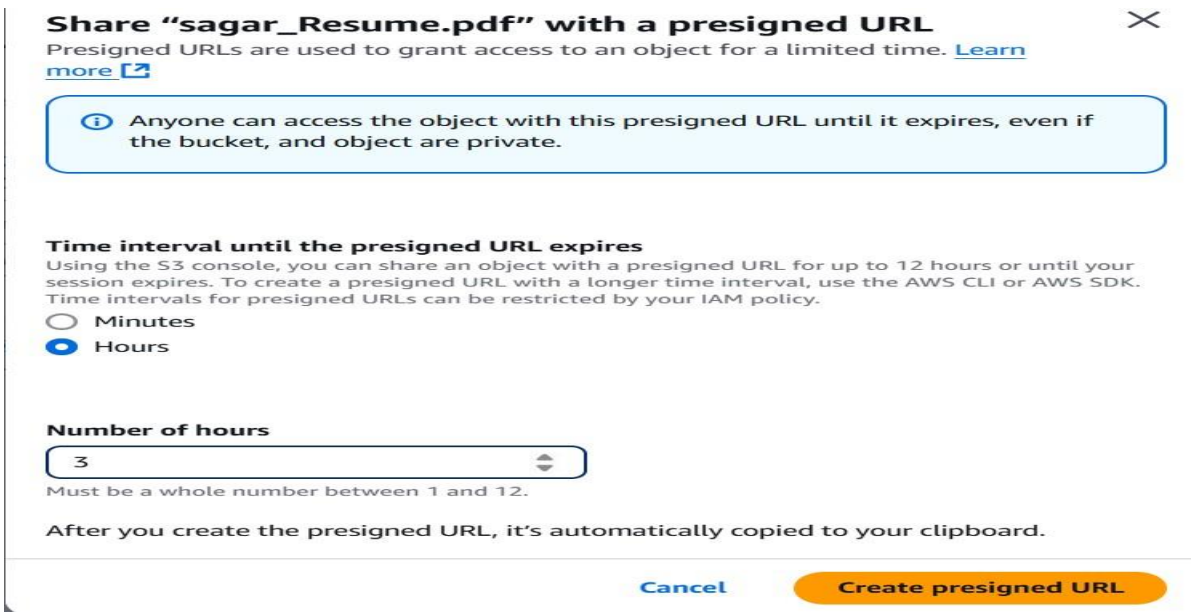
//create bucket name as s3-logoutafter-3hor

//uplode your resume on their

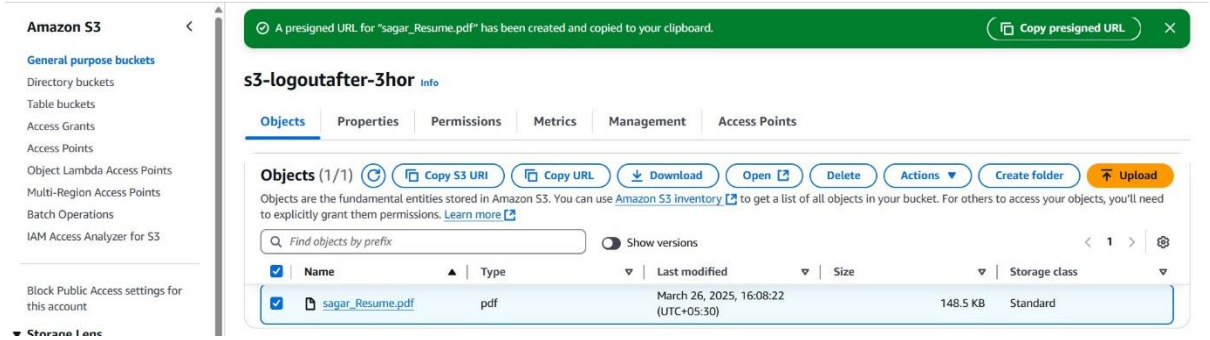
//select the resume click -> action -> share with a presigned url



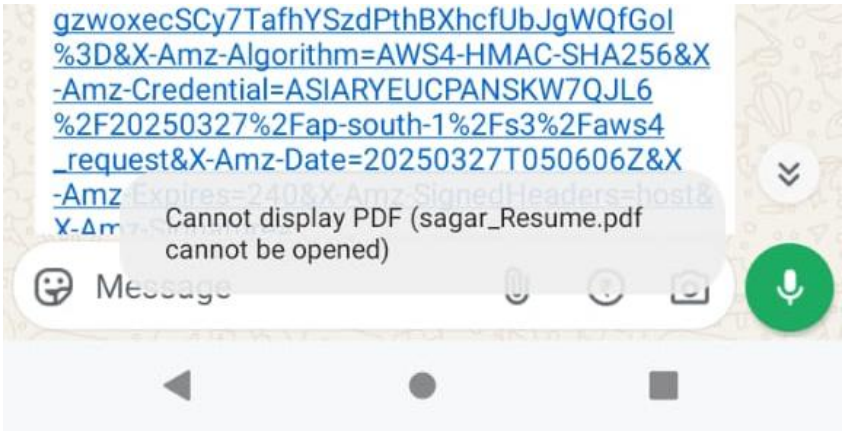
//set time of 3 hour



// after that it generate presigned url send this url to other



//after 3 hour while clicking this link it gives error like cannot display pdf (sagar_resume.pdf cannot be opened)



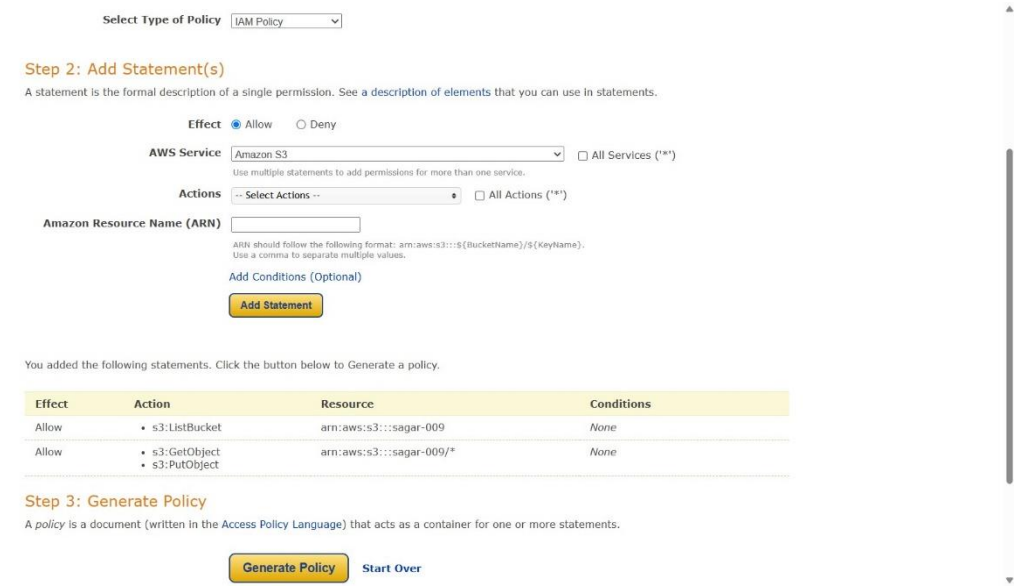
Task No:5

Name: sagar

* S3-to grant specific user permition-read,write,list.

Step:1 create user wiper.

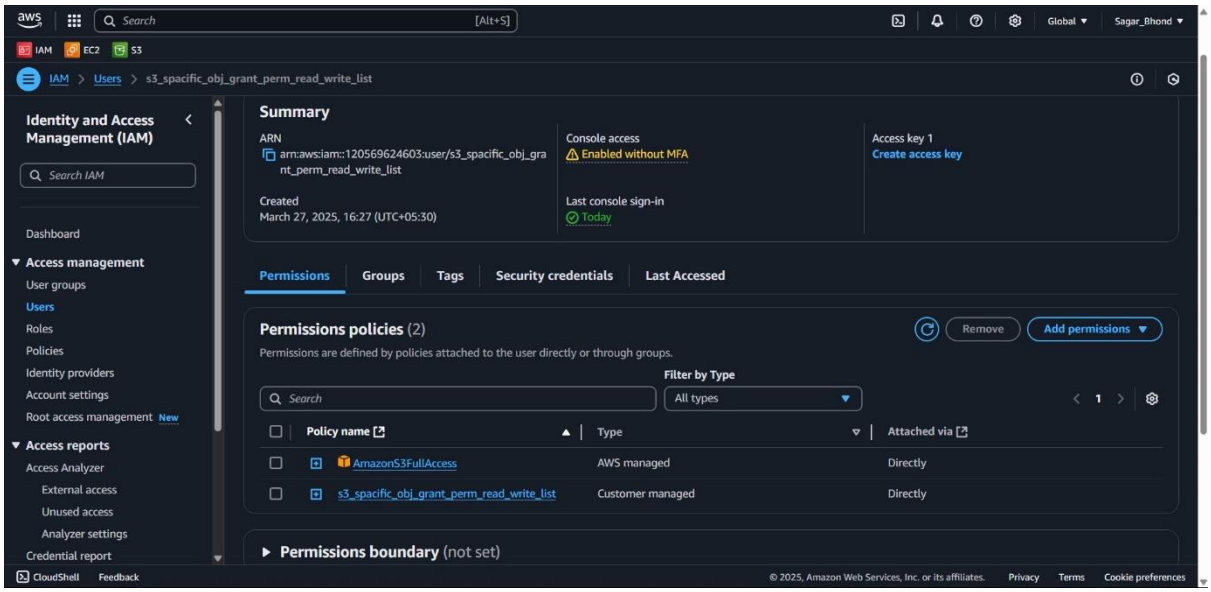
Step:2 create policy.



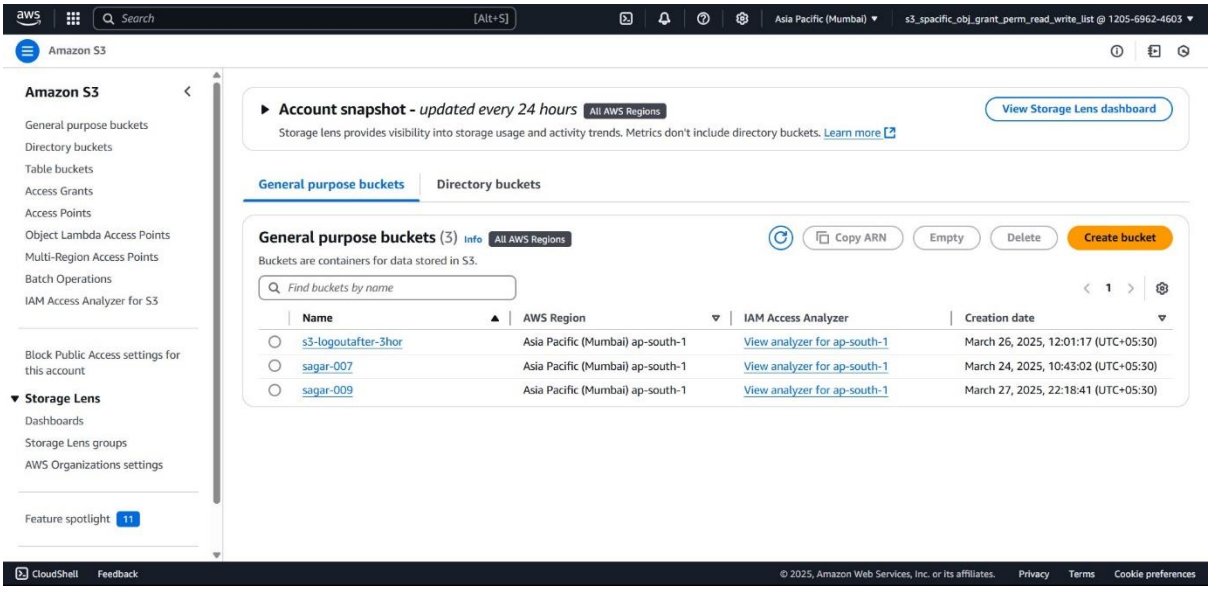

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1743094718987",
      "Action": [
        "s3:ListBucket"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::sagar-009"
    },
    {
      "Sid": "Stmt1743094779514",
      "Action": [
        "s3:GetObject",
        "s3:PutObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::sagar-009/*"
    }
  ]
```


Close

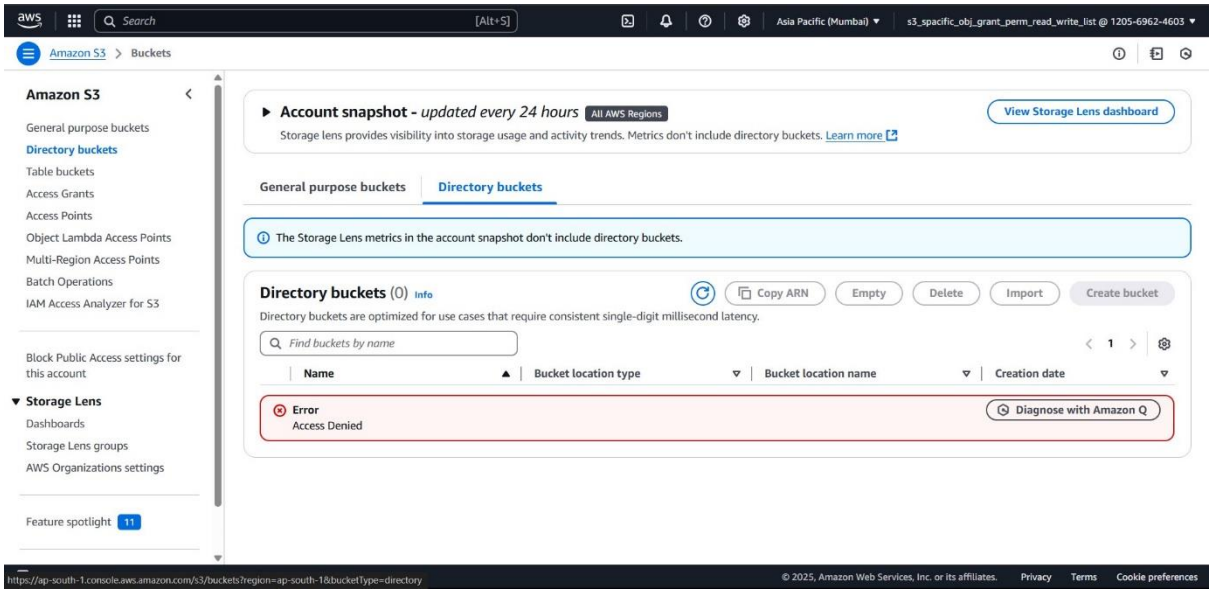
Step:3 give access to only s3_spacific_obj_grant_perm_read_write_list.



//previously when I give only s3fullacces it look like this type

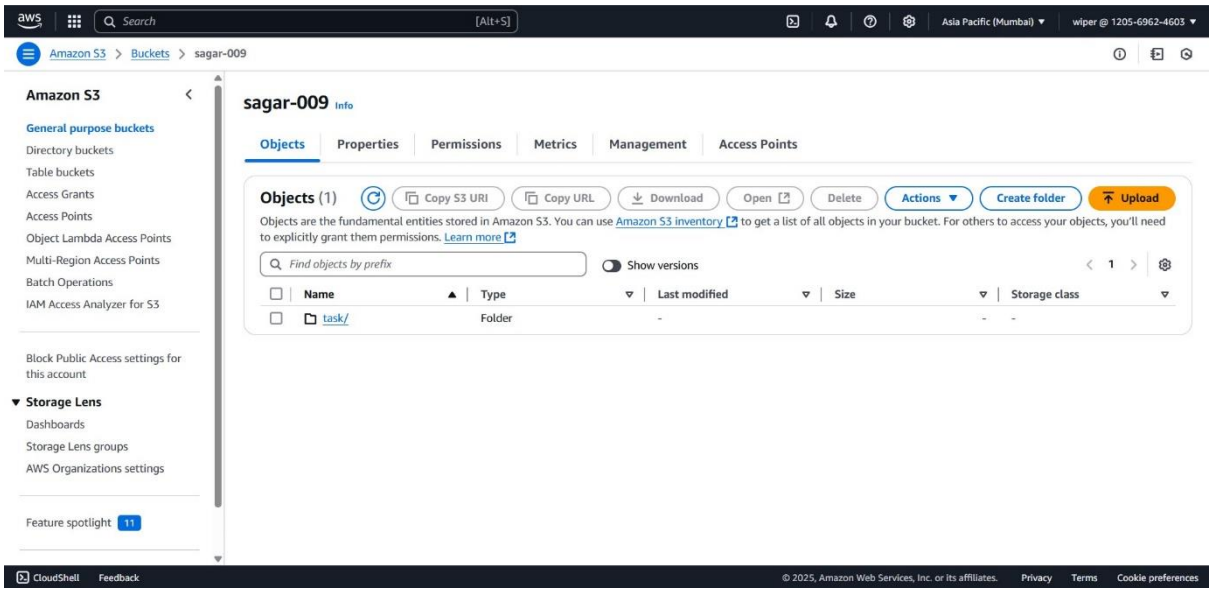


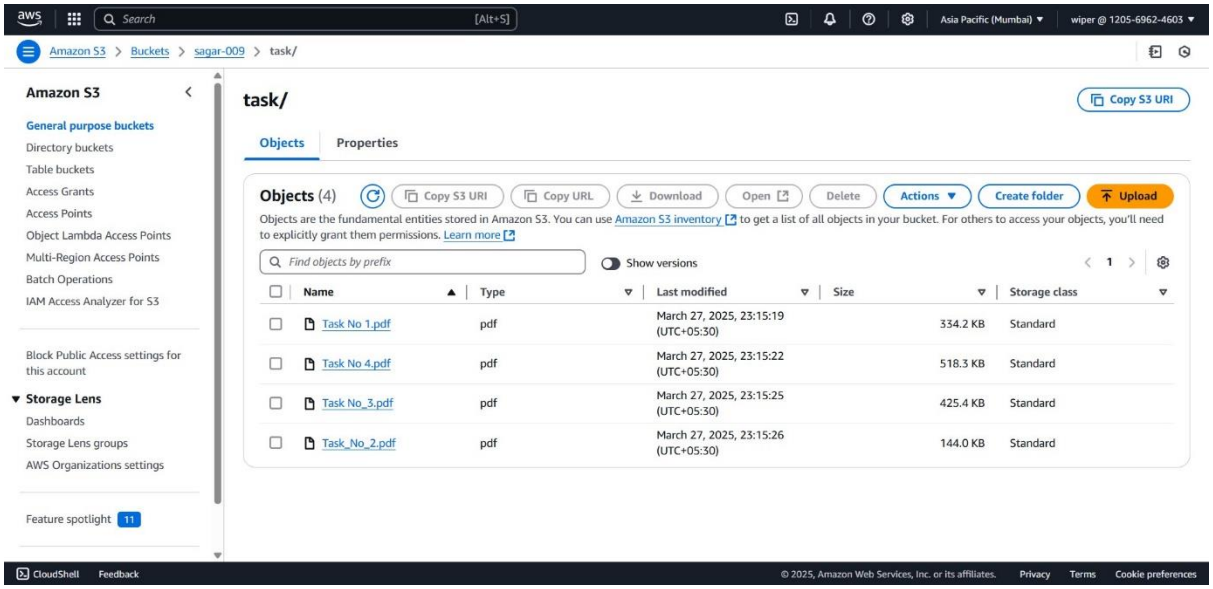
//after I add only s3_spacific_obj_grant_perm_read_write_list it look like.



//edit the url bucket name in which bucket did you apply permition.

//after that it give the sagar-009 bucket access.




Search
[Alt+S]
Asia Pacific (Mumbai)
wiper @ 1205-6962-4603
Amazon S3 > Buckets > sagar-009 > task/
Amazon S3
General purpose buckets
Directory buckets
Table buckets
Access Grants
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
Storage Lens groups
AWS Organizations settings
Feature spotlight 11
task/
Copy S3 URI
Objects
Properties
Objects (4)
Copy S3 URI
Copy URL
Download
Open
Delete
Actions
Create folder
Upload
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
Find objects by prefix
Show versions
1
Name
Type
Last modified
Size
Storage class
Task No 1.pdf
pdf
March 27, 2025, 23:15:19 (UTC+05:30)
334.2 KB
Standard
Task No 4.pdf
pdf
March 27, 2025, 23:15:22 (UTC+05:30)
518.3 KB
Standard
Task No_3.pdf
pdf
March 27, 2025, 23:15:25 (UTC+05:30)
425.4 KB
Standard
Task_No_2.pdf
pdf
March 27, 2025, 23:15:26 (UTC+05:30)
144.0 KB
Standard
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Task No:6

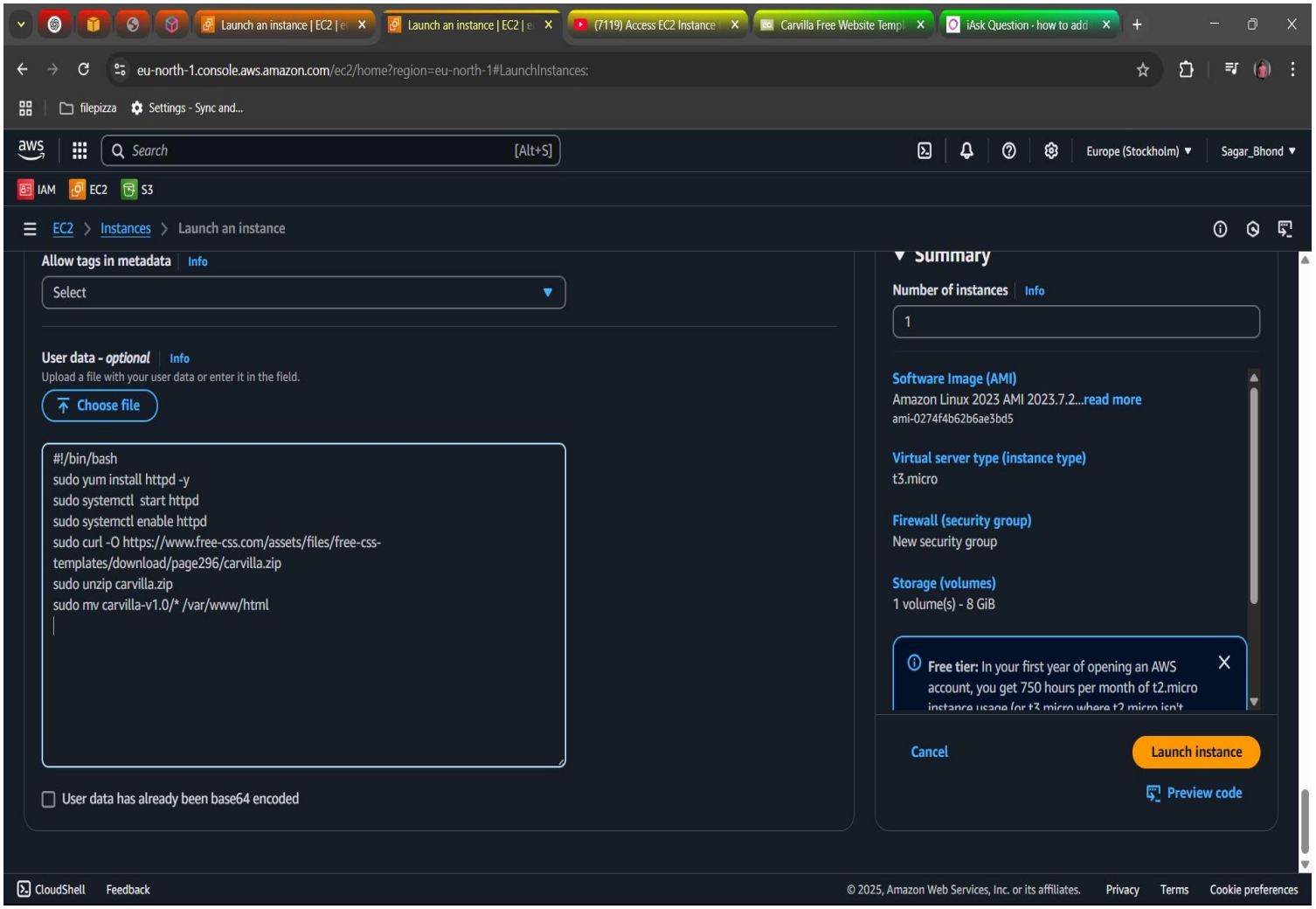
Name: sagar
ec2 instance name:- scripted-web-hosting

//after that Allow HTTP traffic from the internet

// go to Advanced details option

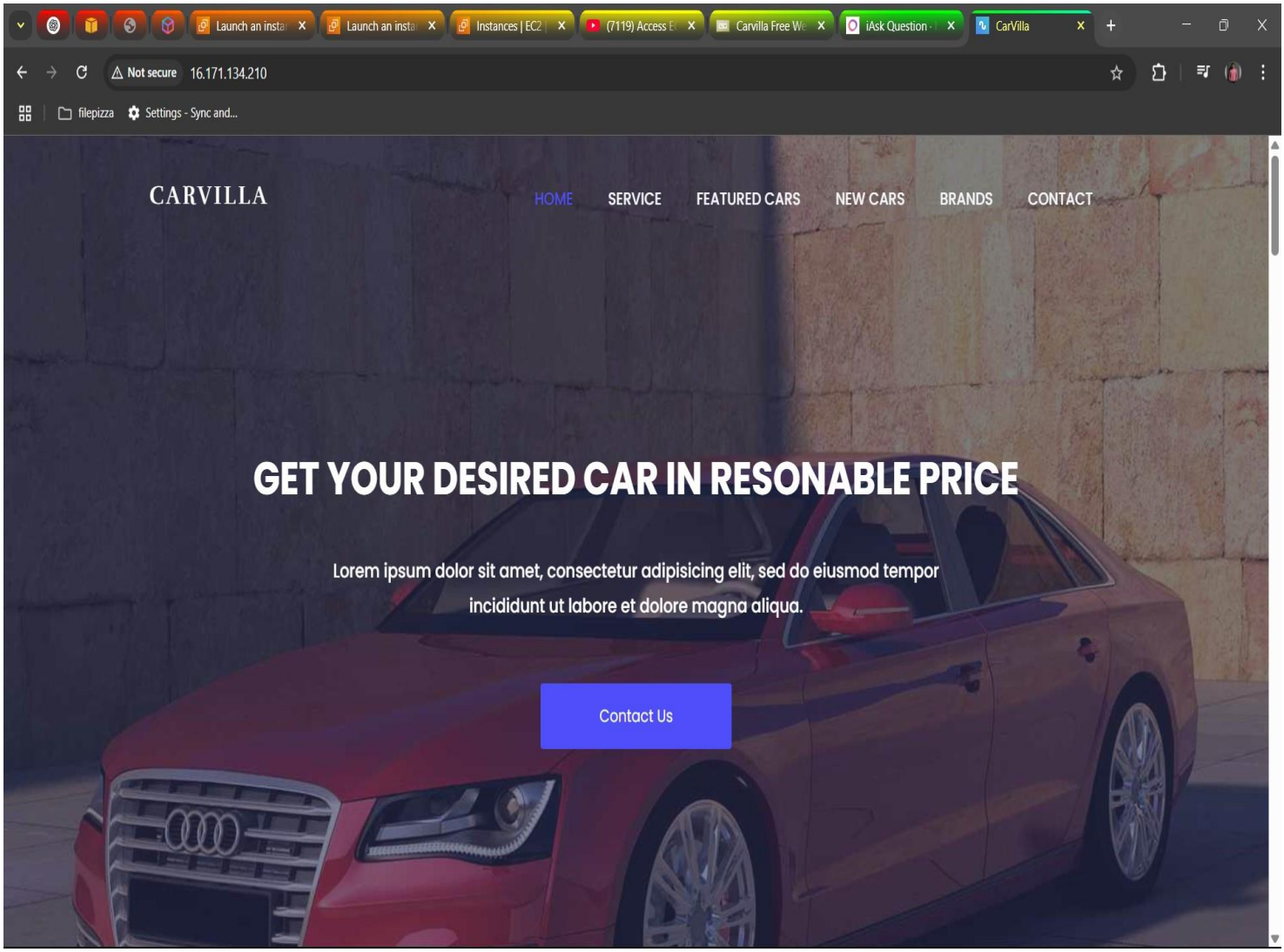
// give key pair name san

// write User data - optional script



//open scripted-web-hosting and go to detail and copy ipv4 public ip address

// after that hit the ipv4 public ip address http://16.171.134.210/

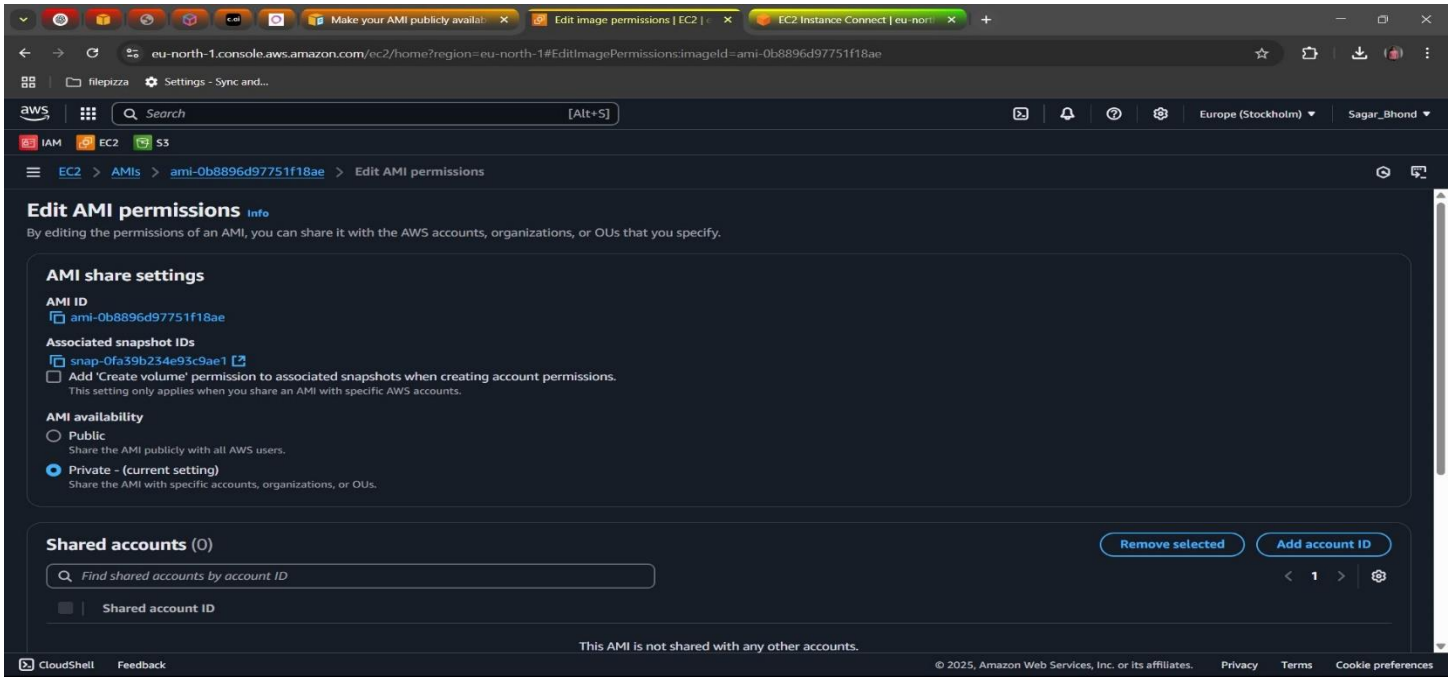


Task No:7

Name: sagar

ec2 :- Unblock Public Access for AMIs

// select ami -> Edit AMI permissions it shows disable public option.



// by using this command it display block-new-sharing or unblocked.

$ aws ec2 get-image-block-public-access-state --region eu-north-1

//To disable block public access

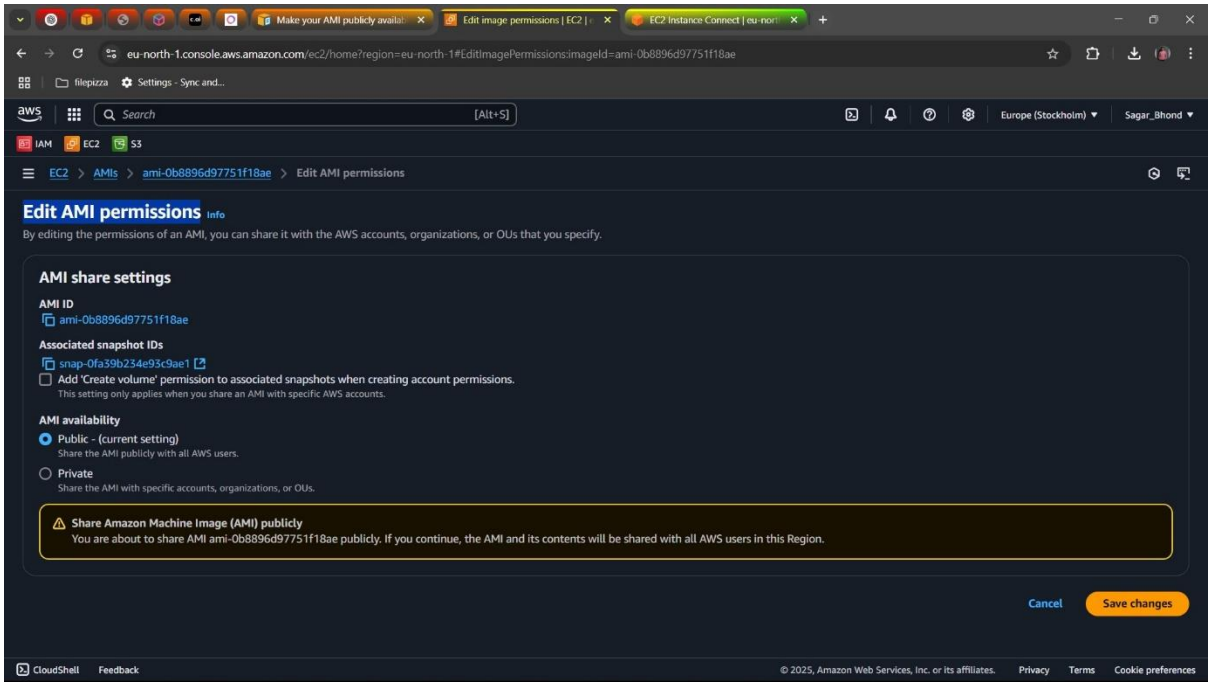
$ aws ec2 disable-image-block-public-access --region eu-north-1

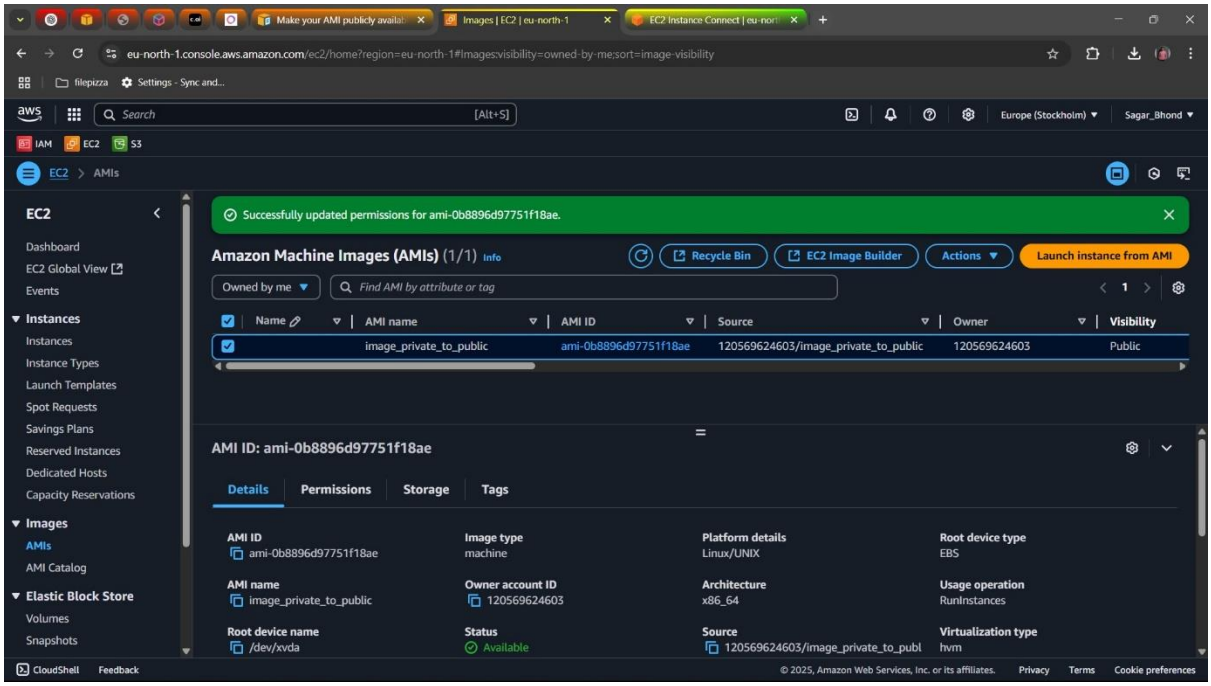
//to see unblocked or not

$ aws ec2 get-image-block-public-access-state --region eu-north-1

```
[ec2-user@ip-172-31-32-133 ~]$ aws ec2 get-image-block-public-access-state --region eu-north-1
{
    "ImageBlockPublicAccessState": "block-new-sharing",
    "ManagedBy": "account"
}
[ec2-user@ip-172-31-32-133 ~]$ aws ec2 disable-image-block-public-access --region eu-north-1
{
    "ImageBlockPublicAccessState": "unblocked"
}
[ec2-user@ip-172-31-32-133 ~]$ aws ec2 get-image-block-public-access-state --region eu-north-1
{
    "ImageBlockPublicAccessState": "unblocked",
    "ManagedBy": "account"
}
[ec2-user@ip-172-31-32-133 ~]$
```

// after that enable public access it enable that option





Task No:8

Name: sagar

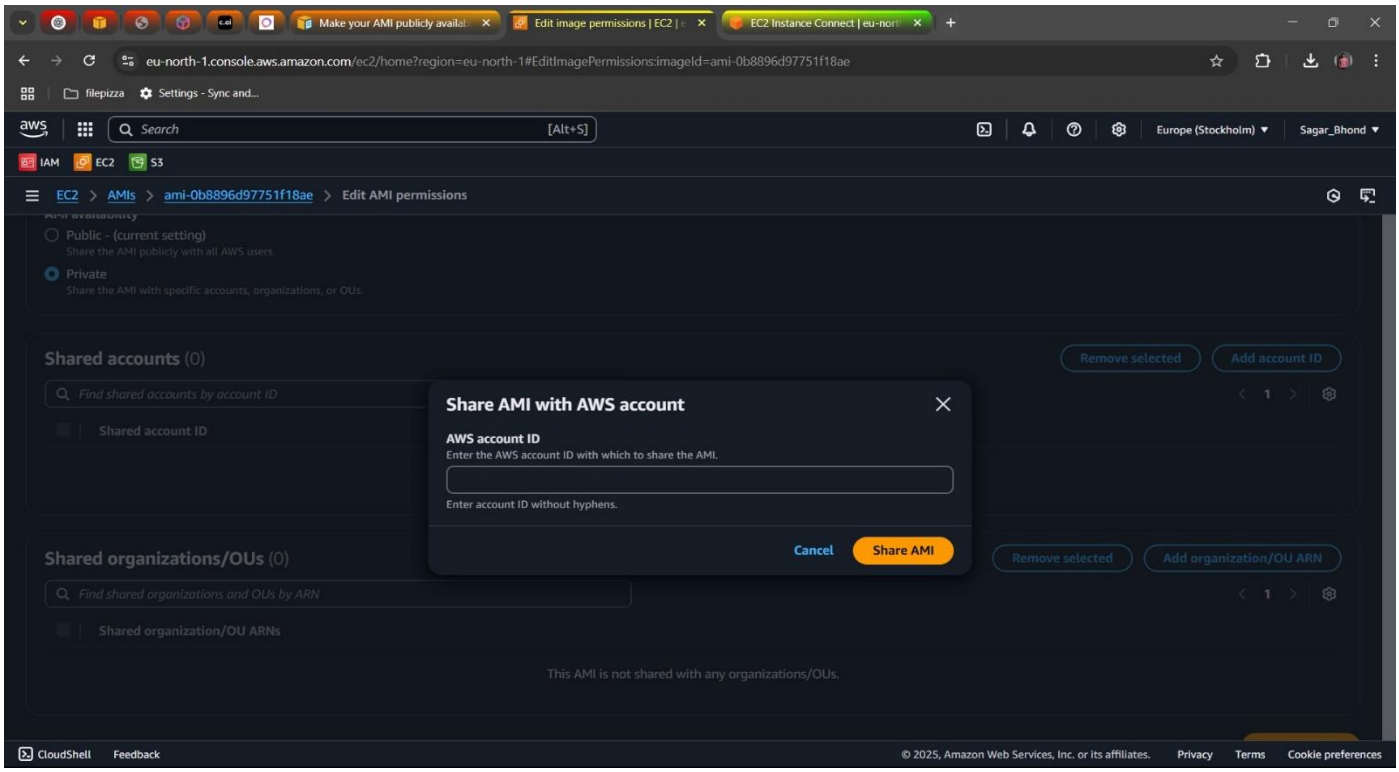
ec2 :- Share an EC2 AMI with a Friend’s AWS Account.

// get your friend aws account id

//go to ec2 service -> ami -> select ami -> action -> Edit AMI permissions.

// AMI availability make private it gives option to share ami resource.

//past Shared accounts ID.



//after that it show This AMI is not shared with any organizations/OUs.

//enable it by using command

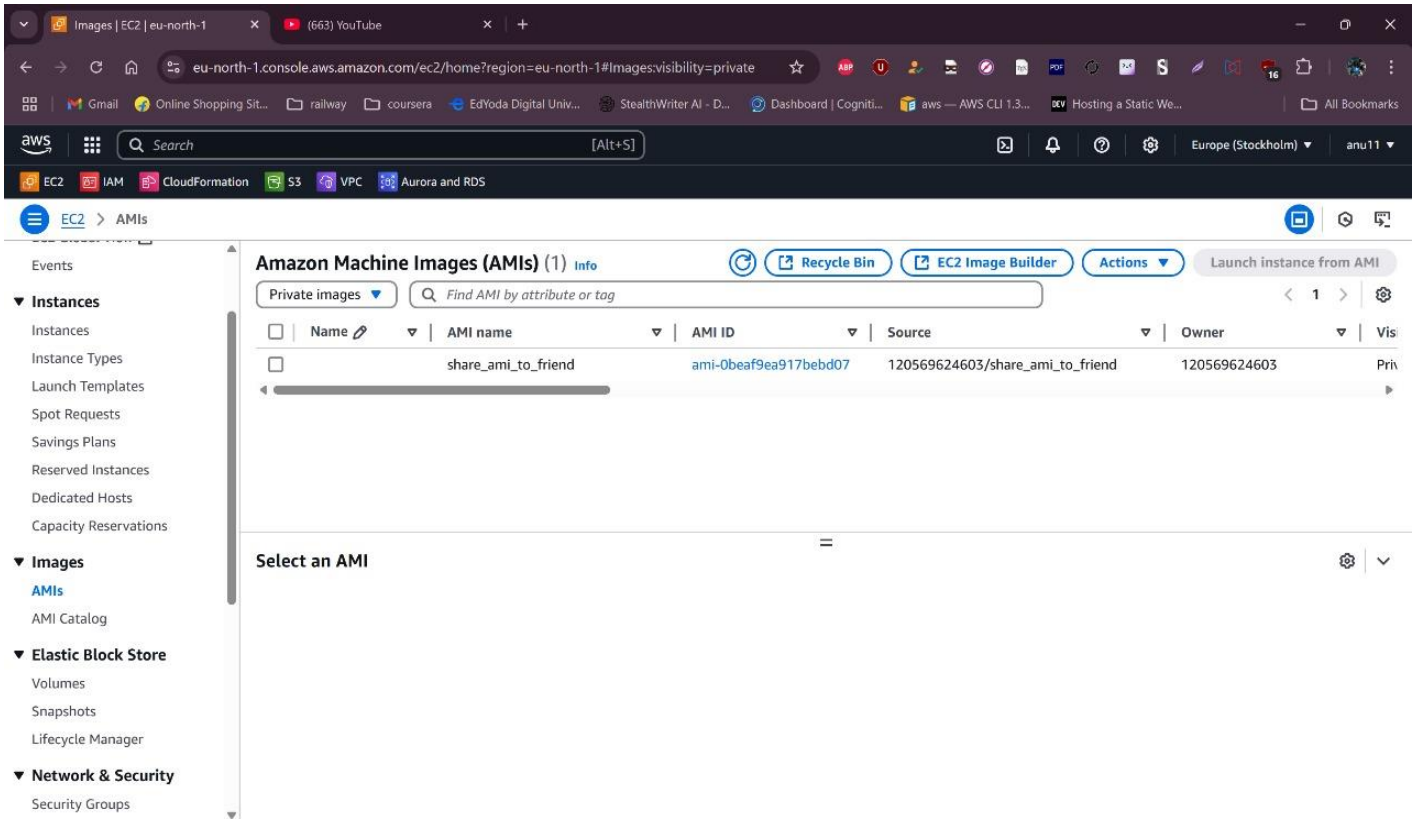
```
[ec2-user@ip-172-31-32-195 ~]$ aws configure
AWS Access Key ID [None]: AKIARYEUCPAN4HJ2TEEQ
AWS Secret Access Key [None]: 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
Default region name [None]:
Default output format [None]:
[ec2-user@ip-172-31-32-195 ~]$ aws ec2 describe-image-attribute --image-id ami-0abcdef1234567890 --attribute launchPermission

An error occurred (InvalidAMIID.Malformed) when calling the DescribeImageAttribute operation: The image ID 'ami-0abcdef1234567890' is malformed
[ec2-user@ip-172-31-32-195 ~]$ aws ec2 describe-image-attribute --image-id ami-0beaf9ea917bebd07 --attribute launchPermission
{
    "ImageId": "ami-0beaf9ea917bebd07",
    "LaunchPermissions": [
        {
            "UserId": "288761773173"
        }
    ]
}
[ec2-user@ip-172-31-32-195 ~]$ aws ec2 modify-image-attribute --image-id
```

//To share an AMI with an organization

```
[ec2-user@ip-172-31-32-195 ~]$ aws ec2 modify-image-attribute --image-id ami-0beaf9ea917bebd07 --launch-permission "Add=[{OrganizationArn=arn:aws:organizations::123456
789012:organization/o-123example}]"
[ec2-user@ip-172-31-32-195 ~]$
```

//after that check friends account ec2 ami resource whether ami is recived or not.



Task No:9

Name: sagar

ec2 :- EC2 recover loss key-pair .

// host web sit in our instance.

// first gen key after that go to git bash and next change dir to that key

```
sagar_c7otrfh@SagarBhond MINGW64 /C/ProgramData/Microsoft/Windows/Start Menu/Pro
grams/Git
$ ssh -i recover-key.pem ec2-user@13.201.83.76
The authenticity of host '13.201.83.76 (13.201.83.76)' can't be established.
ED25519 key fingerprint is SHA256:bHAinIXT7NGm/HX1TO2pZUNKrnY1dQZp682vxRll3Xs.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:1: ec2-13-201-83-76.ap-south-1.compute.amazonaws.com
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '13.201.83.76' (ED25519) to the list of known hosts.
       #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Thu Apr 10 10:03:29 2025 from 13.233.177.3
[ec2-user@ip-172-31-11-200 ~]$
```

//after that logout the login exit

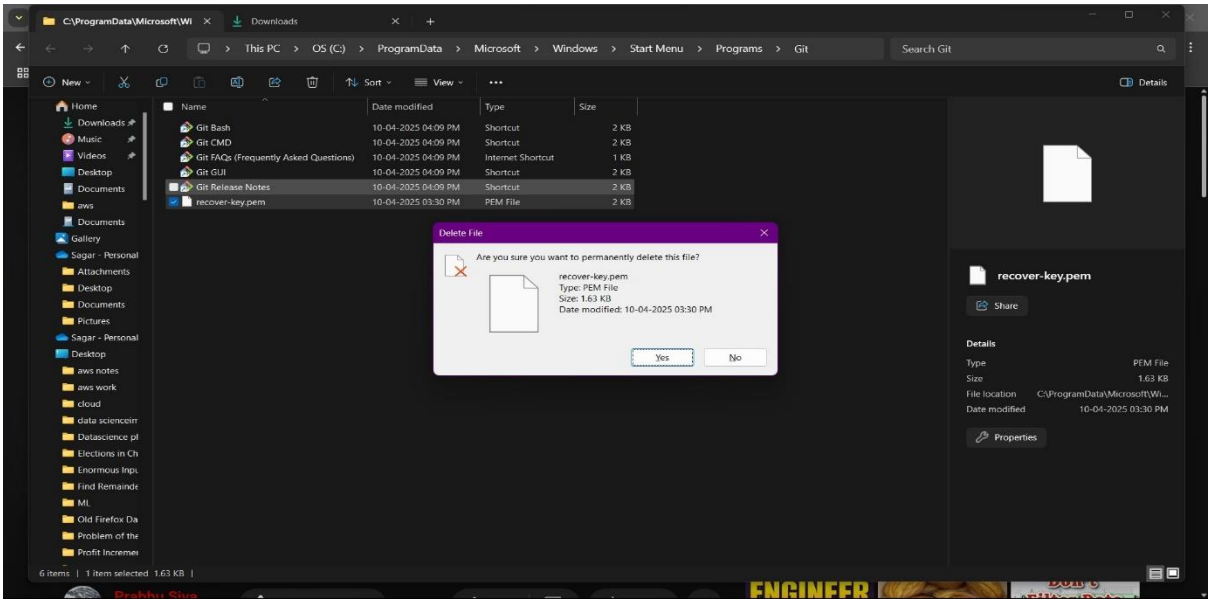
//after that delete key-pair permanently.

// in instance go to security -> security group link

// change inbound ssh custom to anyware.

// after that connect your instance to server of aws.

// how to create new key-pair ,first of all delete old key-pair.



// create new key-file

```
[ec2-user@ip-172-31-11-200 ~]$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ec2-user/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
```

```
[ec2-user@ip-172-31-11-200 ~]$ cd .ssh/
[ec2-user@ip-172-31-11-200 .ssh]$ ls
authorized_keys  id_rsa  id_rsa.pub
[ec2-user@ip-172-31-11-200 .ssh]$ cat id_rsa
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn
NhAAAAAwEAAQAAAYEAwbnrzbIk71yOJQTLXPsQ5BkbbTCgrXl12VnXyLzim3XnhSB/Hr4z
eOMM6YNmE97ZlgGBkIdBGYwPRWUiM184K14+LnzWrr1Iozy6XOOyWbjz6QB8trSiYKu8pK
GCzumVgmWegSkKPHC6Wpxy+avjCBGNBo3wfpemYIbwbHlqp6/WxXR2vN6SyxdluhRYnTl+
```

// private key convert into new formate.

```
[ec2-user@ip-172-31-11-200 .ssh]$ ssh-keygen -p -m PEM -f id_rsa
Key has comment 'ec2-user@ip-172-31-11-200.ap-south-1.compute.internal'
Enter new passphrase (empty for no passphrase):
Enter same passphrase again:
```

//change private key format pest in vs code lflr into lr formate and create private key in your device with .pem extention.

//change your ssh to anywhearipv-4.

// authorized_key have there old public key override the new key-pair whith authorize key

```
[ec2-user@ip-172-31-11-200 .ssh]$ cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDSwPXy56xp3PqvENf8J8d7YjxLNPJfcbWTEfq/2rDJJLjsPQAoBeANY9bnI4+67z4vxJ3zQtO0aFHj
5nGwI4BraCTYTC/BVNXb9moiVXzv02oM15ZVgL/HlYUimJRRs/btCTH4xJMmdkLeOEX7F5oAdVHywdznPZHjxCeOgJNvytDf8voDebJeDHF3xesCgFsl
zFj7Sjy6VxjLSw4gTgjC1w3ckqWAAH5Vb2DiinFuZ0jI3V recover-key
[ec2-user@ip-172-31-11-200 .ssh]$ cp id_rsa.pub authorized_keys
[ec2-user@ip-172-31-11-200 .ssh]$ cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDBuevNsiTvXI4lBMtc+xDkGRttMKCteXXZWdfIvOKbdeeFIH8evjN44wzpg2YT3tnWAYGQh0EZjA9
CZZ6BKQo8cLpanHL5q+MIEY0GjfB+l6ZghvBseWqnr9bFdHa83pLLF2W6FFidOX7UlYIN6HUXADa2qXmc2/7MvSO2MY6pc3/zWVviPTFqeFmq3mtiEn
QyYFfiOqLgO5DgYHqgizRNnntoDs9M8oD884KmCw+7OX+wD1ZO8PbTj2NPKk07ngo/jHEMcbNRI+zEMVbRY2osOr7+wqOqAcxEPFOYNgVDs9UVXwiH1
3+jGgozI/TkqFdryHx59pejBoUJ2G51TkqOa+8quSOtV7NeMJU= ec2-user@ip-172-31-11-200.ap-south-1.compute.internal
```

// login to your ec2 instance.

```
sagar_c7otrfh@SagarBhond MINGW64 /C/ProgramData/Microsoft/Windows/Start Menu/Pro
grams/Git
$ ssh -i new-privkey.pem ec2-user@13.201.83.76
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Thu Apr 10 11:27:55 2025 from 13.233.177.5
[ec2-user@ip-172-31-11-200 ~]$
```

//check your hosted web site are display or not.