AWS

1.Create keypair

2.Create Security group

3. Create linux EC2 instance in AWS

4. Create EC2 windows instance in AWS

5. Create an elastic IP Address on AWS using AWS Elastic IP

6. Create EBS volume and attach(mount) to Ec2

7.How to Use a Snapshot to Restore an AWS EC2 Instance

8.Introduction to AWS Elastic Load Balancing 9.Monolith and Microservice in AWS

10.RDS and connect MySQL

11. Create bucket and host static website

12.Multi-Factor Authentication for IAM User

13.Create IAM User with Multi-Factor Authentication

14.Creating IAM Users and Groups on AWS

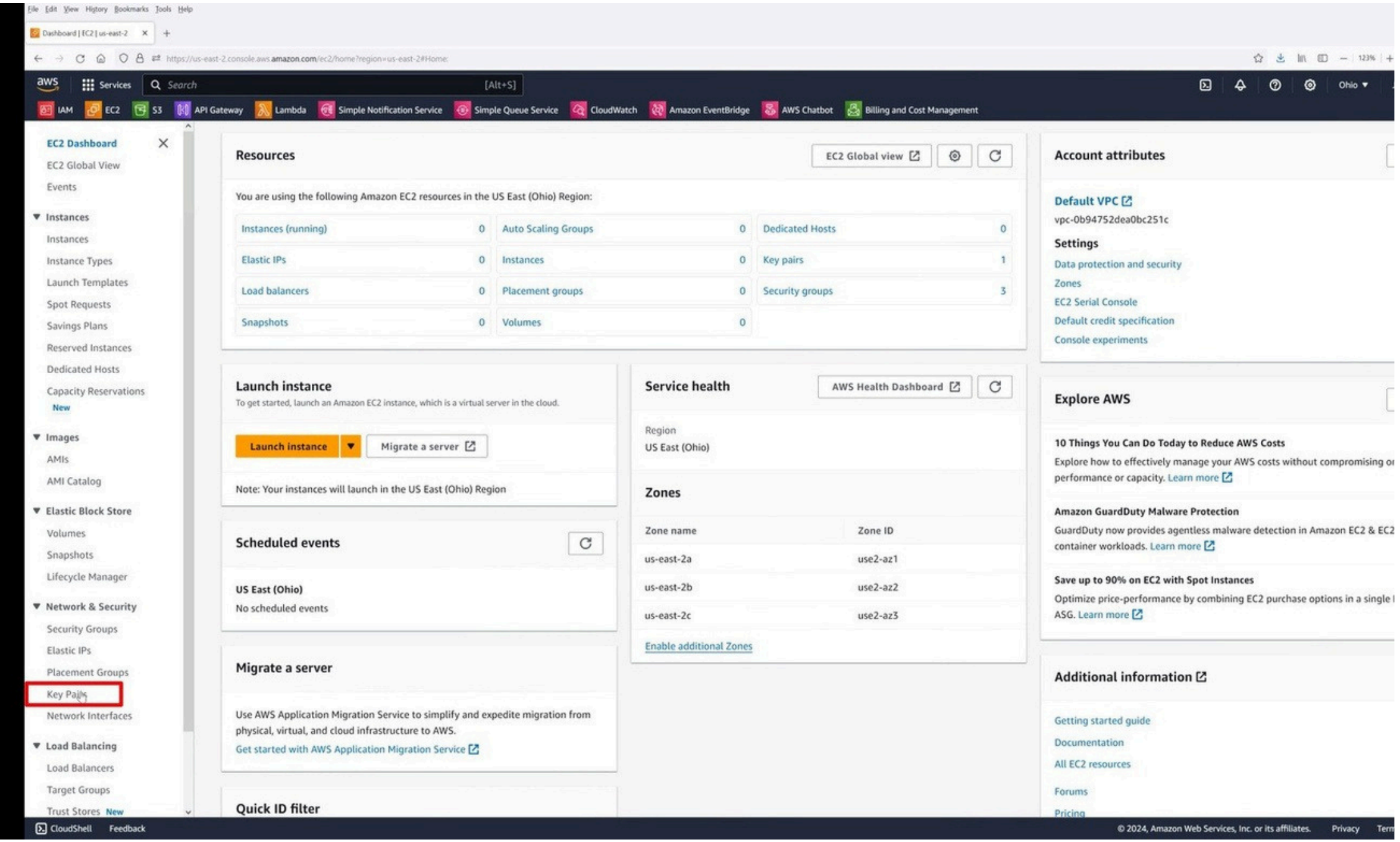
15.Create Role, Create policy, attach policy to Role and attach policy to IAM user

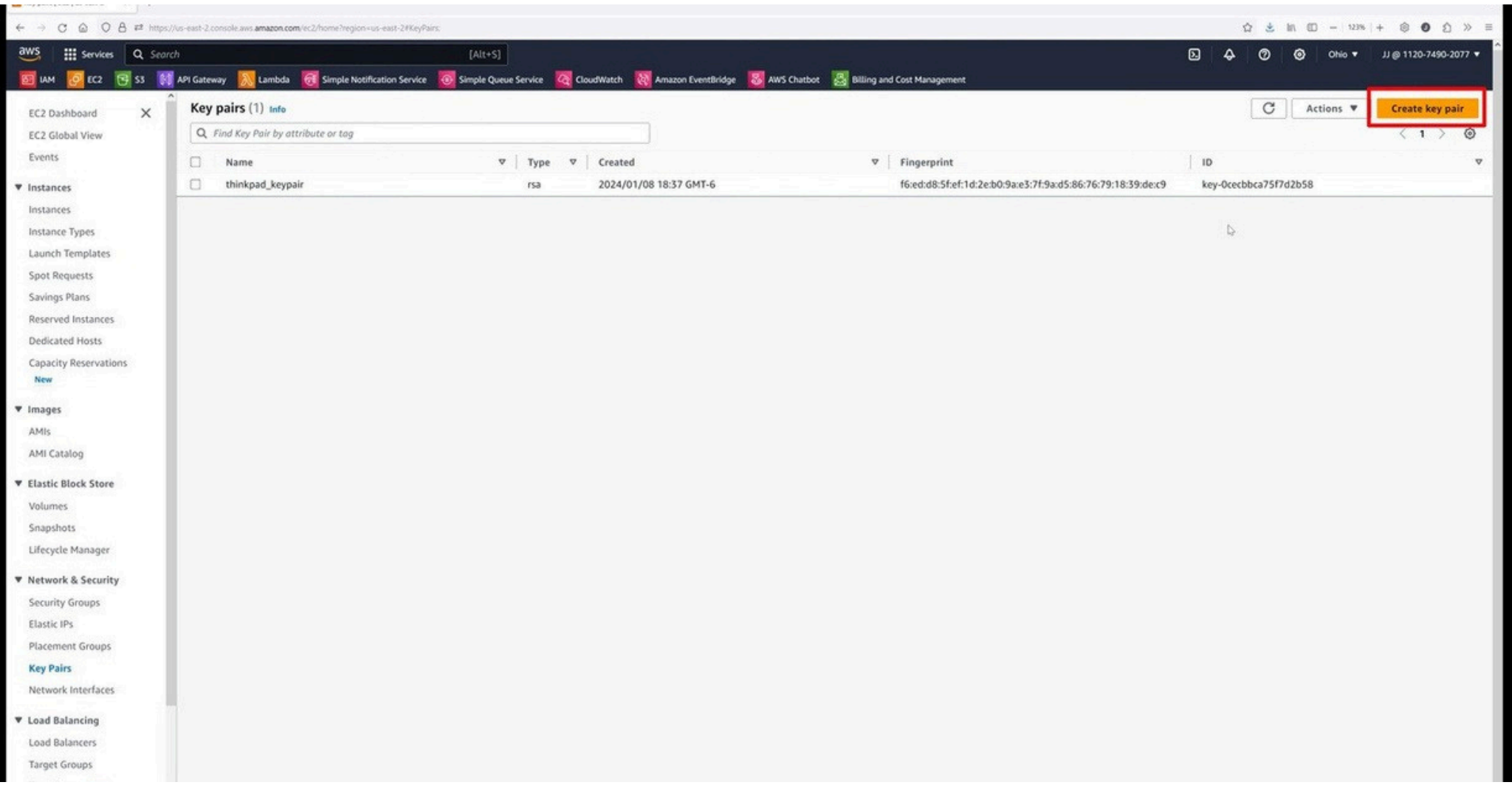
16.AWS CLI configuration

17.AWS cloud watch

18.Create Elastic Beanstalk

Create keypair




https://us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#KeyPairs:
Services
Search
[Alt+S]
Ohio
JJ @ 1120-7490-2077
IAM
EC2
S3
API Gateway
Lambda
Simple Notification Service
Simple Queue Service
CloudWatch
Amazon EventBridge
AWS Chatbot
Billing and Cost Management
EC2 Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
New
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Load Balancers
Target Groups
Key pairs (1) Info
Actions
Create key pair
Find Key Pair by attribute or tag
1
Name
Type
Created
Fingerprint
ID
thinkpad_keypair
rsa
2024/01/08 18:37 GMT-6
f6:ed:d8:5f:ef:1d:2e:b0:9a:e3:7f:9a:d5:86:76:79:18:39:de:c9
key-0cecbbca75f7d2b58

EC2 > Key pairs > Create key pair

Create key pair Info

Key pair

A key pair, consisting of a private key and a public key, is a set of security credentials that you use to prove your identity when connecting to an instance.

Name

JJ-Demo-TodaysDate

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type Info

- (•) RSA
- ( ) ED25519

Private key file format

- ( ) .pem
  For use with OpenSSH
- (•) .ppk
  For use with PuTTY

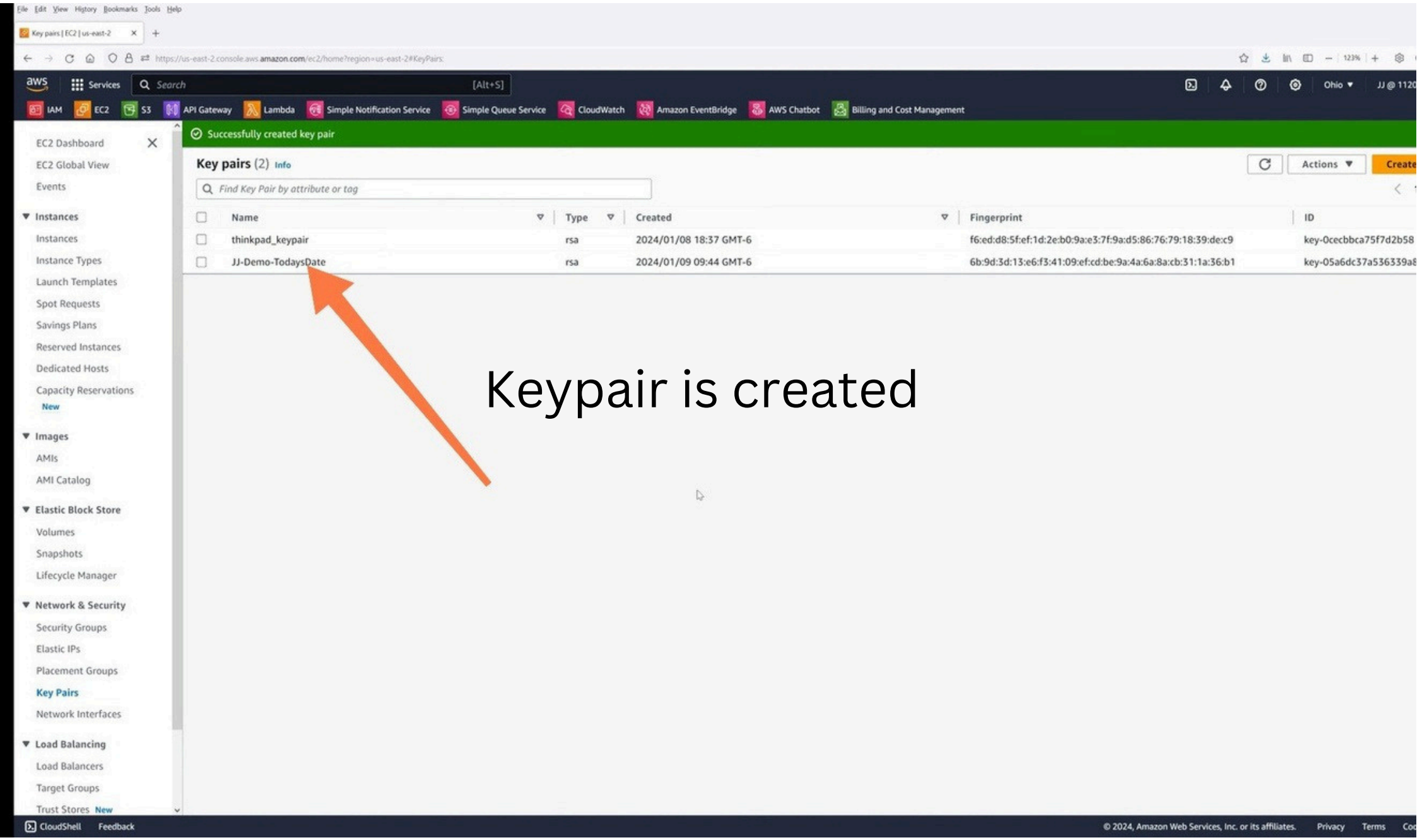
Tags - *optional*

No tags associated with the resource.

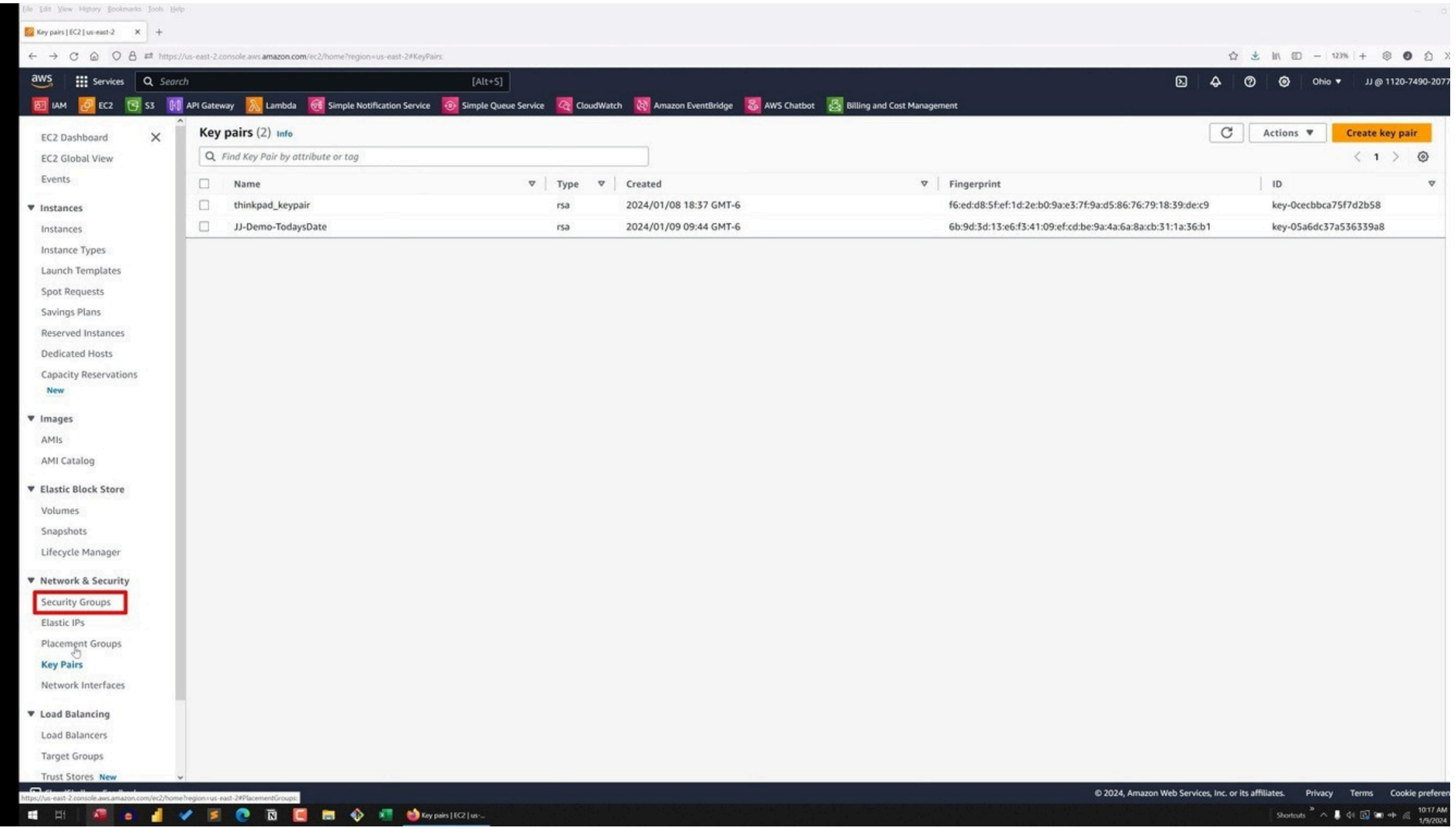
Add new tag

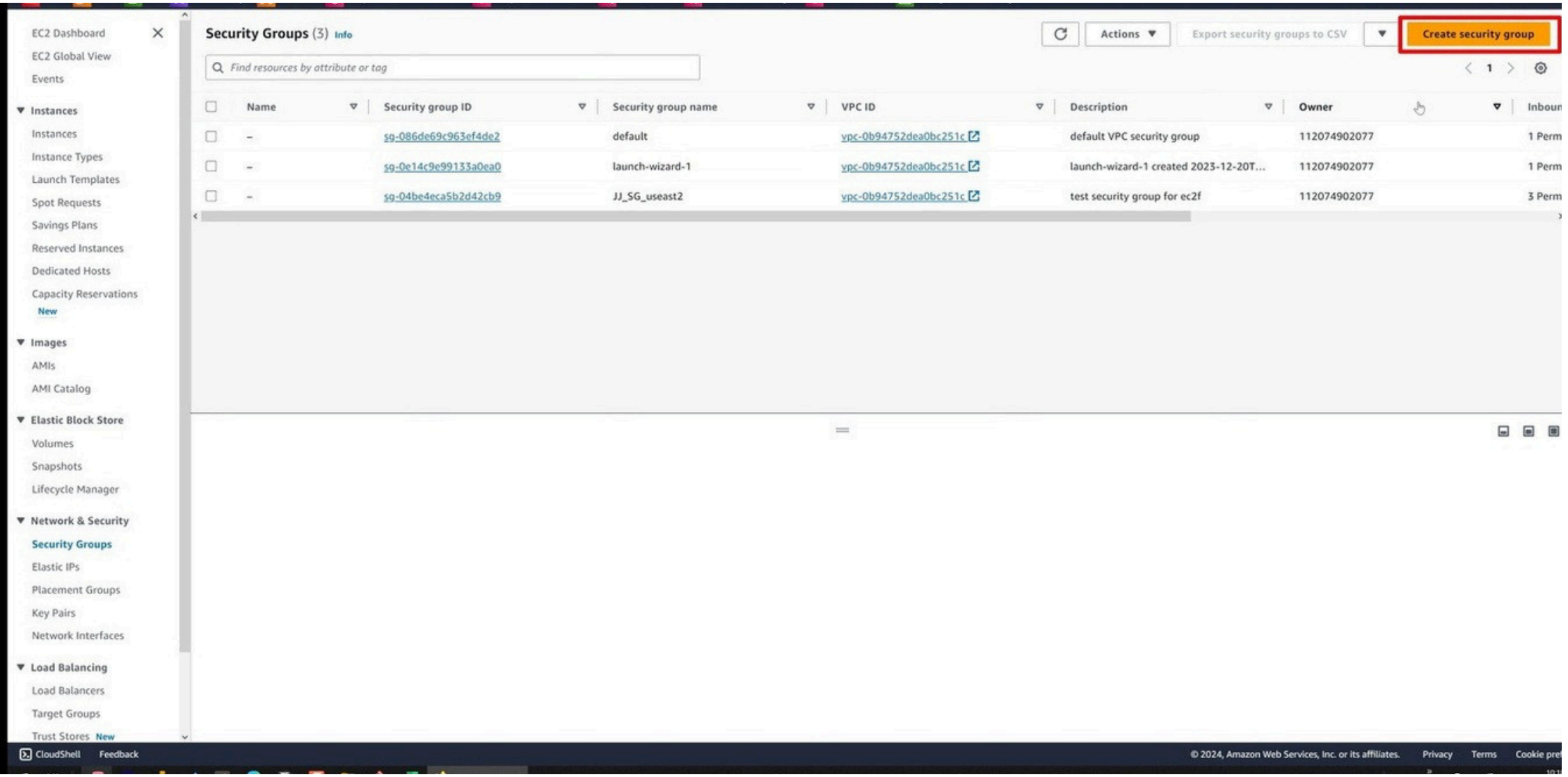
You can add up to 50 more tags.

Cancel | Create key pair


Successfully created key pair
Key pairs (2) Info
Actions
Create
Find Key Pair by attribute or tag
Name
Type
Created
Fingerprint
ID
thinkpad_keypair
rsa
2024/01/08 18:37 GMT-6
f6:ed:d8:5f:ef:1d:2e:b0:9a:e3:7f:9a:d5:86:76:79:18:39:de:c9
key-0cecbbca75f7d2b58
JJ-Demo-TodaysDate
rsa
2024/01/09 09:44 GMT-6
6b:9d:3d:13:e6:f3:41:09:ef:cd:be:9a:4a:6a:8a:cb:31:1a:36:b1
key-05a6dc37a536339a8
Keypair is created
EC2 Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
New
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Load Balancers
Target Groups
Trust Stores New
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms

Now create security group




EC2 Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
New
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Load Balancers
Target Groups
Trust Stores New
Security Groups (3) Info
Actions
Export security groups to CSV
Create security group
Find resources by attribute or tag
1
Name
Security group ID
Security group name
VPC ID
Description
Owner
Inboun
– sg-086de69c963ef4de2 default vpc-0b94752dea0bc251c default VPC security group 112074902077 1 Perm
– sg-0e14c9e99133a0ea0 launch-wizard-1 vpc-0b94752dea0bc251c launch-wizard-1 created 2023-12-20T... 112074902077 1 Perm
– sg-04be4eca5b2d42cb9 JJ_SG_useast2 vpc-0b94752dea0bc251c test security group for ec2f 112074902077 3 Perm
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie pref

EC2 > Security Groups > Create security group

Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name Info

JJ_SG_demo_useast2

Name cannot be edited after creation.

Description Info

Security Group Tutorial Demo

VPC Info

vpc-0b94752dea0bc251c

Inbound rules Info

This security group has no inbound rules.

Add rule

Outbound rules Info

Type Info	Protocol Info	Port range Info	Destination Info		Description - optional Info
All traffic	All	All	Custom	0.0.0.0/0	

Add rule

Inbound rules Info

Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info	
RDP	TCP	3389	My IP		Delete

Add rule

Outbound rules Info

Type Info	Protocol Info	Port range Info	Destination Info	Description - optional Info	
All traffic	All	All	Custom 0.0.0.0/0		Delete

Add rule

⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Tags - *optional*

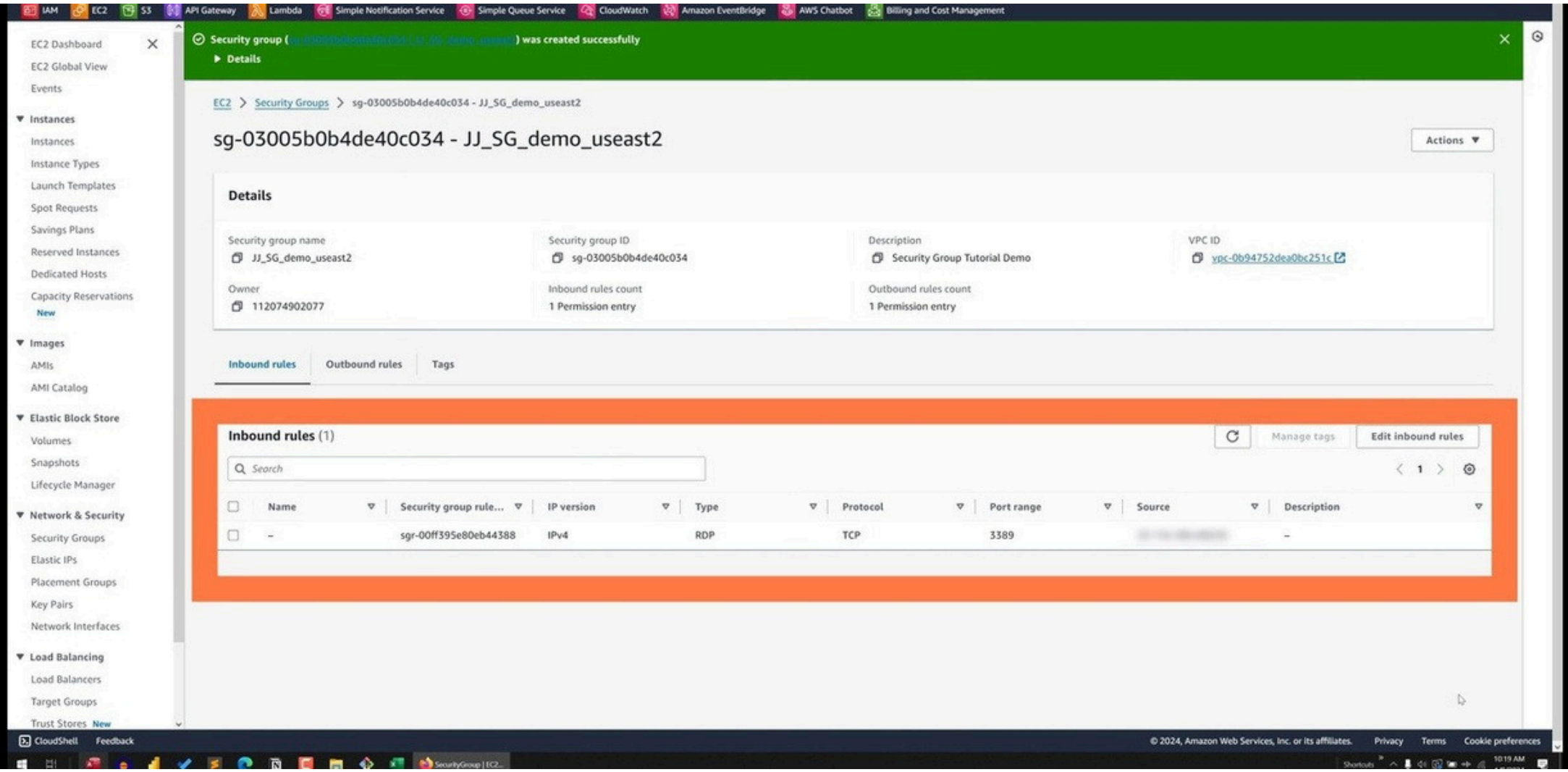
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

No tags associated with the resource.

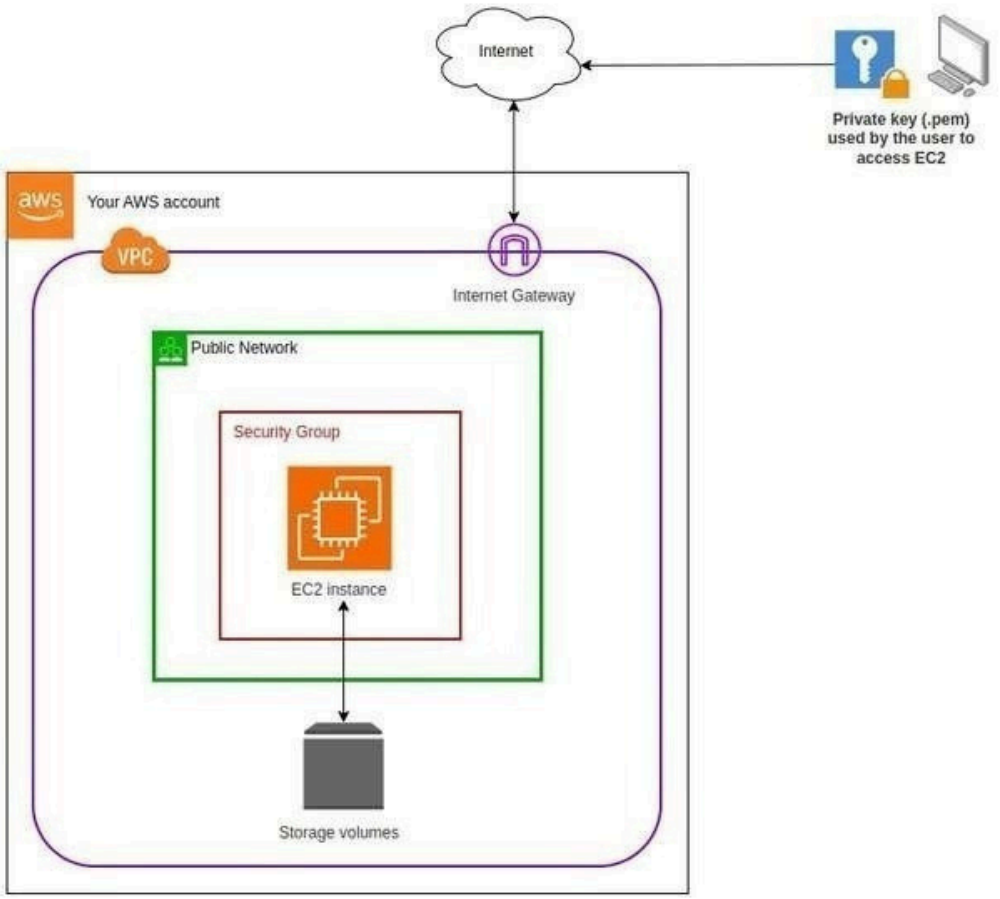
Add new tag

You can add up to 50 more tags

Cancel | Create security group


IAM
EC2
S3
API Gateway
Lambda
Simple Notification Service
Simple Queue Service
CloudWatch
Amazon EventBridge
AWS Chatbot
Billing and Cost Management
EC2 Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
New
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Load Balancers
Target Groups
Trust Stores New
Security group ( ) was created successfully
Details
EC2 > Security Groups > sg-03005b0b4de40c034 - JJ_SG_demo_useast2
sg-03005b0b4de40c034 - JJ_SG_demo_useast2
Actions
Details
Security group name
JJ_SG_demo_useast2
Security group ID
sg-03005b0b4de40c034
Description
Security Group Tutorial Demo
VPC ID
vpc-0b94752dea0bc251c
Owner
112074902077
Inbound rules count
1 Permission entry
Outbound rules count
1 Permission entry
Inbound rules
Outbound rules
Tags
Inbound rules (1)
Manage tags
Edit inbound rules
Search
1
Name
Security group rule...
IP version
Type
Protocol
Port range
Source
Description
-
sgr-00ff395e80eb44388
IPv4
RDP
TCP
3389
-
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Create linux EC2 instance in AWS



So, let's get started with the fundamentals and gradually explore advanced configurations to unlock the full potential of AWS EC2!

Step 1: Sign In to AWS

Log in to your AWS Management Console (https://aws.amazon.com/console/) using your AWS account credentials.

From the AWS Management Console, locate the "Services" dropdown and select "EC2" under the "Compute" section.

Step 2: Launch an Instance

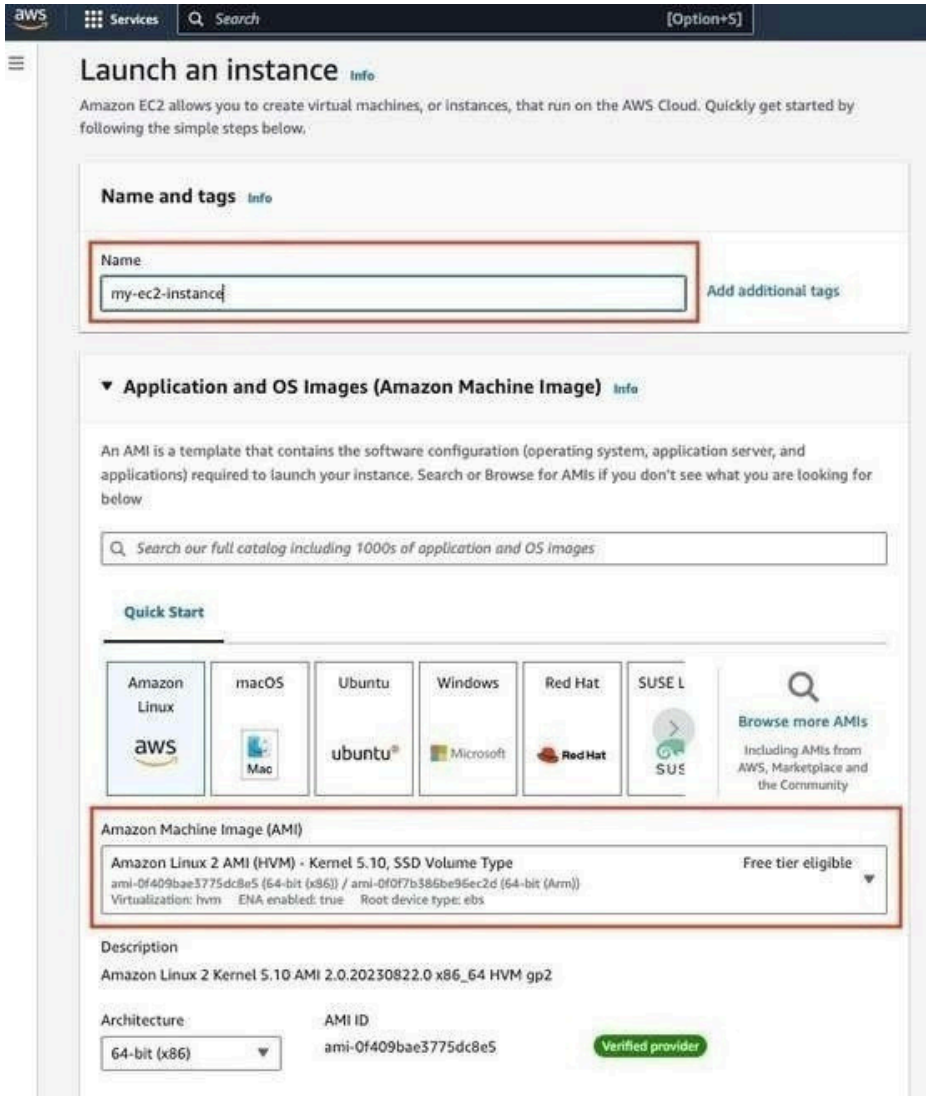
In the EC2 Dashboard, click the "Instances" link in the left navigation pane, then click the orange "Launch Instance" button.

Step 3: Choose an Amazon Machine Image (AMI)

Give a name to your instance.

- Choose an AMI that matches your requirements.
-

AMIs are pre-configured templates that include an operating system and other software. Most of the world's servers use Linux because it's open-source, reliable, and efficient. It's commonly used for web servers, databases, and more. For this guide, we'll choose a Linux AMI.

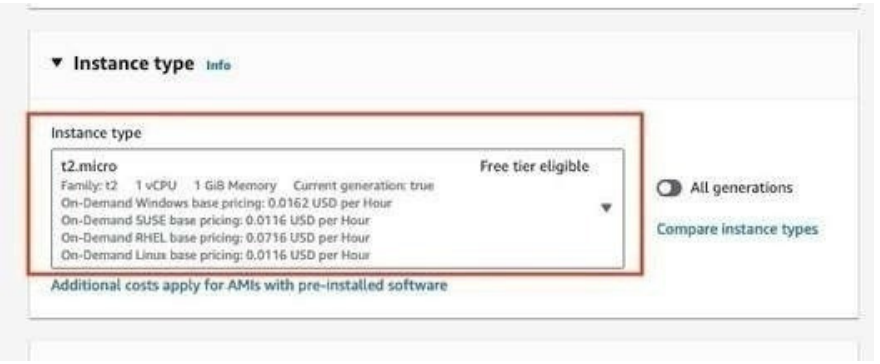


Step 4: Choose an Instance Type

Give a name to your instance.

- Choose an AMI that matches your requirements.
-

AMIs are pre-configured templates that include an operating system and other software. Most of the world's servers use Linux because it's open-source, reliable, and efficient. It's commonly used for web servers, databases, and more. For this guide, we'll choose a Linux AMI.



Step 5: Create or Select a Key Pair

You need a key pair to securely connect to your instance using SSH. Create a new key pair or select an existing one. Make sure to safely store the private key file, as it's the key to accessing your server. If you lose it, you won't be able to access the instance. You can add other key pairs later if needed.

Select create a new key pair and give a name to your *.pem* key.

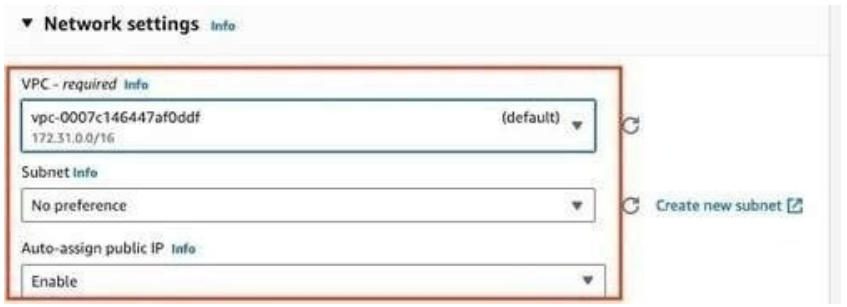
- Click on the “Create key pair” orange button.
-

A *.pem* key will get downloaded into your local computer.

Step 6: Configure Network Details

Here, you can choose the network settings (like the VPC and subnet), and more. For now, we'll use

the default VPC and subnet. Think of VPC as a private network, and we'll dive deeper into it later.

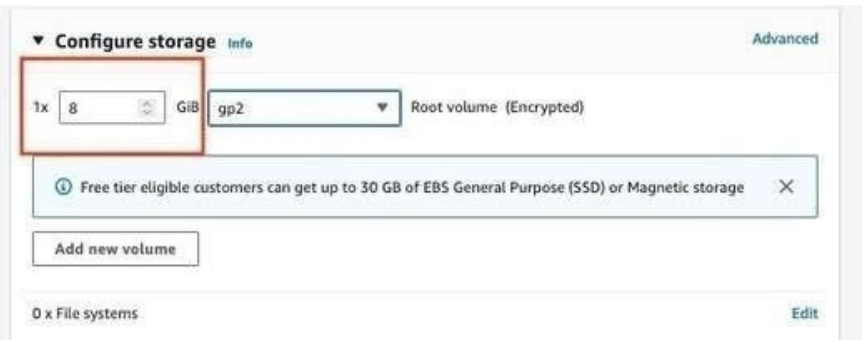


Step 7: Configure Security Group

Security groups act as virtual firewalls for your instance. Define inbound and outbound rules to control network traffic. For now, use the default security group. We can delve deeper into it later on how we use security groups for security.

Step 8: Add Storage

Configure the amount and type of storage for your instance. You can add additional storage volumes if needed.



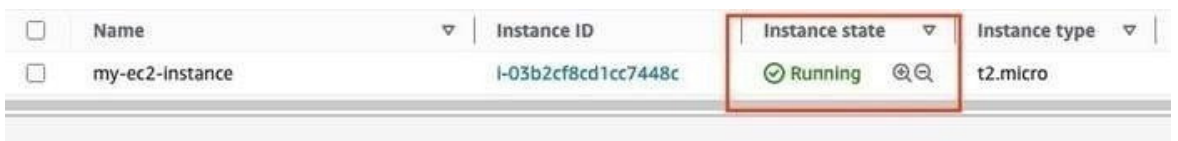
Step 9: Review and Launch

Review your instance configuration settings. If everything looks good, click the "Launch" button.

Step 10: Launch Status

As you delve deeper into the EC2 service, you will gradually grasp and gain insights into additional settings, expanding your understanding and knowledge over time.

- Once you click "Launch," your instance will start launching. You'll see its status change to "running" in the EC2 Dashboard.



Step 11: Access Your Instance using "ssh"

Once your instance is running, go to the EC2 Dashboard, select your instance, and click the "Connect" button. You'll find instructions for both Linux and Windows connections below:

Connecting from Mac and Linux:

- Open a terminal on your local machine.
- Navigate to the directory where you saved your private key file (.pem).
- Use the "*chmod*" command to set the appropriate permissions on the key file:
- *chmod 400 <your-key-file>.pem*
- Copy the SSH command from the EC2 instance connect page.
- Paste the command into your terminal and press Enter
-

Connecting from Windows:

- Download and install an SSH client such as PuTTY.
- Convert your .pem key file to a .ppk key file using PuTTYgen.
- Open PuTTY and enter the public IP address of your instance in the "Host Name" field.
- Load your .ppk key file in the "Connection > SSH > Auth" settings.
- Click "Open" to start the SSH session.
-

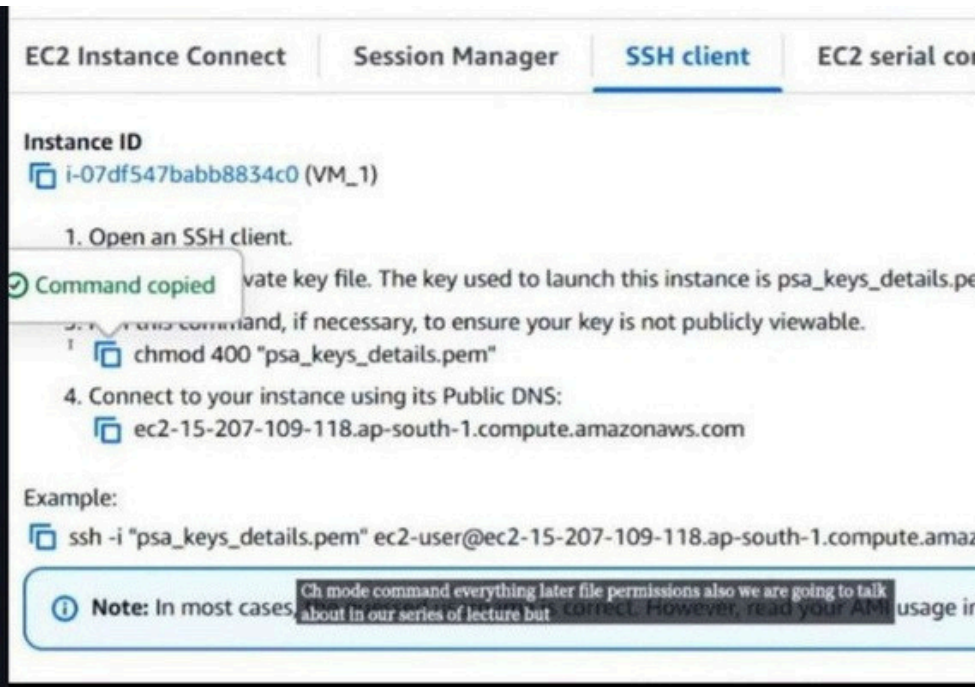
Step 13: Access Your Instance using "AWS Session Manager"

You can access now the instance through the AWS console using Session Manager.

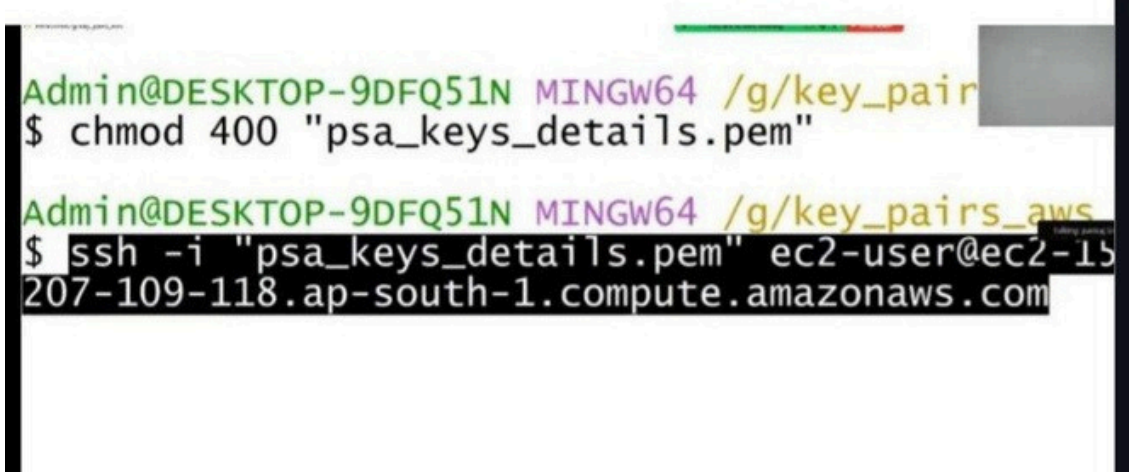
This provides a browser-based session similar to EC2 Instance Connect, eliminating the need to open port 22.

Step 1:Right click keypair file and open gitbash

Step 2 copy 2 commands
and paste
in gitbash as shown step 3



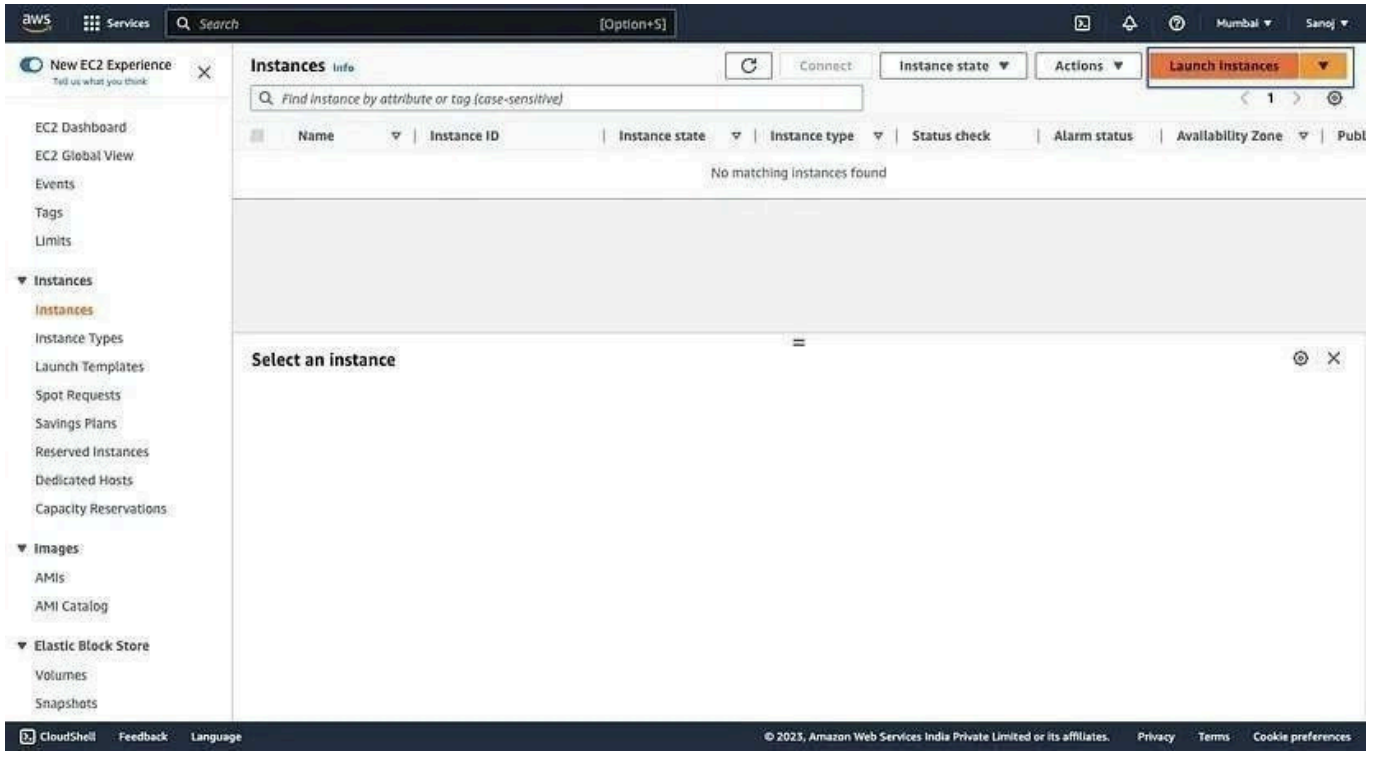
Step 3



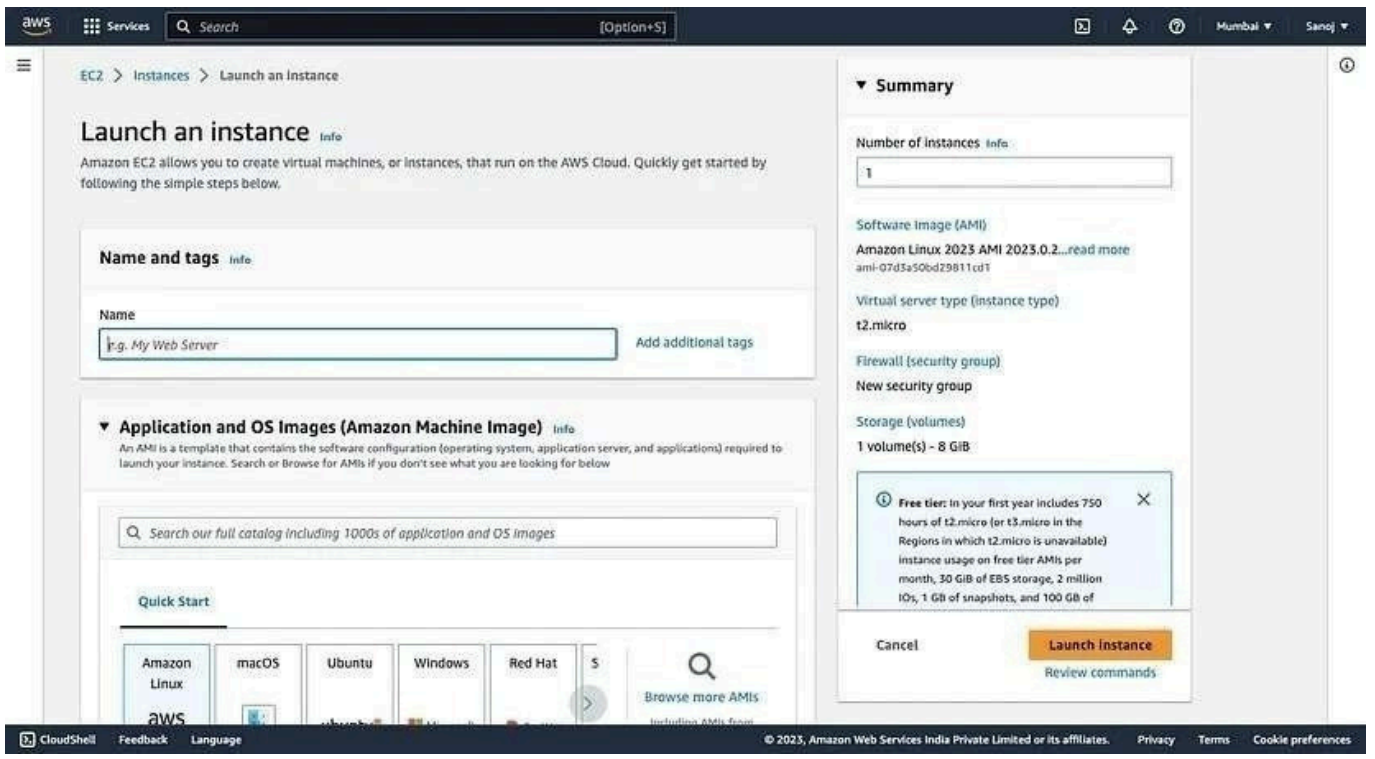
----> Done

How to create a Windows Ec2 instance in AWS?

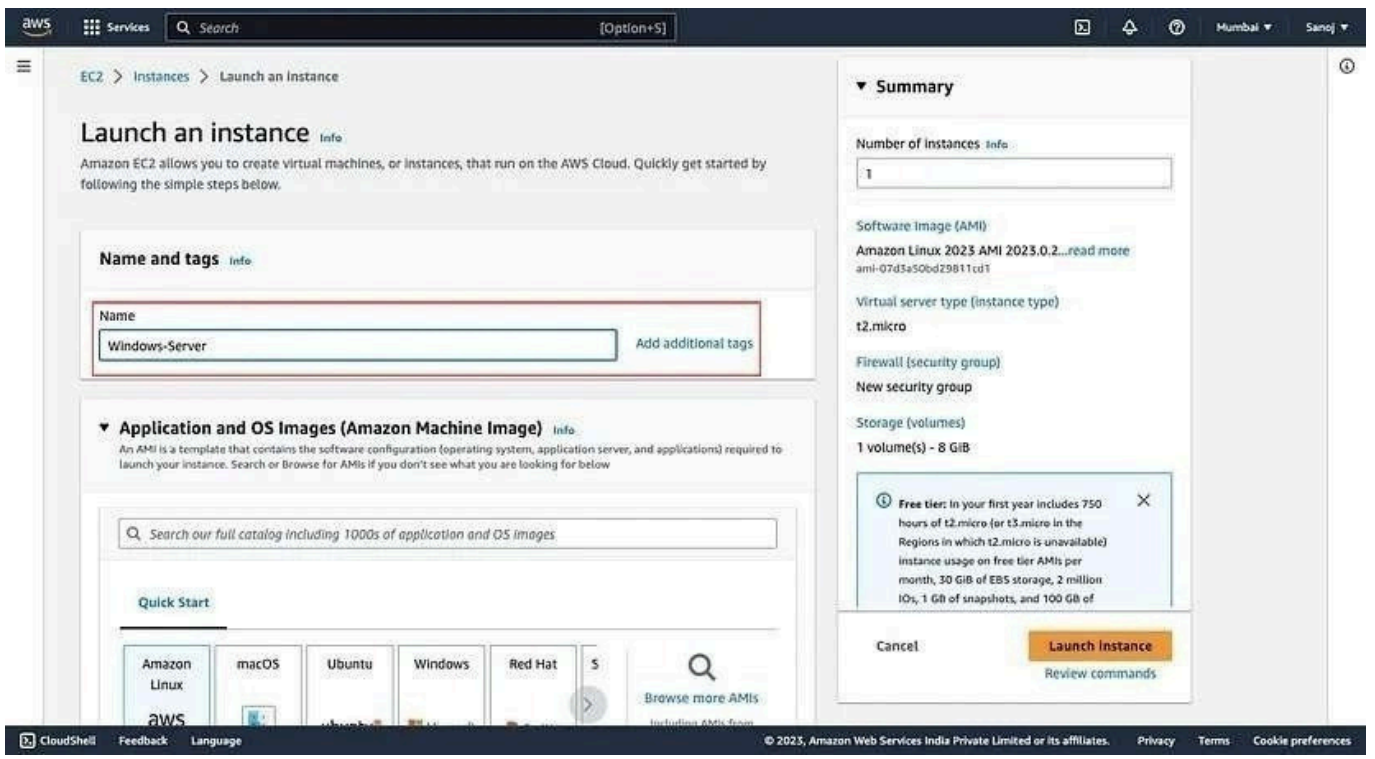
Step 1: Sign in to the AWS Management Console and click on the “Launch Instances “



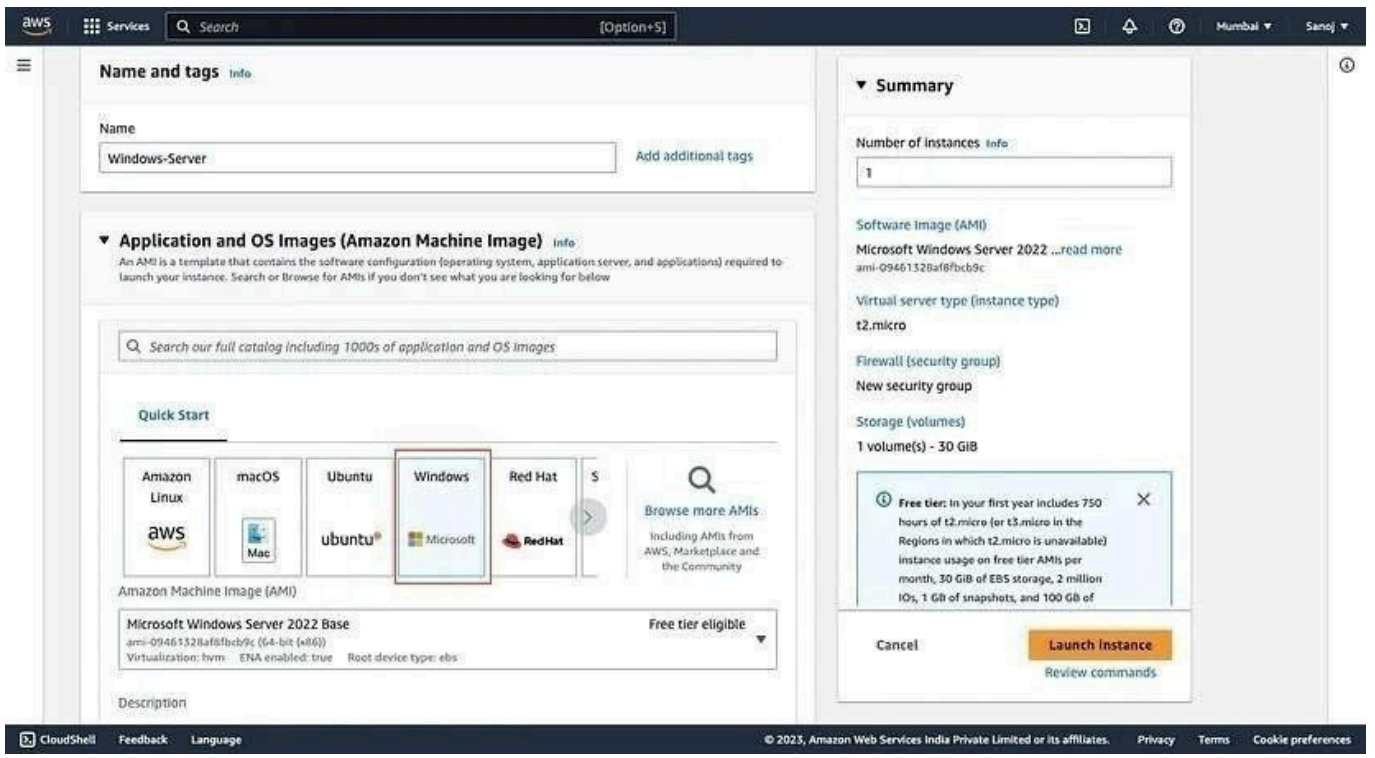
After clicking on the “Launch Instances” the following interface will appear.



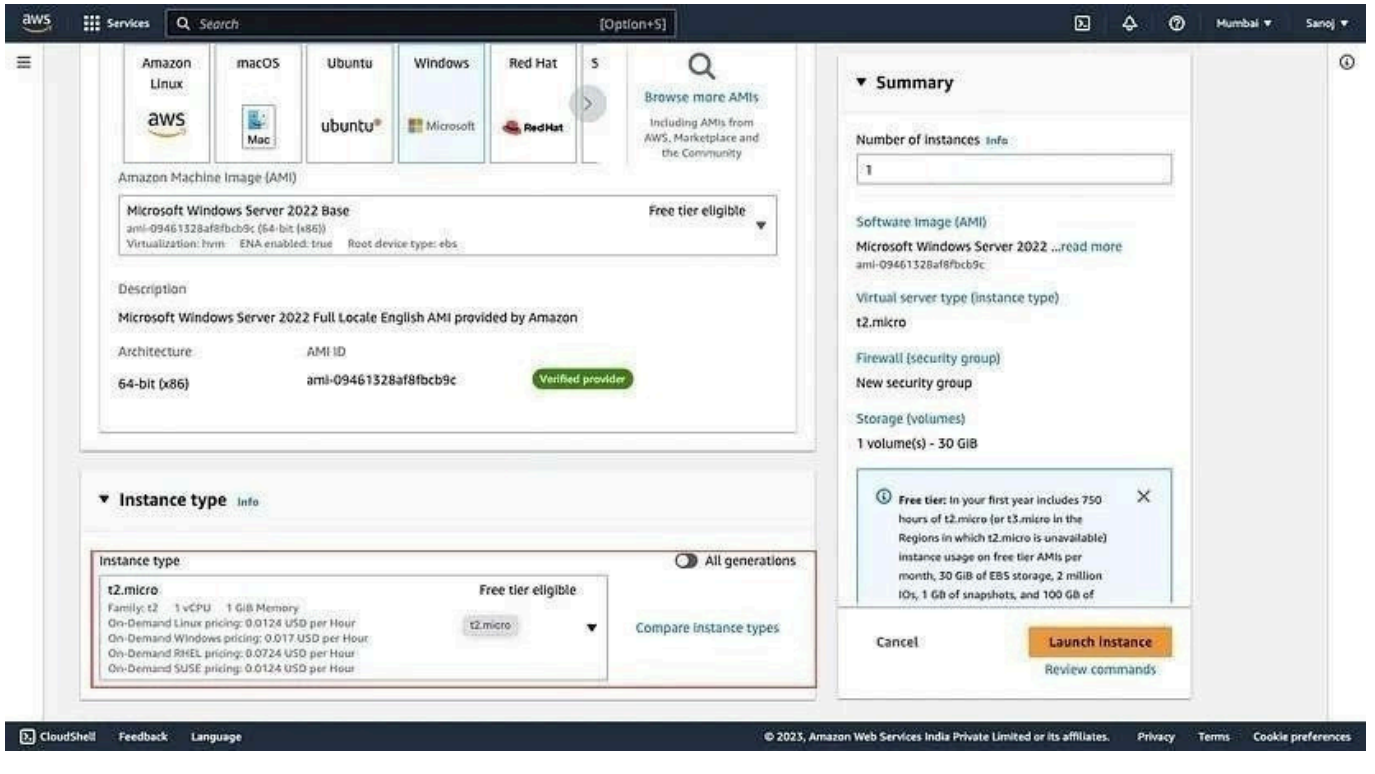
Step 2: Now give the Instance name, I gave the name “Windows-Server”



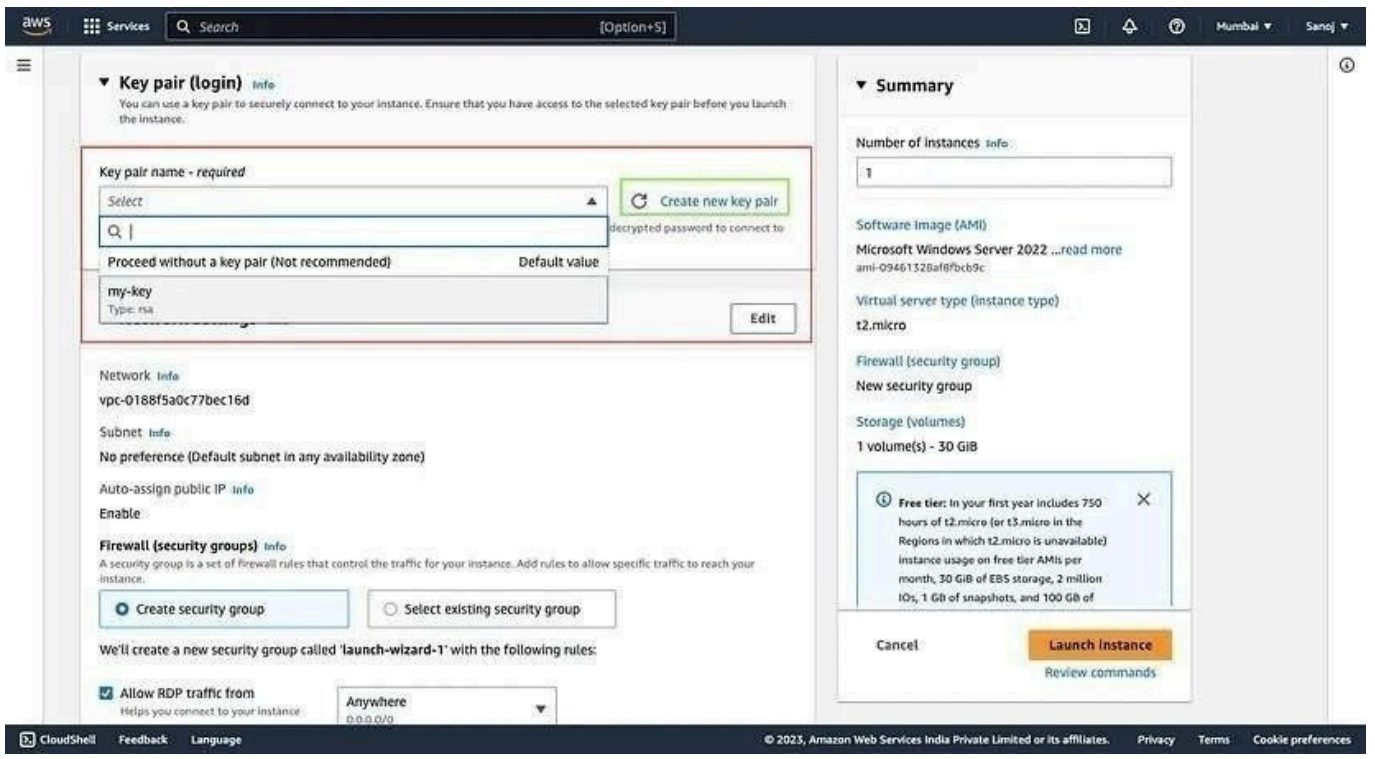
Step 3: Select the AMI (Amazon Machine Image) or OS.



Step 4: Select the Instance Type or Hardware Type, I selected t2.micro.

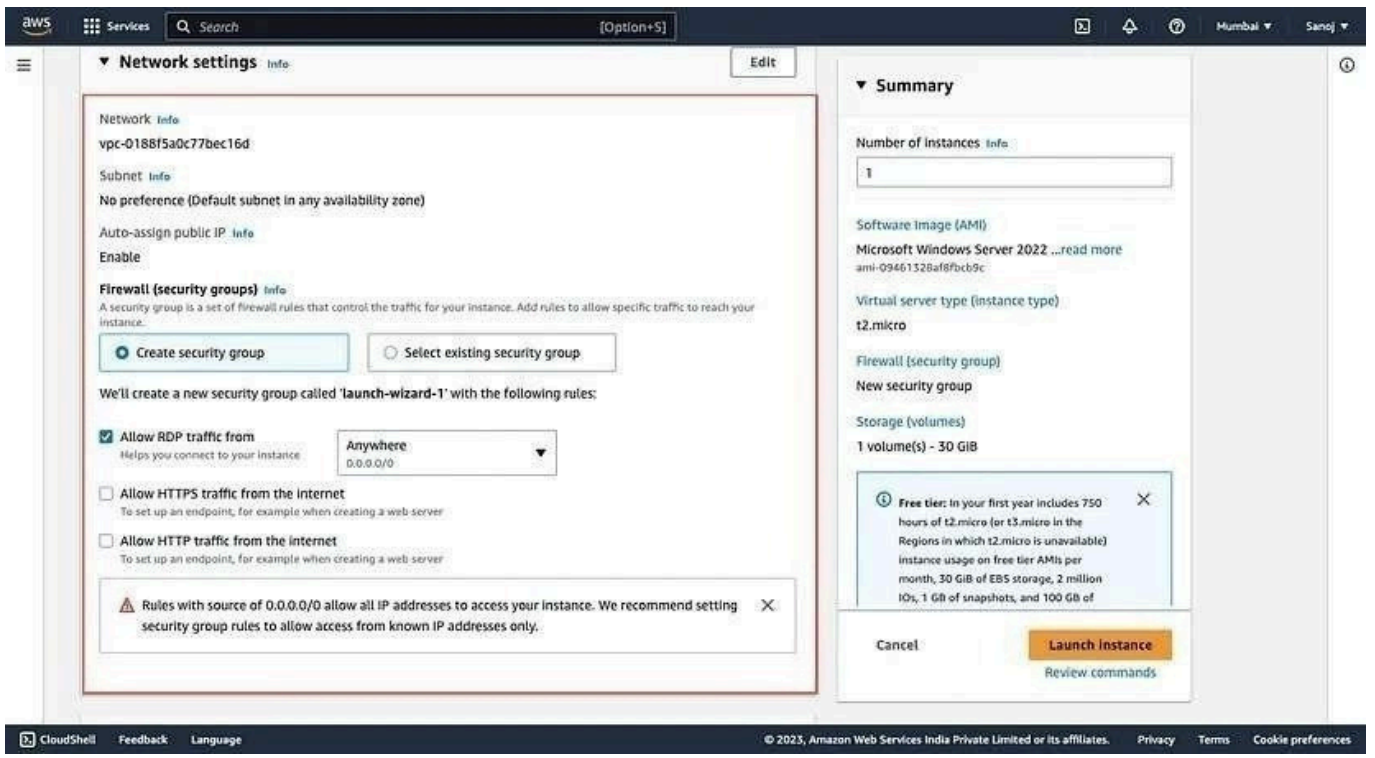


Step 5: Select your “Key pair” by clicking on the Drop down menu, if you don’t have , create new “key pair”.

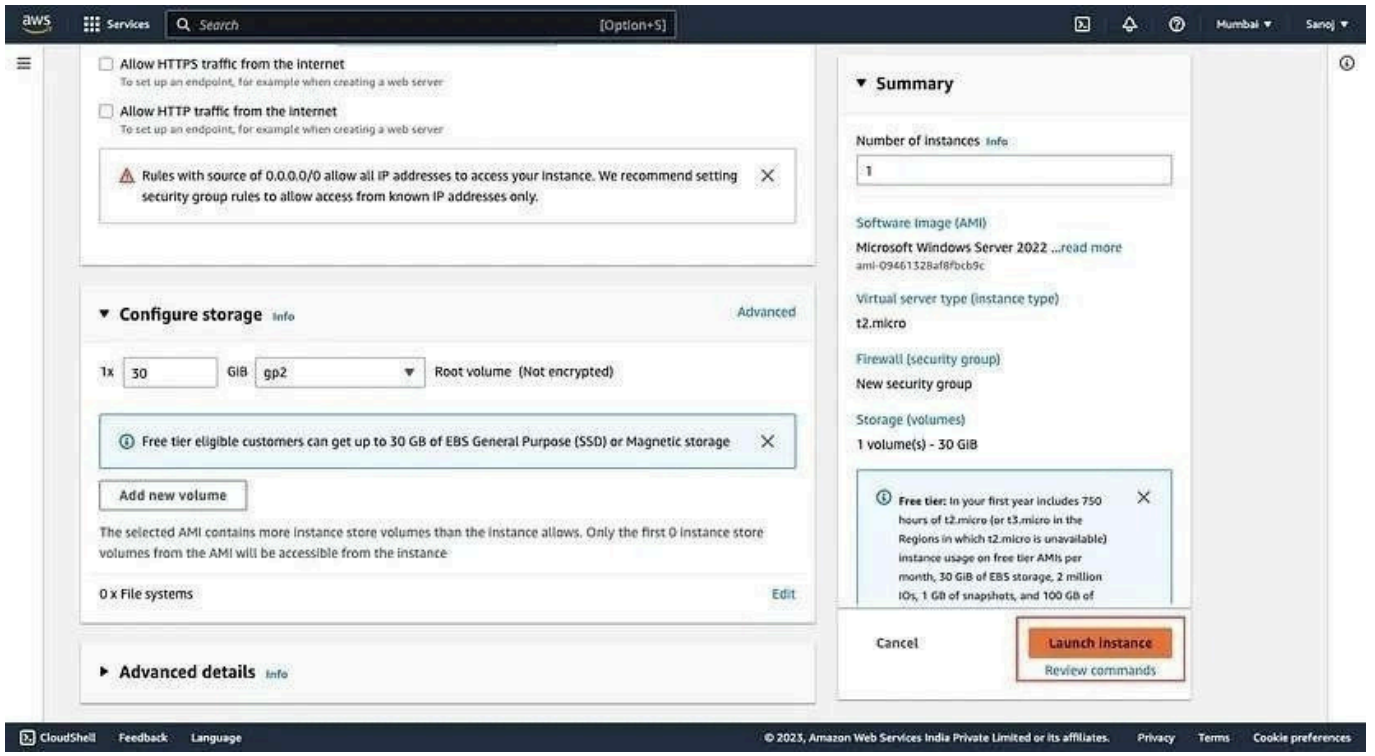


Step 6: Select your security group if you have already if you don’t have a, create a new security group.

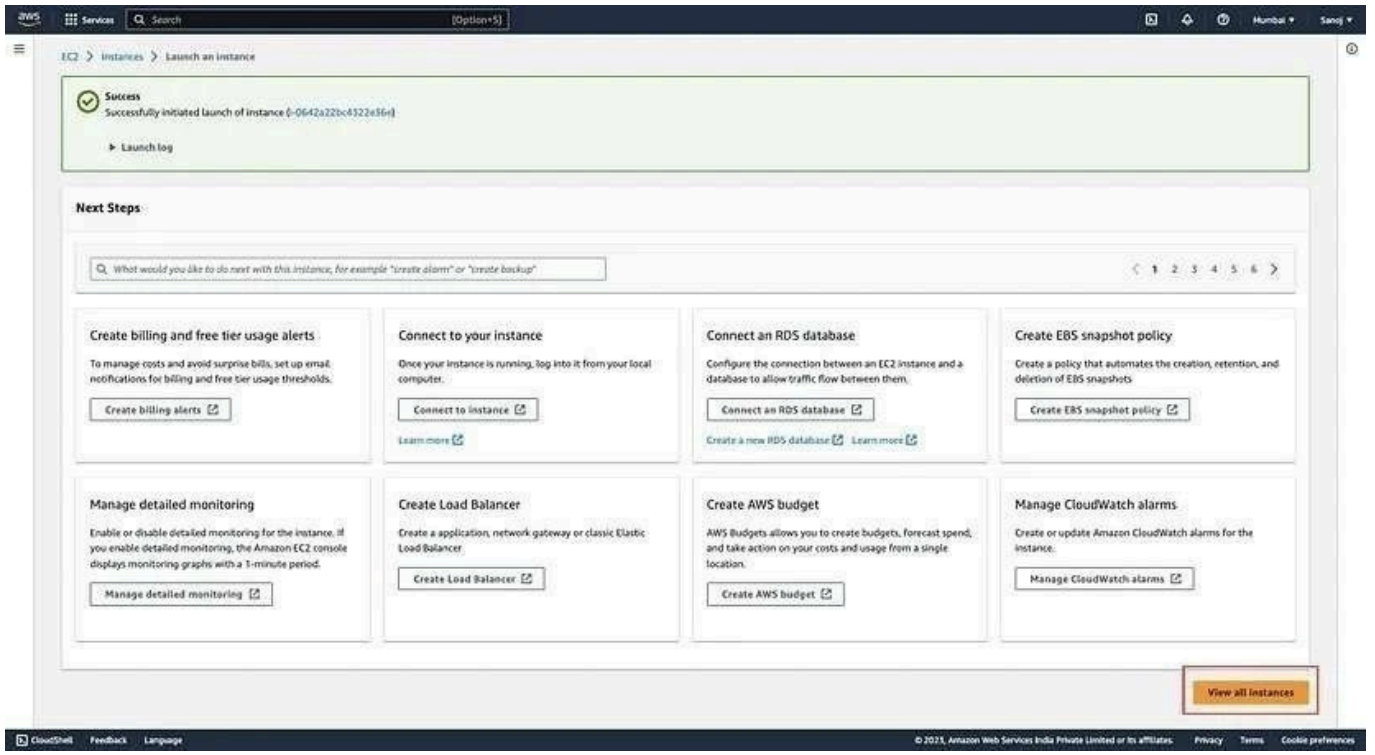
Note: If you keep it as it is, AWS will create a new security group for you.

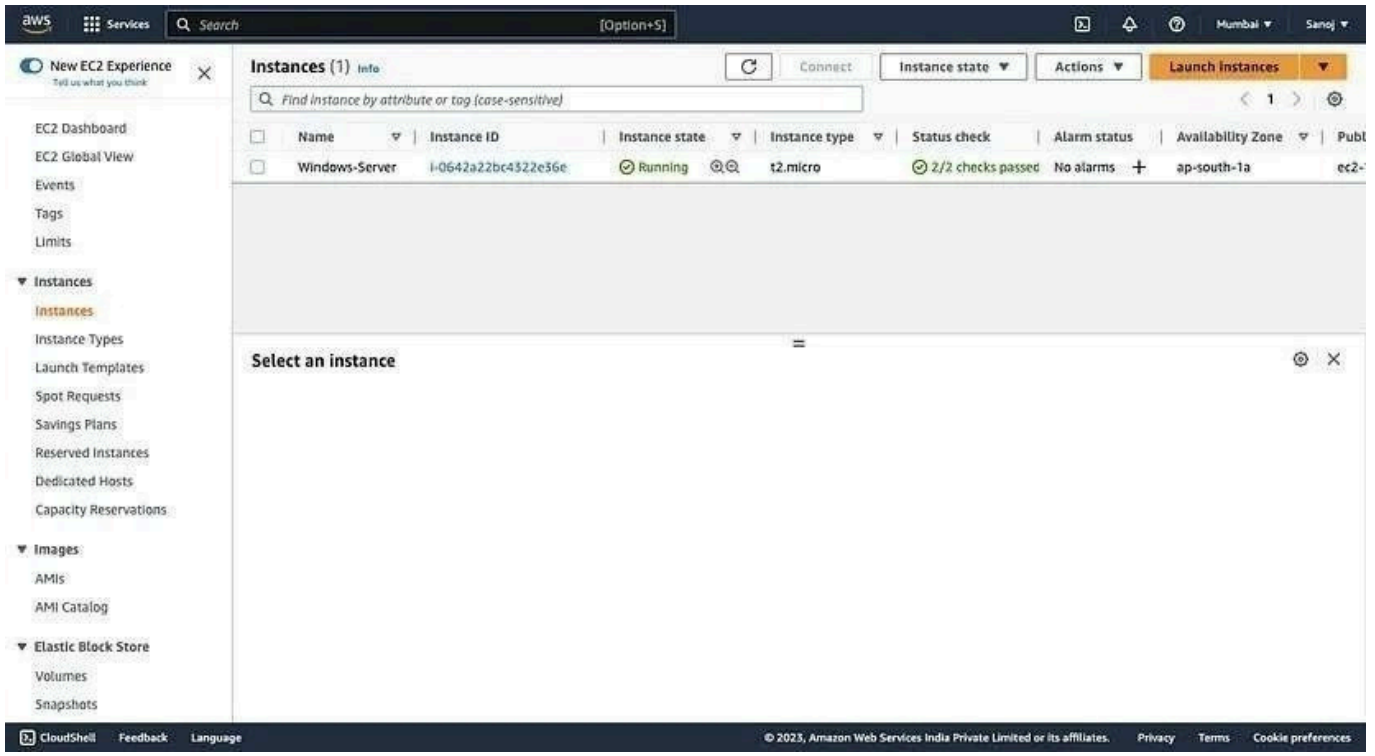


Step 7: Click on the “Launch Instance”.



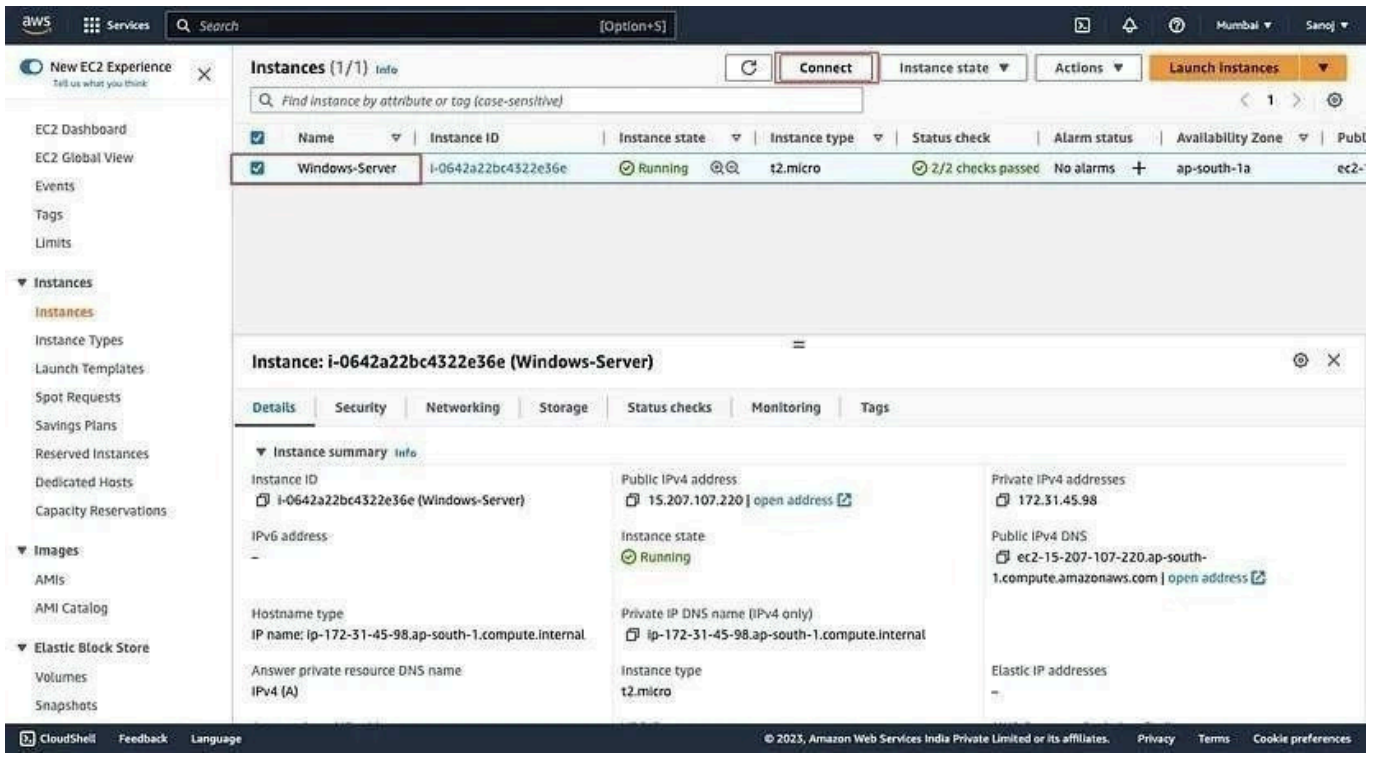
After clicking on the “launch Instance” the following interface will appear, just click on the “View all instances”





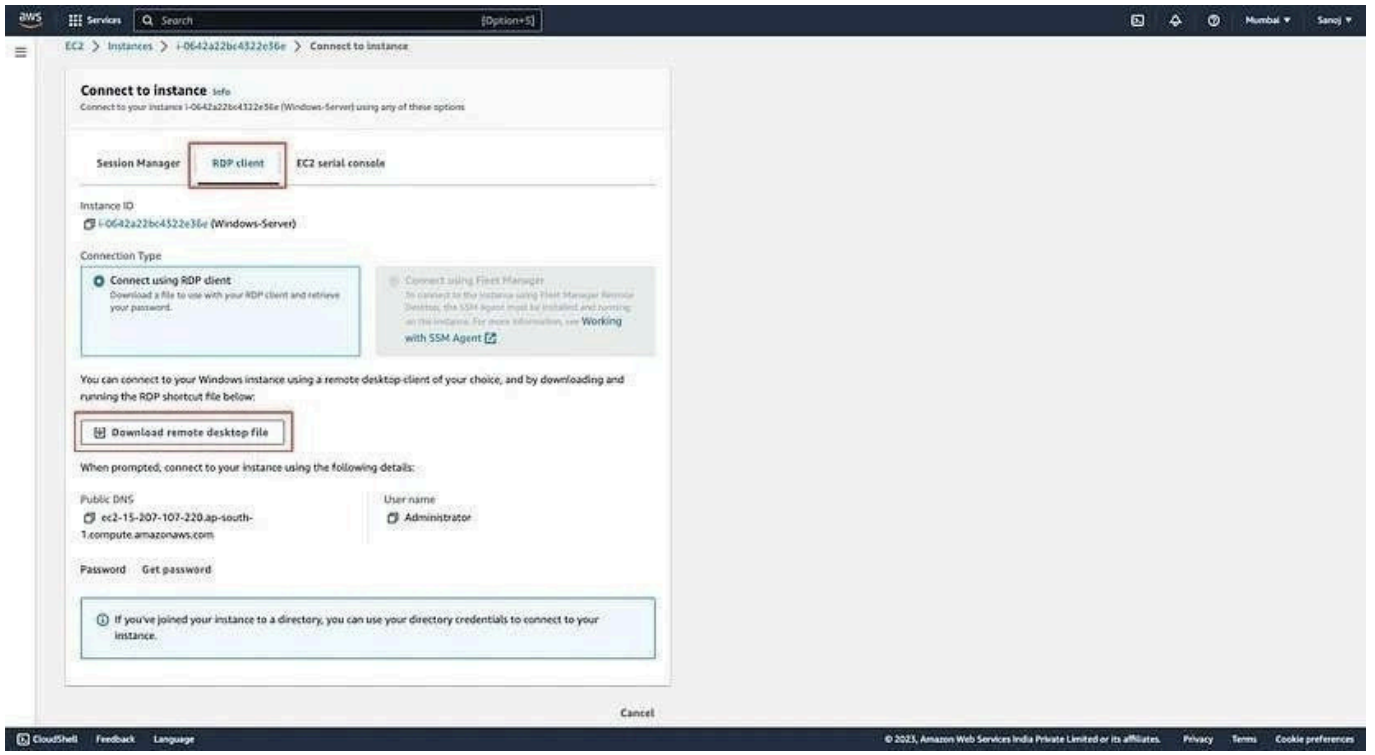
Now it's time to connect our Ec2 instance that we have recently launched.

Step 8: Select your Windows Instance and Click on the “Connect”.

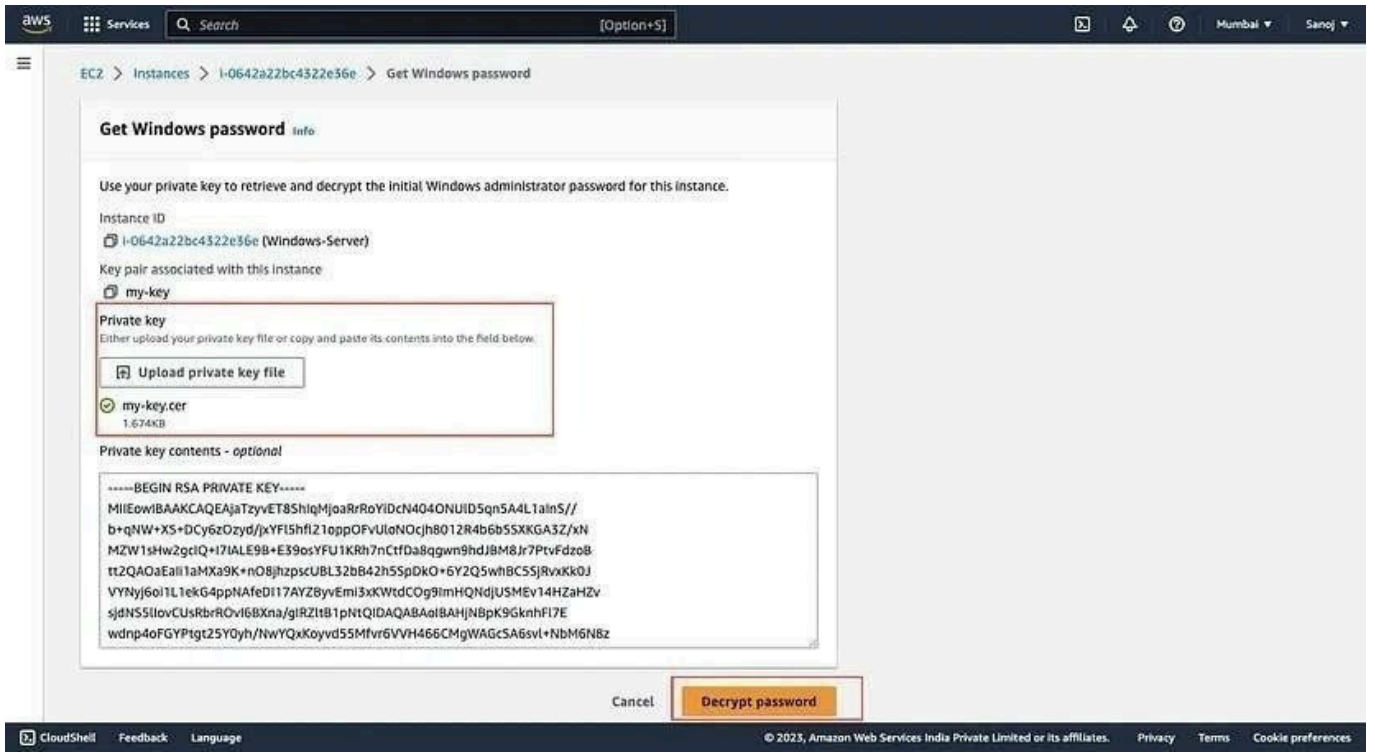


Step 9: After Clicking on “Connect” the following interface will appear. just select the “RDP Client” Tab and download the “Remote desktop file”.

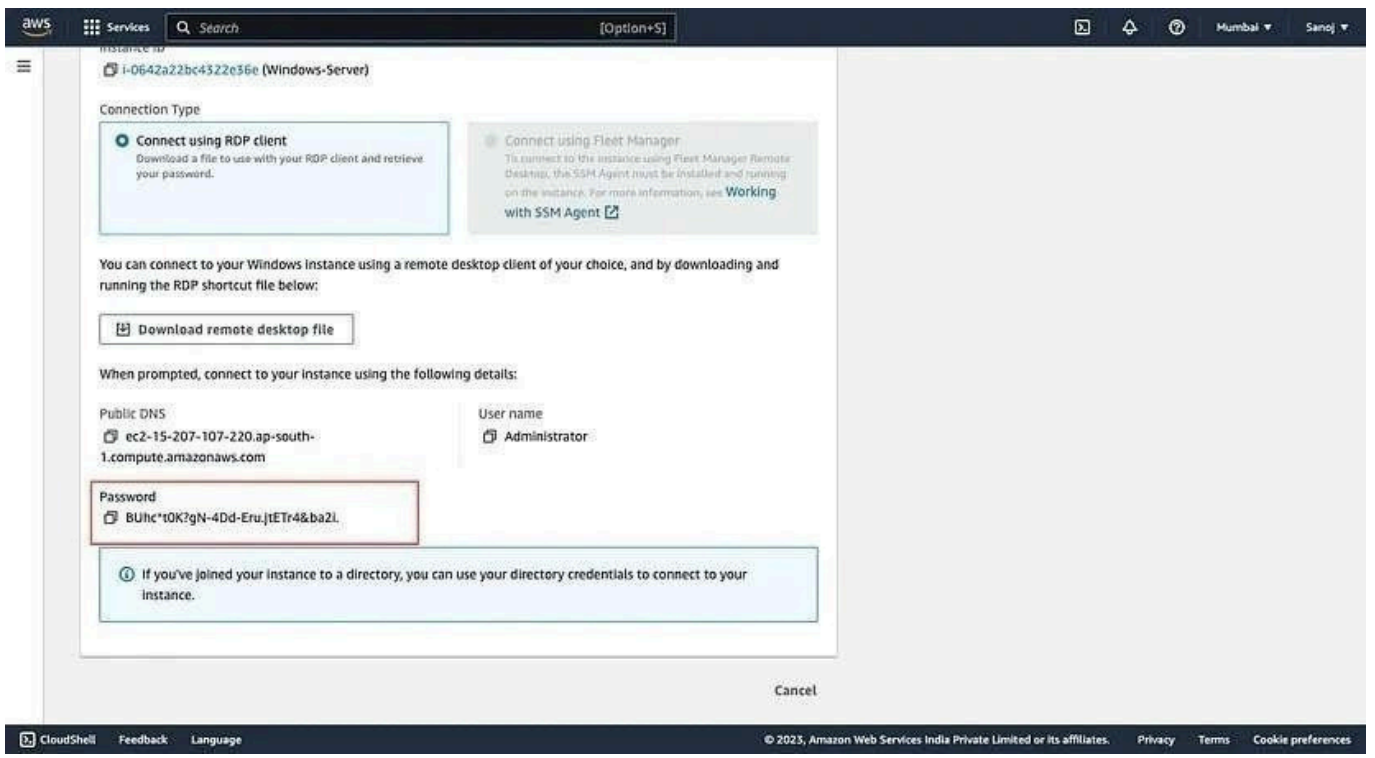
Note: In the case of Windows only, Linux and mac have different mechanisms to connect RDP Connection.



Step 10: Now click on the “Get password”, select your “key pair” and decrypt the password.



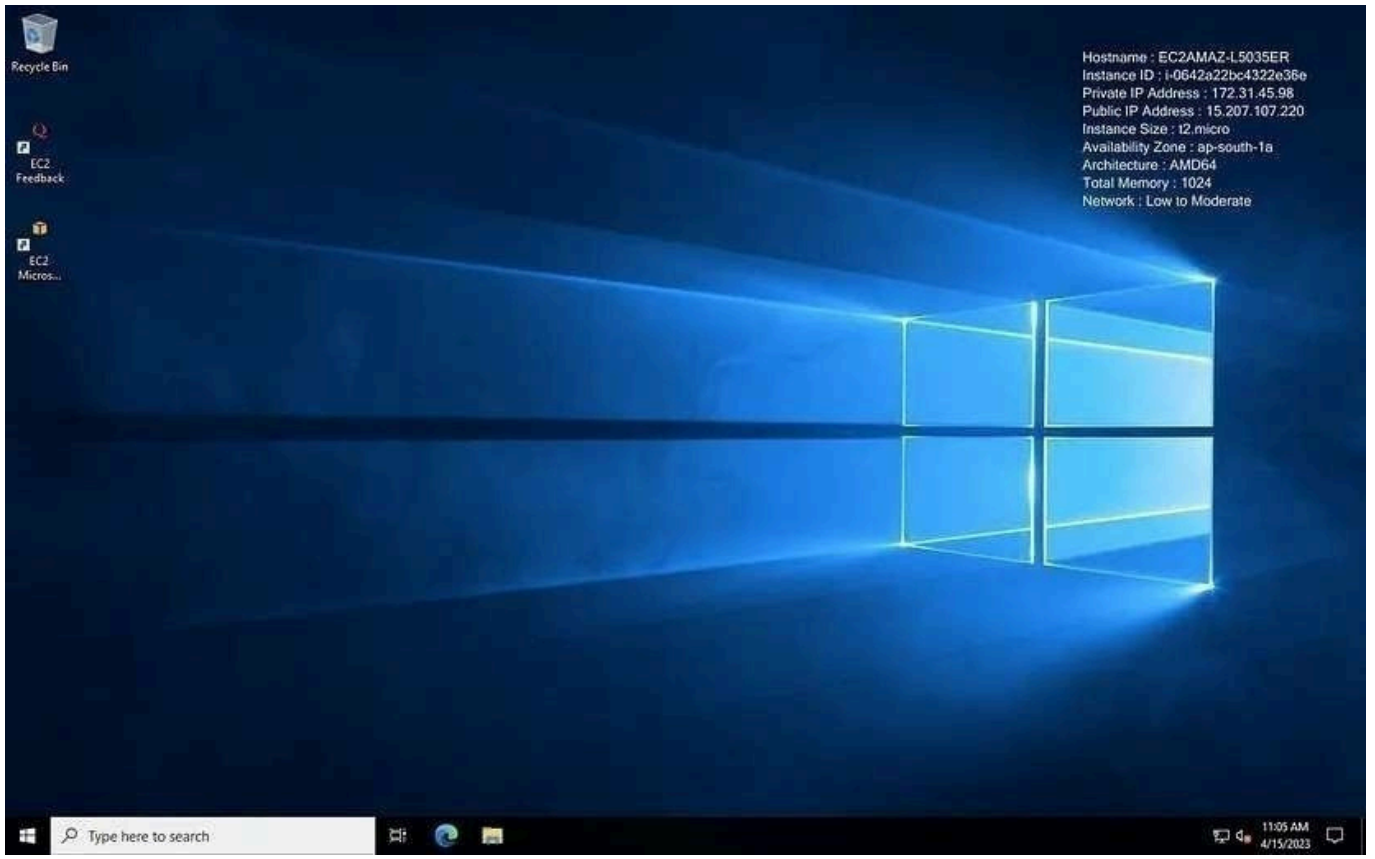
Step 11: After clicking on the “Decrypt password” the following interface will appear to copy the password and use that password to connect your Windows EC2 instance



Step 12: Open your Downloaded file “RDP file” and put, the password that you decrypted.

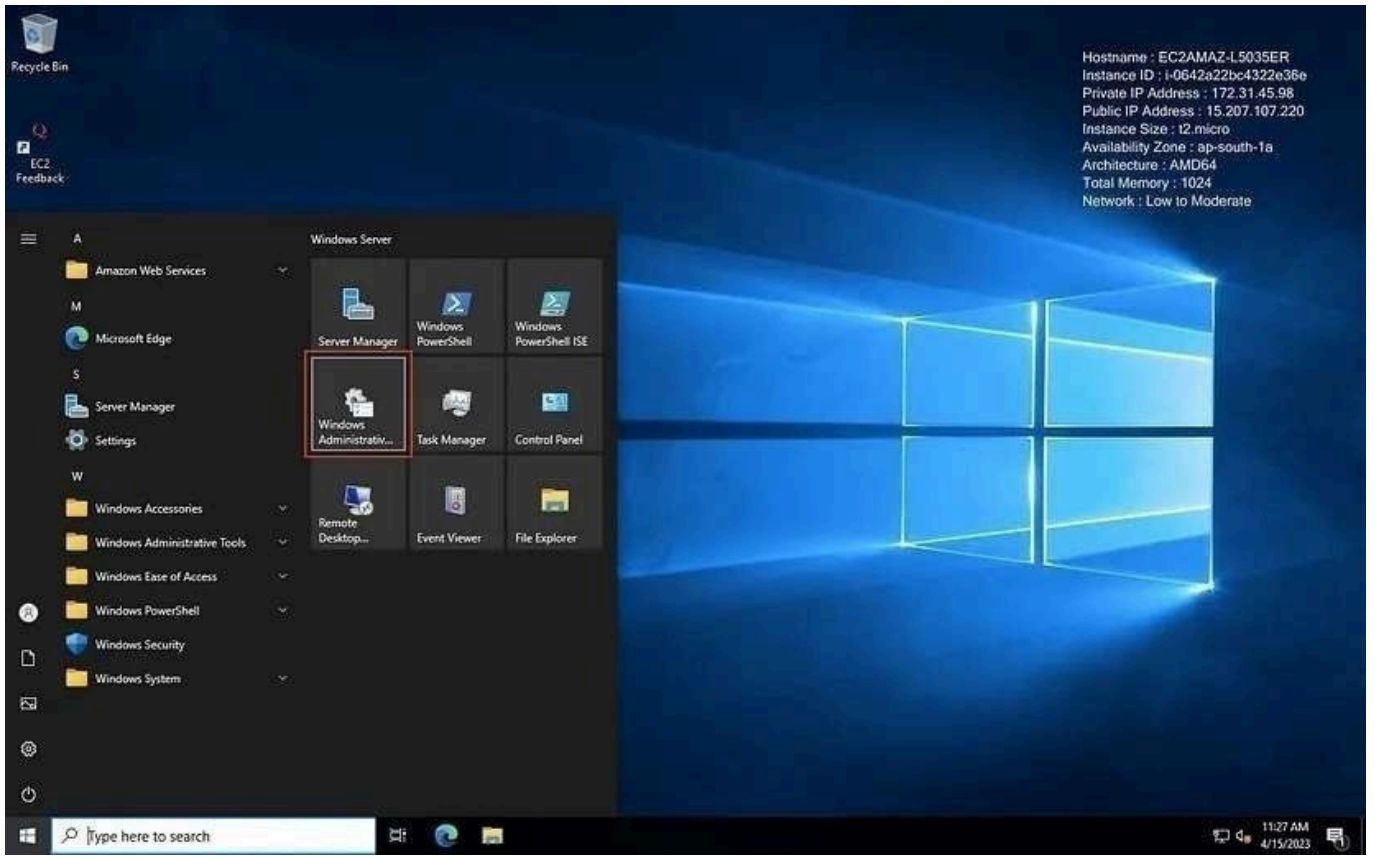
The default user name will be “Administrator” and the password is that you have decrypted and shown in the above screenshot.

> Note: Every time whenever you want to connect after closing the session you need to regenerate the password or decrypt the using “Key pair”

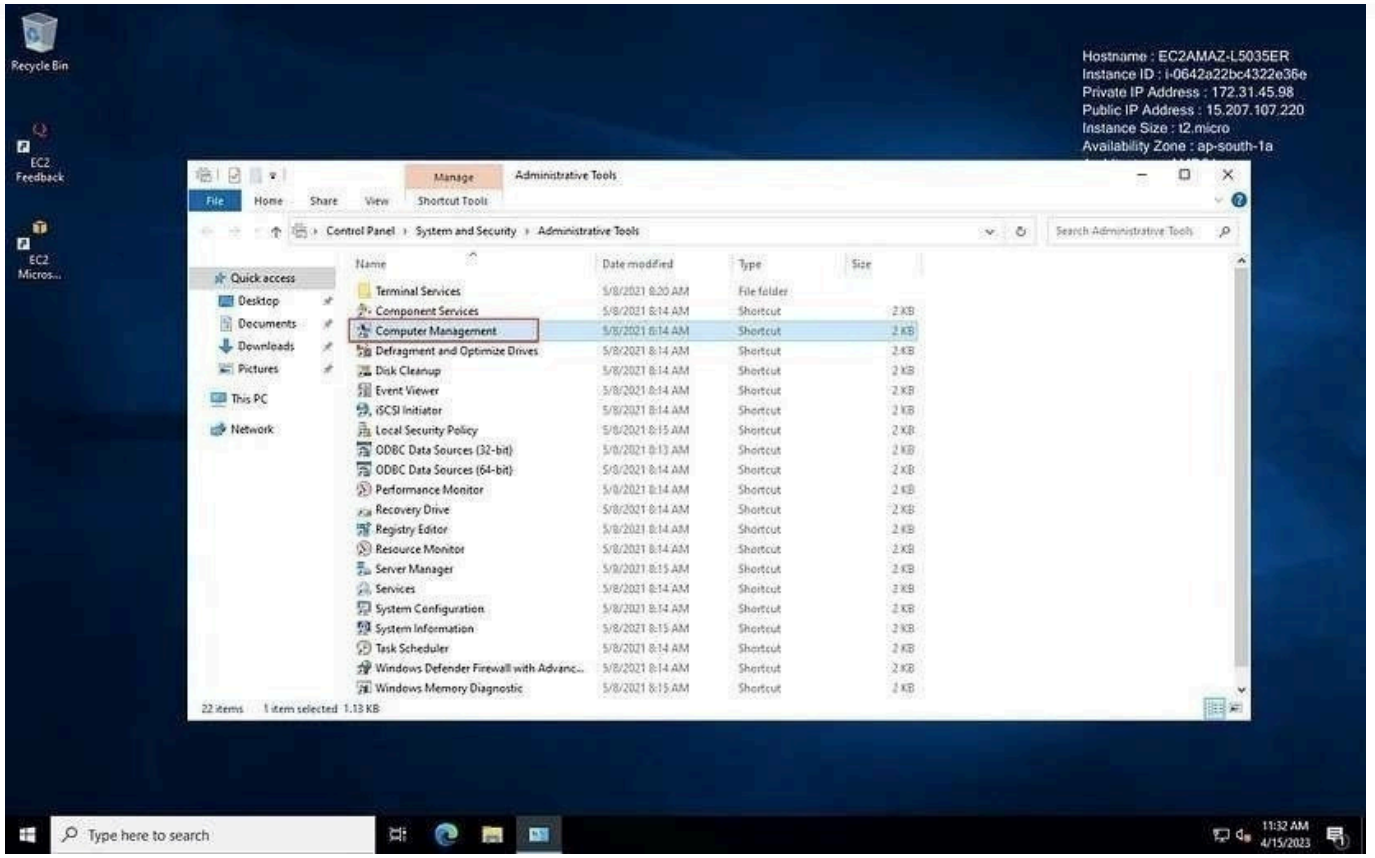


> Now we don’t want to decrypt the password every time to connect the Windows EC2 instance for that just change the Windows Password.
>
> Now the question is how we can do that.

Step 13: Press Windows Button or Click on the “Windows Icon”, then click on “Windows Administrative tool”.

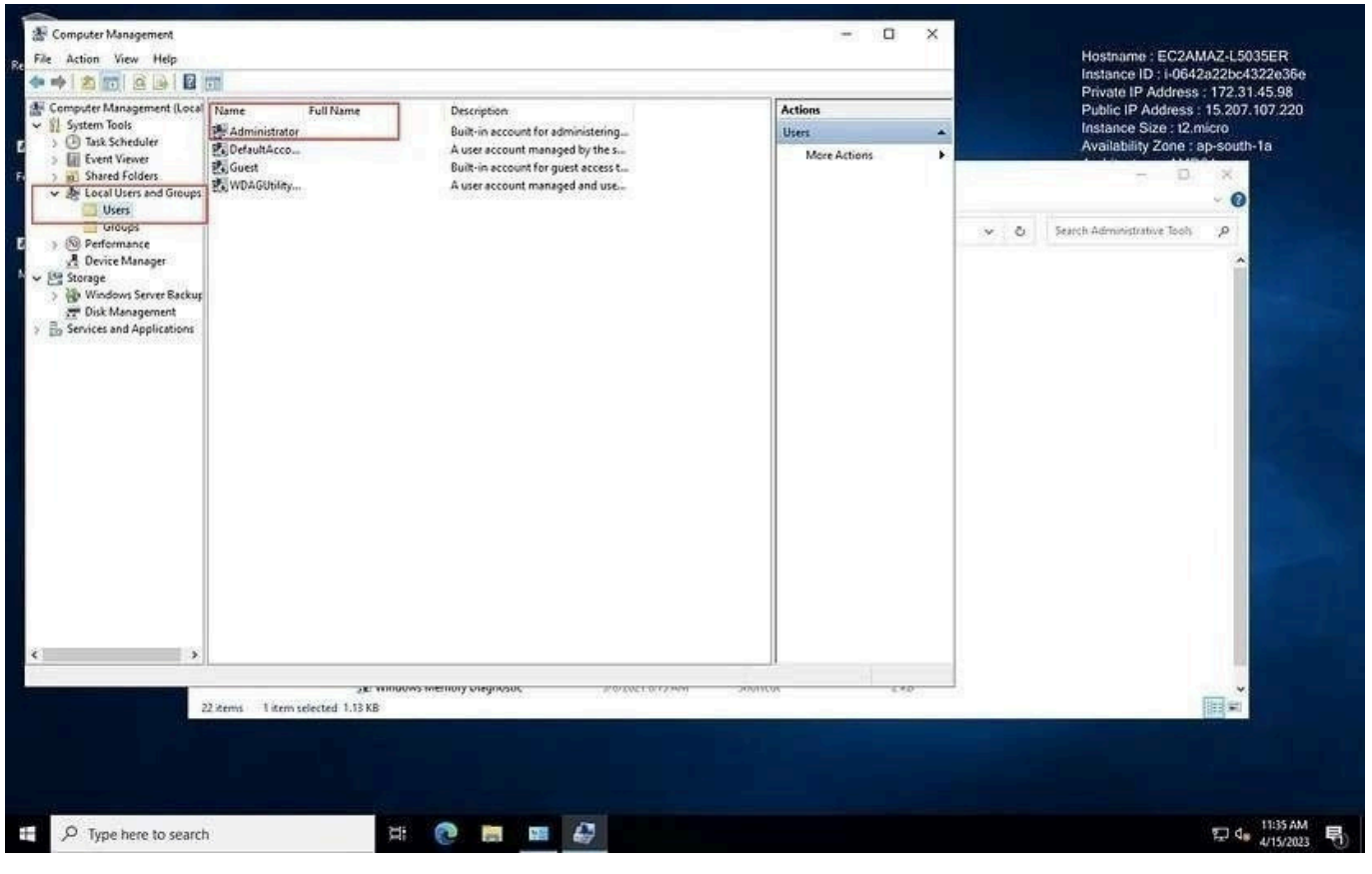


After clicking the “Windows Administrative tool” the following interface will appear, just click on “Computer Management”



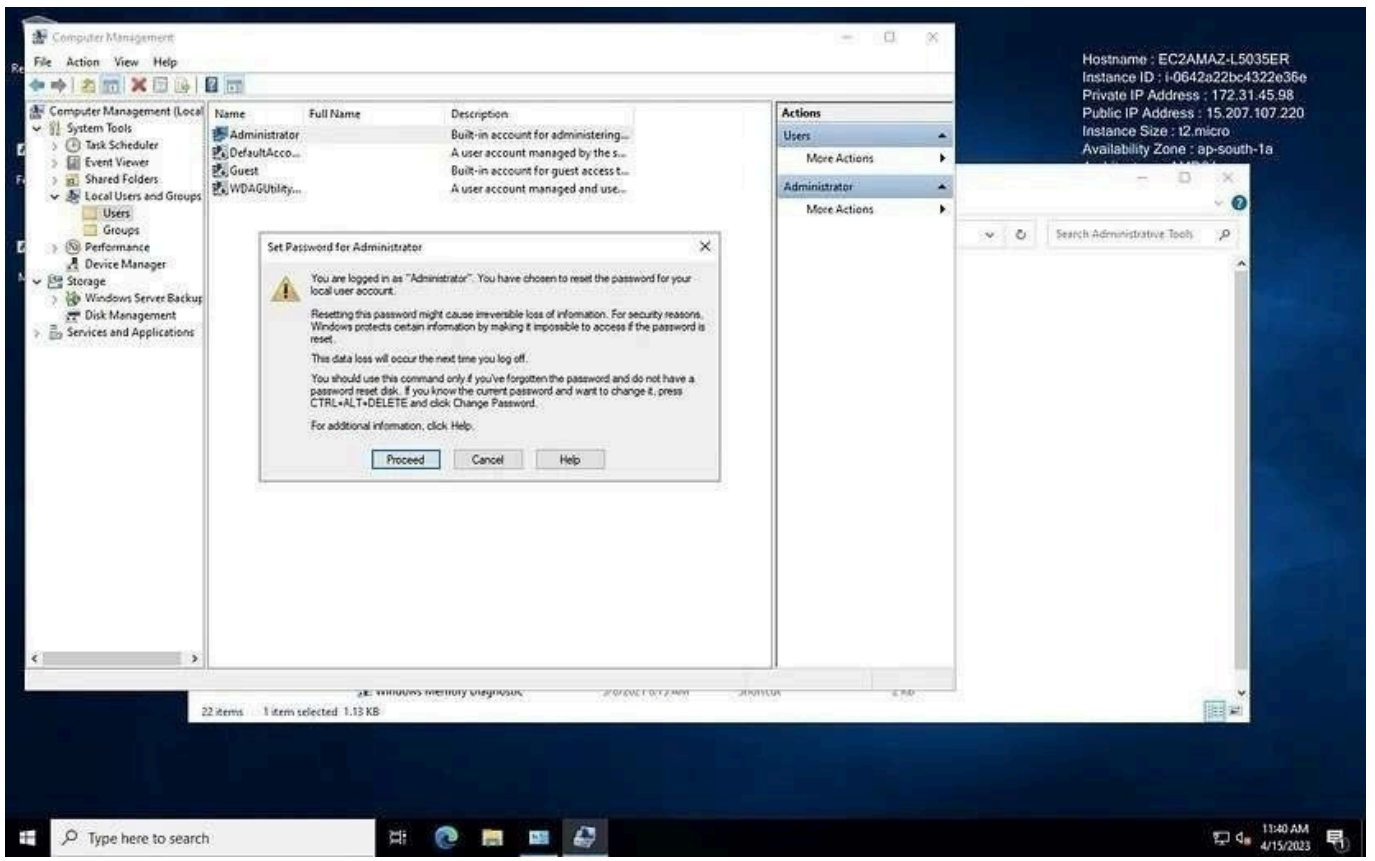
Step 14: After clicking on “Computer Management” the following interface will appear. Just click on “Local Users and Groups” then click on “Users”.

Then right-click on the “Administrator” and set your password

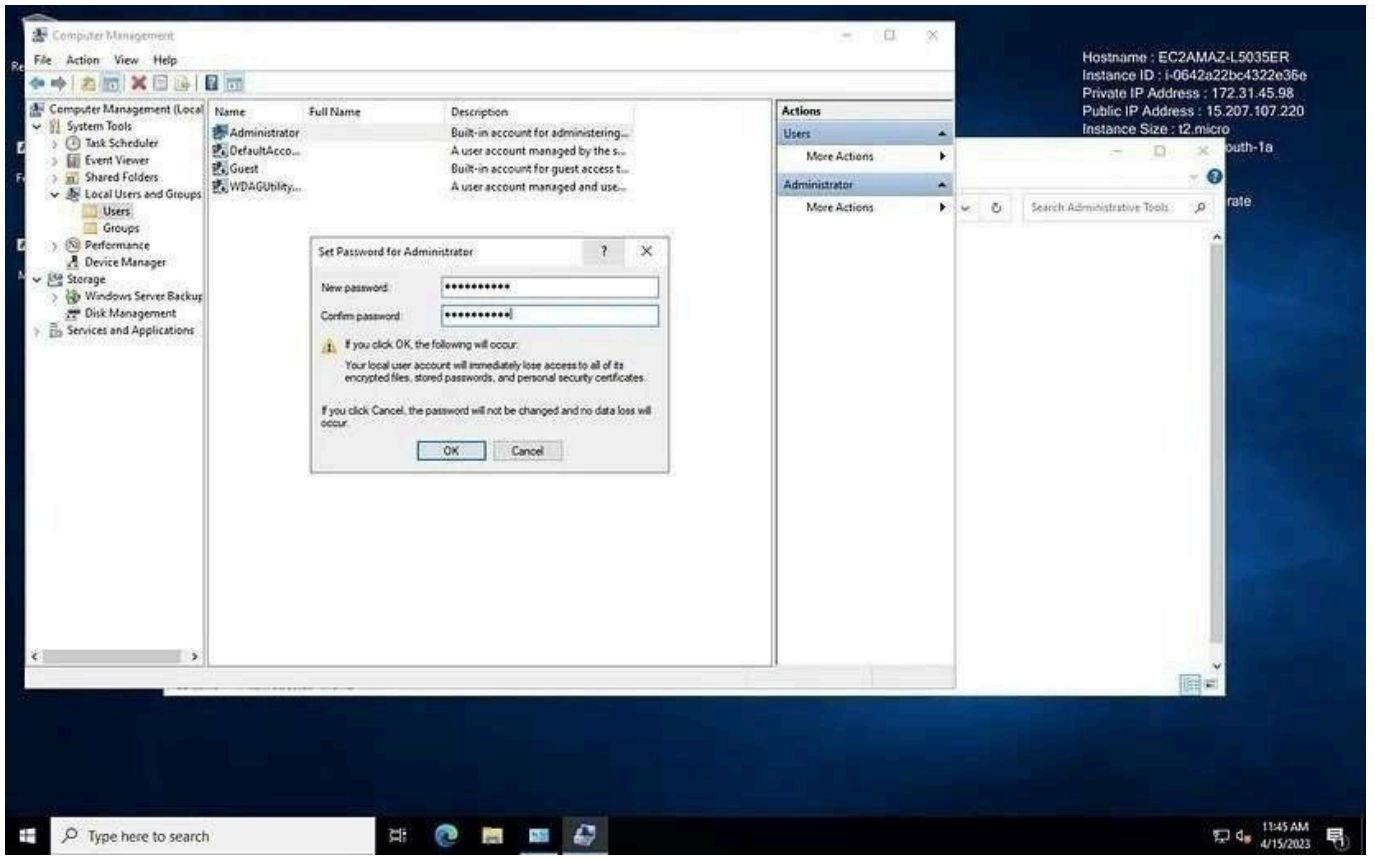

Computer Management
File Action View Help
Computer Management (Local
System Tools
Task Scheduler
Event Viewer
Shared Folders
Local Users and Groups
Users
Groups
Performance
Device Manager
Storage
Windows Server Backup
Disk Management
Services and Applications
Name
Full Name
Description
Administrator
Built-in account for administering...
DefaultAcco...
A user account managed by the s...
Guest
Built-in account for guest access t...
WDAGUtility...
A user account managed and use...
Actions
Users
More Actions
Search Administrative Tools
22 items
1 item selected 1.13 KB
Hostname : EC2AMAZ-L5035ER
Instance ID : i-0642a22bc4322e36e
Private IP Address : 172.31.45.98
Public IP Address : 15.207.107.220
Instance Size : t2.micro
Availability Zone : ap-south-1a
Type here to search
11:35 AM
4/15/2023

Computer Management
File Action View Help
Computer Management (Local
System Tools
Task Scheduler
Event Viewer
Shared Folders
Local Users and Groups
Users
Groups
Performance
Device Manager
Storage
Windows Server Backup
Disk Management
Services and Applications
Name
Full Name
Description
Administrator
Built-in account for administering...
DefaultAcc
A user account managed by the s...
Guest
Built-in account for guest access t...
WDAGUtili
A user account managed and use...
Set Password...
All Tasks
Delete
Rename
Properties
Help
Actions
Users
More Actions
Administrator
More Actions
Search Administrative Tools
Set the user's password.
22 items
1 item selected 1.13 KB
Hostname : EC2AMAZ-L5035ER
Instance ID : i-0642a22bc4322e36e
Private IP Address : 172.31.45.98
Public IP Address : 15.207.107.220
Instance Size : t2.micro
Availability Zone : ap-south-1a
Type here to search
11:39 AM
4/15/2023

Click on the “Proceed”



After setting a new password just click on “ok”



Congratulations, you have successfully created your own Windows EC2 instance in AWS! By following these step-by-step instructions, you can now leverage the power of the cloud to run your Windows-based applications and workloads with ease. Happy computing!

Ec2 Instance | Ec2 Instance Connect | Windows | Launch Ec2 Instance

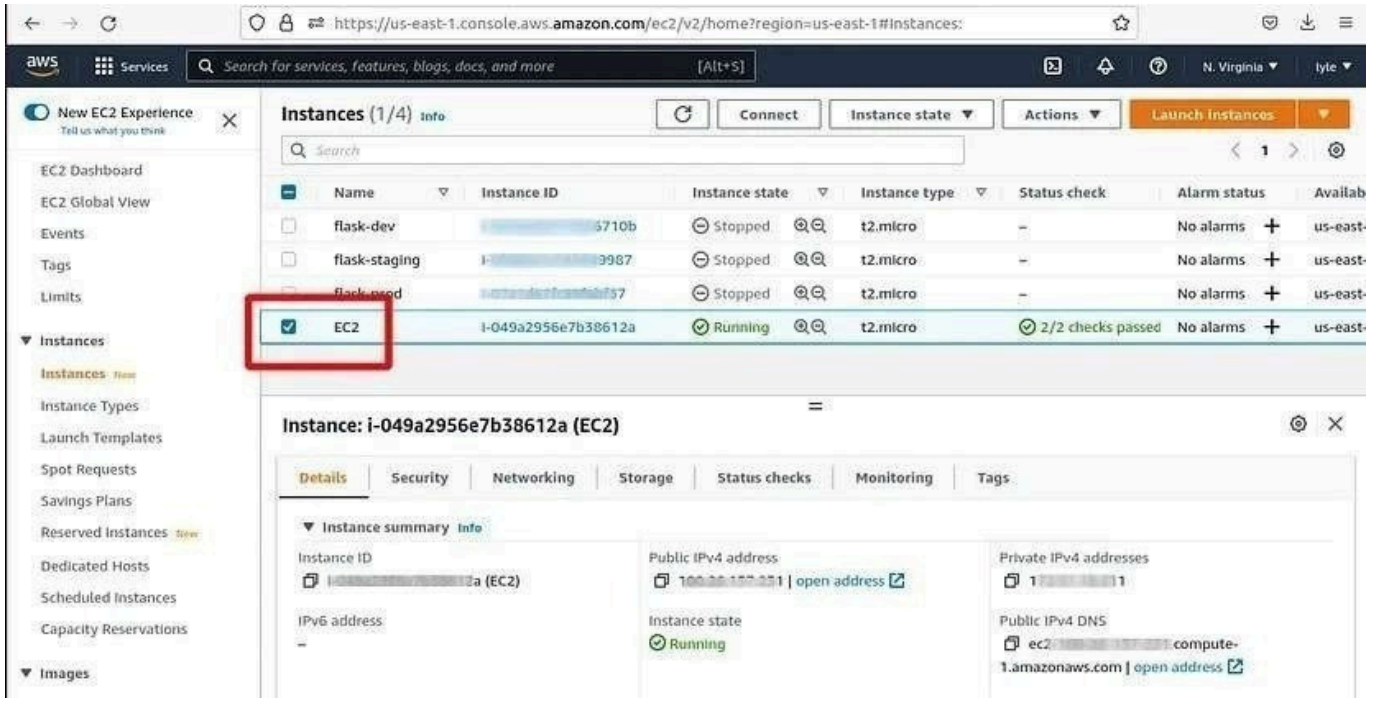
Written by Sanoj

85 Followers·50 Following

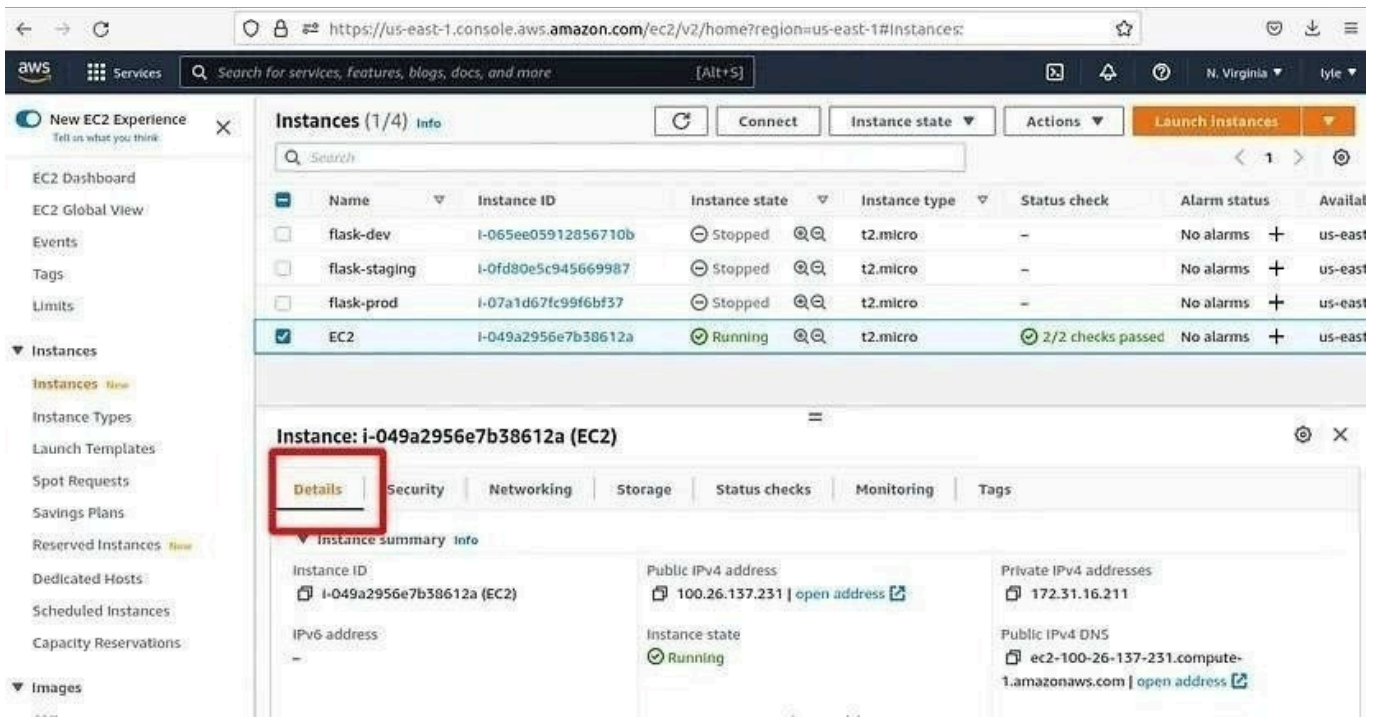
AWS/DevOps Engineer

Follow

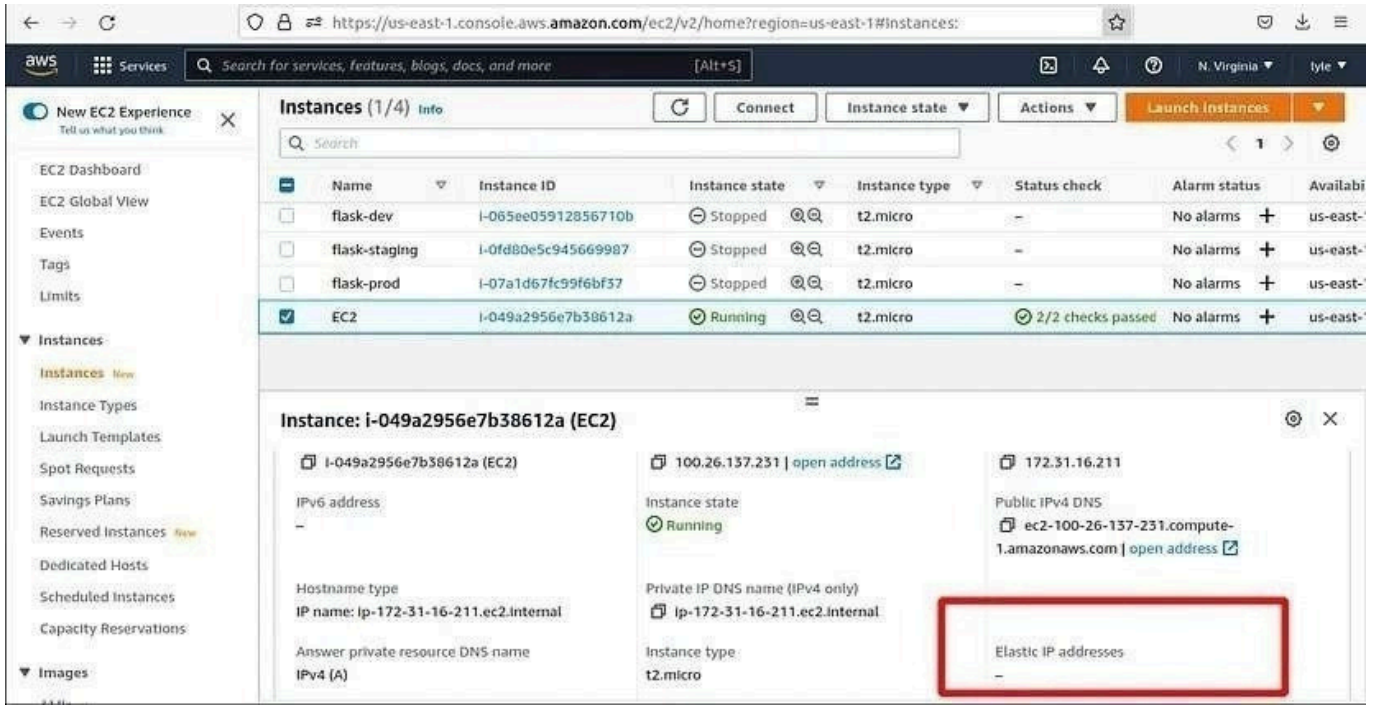
Create an elastic IP Address on AWS



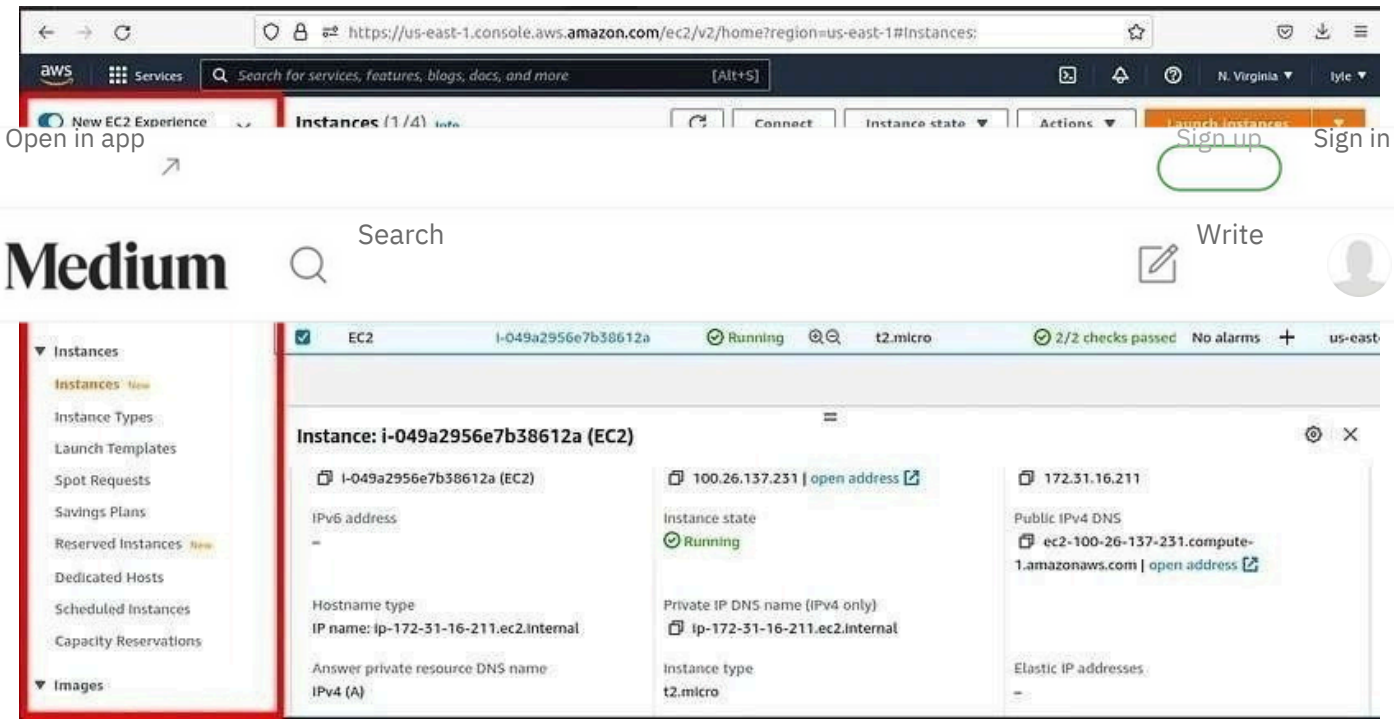
Select the Details tab:



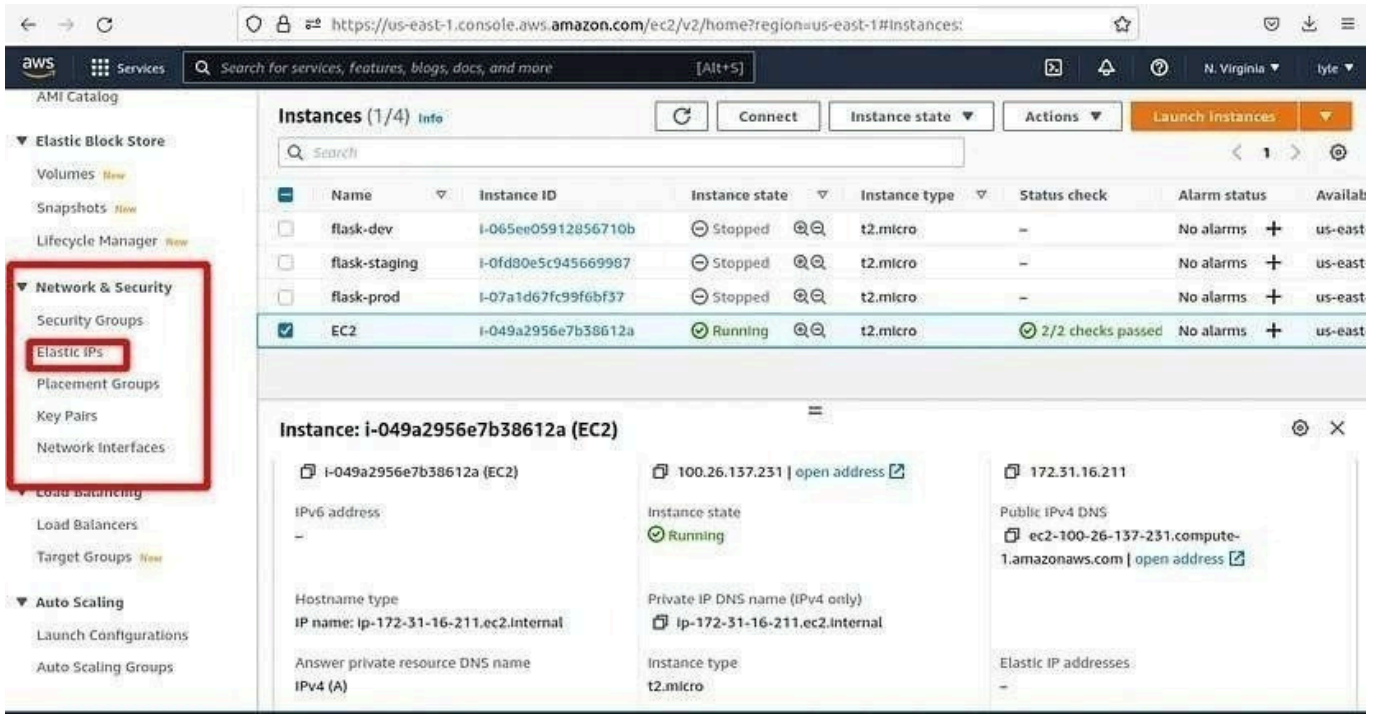
Then Under the Instance summary, scroll down to the Elastic IP addresses section; it should be empty:



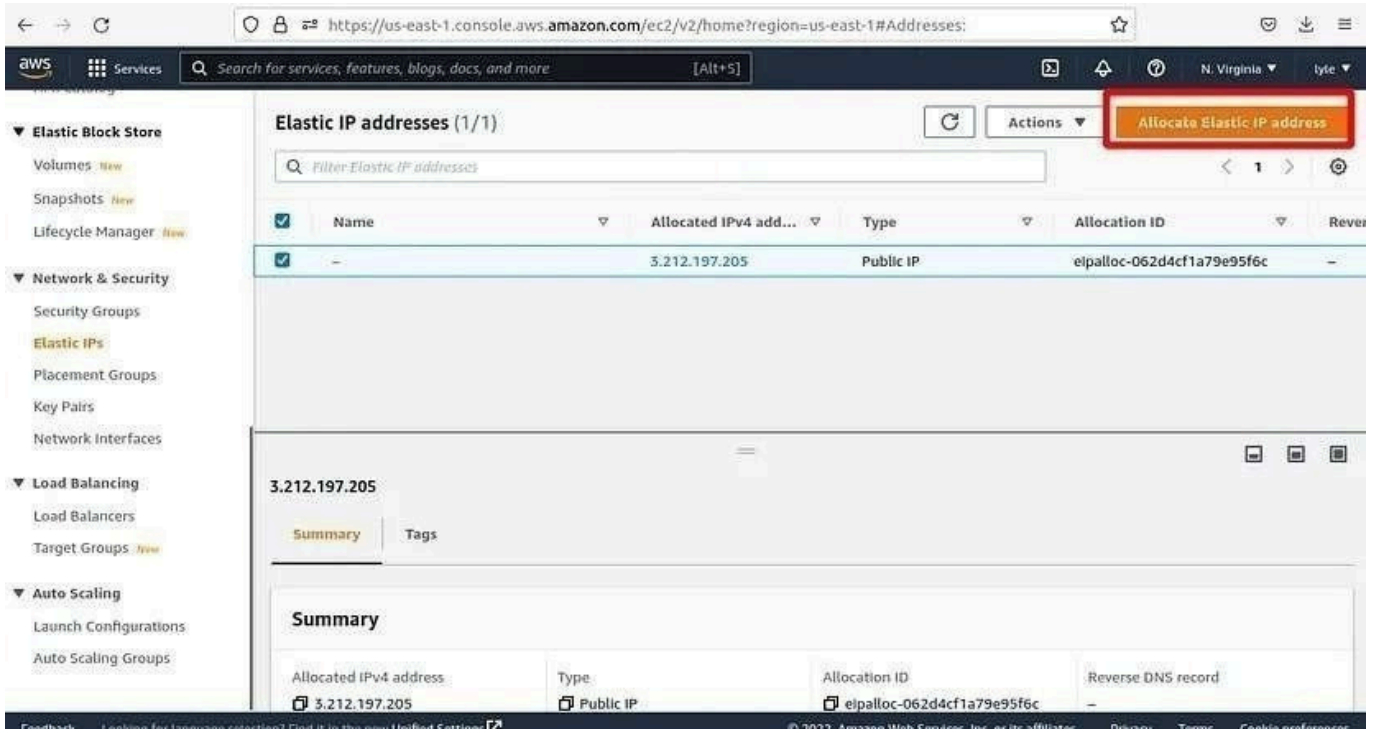
In the same tab in the navigation panel to the left:



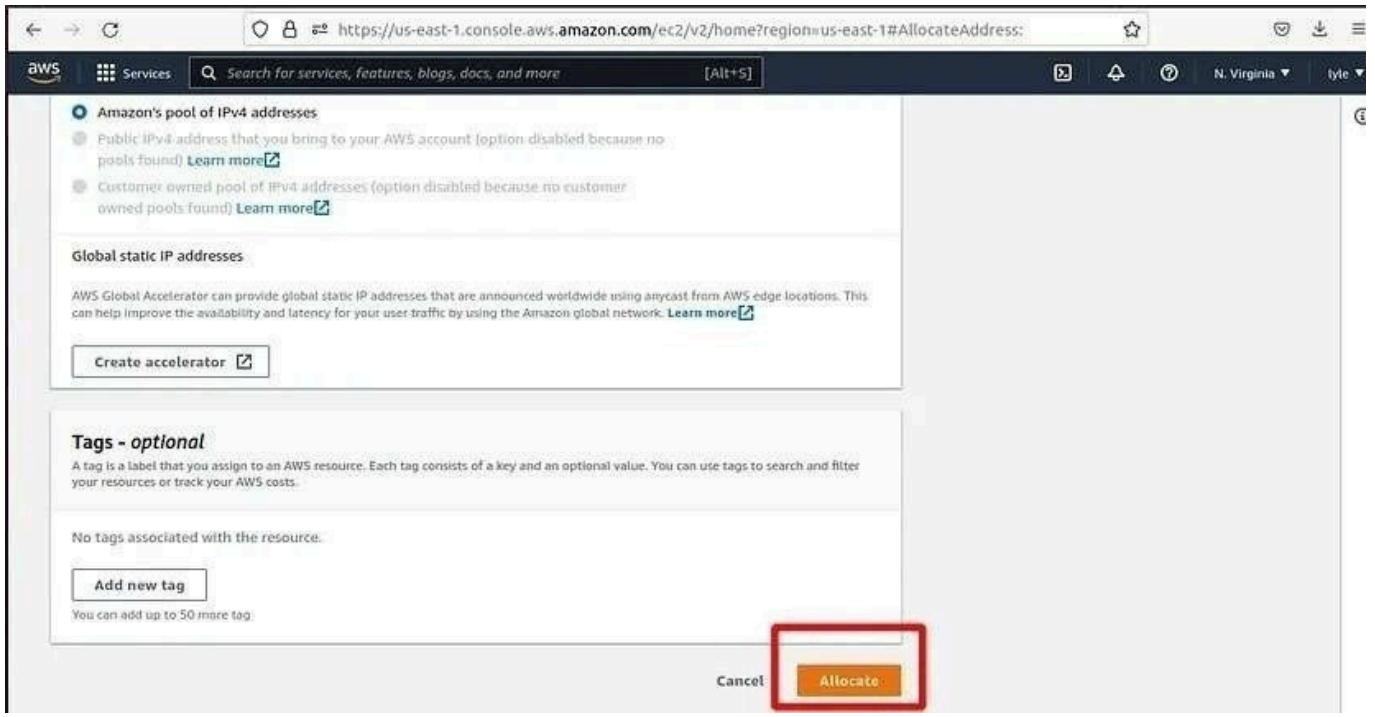
Scroll down to the Network and Security section; within the drop-down, select Elastic IPs:



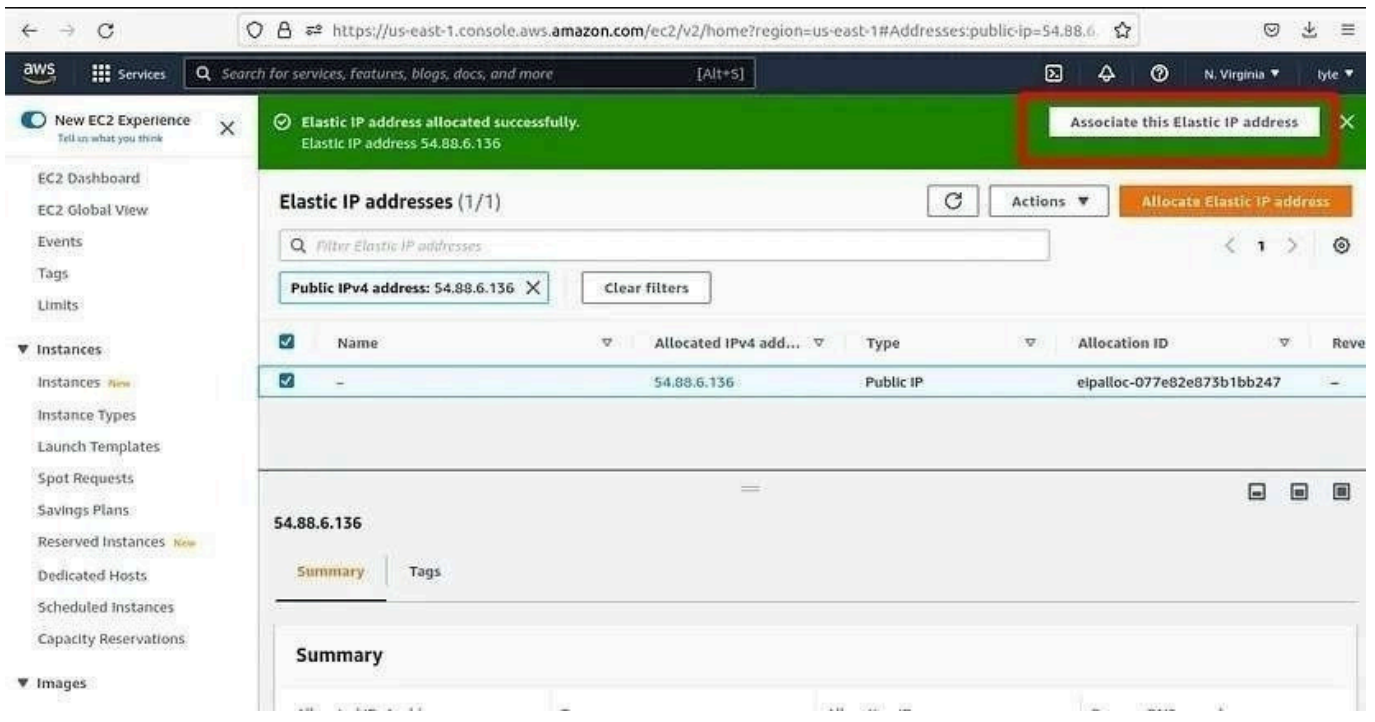
Click on the Allocate Elastic IP address button:



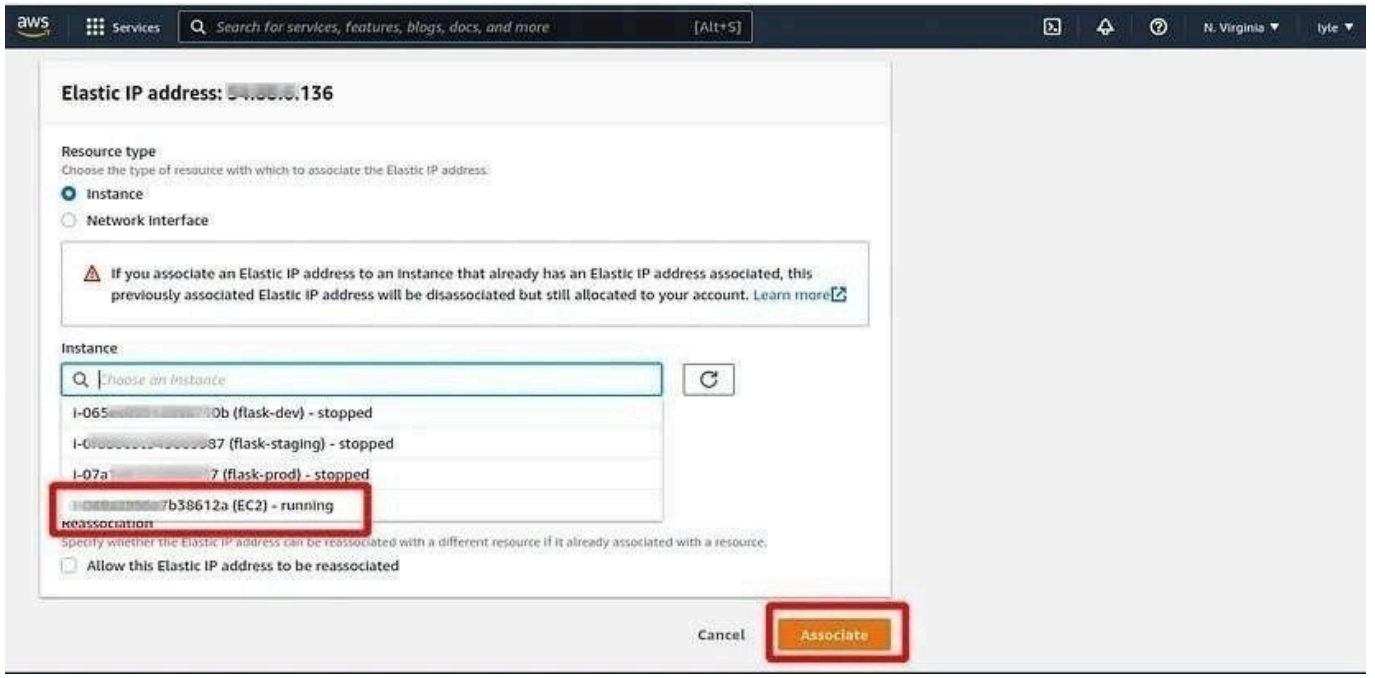
Leave the defaults and click on Allocate:



Then select the Associate this Elastic IP address prompt:



Select the instance to associate it with in the Instance drop-down, then in the Private IP address drop-down, select the instance's private IP address. then click the Associate button:

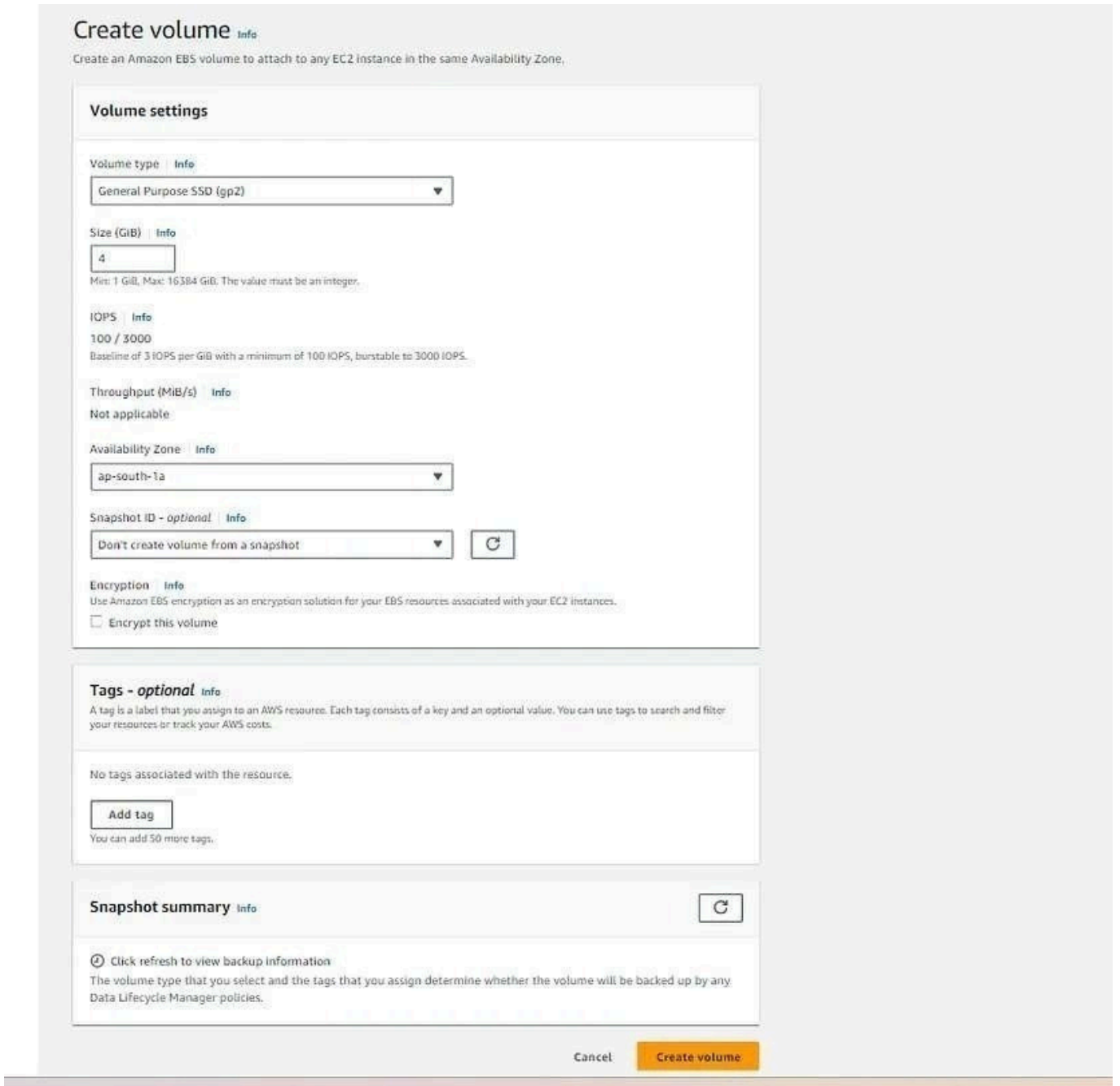


And with that, we have successfully allocated an Elastic IP address to our EC2 server:

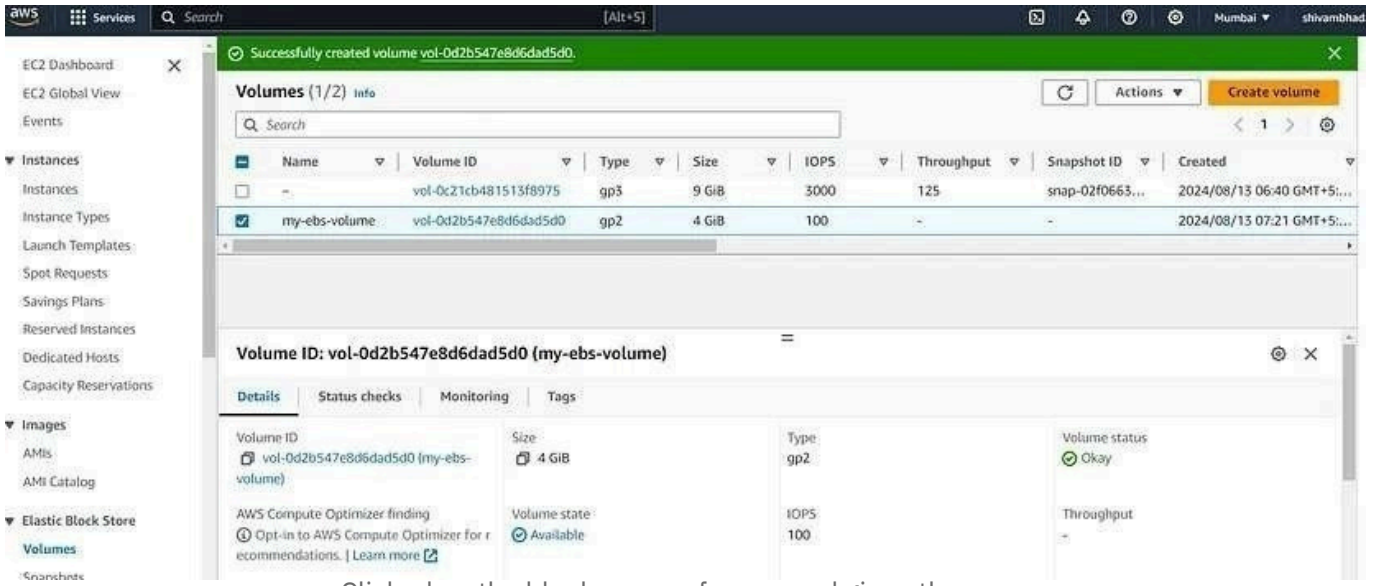
Create EBS volume and attach(mount) to Ec2

Back to the topic, Click on “Create Volume” as you can see in the upper right corner.

2. I selected the Volume type as gp2 and gave 4 GB size. And the same region “ap-south-1a” as our instance. Then click on “Create Volume” on the bottom.

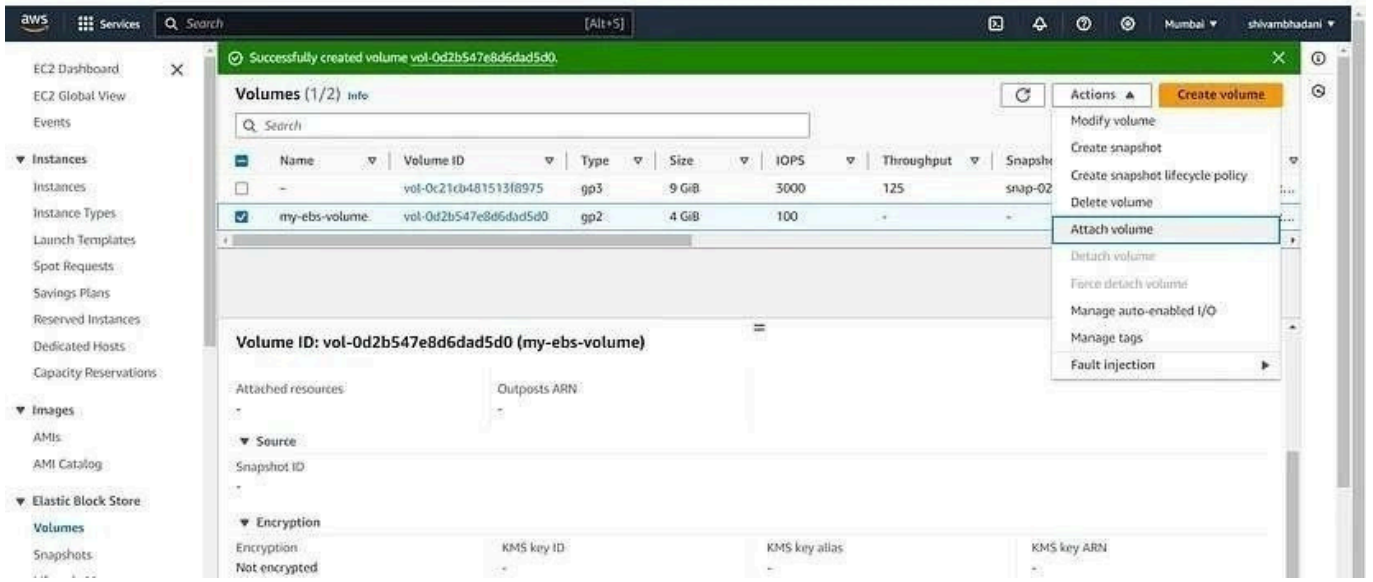


3. Just gave the name “my-ebs-volume” to our newly created EBS Volume for simplicity and understanding.

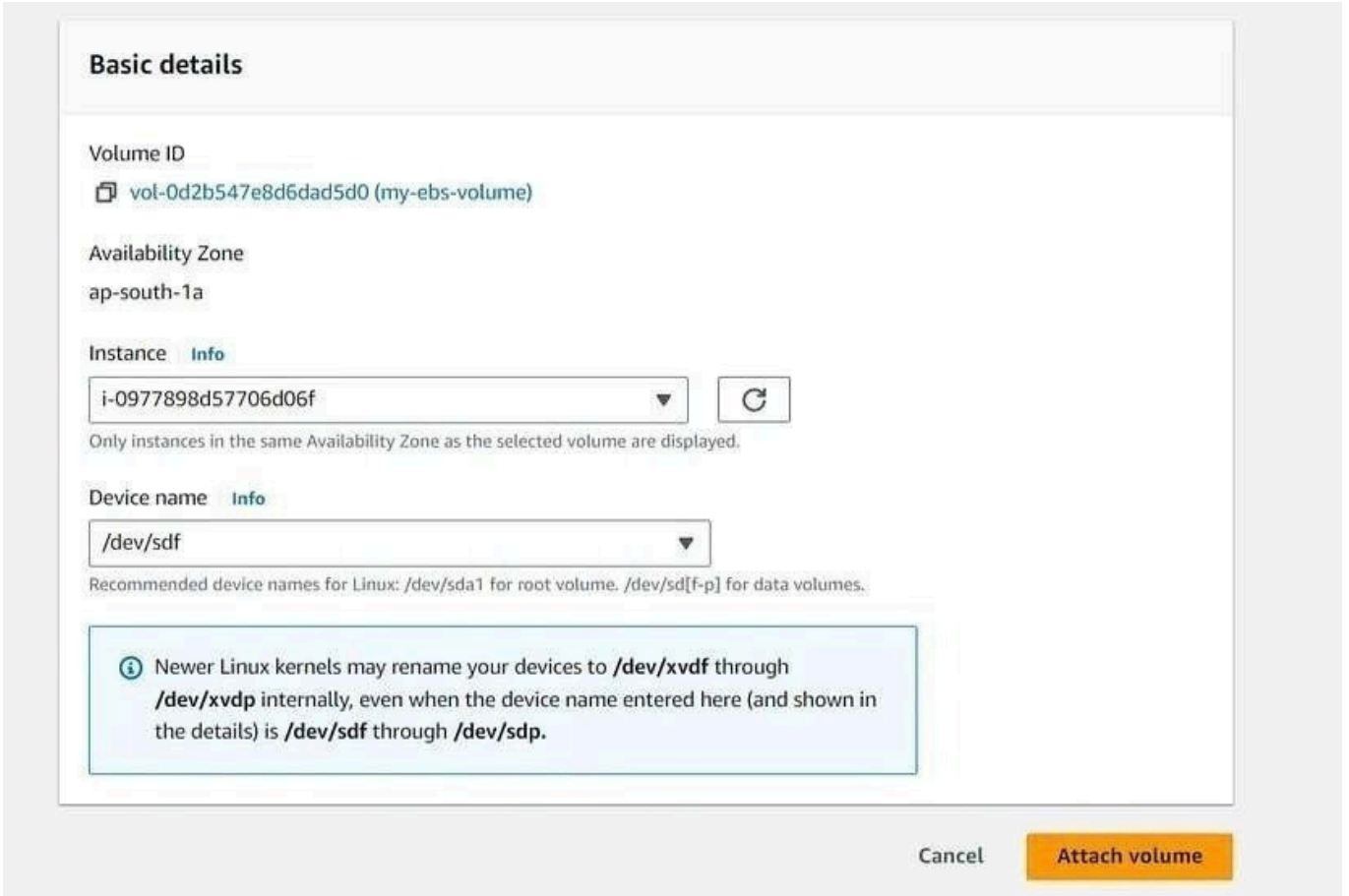


Clicked on the blank space of name and given the name

4. Select “my-ebs-volume” checkbox then Click on “Actions” then click on “Attach volume” to attach it to our ec2 instance.



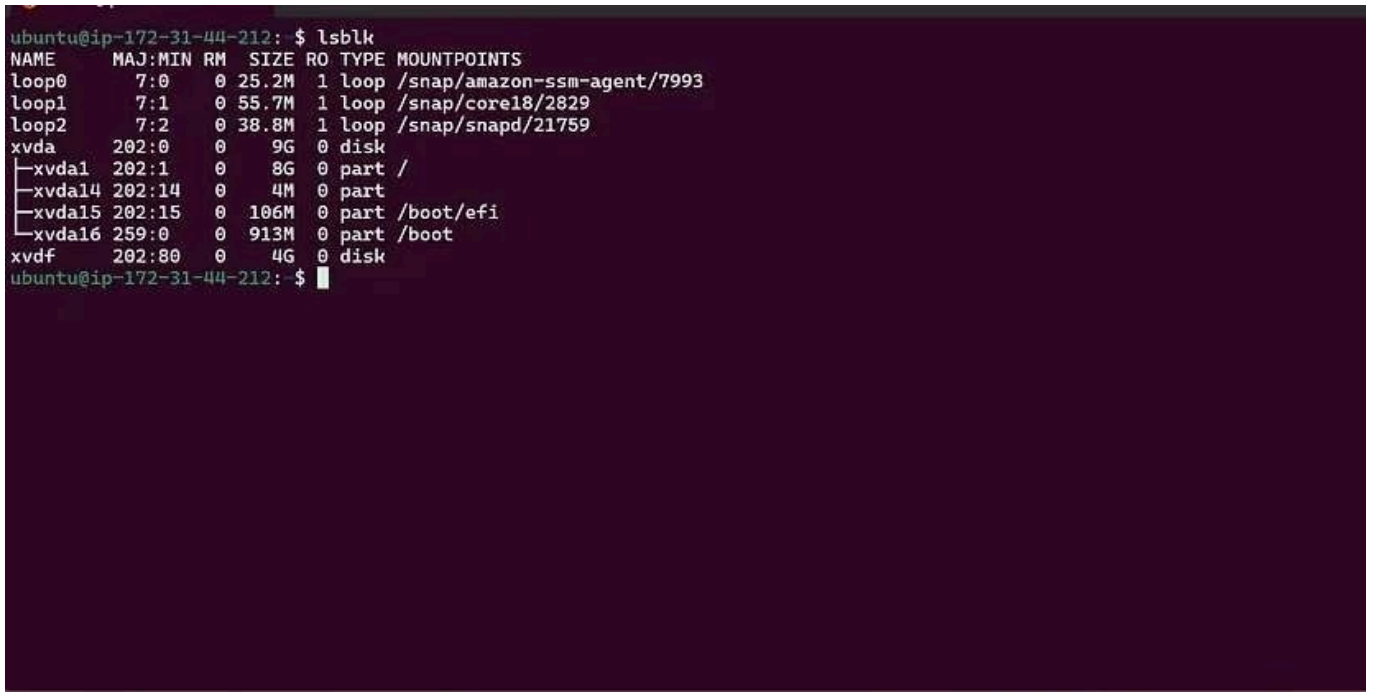
5. Select the instance (in the same availability zone) for which you want to attach this Volume. I have selected the same instance “shivam-ebs-tutorial” as I launched above.
You can select any device type, it won’t matter. Then click on “Attach Volume” button.



6. Now, volume “my-ebs-volume” is attached to the instance “shivam-ebs-tutorial”. We have reached till image 2 of our visualization tutorial.
SSH to my EC2 machine to download code on “my-ebs-volume” to reach image 3.

7. Everything we will do on this EC2 instance, we have to do in the terminal because there is no GUI available here.
We have to check, what block devices (hard disk, EBS Volumes) etc attached to this machine. The below command is for this:

```
lsblk
```



You can see, there are two volumes: one is “xvda” of 9 GB (this is the default root volume that we gave during the launch instance) and the second is “xvdf” of 4 GB (this is “my-ebs-volume” we created. Here we will download our code).

If it is your first time and you don’t know all the below steps that I am doing with Linux commands then just cram it here. If you know Linux then these are very common for you.

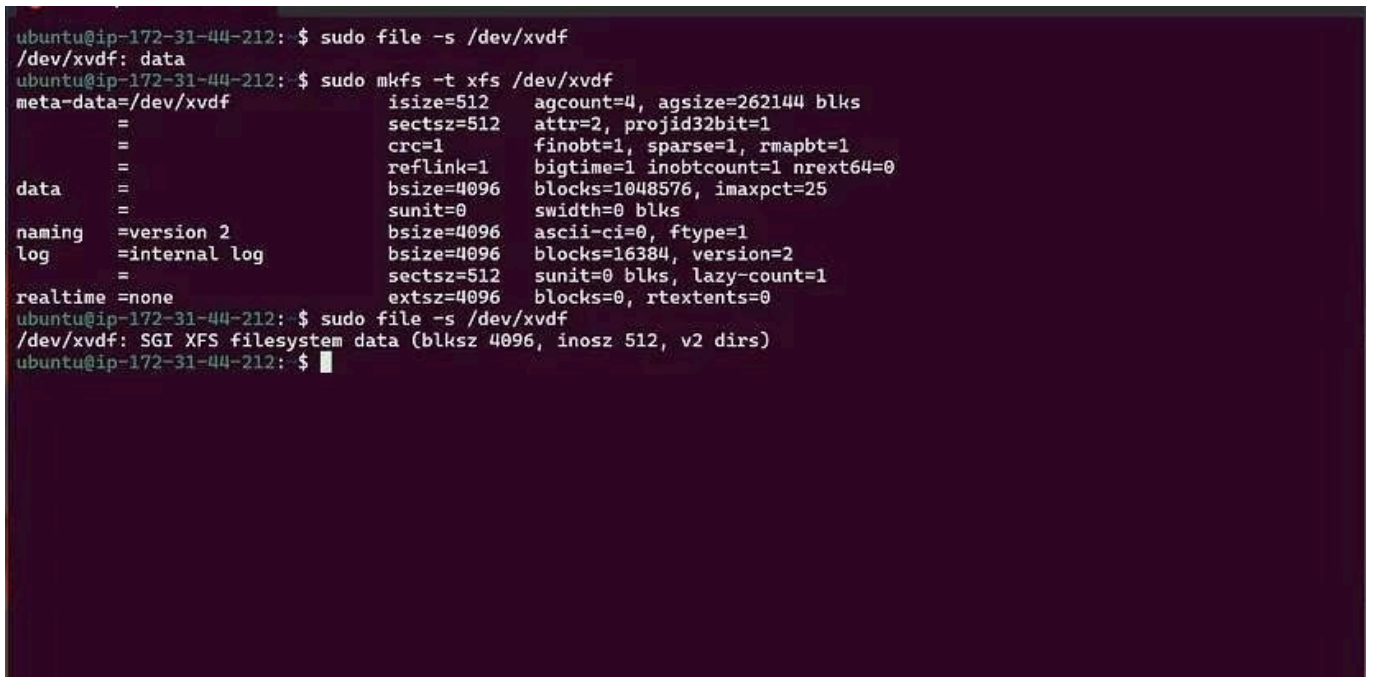
- First, we have to enable the File System for newly created xvdf storage. If you are attaching any existing EBS volume then you don’t need to do this. But, since we created this EBS Volume fresh and haven’t enabled the file system till now, we have to do it. To check file system is enabled or not, type the below command

```
sudo file -s /dev/xvdf
```

If the response comes as “/dev/xvdf: data”, then it is not enabled.

- Enable File System, type the below command

```
sudo mkfs -t xfs /dev/xvdf
```



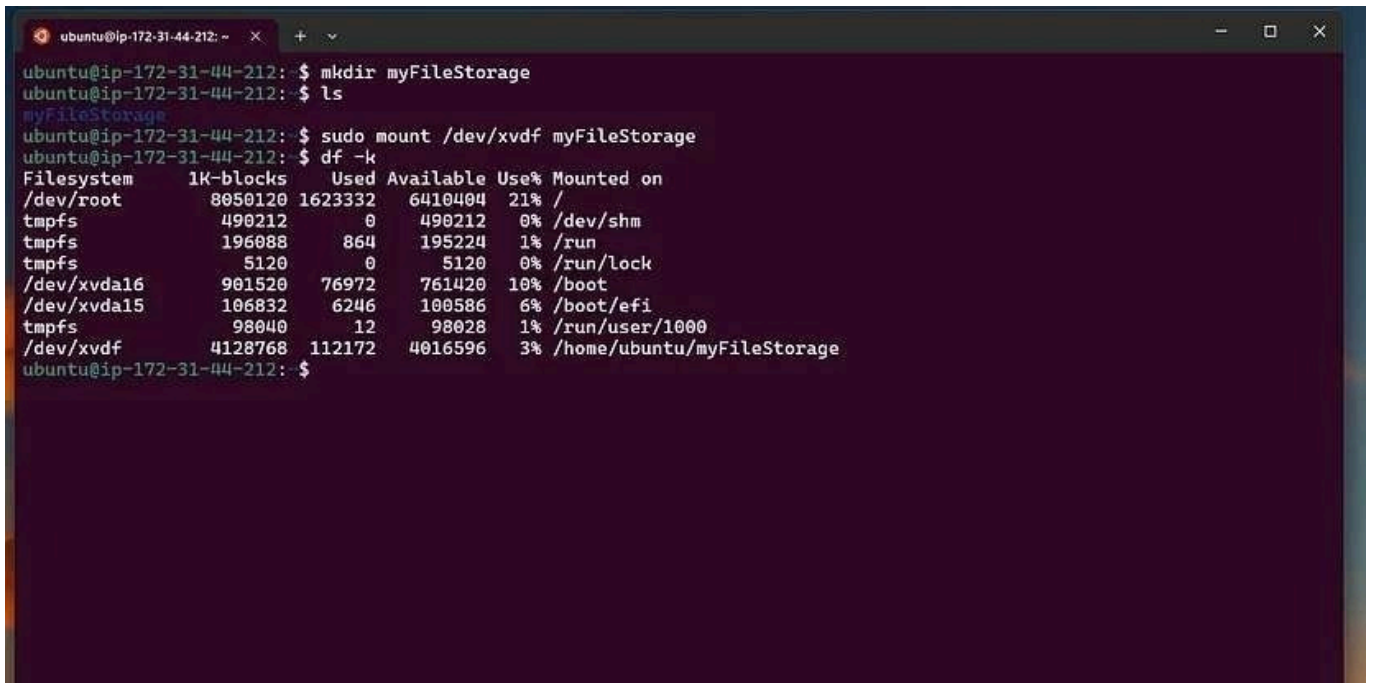
- Make a folder and mount that file system into this folder. Now all the things that you download or make in this folder will be created in the file system of EBS Volume.

Make a folder with the name “myFileStorage” from the below command:

```
mkdir myFileStorage
```

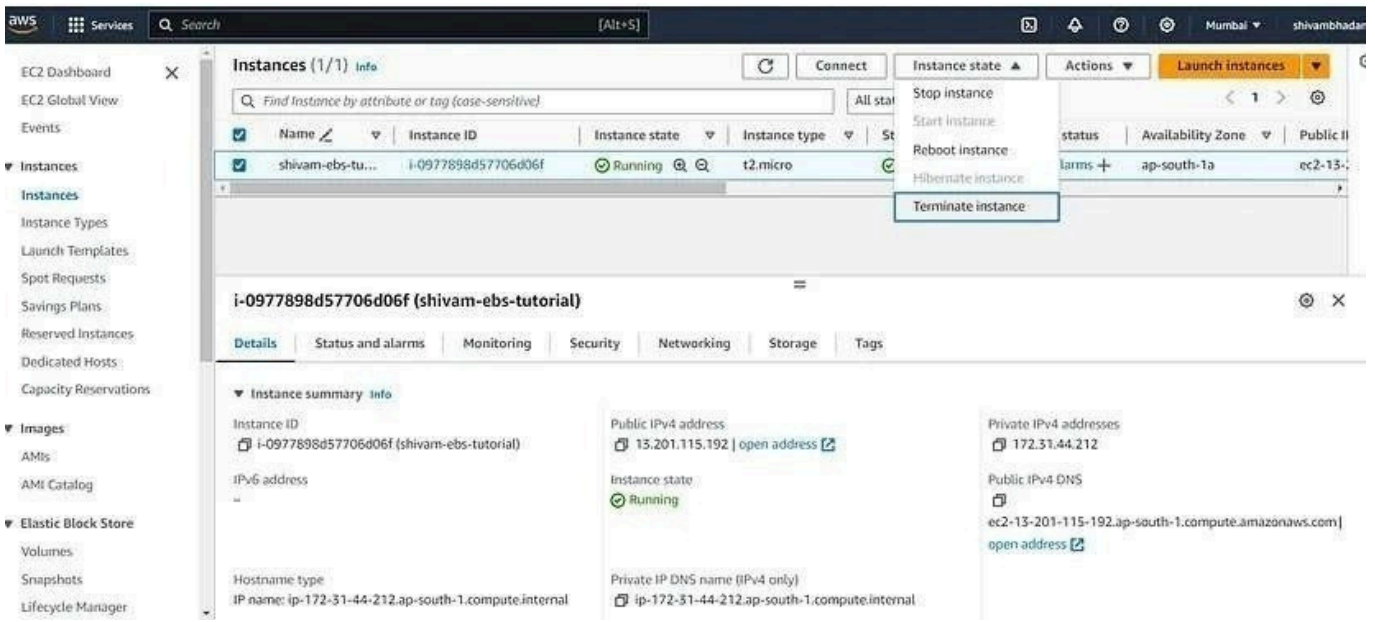
Mount the file system of “xvdf” into this folder from below command:

```
sudo mount /dev/xvdf myFileStorage
```



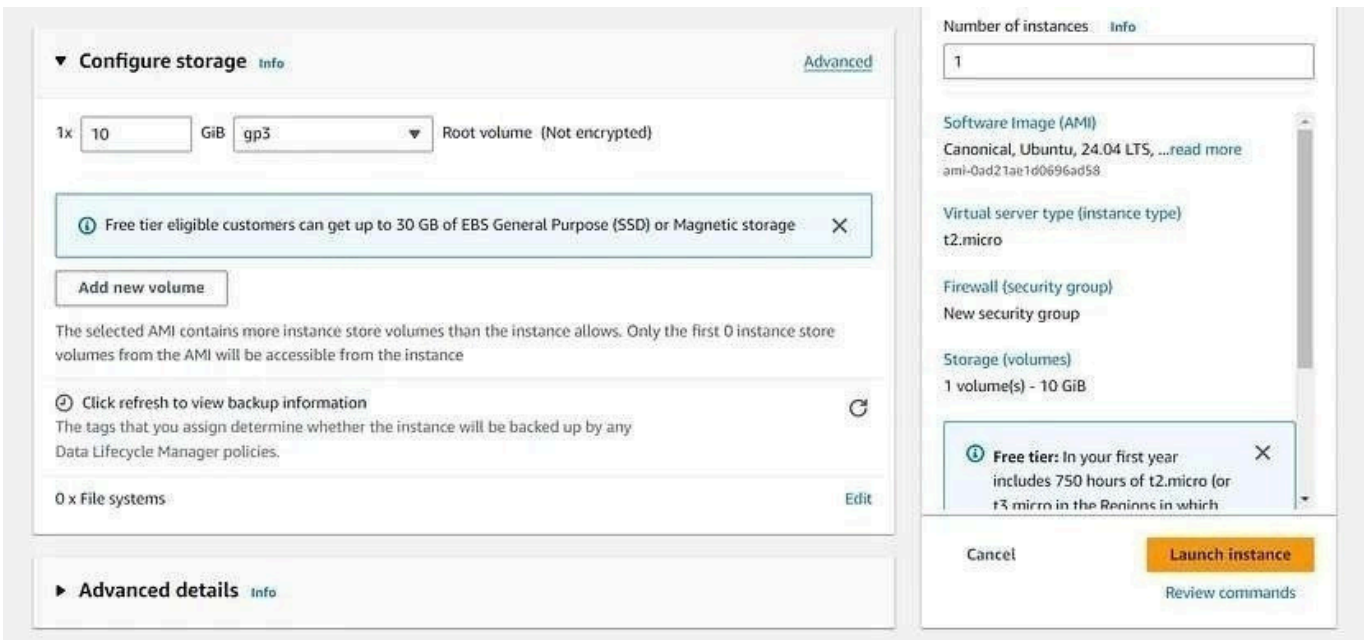
Now, our EBS volume is mounted to folder “myFileStorage”. Whatever we make inside this folder will be kept inside our EBS Volume.

8. Terminate the instance “shivam-ebs-tutorial”.

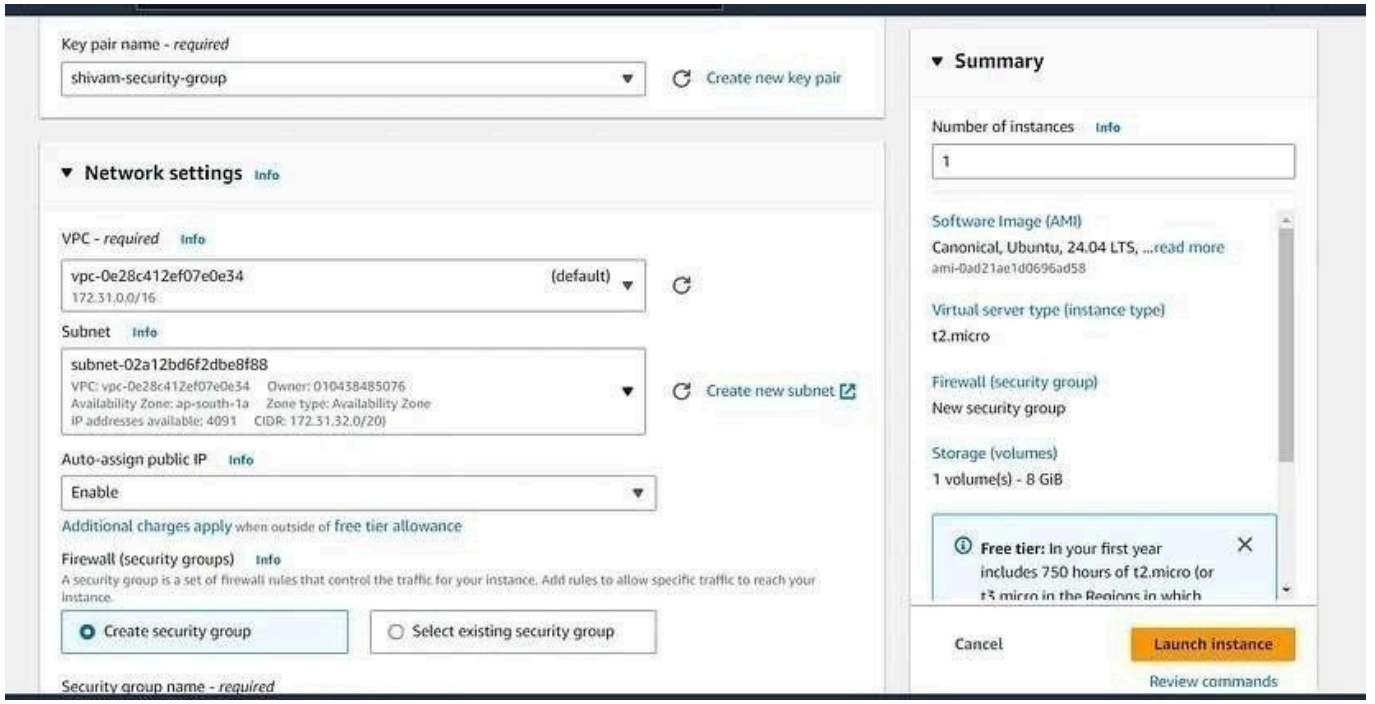


Completed till image 4 of the visualization tutorial.

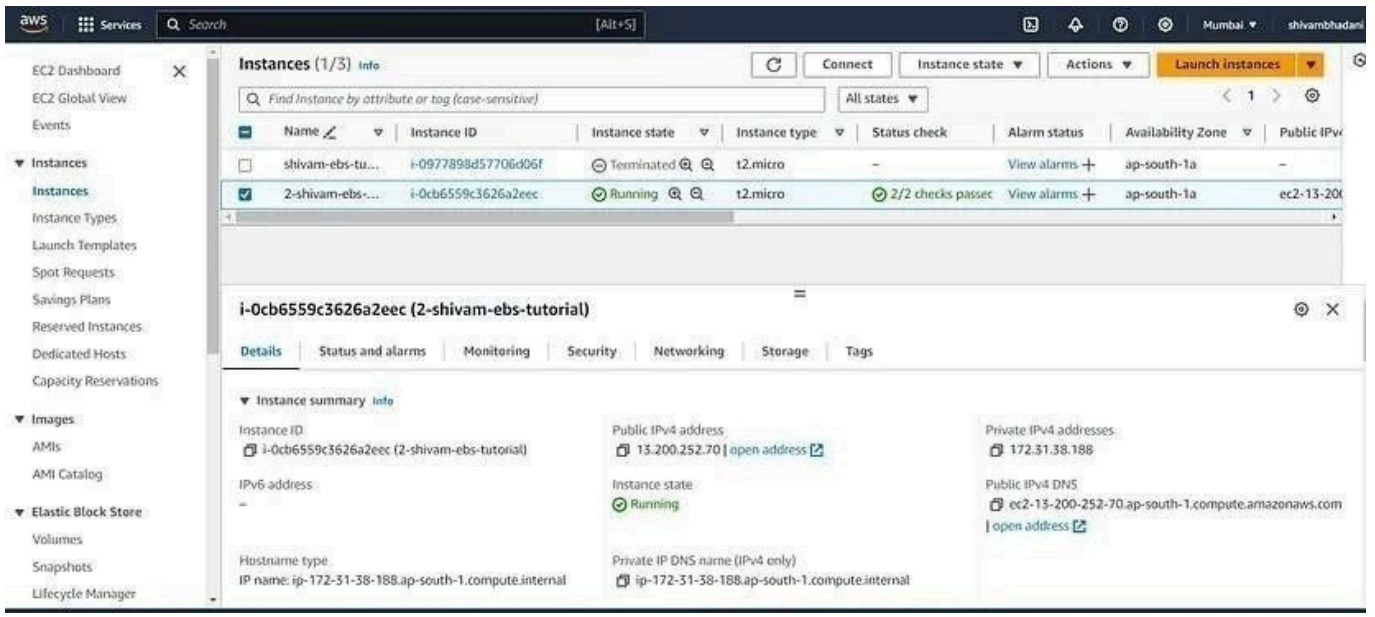
9. Launch a new EC2 Instance with name “2-shivam-ec2-tutorial” with default root EBS Volume 10 GB.



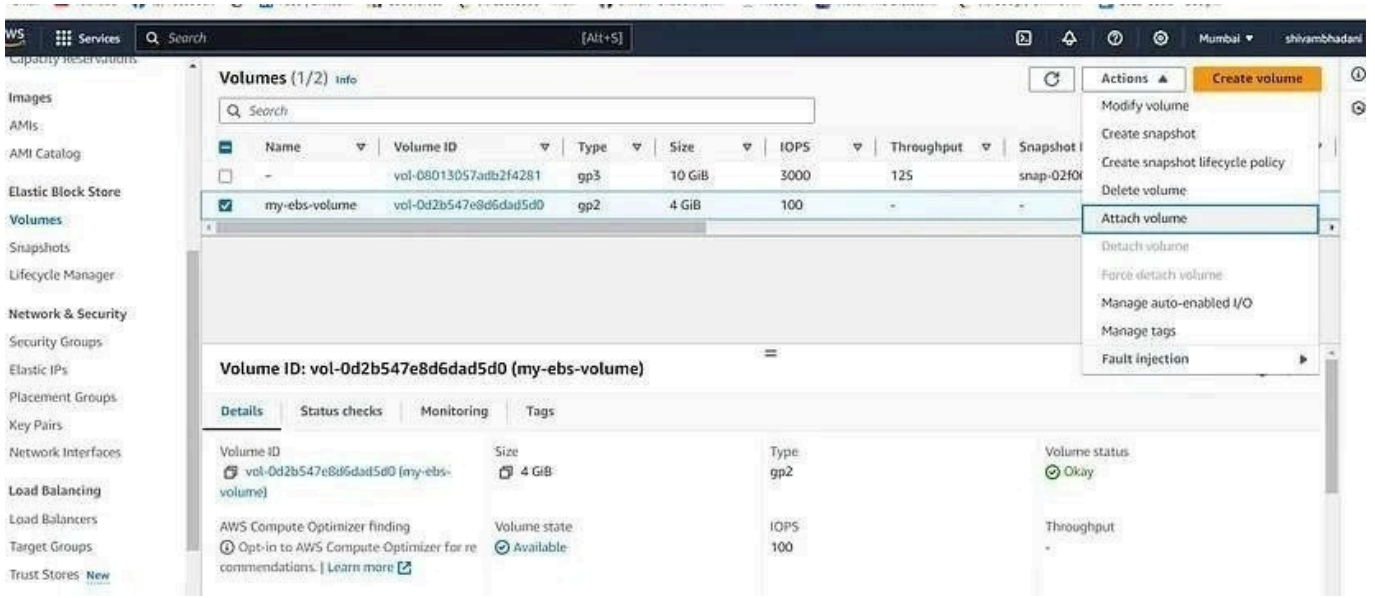
While launching EC2 Instance, go to the network setting and select a subnet of “ap-south-1a”. It will launch our instance in ap-south-1a. We are doing it because our EBS Volume is in ap-south-1a. Don’t worry, if you don’t know about the subnet. We will study it in detail in upcoming blogs just like we are studying EBS Volume now.



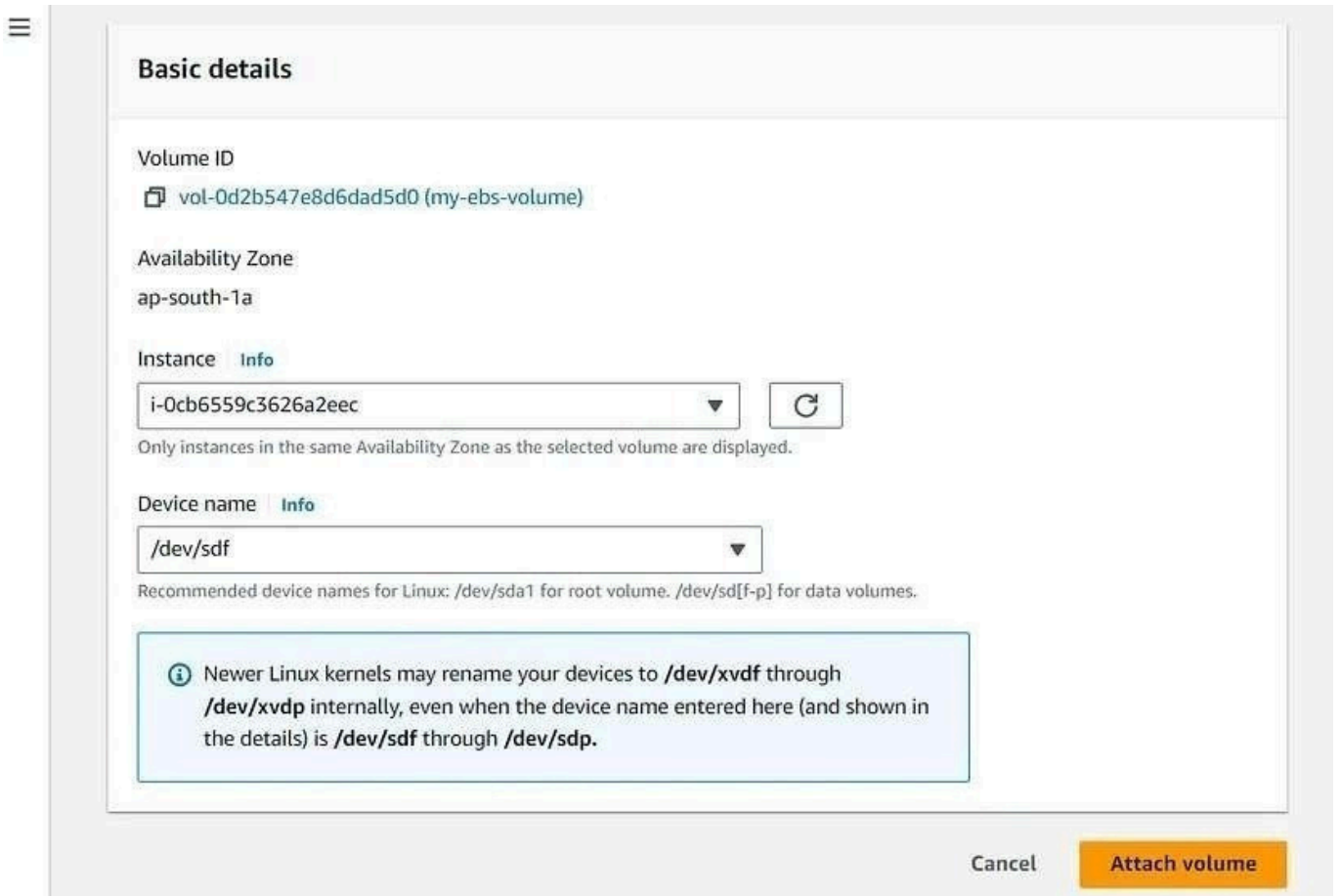
Click on “Launch Instance” and our instance named “2-shivam-ebs-tutorial” launched in ap-south-1a.



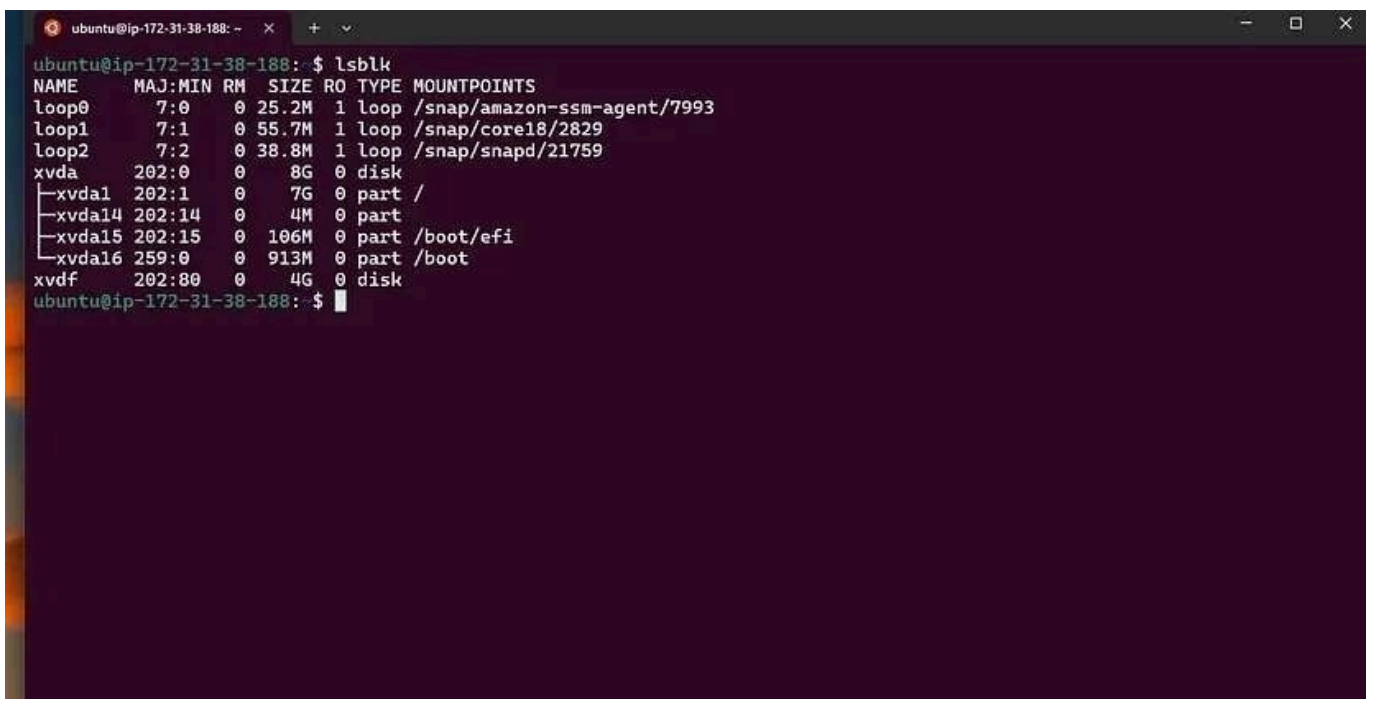
10. Now, go to Volumes and select the same EBS Volume “my-ebs-volume” to attach it to our new EC2 instance.



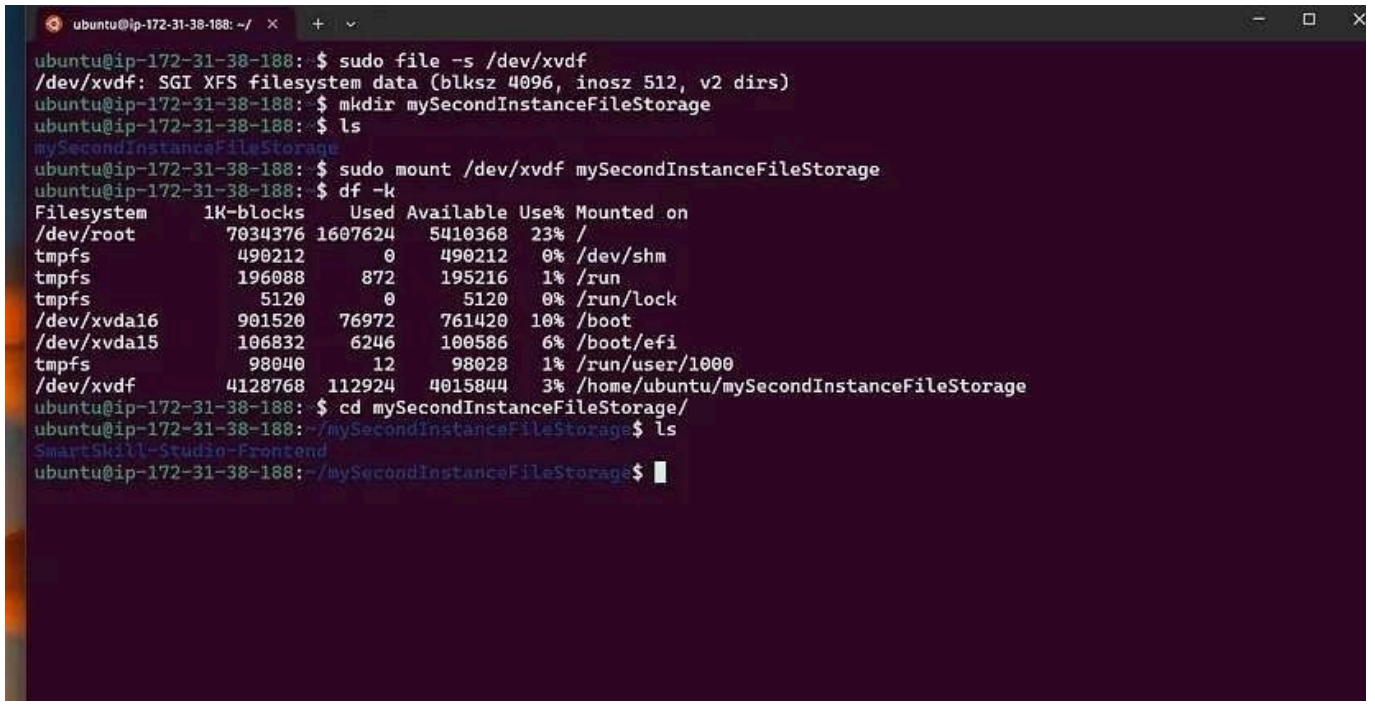
Select the newly created EC2 Instance and any device type.



11. Do SSH in the new EC2 instance “2-shivam-ebs-tutorial” and run lsblk to check if the volume is correctly attached or not.



Since we already created the file system of this EBS Volume so we don’t need to do it now. Just make any folder in this EC2 instance and attach this file system to that folder. Then check if the code that we installed previously persists or not.



df -k command is to check the mount folder.

You can see in the above image that when we do “ls”, we get the same data that we downloaded previously. This shows data persists.

And finally, we complete image 5 of the Visualization Tutorial.

This was a little boring but if you read carefully till now, then I hope you understood everything.

Changing the size of existing EBS volume without downtime

1. Select the EBS Volume by clicking the checkbox
2. Click to “Action” button on the top right corner
3. Click on resize volume then give the new volume size.
   Note: You can only increase the size, which means if you have initially given 4 GB then you can only give a size which is greater than 4GB.

How to Use a Snapshot to Restore an AWS EC2 Instance

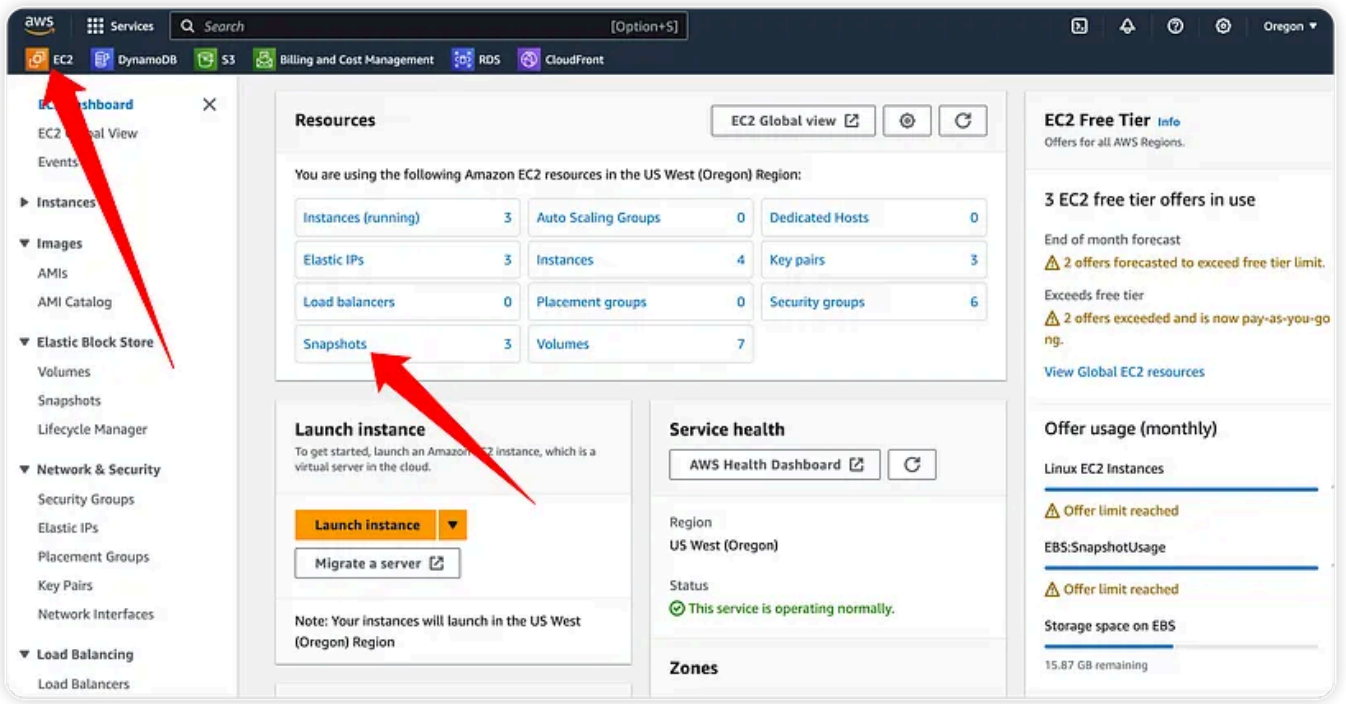
Step 1: Create a Snapshot

If you don’t already have a snapshot, you’ll need to create one from your existing EBS volume.

Navigate to the EC2 Dashboard:

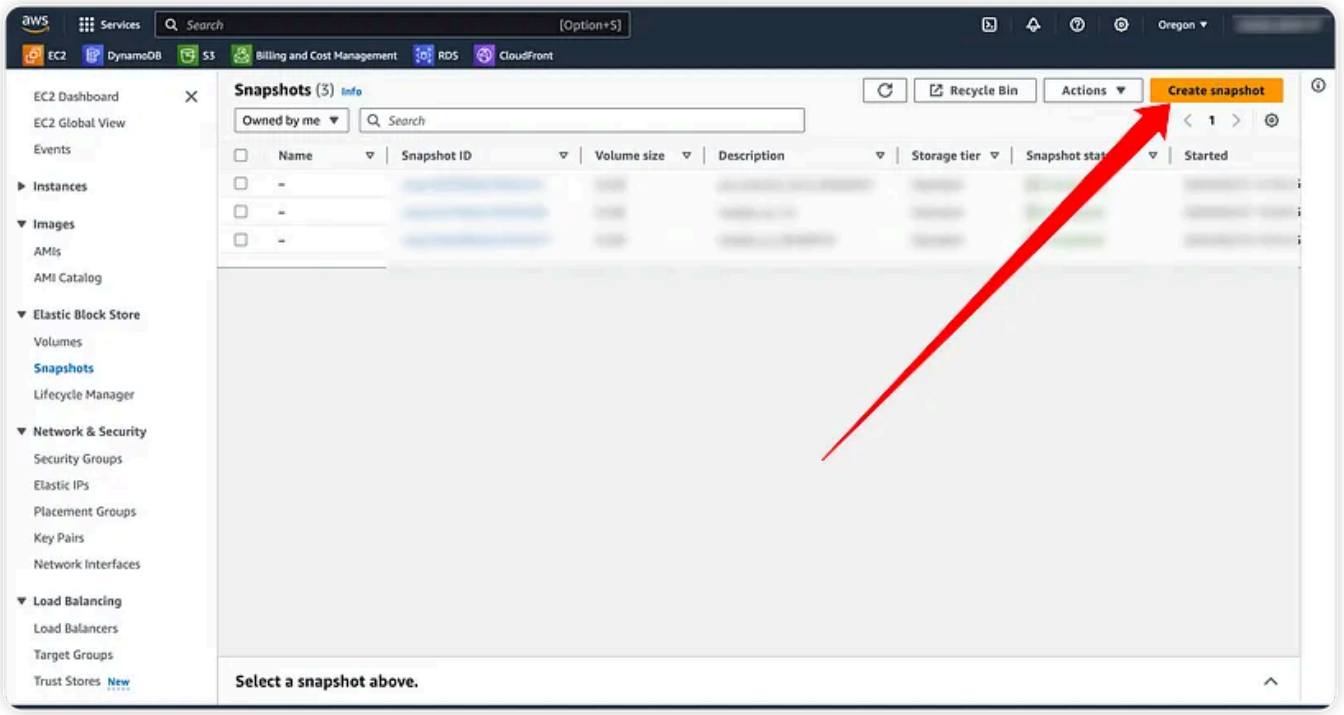
Go to the AWS Management Console.

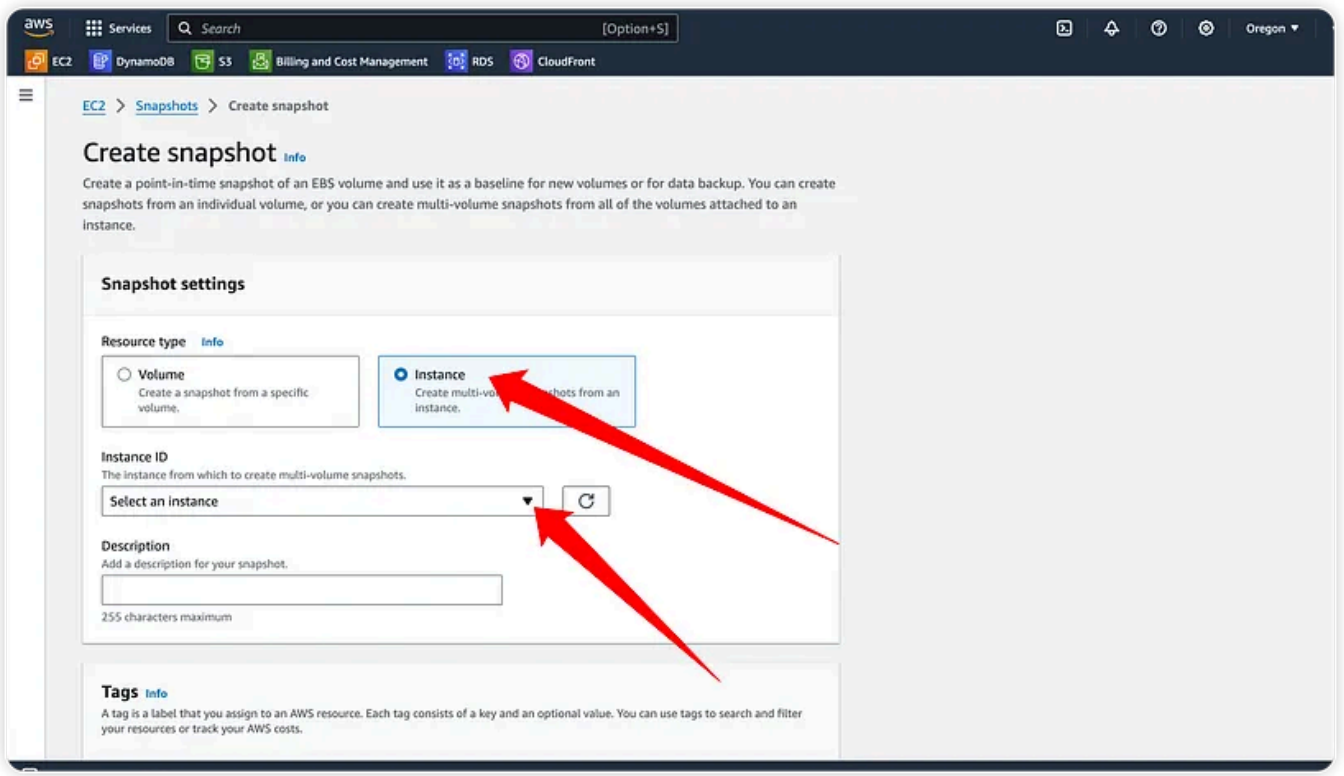
Select EC2 from the Services menu.



Create a Snapshot:

- In the left-hand navigation pane, click on Snapshots under the Elastic Block Store section.
- Click on the Create Snapshot button.
- Select the volume you want to create a snapshot of and provide a description.
- Click Create Snapshot.





Step 2: Create a Volume from the Snapshot

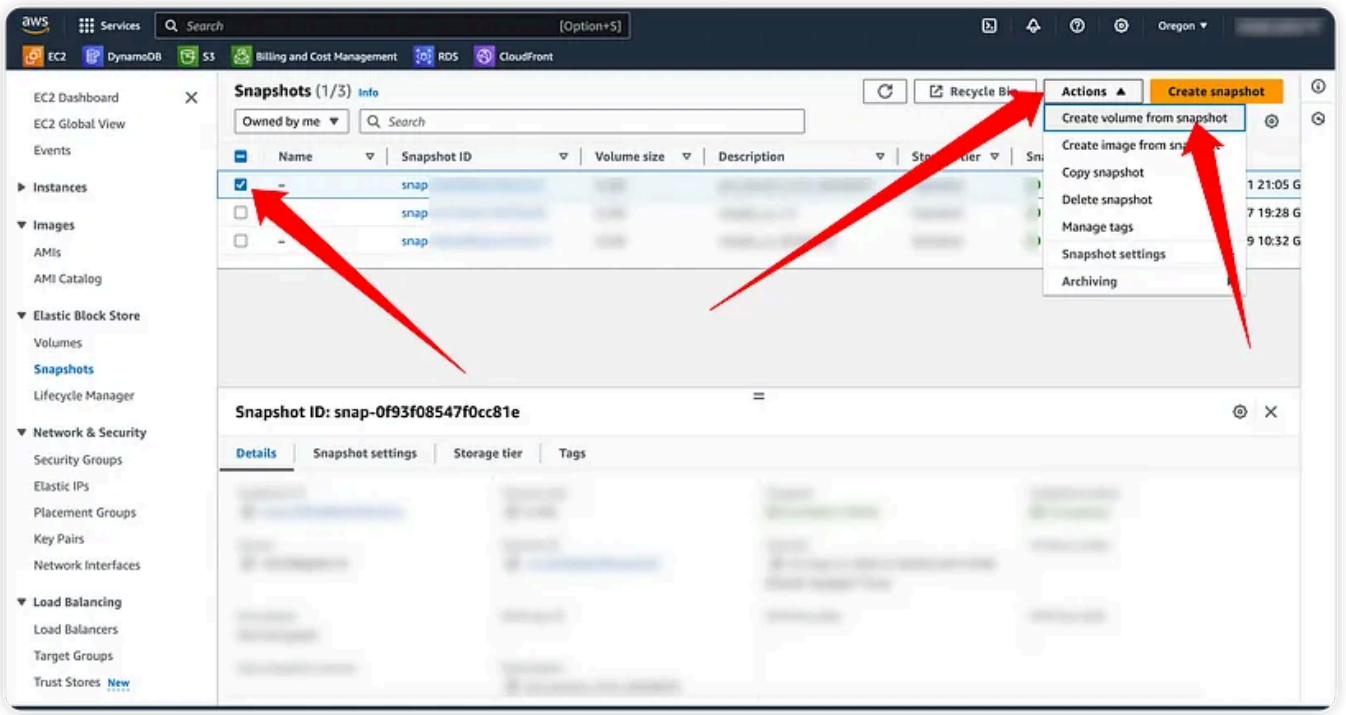
Once you have the snapshot, the next step is to create a new EBS volume from the snapshot.

Navigate to Snapshots:

- In the EC2 Dashboard, select Snapshots .

Create Volume:

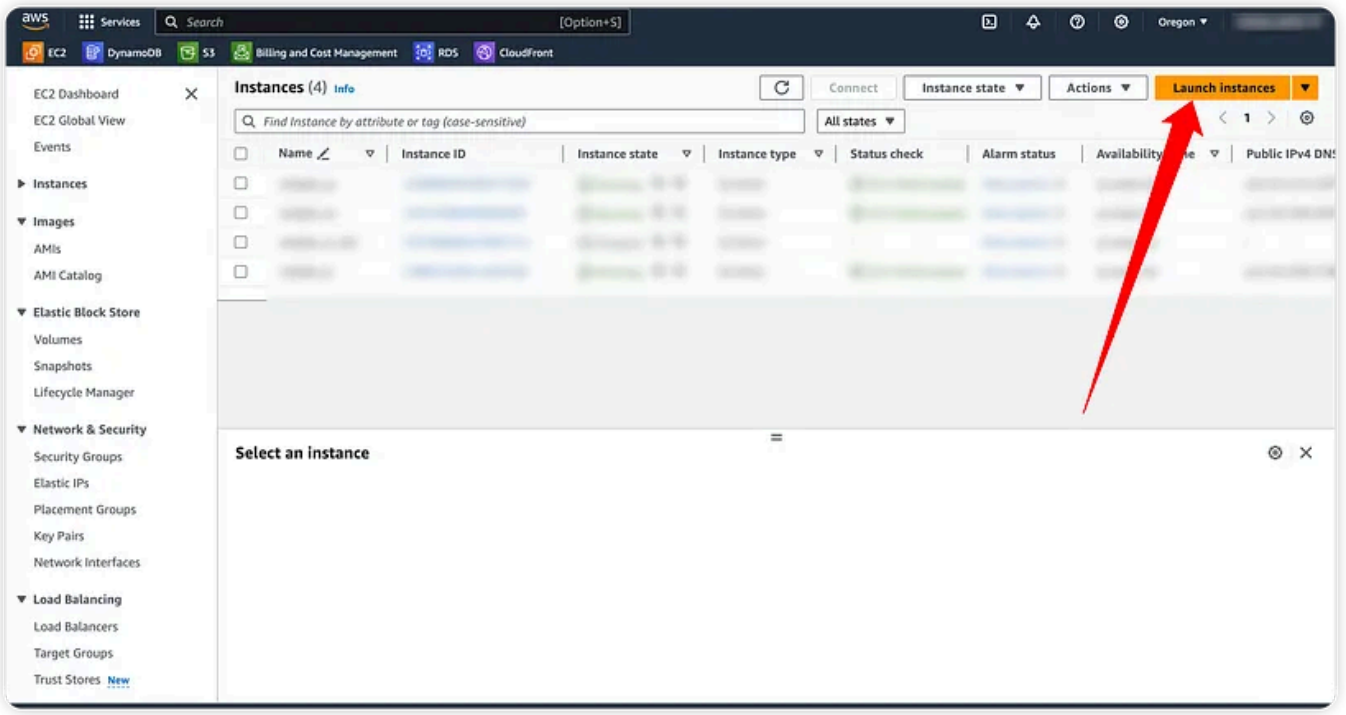
- Select the snapshot you created.
- Click Actions and choose Create Volume.
- In the Create Volume dialog:
- Ensure the Availability Zone matches where you want to launch the new instance.
- Click Create Volume.



Step 3: Launch a New EC2 Instance

Launch an Instance:

- Go to the Instances section in the EC2 Dashboard.
- Click Launch Instance .



Select an AMI:

- Choose an Amazon Machine Image (AMI) that matches your application’s requirements.

Choose an Instance Type:

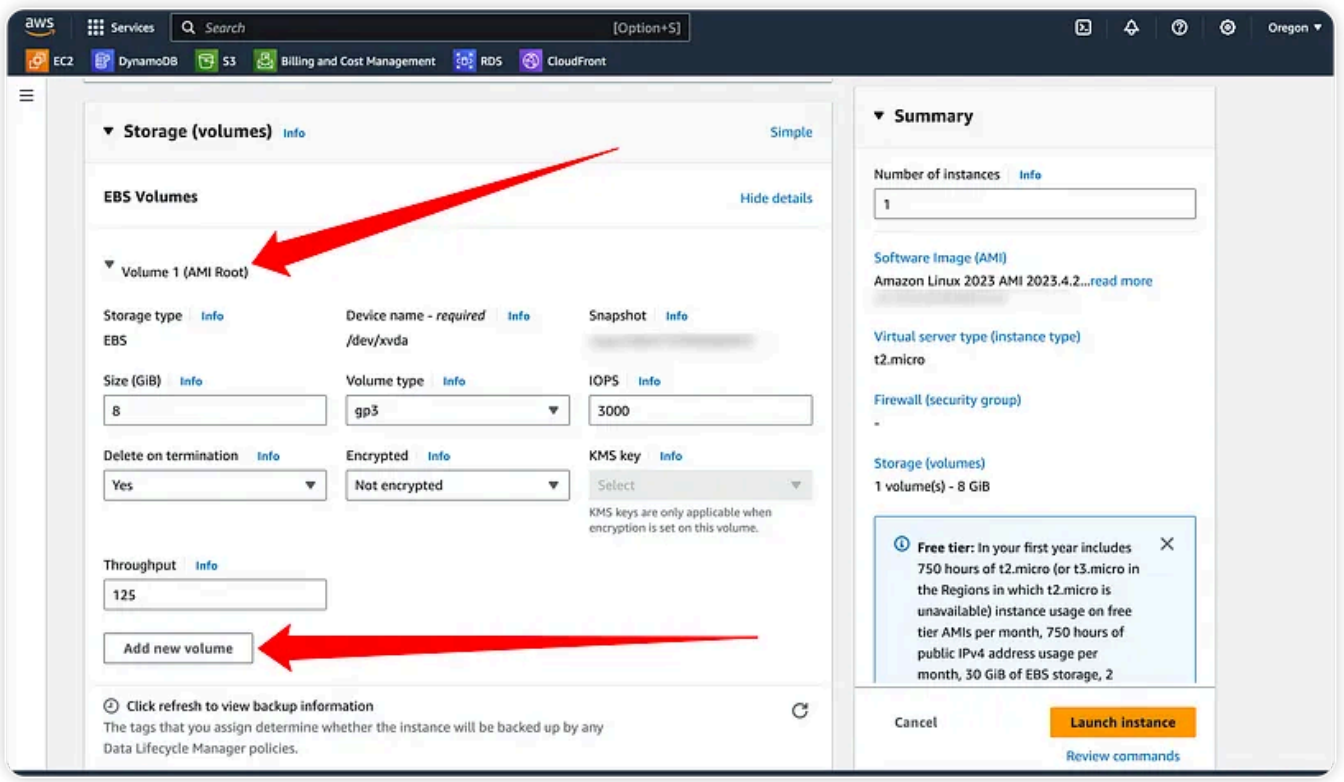
- Select the instance type that best suits your needs.

Configure Instance Details:

- Ensure the instance is in the sameAvailability Zone as the volume created from the snapshot.
- Complete the configuration and click Next: Add Storage.

Add Storage:

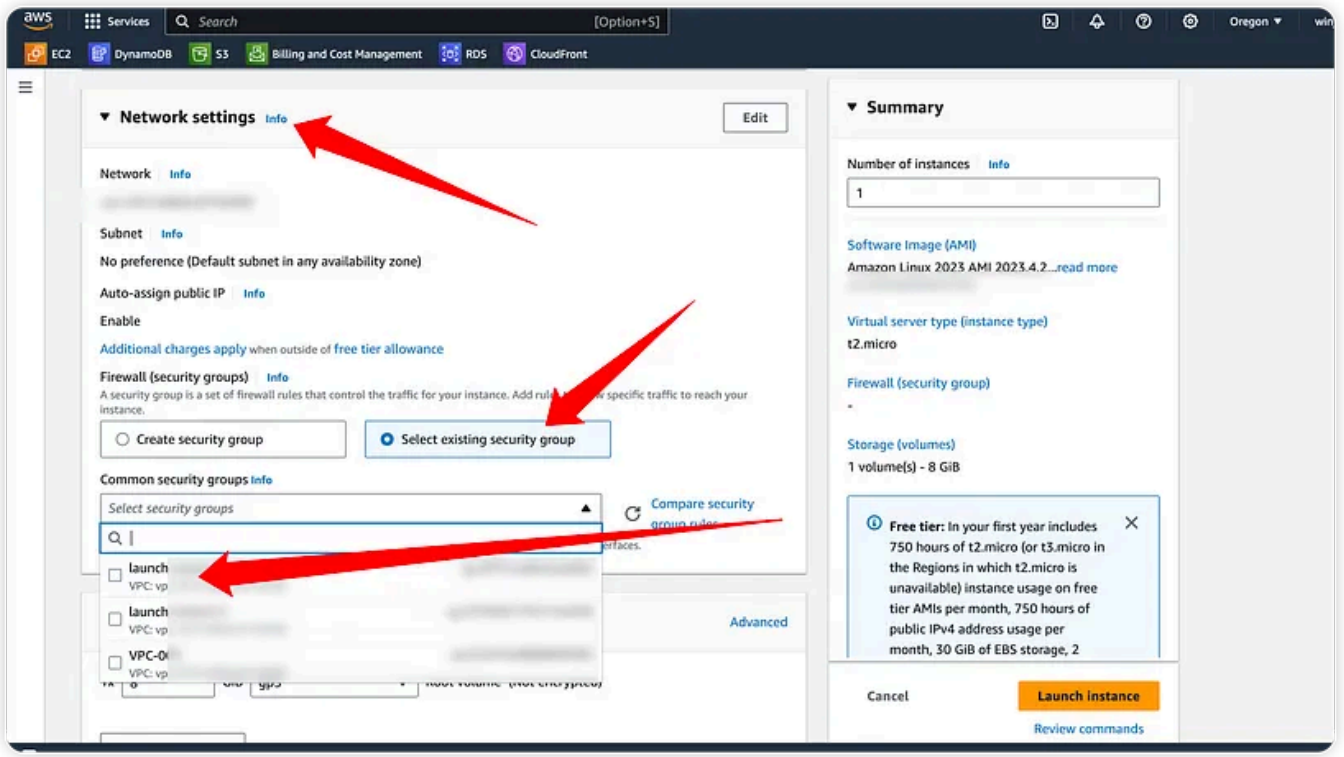
- Remove the default root volume (if possible. not possible in my case).



can’t delete the root volume, we will handle it later

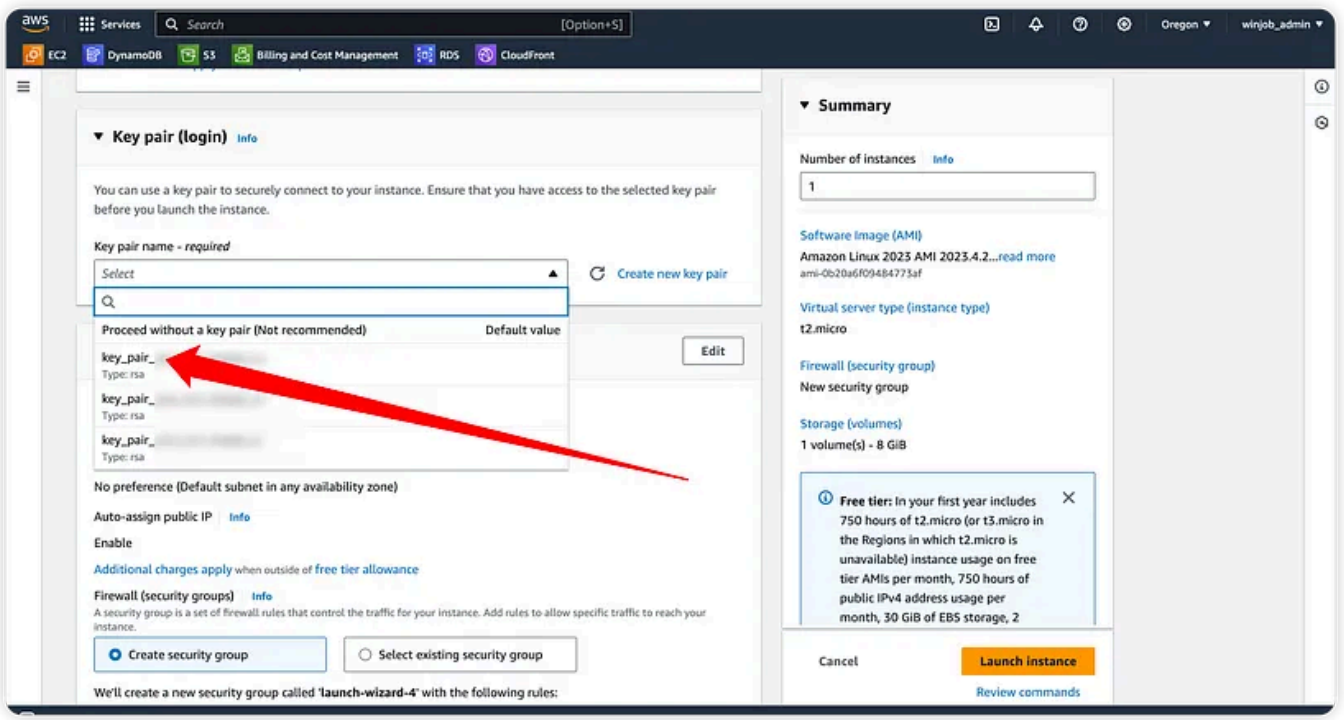
Configure Security Group:

- Select the existing security group



Key pair

- Select the old key pair

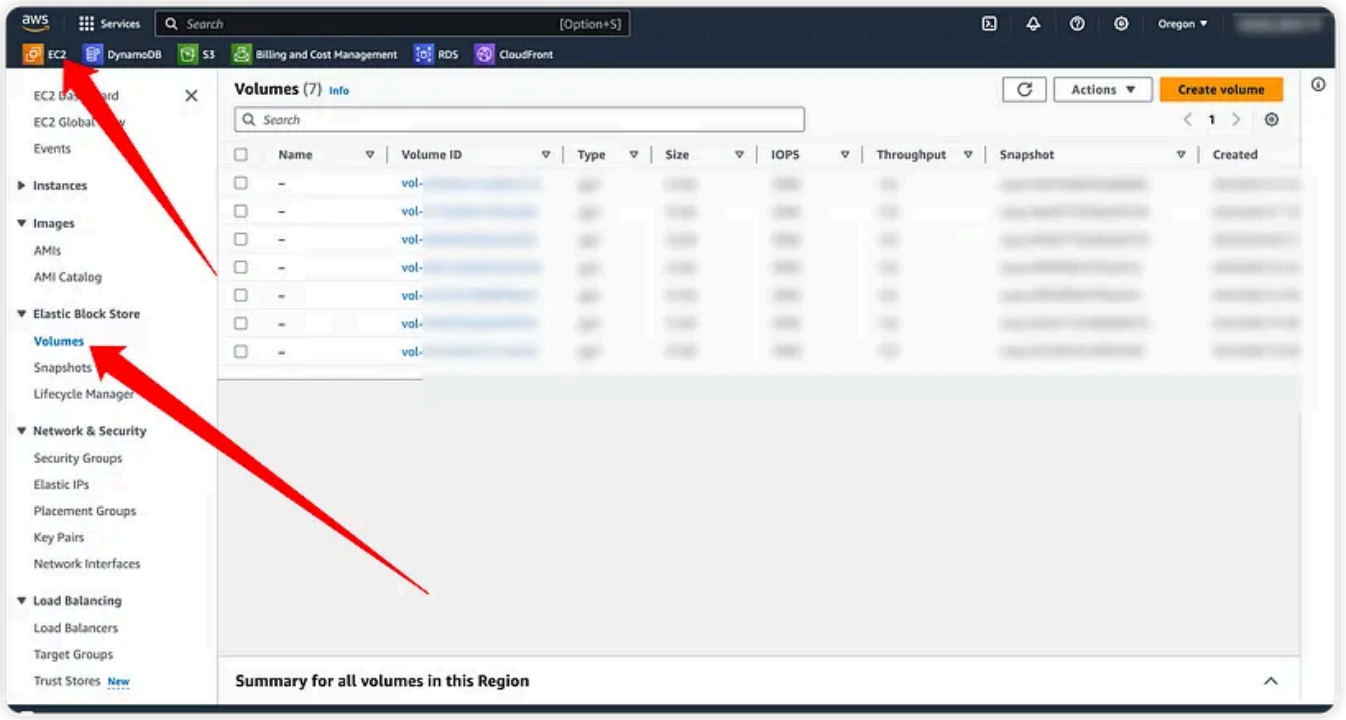


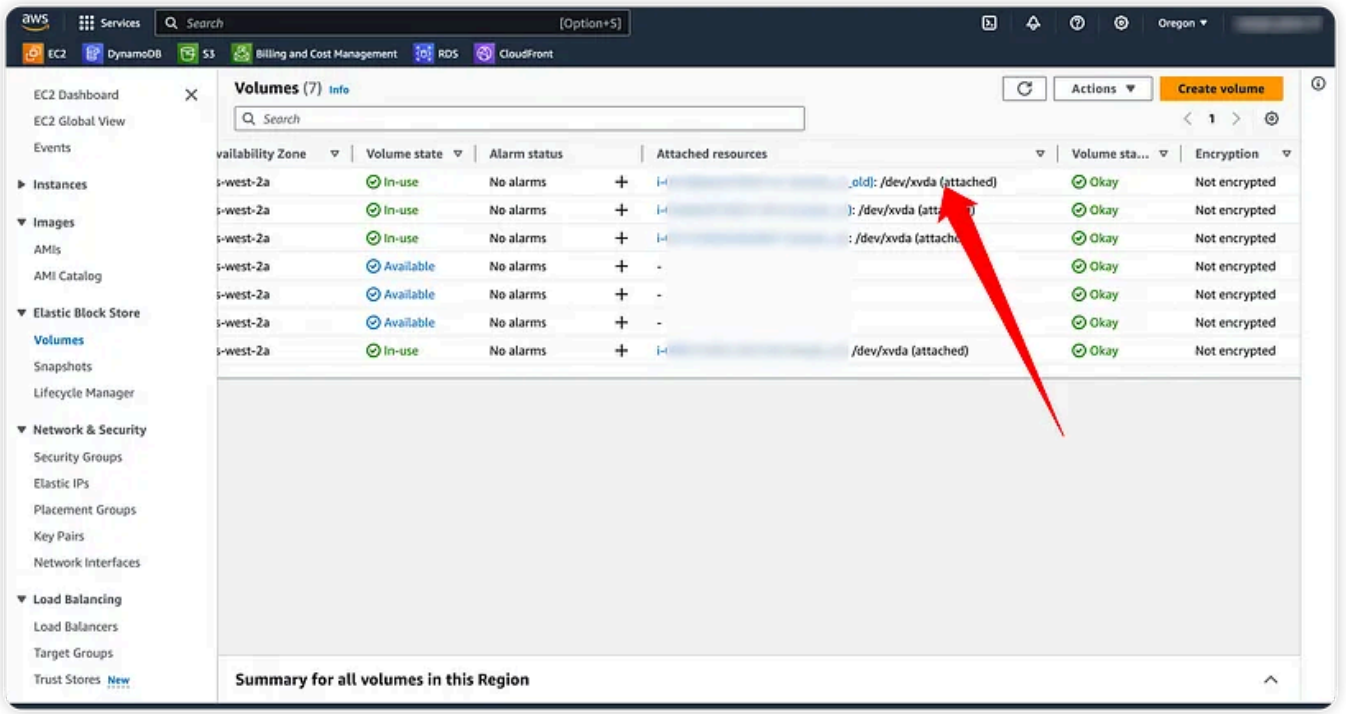
Review and Launch:

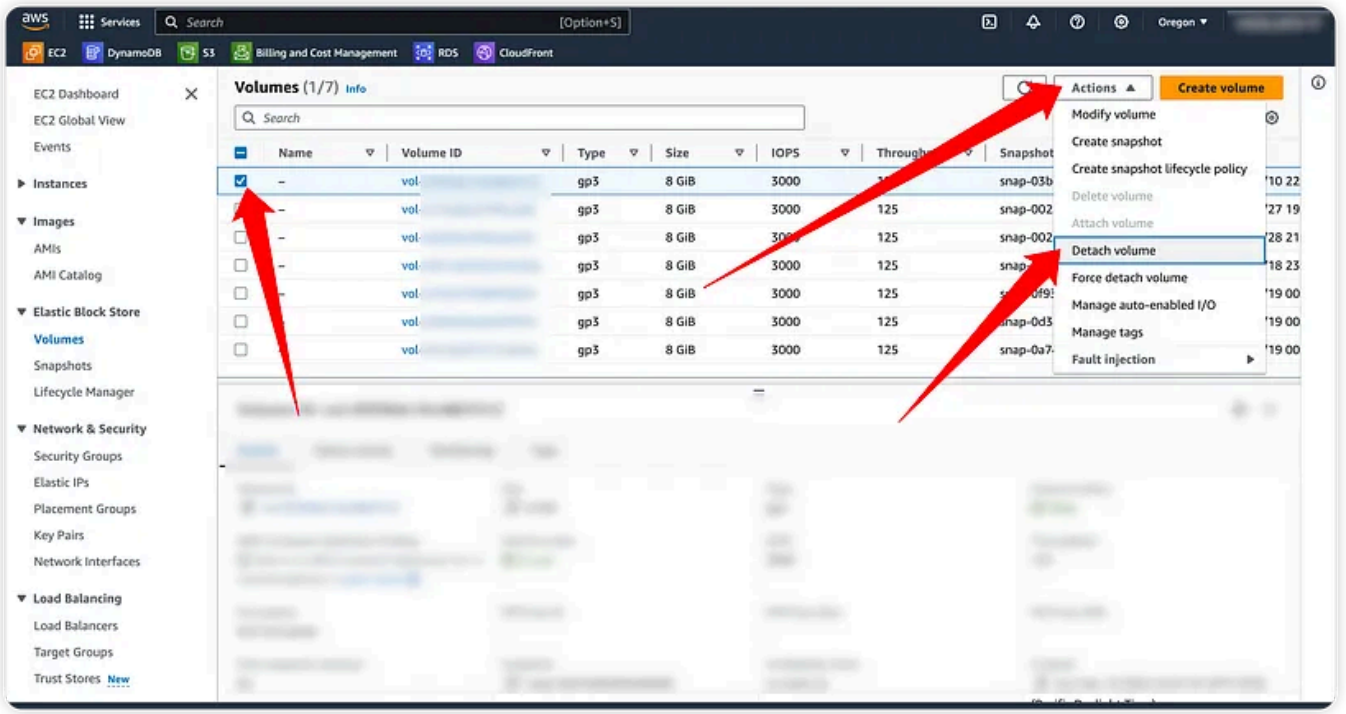
- Review your configuration and click Launch.
- Select or create a key pair and acknowledge that you have access to the selected key pair.
  Click Launch Instances.
-

Step 4: Connect the snapshot volumn to Instance

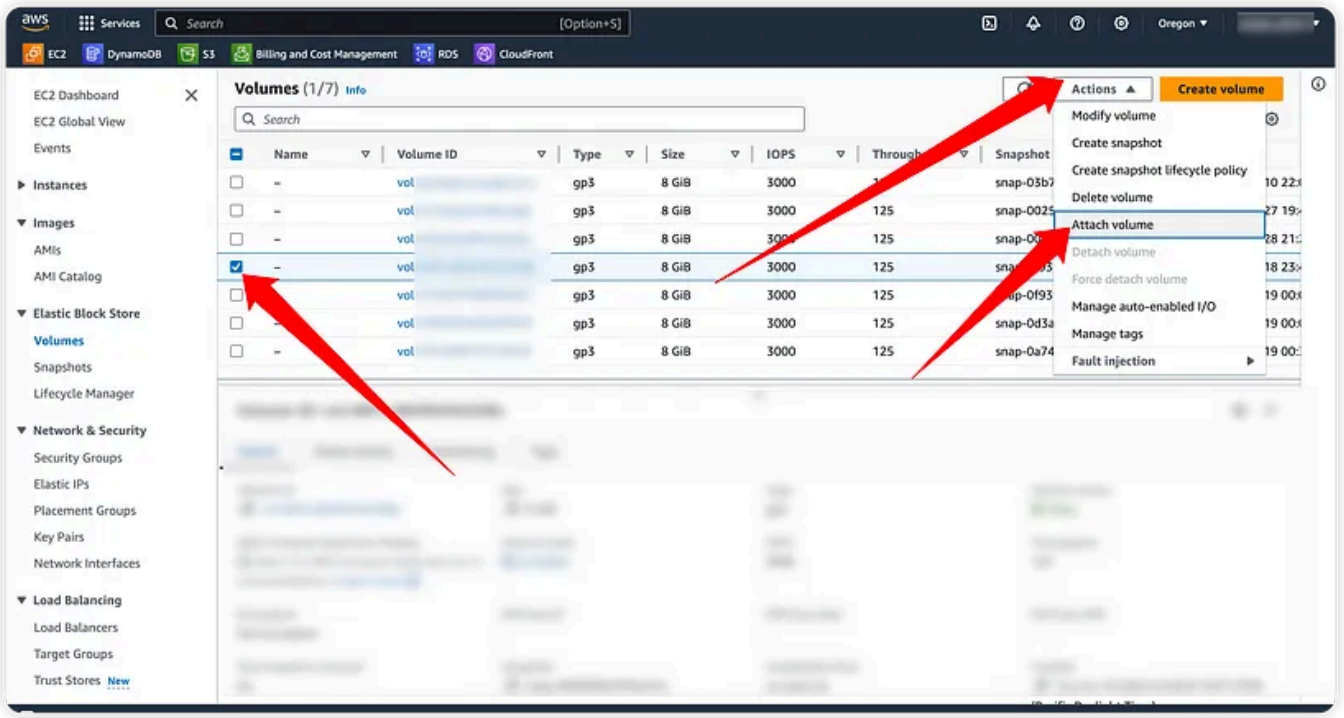
Stop the Instance and Detach the Default Root Volume

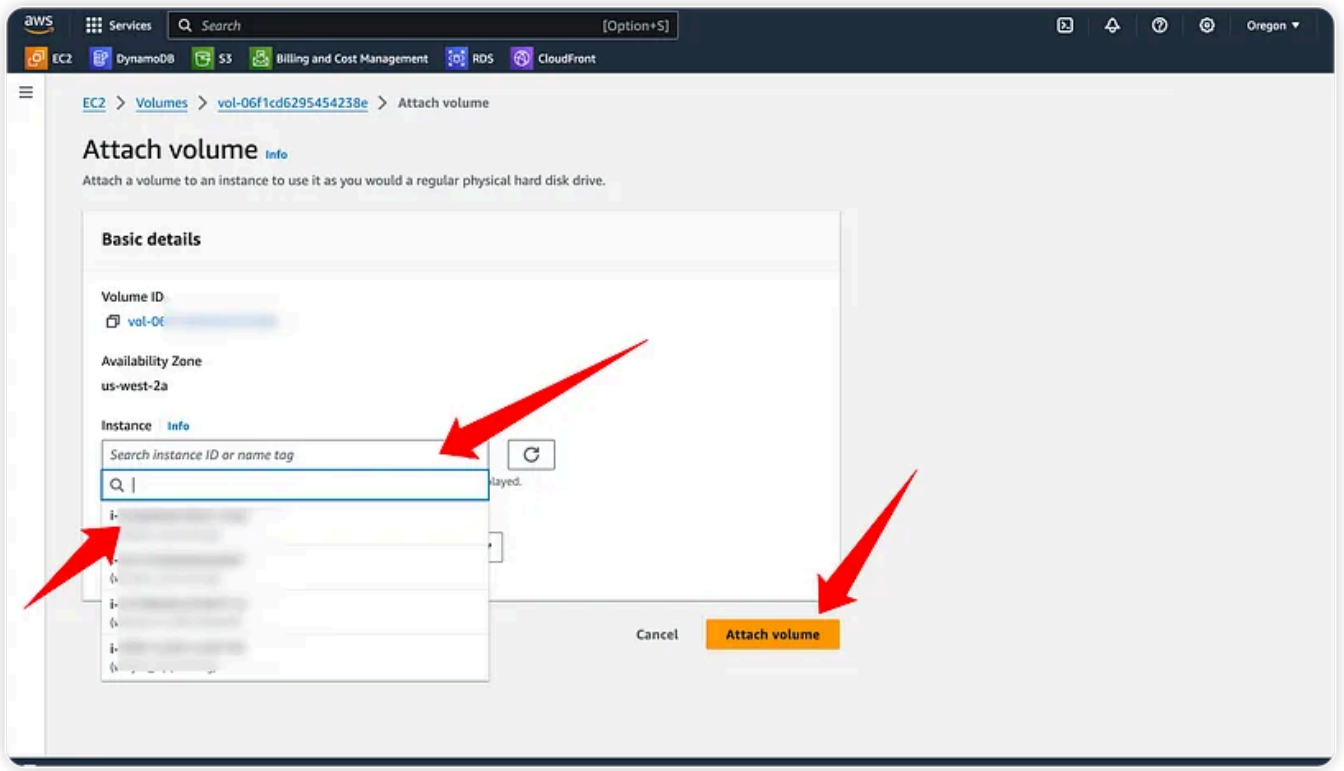






Attach the New Volume Created from the Snapshot





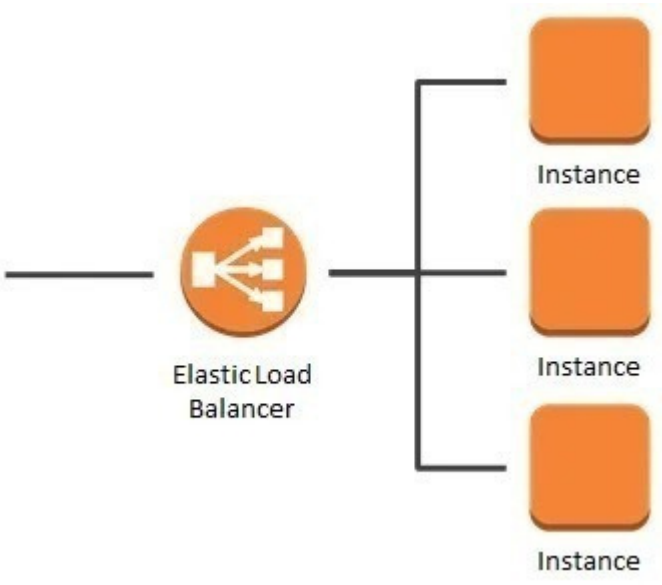
Start the instance again.

Step 5: Steps to Unassociate and Reassociate an Elastic IP

Introduction to AWS Elastic Load Balancing

Introduction to AWS Load Balancing

Meriem Terki·Follow
Published inAWS Tip· 10 min read· Jan 18, 2024



Looking to learn and practice about Elastic Load Balancing ?

incoming application traffic across multiple Amazon EC2 instances in the cloud. We will demonstrate elastic load balancing with 2 EC2 Instances.

Introduction

What is Elastic Load Balancing?

- ELB is a service that automatically distributes incoming application traffic and scale resources to meet traffic demands.
- ELB helps in adjusting capacity according to incoming application and network traffic.
- ELB can be enabled within a single availability zone or across multiple availability zones to maintain consistent application performance.

What are ELB features?

- Detection of unhealthy EC2 instances.
- Spreading EC2 instances across healthy channels only.
- Centralized management of SSL certificates.
- Optional public key authentication.
- Support for both IPv4 and IPv6.
- ELB accepts incoming traffic from clients and routes requests to its registered targets.
- When an unhealthy target or instance is detected, ELB stops routing traffic to it and resumes only when the instance is healthy again.
- ELB monitors the health of its registered targets and ensures that the traffic is routed only to healthy instances.
-

- ELB's are configured to accept incoming traffic by specifying one or more listeners. A listener is a process that checks for connection requests.
- Listeners are configured with a protocol and port number from the client to the ELB and vice-versa i.e., back from ELB to the client.

What are ELB types of load balancers ?

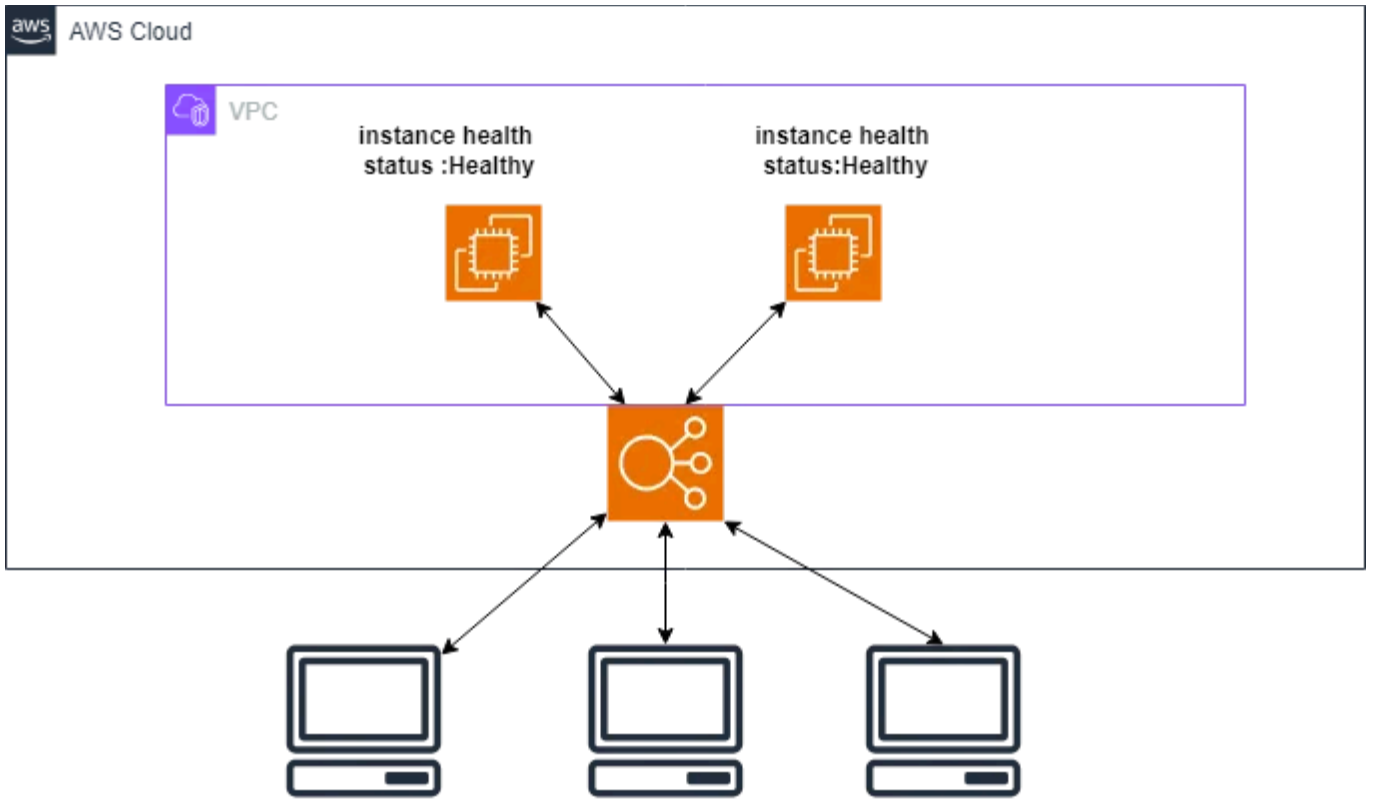
- Application Load Balancers
- Network Load Balancers
- Classic Load Balancers
- Each load balancer is configured differently.
- For Application and Network Load Balancers, you register targets in target groups and route traffic to target groups.
- For Classic Load Balancers, you register instances with the load balancer.
- AWS recommends users to work with Application Load Balancer to use multiple Availability Zones because if one availability zone fails, the load balancer can continue to route traffic to the next available one.

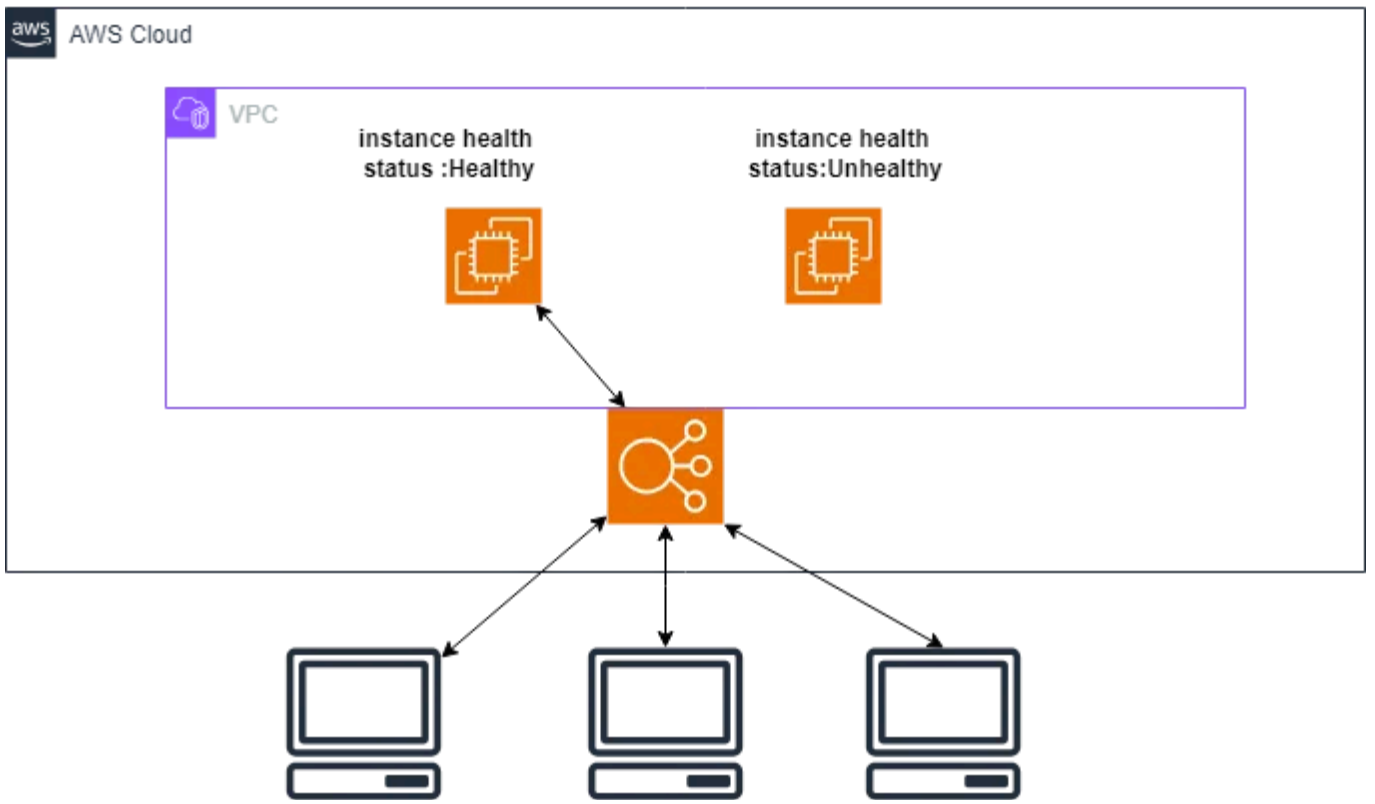
Some more features

- We can have our load balancer be either internal or internet-facing.
- The nodes of an internet-facing load balancer have Public IP addresses, and the DNS name is publicly resolvable to the Public IP addresses of the nodes.

- Due to the point above, internet-facing load balancers can route requests from clients over the Internet.
- The nodes of an internal load balancer have only Private IP addresses, and the DNS name is publicly resolvable to the Private IP addresses of the nodes.
- Due to the point above, internal load balancers can only route requests from clients with access to the VPC for the load balancer.
- Both internet-facing and internal load balancers route requests to your targets using Private IP addresses.
- Your targets do not need Public IP addresses to receive requests from an internal or an internet-facing load balancer.

Architecture Diagram





Task Steps

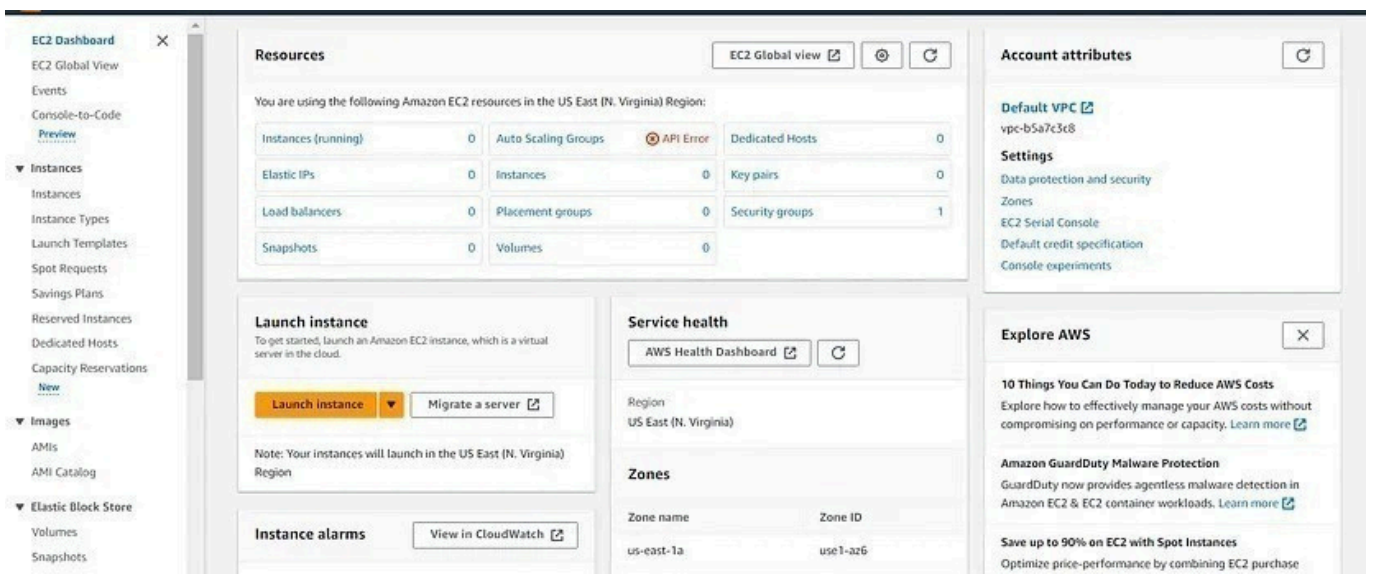
Step 1: Sign in to AWS Management Console

1. On the AWS sign-in page ,enter your credentials to log in to your AWS account and click on the Sign in button.
2. Once Signed In to the AWS Management Console, Make the default AWS Region as US East (N. Virginia) us-east-1

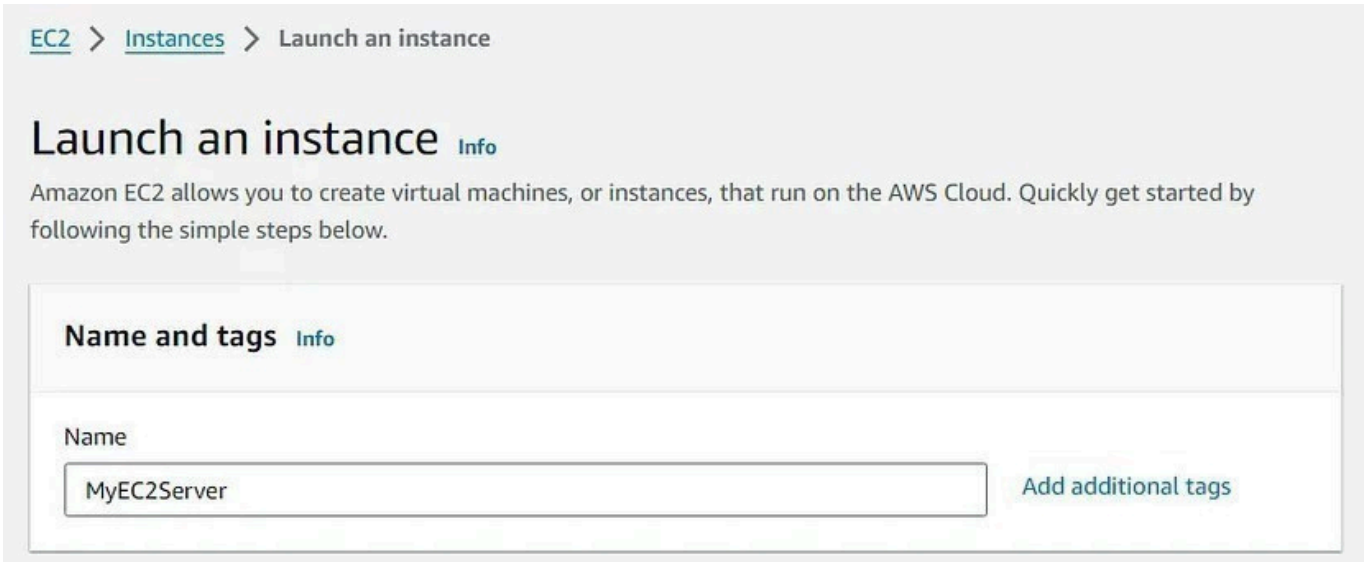
Step 2: Launching First EC2 Instance

1. Make sure you are in the US East (N. Virginia) us-east-1 Region.
2. Navigate to EC2 by clicking on the Services menu in the top, then click on EC2 in the Compute section.

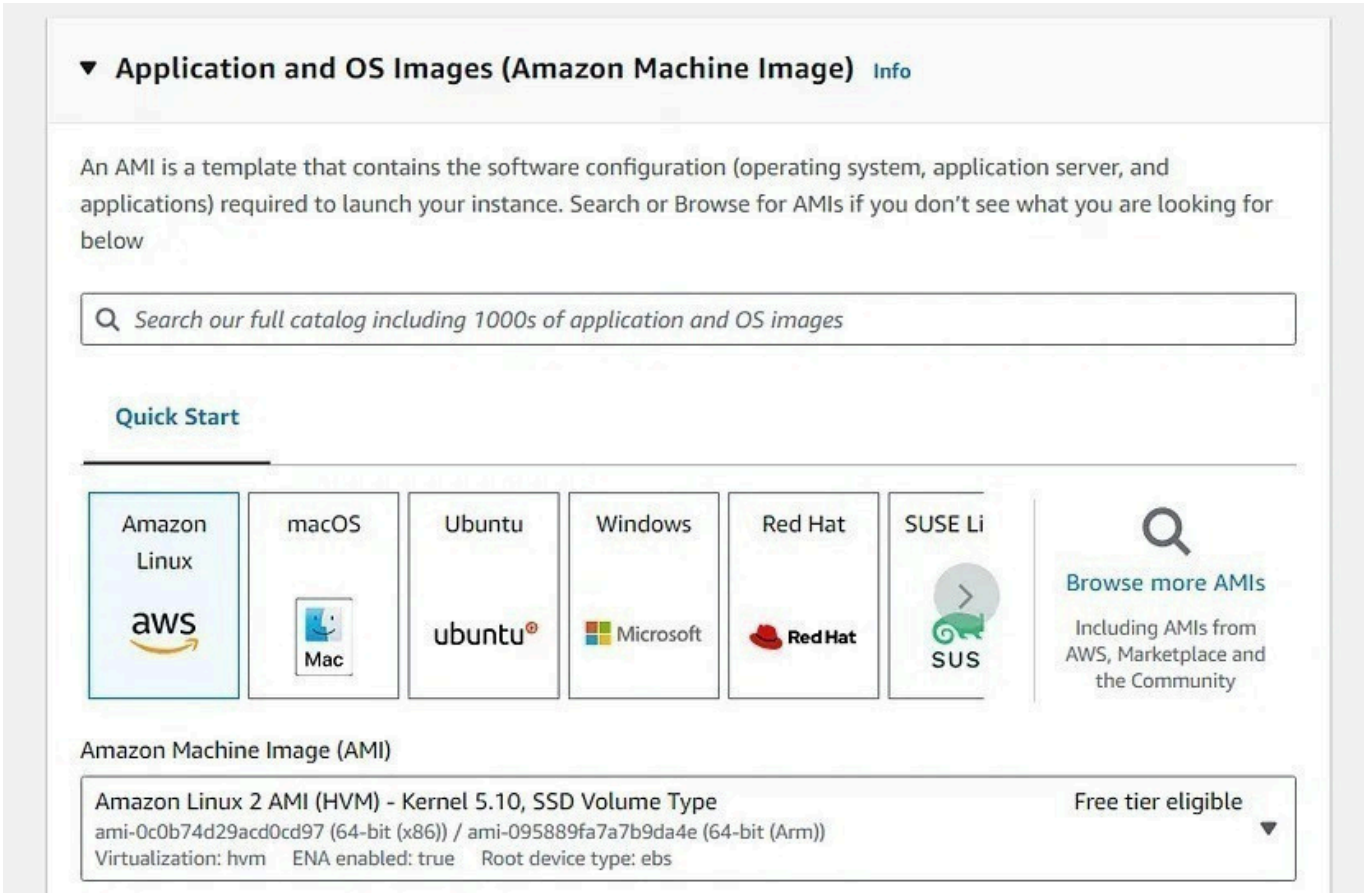
3. Navigate to Instances on the left panel and click on Launch Instances button.



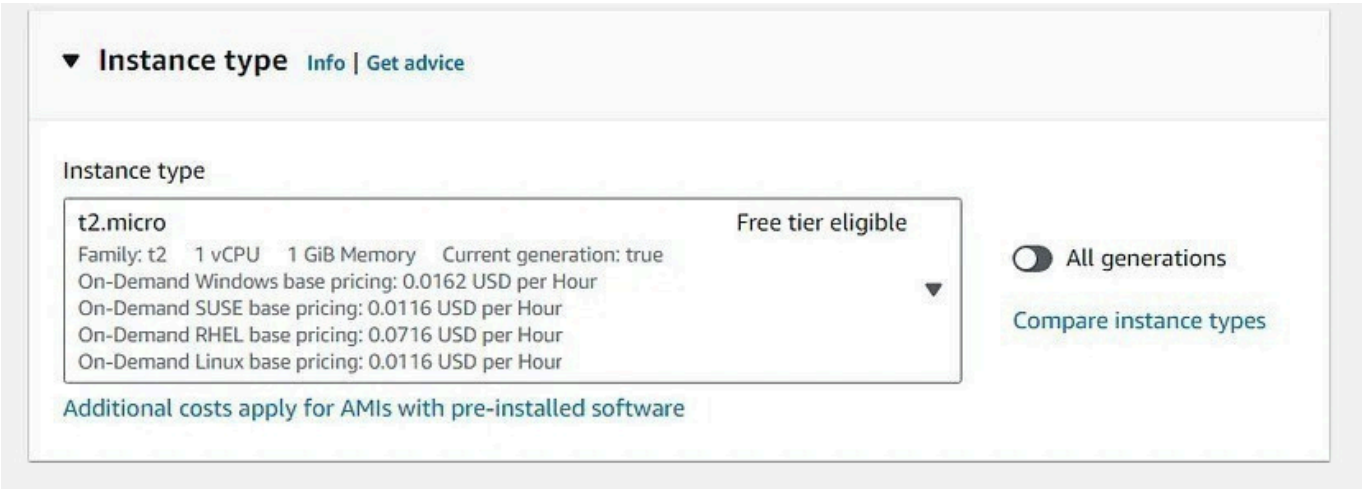
3. Name: Enter MyEC2Server



4. For Amazon Machine Image (AMI): Search for Amazon Linux 2 AMI in the search box and click on the Select button.



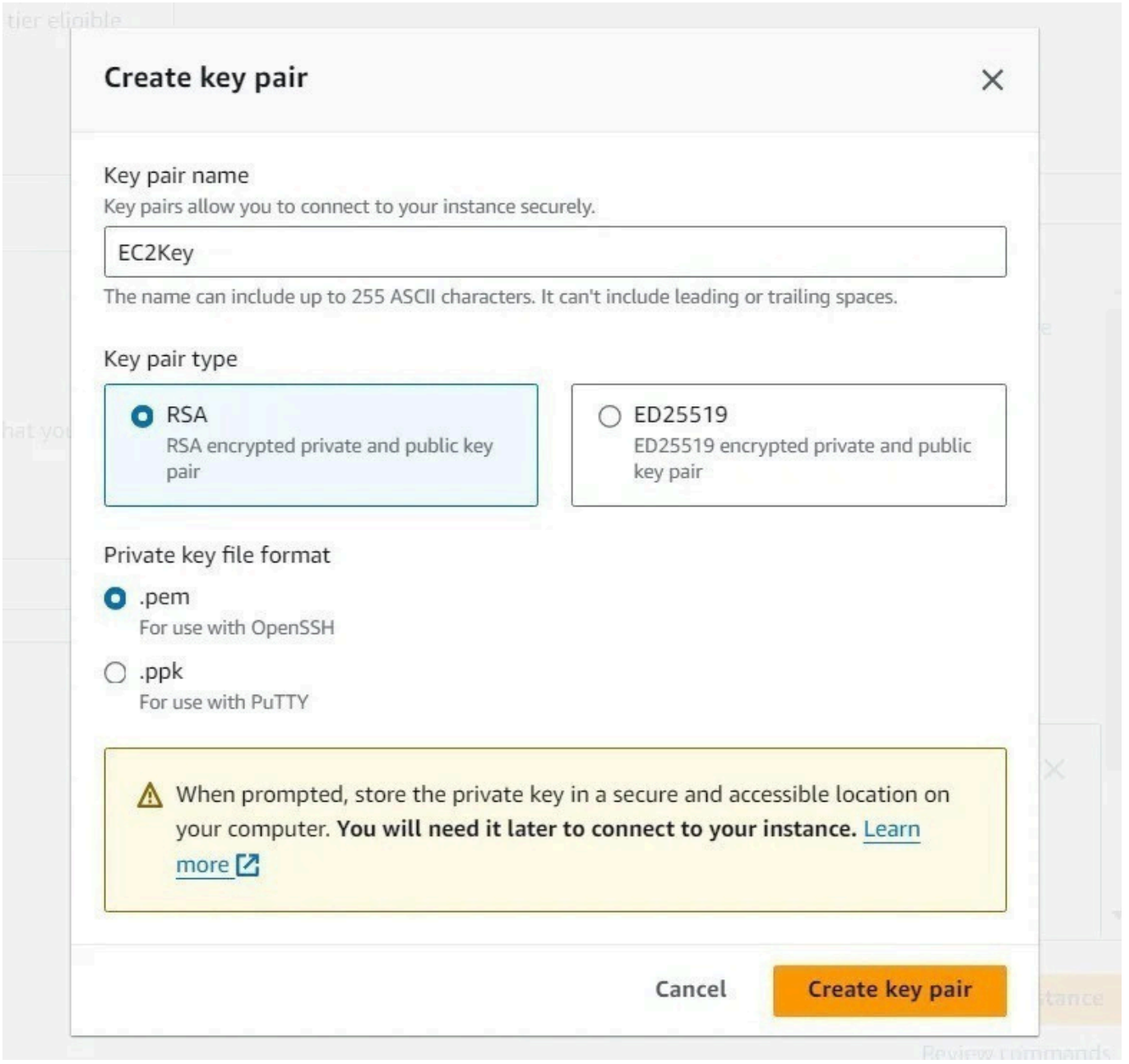
6. For Instance Type: Select t2.micro



7. For Key pair: Select Create a new key pair Button

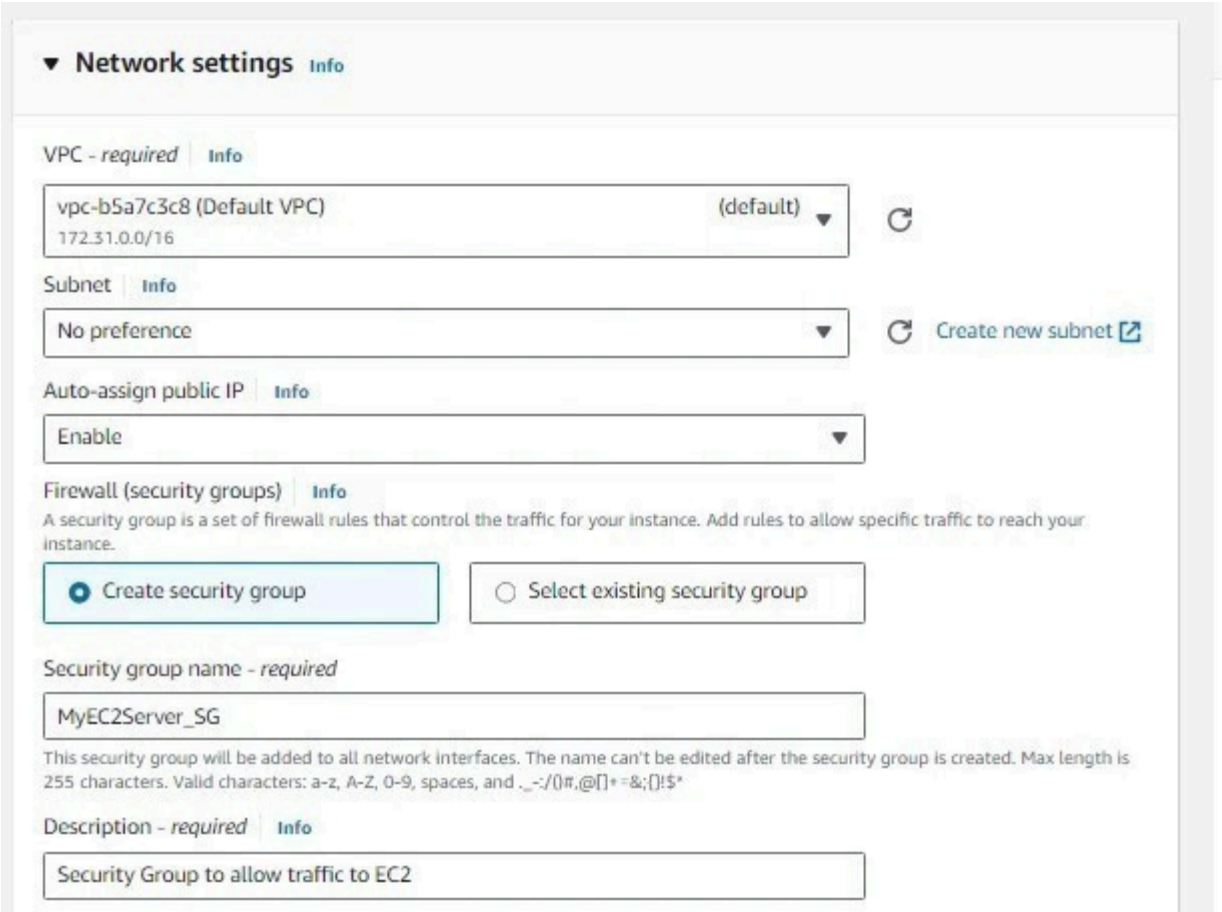
- Key pair name: EC2Key
- Key pair type: RSA
- Private key file format: .pem

8. Select Create key pair Button.



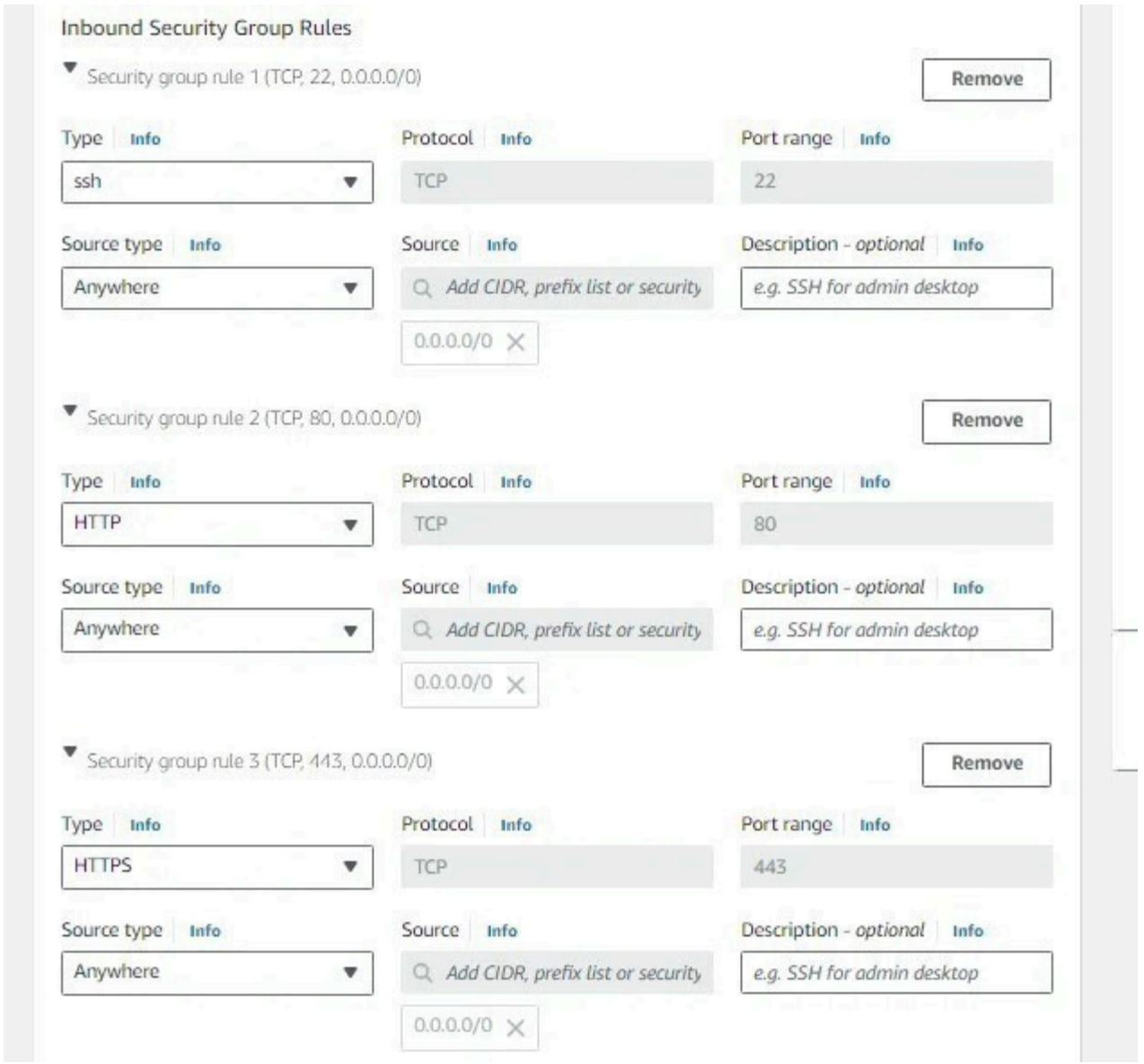
9. In Network Settings Click on Edit button:

- Auto-assign public IP: Enable
- Select Create new Security group
- Security group name : Enter MyEC2Server_SG
- Description : Enter Security Group to allow traffic to EC2



10. Check Allow SSH from and Select Anywhere from dropdown

- To add SSH,
- Choose Type: SSH
- Source: Select Anywhere
- For HTTP, Click on Add security group rule
- Choose Type: HTTP
- Source: Select Anywhere
- For HTTPS, Click on Add security group rule
- Choose Type: HTTPs
- Source: Select Anywhere



11. Click on Advanced details and under the User data: section, Enter the following script to create an HTML page served by an Apache httpd web server.

```
#!/bin/bash
sudo su
yum update -y
yum install httpd -y
echo "<html><h1> Welcome to Server 1 </h1><html>" > /var/www/html/index.html
systemctl start httpd
systemctl enable httpd
```

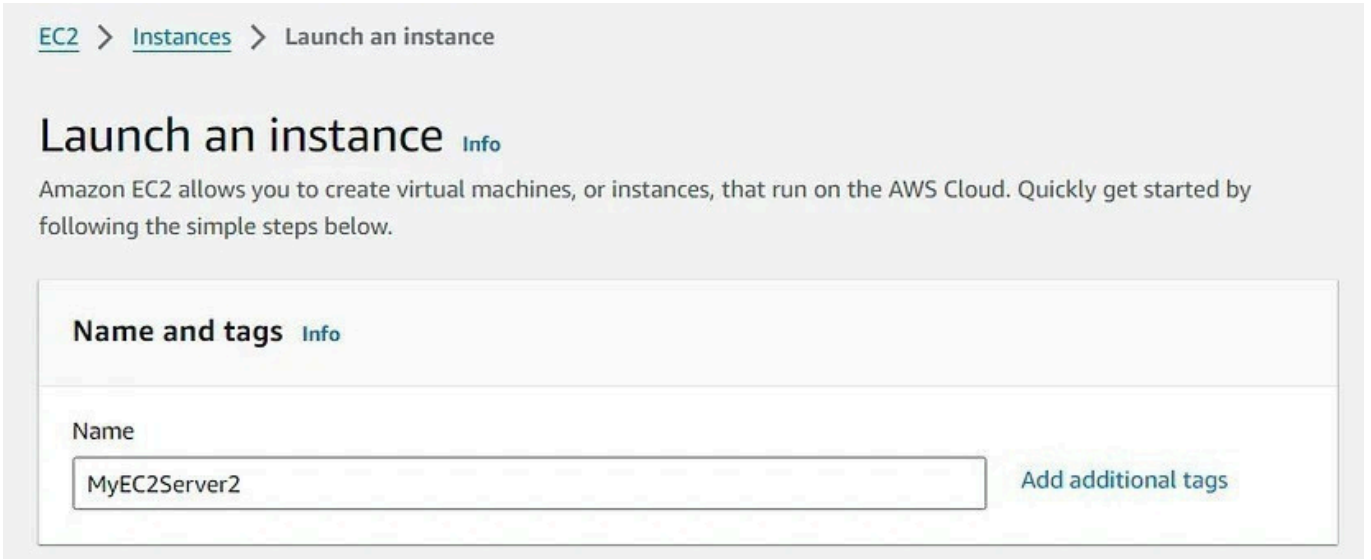
12. Keep Rest thing Default and Click on Launch Instance Button.

13. Select View all Instances to View Instance you Created.
14. Launch Status: Your instance is now launching, Click on the instance ID and wait for complete initialization of the instance till status changes to Running.

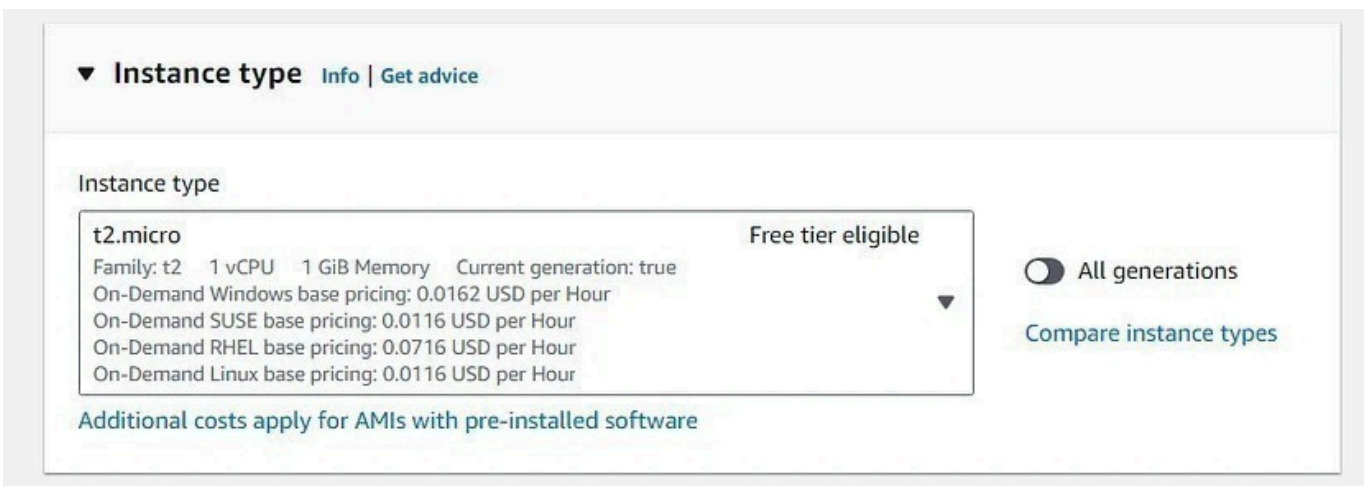
Step3 : Launching Second EC2 Instance

1. Make sure you are in the US East (N. Virginia) us-east-1 Region.
2. Navigate to EC2 by clicking on the Services menu in the top, then click on EC2 in the Compute section.
3. Navigate to Instances on the left panel and click on Launch Instances button.

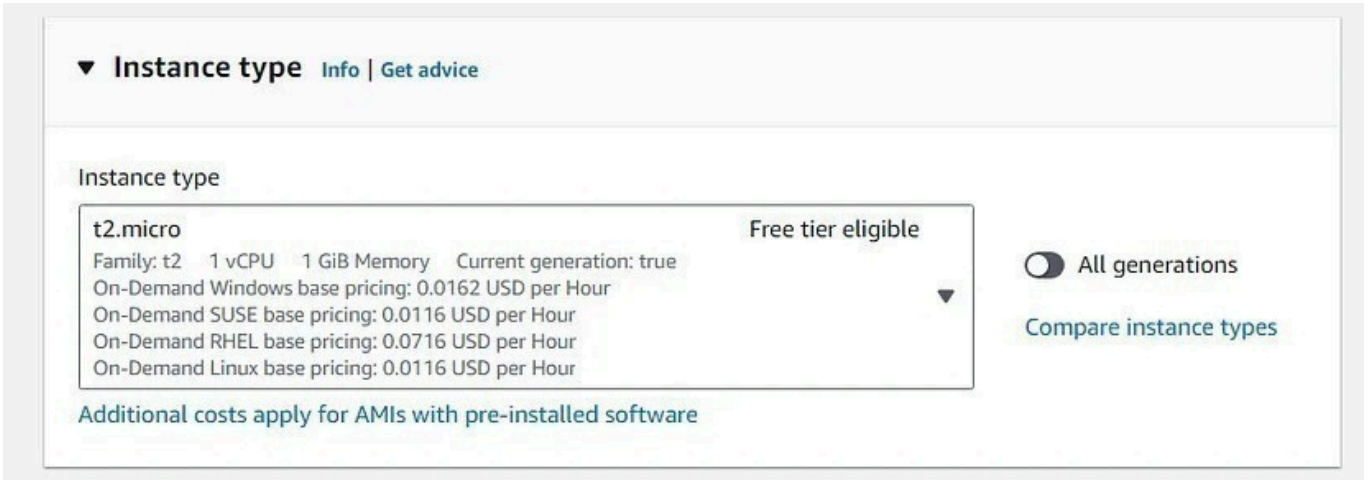
4. Name : Enter MyEC2Server2



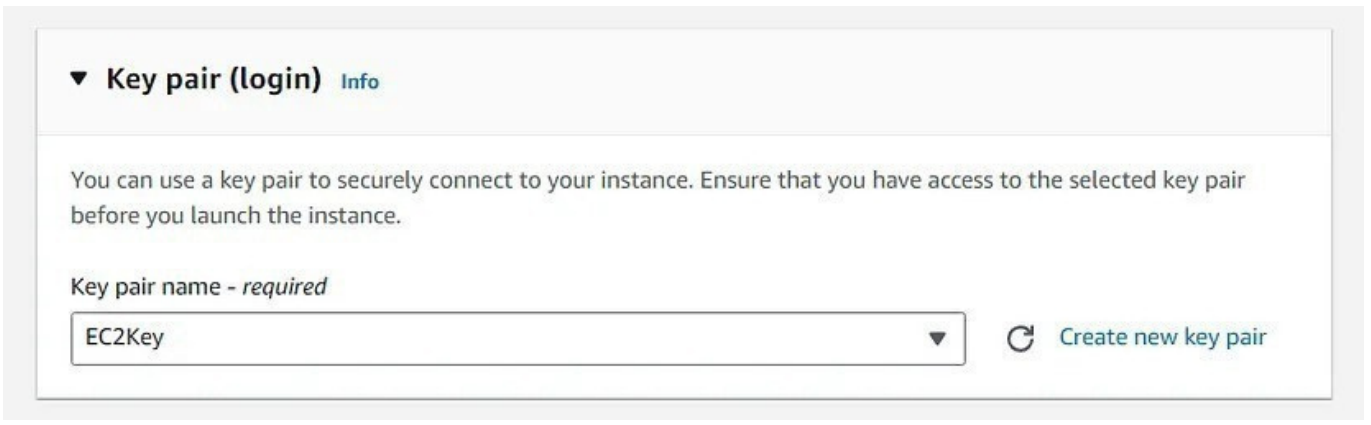
5. . For Amazon Machine Image (AMI): Search for Amazon Linux 2 AMI in the search box and click on the Select button.



6. For Instance Type: select t2.micro

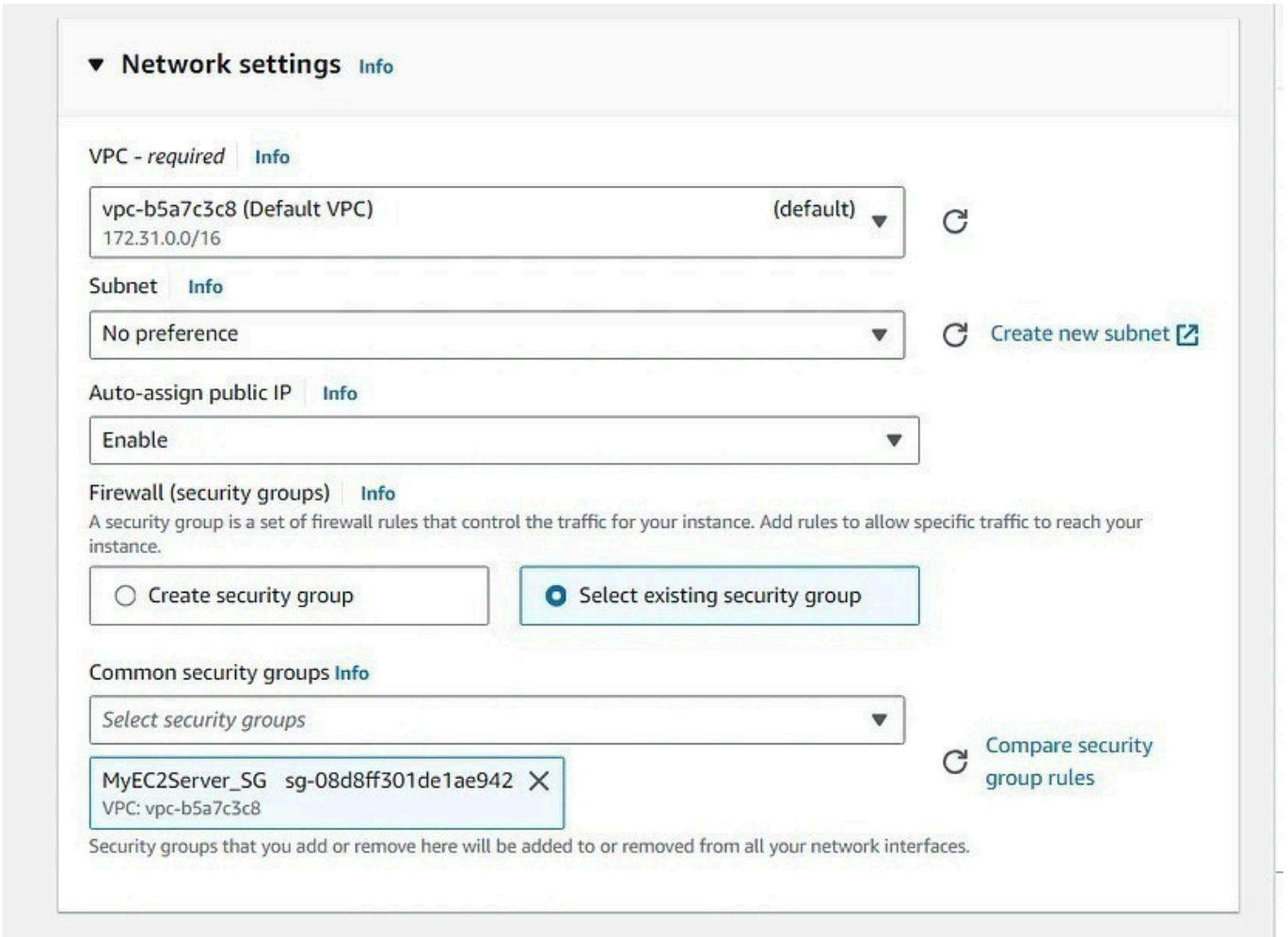


7. For Key pair: Select the key you created previously.



8. In Network Settings Click on Edit:

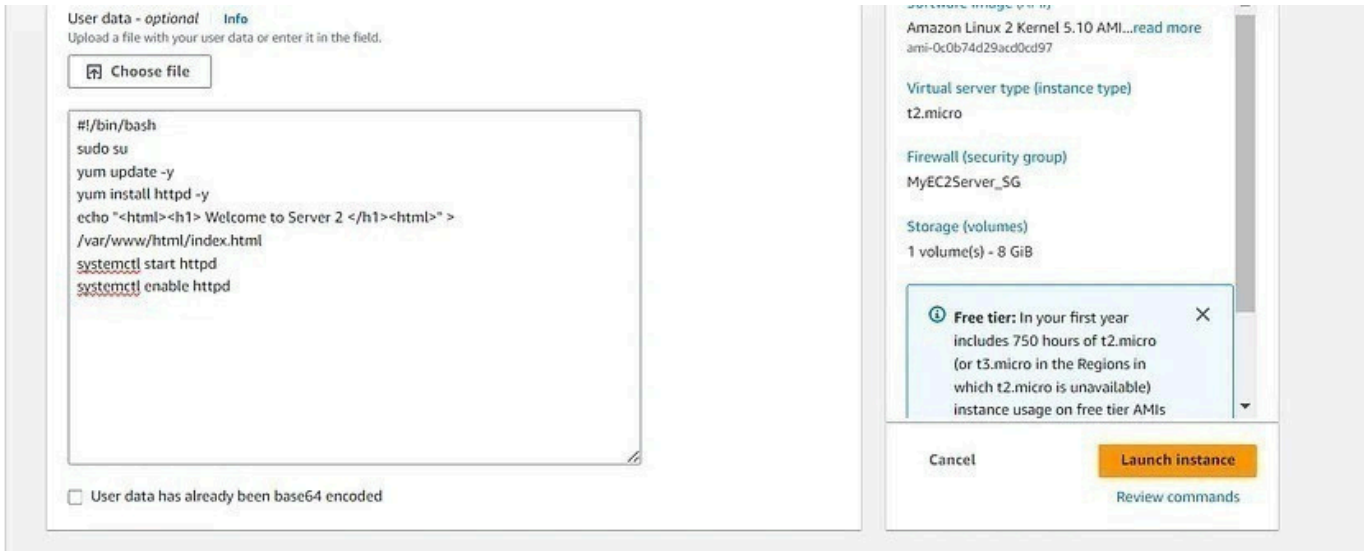
- Auto-assign public IP: Enable
- Select Select existing security group and Select the group created earlier.



9. Click on Advanced details and under the User data: section, Enter the following script to create an HTML page served by Apache httpd web server:

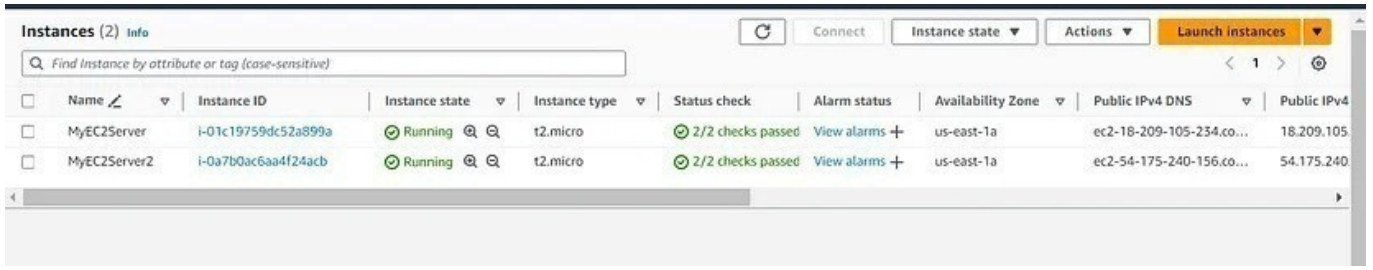
```
#!/bin/bash sudo su yum update -y yum install httpd -y echo "<html><h1> Welcome to Server 2 </h1><html>" > /var/www/html/index.html systemctl start httpd
systemctl enable httpd
```

10. Keep Rest thing Default and Click on Launch Instance Button.



11. Select View all Instances to View Instance you Created

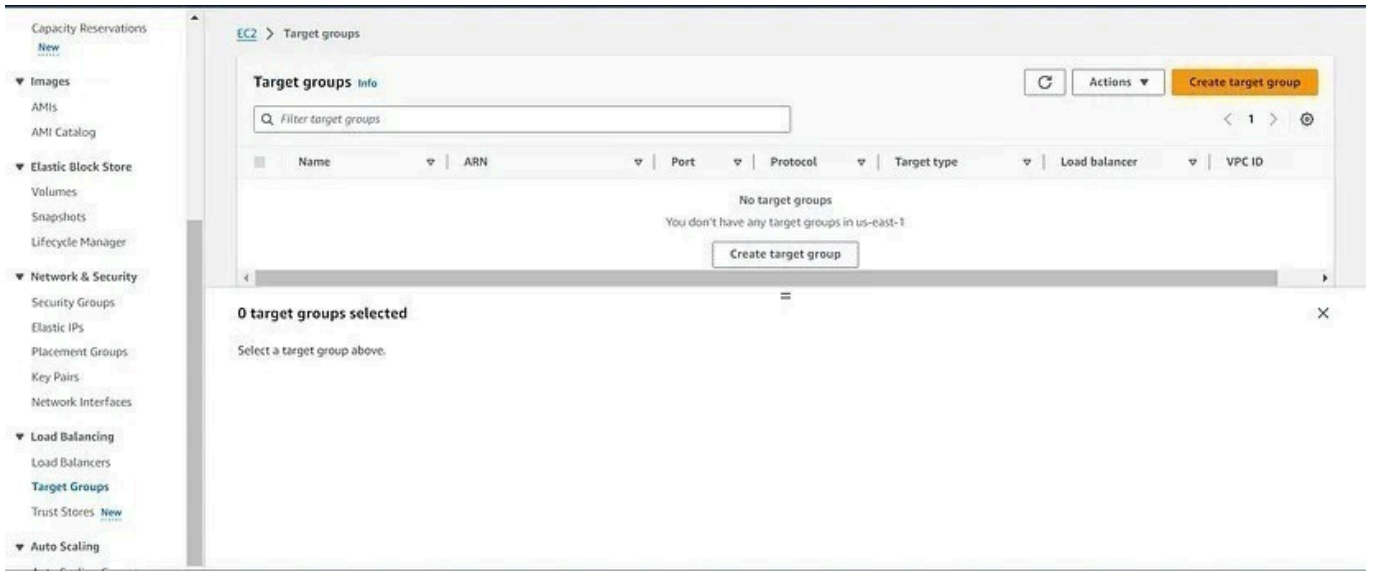
12. Launch Status: Your instance is now launching. Click on View Instances. In the dashboard find your instance and wait for complete initialization of the instance until the instance state changes to running.



Step 4: Creating the Target Group and Load Balancer

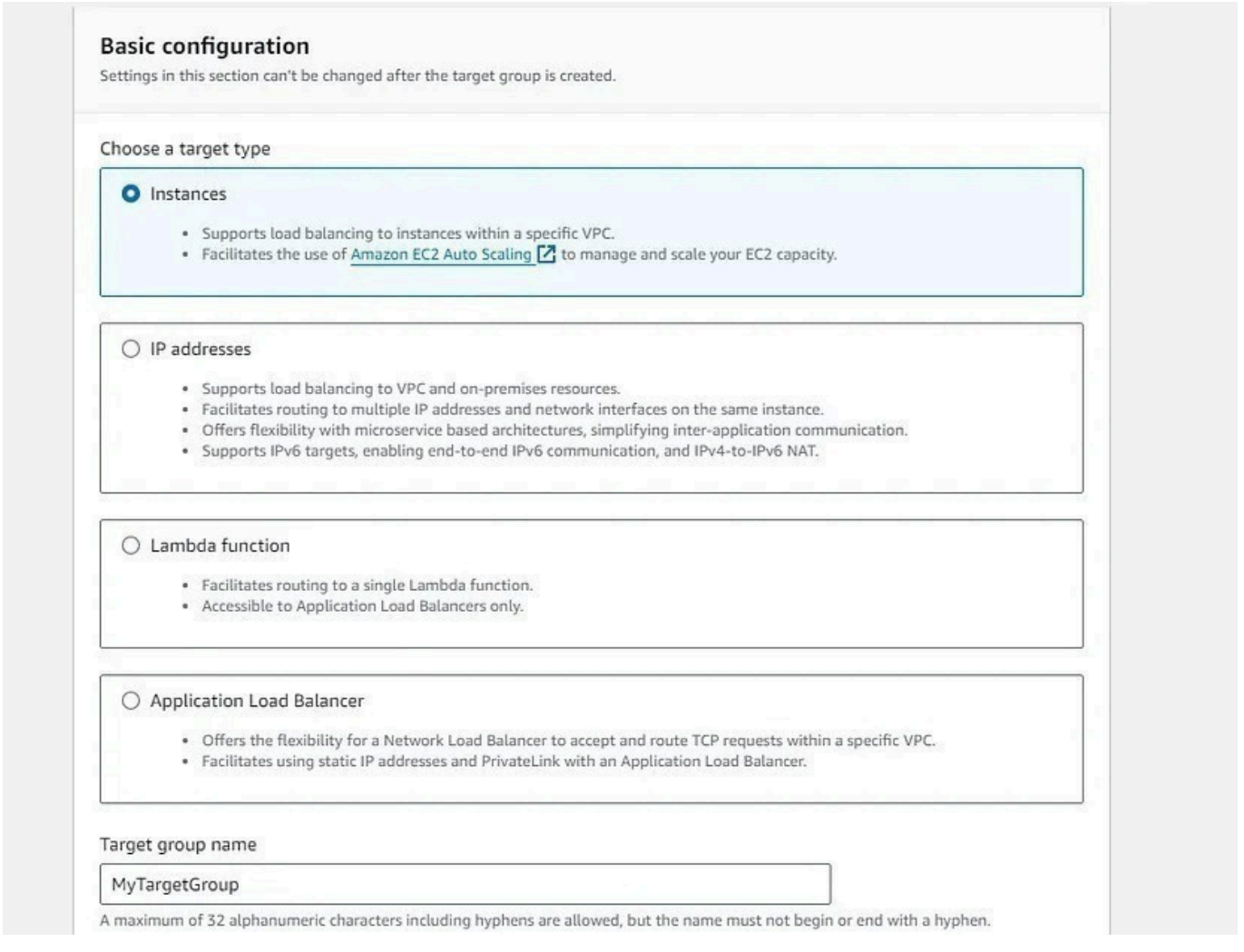
1. In the EC2 Console, Navigate to Target Groups, present in the left panel under Load Balancing.

2. Click on the Create target group button.



3. For Step 1, Specify group details

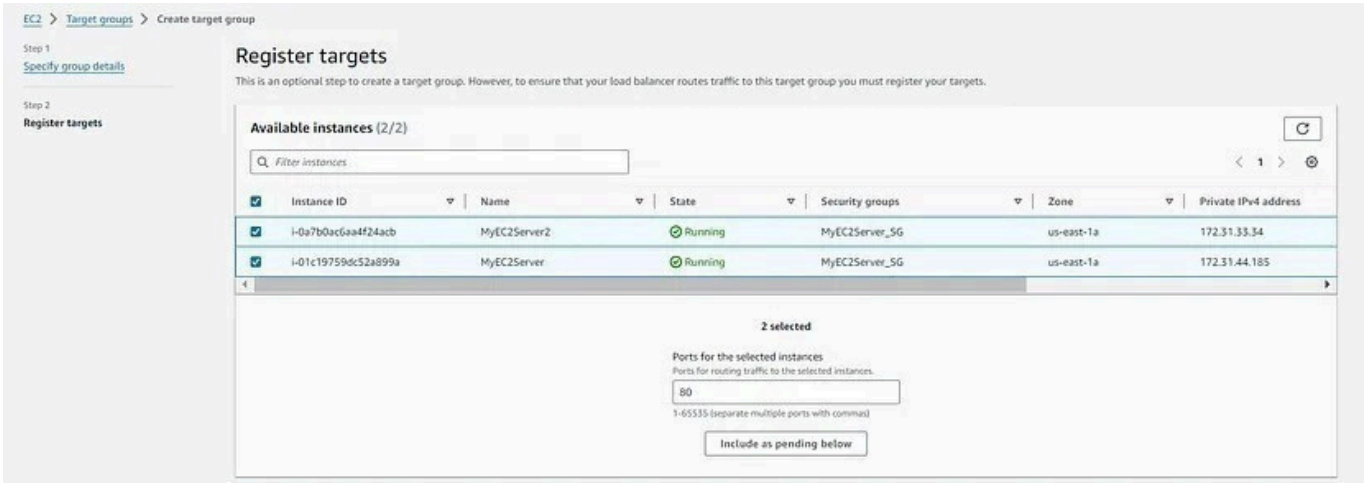
- Under Basic configurations,
- Choose a target group: Choose Instances
- Target group name: Enter MyTargetGroup



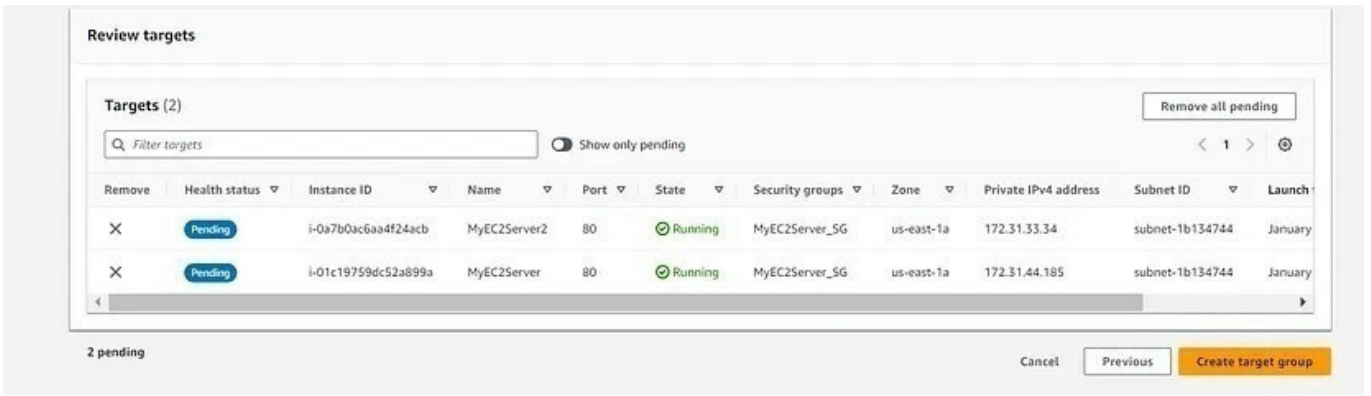
- Keep all the settings as default.
- Scroll to the end of the page and click on the Next button.

4. For Step 2, Register targets

- Select both instances and click on the Include as pending below button.



- Instances will be present in the Review targets part, having health status as Pending.

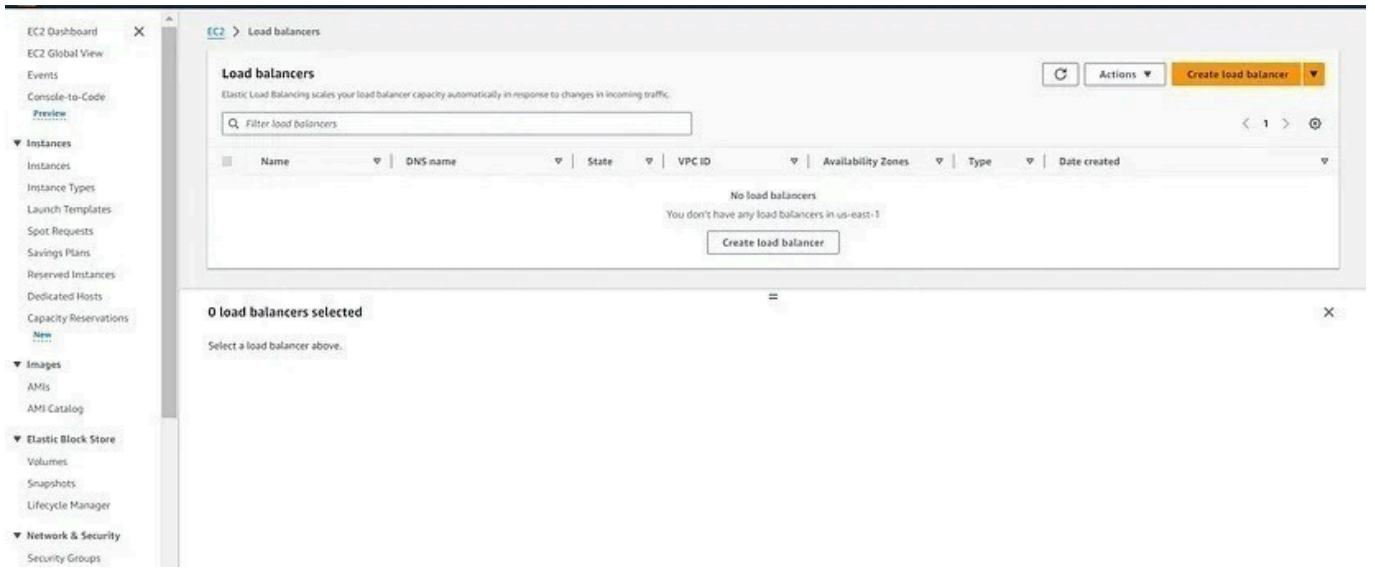


- Click on the Create target group button.

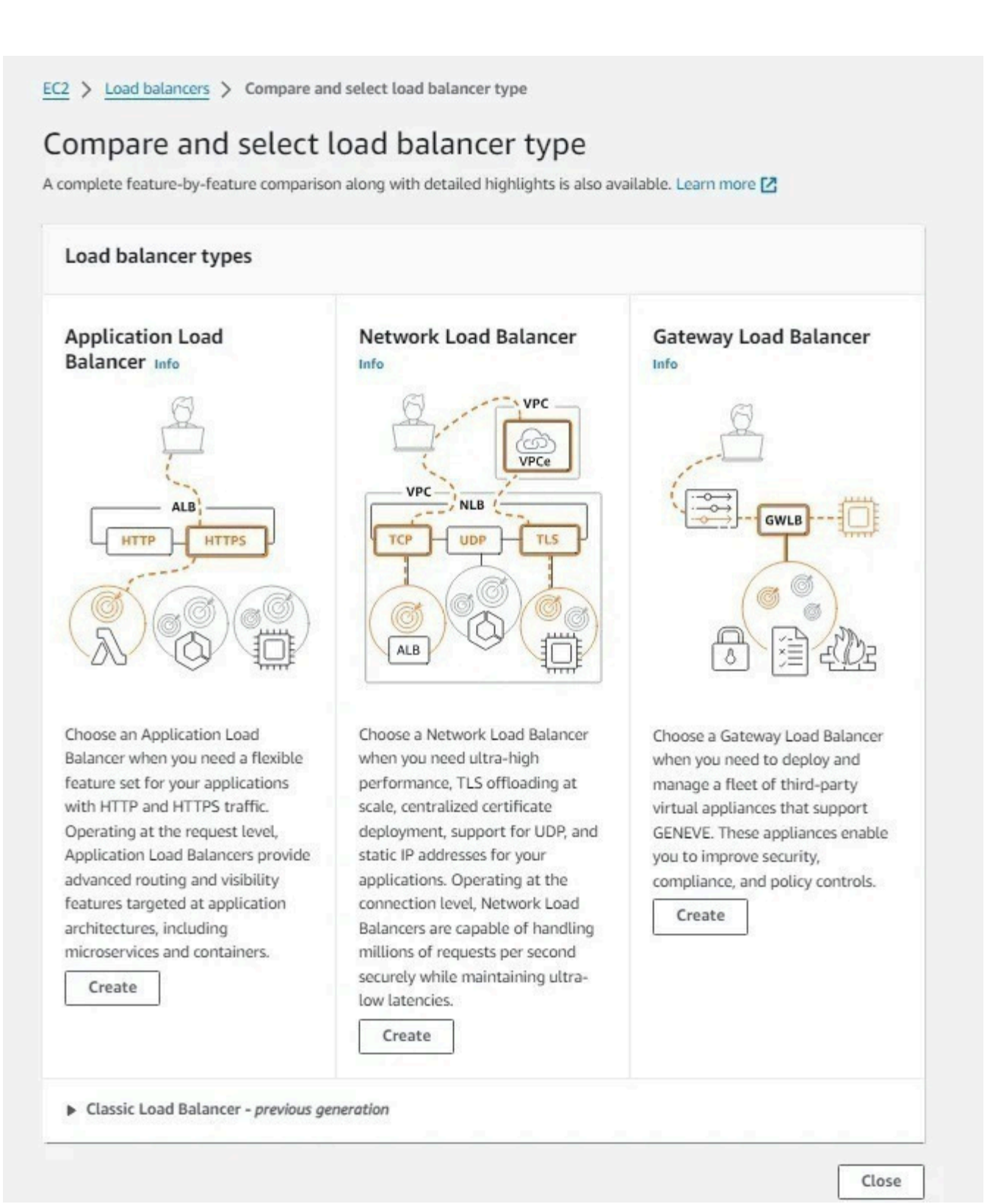
5. The Target group is now created.

6. In the EC2 console, navigate to Load balancers in the left-side panel.

7. Click on Create Load Balancer button at the top-left to create a new load balancer for our web servers.



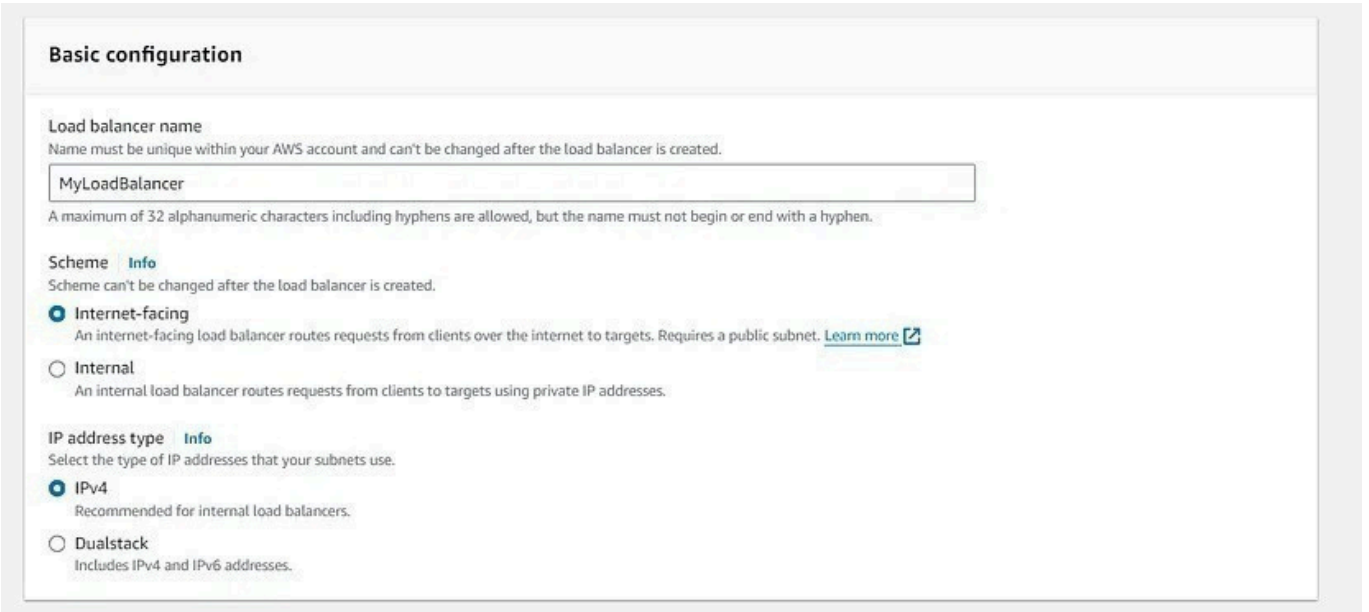
8. Select Load Balancer Type: Under the Application load balancer, click on Create button



9. To create an Application load balancer, configuring the load balancer as below

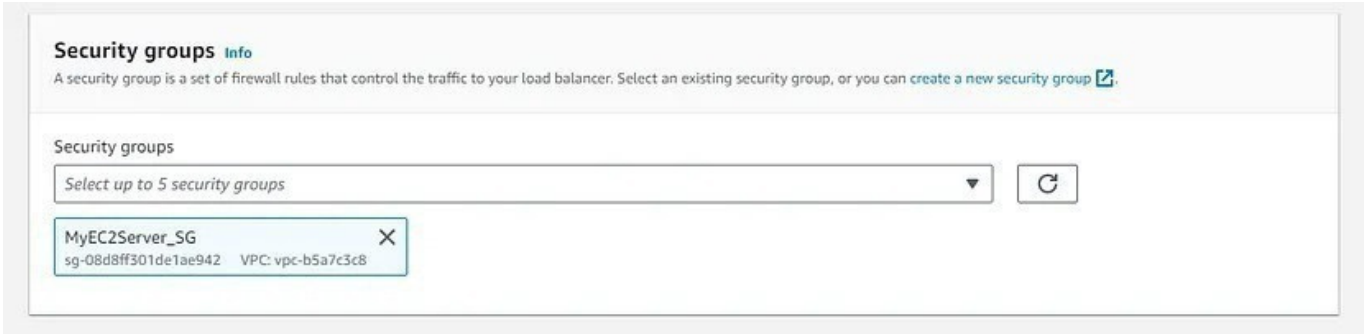
- For the Basic configuration section,

- Name: Enter MyLoadBalancer
- Scheme: Select Internet-facing
- IP address type: Choose IPv4

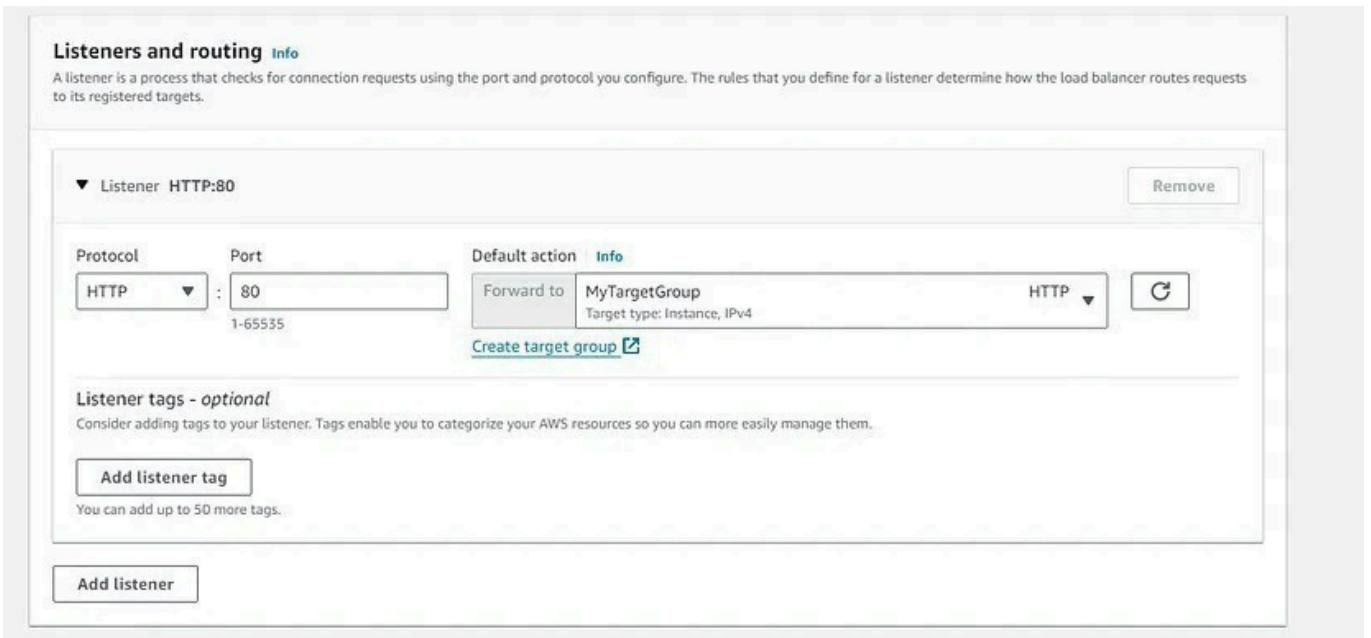


- For the Network mapping section:
- VPC: Select Default
- Mappings: Select all the Availability zone present

- For the Security groups section,
- Select the MyEC2Server_SG Security group from the dropdown and remove the default security group.

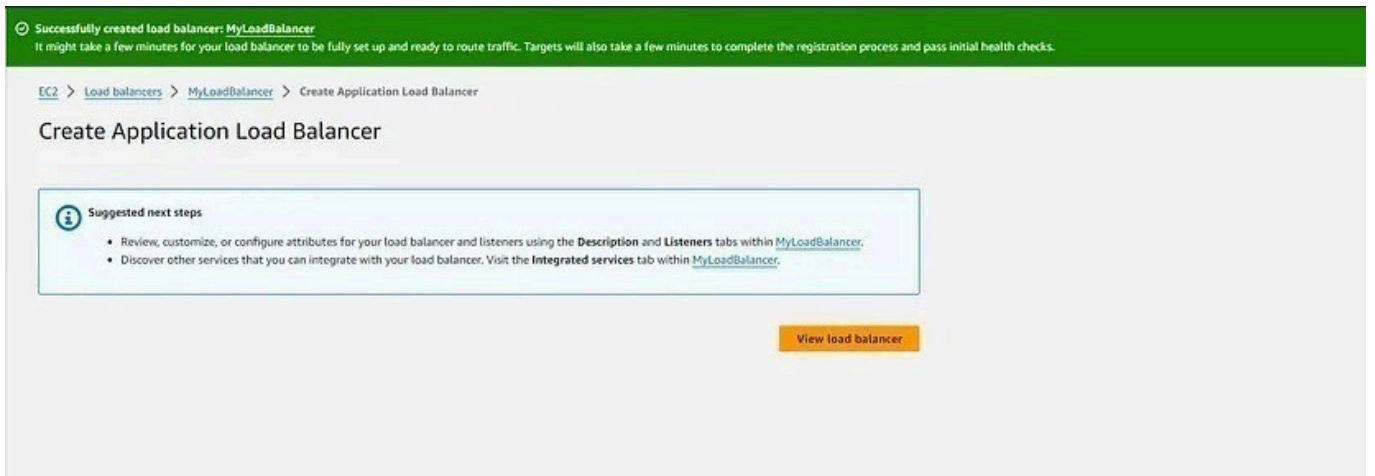


- For the Listeners and routing section,
- The listener is already present with Protocol HTTP and Port 80.
- Select the target group MyTargetGroup for the Default action forwards to option.

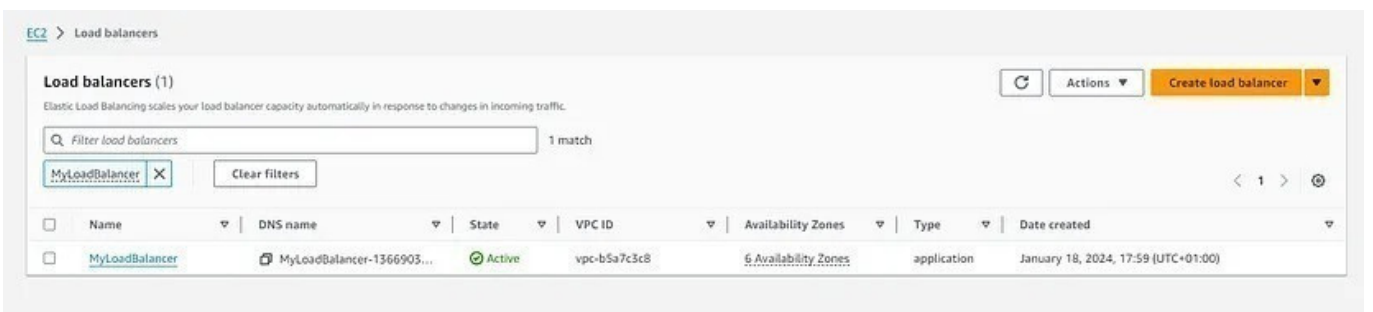


10. Keep the tags as default and click on the Create load balancer button.

11. You have successfully created the Application Load balancer. Click on the View load balancers button.

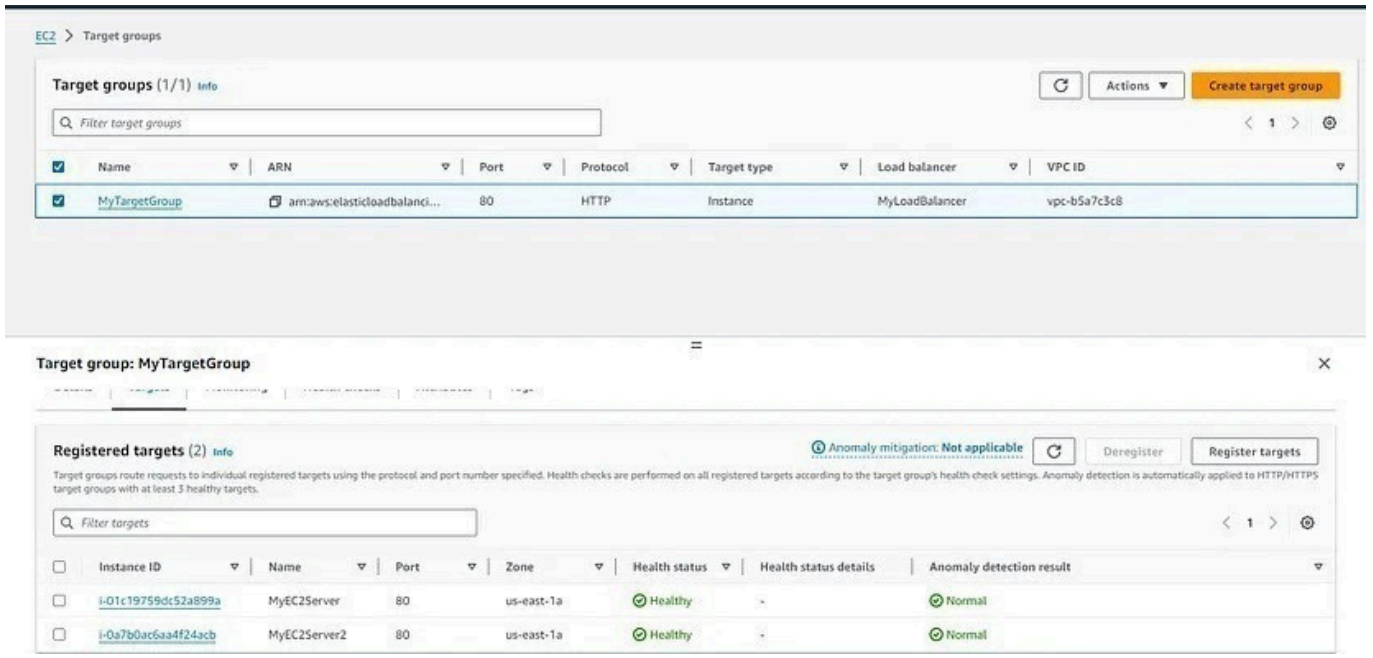


12. Wait for 2 to 3 minutes for the load balancer to become Active.



Step 5: Testing the Elastic Load Balancer

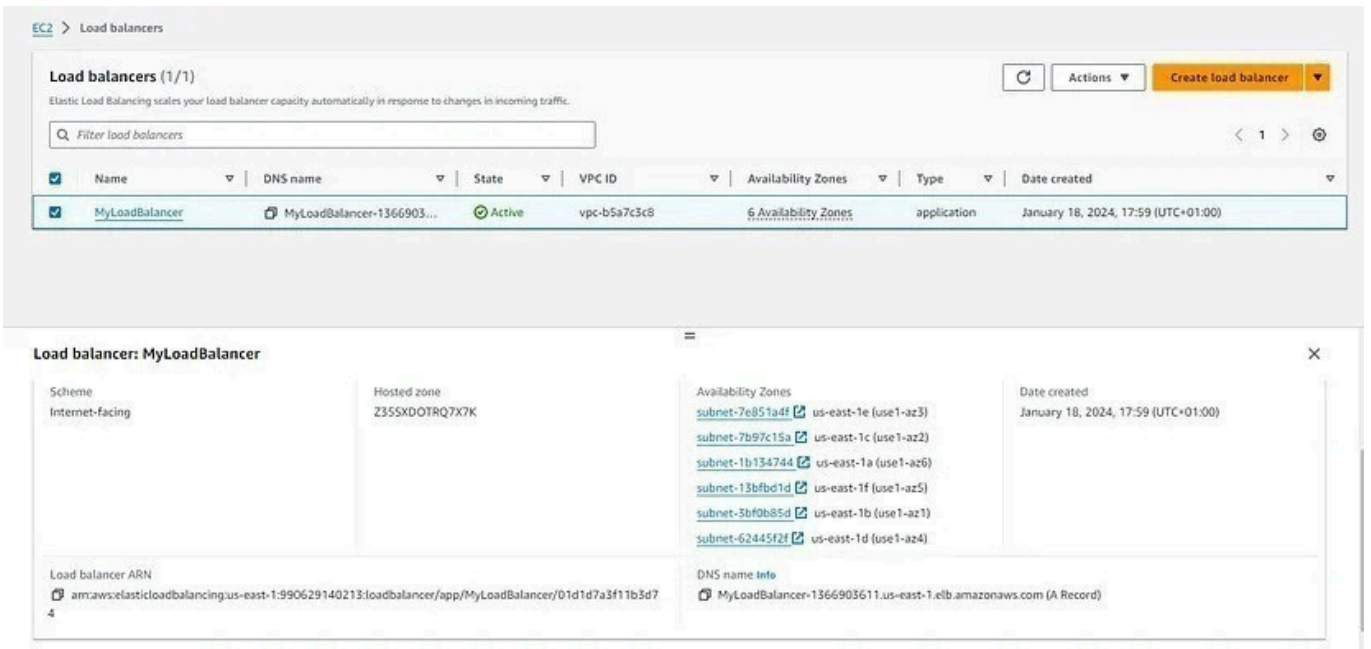
1. Click on Target groups from the left menu section.
2. Select MyTargetGroup and navigate to the Targets tab below.
3. Wait until the status column of the instances changes to healthy (this means both web servers have passed ELB health check)



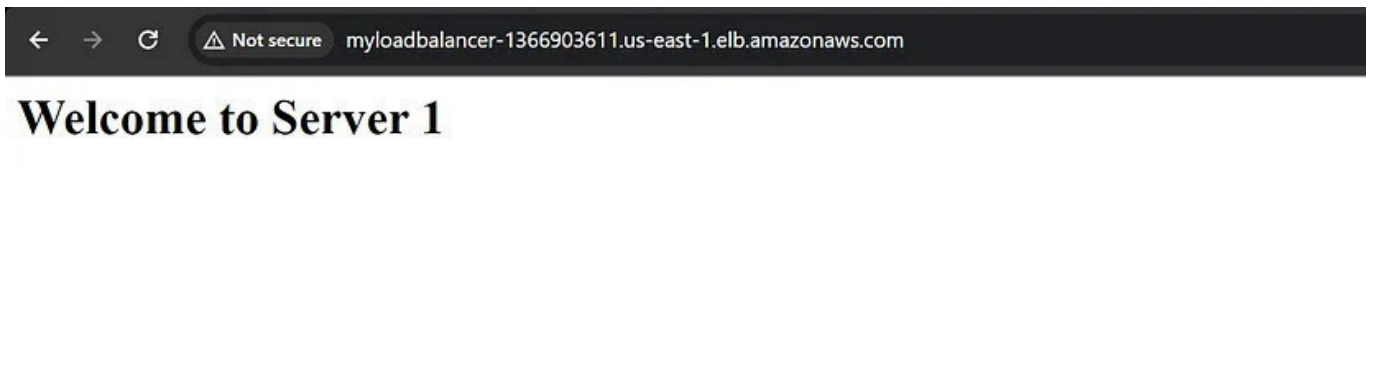
4. Next, navigate to Load Balancers and notice the state of ELB is active.

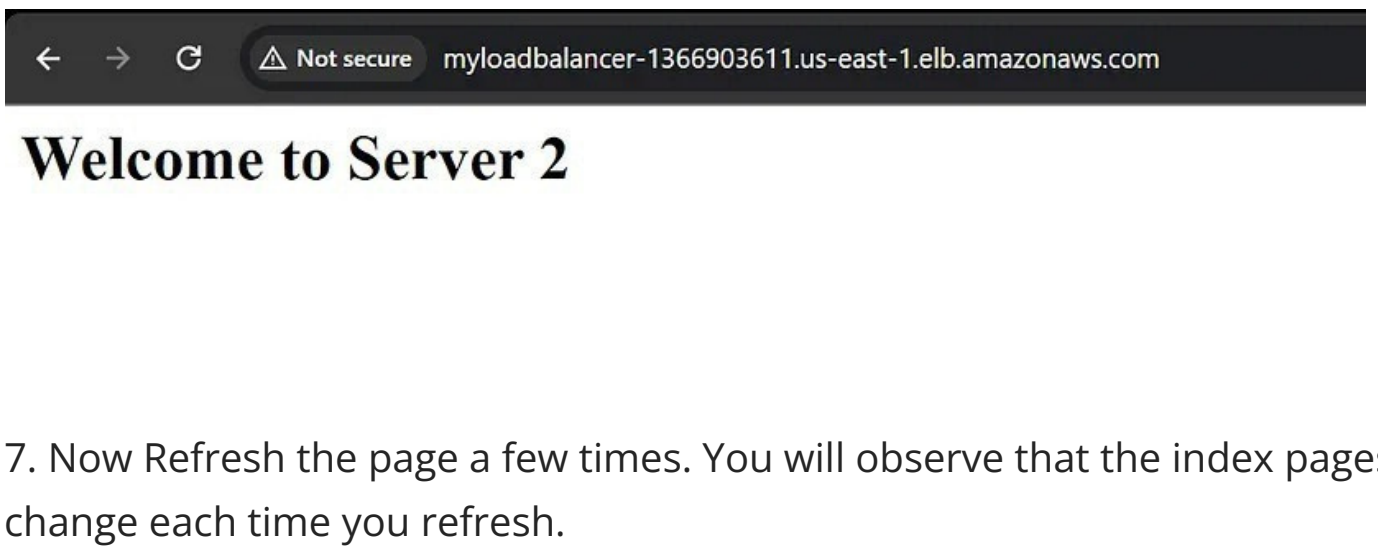
5. Copy the DNS name of the ELB and enter the address in the browser.

- DNS Example: myloadbalancer-1366903611.us-east-1.elb.amazonaws.com



6. You should see the index.html page content of Web Server 1 or Web Server 2



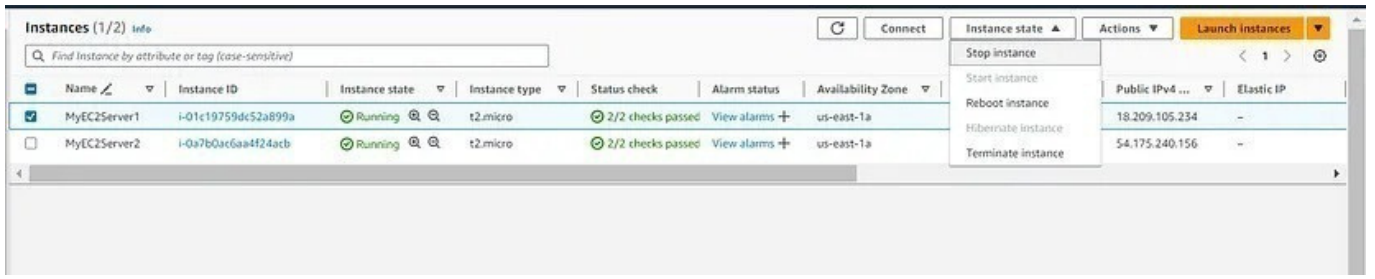


7. Now Refresh the page a few times. You will observe that the index pages change each time you refresh.

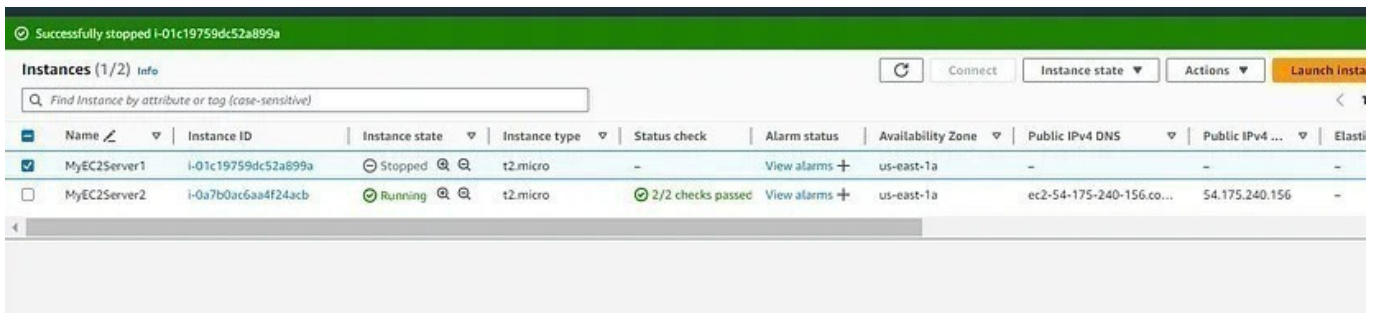
Note: The ELB is equally dividing the incoming traffic to both servers in a Round Robin manner.

8. For testing, if ELB is working properly,

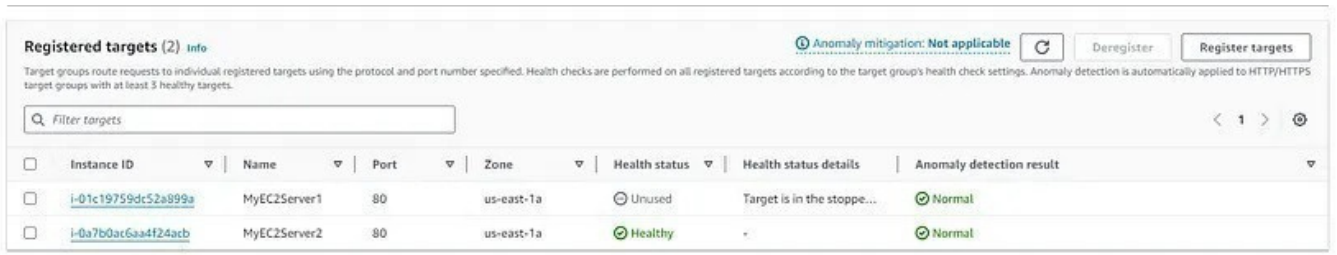
- In the left side menu, scroll up and navigate back to the Instances
- page. Select MyEC2Server1, click on Instance State and click on Stop instance
  to stop the EC2 instance.



- Once MyEC2Server1 is stopped, navigate to Target Groups. Select the MyTargetGroup, Click on the Targets.



- It will say that the stopped instance MyEC2Server1 is unused.



- Refresh the ELB domain name URL in Browser, and notice the HTML webpage remains visible. The ELB is only rendering the HTML page from the MyEC2Server2 instance.

Step 7: Delete AWS Resources

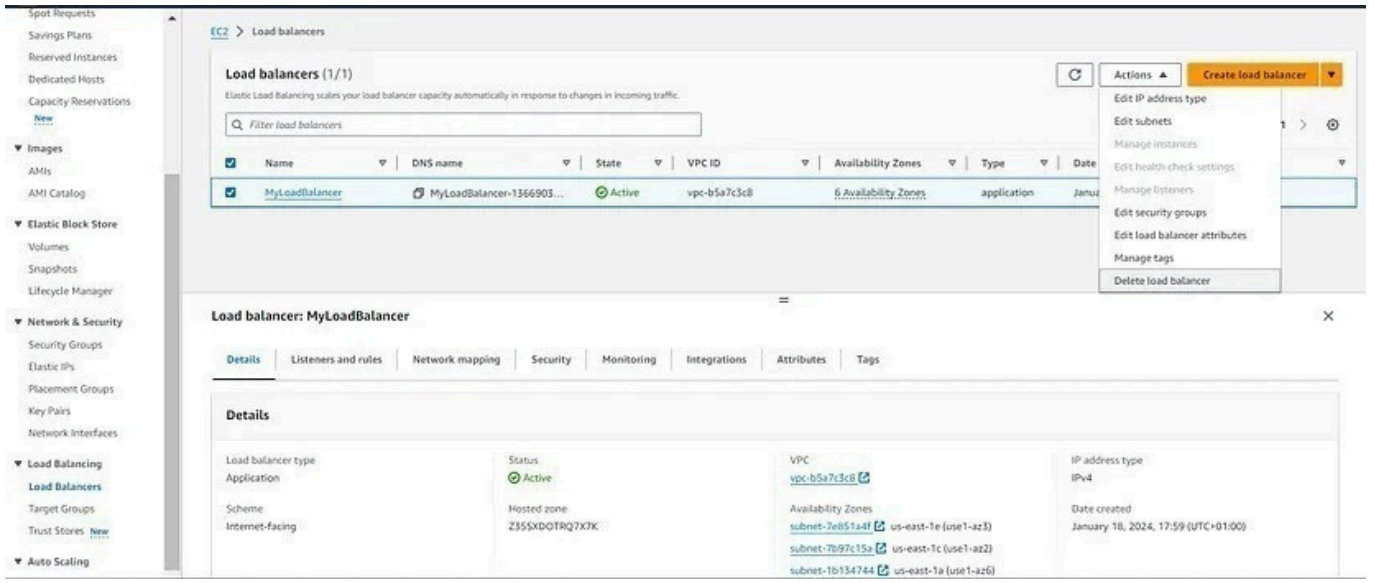
Deleting Load balancer

1. In the EC2 console, navigate to Load Balancers in the left-side panel.

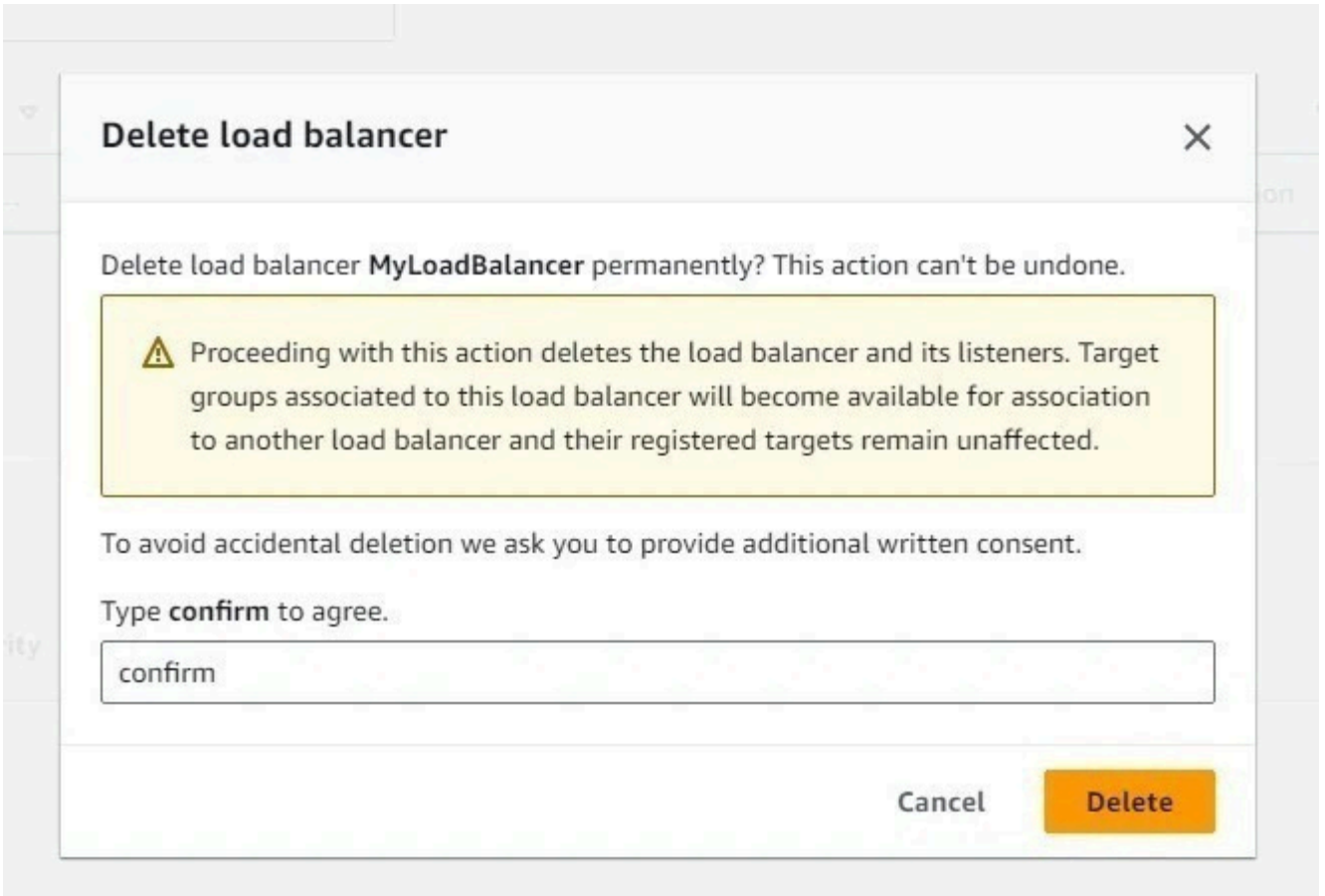
2. MyLoadBalancer will be listed here.

3. To delete the load balancer, you need to perform the following actions:

- Select the load balancer,
- Click on the Actions button,
- Select the Delete option.



4. Confirm by typing confirm when a pop-up is shown. Click on Delete button.



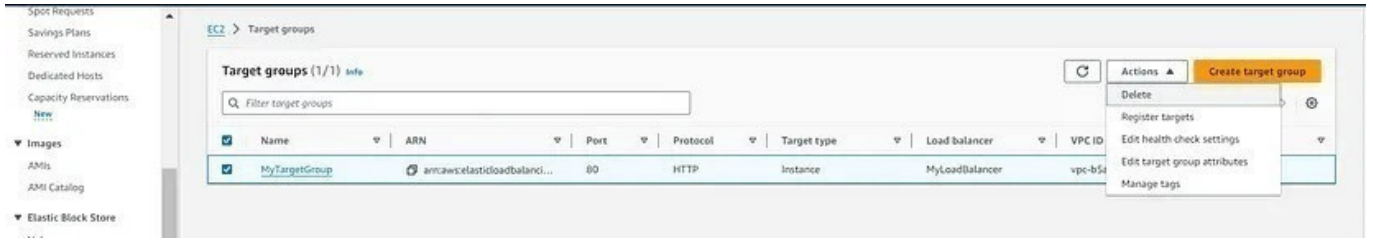
5. Web-server-LG will be deleted immediately.

Deleting Target groups

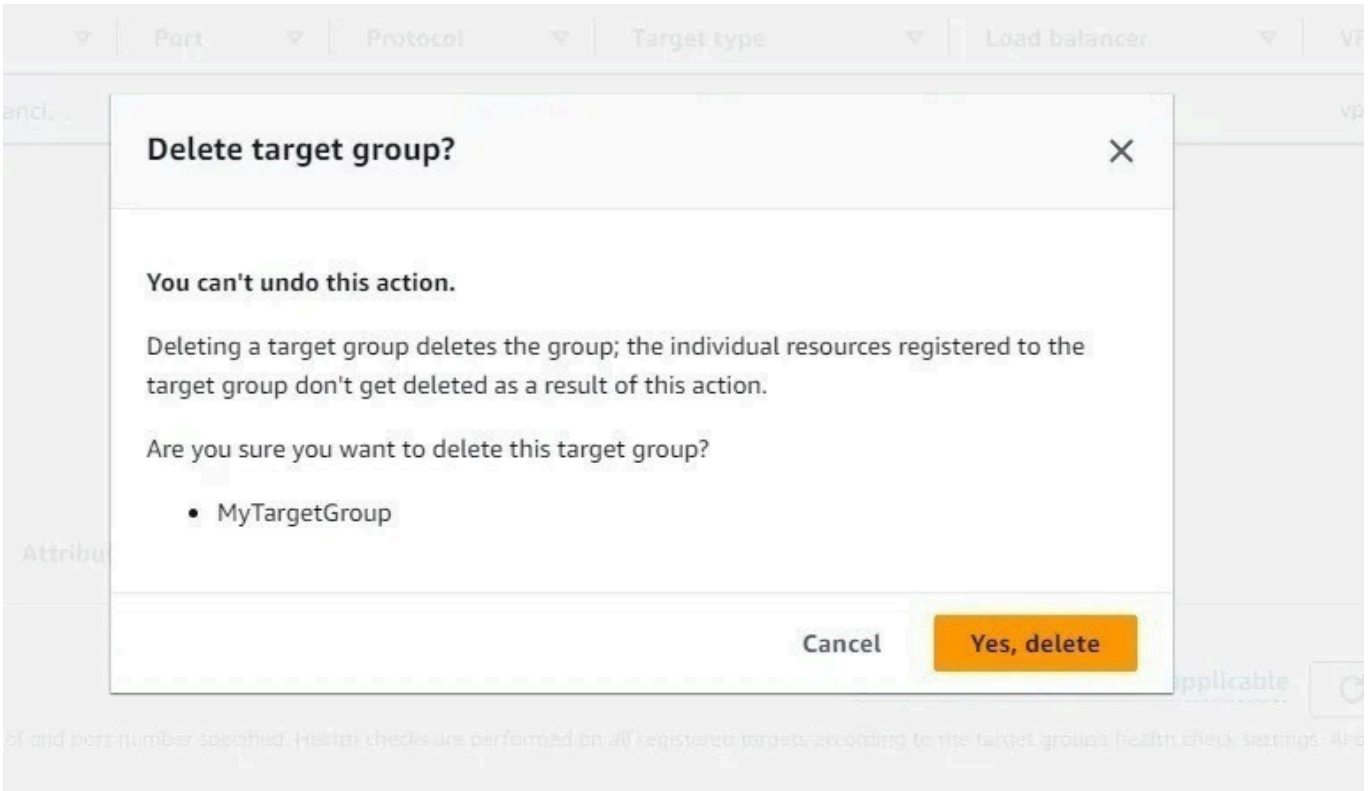
1. In the EC2 console, navigate to Target groups in the left-side panel.
2. MyTargetGroup will be listed here.

3. To delete the target group,you need to perform the following actions:

- Select the target group
- Click on the Actions button,
- Select the Delete option.



4. Confirm by clicking on the Yes, delete button when a pop-up is shown.



5. MyTargetGroup will be deleted immediately.

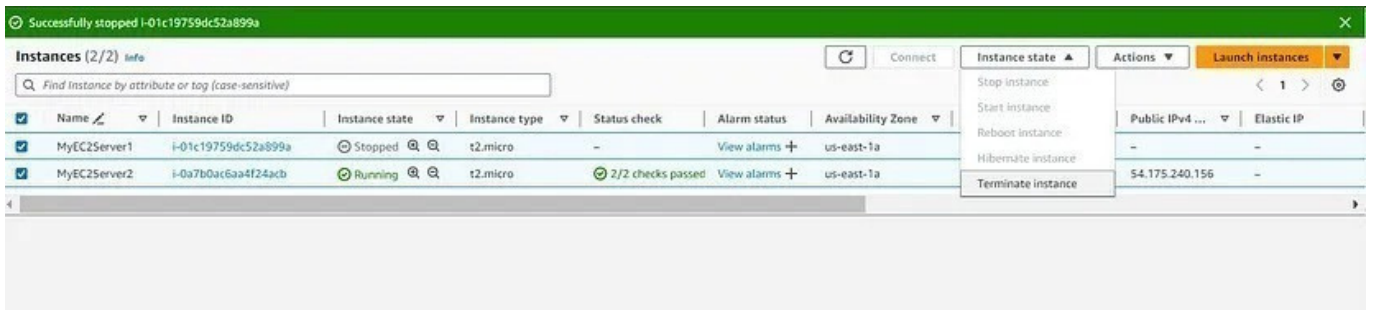
Deleting EC2 Instances

1. In the EC2 console, navigate to Instances in the left-side panel.
2. Two EC2 Instances MyEC2Server1 and MyEC2Server2 will be listed here.

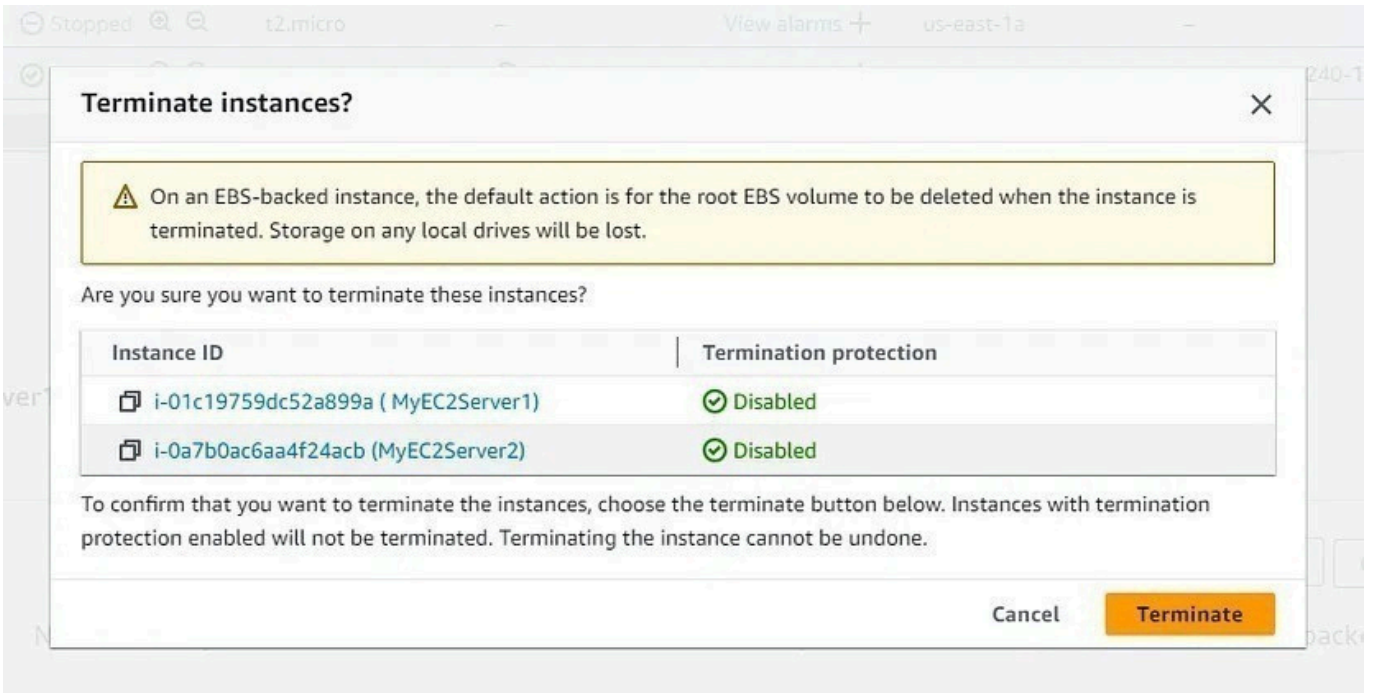
3. To terminate the EC2 Instances, need to perform the following actions:

- Select the EC2 instance

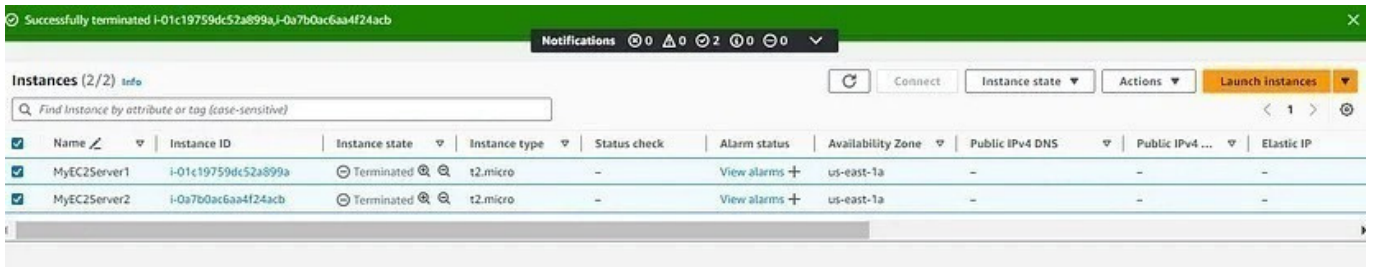
- Click on the Instance state button,
- Select the Terminate instance option



4. Confirm by clicking on the Terminate button when a pop-up is shown.



5. EC2 Instances will be terminated immediately.



You’re all done! Congratulations!

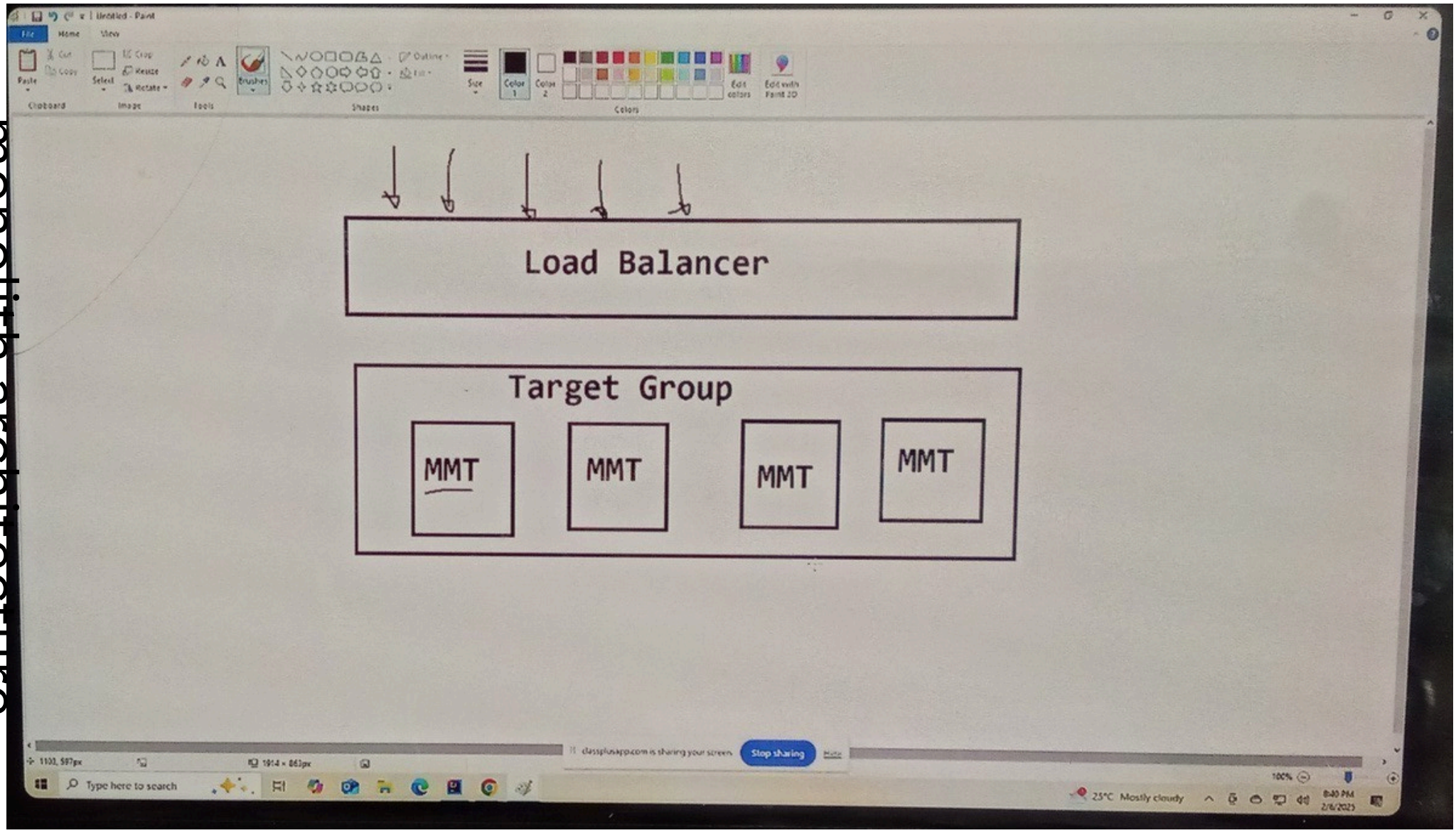
That’s all I have for today folks. Thank you for reading and/or following along! I hope this project was helpful and worth your while. Stay tuned for my next project on this journey into the cloud.

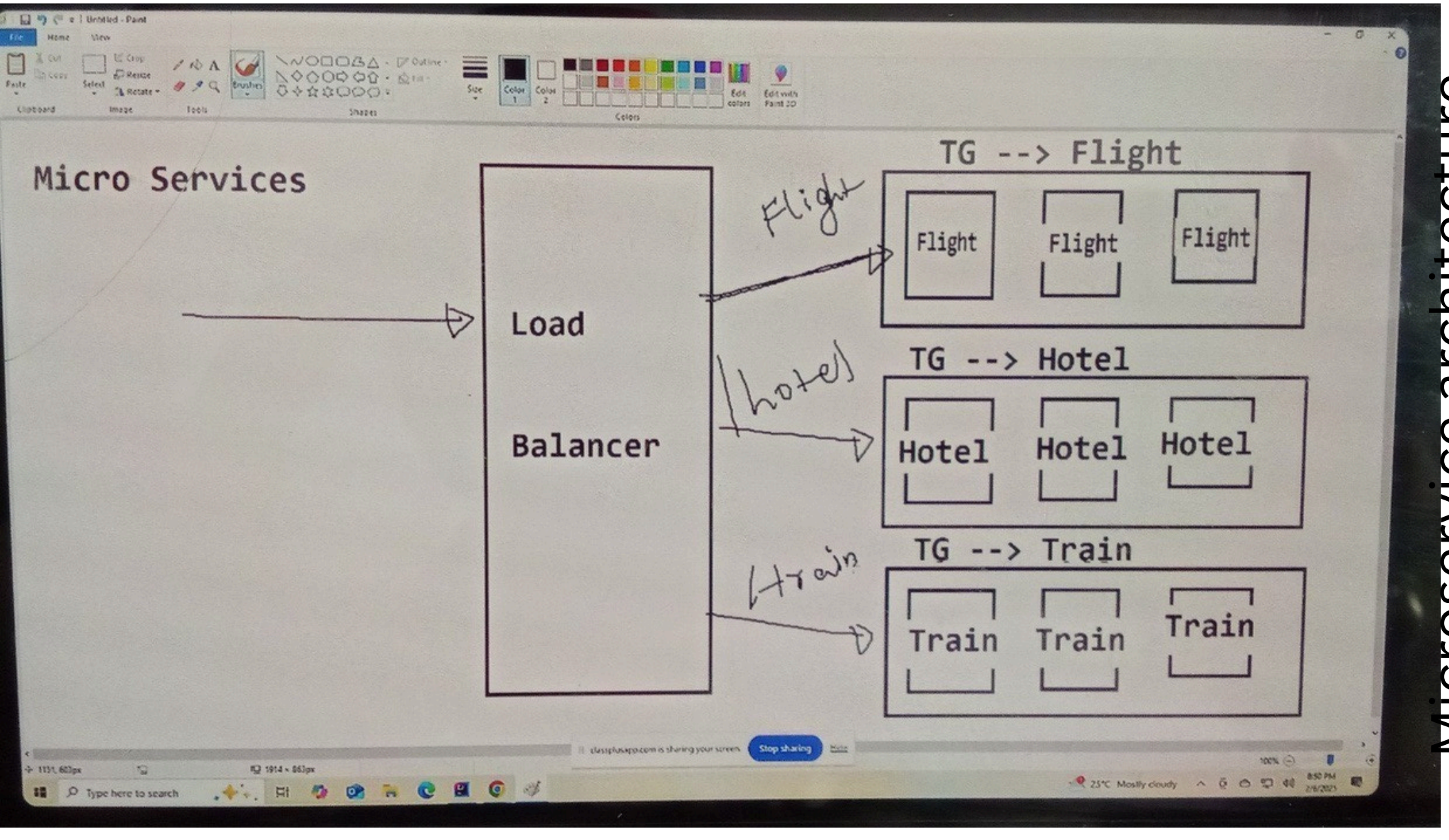
Let’s connect on LinkedIn! 👉 https://www.linkedin.com/in/meriemterki/

AWS Ec2 Elb Elastic Load Balancer

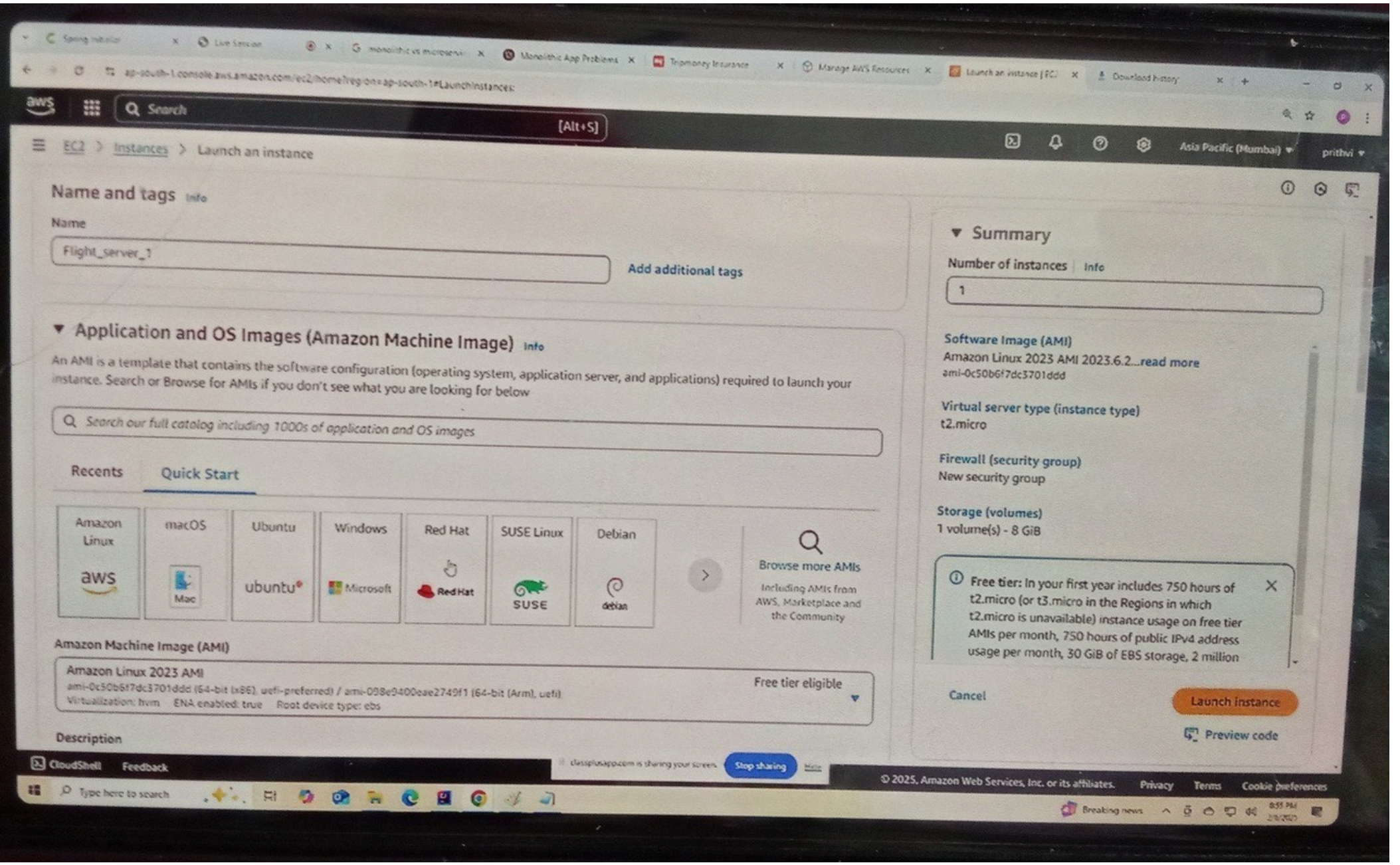
Monolith and Microservice in AWS

monolith architecture

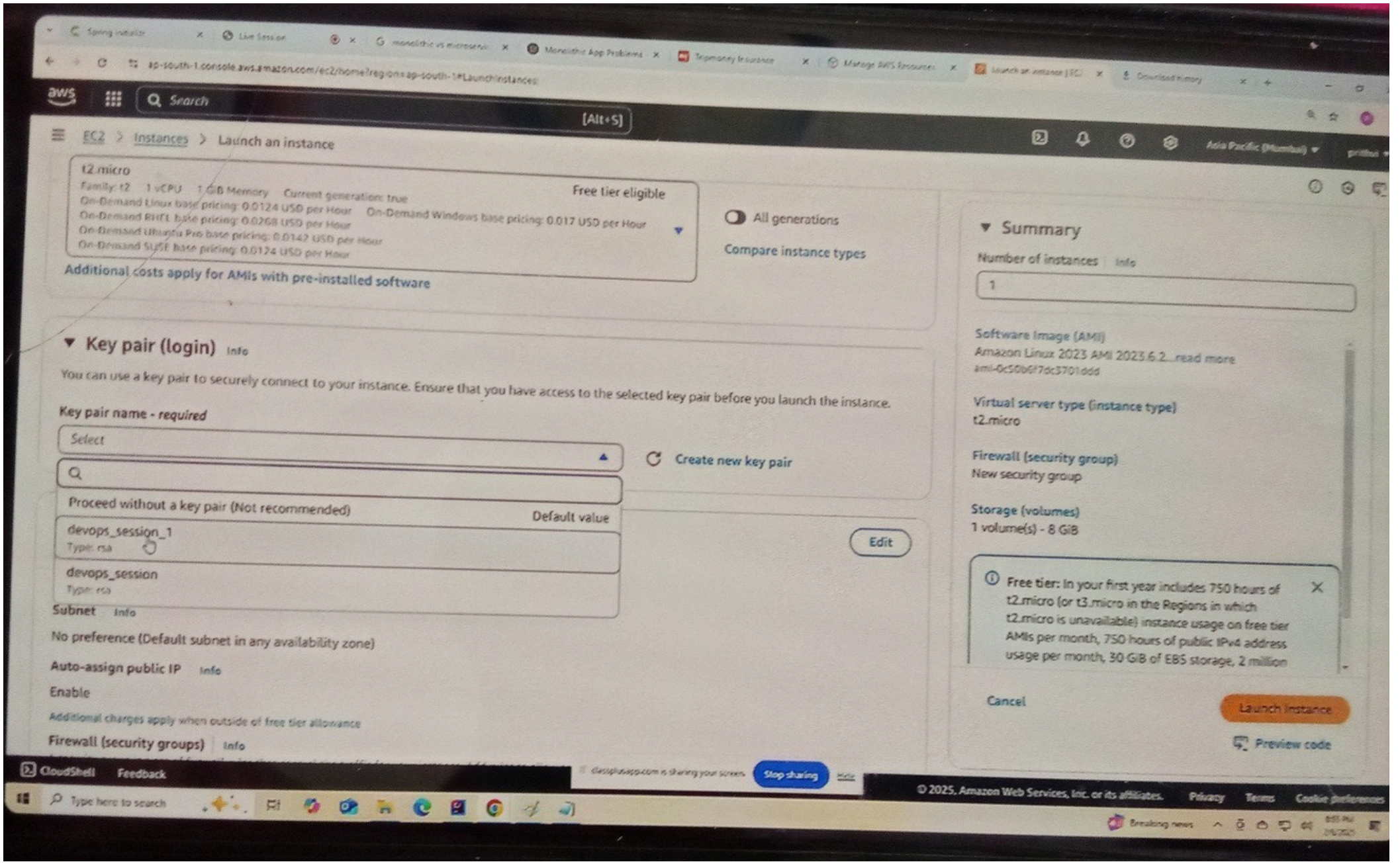


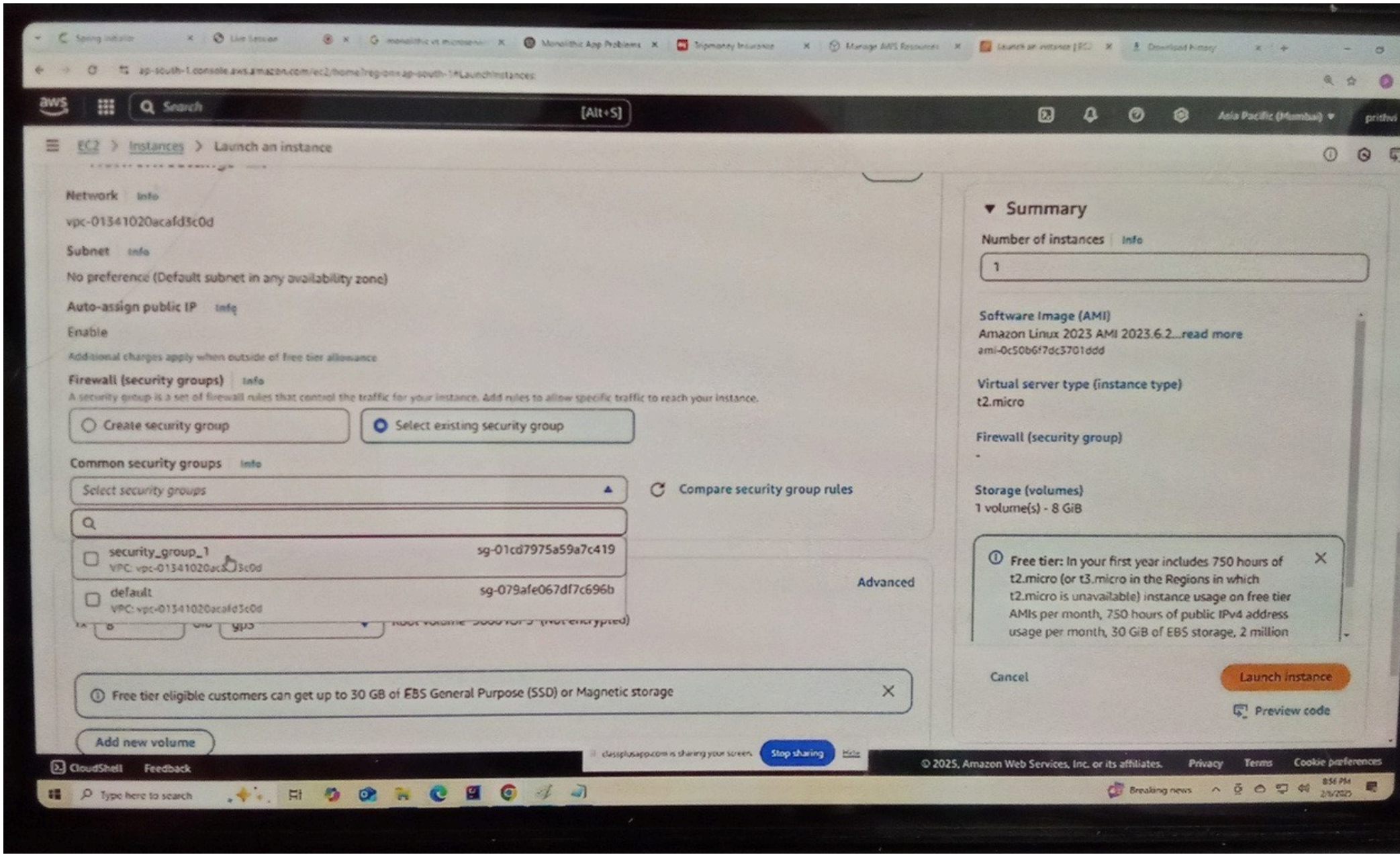


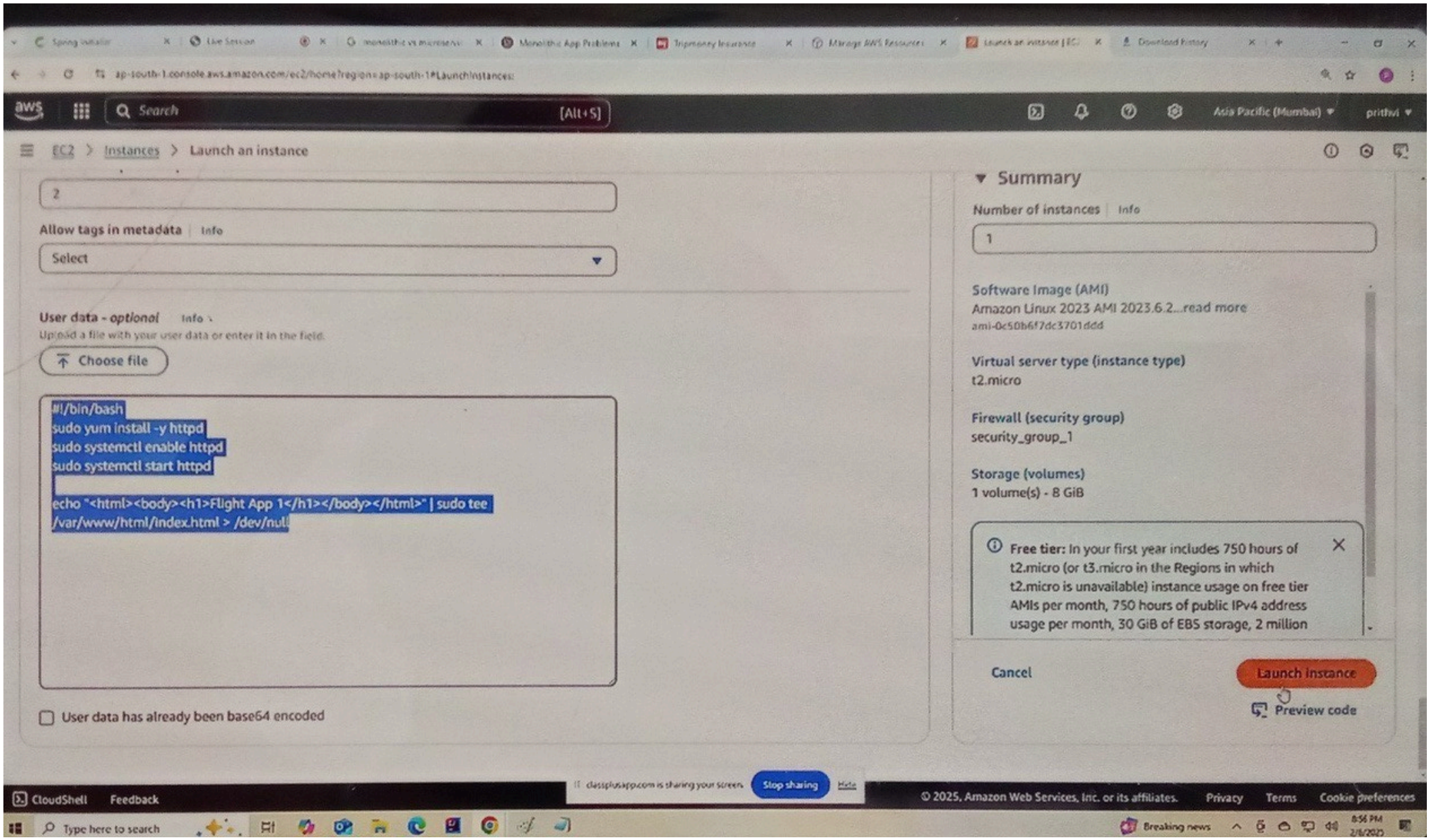
Microservice architecture



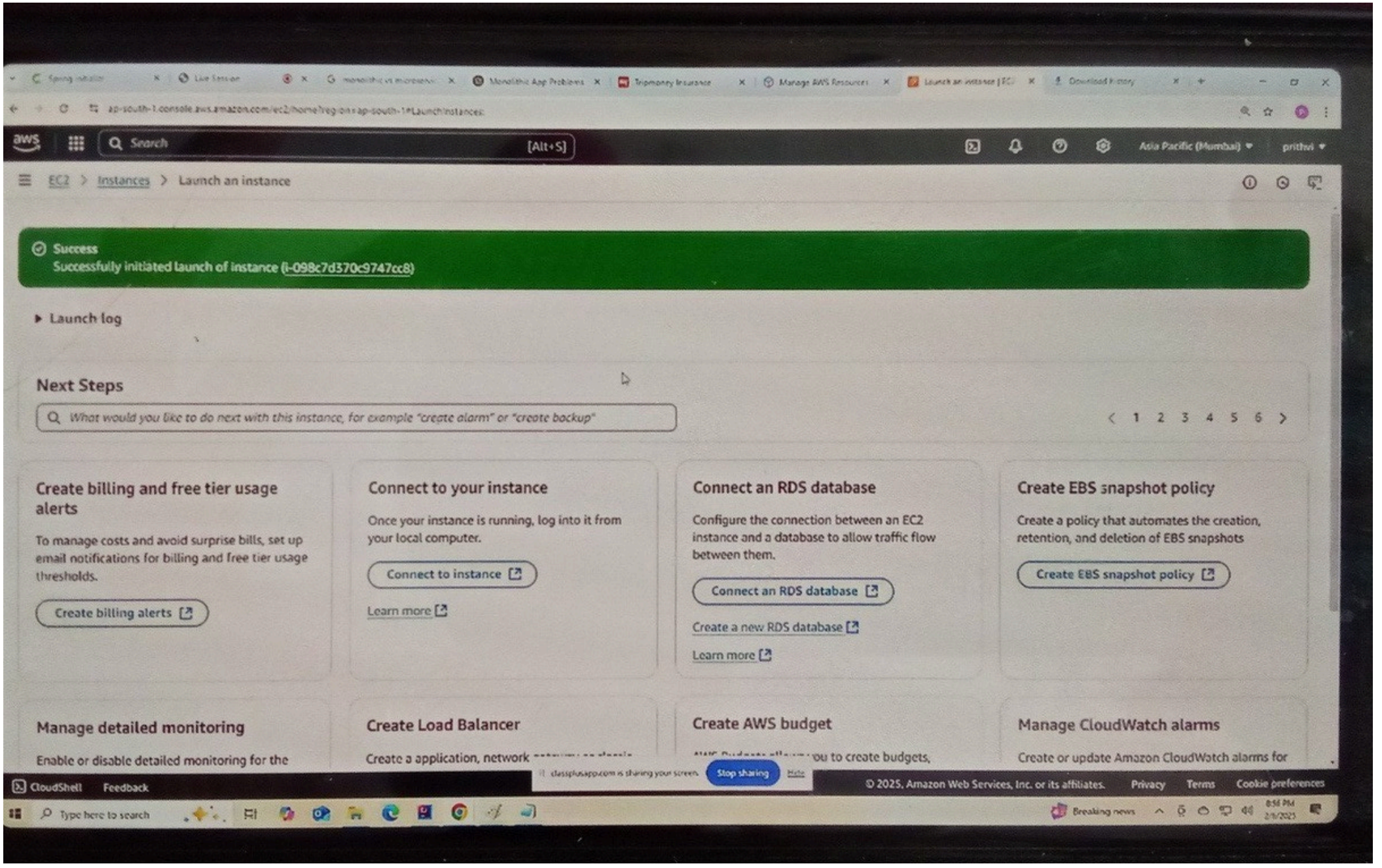
step 1: create Flight_ server_2ec2 instance

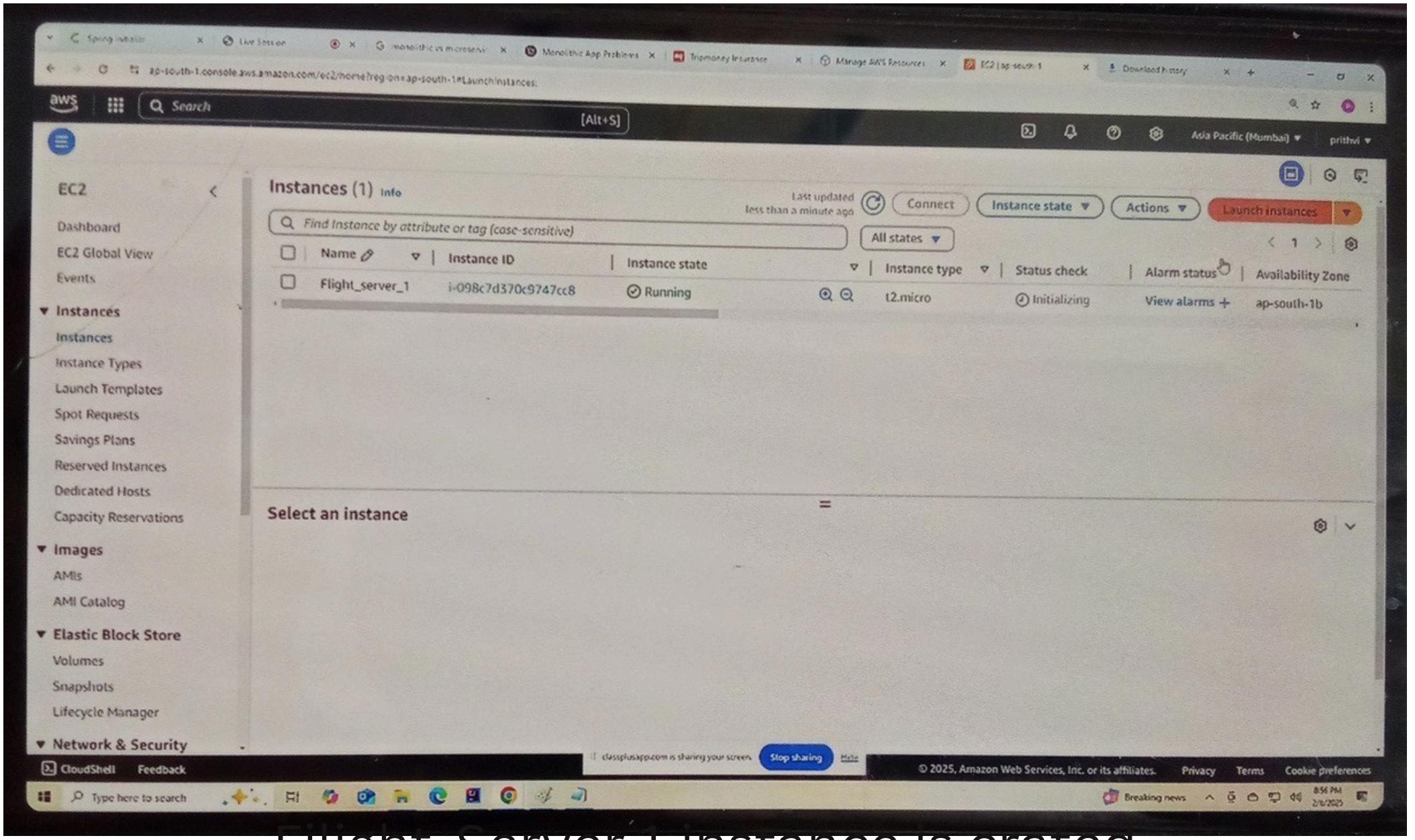

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:
Search
[Alt+S]
EC2 > Instances > Launch an instance
Asia Pacific (Mumbai)
t2.micro
Family: t2 1 vCPU 1 GiB Memory Current generation: true
Free tier eligible
On-Demand Linux base pricing: 0.0124 USD per Hour
On-Demand Windows base pricing: 0.017 USD per Hour
On-Demand RHEL base pricing: 0.0268 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0142 USD per Hour
On-Demand SUSE base pricing: 0.0124 USD per Hour
Additional costs apply for AMIs with pre-installed software
All generations
Compare instance types
Key pair (login) Info
You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.
Key pair name - required
Select
Create new key pair
Proceed without a key pair (Not recommended)
Default value
devops_session_1
Type: rsa
devops_session
Type: rsa
Edit
Subnet Info
No preference (Default subnet in any availability zone)
Auto-assign public IP Info
Enable
Additional charges apply when outside of free tier allowance
Firewall (security groups) Info
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2023 AMI 2023.6.2...read more
ami-0c50b6f7dc3701ddd
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million
Cancel
Launch instance
Preview code
CloudShell
Feedback
Stop sharing
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
Breaking news





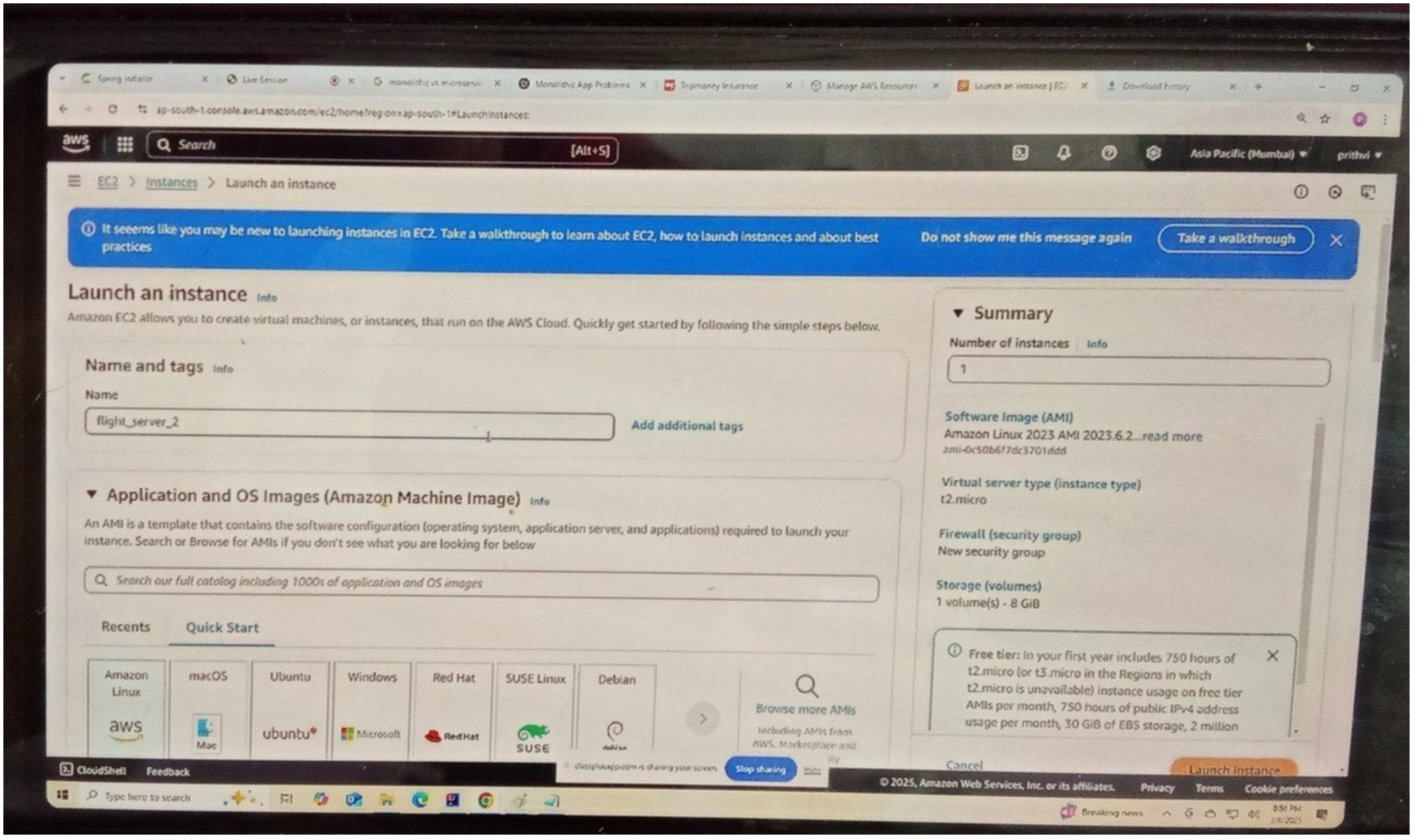
step 2 : add below command in User data


Search [Alt+S]
Asia Pacific (Mumbai)
prithvi
EC2 > Instances > Launch an instance
Success
Successfully initiated launch of instance (i-098c7d370c9747cc8)
Launch log
Next Steps
What would you like to do next with this instance, for example "create alarm" or "create backup"
Create billing and free tier usage alerts
To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds.
Create billing alerts
Connect to your instance
Once your instance is running, log into it from your local computer.
Connect to instance
Learn more
Connect an RDS database
Configure the connection between an EC2 instance and a database to allow traffic flow between them.
Connect an RDS database
Create a new RDS database
Learn more
Create EBS snapshot policy
Create a policy that automates the creation, retention, and deletion of EBS snapshots
Create EBS snapshot policy
Manage detailed monitoring
Enable or disable detailed monitoring for the
Create Load Balancer
Create a application, network
Create AWS budget
ou to create budgets,
Manage CloudWatch alarms
Create or update Amazon CloudWatch alarms for
Stop sharing
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
Breaking news



Filight_Server 1 instance is crated

Similarly create Filight_Server 2 instance is crated



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

aws | Search [Alt+S] | Asia Pacific (Mumbai) | prithvi

EC2 > Instances > Launch an instance

t2.micro — Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Linux base pricing: 0.0124 USD per Hour On-Demand Windows base pricing: 0.017 USD per Hour
On-Demand RHEL base pricing: 0.0268 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0142 USD per Hour
On-Demand SUSE base pricing: 0.0124 USD per Hour

Additional costs apply for AMIs with pre-installed software

All generations

Compare instance types

Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

devops_session

Create new key pair

Network settings Info

Edit

Network Info
vpc-01341020acafd3c0d

Subnet Info
No preference (Default subnet in any availability zone)

Auto-assign public IP Info
Enable
Additional charges apply when outside of free tier allowance

Firewall (security groups) Info

Summary

Number of instances Info
1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.6.2...read more
ami-0c50b6f7dc3701ddd

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

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

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million

Cancel

Launch instance

Preview code

classplusapp.com is sharing your screen. Stop sharing Hide

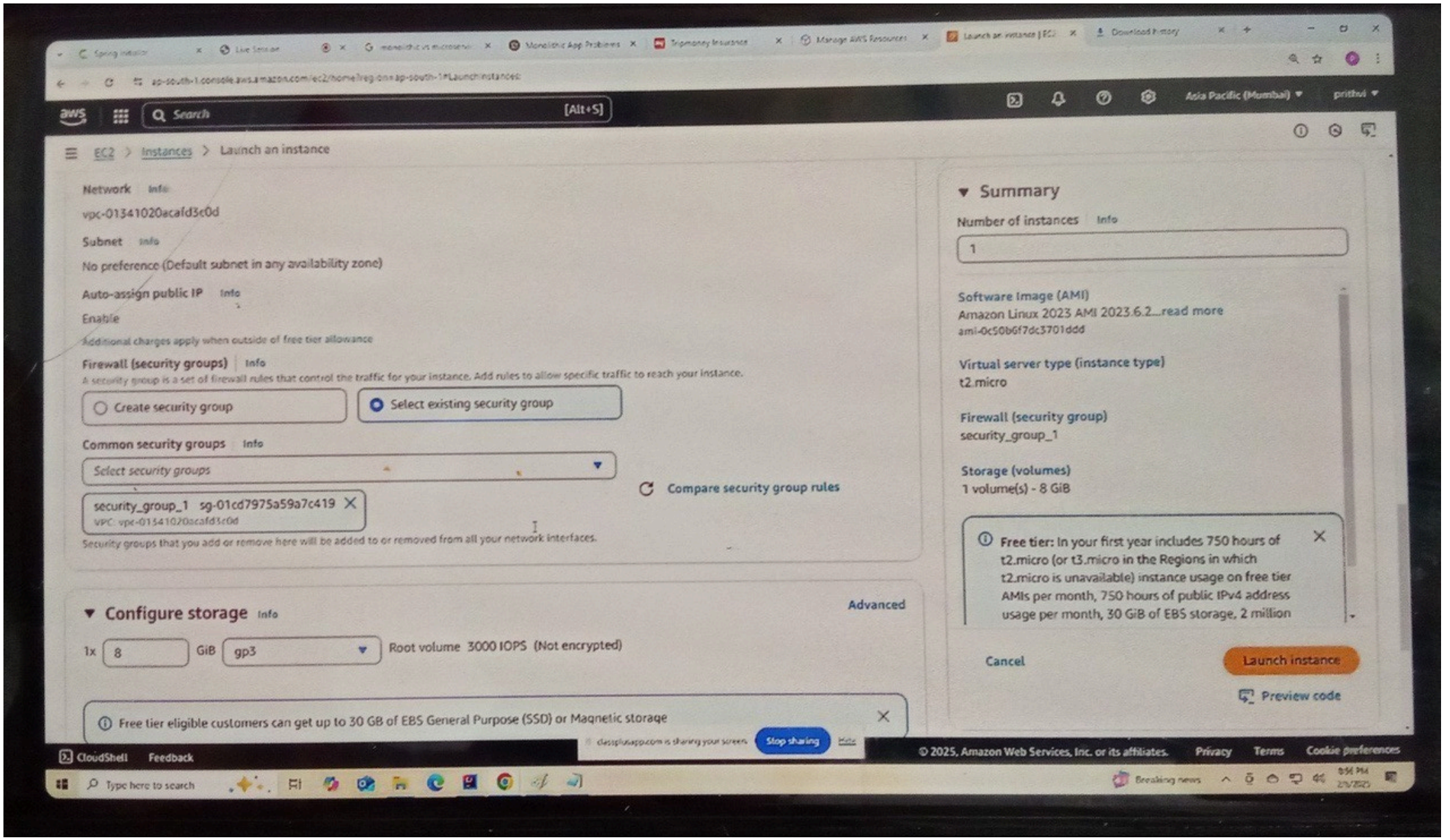
CloudShell Feedback

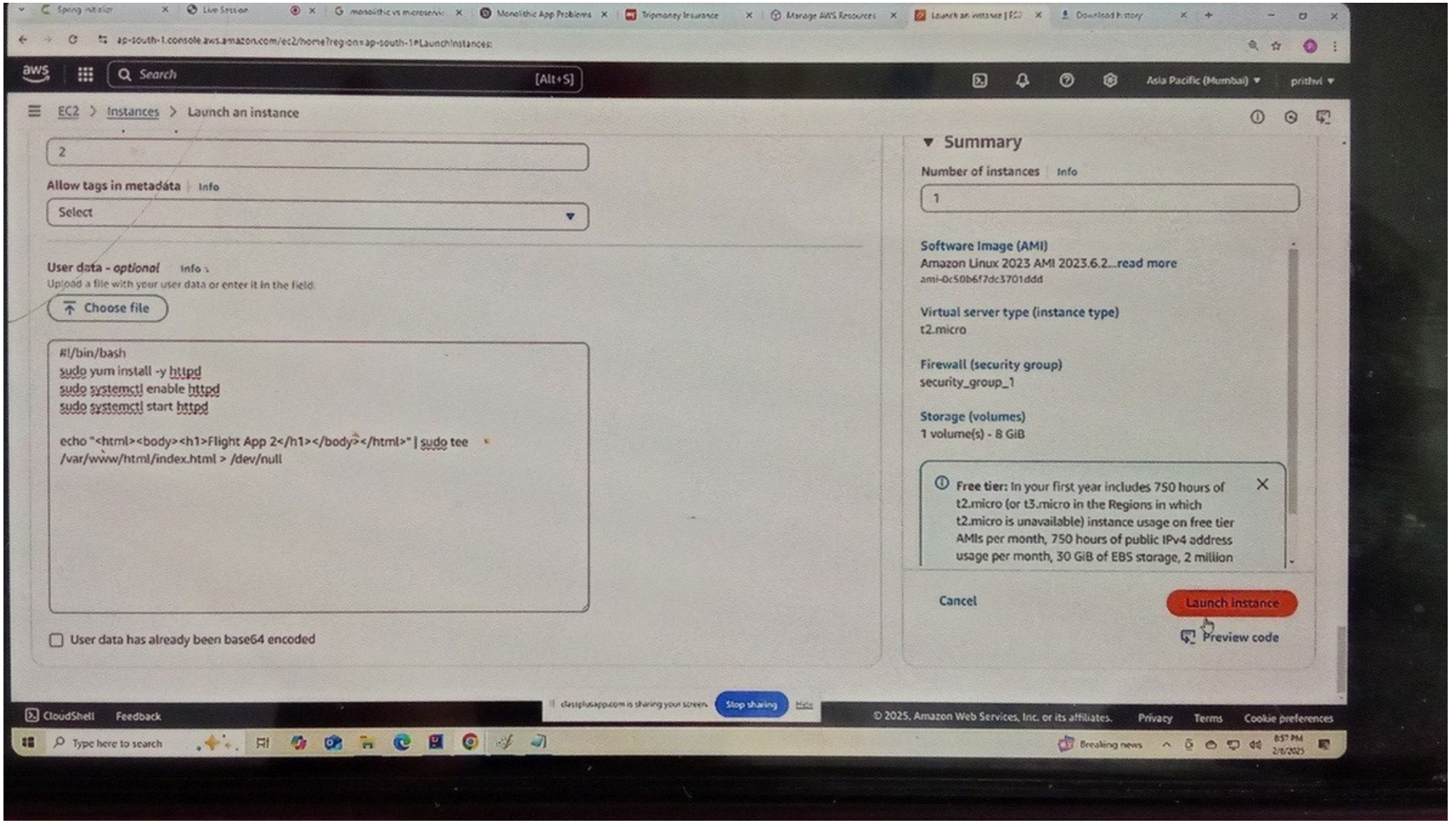
Privacy Terms Cookie preferences

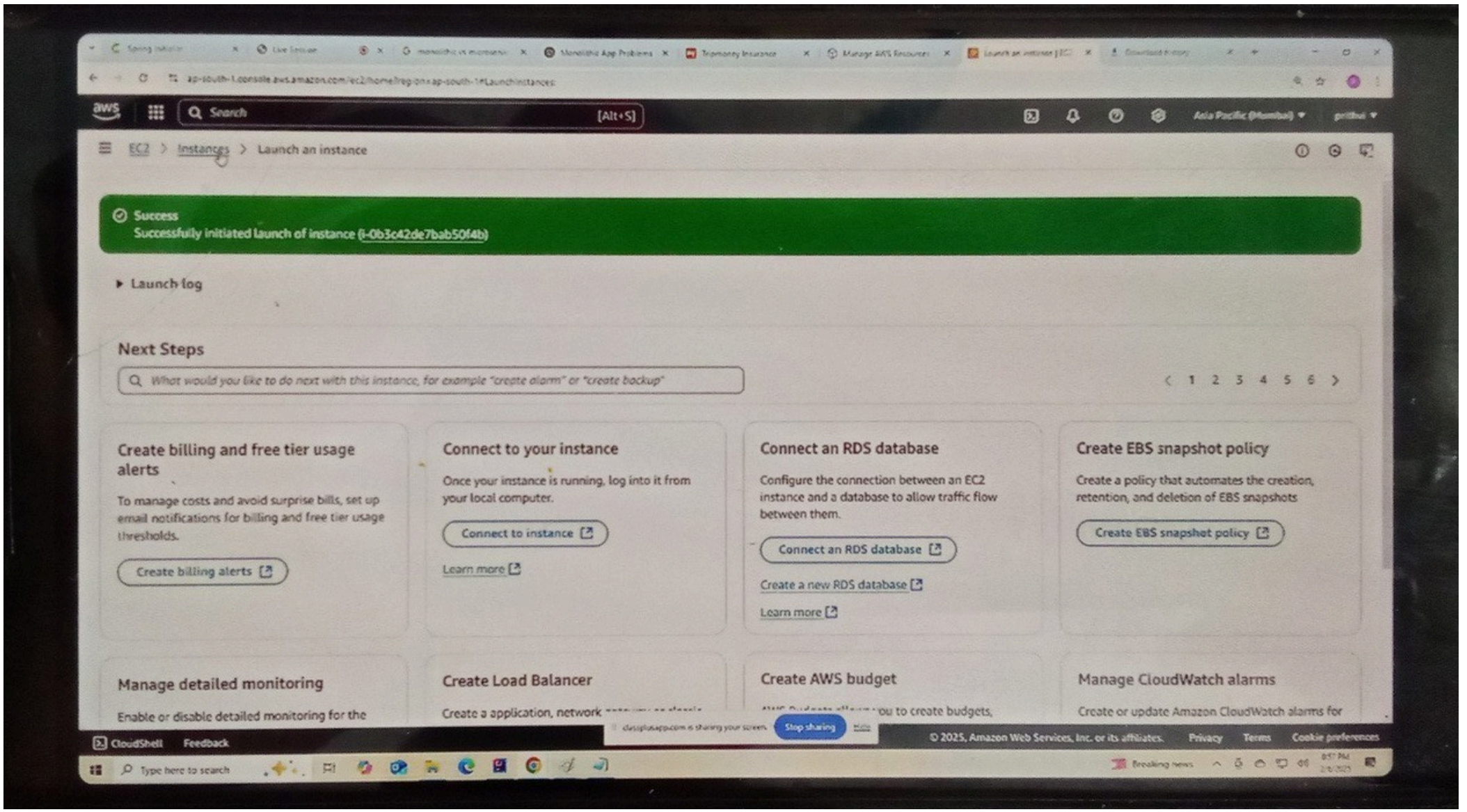
Type here to search

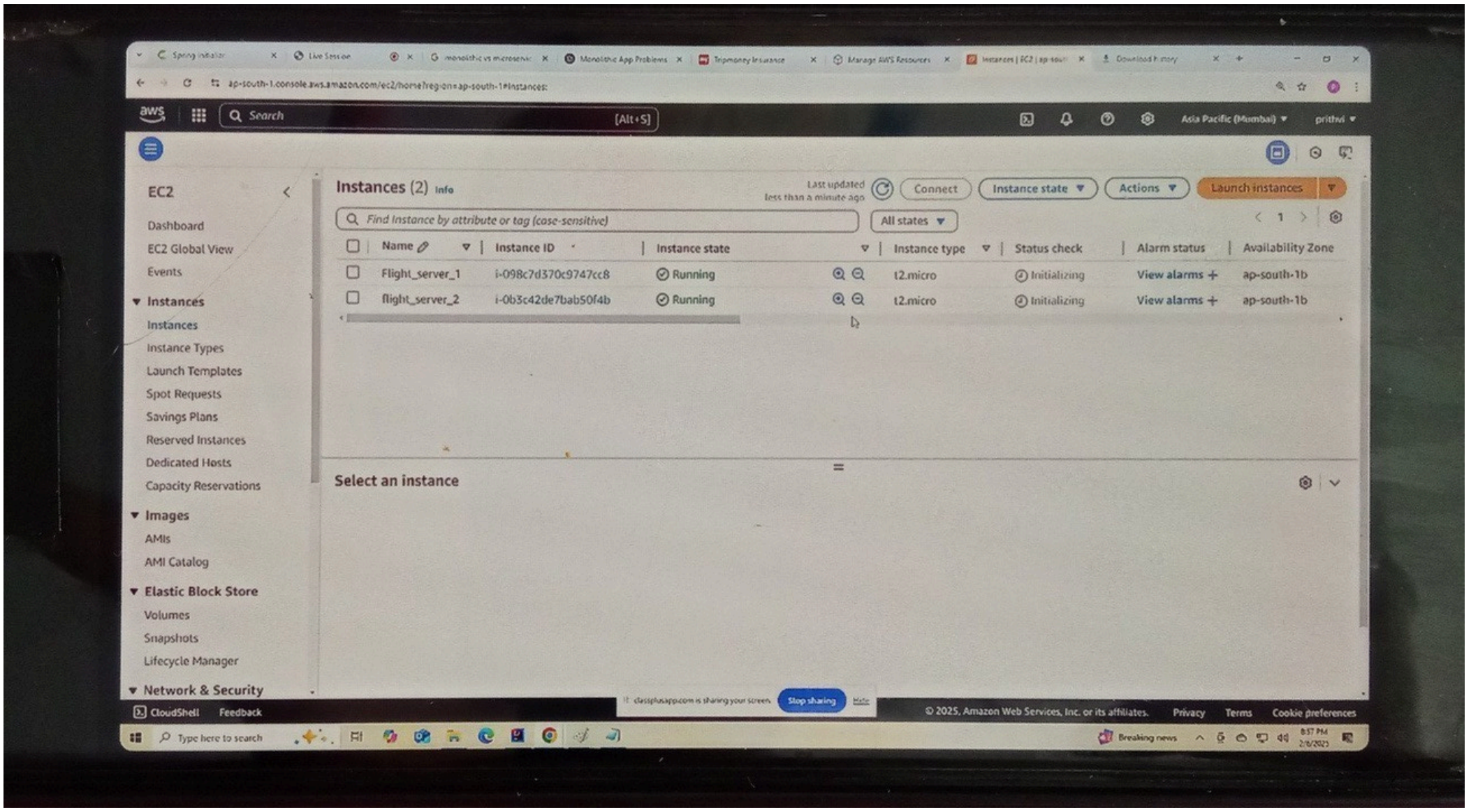
Breaking news

8:56 PM
2/9/2025


Search
[Alt+S]
Asia Pacific (Mumbai)
prithvi
EC2 > Instances > Launch an instance
Network Info
vpc-01341020acafd3c0d
Subnet Info
No preference (Default subnet in any availability zone)
Auto-assign public IP Info
Enable
Additional charges apply when outside of free tier allowance
Firewall (security groups) Info
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.
Create security group
Select existing security group
Common security groups Info
Select security groups
security_group_1 sg-01cd7975a59a7c419
VPC: vpc-01341020acafd3c0d
Compare security group rules
Security groups that you add or remove here will be added to or removed from all your network interfaces.
Configure storage Info
Advanced
1x 8 GiB gp3 Root volume 3000 IOPS (Not encrypted)
Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2023 AMI 2023.6.2...read more
ami-0c50b6f7dc3701ddd
Virtual server type (instance type)
t2.micro
Firewall (security group)
security_group_1
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million
Cancel
Launch instance
Preview code
CloudShell
Feedback
Stop sharing
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
Breaking news

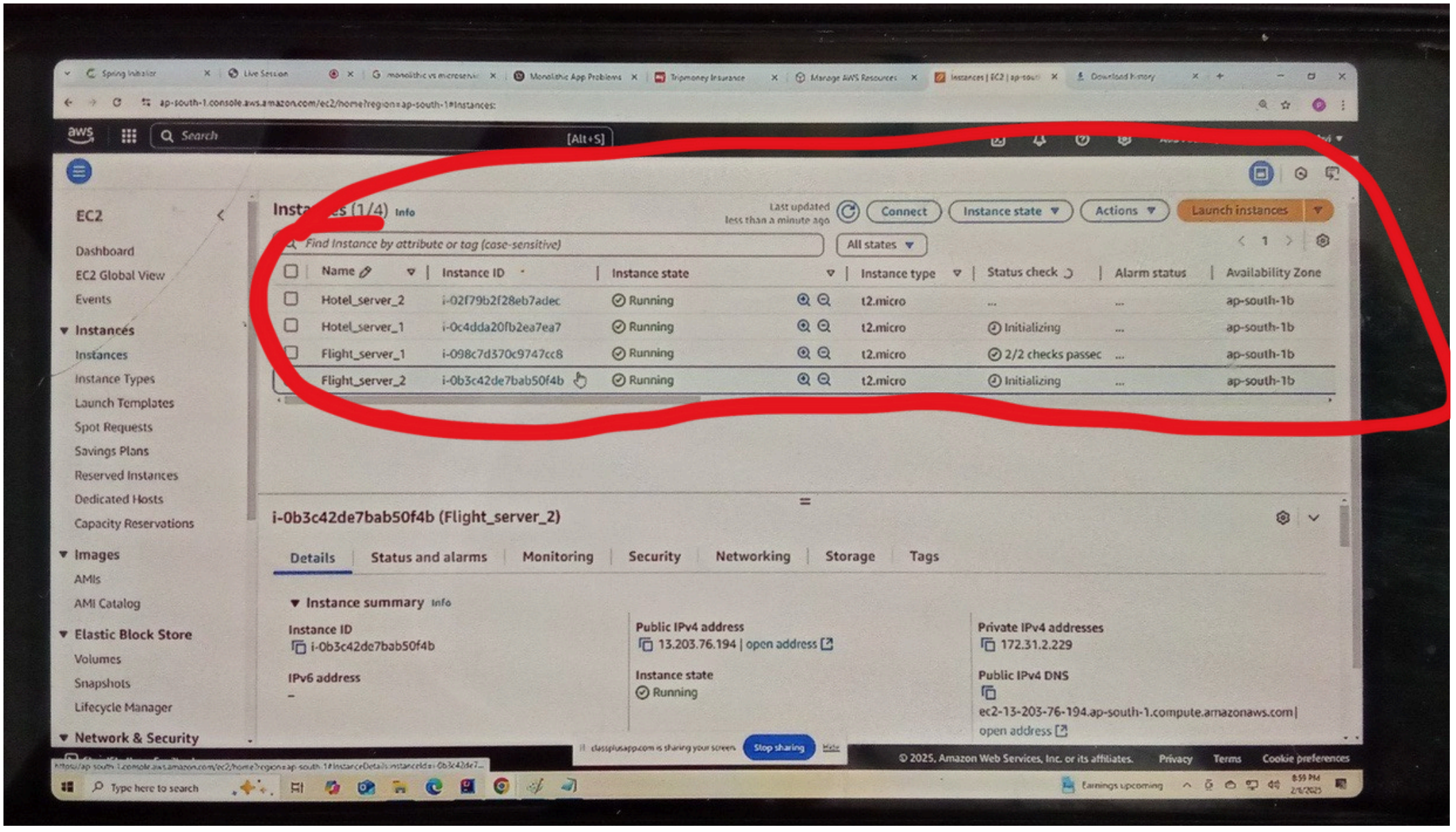

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:
Search [Alt+S]
Asia Pacific (Mumbai)
prithvi
EC2 > Instances > Launch an instance
2
Allow tags in metadata | Info
Select
User data - optional Info
Upload a file with your user data or enter it in the field.
Choose file
#!/bin/bash
sudo yum install -y httpd
sudo systemctl enable httpd
sudo systemctl start httpd
echo "<html><body><h1>Flight App 2</h1></body></html>" | sudo tee /var/www/html/index.html > /dev/null
User data has already been base64 encoded
Summary
Number of instances | Info
1
Software Image (AMI)
Amazon Linux 2023 AMI 2023.6.2...read more
ami-0c50b6f7dc3701ddd
Virtual server type (instance type)
t2.micro
Firewall (security group)
security_group_1
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million
Cancel
Launch instance
Preview code
CloudShell
Feedback
classplusapp.com is sharing your screen.
Stop sharing
Hide
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
Breaking news
8:57 PM
2/6/2025





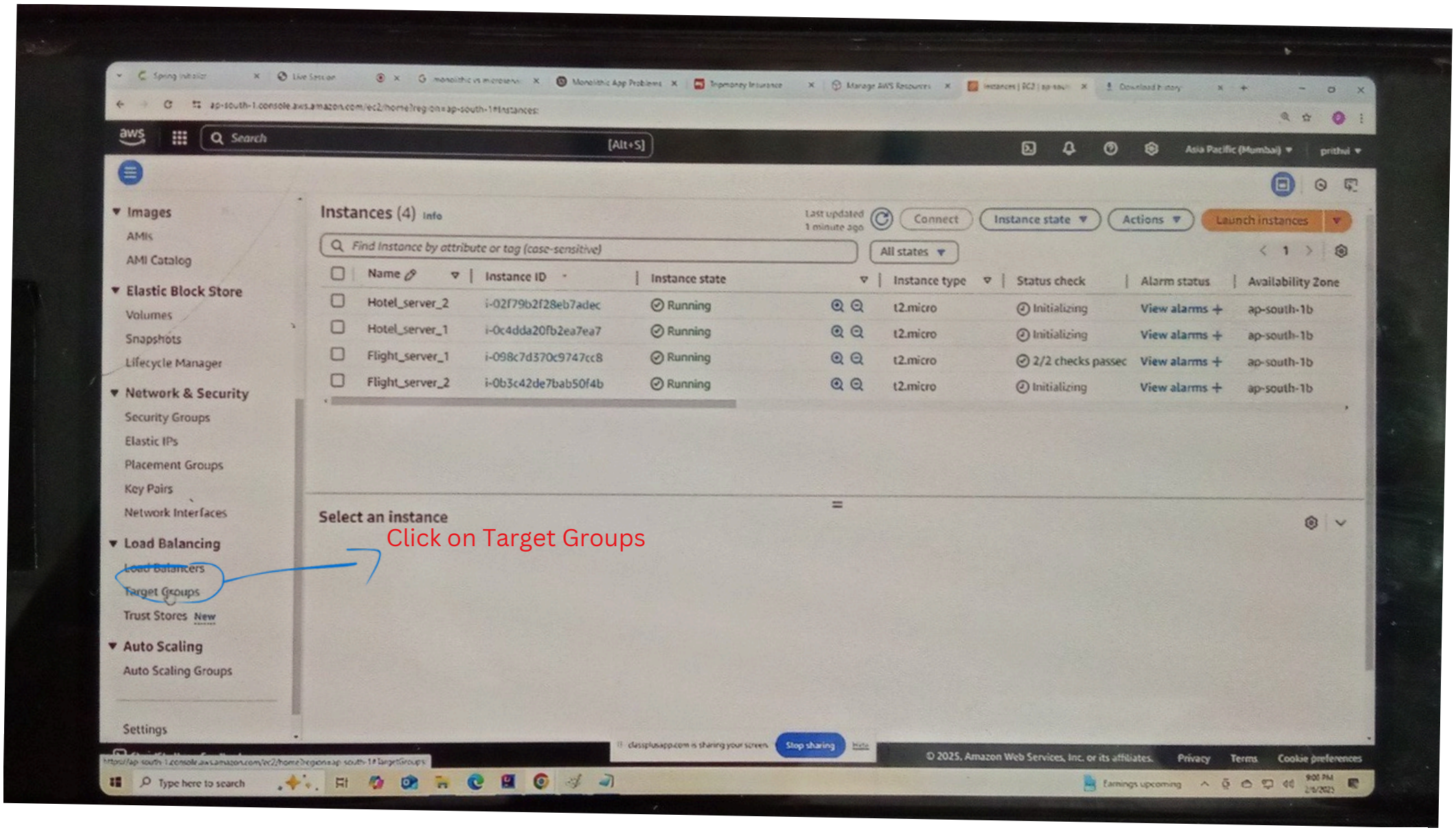
Now see Flight_server1 and Flight_server2 is created

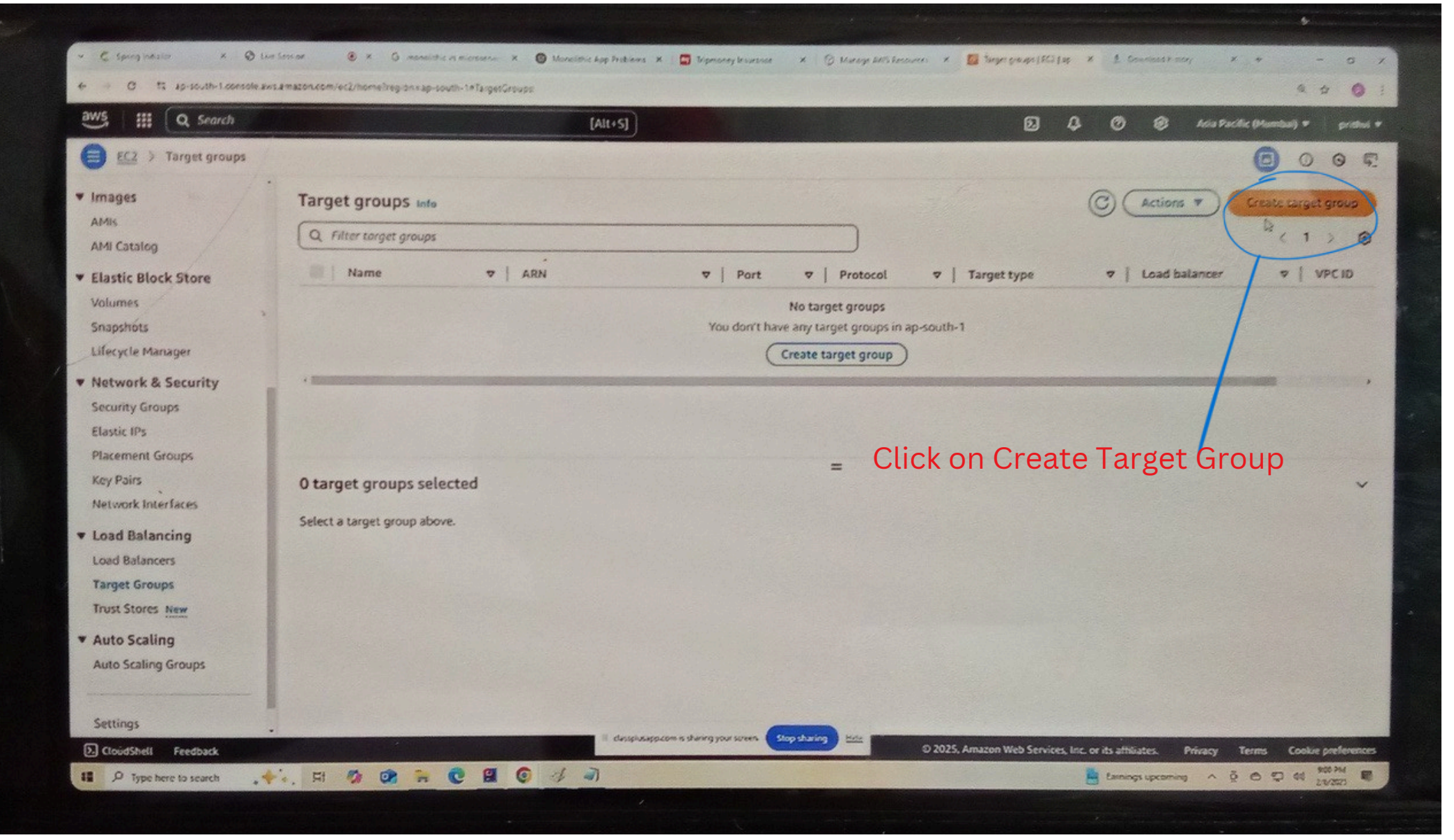
Now see Similarly do Hotel_server1 and Hotel_server2 is created

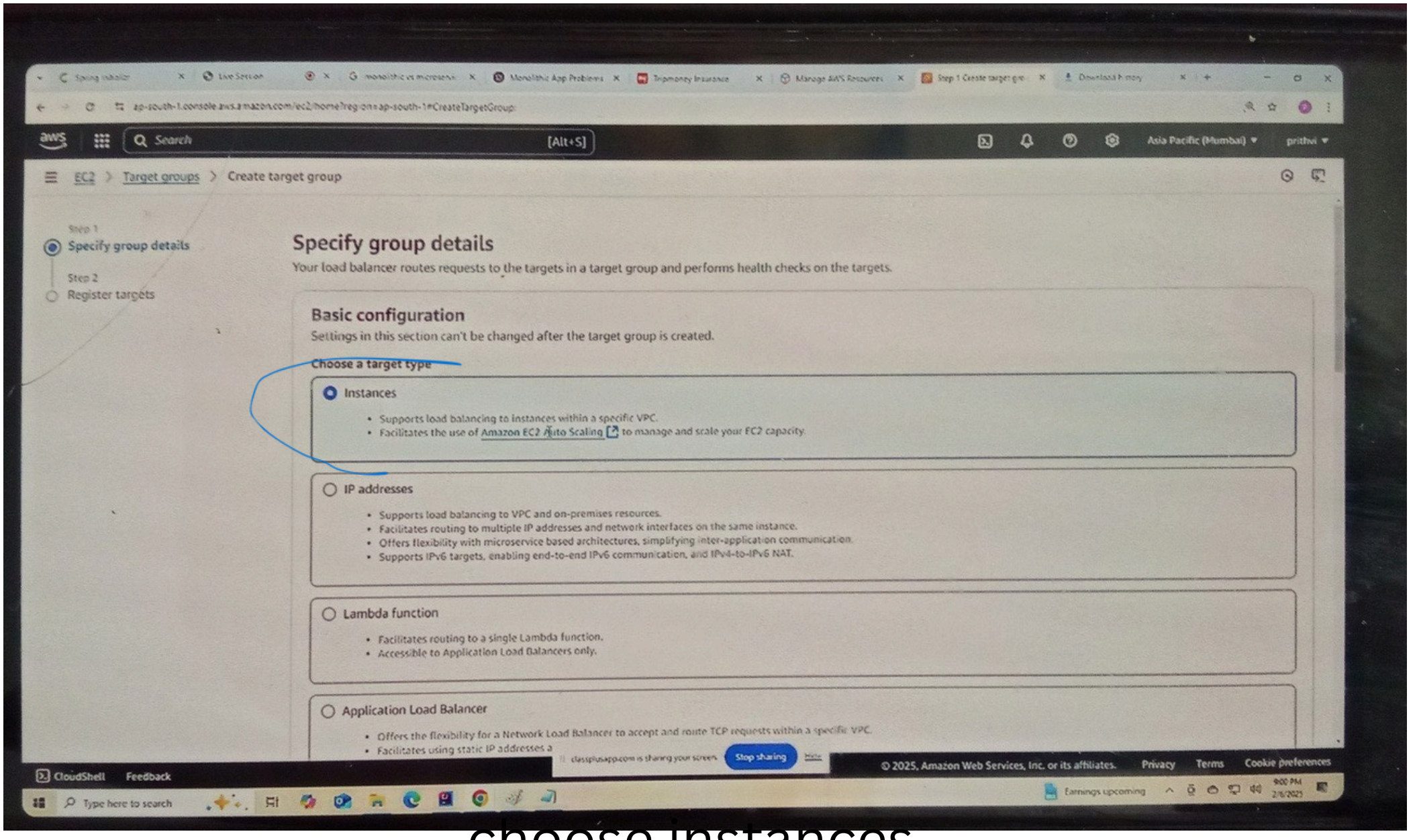


Now see above 4 instance is created

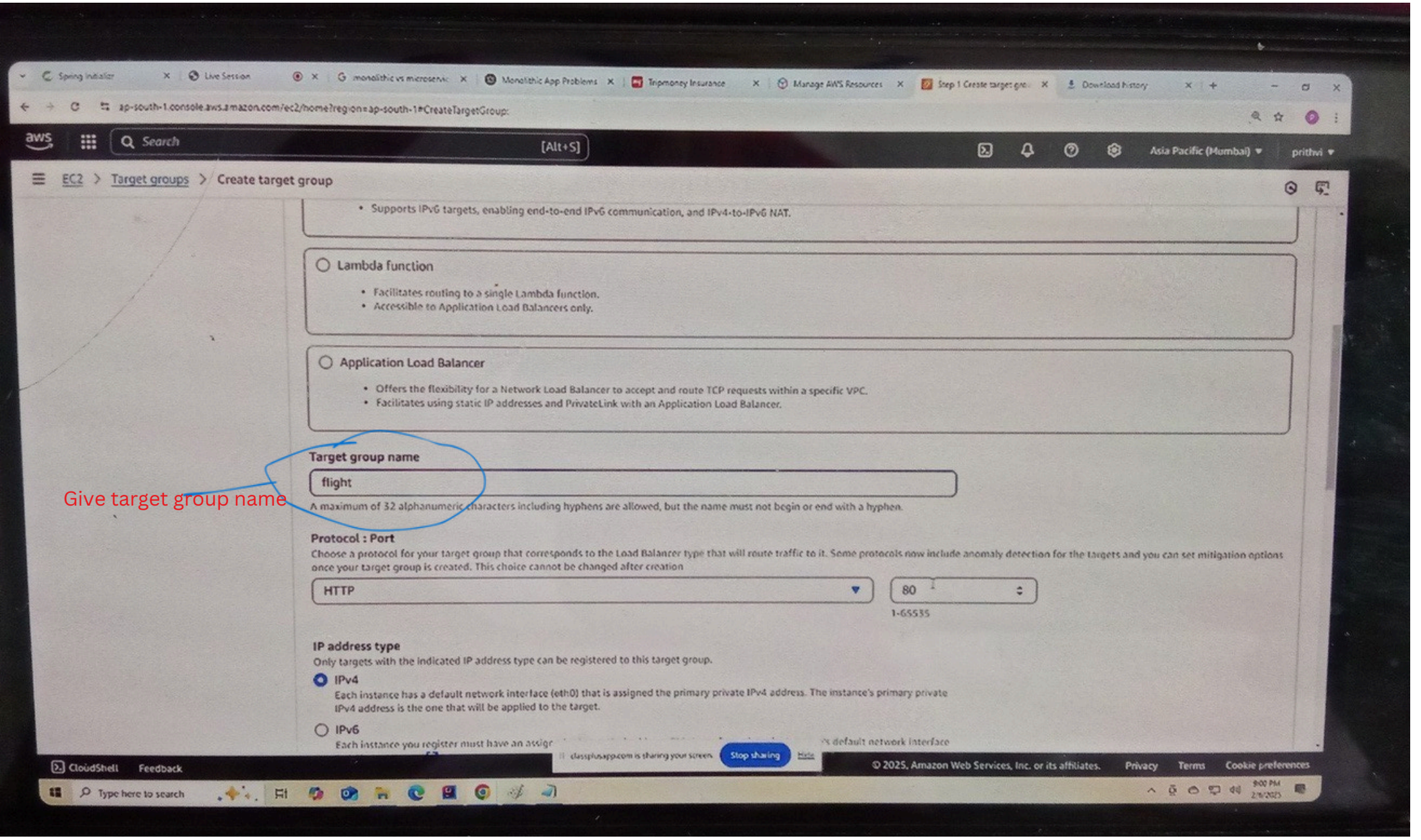
Create Target group of flight

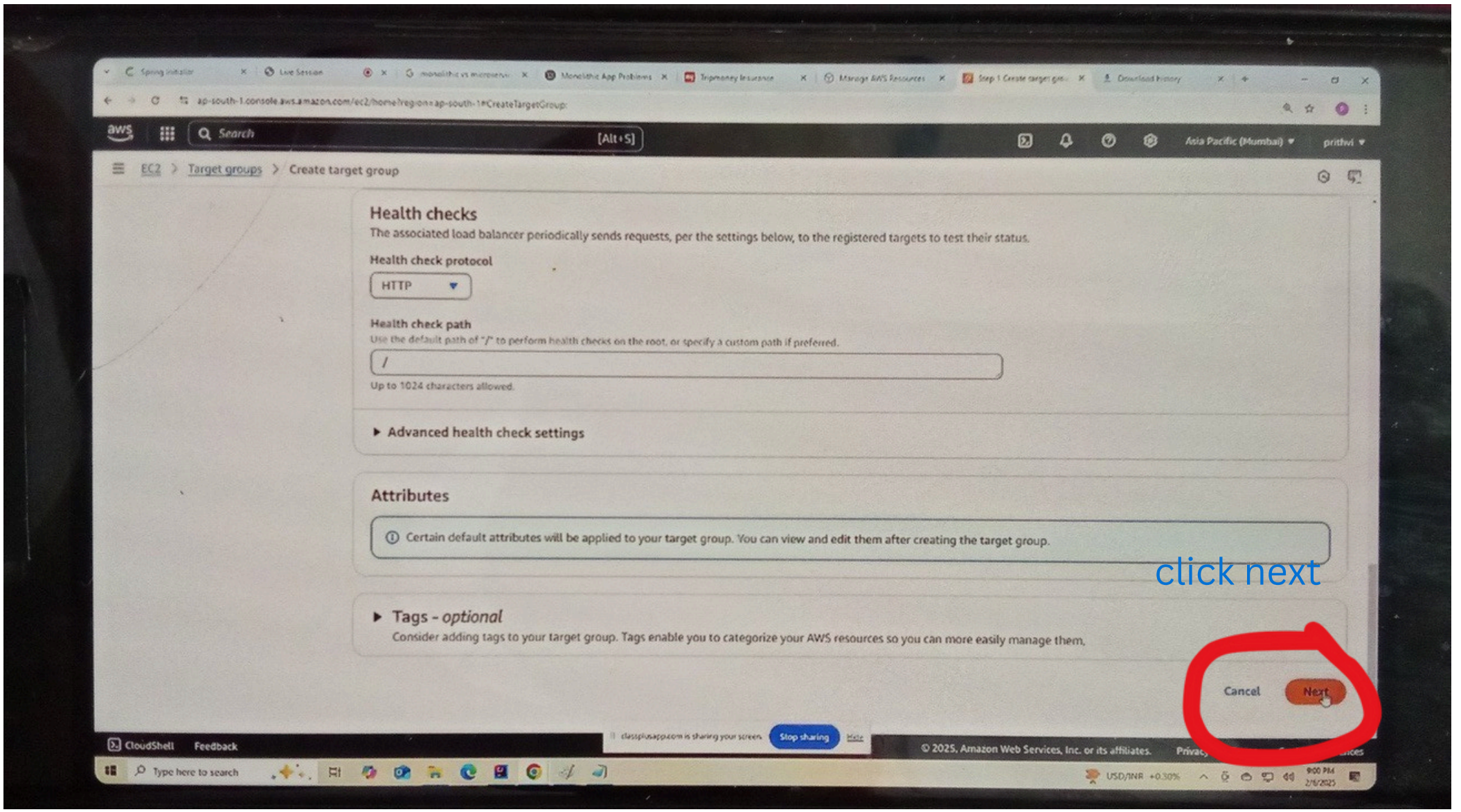



Search
[Alt+S]
EC2 > Target groups
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Load Balancers
Target Groups
Trust Stores New
Auto Scaling
Auto Scaling Groups
Settings
Target groups Info
Actions
Create target group
Filter target groups
Name
ARN
Port
Protocol
Target type
Load balancer
VPC ID
No target groups
You don't have any target groups in ap-south-1
Create target group
0 target groups selected
Select a target group above.
Click on Create Target Group
Stop sharing
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

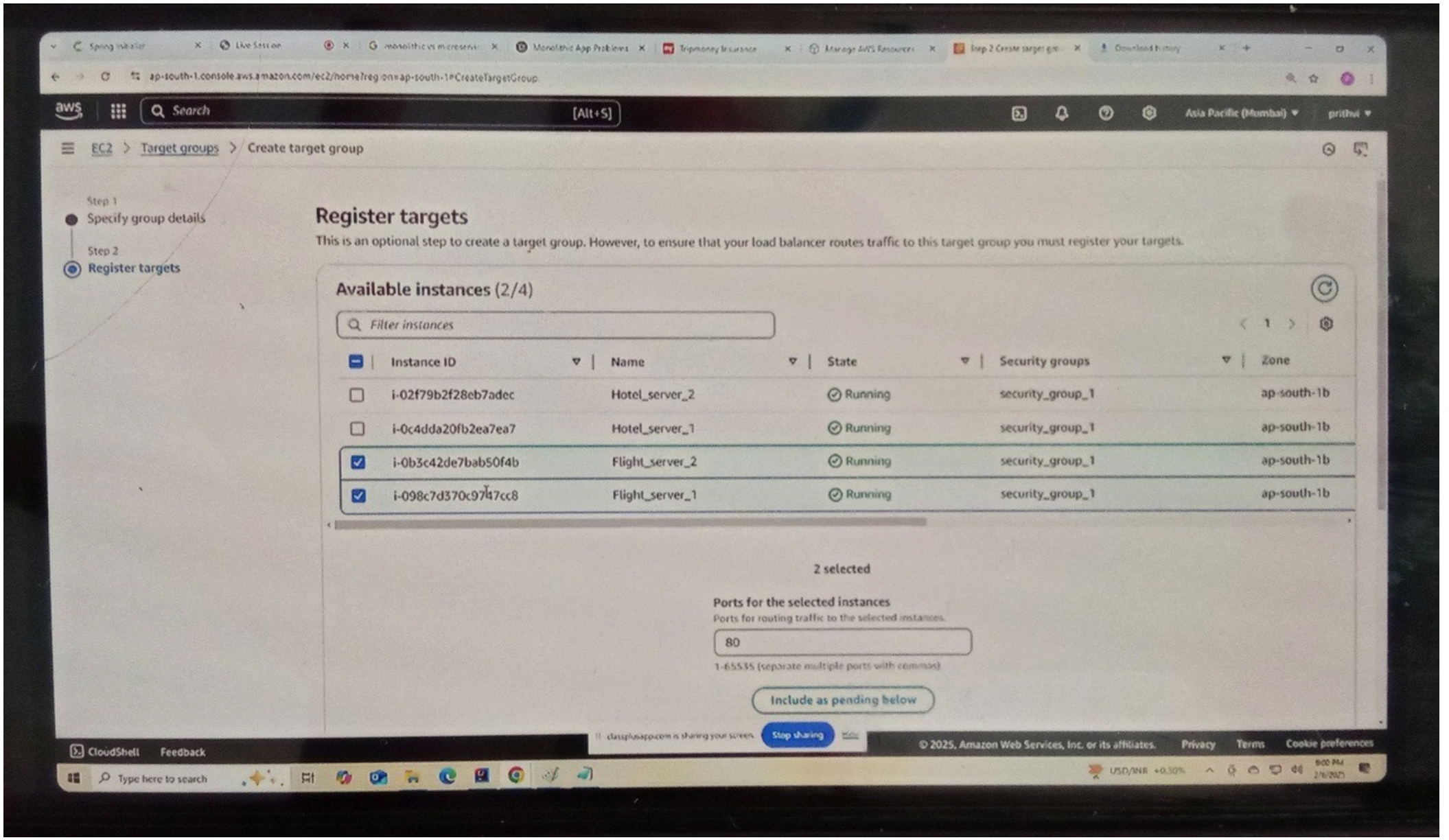


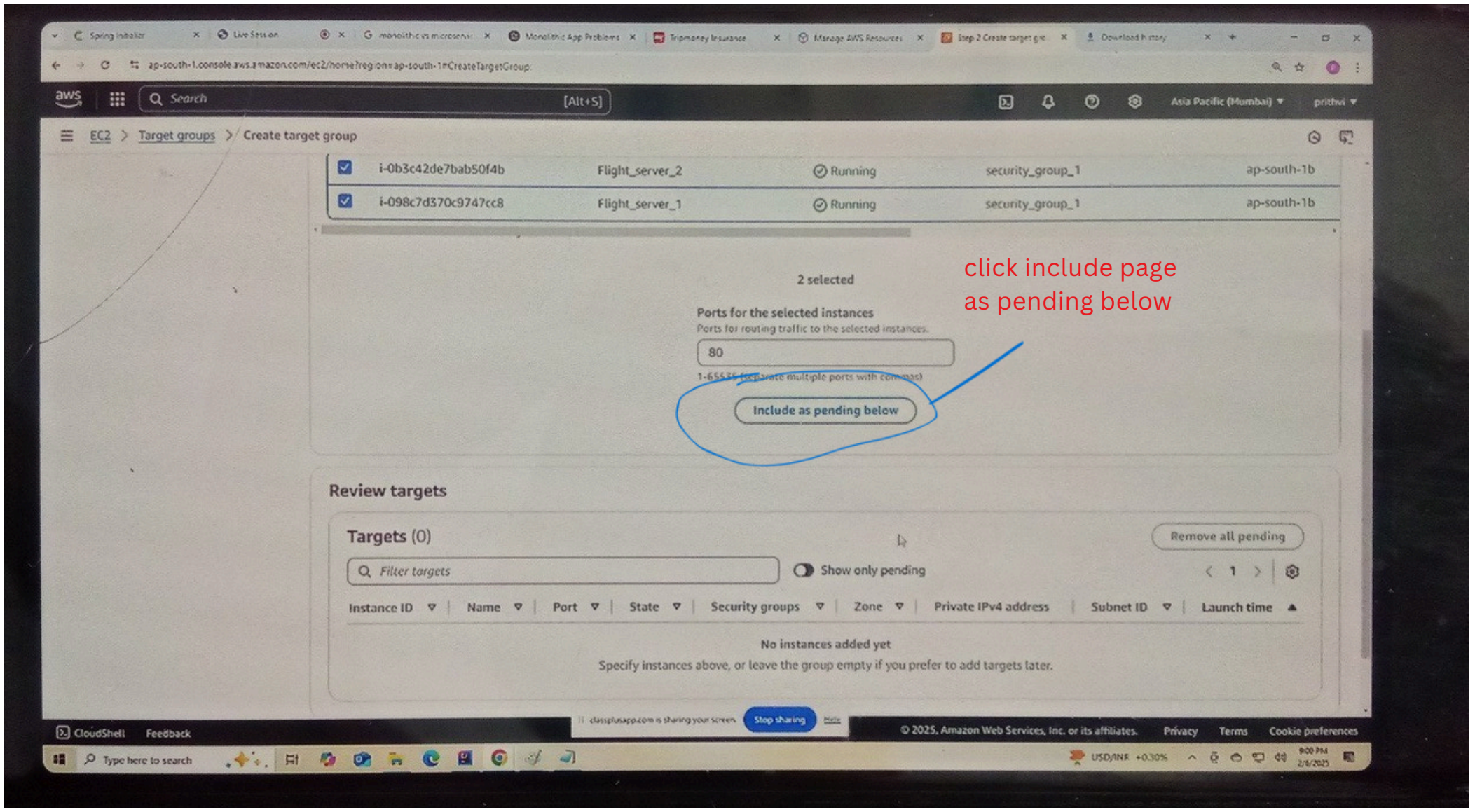
choose instances

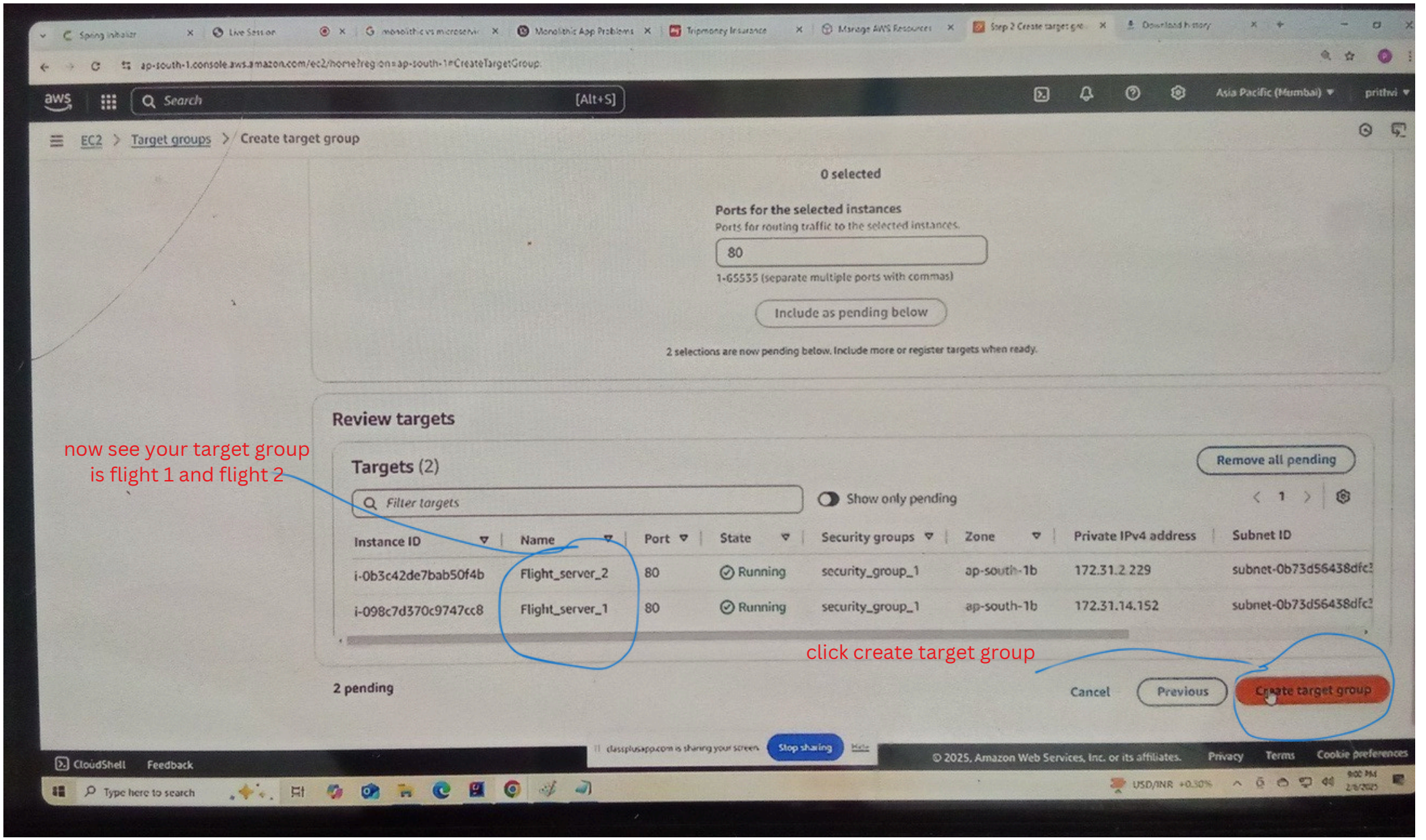


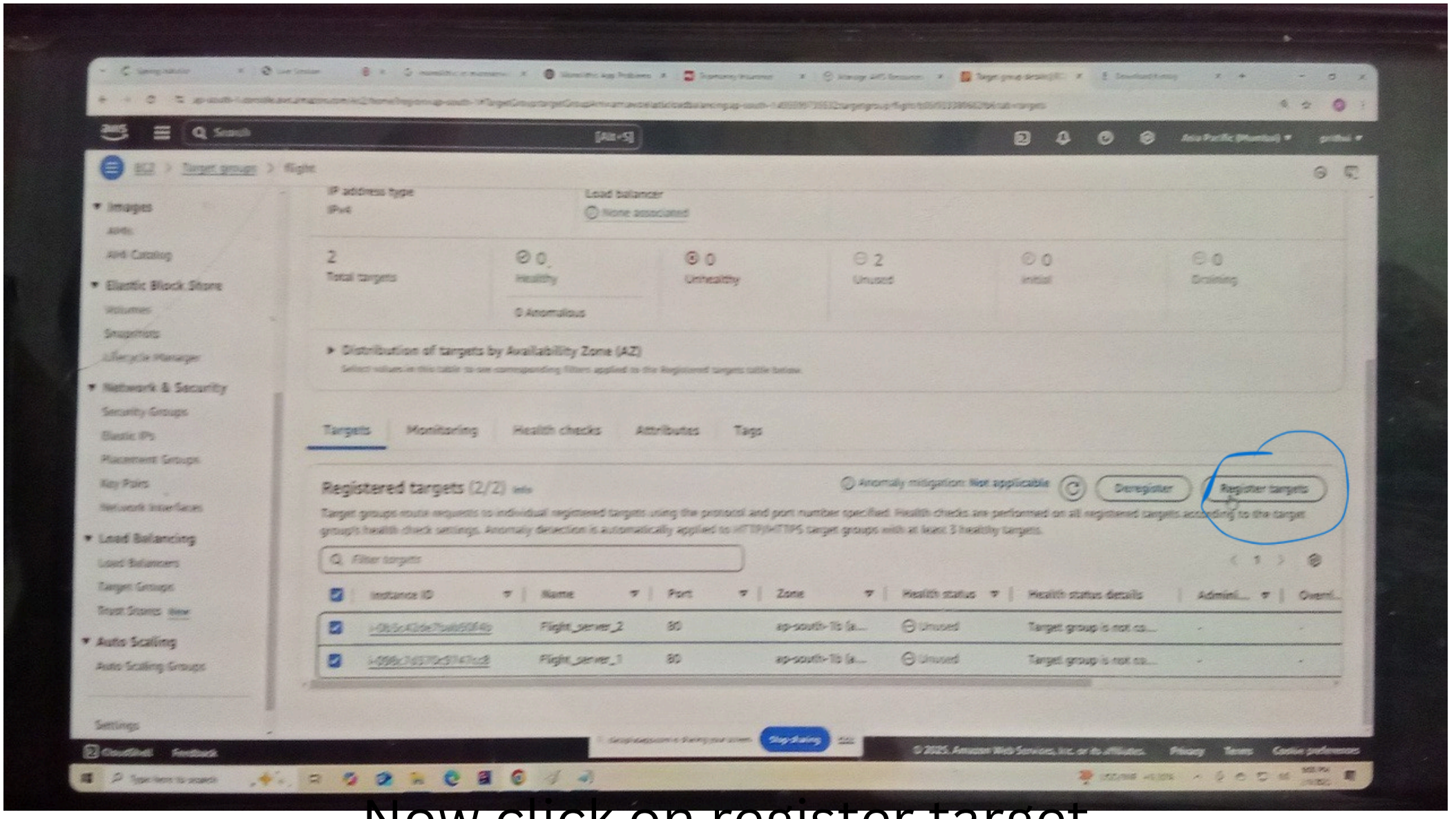


create seperate group of flight

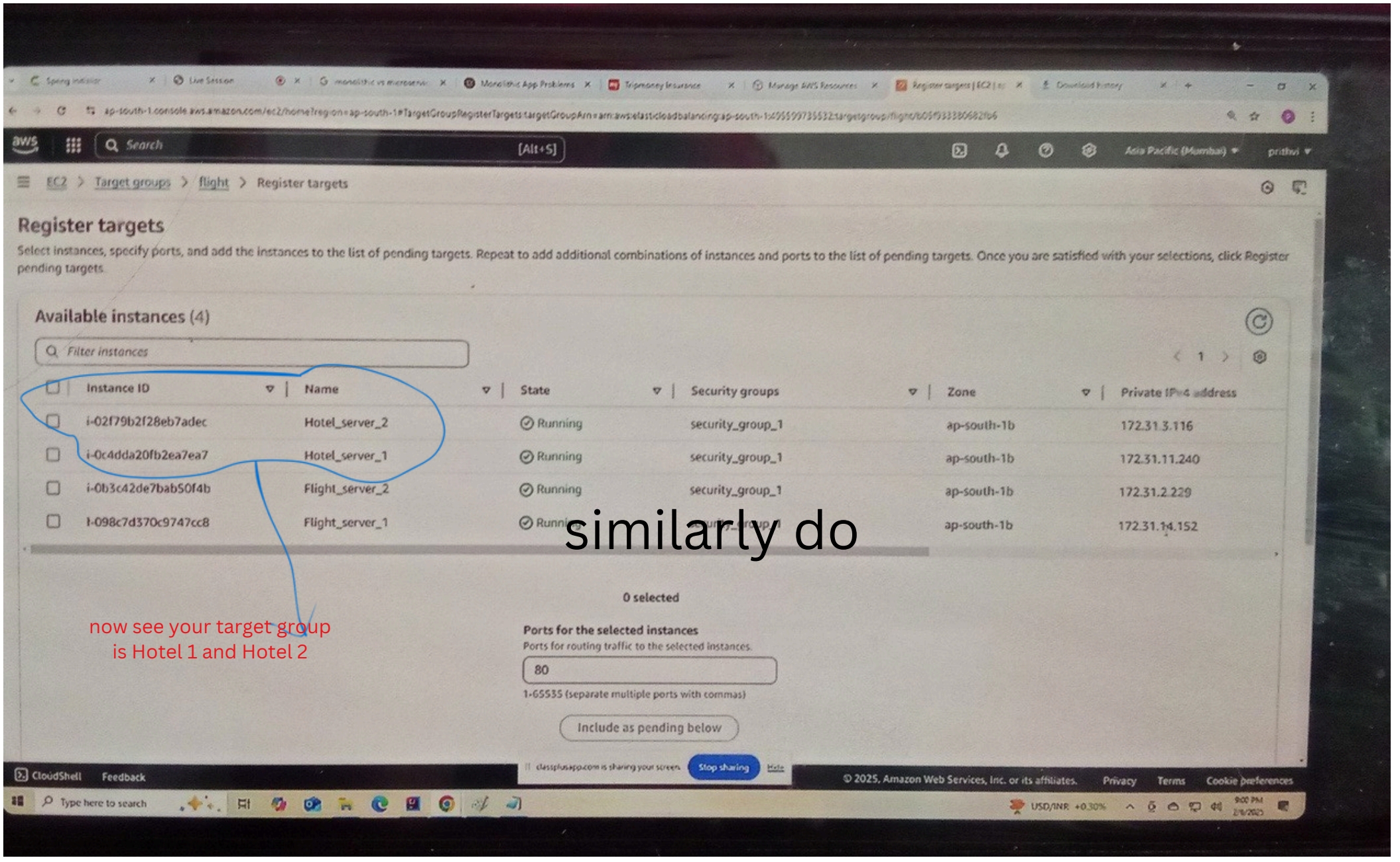


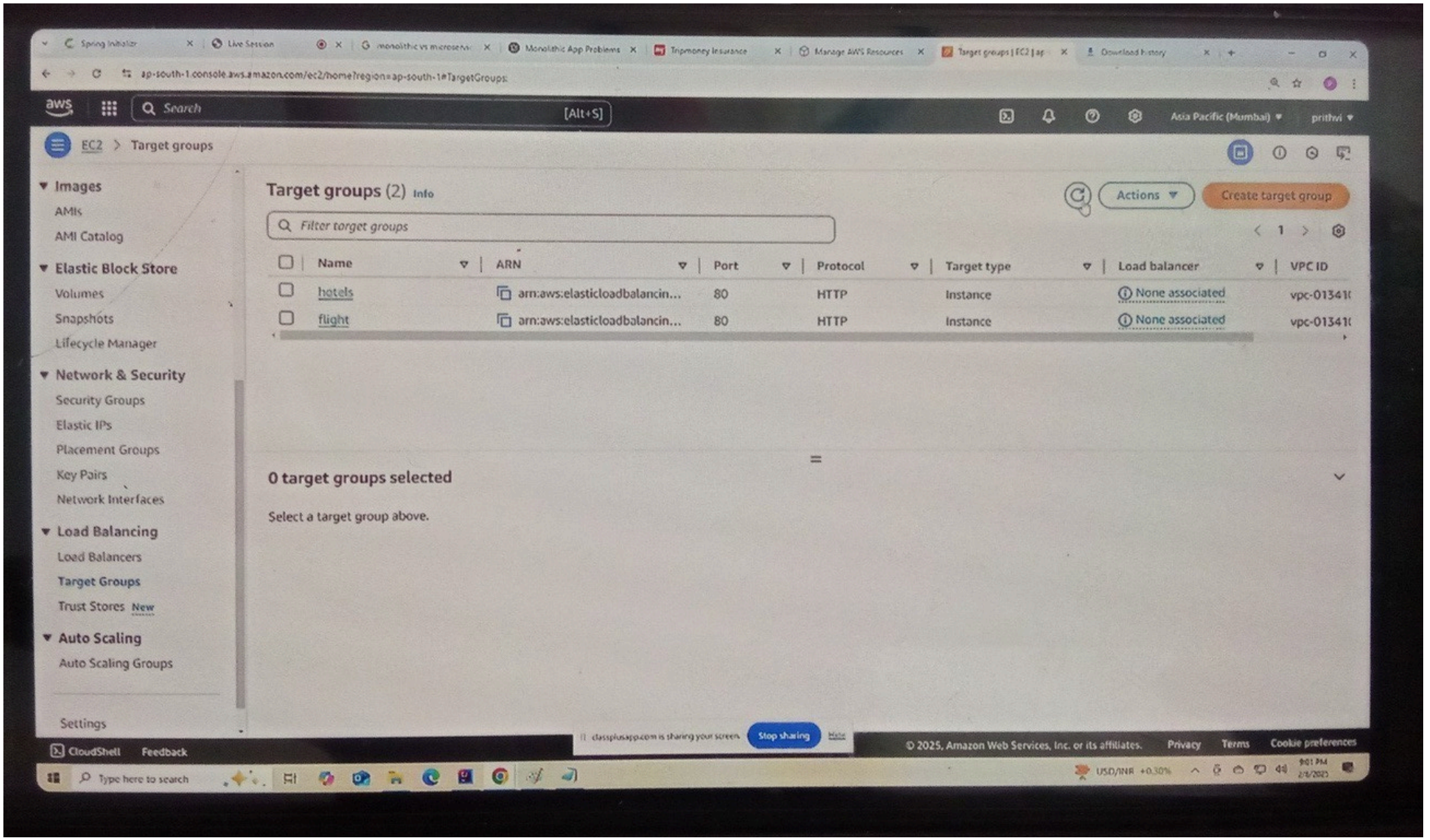

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateTargetGroup:
Search
[Alt+S]
Asia Pacific (Mumbai)
prithvi
EC2 > Target groups > Create target group
i-0b3c42de7bab50f4b Flight_server_2 Running security_group_1 ap-south-1b
i-098c7d370c9747cc8 Flight_server_1 Running security_group_1 ap-south-1b
2 selected
Ports for the selected instances
Ports for routing traffic to the selected instances.
80
Include as pending below
click include page as pending below
Review targets
Targets (0)
Remove all pending
Filter targets
Show only pending
Instance ID
Name
Port
State
Security groups
Zone
Private IPv4 address
Subnet ID
Launch time
No instances added yet
Specify instances above, or leave the group empty if you prefer to add targets later.
CloudShell
Feedback
classplusapp.com is sharing your screen.
Stop sharing
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
USD/INR +0.30%
9:00 PM
2/6/2025


ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateTargetGroup:
aws
Search
[Alt+S]
Asia Pacific (Mumbai)
prithvi
EC2 > Target groups > Create target group
0 selected
Ports for the selected instances
Ports for routing traffic to the selected instances.
80
1-65535 (separate multiple ports with commas)
Include as pending below
2 selections are now pending below. Include more or register targets when ready.
Review targets
now see your target group is flight 1 and flight 2
Targets (2)
Remove all pending
Filter targets
Show only pending
1
Instance ID
Name
Port
State
Security groups
Zone
Private IPv4 address
Subnet ID
i-0b3c42de7bab50f4b
Flight_server_2
80
Running
security_group_1
ap-south-1b
172.31.2.229
i-098c7d370c9747cc8
Flight_server_1
80
Running
security_group_1
ap-south-1b
172.31.14.152
click create target group
2 pending
Cancel
Previous
Create target group
CloudShell
Feedback
Stop sharing
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
USD/INR

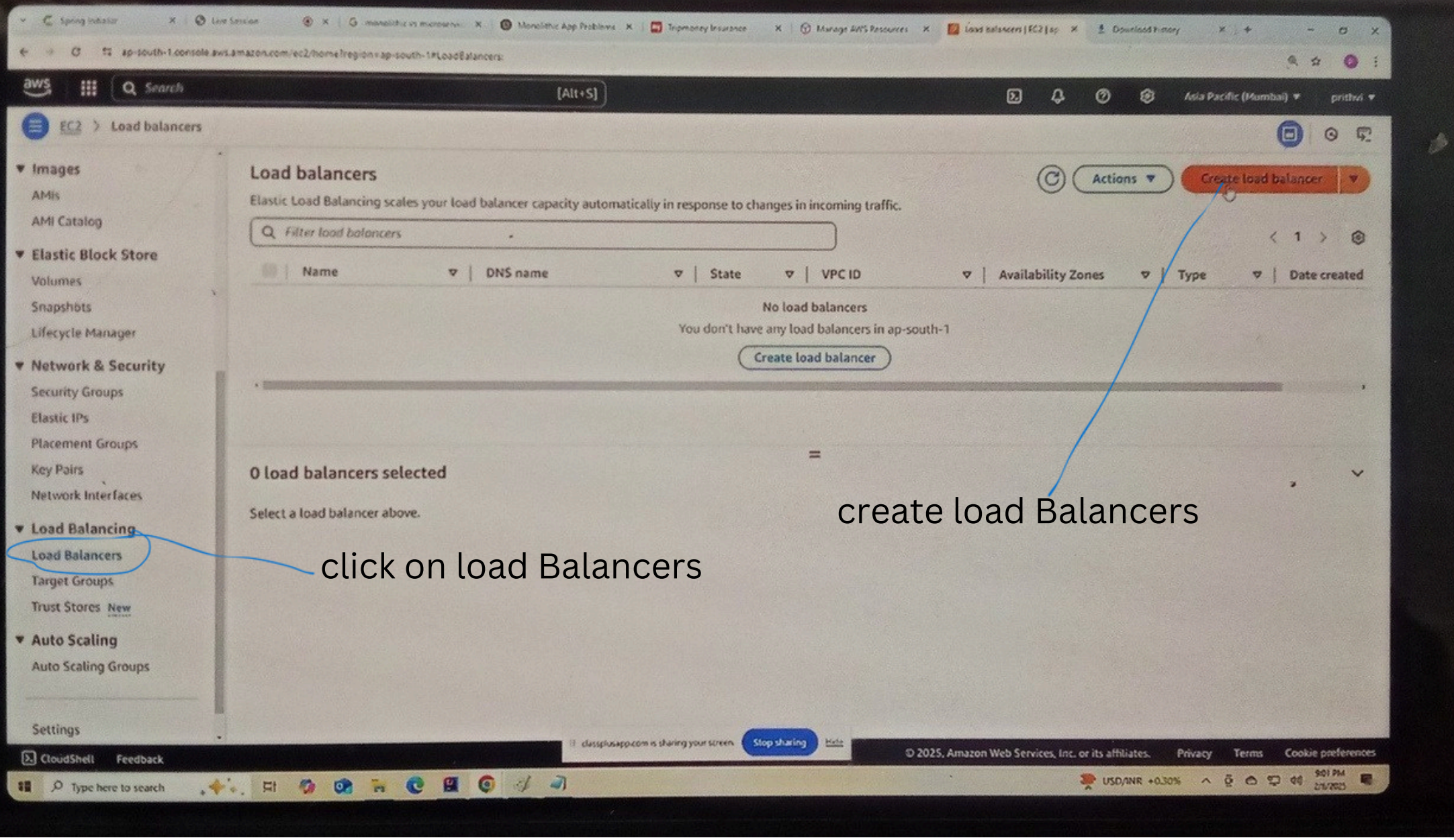


Now click on register target





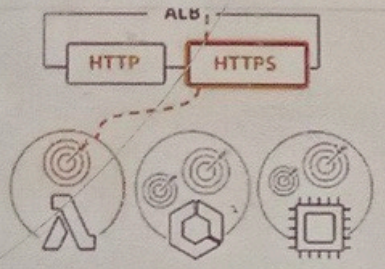
now see 2 target group is created


Load balancers
Actions
Create load balancer
Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.
Filter load balancers
Name
DNS name
State
VPC ID
Availability Zones
Type
Date created
No load balancers
You don't have any load balancers in ap-south-1
Create load balancer
0 load balancers selected
Select a load balancer above.
create load Balancers
click on load Balancers
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Load Balancers
Target Groups
Trust Stores New
Auto Scaling
Auto Scaling Groups
Settings
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#SelectCreateELBWizard:

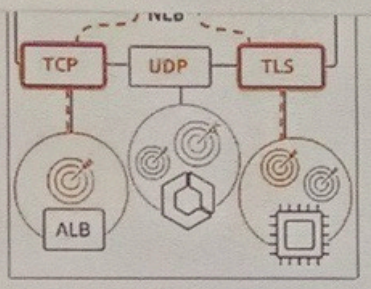
aws | Search [Alt+S] | Asia Pacific (Mumbai) | prithvi

EC2 > Load balancers > Compare and select load balancer type



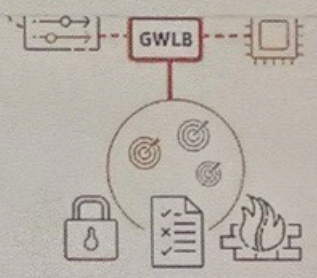
Choose an Application Load Balancer when you need a flexible feature set for your applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing and visibility features targeted at application architectures, including microservices and containers.

Create



Choose a Network Load Balancer when you need ultra-high performance, TLS offloading at scale, centralized certificate deployment, support for UDP, and static IP addresses for your applications. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second securely while maintaining ultra-low latencies.

Create



Choose a Gateway Load Balancer when you need to deploy and manage a fleet of third-party virtual appliances that support GENEVE. These appliances enable you to improve security, compliance, and policy controls.

Create

▶ Classic Load Balancer - *previous generation*

click on create

Close

classplusapp.com is sharing your screen. Stop sharing Hide

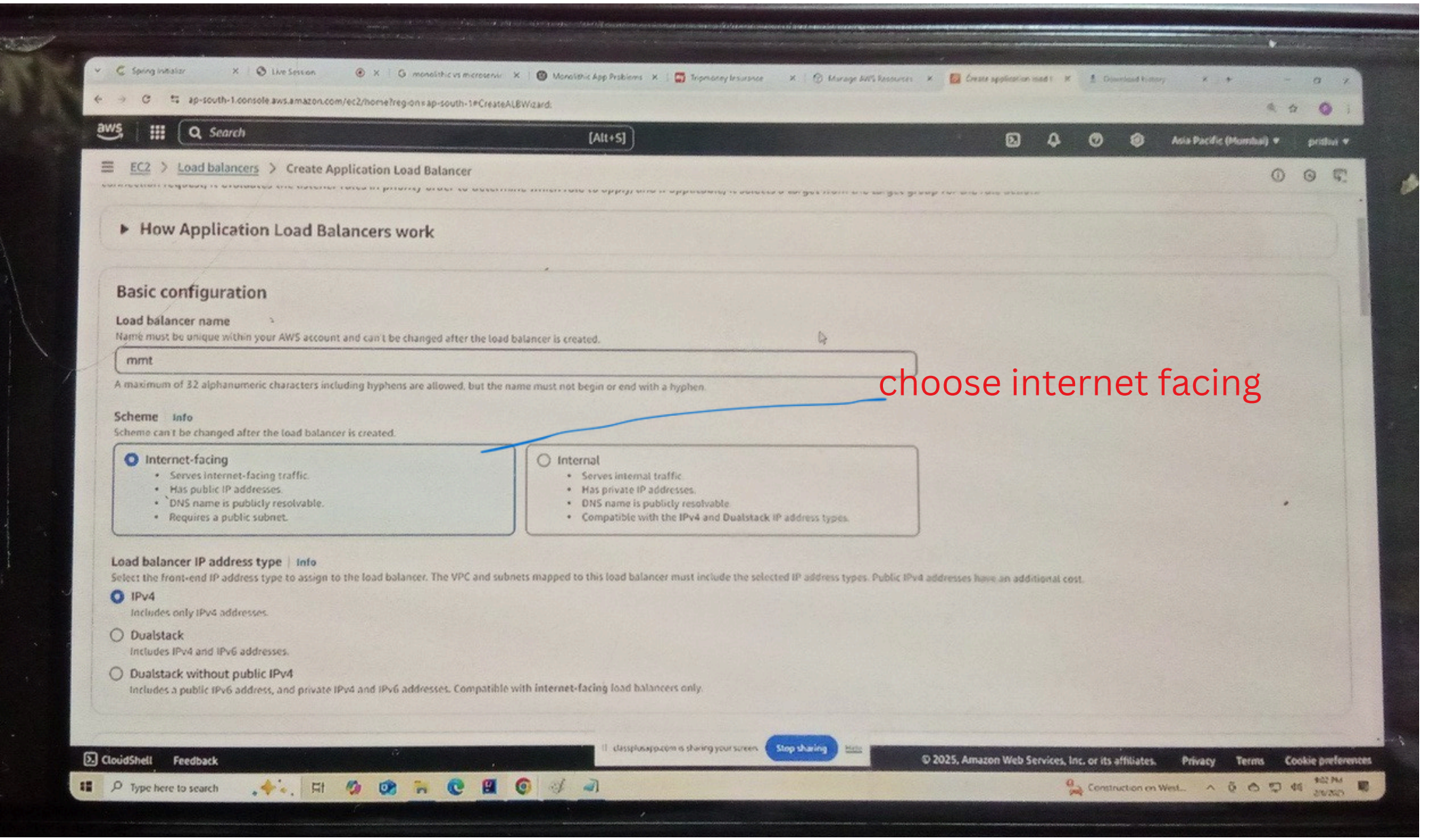
CloudShell Feedback

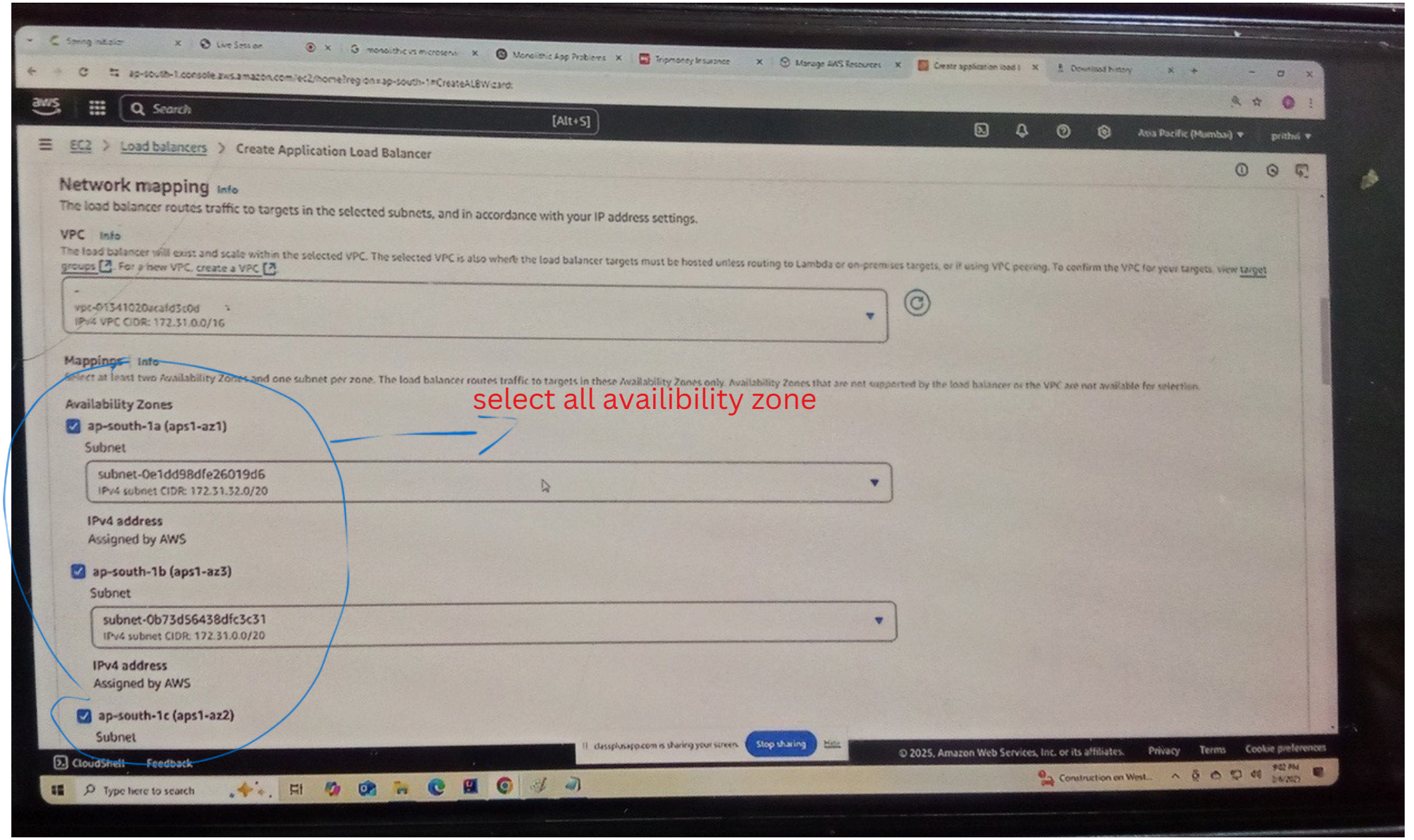
© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

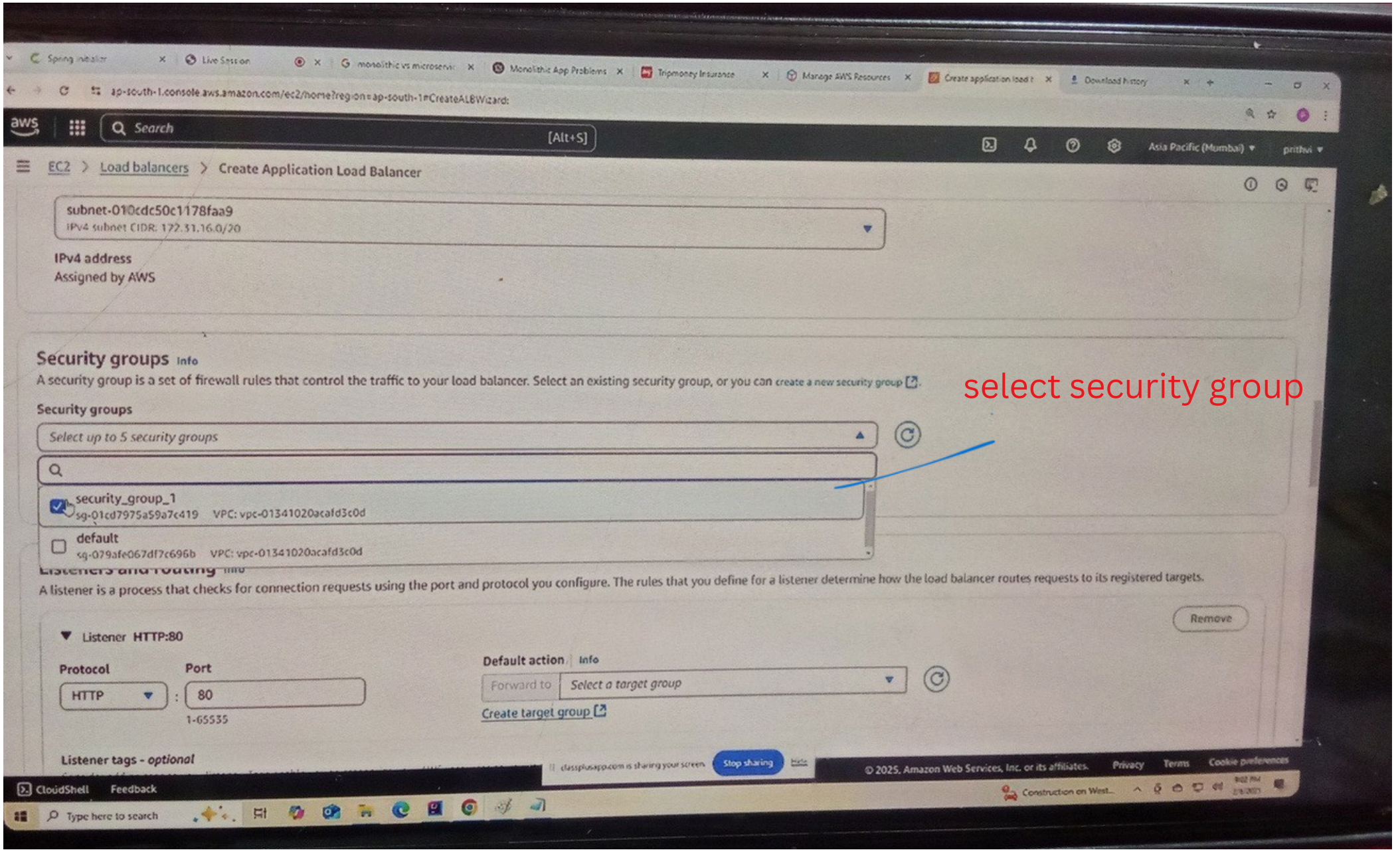
Type here to search

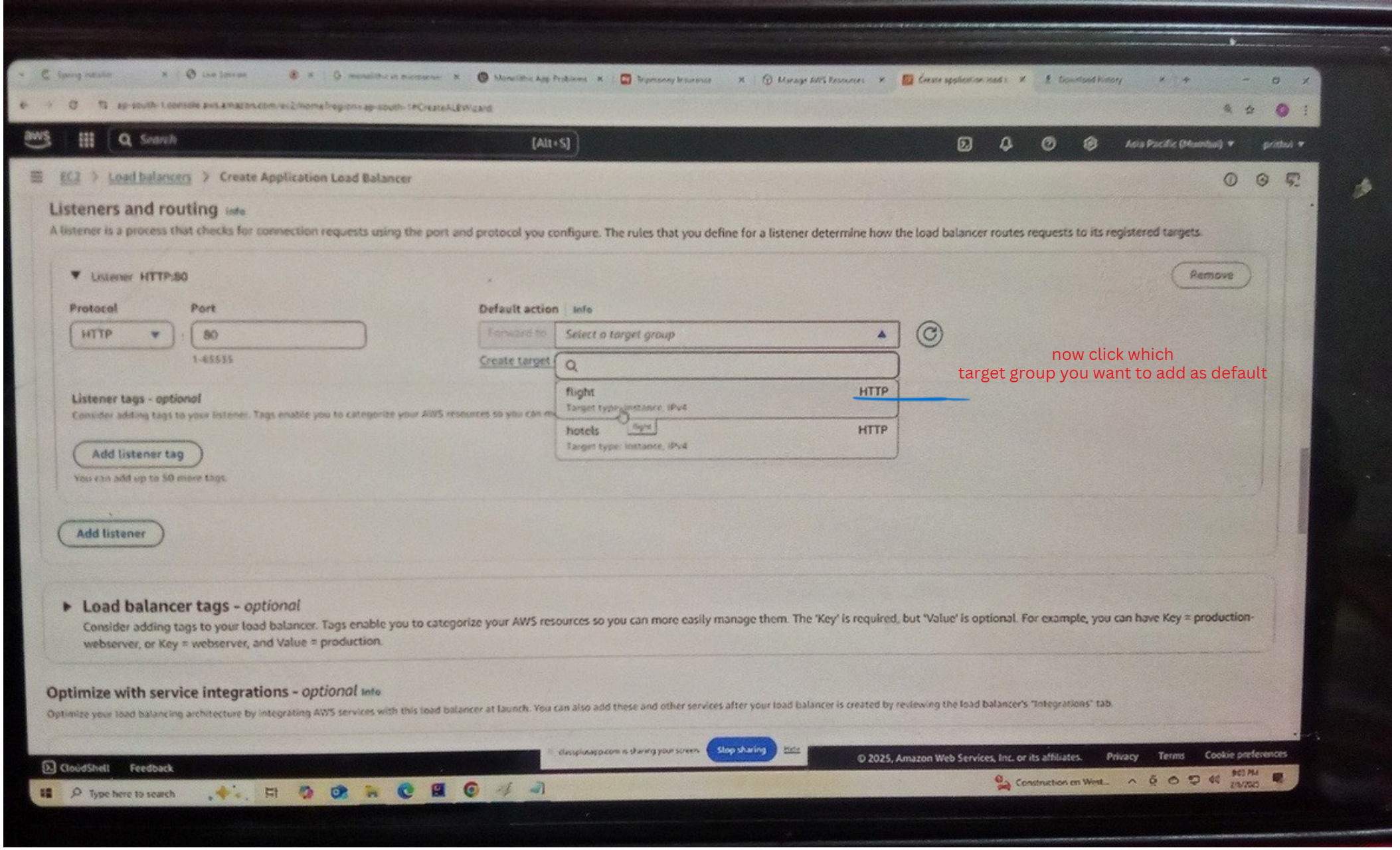
USD/INR +0.30%

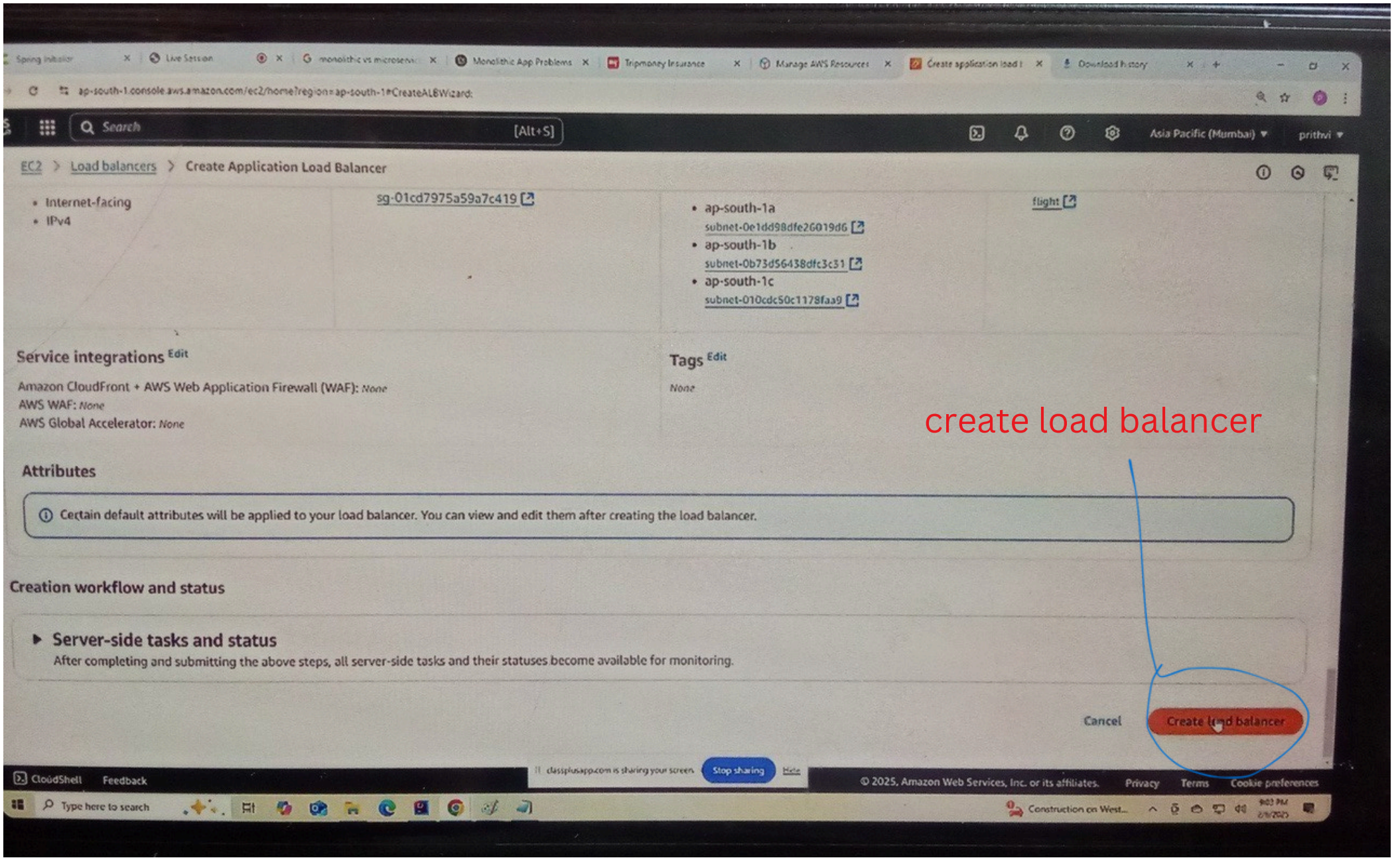
9:02 PM 2/6/2025

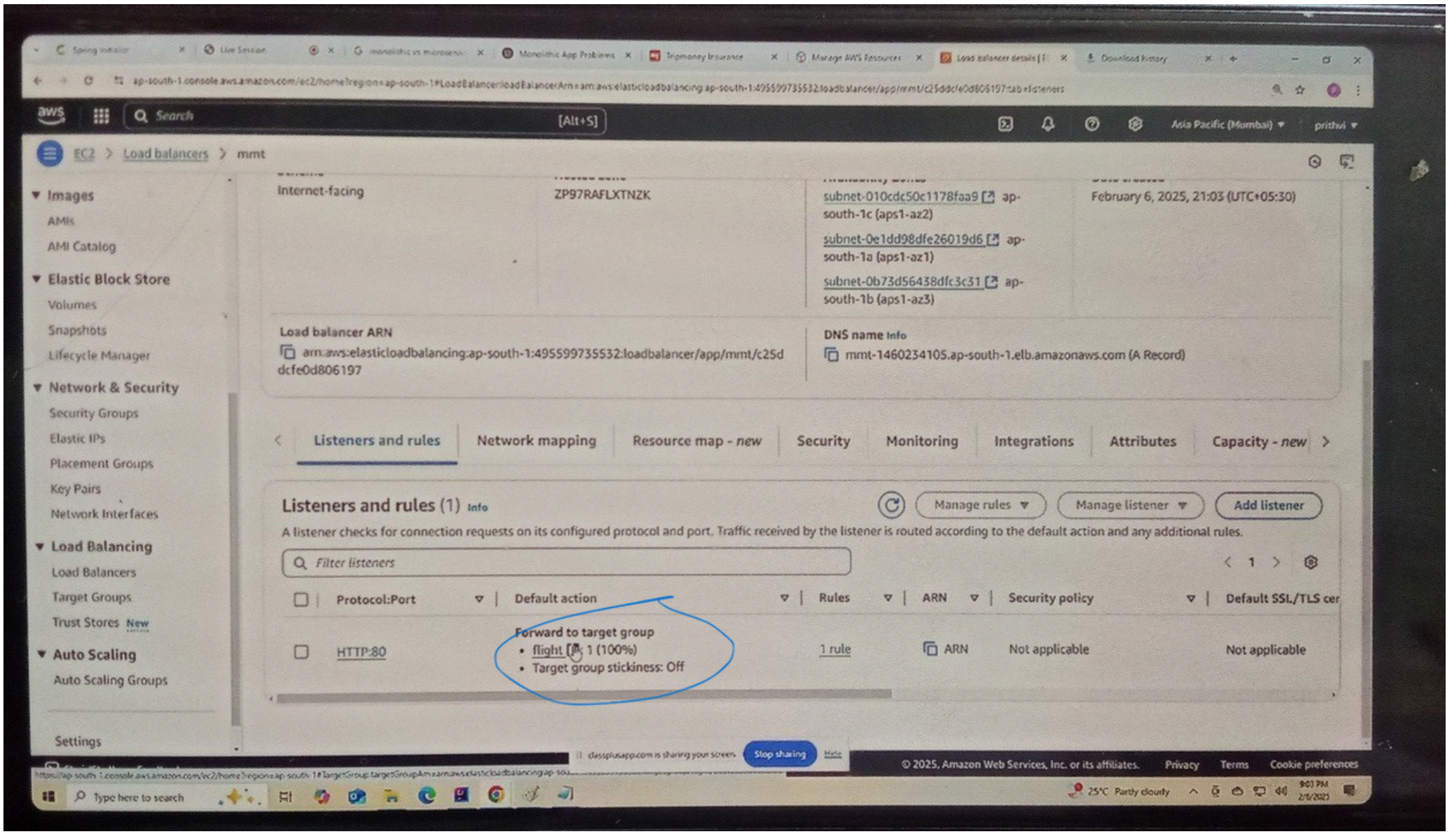

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateALBWizard:
Search
[Alt+S]
Asia Pacific (Mumbai)
EC2 > Load balancers > Create Application Load Balancer
How Application Load Balancers work
Basic configuration
Load balancer name
Name must be unique within your AWS account and can't be changed after the load balancer is created.
mmt
A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.
choose internet facing
Scheme Info
Scheme can't be changed after the load balancer is created.
Internet-facing
Serves internet-facing traffic.
Has public IP addresses.
DNS name is publicly resolvable.
Requires a public subnet.
Internal
Serves internal traffic.
Has private IP addresses.
DNS name is publicly resolvable.
Compatible with the IPv4 and Dualstack IP address types.
Load balancer IP address type Info
Select the front-end IP address type to assign to the load balancer. The VPC and subnets mapped to this load balancer must include the selected IP address types. Public IPv4 addresses have an additional cost.
IPv4
Includes only IPv4 addresses.
Dualstack
Includes IPv4 and IPv6 addresses.
Dualstack without public IPv4
Includes a public IPv6 address, and private IPv4 and IPv6 addresses. Compatible with internet-facing load balancers only.
CloudShell
Feedback
Stop sharing
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search

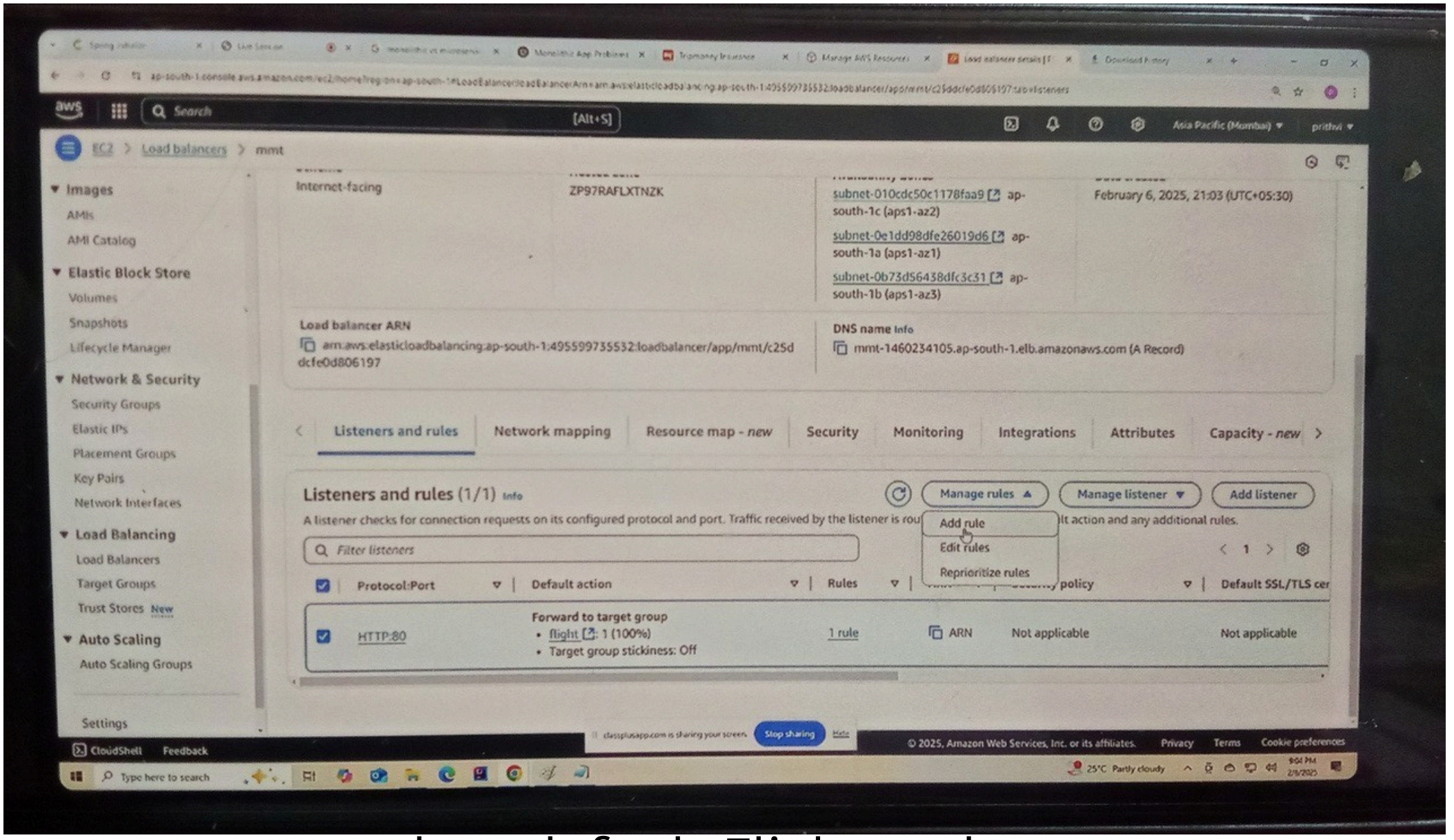

select all availibility zone
Network mapping Info
The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.
VPC Info
Availability Zones
ap-south-1a (aps1-az1)
Subnet
subnet-0e1dd98dfe26019d6
IPv4 subnet CIDR: 172.31.32.0/20
IPv4 address
Assigned by AWS
ap-south-1b (aps1-az3)
Subnet
subnet-0b73d56438dfc3c31
IPv4 subnet CIDR: 172.31.0.0/20
IPv4 address
Assigned by AWS
ap-south-1c (aps1-az2)
Subnet


ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateALBWizard:
Search
[Alt+S]
Asia Pacific (Mumbai)
prithvi
EC2 > Load balancers > Create Application Load Balancer
subnet-010cdc50c1178faa9
IPv4 subnet CIDR: 172.31.16.0/20
IPv4 address
Assigned by AWS
Security groups Info
A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can create a new security group.
select security group
Security groups
Select up to 5 security groups
security_group_1
sg-01cd7975a59a7c419 VPC: vpc-01341020acafd3c0d
default
sg-079afe067df7c696b VPC: vpc-01341020acafd3c0d
A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.
Remove
Listener HTTP:80
Protocol
HTTP
Port
80
1-65535
Default action Info
Forward to
Select a target group
Create target group
Listener tags - optional
classplusapp.com is sharing your screen.
Stop sharing
Hide
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
CloudShell
Feedback
Construction on West...
Type here to search

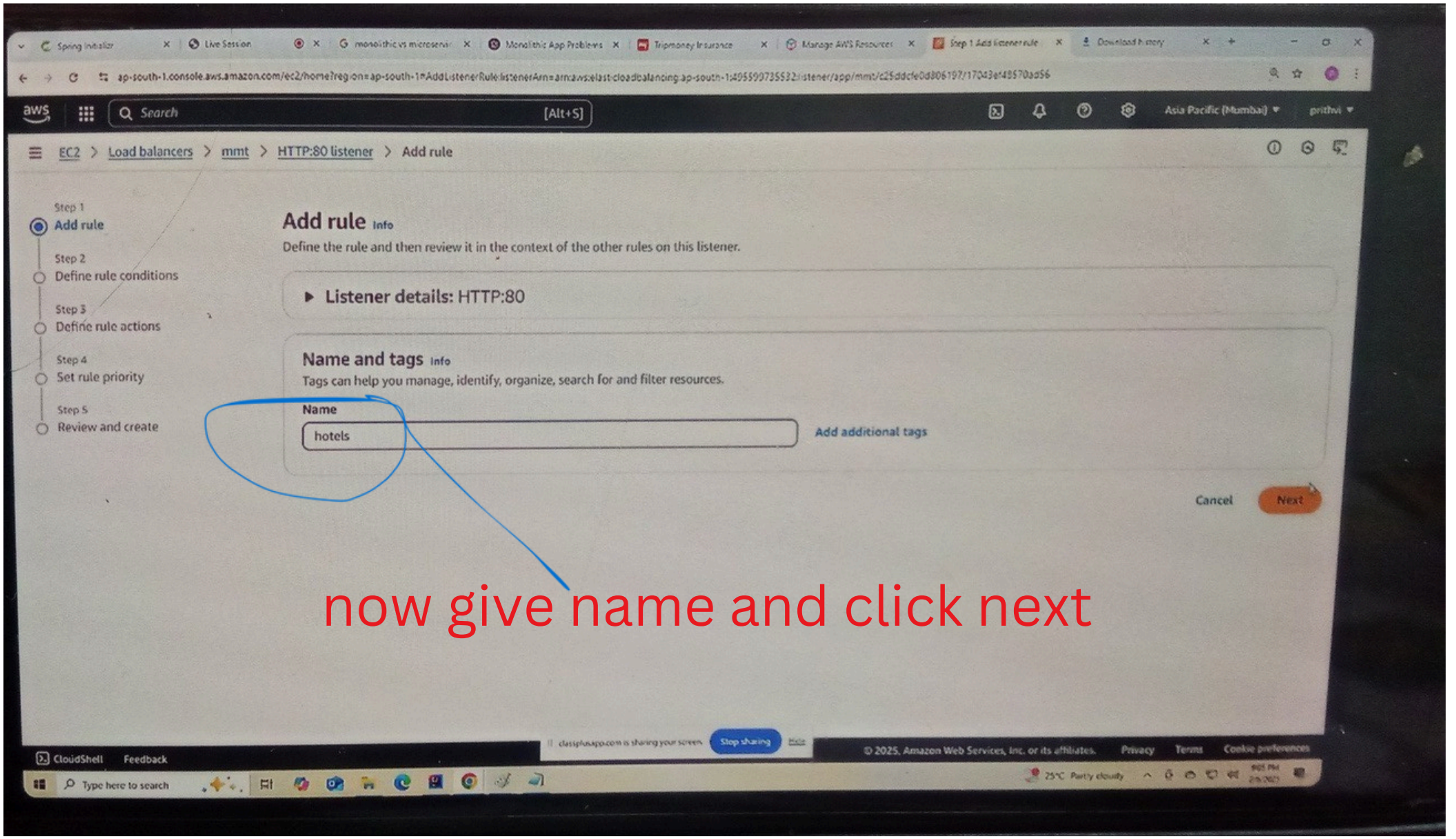

EC2 > Load balancers > Create Application Load Balancer
Listeners and routing info
A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.
Listener HTTP:80
Remove
Protocol
Port
HTTP
80
1-65535
Default action info
Forward to
Select a target group
Create target
flight
HTTP
Target type: instance, IPv4
hotels
HTTP
Target type: instance, IPv4
now click which
target group you want to add as default
Listener tags - optional
Add listener tag
Add listener
Load balancer tags - optional
Consider adding tags to your load balancer. Tags enable you to categorize your AWS resources so you can more easily manage them. The 'Key' is required, but 'Value' is optional. For example, you can have Key = production-webserver, or Key = webserver, and Value = production.
Optimize with service integrations - optional info
Optimize your load balancing architecture by integrating AWS services with this load balancer at launch. You can also add these and other services after your load balancer is created by reviewing the load balancer's "Integrations" tab.
Stop sharing
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

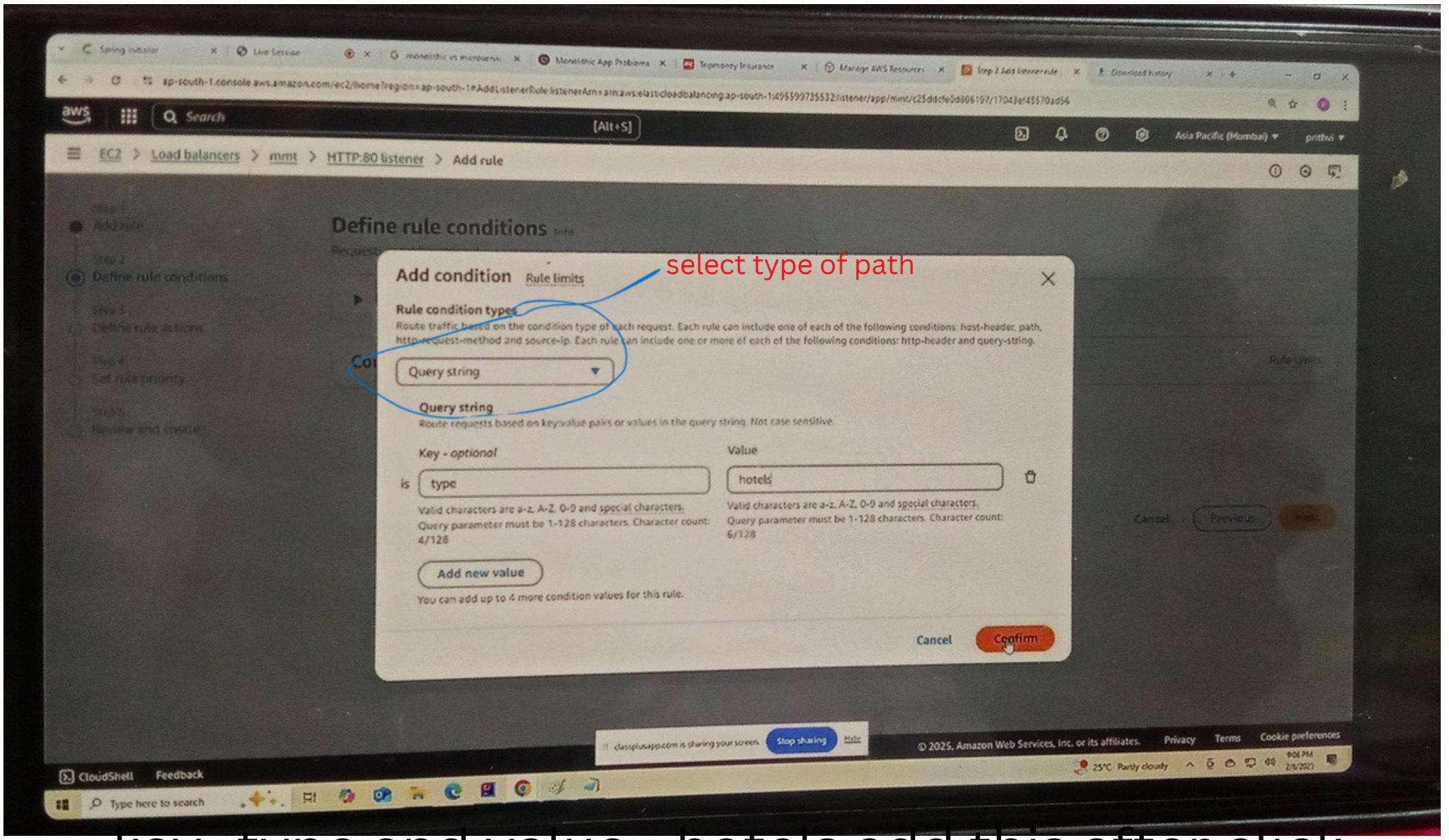

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#CreateALBWizard:
Search
[Alt+S]
Asia Pacific (Mumbai)
prithvi
EC2 > Load balancers > Create Application Load Balancer
Internet-facing
IPv4
sg-01cd7975a59a7c419
ap-south-1a
subnet-0e1dd98dfe2G019d6
ap-south-1b
subnet-0b73d56438dfc3c31
ap-south-1c
subnet-010cdc50c1178faa9
flight
Service integrations Edit
Amazon CloudFront + AWS Web Application Firewall (WAF): None
AWS WAF: None
AWS Global Accelerator: None
Tags Edit
None
create load balancer
Attributes
Certain default attributes will be applied to your load balancer. You can view and edit them after creating the load balancer.
Creation workflow and status
Server-side tasks and status
After completing and submitting the above steps, all server-side tasks and their statuses become available for monitoring.
Cancel
Create load balancer
CloudShell
Feedback
classplusapp.com is sharing your screen.
Stop sharing
Hide
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
Construction on West...


aws
Search
[Alt+S]
Asia Pacific (Mumbai)
prithvi
EC2 > Load balancers > mmt
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Load Balancers
Target Groups
Trust Stores New
Auto Scaling
Auto Scaling Groups
Settings
Internet-facing
ZP97RAFLXTNZK
subnet-010cdc50c1178faa9 ap-south-1c (aps1-az2)
subnet-0e1dd98dfe26019d6 ap-south-1a (aps1-az1)
subnet-0b73d56438dfc3c31 ap-south-1b (aps1-az3)
February 6, 2025, 21:03 (UTC+05:30)
Load balancer ARN
arn:aws:elasticloadbalancing:ap-south-1:495599735532:loadbalancer/app/mmt/c25ddcfe0d806197
DNS name Info
mmt-1460234105.ap-south-1.elb.amazonaws.com (A Record)
Listeners and rules
Network mapping
Resource map - new
Security
Monitoring
Integrations
Attributes
Capacity - new
Listeners and rules (1) Info
Manage rules
Manage listener
Add listener
A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.
Filter listeners
Protocol:Port
Default action
Rules
ARN
Security policy
Default SSL/TLS cer
HTTP:80
Forward to target group
flight: 1 (100%)
Target group stickiness: Off
1 rule
ARN
Not applicable
Not applicable
classplusapp.com is sharing your screen.
Stop sharing
Hide
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
25°C Partly cloudy
9:03 PM
2/6/2025

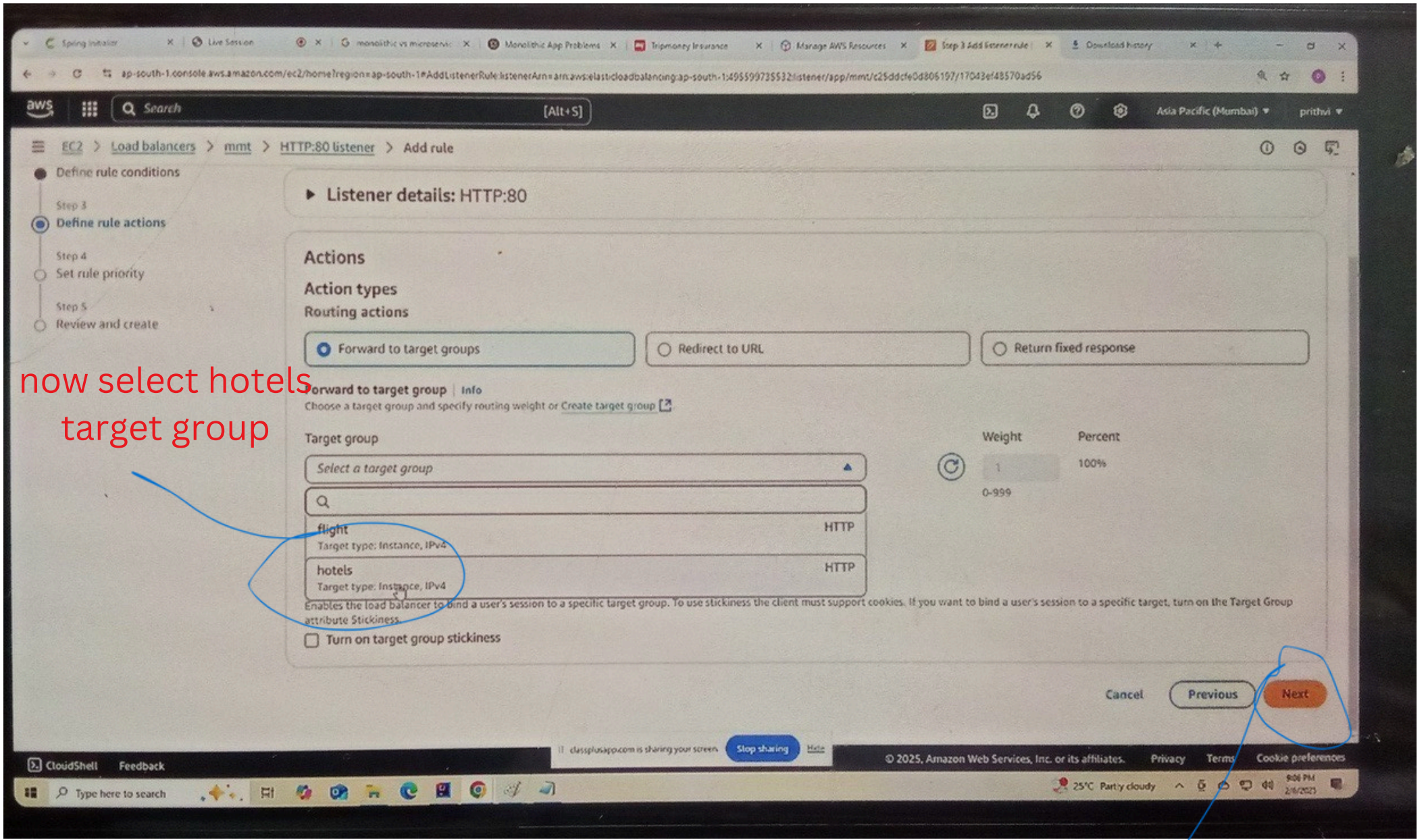


select default Flight and
after that select add rule


Search [Alt+S]
Asia Pacific (Mumbai)
prithvi
EC2 > Load balancers > mmt > HTTP:80 listener > Add rule
Step 1
Add rule
Step 2
Define rule conditions
Step 3
Define rule actions
Step 4
Set rule priority
Step 5
Review and create
Add rule Info
Define the rule and then review it in the context of the other rules on this listener.
Listener details: HTTP:80
Name and tags Info
Tags can help you manage, identify, organize, search for and filter resources.
Name
hotels
Add additional tags
Cancel
Next
now give name and click next
CloudShell
Feedback
Stop sharing
Hide
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
25°C Partly cloudy

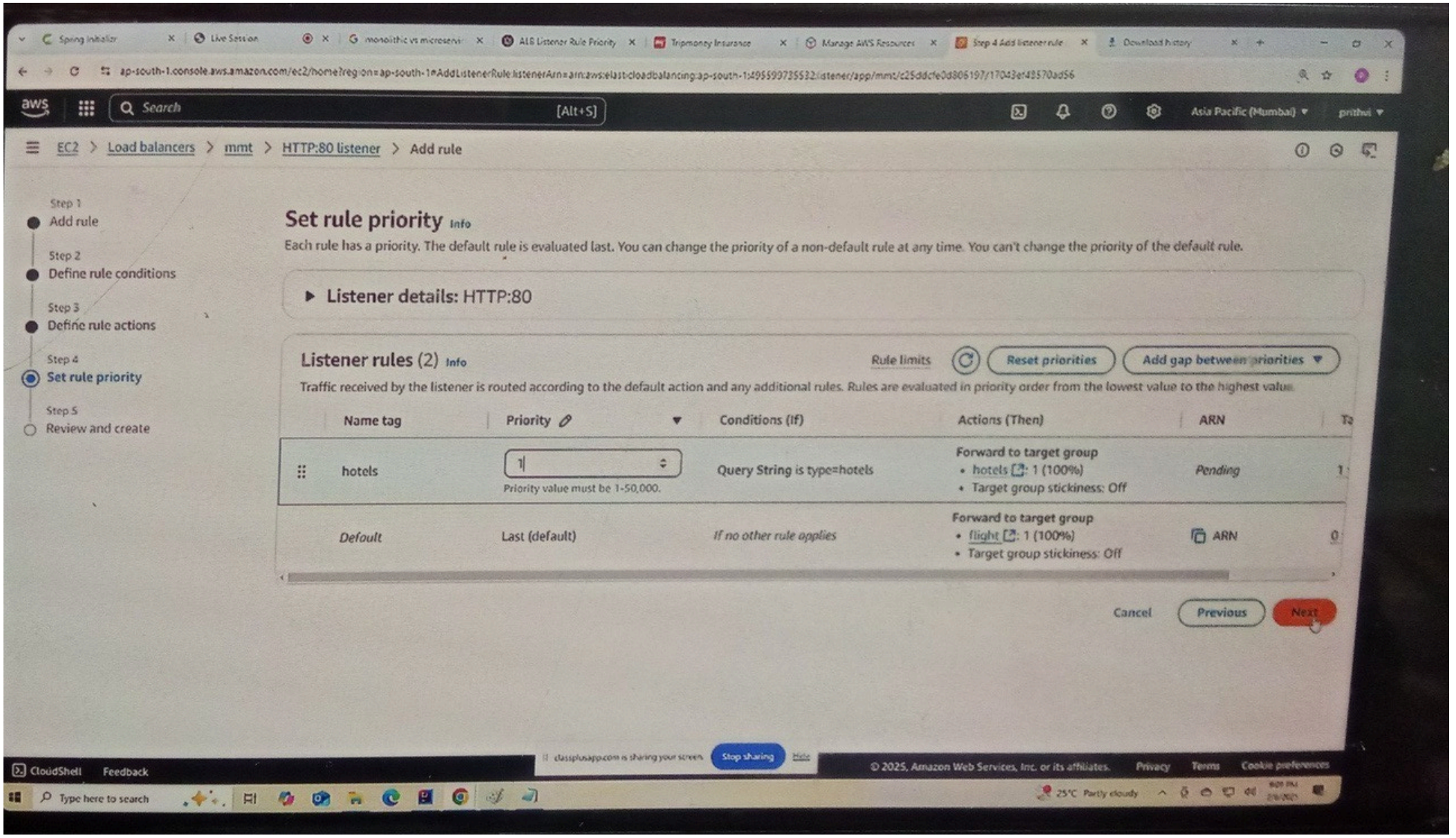


key -type and value - hotels add this after click confirm

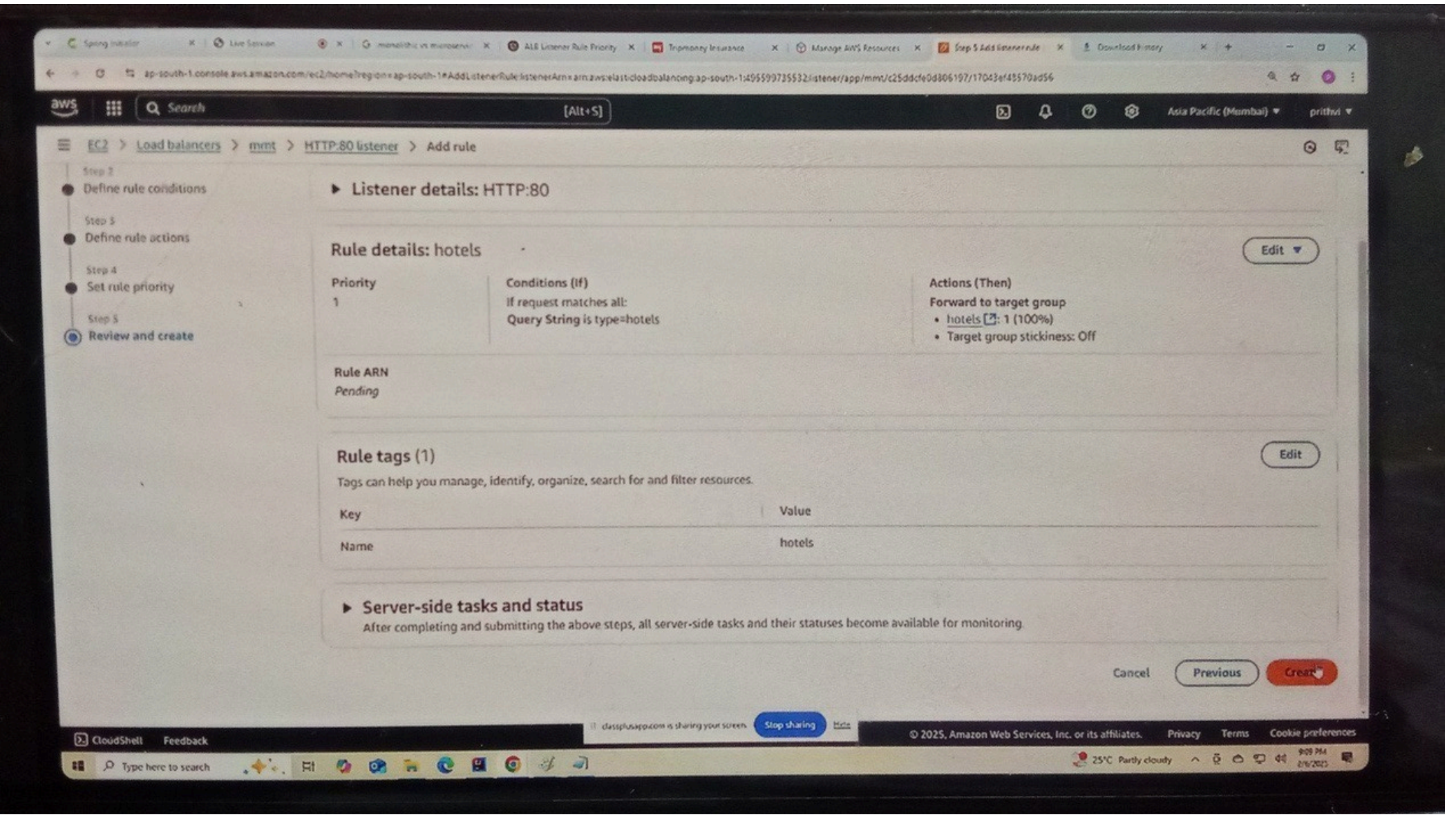


now select hotels target group

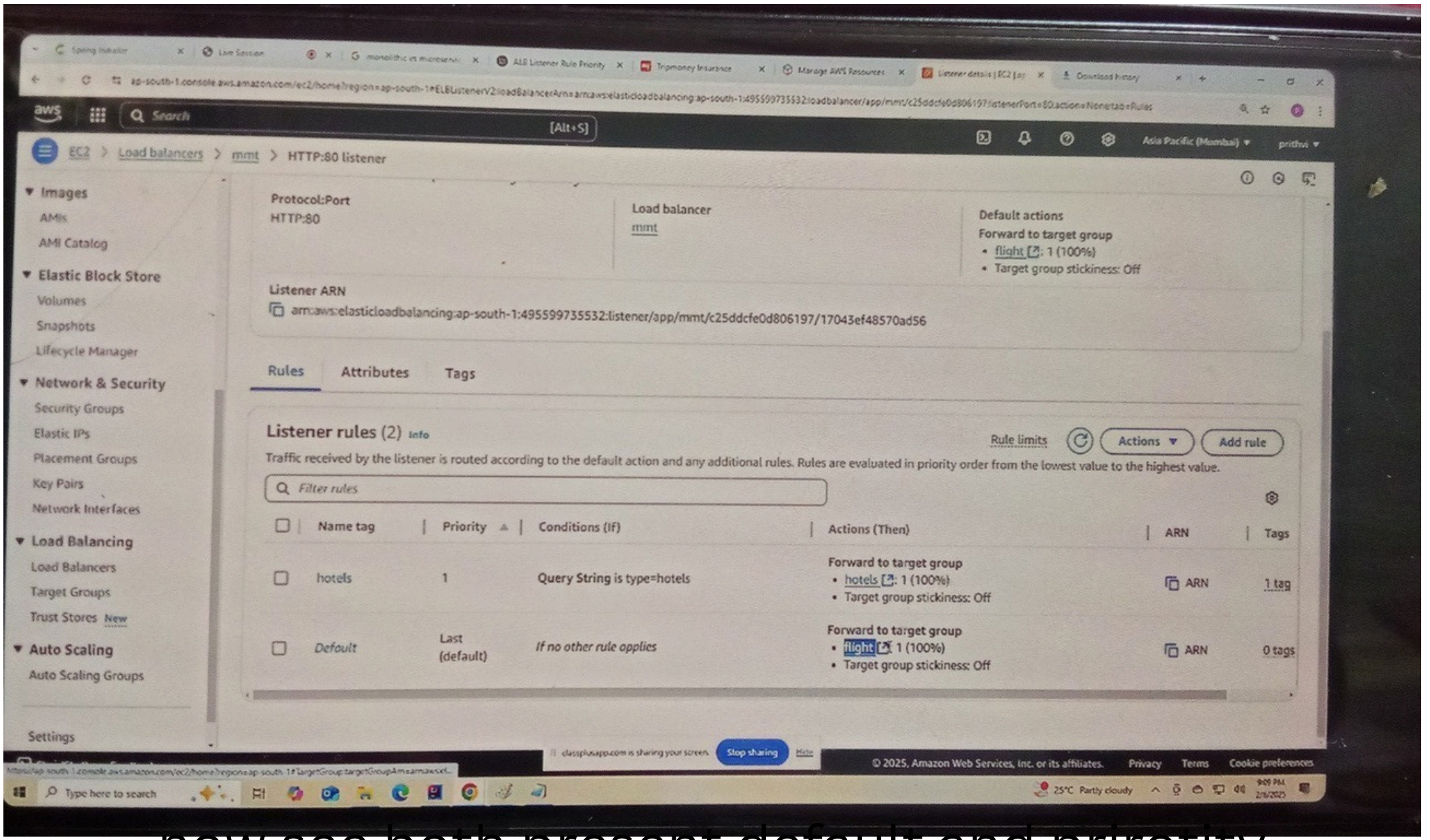
click next



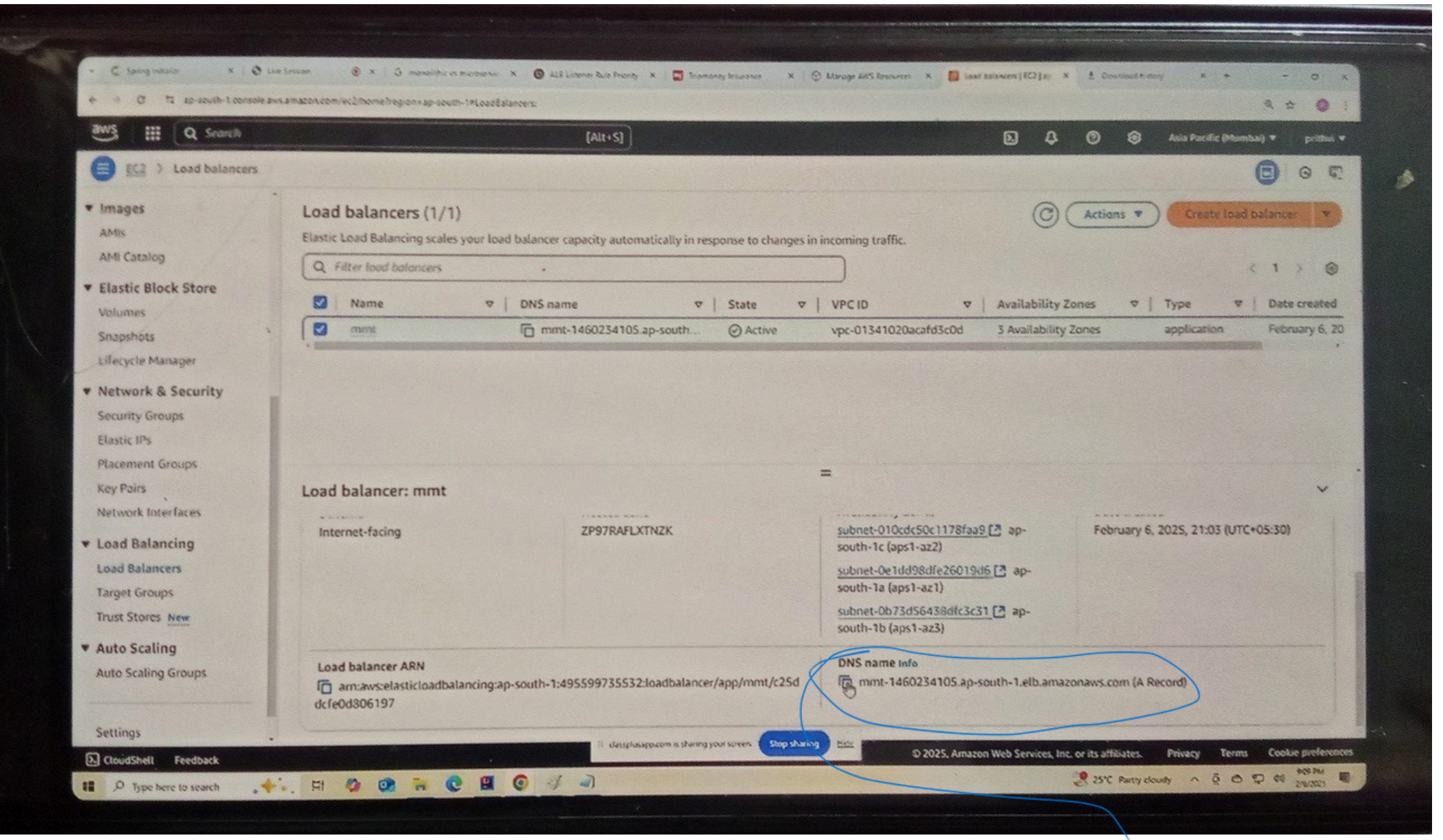
set priority and click next



Review and submit

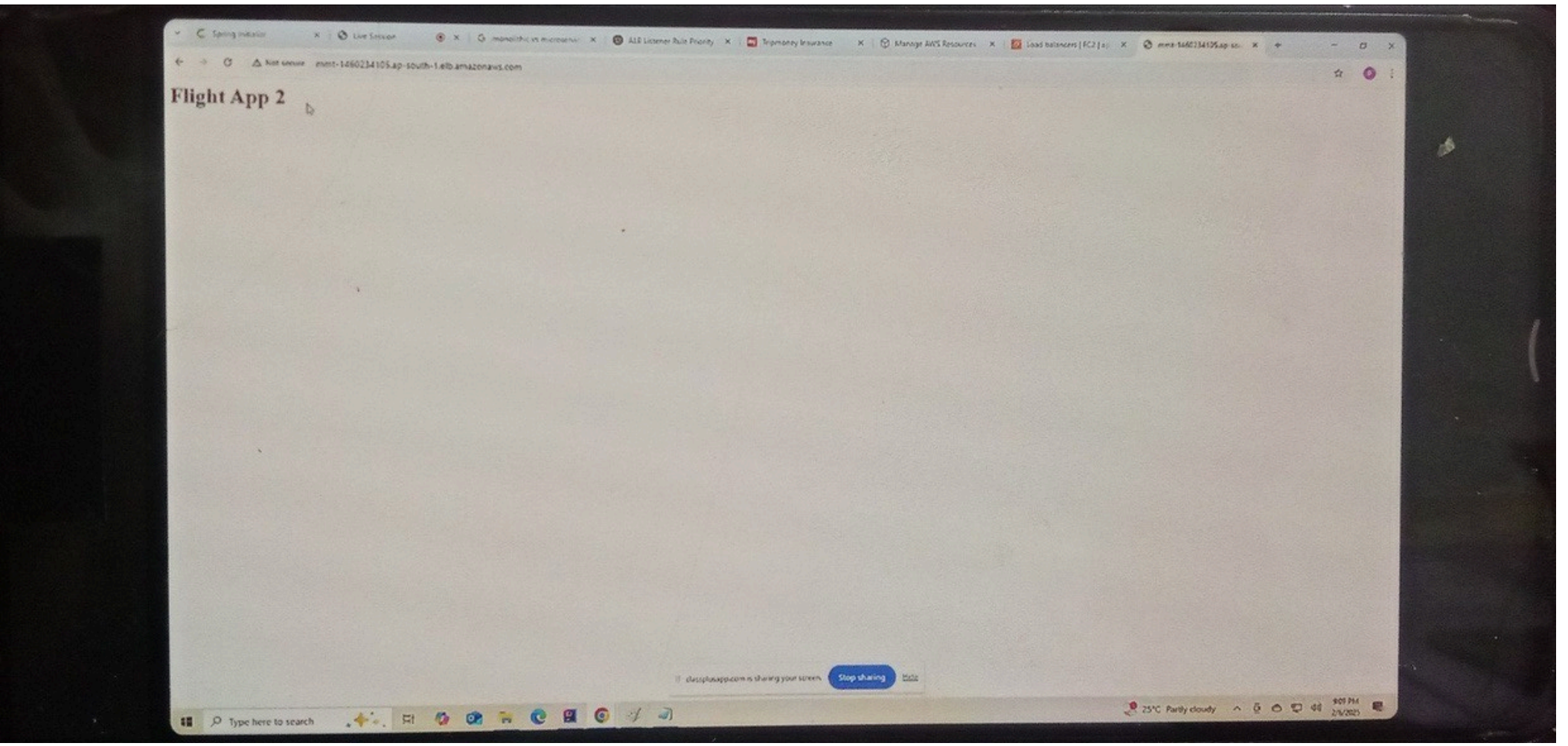


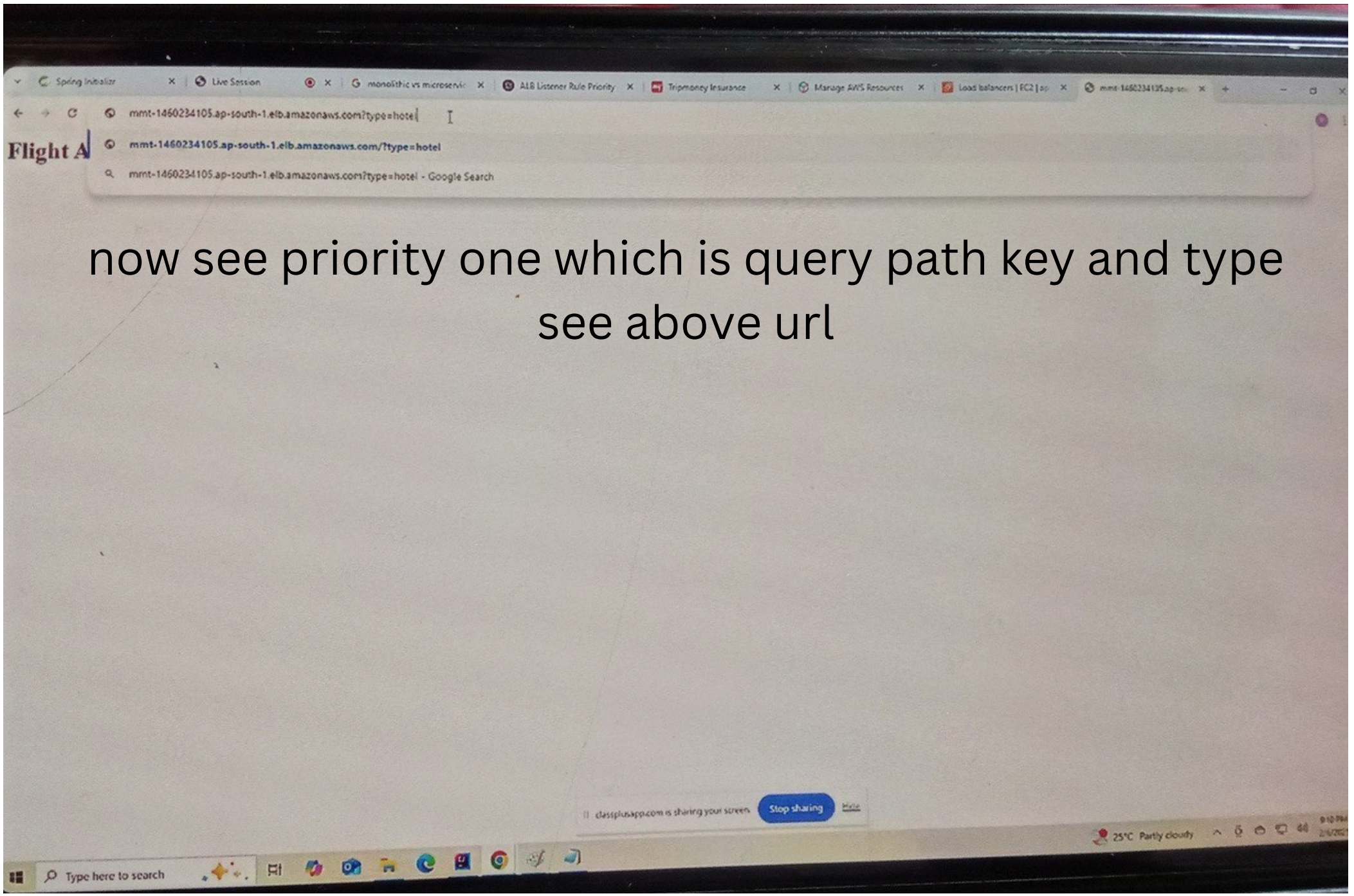
now see both present default and prirotity

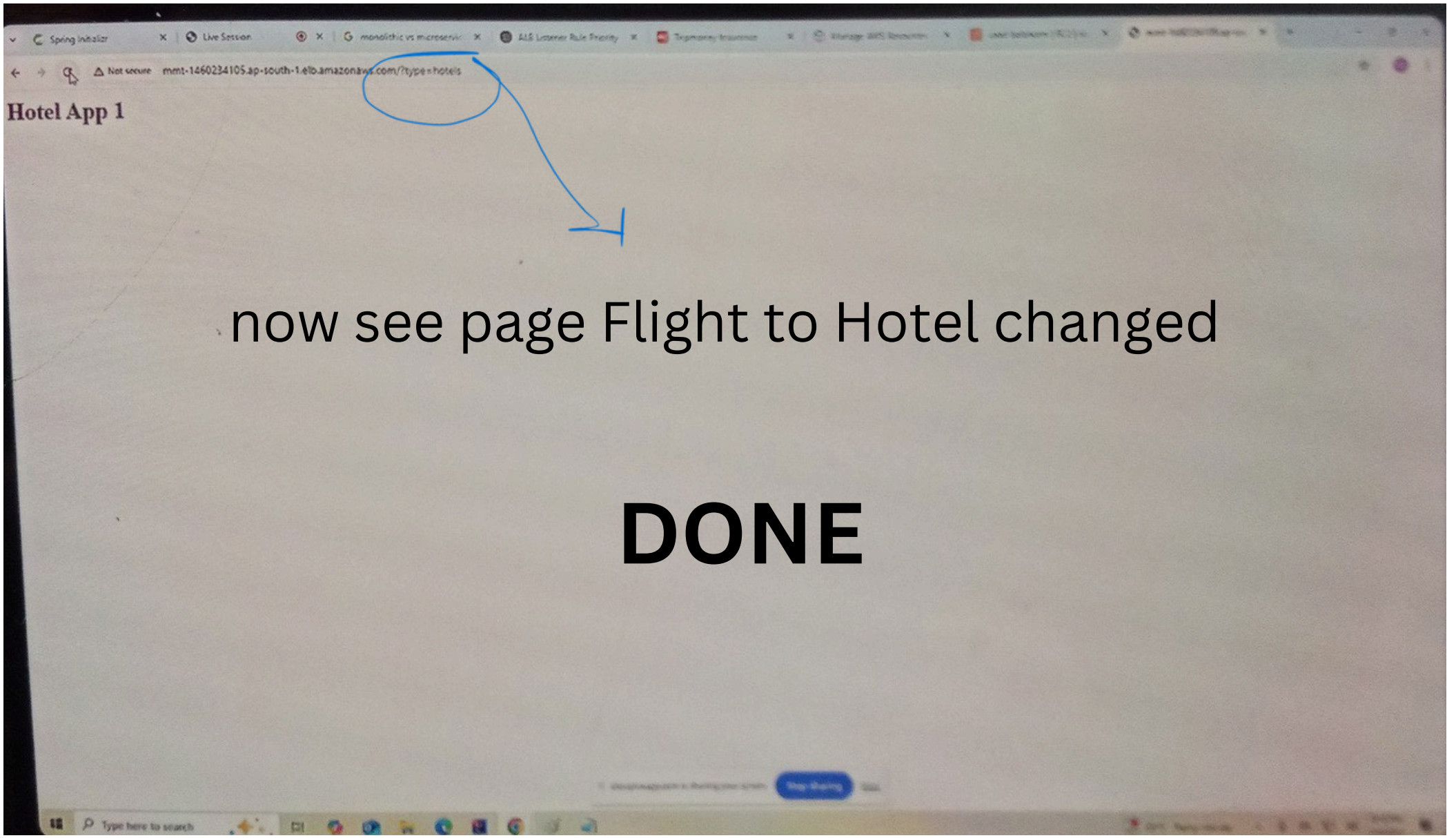


now go to load balancer and select DNS

default is flight







Create RDS and Connect MySQL

Introduction:

Amazon Web Services (AWS) provides a powerful and scalable cloud-based solution for managing databases through its Relational Database Service (RDS). In this tutorial, we will walk through the process of setting up an AWS RDS MySQL database, configuring its access, and connecting to it using MySQL Workbench. Whether you're new to AWS or looking to expand your knowledge, this guide will help you get started with RDS databases quickly and efficiently.

Prerequisites:

Before diving into the setup process, make sure you have the following prerequisites in place:

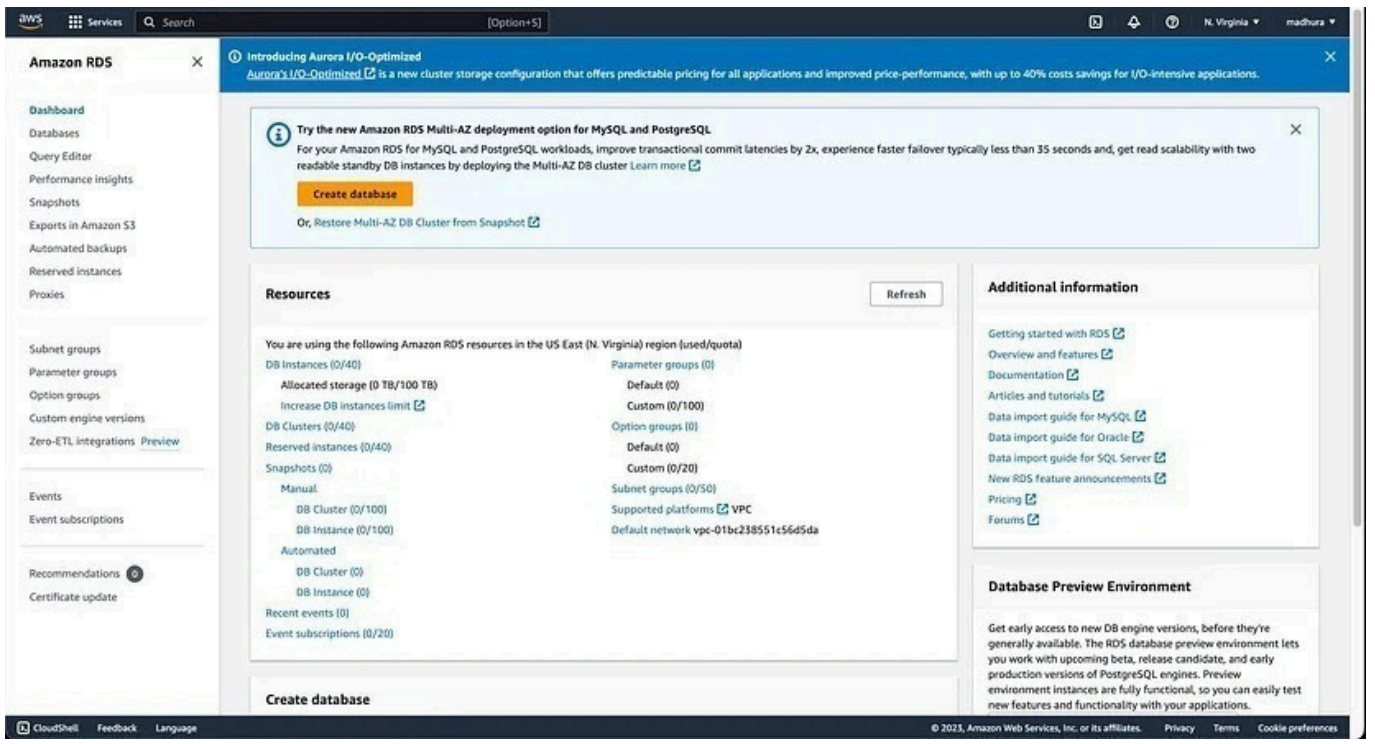
- An active AWS account with appropriate access permissions.
- Basic familiarity with AWS services and the AWS Management Console.
- A computer with an internet connection.

Creating an AWS RDS MySQL Database:

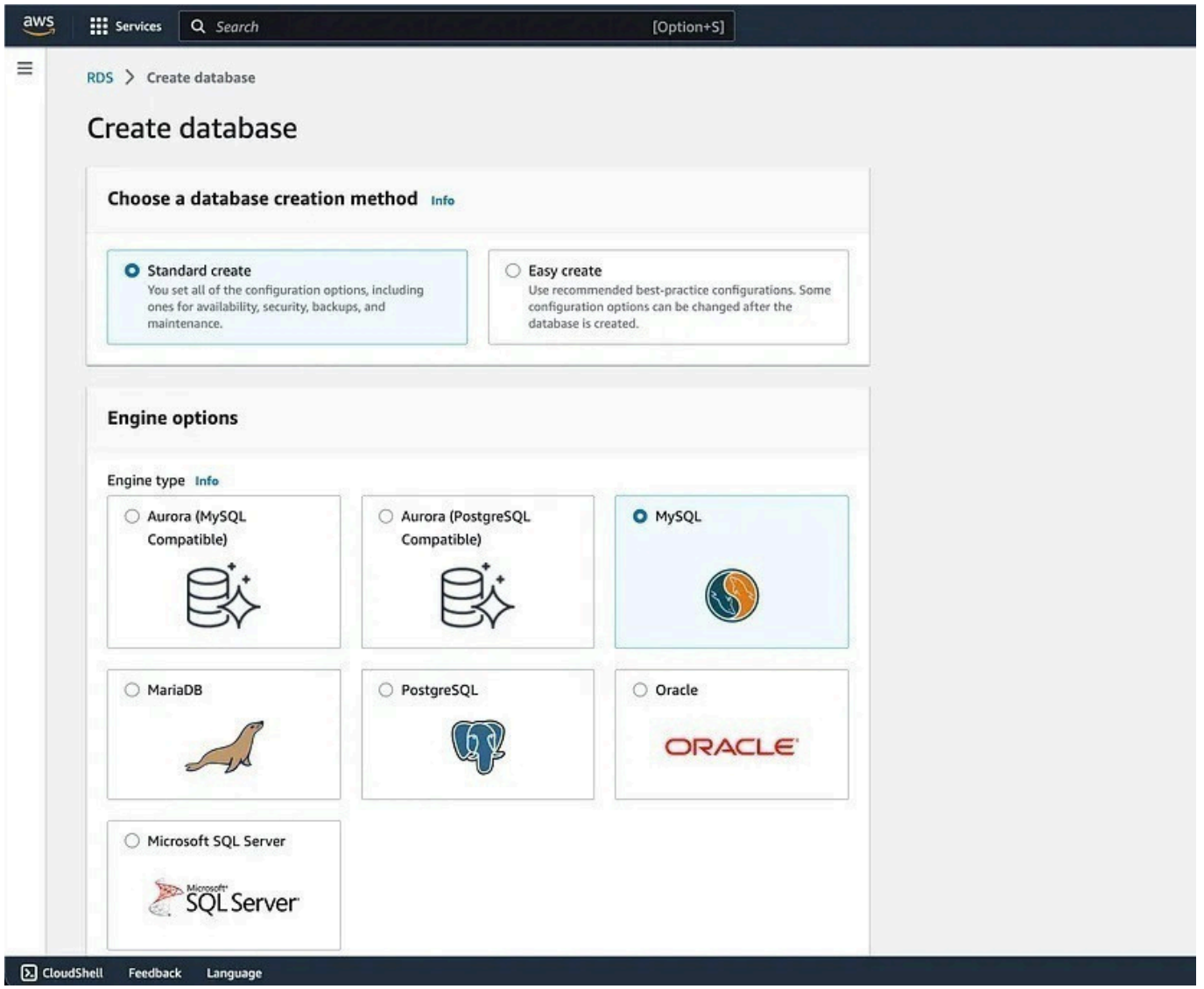
Step 1: Accessing AWS Console and Navigating to RDS

To begin, log in to your AWS Management Console and search for “RDS” in the service search bar. Click on the “RDS” service to open the RDS Dashboard.

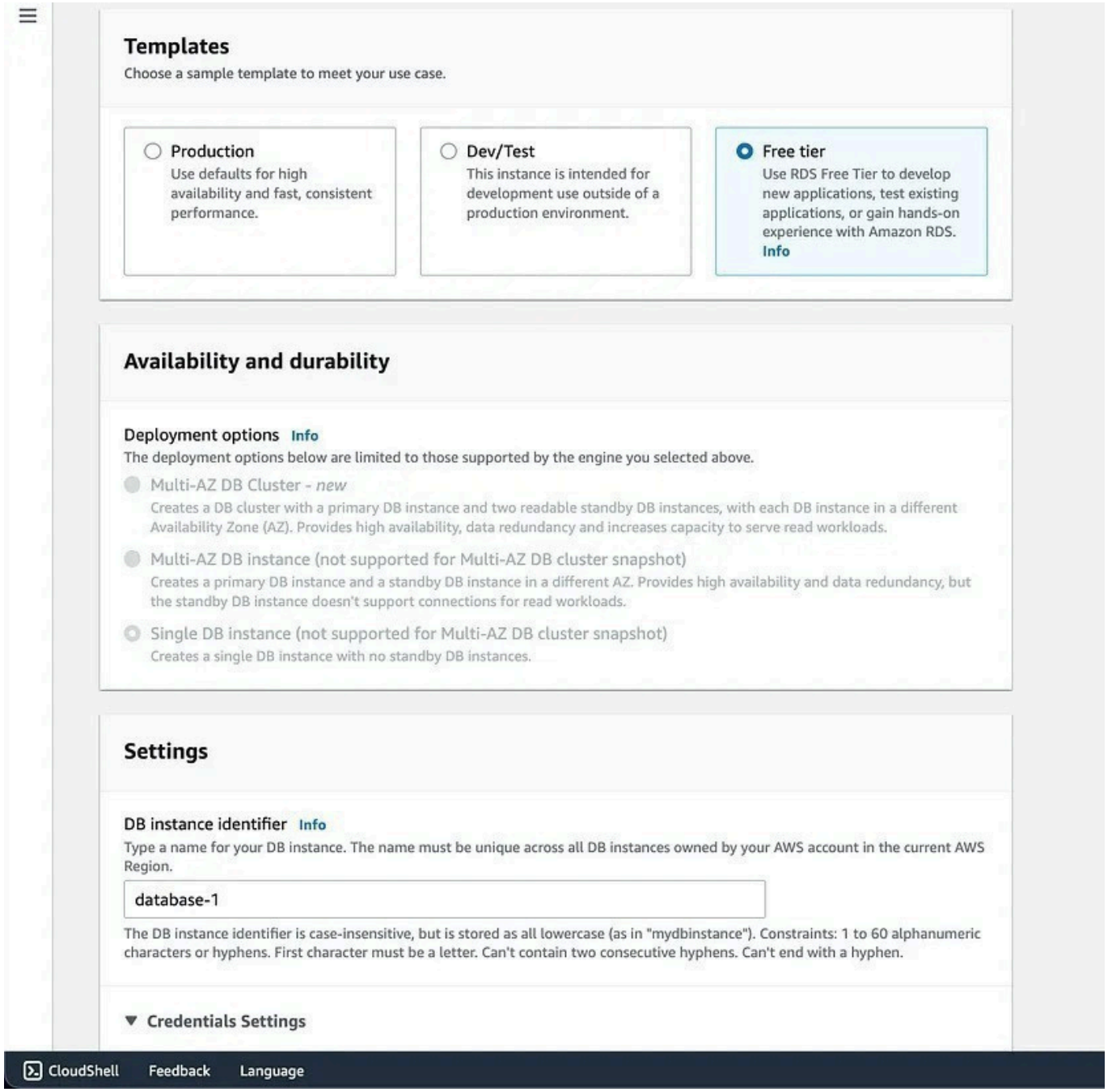
Step 2: Creating a MySQL Database



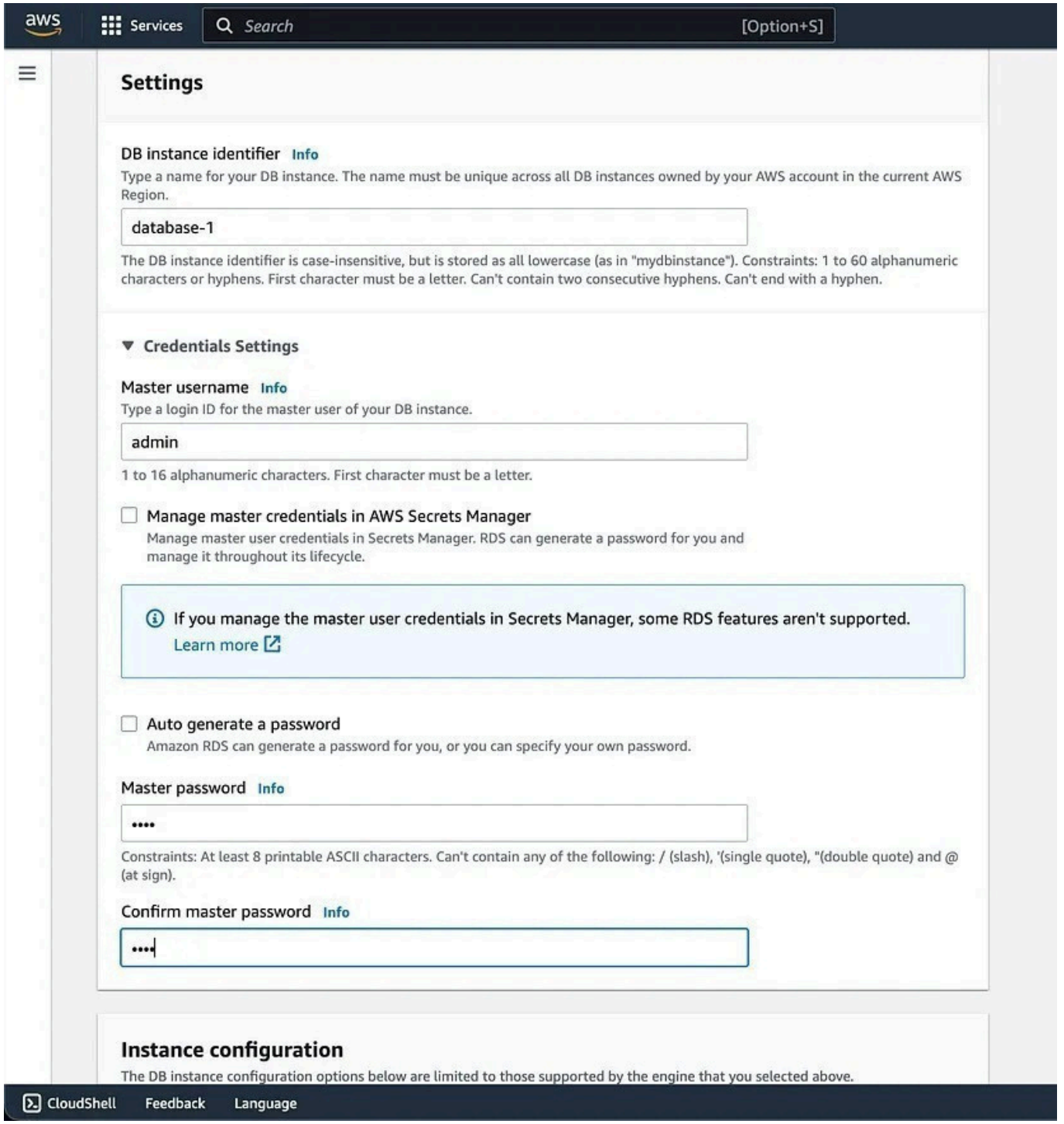
Within the RDS Dashboard, click on the “Create database” button and select “MySQL” as the database engine.



Provide a name for your database and choose the “Free tier” option.

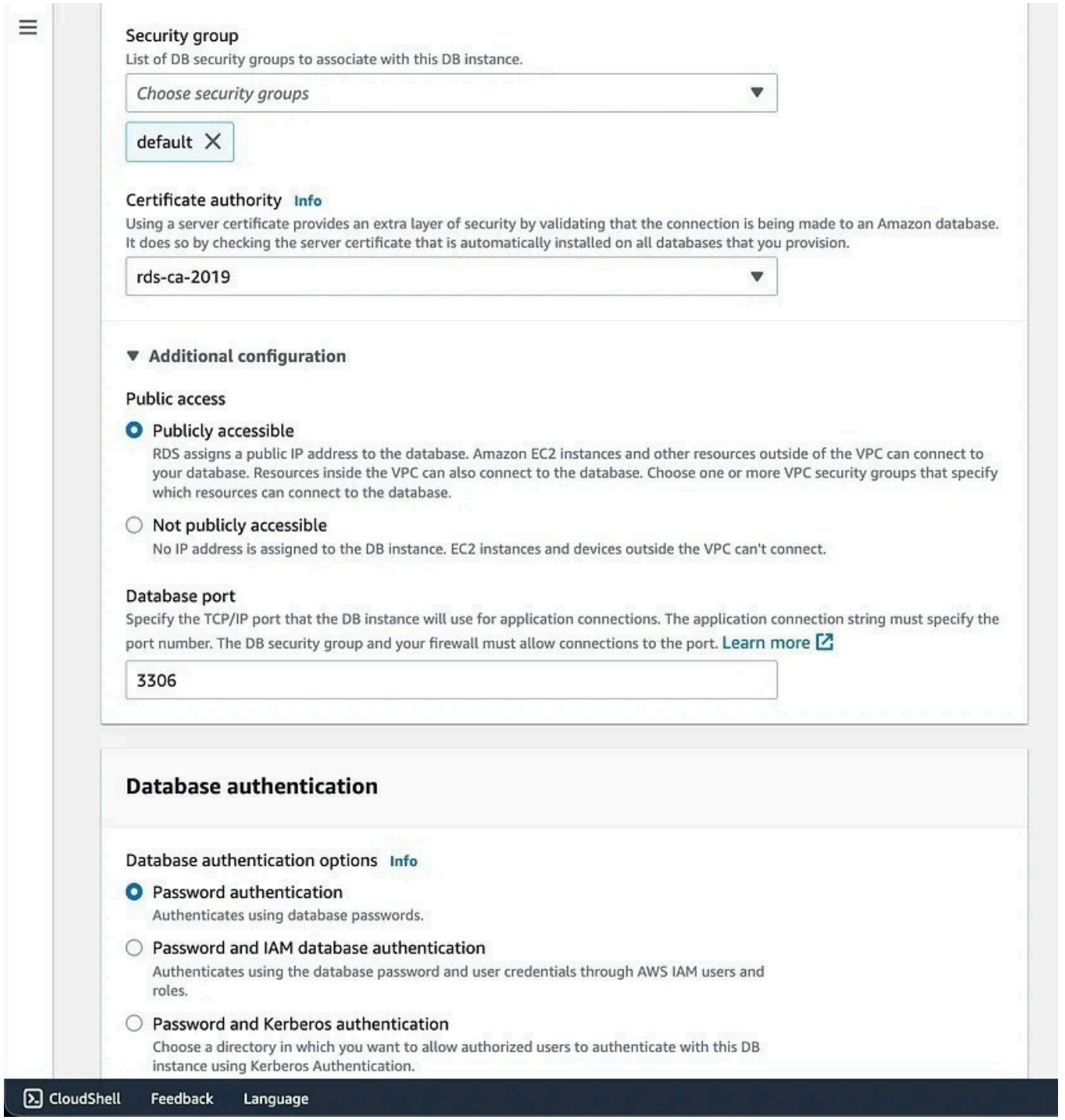


Set a password for accessing the database and proceed to create the database. Note that the creation process might take a few minutes.



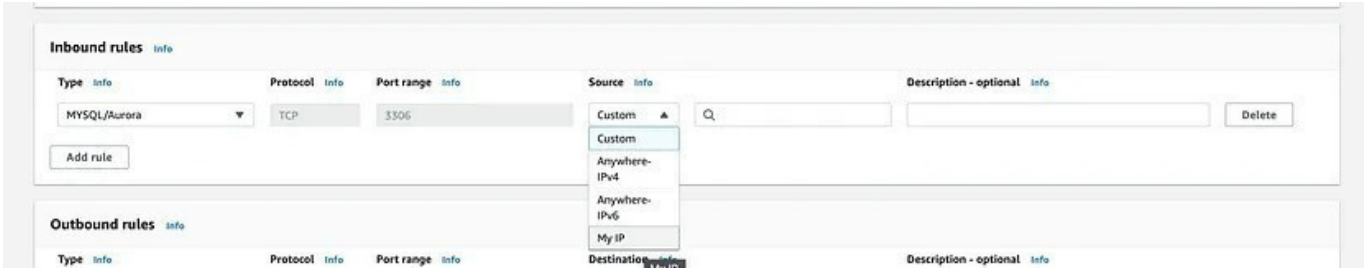
Step 3: Modifying Database Configuration

Once the database is created, go to the “Modify” option for your database. Expand the “Additional configuration” section and select the “Publicly accessible” option. Save the changes.



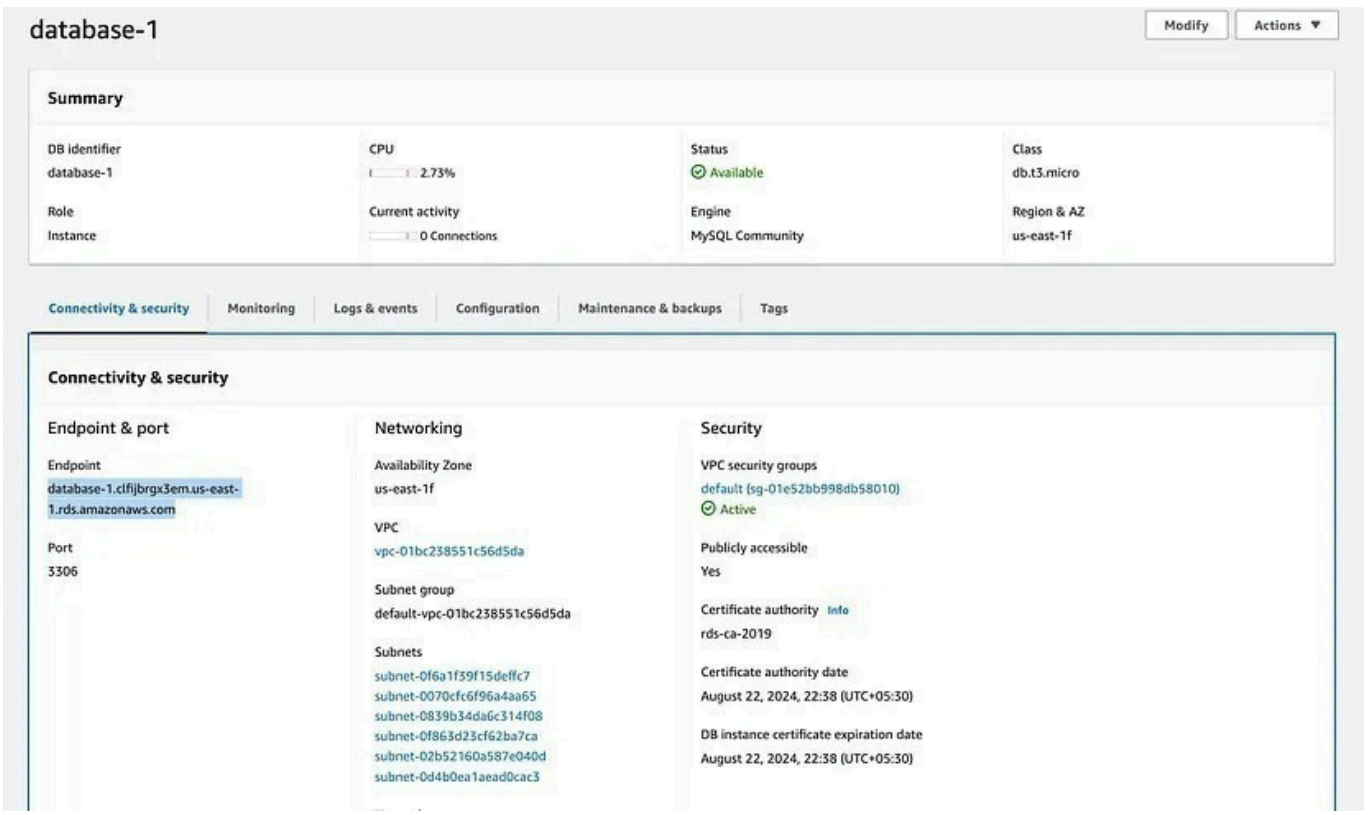
Step 4: Configuring Security Group Rules

Next, navigate to the “Security Group Rules” section for your database. In the “Inbound rules” tab, add a rule for MySQL. Specify the source as your IP address or any desired location. Save the changes.



Step 5: Obtaining the Database Endpoint

After saving the security group rules, allow some time for the changes to propagate. Once completed, you should see the “Publicly accessible” option set to “true” for your database. Copy the endpoint provided for your database. This endpoint will be used to connect to the database.



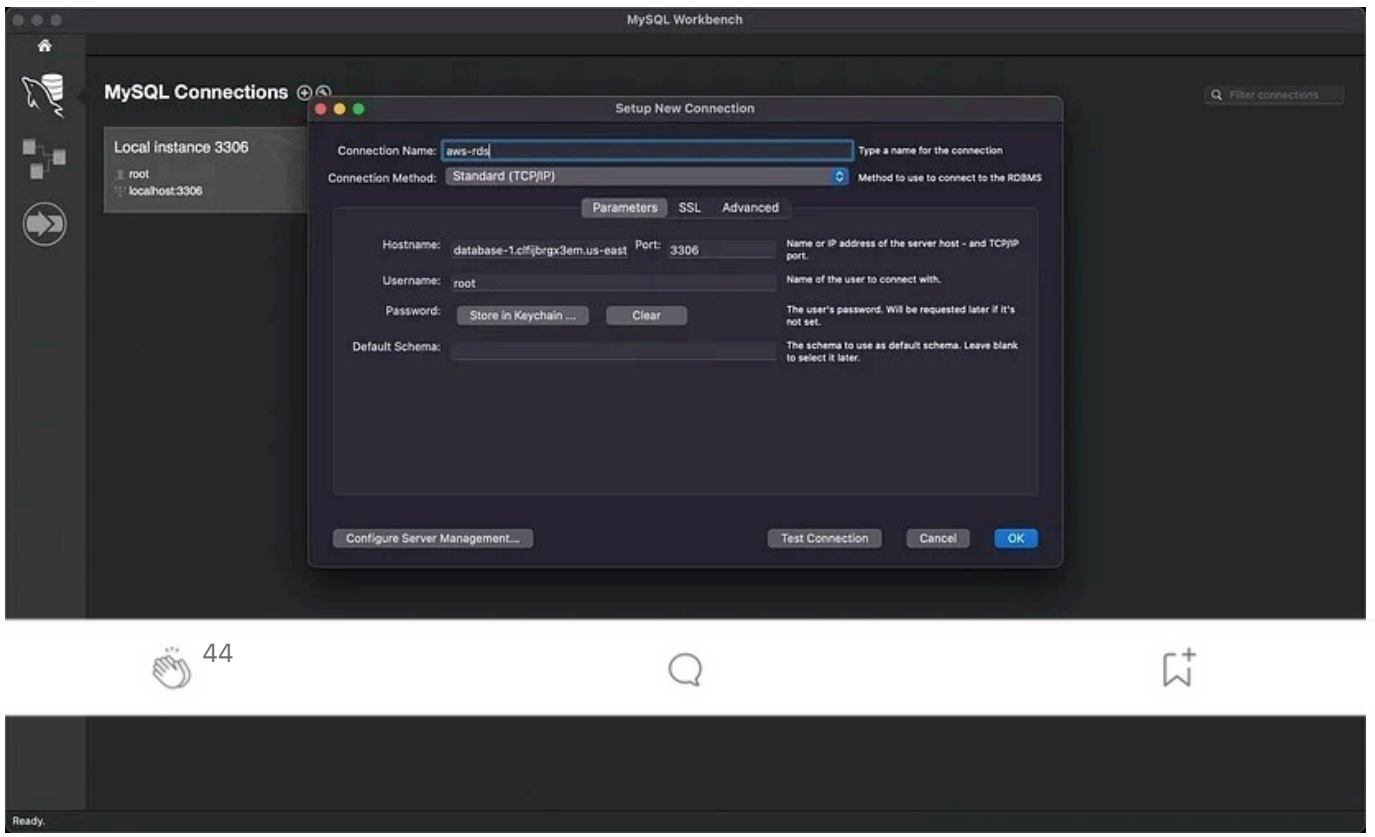
Connecting to the AWS RDS MySQL Database:

Step 1: Downloading MySQL Workbench

To connect to the RDS database, download and install MySQL Workbench on your local machine. MySQL Workbench is a popular tool for managing and interacting with MySQL databases.

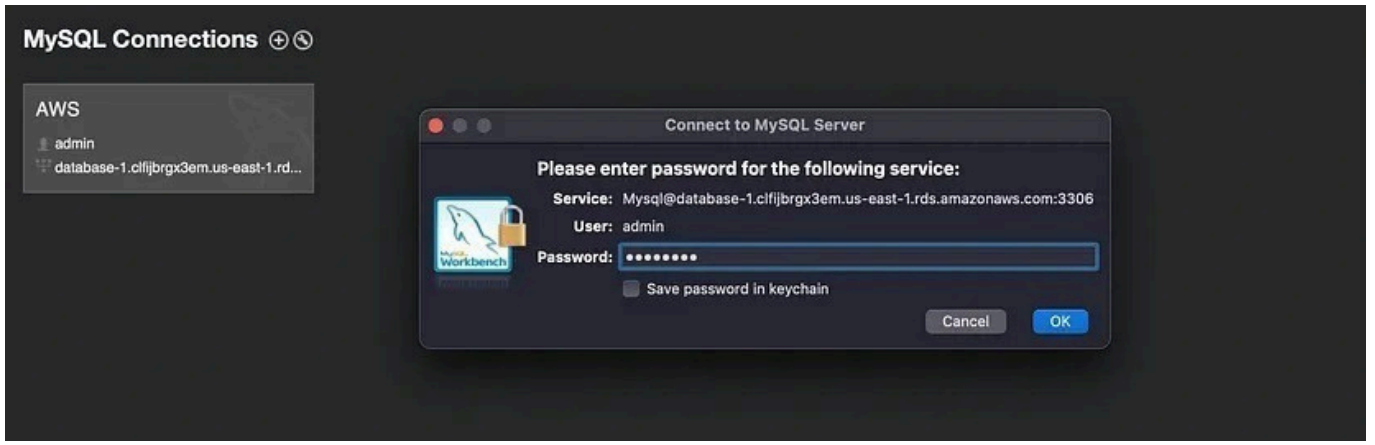
Step 2: Creating a New Connection

Launch MySQL Workbench and click on the “+” button next to “MySQL Connections” to create a new connection. Provide a name for the connection and enter the database endpoint copied earlier. Set the port to 3306 and leave the other settings as default.

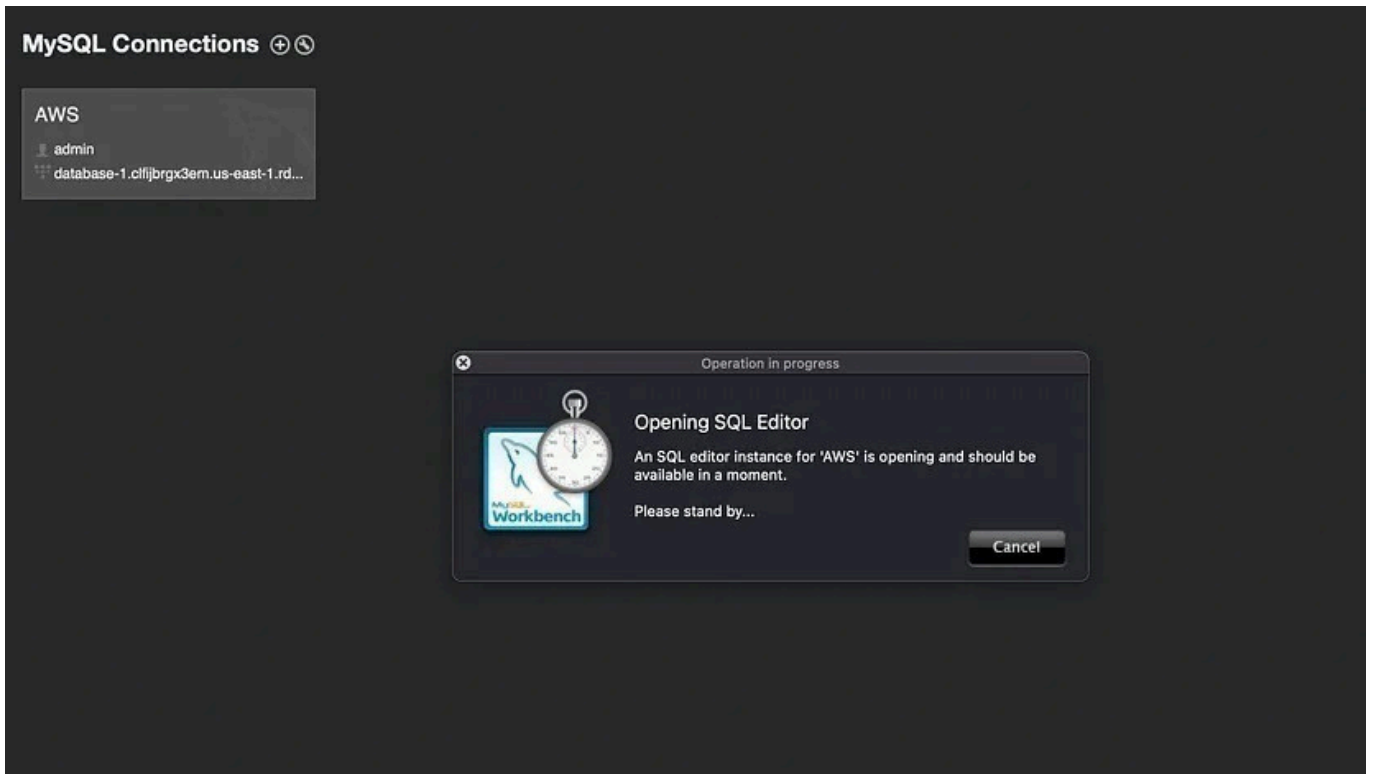


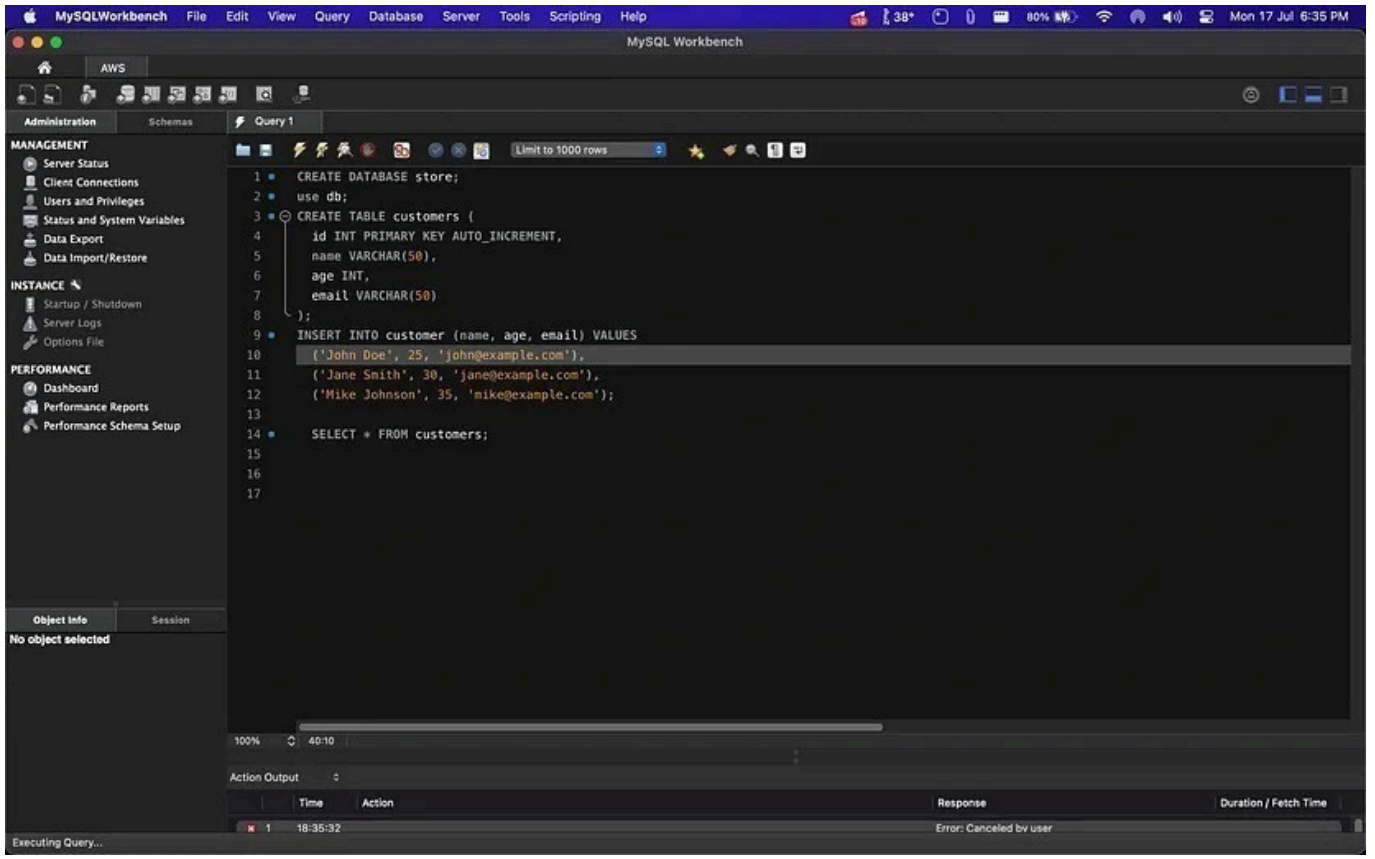
Step 3: Logging Into the Database

Click on the newly created connection and enter the password you set during the database creation process. Click “Test Connection” to verify the connection settings. If successful, click “OK” to establish the connection.



Now you can use MySQL like it’s on your local machine!




MySQLWorkbench File Edit View Query Database Server Tools Scripting Help
Mon 17 Jul 6:35 PM
MySQL Workbench
AWS
Administration
Schemas
Query 1
MANAGEMENT
Server Status
Client Connections
Users and Privileges
Status and System Variables
Data Export
Data Import/Restore
INSTANCE
Startup / Shutdown
Server Logs
Options File
PERFORMANCE
Dashboard
Performance Reports
Performance Schema Setup
Limit to 1000 rows
1 CREATE DATABASE store;
2 use db;
3 CREATE TABLE customers (
4 id INT PRIMARY KEY AUTO_INCREMENT,
5 name VARCHAR(50),
6 age INT,
7 email VARCHAR(50)
8);
9 INSERT INTO customer (name, age, email) VALUES
10 ('John Doe', 25, 'john@example.com'),
11 ('Jane Smith', 30, 'jane@example.com'),
12 ('Mike Johnson', 35, 'mike@example.com');
13
14 SELECT * FROM customers;
15
16
17
Object Info
Session
No object selected
100%
40:10
Action Output
Time
Action
Response
Duration / Fetch Time
1 18:35:32
Error: Canceled by user
Executing Query...

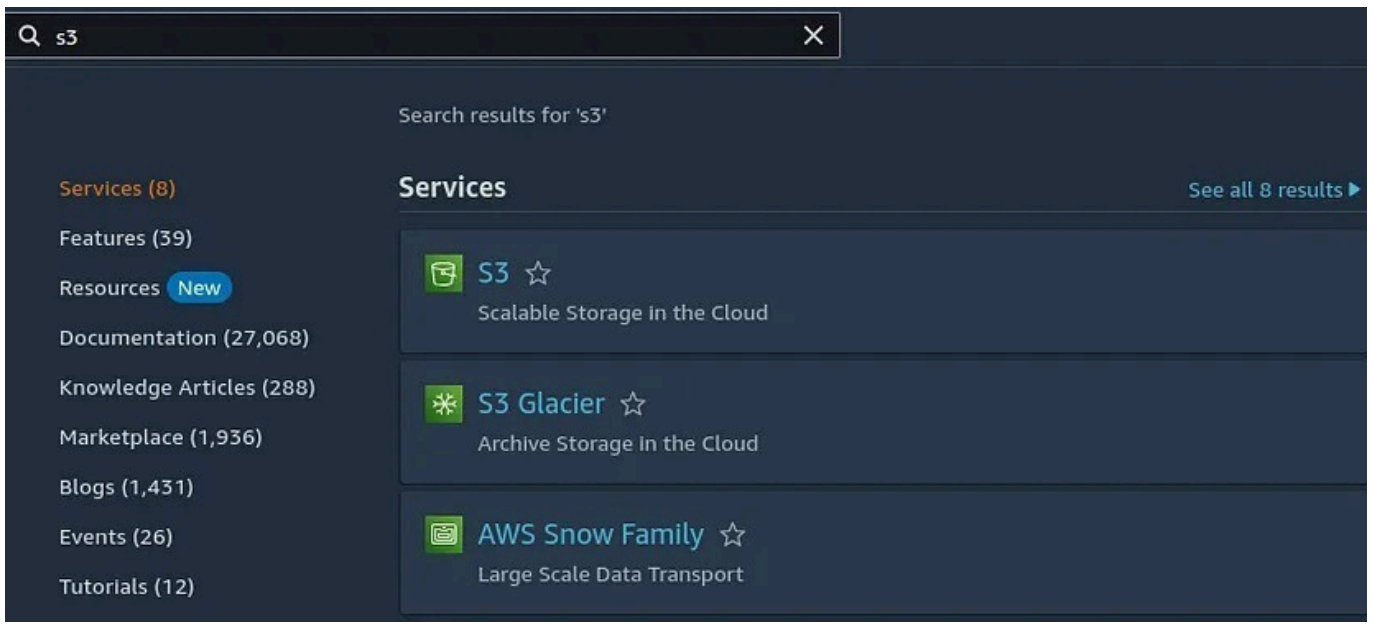
MySQLWorkbench File Edit View Query Database Server Tools Scripting Help
Mon 17 Jul 6:39 PM
MySQL Workbench
AWS
Administration
Schemas
Query 1
SCHEMAS
Filter objects
sys
Tables
Views
Stored Procedures
Functions
Limit to 1000 rows
1 CREATE DATABASE IF NOT EXISTS store;
2 use store;
3 CREATE TABLE customers (
4 id INT PRIMARY KEY AUTO_INCREMENT,
5 name VARCHAR(50),
6 age INT,
7 email VARCHAR(50)
8);
9 INSERT INTO customers (name, age, email) VALUES
10 ('John Doe', 25, 'john@example.com'),
11 ('Jane Smith', 30, 'jane@example.com'),
12 ('Mike Johnson', 35, 'mike@example.com');
13
14 SELECT * FROM customers;
15
100%
27:14
Result Grid
Filter Rows:
Search
Edit:
Export/Import:
id
name
age
email
1
John Doe
25
john@example.com
2
Jane Smith
30
jane@example.com
3
Mike Johnson
35
mike@example.com
Result Grid
Form Editor
Object Info
Session
No object selected
customers 1
Apply
Revert
Action Output
Time
Action
Response
Duration / Fetch Time
1 18:39:00 CREATE DATABASE IF NOT EXISTS store 1 row(s) affected 0.315 sec
2 18:39:00 use store 0 row(s) affected 0.313 sec
3 18:39:01 CREATE TABLE customers (id INT PRIMARY KEY AUTO_INCREMENT, name VARCHAR(50), age INT, email VAR... 0 row(s) affected 0.363 sec
4 18:39:01 INSERT INTO customers (name, age, email) VALUES ('John Doe', 25, 'john@example.com'), ('Jane Smith', 30, 'jan... 3 row(s) affected Records: 3 Duplicates: 0 Warnings... 0.329 sec
5 18:39:01 SELECT * FROM customers LIMIT 0, 1000 3 row(s) returned 0.337 sec / 0.000062...
Query Completed

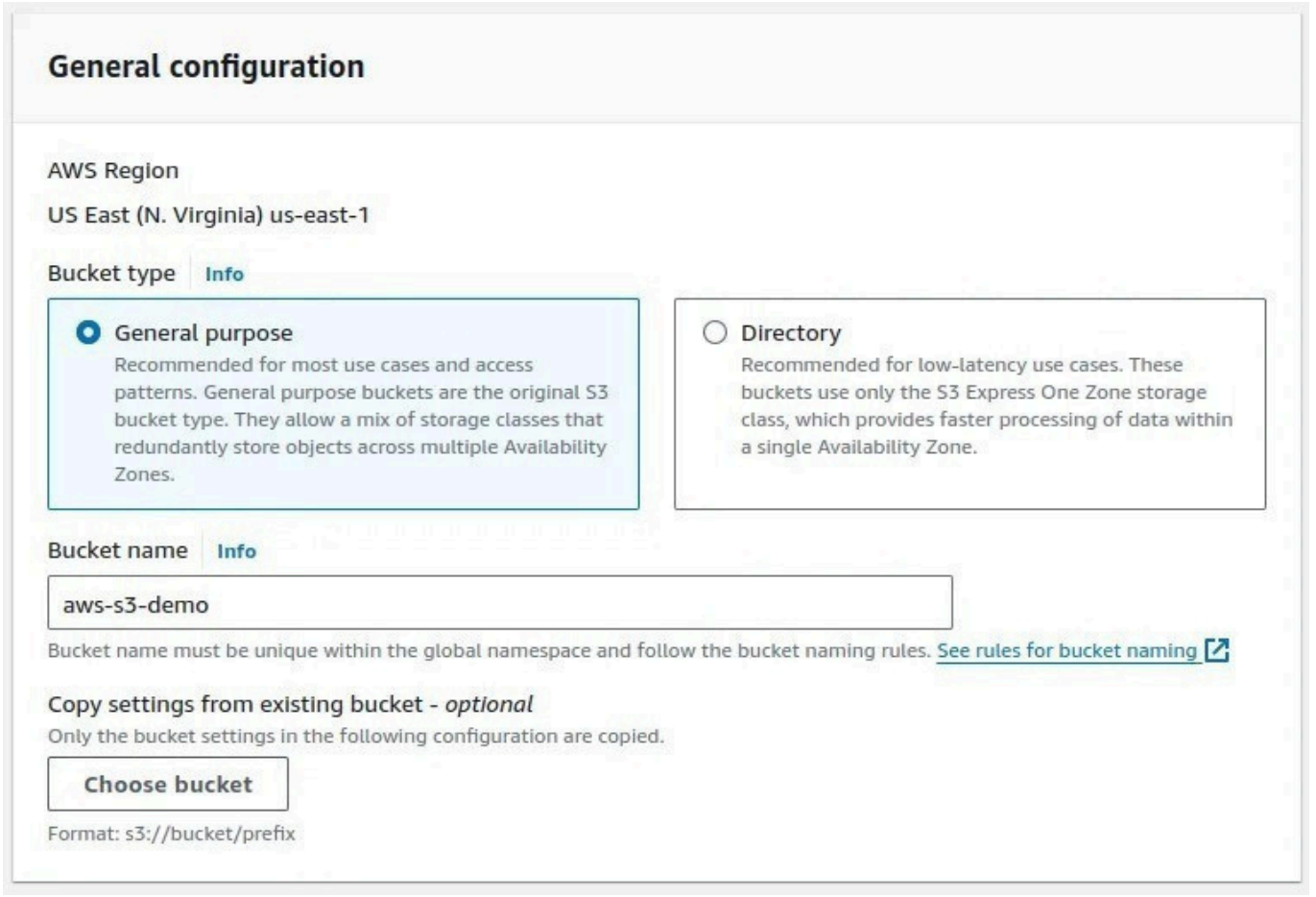
Conclusion:

Congratulations! You have successfully set up an AWS RDS MySQL database and connected to it using MySQL Workbench. AWS RDS provides a robust and scalable solution for managing databases in the cloud, and with the steps outlined in this tutorial, you can start building and deploying your applications with ease.

Create bucket in S3 and host static website

Select AWS S3 service in the management console.





General Configurations for Amazon S3

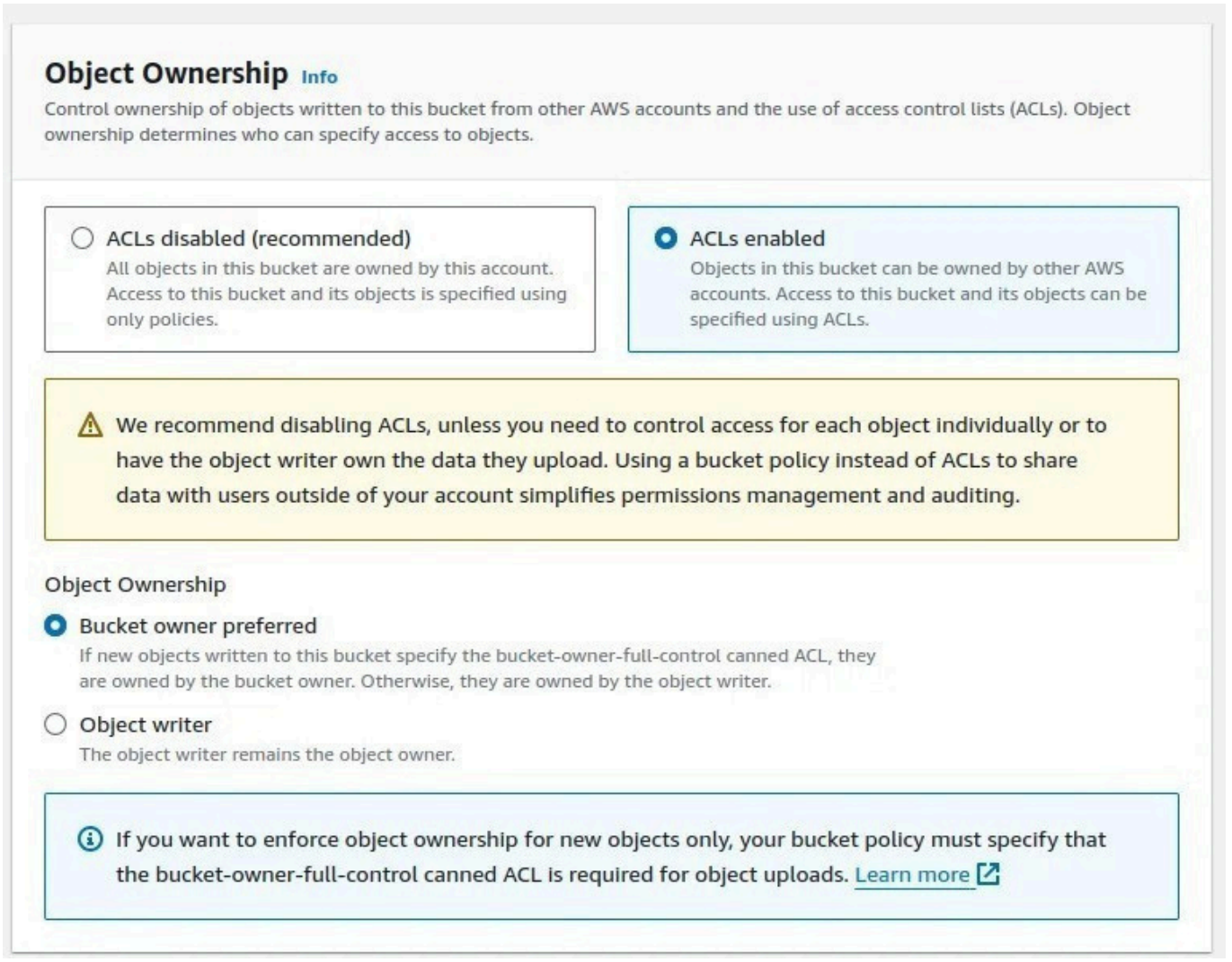
- After that, we have to define the ownership of bucket objects through the ACL options. Here, we have two options: we can either enable or disable ACLs. Let’s see what Access Control Lists are.

Access Control Lists

- An Access Control List (ACL) is a set of rules determining who can access a resource. In this S3 setup, enabling ACLs lets you control who can access and manage the objects (e.g., website files) you upload to your bucket.
- ACLs enable different AWS accounts to own and control specific files(objects) within your bucket.

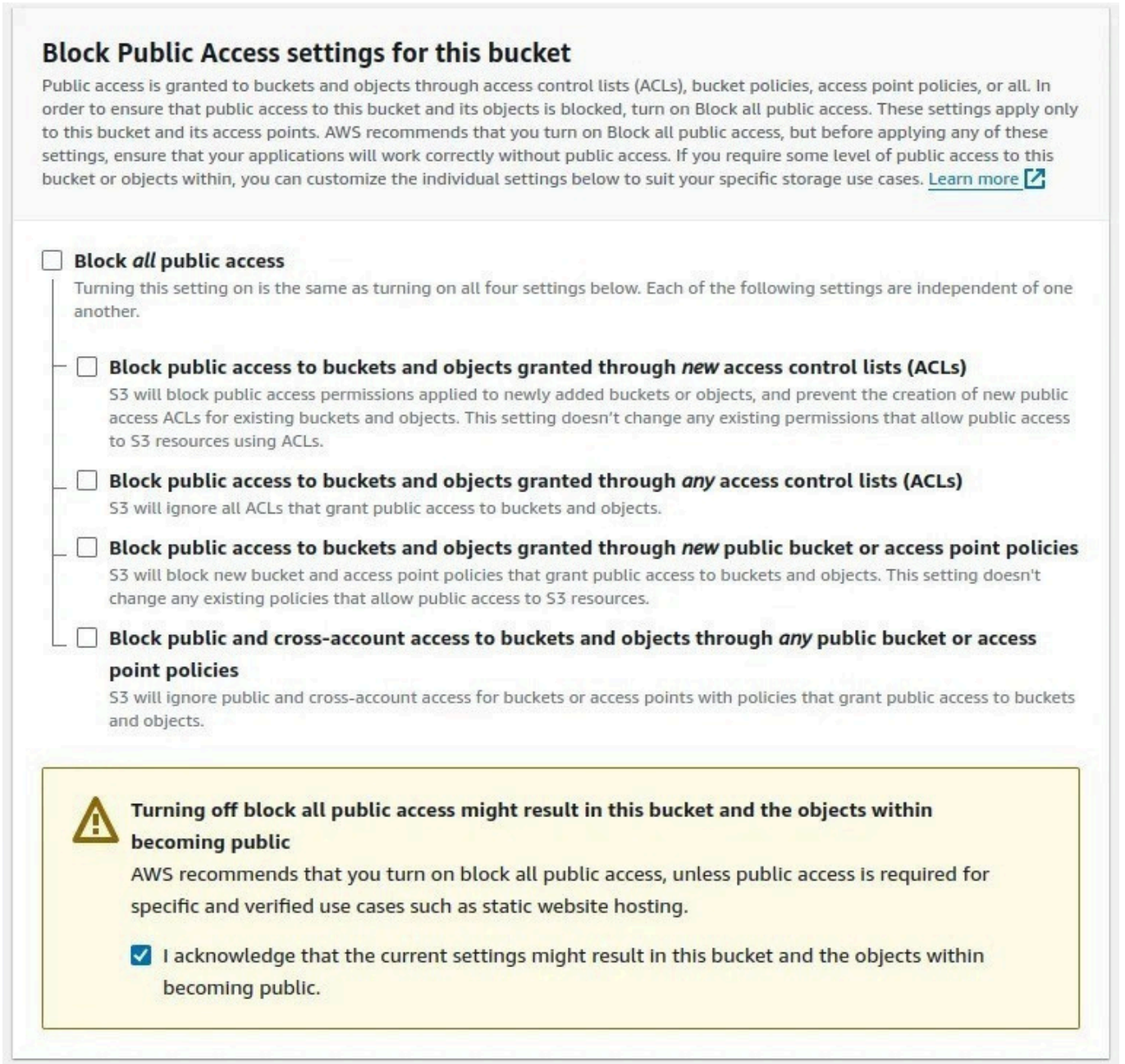
- However, a warning may suggest using “bucket policies” instead, as they simplify access control by applying permissions to the entire bucket (e.g., publishing the entire bucket and its objects). Unlike ACLs, bucket policies do not offer fine-grained control over individual objects.
- In summary, if your bucket contains no sensitive information, a bucket policy is sufficient. However, if even one object in the bucket needs to remain private, using ACLs is advisable.

So, I chose the “ACLs enabled” option and assigned the object ownership to the bucket owner.



Defining object ownership in S3 bucket

- Now, we need to decide whether this bucket will be public or not. Since this bucket will be used to deploy a static website, I made the bucket public by clearing all the checkboxes to enable access for everyone.

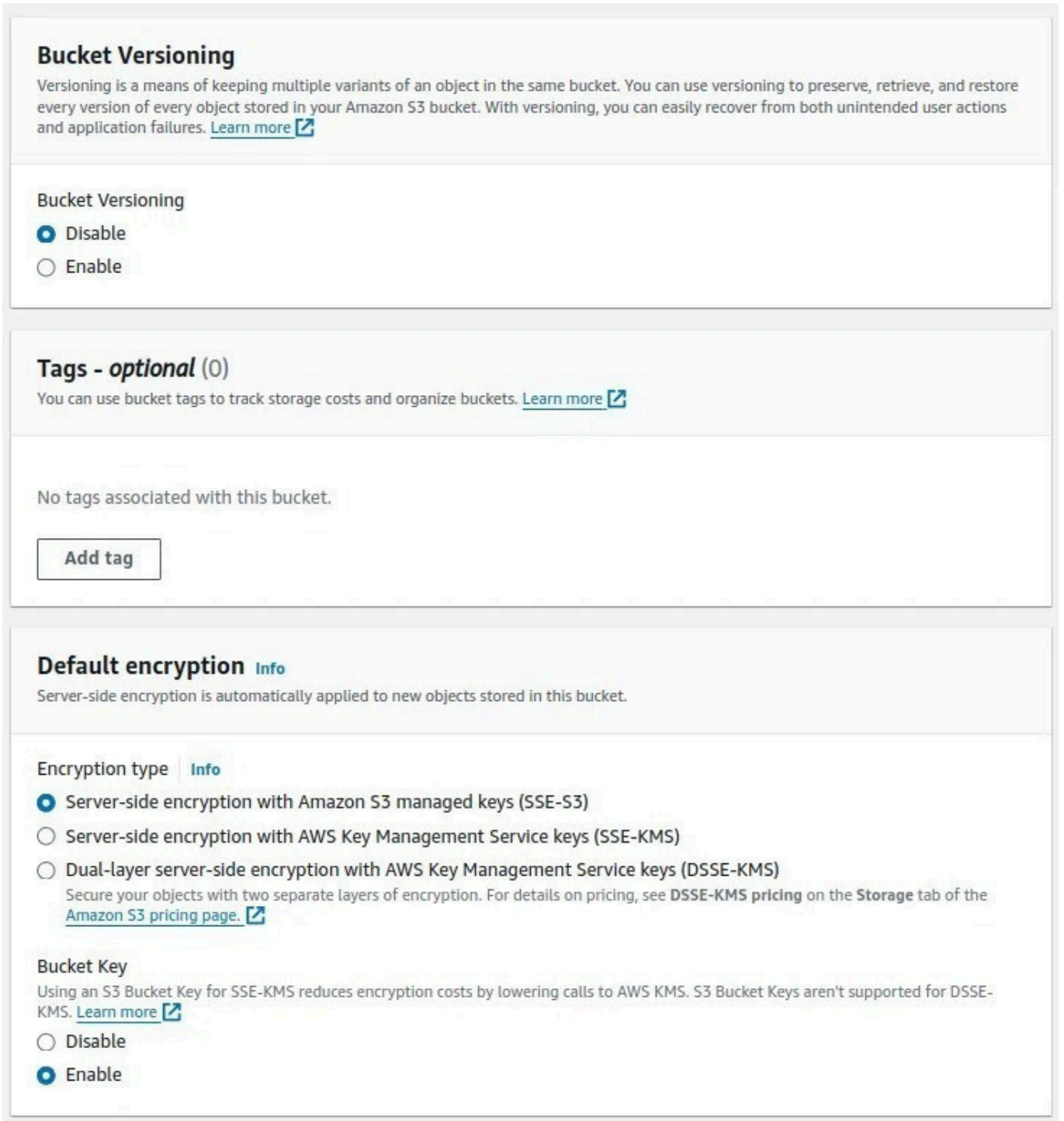


Defining public access setting for the S3 bucket

- Next, there is an option to manage bucket versions, which allows saving previous objects as a history when the bucket is updated with new ones. This feature helps track previous objects (backup copies), their URLs, and

last modified dates. It is helpful because it allows recovering a previous object if it is mistakenly deleted. Therefore, I chose to enable this option.

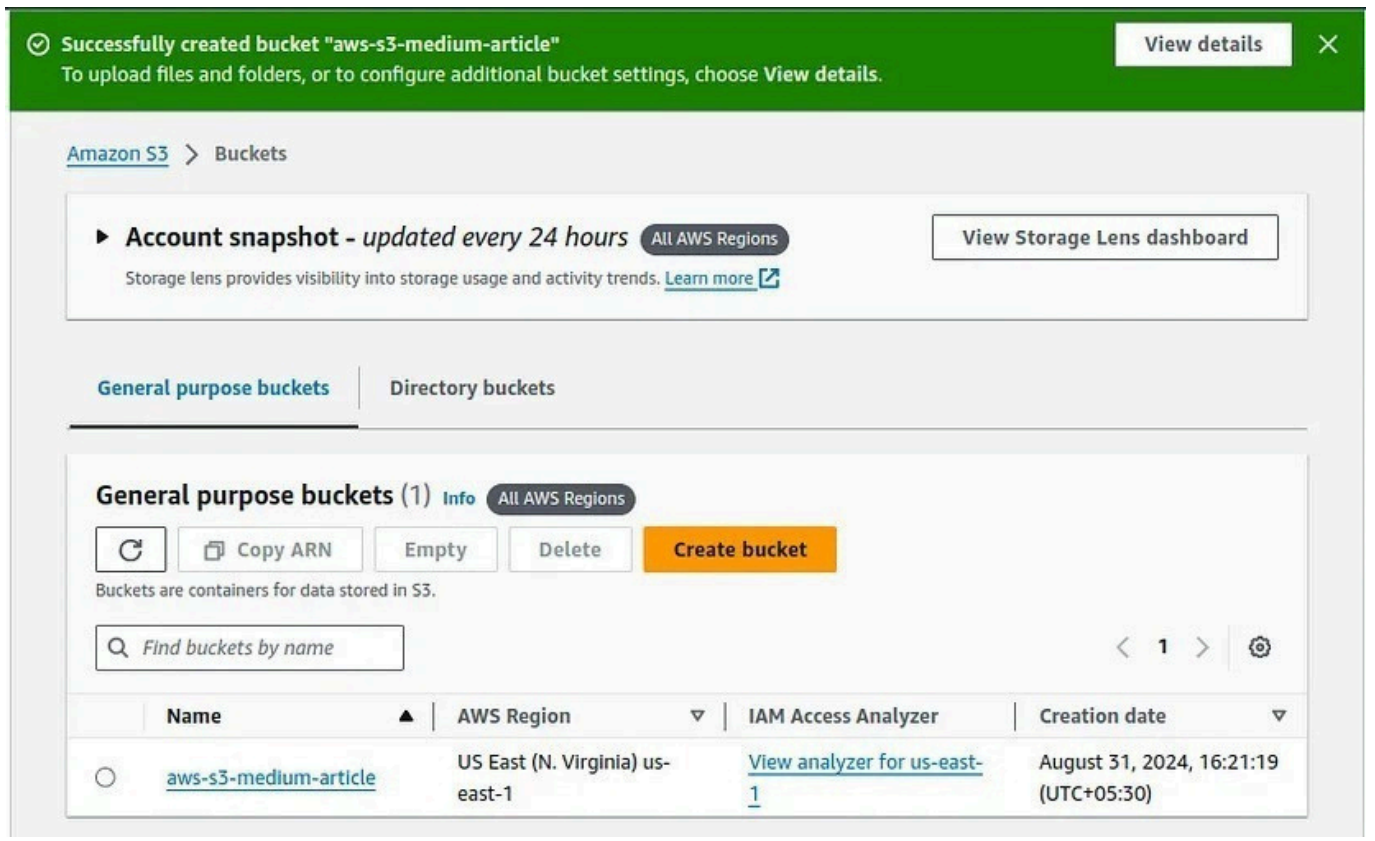
- After that, we can optionally add tags to manage cost allocations. I leave that field empty and keep the default settings for encryption.



Versioning and encryption in S3 bucket

- Now, choose the Create Bucket option.

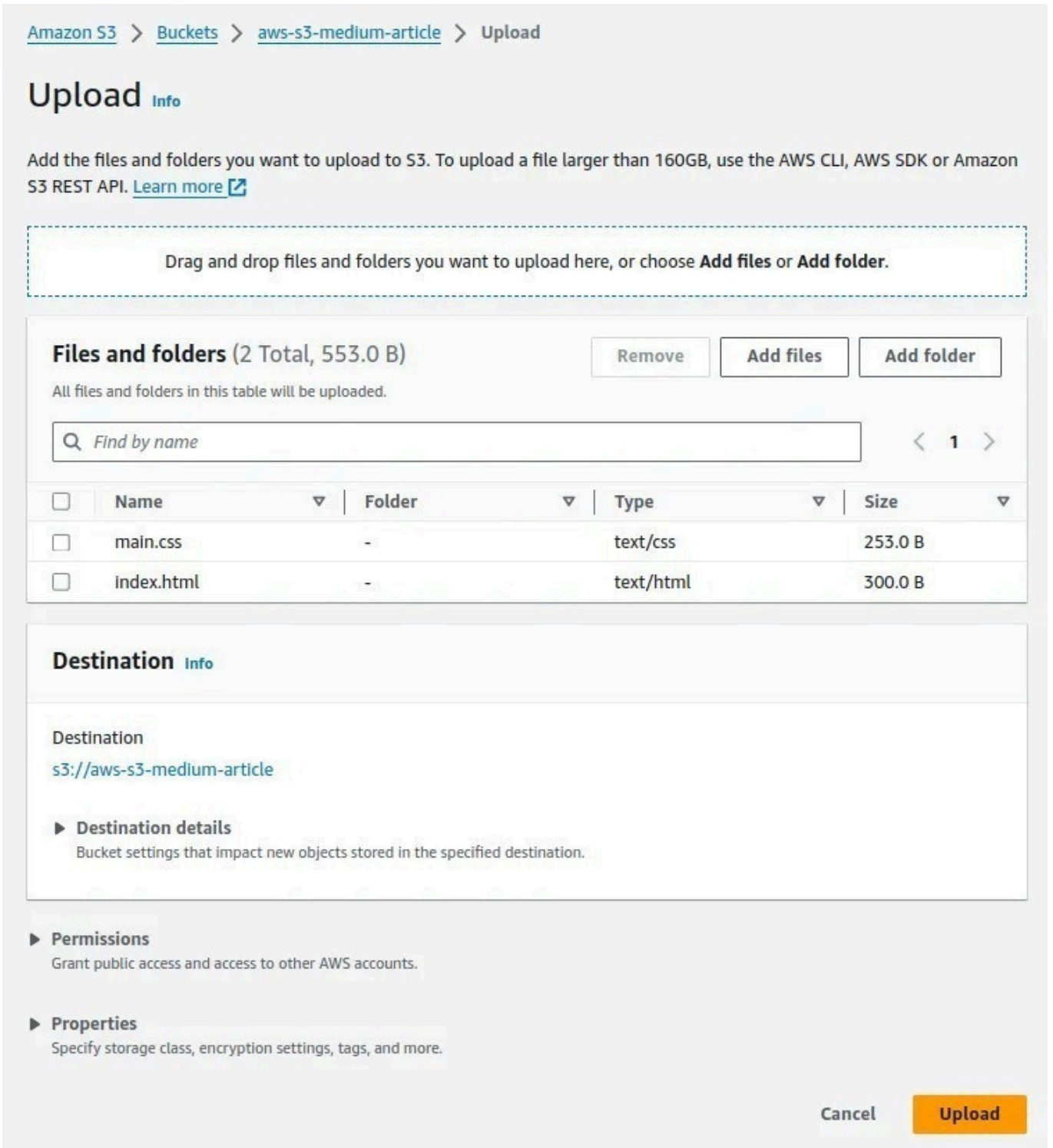
Excellent! You just created an S3 bucket. Now, let’s upload your static website to it.



Amazon S3 bucket creation

🎯 Step 2: Upload Contents to the Amazon S3

1. Now select the created bucket in the Buckets section.
2. Then, choose and upload the files to the bucket.

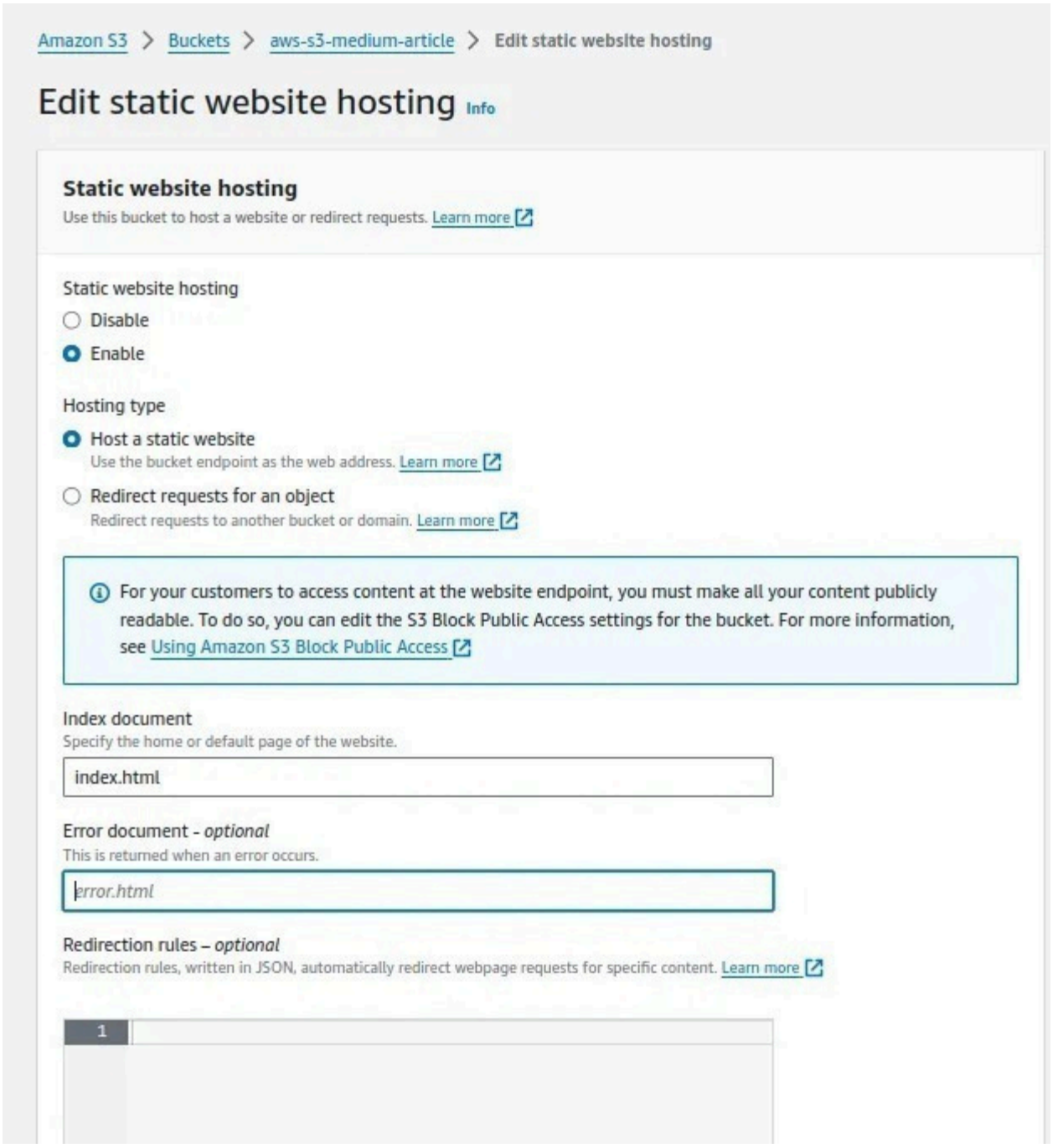


Upload files to the S3 bucket.

🎯 Step 3: Configure a static web hosting on Amazon S3

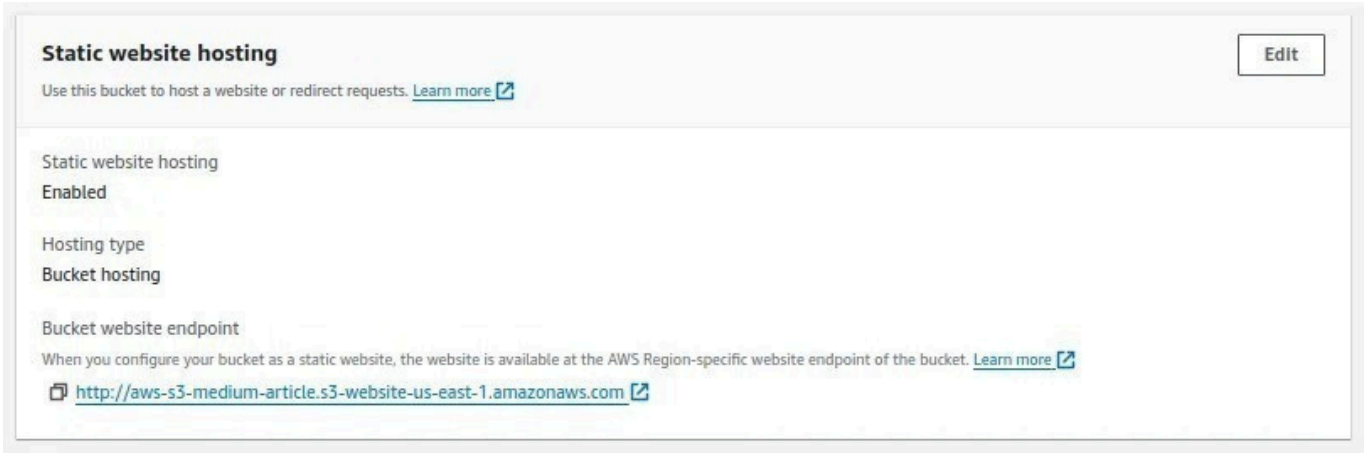
- We have already uploaded our static website contents to the S3 bucket. However, we still need a URL to access our website. Now, let's set up the bucket for static website hosting.

- By configuring your S3 bucket for hosting, you're instructing it to create a URL that will take anyone to a page displaying the files you've uploaded.
- Once again, move to your bucket page and choose the properties tab. Then scroll down until you find the "Static Website Hosting" Panel and select the "Edit" option.
- Now, "Enable" the static website hosting option.
- Configure the following settings:
  — Hosting type: Host a static website
  — Index document: In my case, the default page/home page is an index.html file. This can be different in your case.
  Keep the optional settings as the default and save the changes.
-



Static hosting configurations

- Now, once again, navigate to the static website hosting section, which is located in the properties tab on your bucket page.
- Now, you will be able to see a bucket website endpoint. Please copy and paste it into any web browser to check whether it is accessible.



Static website hosting section

Oops!!! Forbidden

403 Forbidden error message: Our files are still private

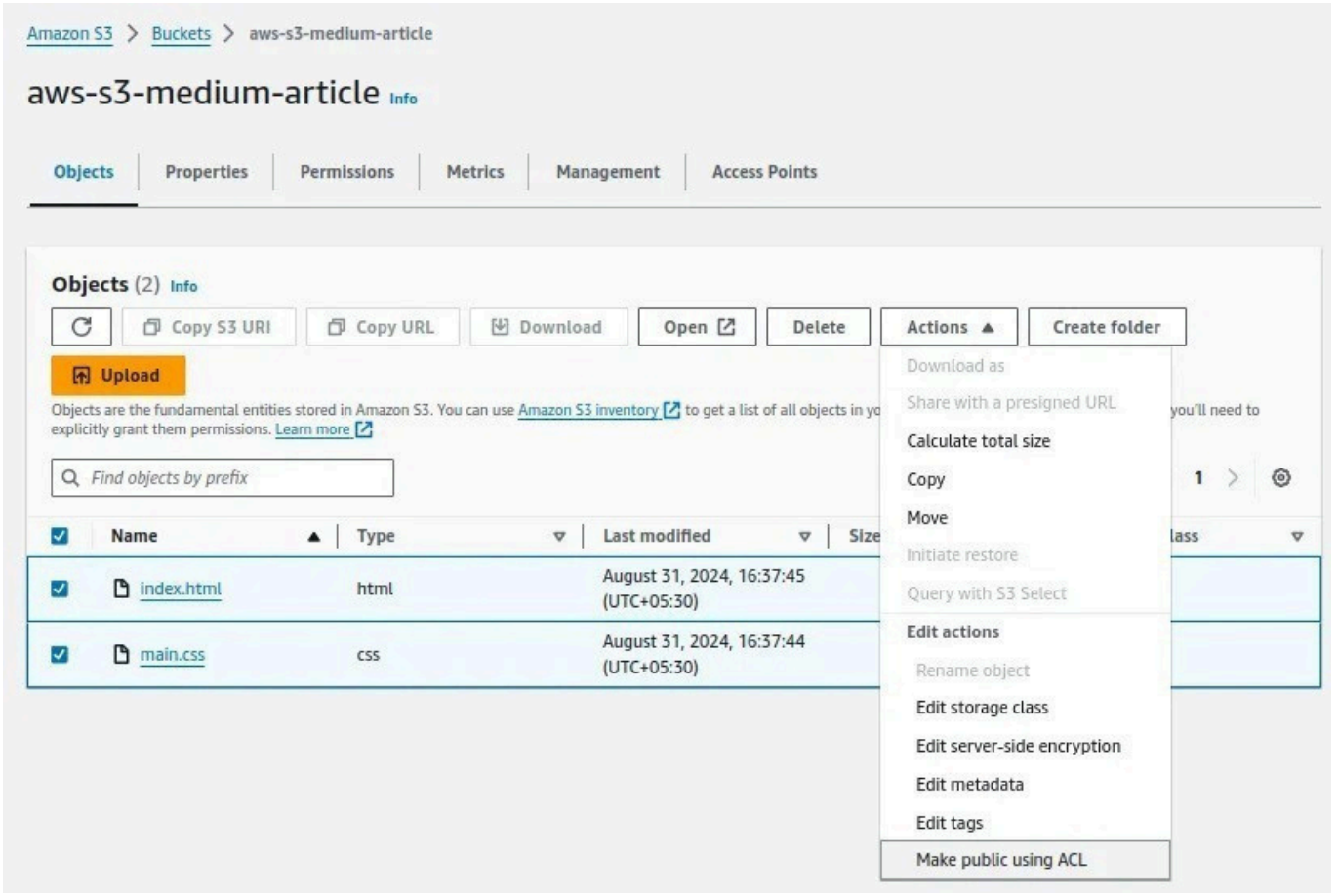
Why did we get this error?🤔

Even if our bucket is public, objects in S3 are private by default to secure your data. The error you're seeing indicates that while your static website is hosted, the uploaded files are still private. It's like having a visible bucket with its contents hidden. We need to set the object permissions to public to fix this, which is why ACLs were enabled previously.

🎯 Step 04: Configure ACLs to set your S3 bucket objects as public.

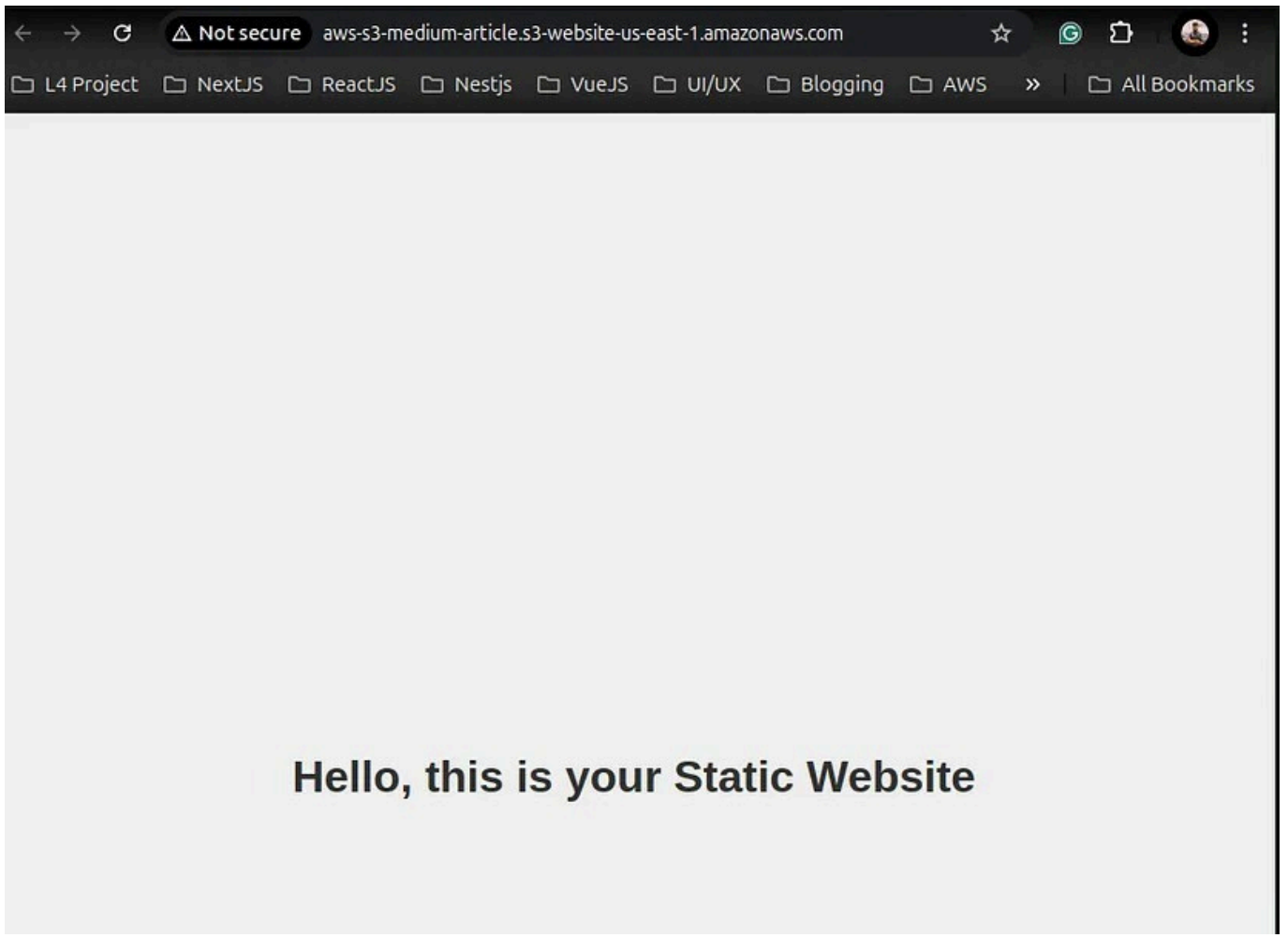
Now, let's make the objects publicly accessible through the ACLs.

- Once again, move into the bucket and select all the objects using checkboxes. Then, in the Actions tab, select the “Make public using ACL” option. Then, Choose Make Public.



Option to make public our files using ACL

Cool! We have set everything up, and your website should now be visible to everyone. Let’s refresh it.



Website preview

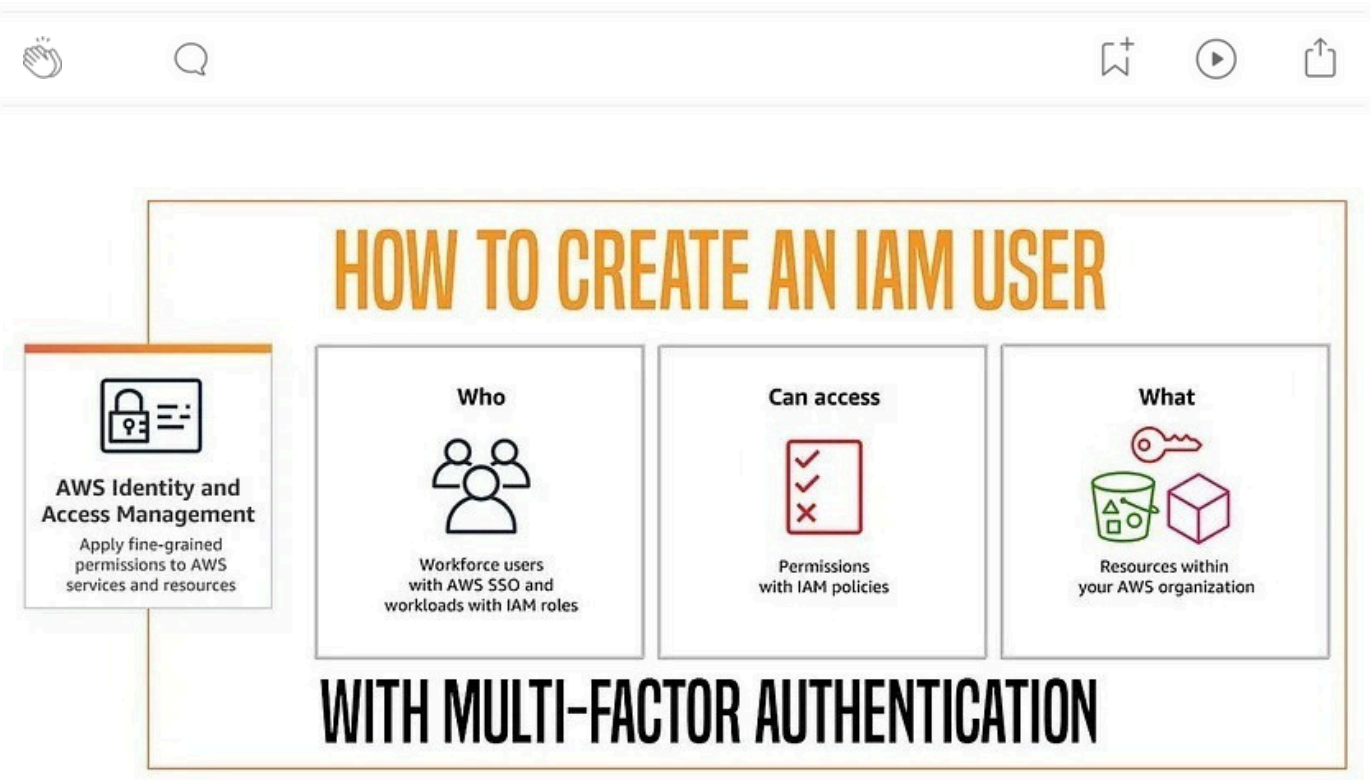
Boom!!! 🎉 Here it is — my website! 😄 Don’t laugh too hard; it’s just a simple demo I whipped up.

We’ve just wrapped up the main part of today’s session and hosted our static website on Amazon S3. Time to celebrate, right? 🎉

Tip: If you’re not using this bucket, be sure to delete it to avoid unnecessary charges.

Update the website

How to secure an AWS IAM User with Multi-Factor Authentication



To enhance security, it’s recommended to activate multi-factor authentication to protect AWS resources. MFA can be enabled on the root user and IAM users. Enabling it on the root user only affects that user. It must be enabled separately on other IAM users since each has independent

MFA configurations. There are 3 ways to set up MFA depending on the device which includes:

- Virtual MFA U2F
- Security key
- Hardware MFA

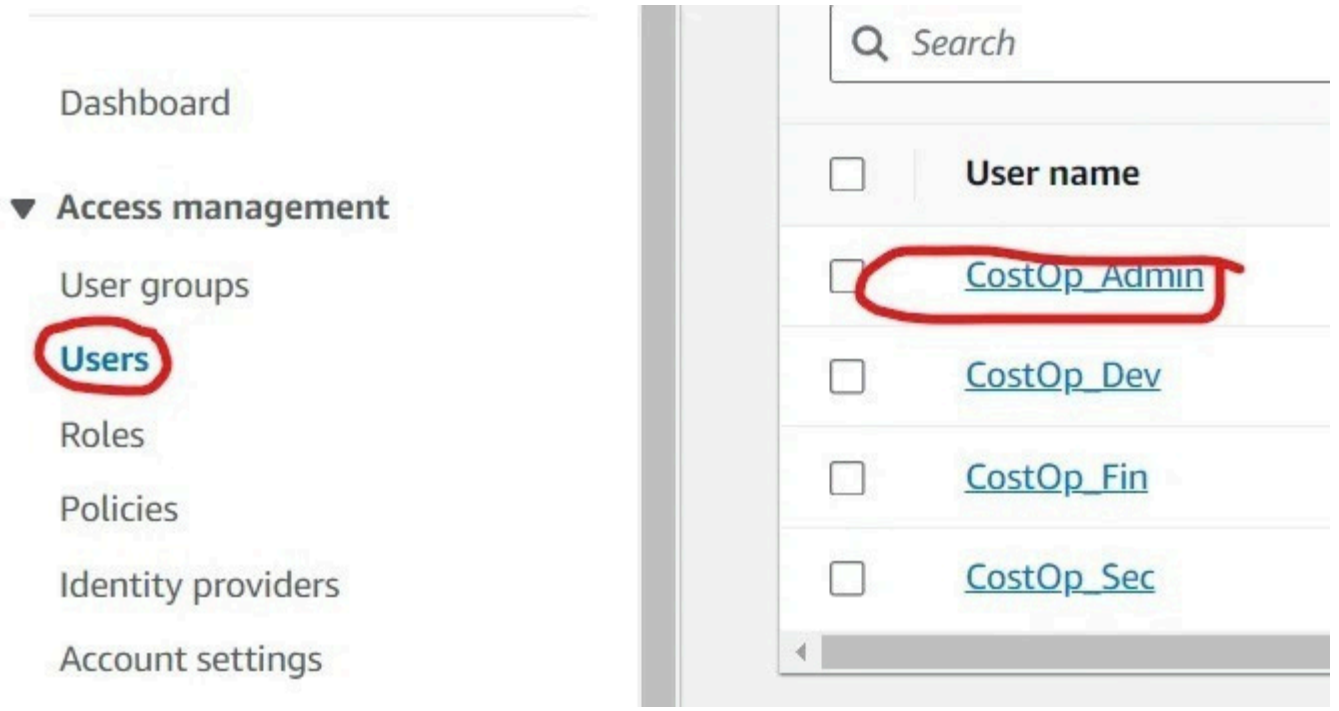
In this article, I showcase the setting up a virtual MFA device since it's the most common method used. It requires installing an authenticator app on your device which is compatible with AWS. You can find the list of authorized apps you can use on this page. We shall use the Google Authenticator for this purpose.

Enabling MFA device

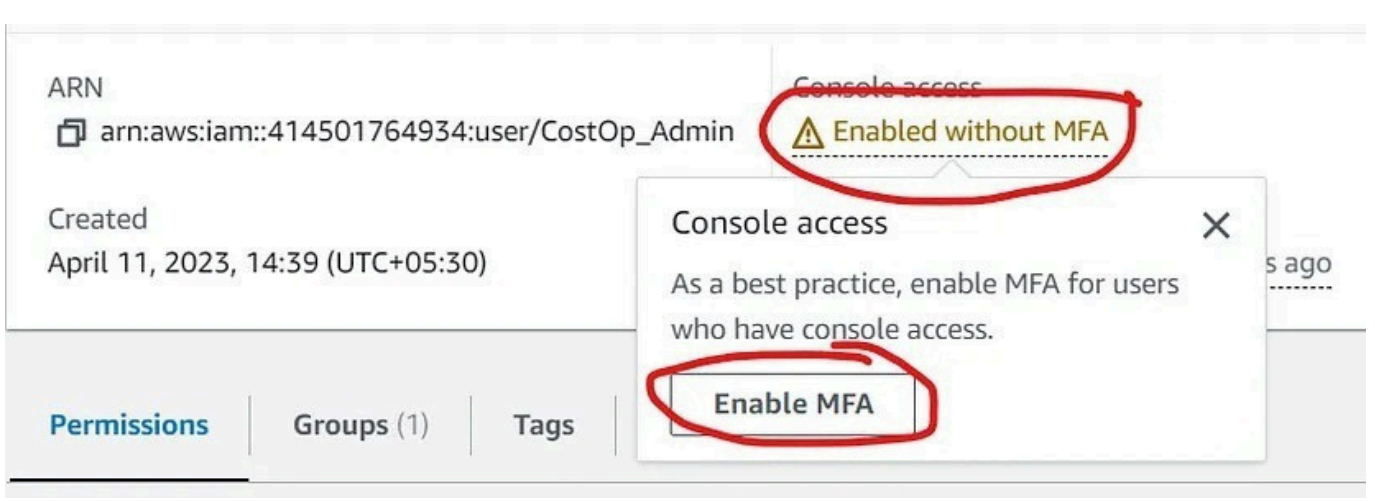
1. Sign in to the IAM administrator user you created previously and open the IAM console.

IAM SERVICES

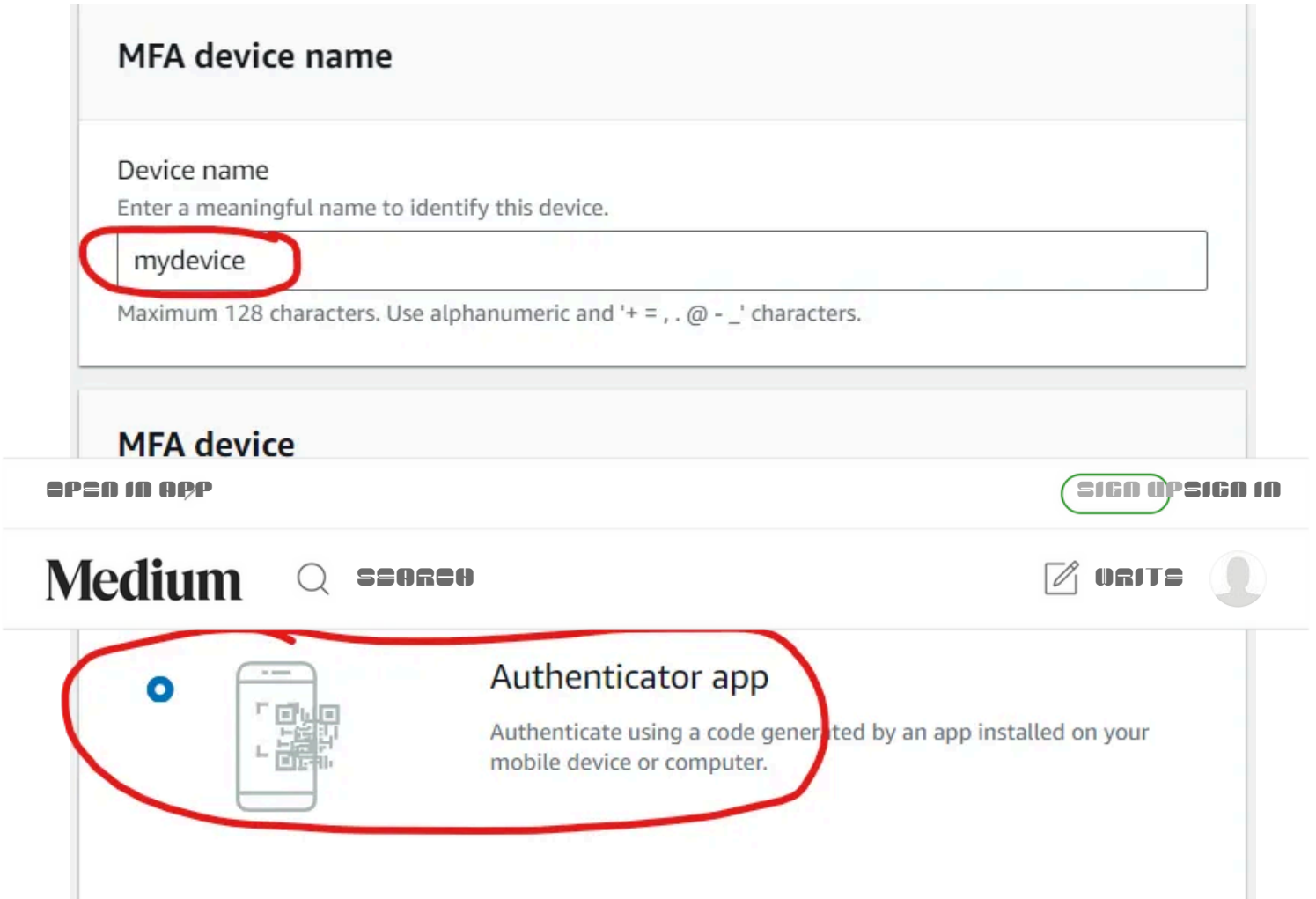
2. Choose Users in the left navigation pane and select "Administrator". Click the link to open it.



3. Click Enabled without MFA and click Enable MFA.



4. When the MFA wizard opens give your device a name such as MyMFADevice. Select Authenticator app and click Next.

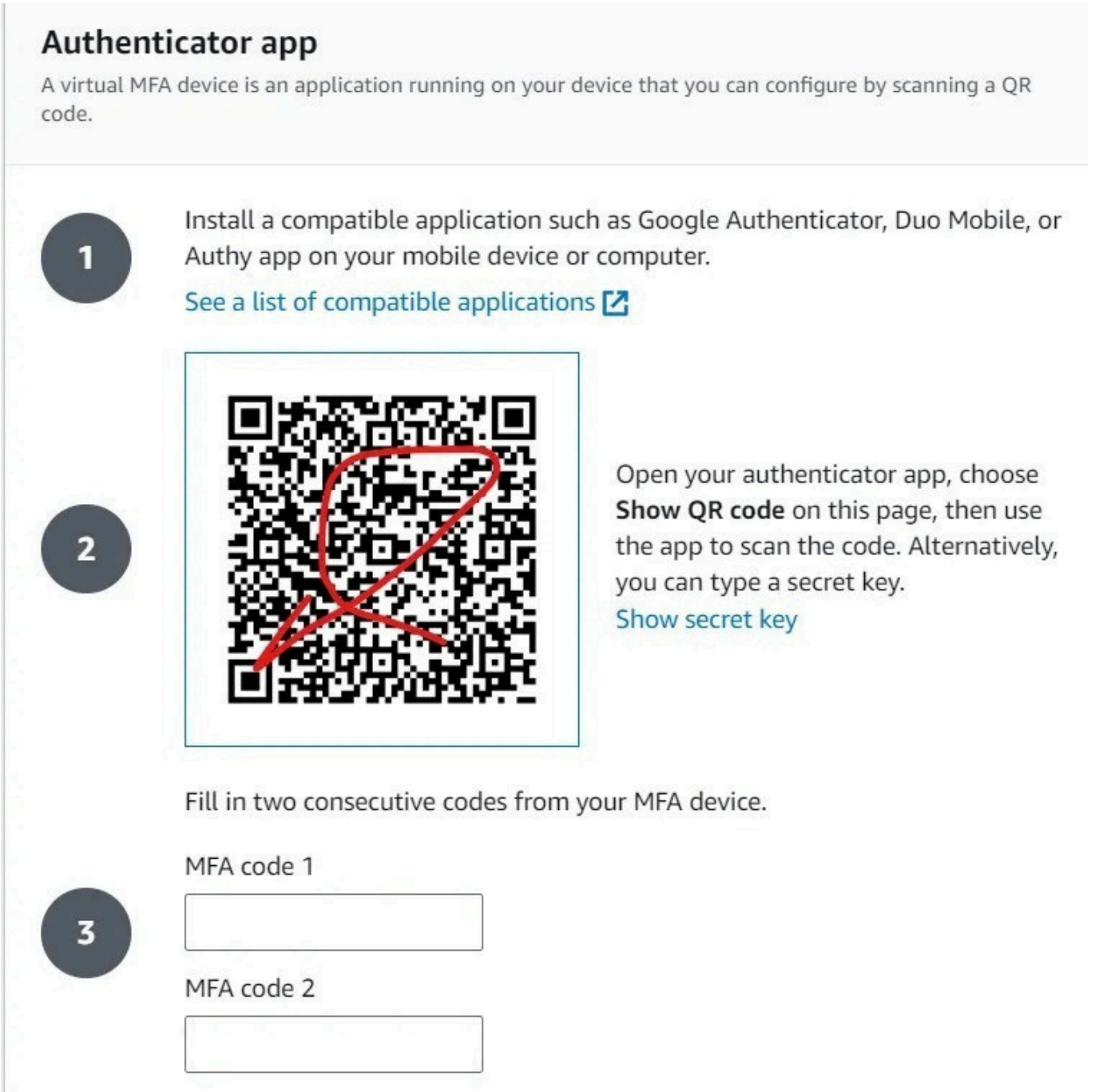


5. The console displays info for the virtual MFA device. This is when you'll add an account on the authenticator app on your device. By now you should've installed the Google Authenticator on your phone. From it you'll get a QR code for the next step. If you've not yet installed the app, please go ahead and do so. You can install apps for your smartphone from the app store that is specific to your type of smartphone. Some app providers also have web and desktop applications available. See the following table for examples:

Android	Twilio Authy Authenticator, Duo Mobile, Microsoft Authenticator, Google Authenticator, Symantec VIP
iOS	Twilio Authy Authenticator, Duo Mobile, Microsoft Authenticator, Google Authenticator, Symantec VIP

6. Open the Google Authenticator on your device. You can add an account and select the QR code option.

7. Click on Show QR Code on the MFA wizard and then use the app to scan the QR code.



8. Once you scan the QR code, the app will start generating one-time passcodes. Use these codes to fill in the MFA code 1 and code 2 boxes with two consecutive codes. Get the first one then wait for 30 seconds to get the next. Once that’s done. Click “Add MFA.”

You’ll get the confirmation that MFA is enabled for the user

To test it, simply logout and log back in with this same “Administrator” user. After entering your credentials, you’ll be prompted to enter a one-time passcode. Open the Google Authenticator and copy the code provided. The app generates codes every 30 seconds.

That’s it for this article. I hope you were enriched. We learned how to create an IAM user with administrator privileges and further secure it by enabling MFA.Thank you for reading and see you at the next one.

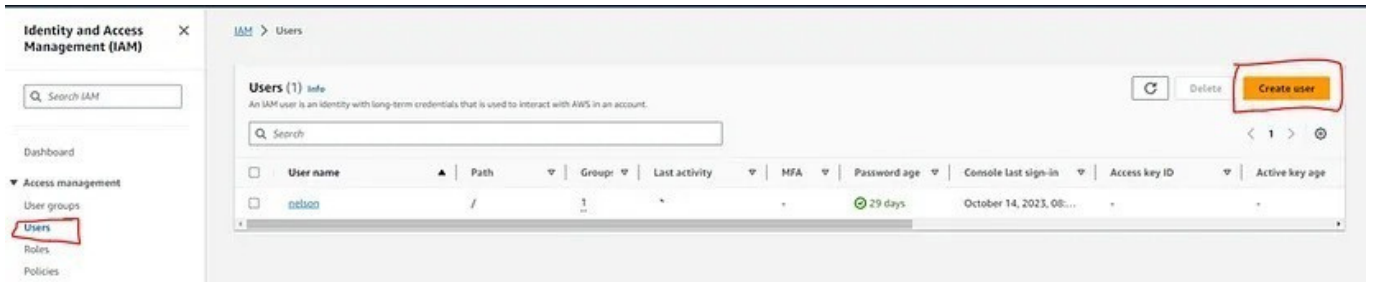
SOURCE: AWS.AMAZON.COM

How to Create an IAM User

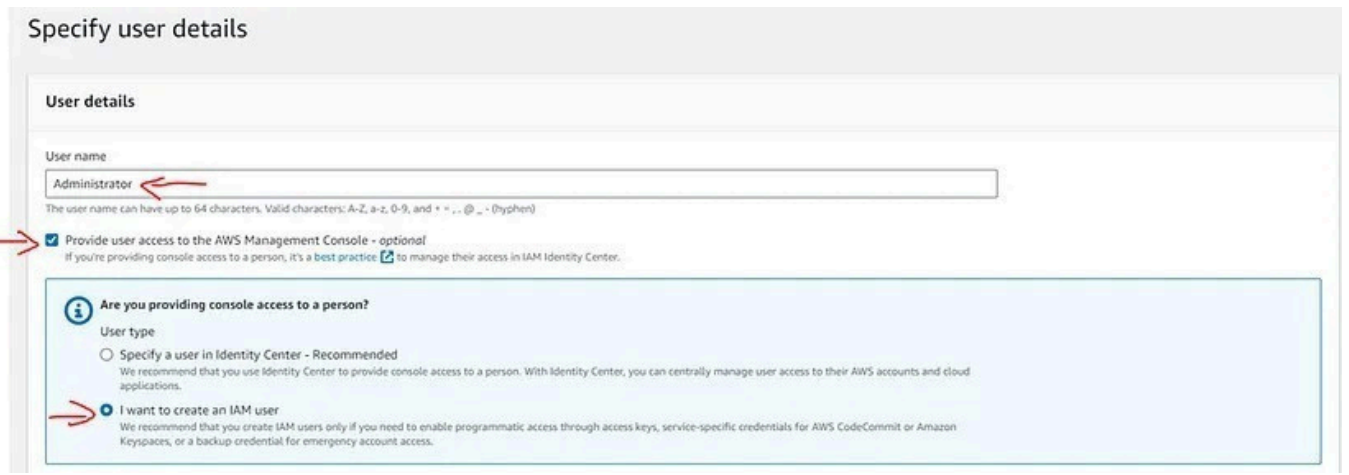
We shall see how to create an IAM User with Administrator privileges in the following steps:

1. First we need to be logged into the AWS Management console. Search forIAM and select it to open the IAM Console

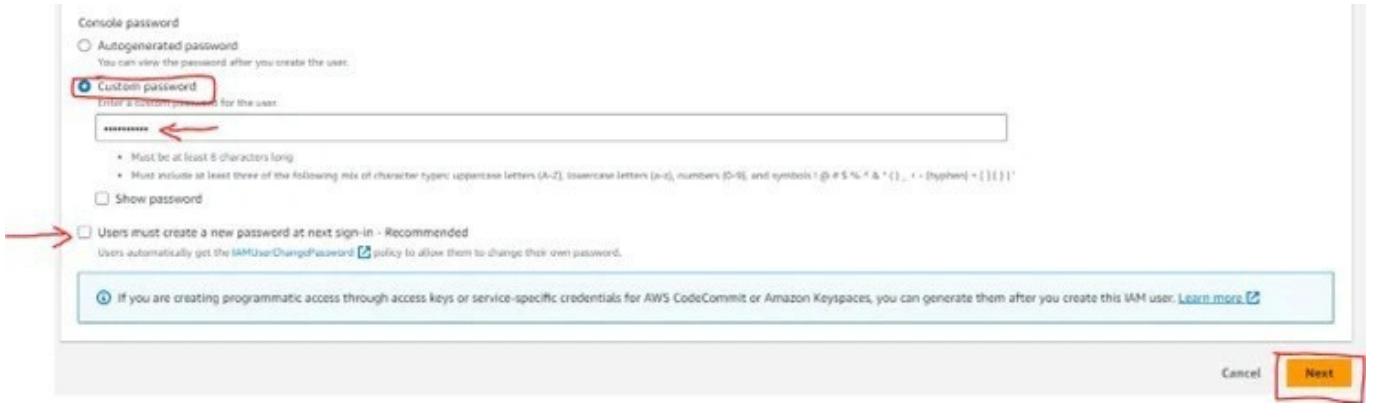
2. On the left navigation pane, click Users, then click “Create User” at the top right corner.



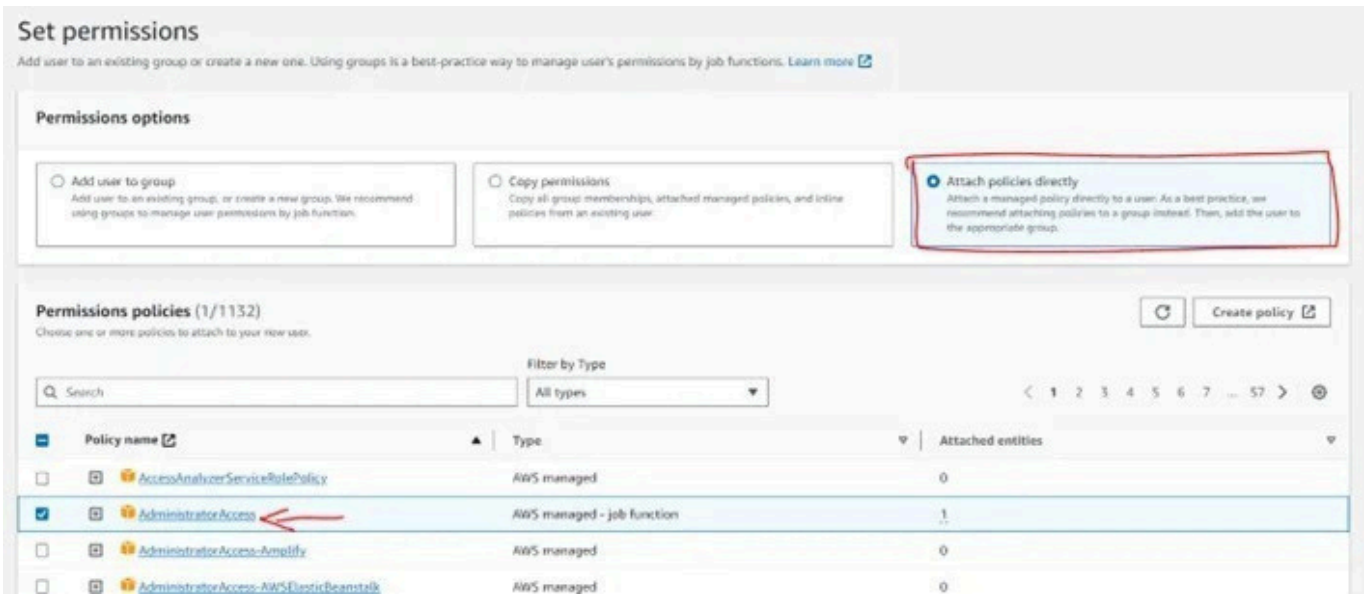
3. Specify “Administrator” as the user name which will be used to sign-in. Since this user should have administrator privileges, check the box “Provide user access to the AWS Management Console”. Check the option “I want to create an IAM user”.



4. It will open the section for us to set the password policy for this user. We can choose either Autogenerated or custom password. Let’s go with “Custom password” to give us control over it. Make sure you enter a password that respects the requirements indicated. Also, uncheck “Users must create a new password at next sign in” then click “Next”

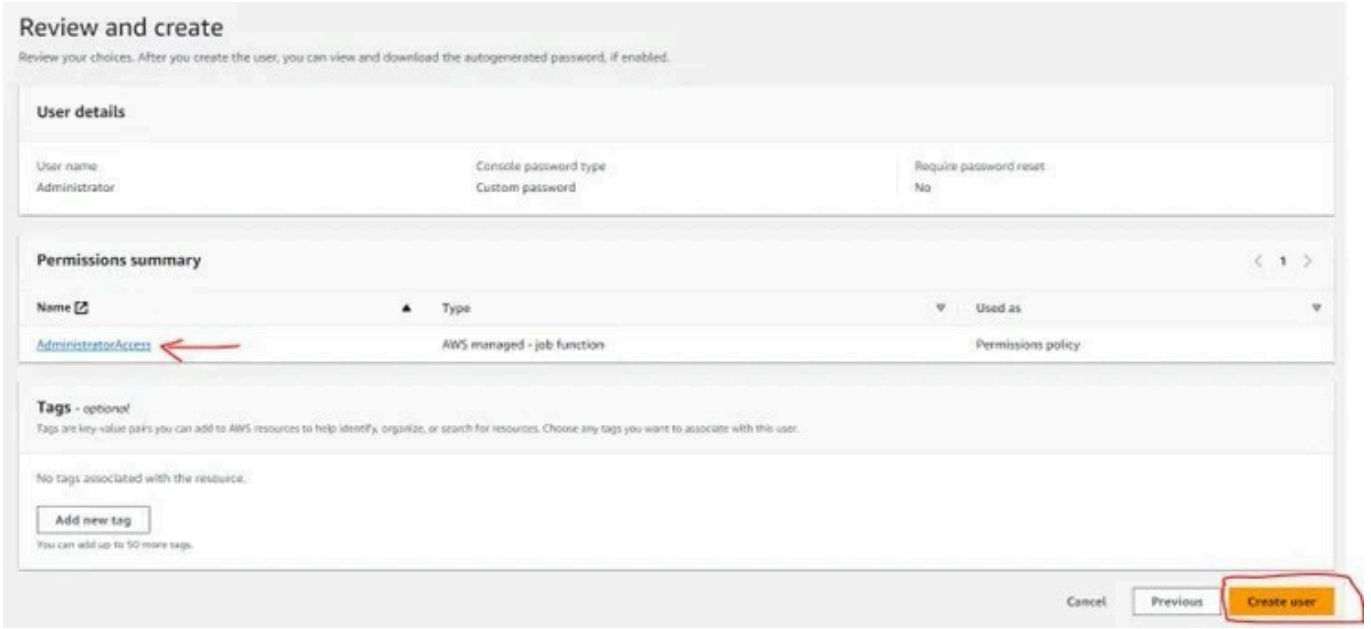


5. This takes us to set permissions for the user to select how we want to assign permission to this group of users. There are 3 options. Since we're creating an administrator user, and you'll likely be doing this for the first time, we shall go with "Attach policies directly". This will open a list of all existing policies. We shall select the "AdministratorAccess" policy to grant the necessary permissions and click "Next".

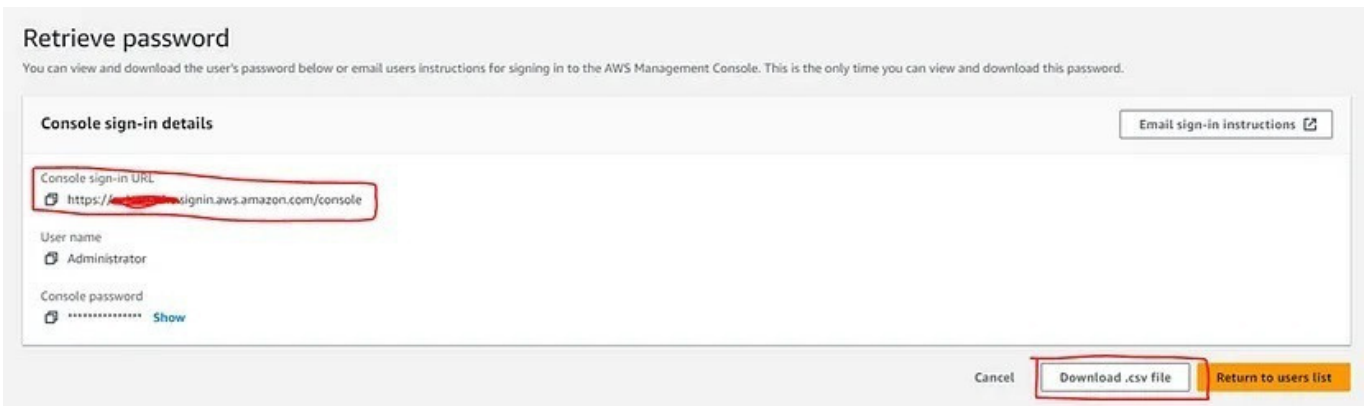


> A policy is essentially a JSON file that defines the permissions a user or group of users have.

6. Now review and take note of the credentials you created. Click "Create user."



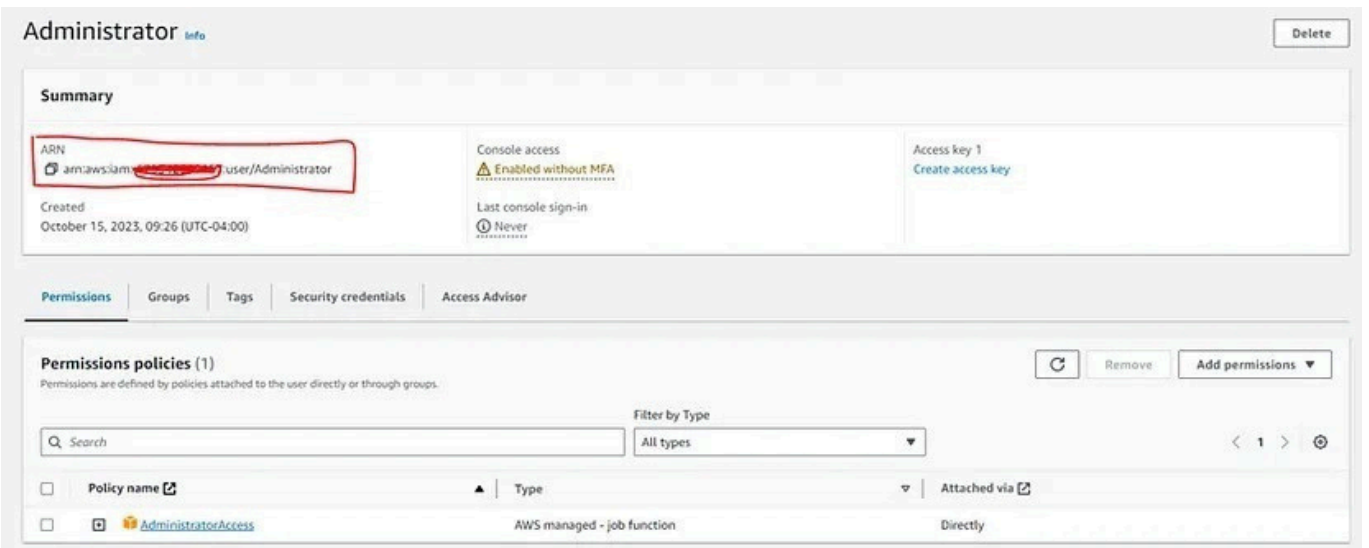
7. At this point download the sign-in details in the .csv and copy the sign-in URL. This is for your records. The URL helps to autofill your information. Then return to the users' list.



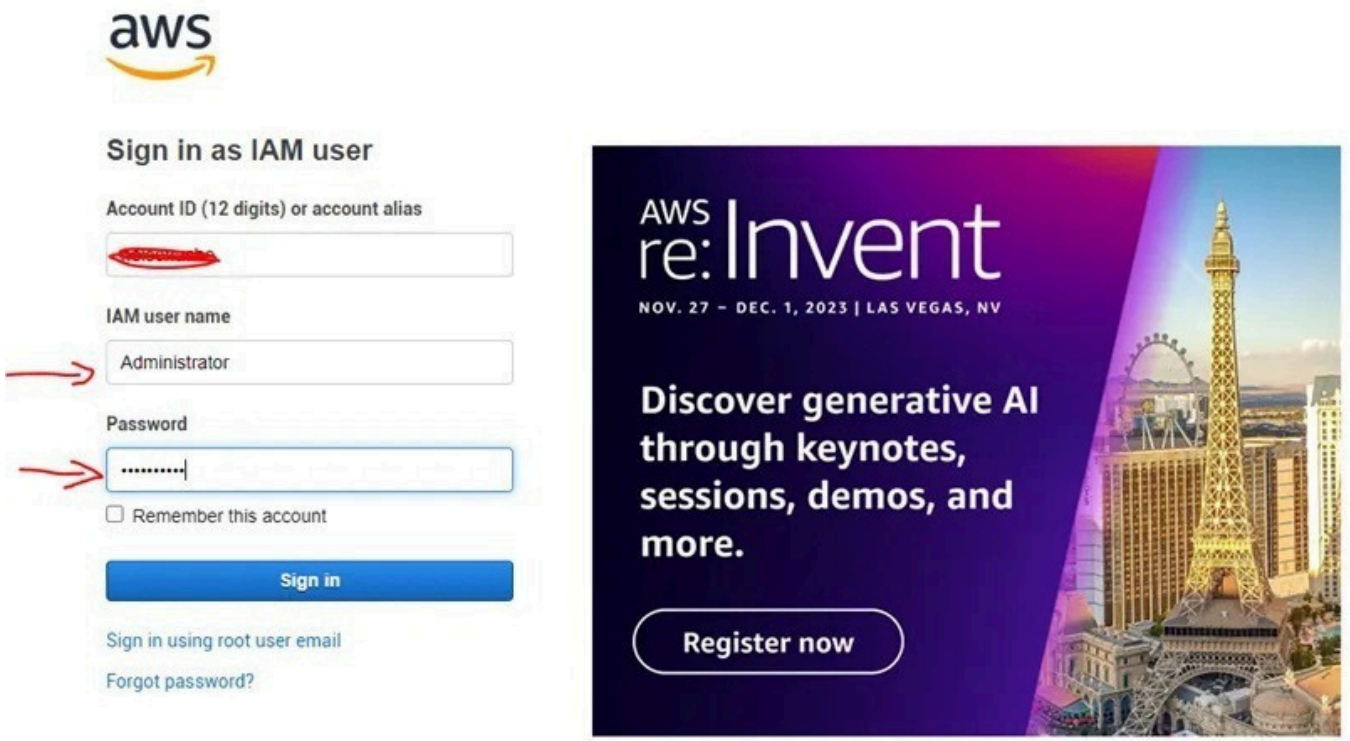
8. You can identify IAM users in the following ways:

- A friendly name used when creating the user "Administrator" in this case
- The Amazon Resource Name (ARN) that uniquely identifies the user across all of AWS.
- A unique identifier for this user.

You can find the ARN by selecting the user we just created in the IAM console.



9. Finally, test the login by navigating to the URL you copied previously. Sign in and test the permissions.



That’s it for the section. We shall dive next into setting up Multi-Factor Authentication for the admin user we just created.

How to Set up MFA on AWS

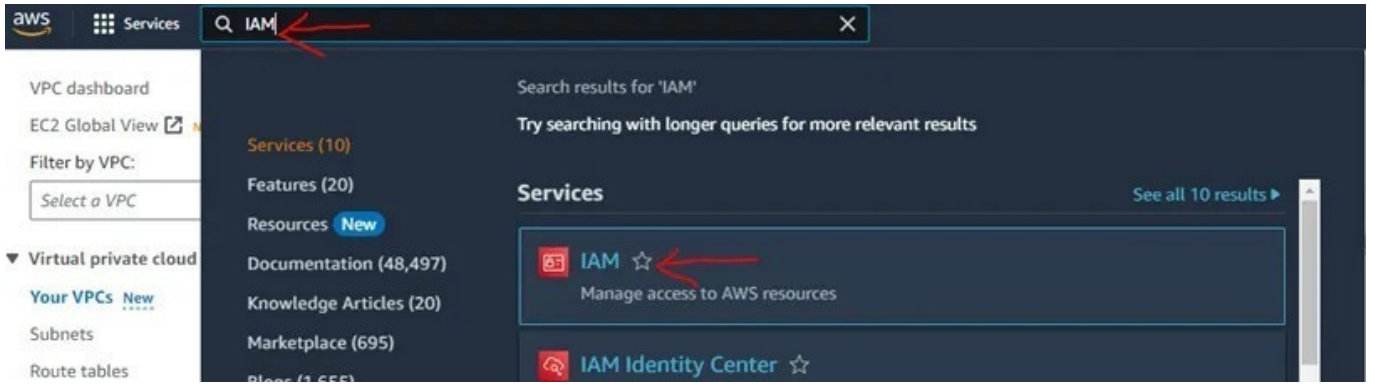
To enhance security, it’s recommended to activate multi-factor authentication to protect AWS resources. MFA can be enabled on the root user and IAM users. Enabling it on the root user only affects that user. It must be enabled separately on other IAM users since each has independent MFA configurations. There are 3 ways to set up MFA depending on the device which includes:

- Virtual MFA U2F
- Security key
- Hardware MFA

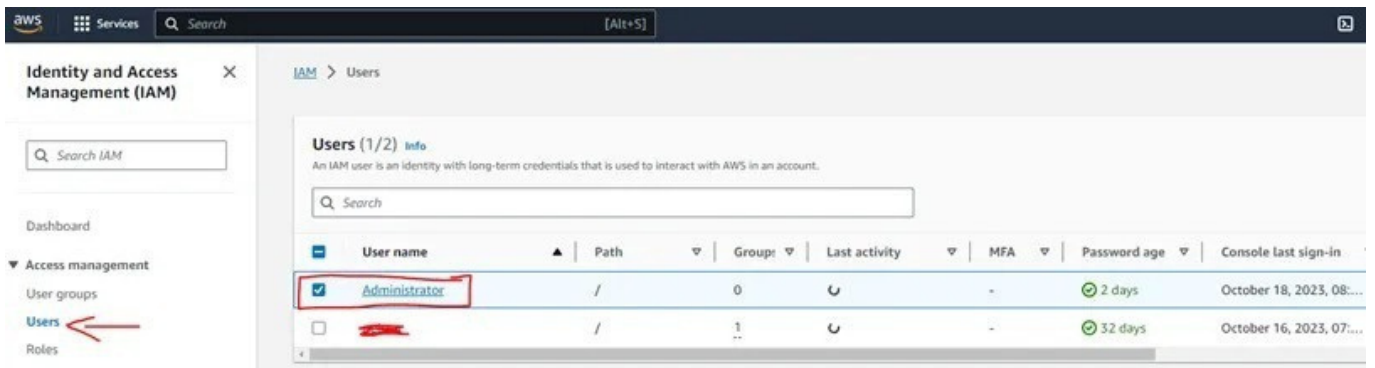
In this lab, we shall set up a virtual MFA device since it’s the most common method used. It requires installing an authenticator app on your device which is compatible with AWS. You can find the list of authorized apps you can use on this page. We shall use the Authy app for this purpose.

Enabling MFA device

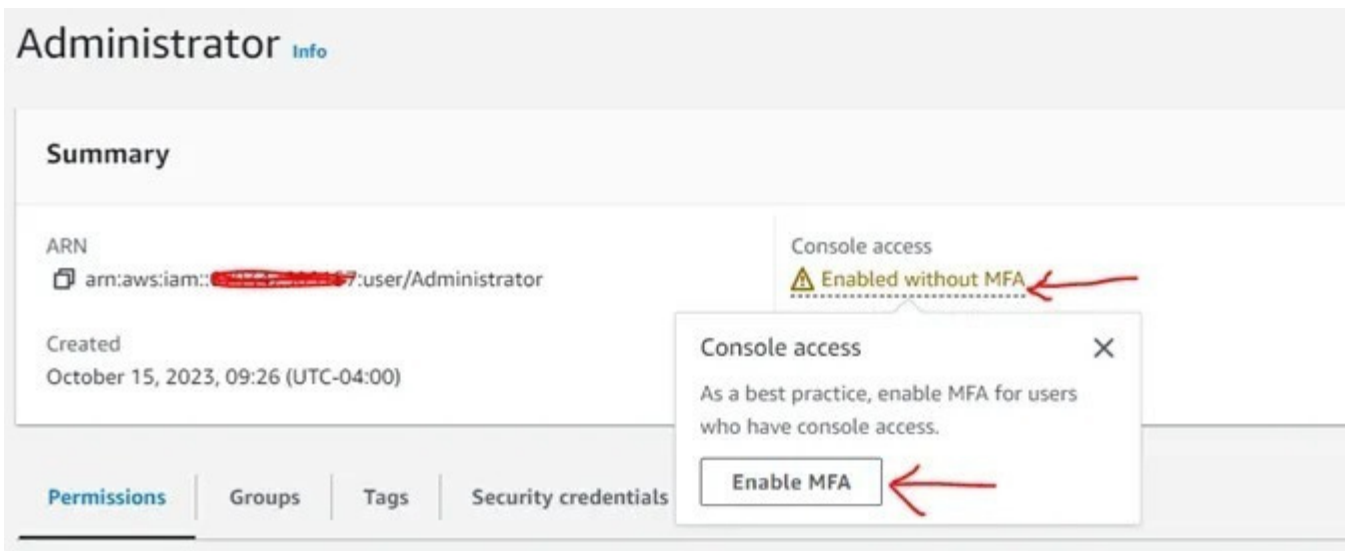
1. Sign in to the IAM administrator user you created previously and open the IAM console.



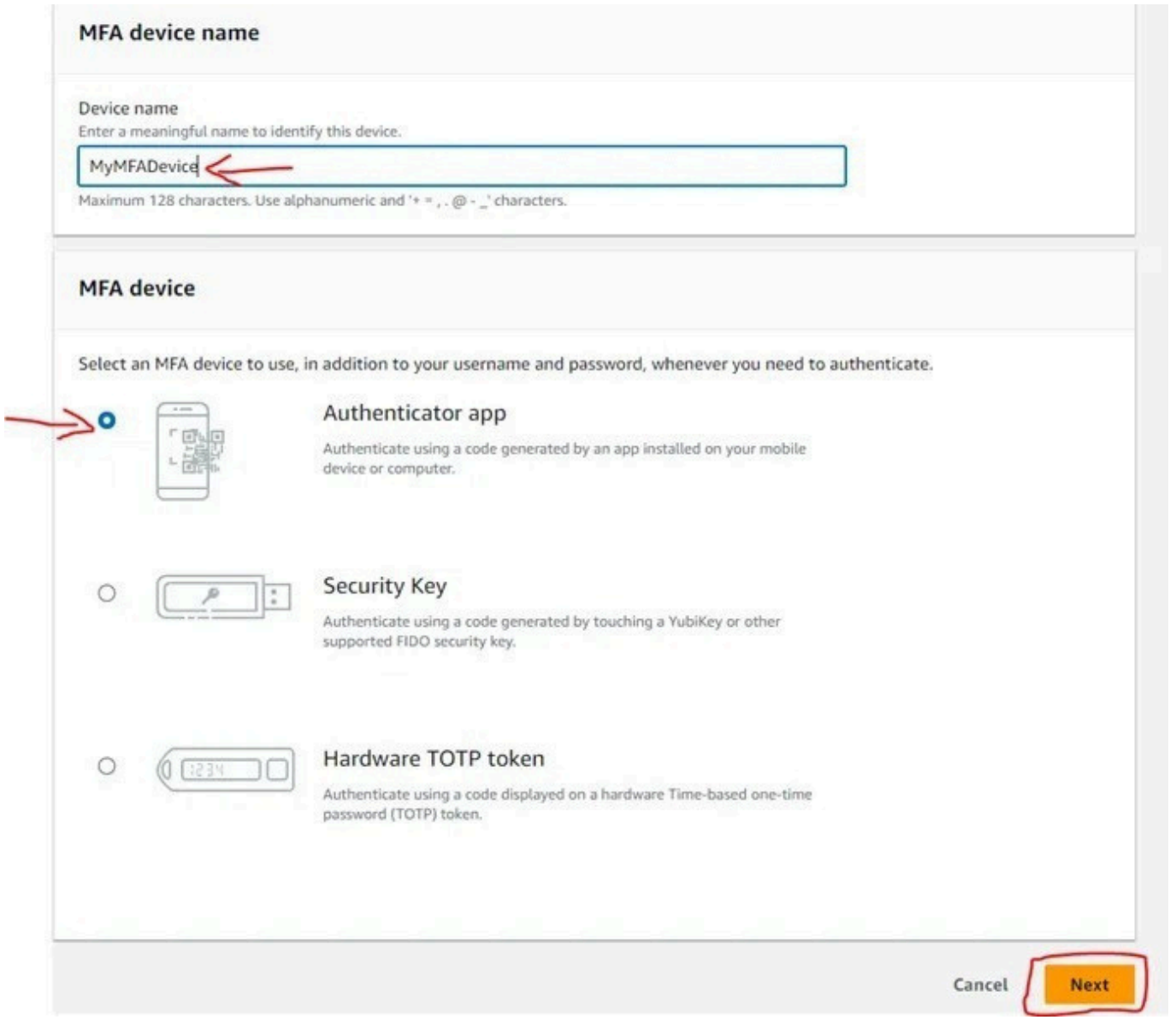
2. Choose Users in the left navigation pane and select “Administrator”. Click the link to open it.



3. Click Enabled without MFA and click Enable MFA.



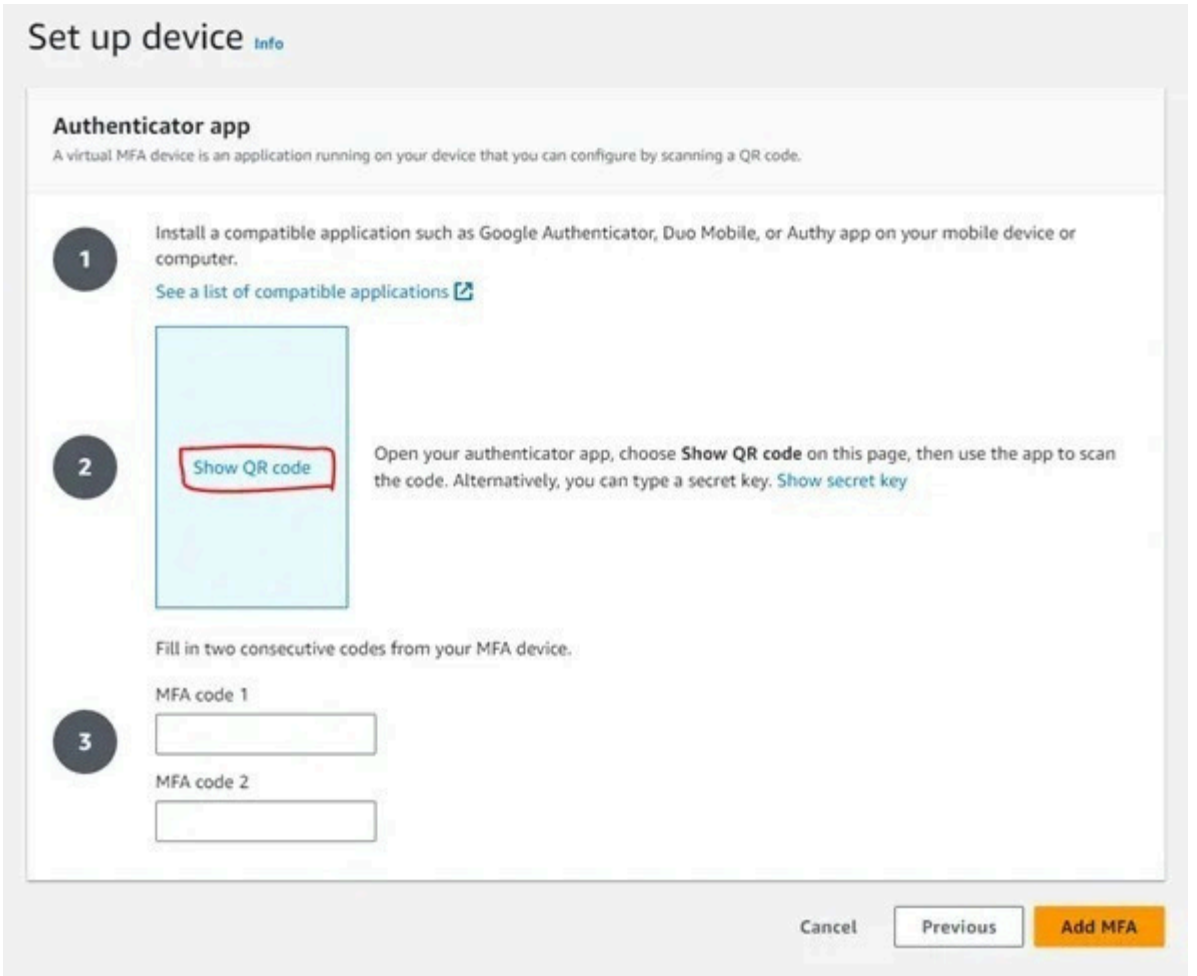
4. When the MFA wizard opens give your device a name such as MyMFADevice. Select Authenticator app and click Next.



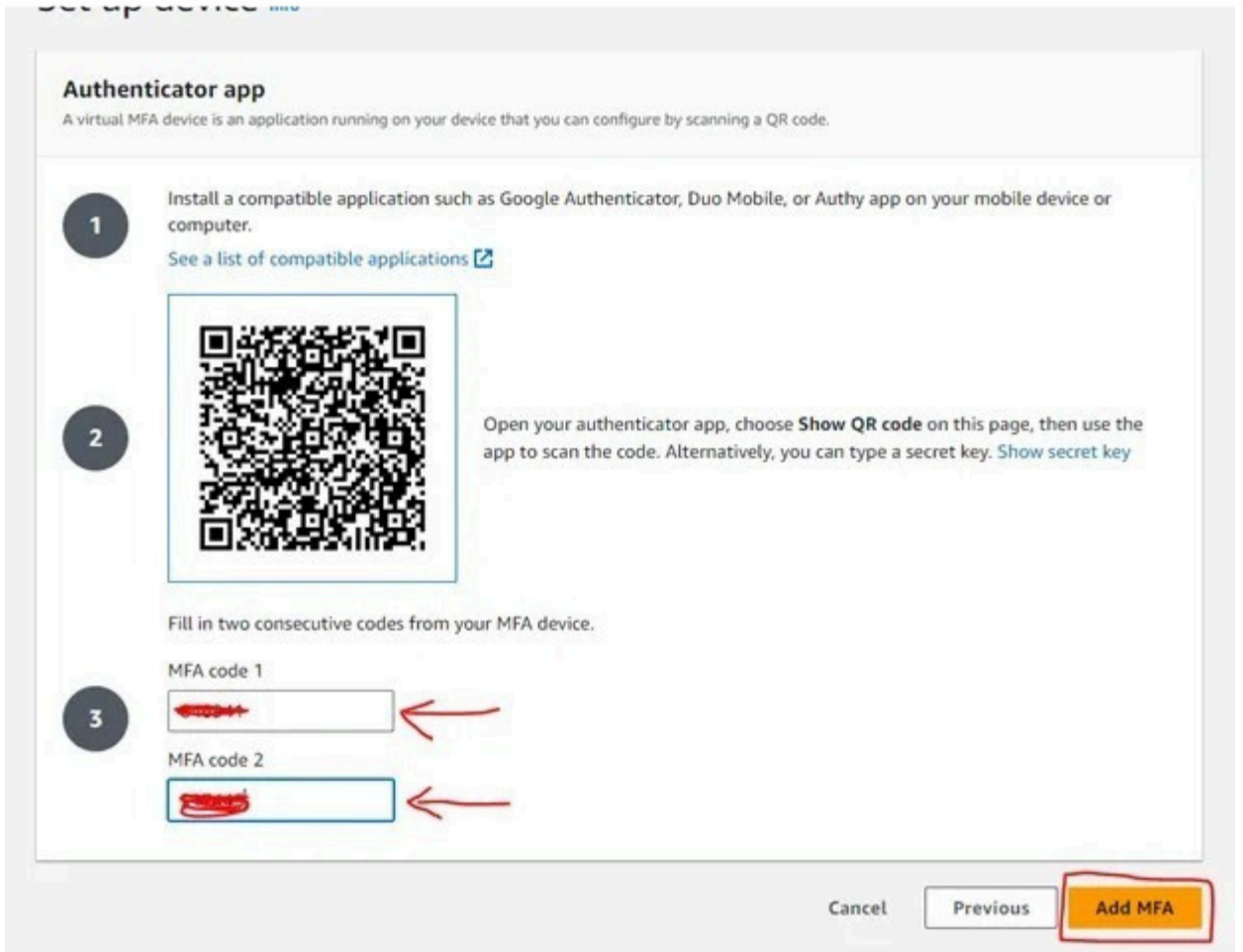
5. The console displays info for the virtual MFA device. This is when you'll add an account on the authenticator app on your device. By now you should've installed the Authy app on your phone. From it you'll get a QR code for the next step. If you've not yet installed the app, please go ahead and do so.

6. Open the Authy app on your device. You can add an account and select the QR code option.

7. Click on Show QR Code on the MFA wizard and then use the app to scan the QR code.



8. Once you scan the QR code, the app will start generating one-time passcodes. Use these codes to fill in the MFA code 1 and code 2 boxes with two consecutive codes. Get the first one then wait for 30 seconds to get the next. Once that’s done. Click “Add MFA.”



You'll get the confirmation that MFA is enabled for the user

To test it, simply logout and log back in with this same "Administrator" user. After entering your credentials, you'll be prompted to enter a one-time passcode. Open the Authy app and copy the code provided. The app generates codes every 30 seconds.

That's it for this lab. I hope you were enriched. We learned how to create an IAM user with administrator privileges and further secure it by enabling MFA. This is a good introduction to the AWS IAM service. Thank you for reading and see you at the next one.

Creating IAM Users and Groups on AWS — A Step-by-Step Guide

Let’s kick things off with a quick overview, shall we?

What is identity and access management (IAM)?

Identity and Access Management (IAM), also known as Identity Management, plays a vital role in securing cloud resources by managing user identities and controlling their access. It acts like a vigilant bouncer at a nightclub entrance, checking the guest list and granting access to the VIP areas only to authorized individuals.

Why even IAM is important?

IAM plays a vital role in keeping data safe and secure while enhancing overall efficiency. Here are the key reasons why IAM matters:

1. Keeping Data Safe: IAM protects sensitive data from hackers anudnauthorized people. It uses strong security measures like passwords and extra verification steps to make sure only the right people can access the data.
2. Making Things Easier: IAM makes it easy for employees to access the things they need quickly. This helps them work faster and saves time by avoiding long access requests.
3. Following Rules: Businesses must follow rules about data protection. IAM helps them meet these rules by controlling who can access data and keeping records of who does.
4. Organizing Access: IAM manages who has access to what in an organized way. It ensures that each person only gets the access they need for their job, reducing the risk of insider issues.

What is an IAM User?

IAM users are like representatives for people or applications using AWS. They can access AWS through the console or programmatically. IAM users help manage access to AWS resources securely, and you can set permissions for them individually or in groups.

They have names and passwords for console access and can create access keys for programmatic access. IAM users ensure only authorized users and apps can access your AWS resources.

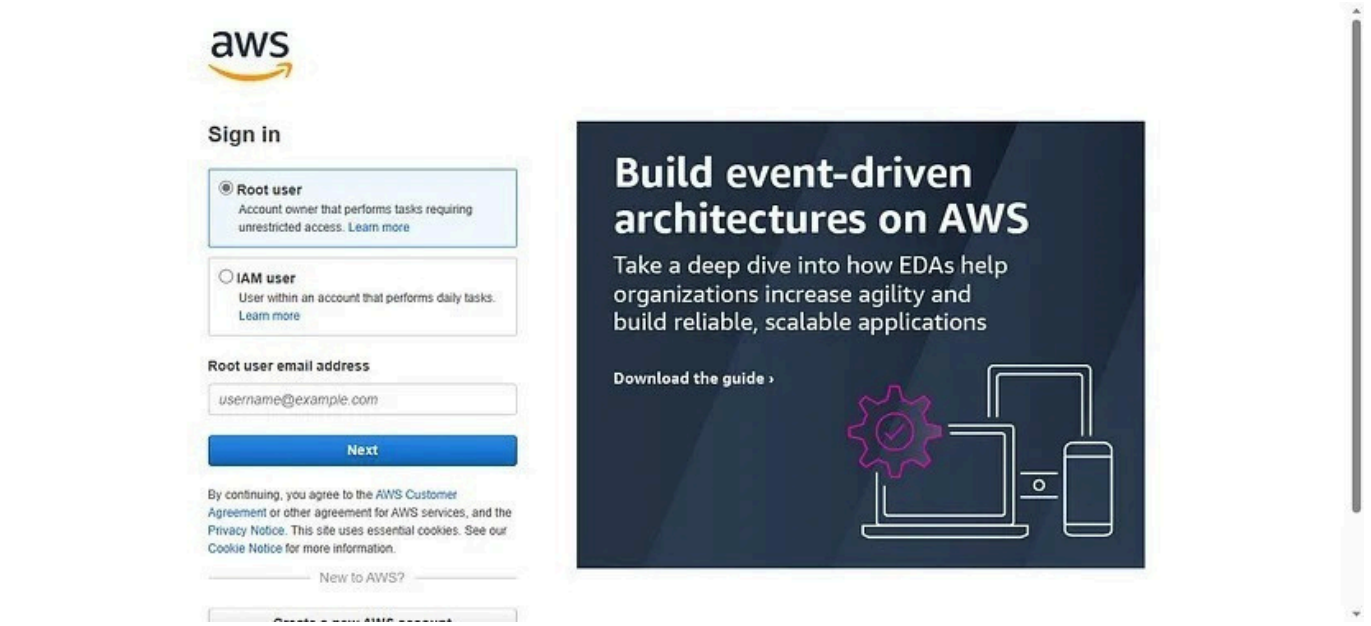
What are IAM Groups?

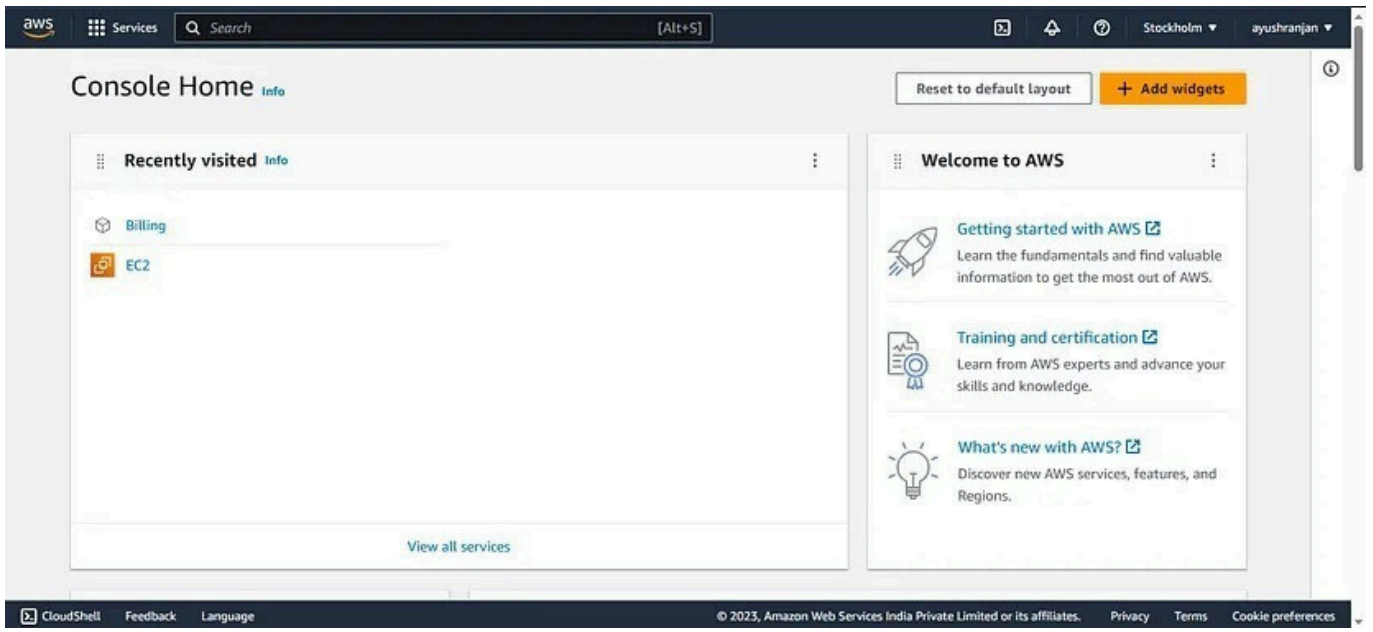
An IAM group is a collection of users who are grouped together based on the same access control policies. These policies define what actions the group members are allowed to perform on specific objects within the group's scope.

For example, let's say you have a group called "Developers." In the group's access control policy, you grant read-only access to all of your EC2 instances. Now, any user added to the "Developers" group will automatically have the permission to view information about those EC2 instances, but they won't have permission to make any changes to them.

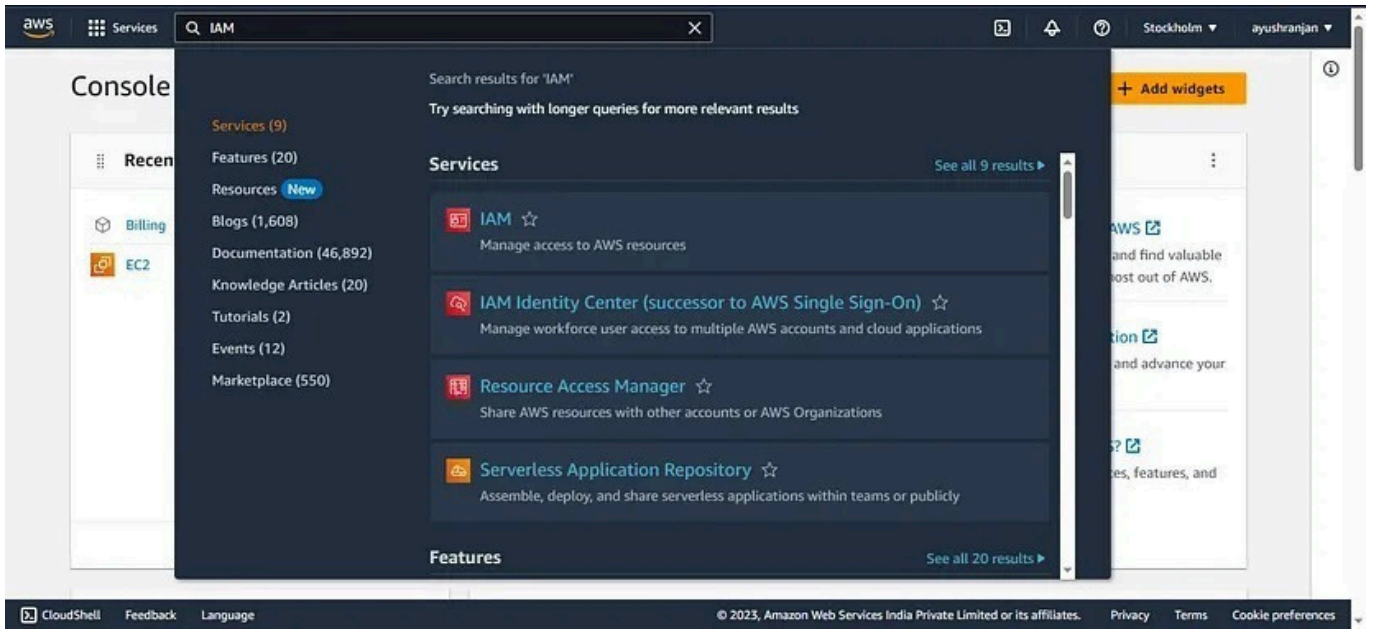
Step-by-Step Guide

Here is a step-by-step guide for creating IAM groups and users.
STEP 1: Signing into your AWS account as a root user.

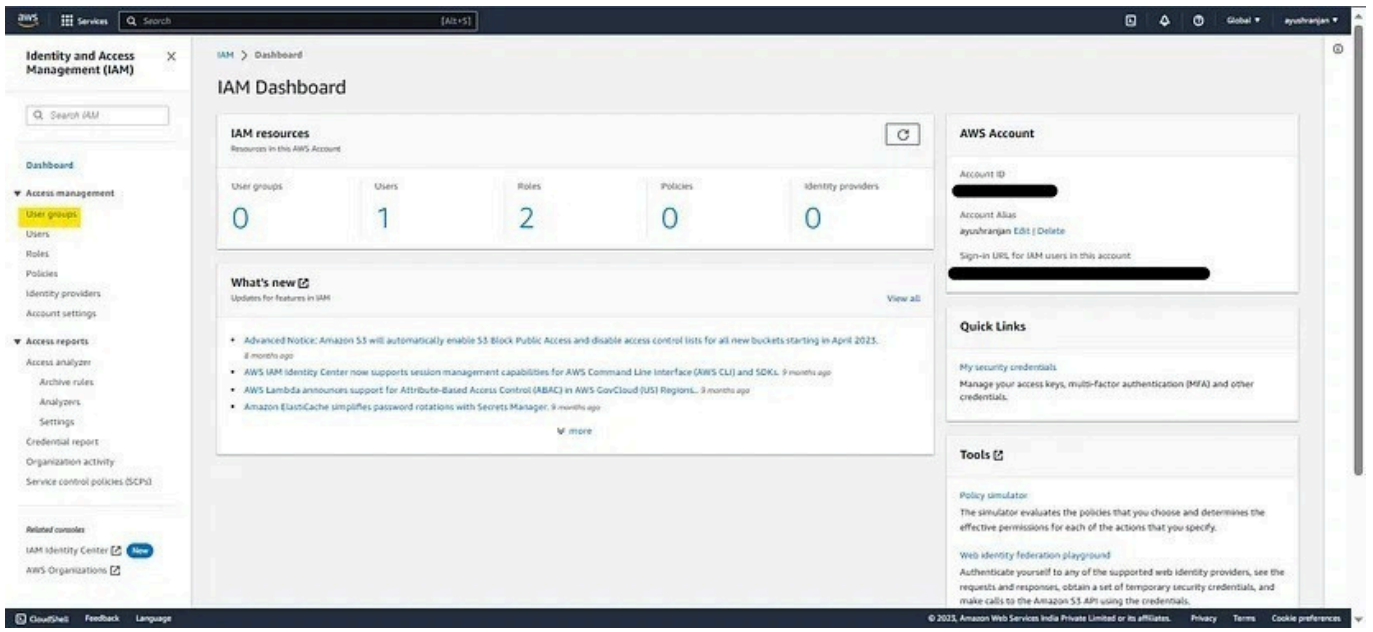




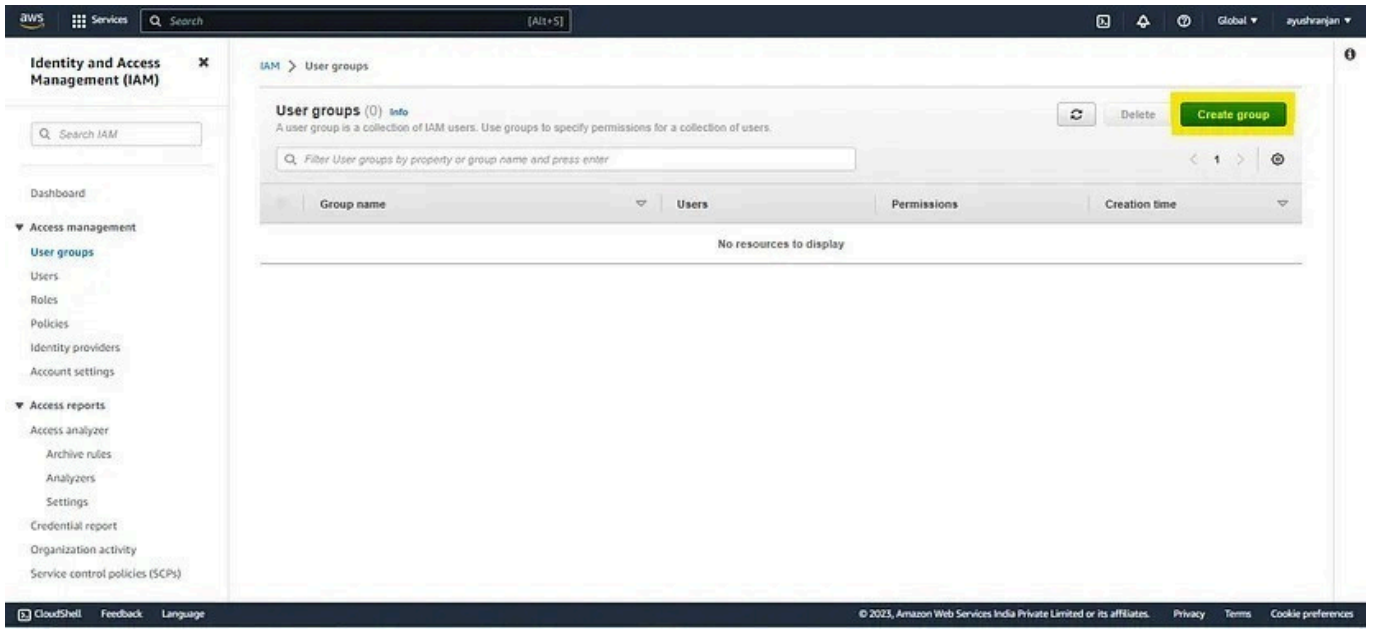
STEP 2: Search for IAM in search bar.



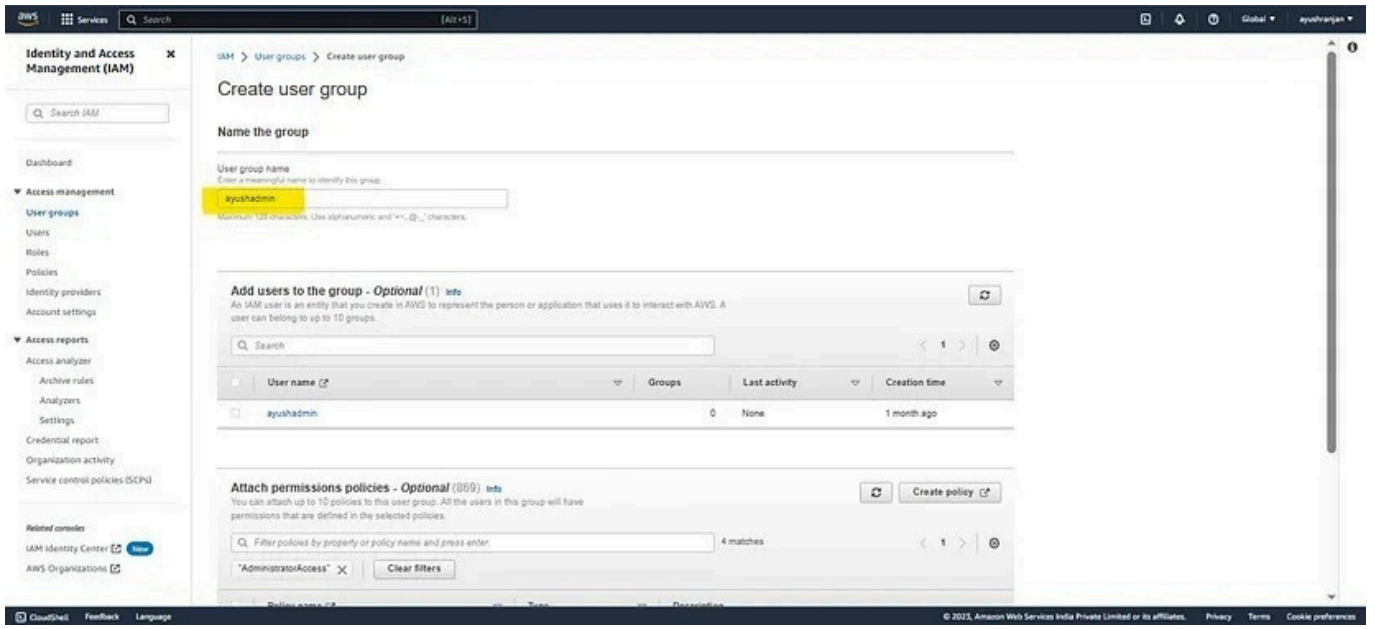
You can see the IAM dashboard as below.



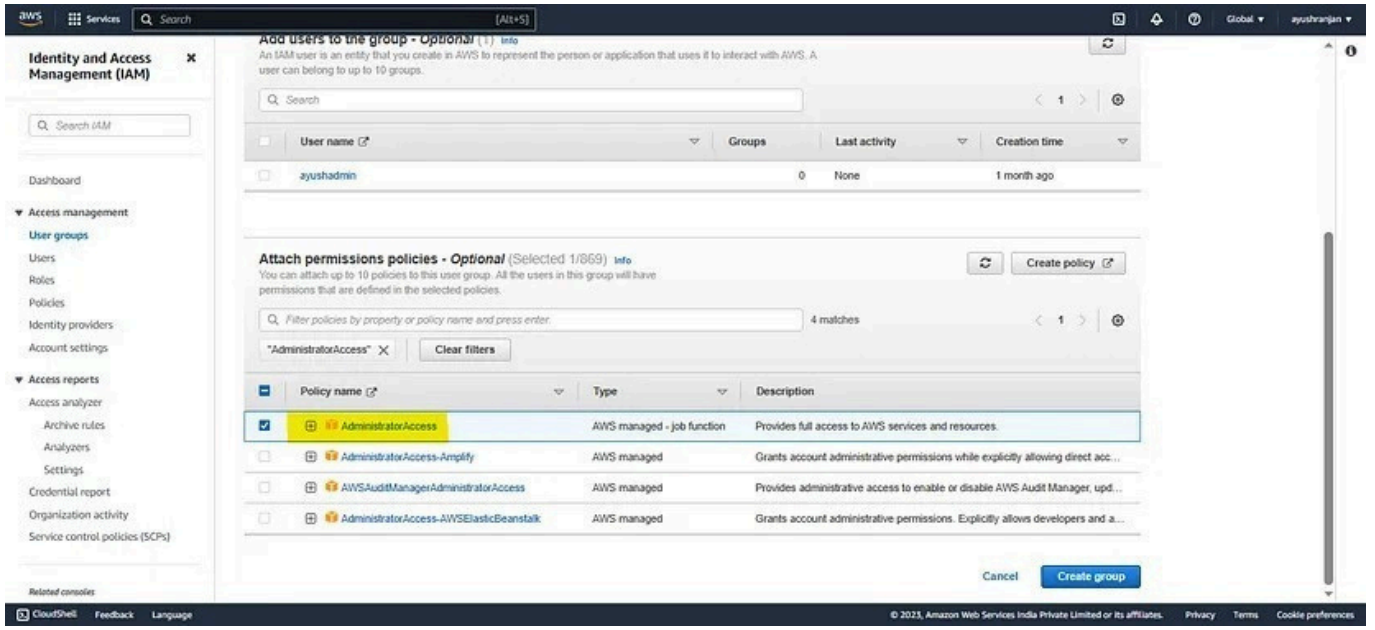
STEP 3: Click on Create Group button.



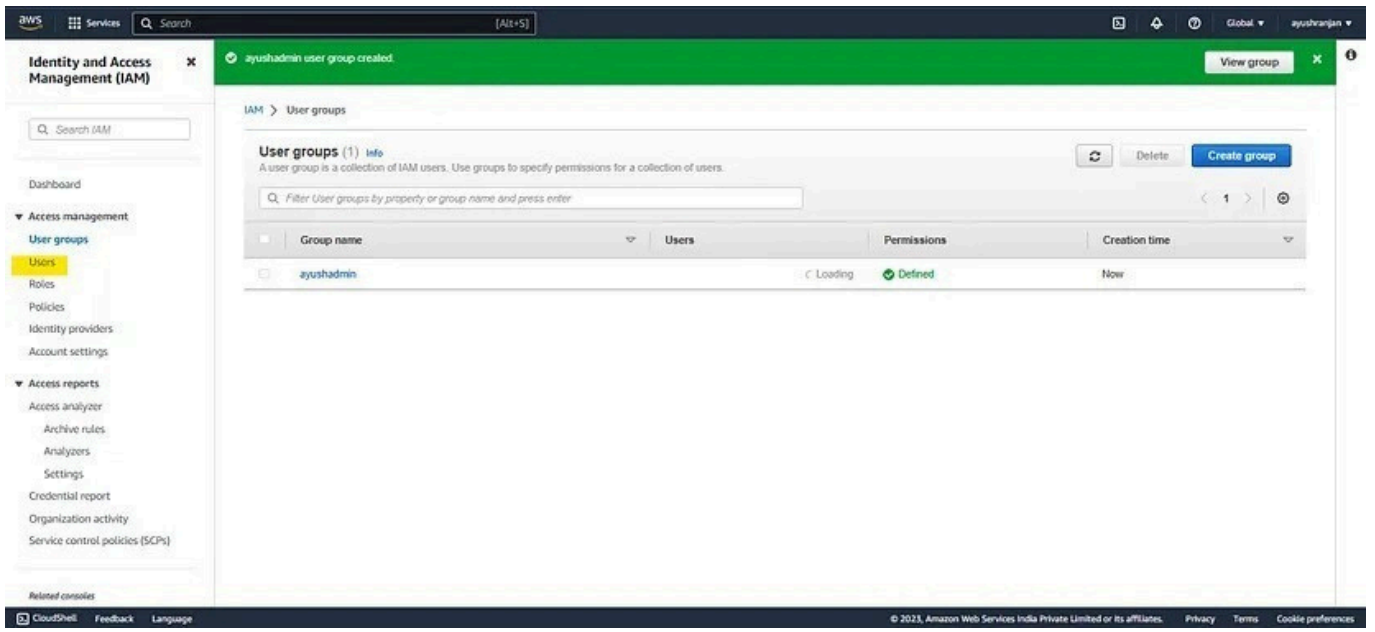
STEP 4: Write the name you want to give to User Group.



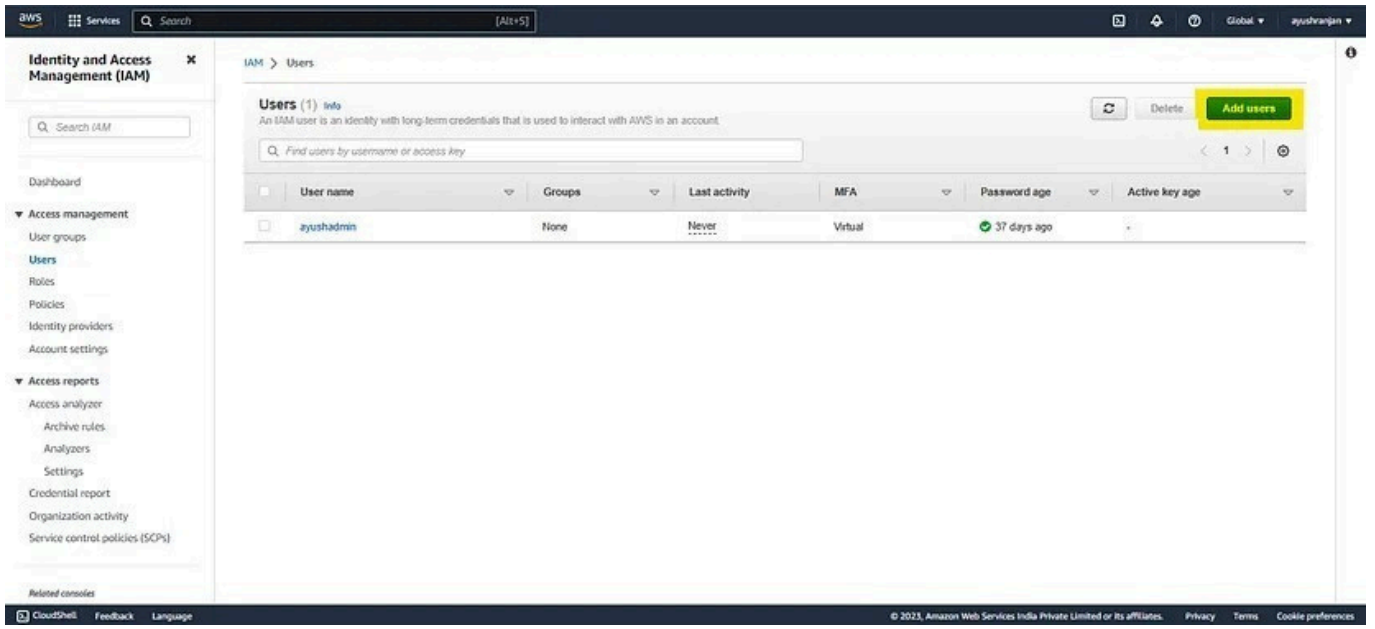
STEP 5: Please select the desired permissions in the permission section. In this case, I have granted administrator access.



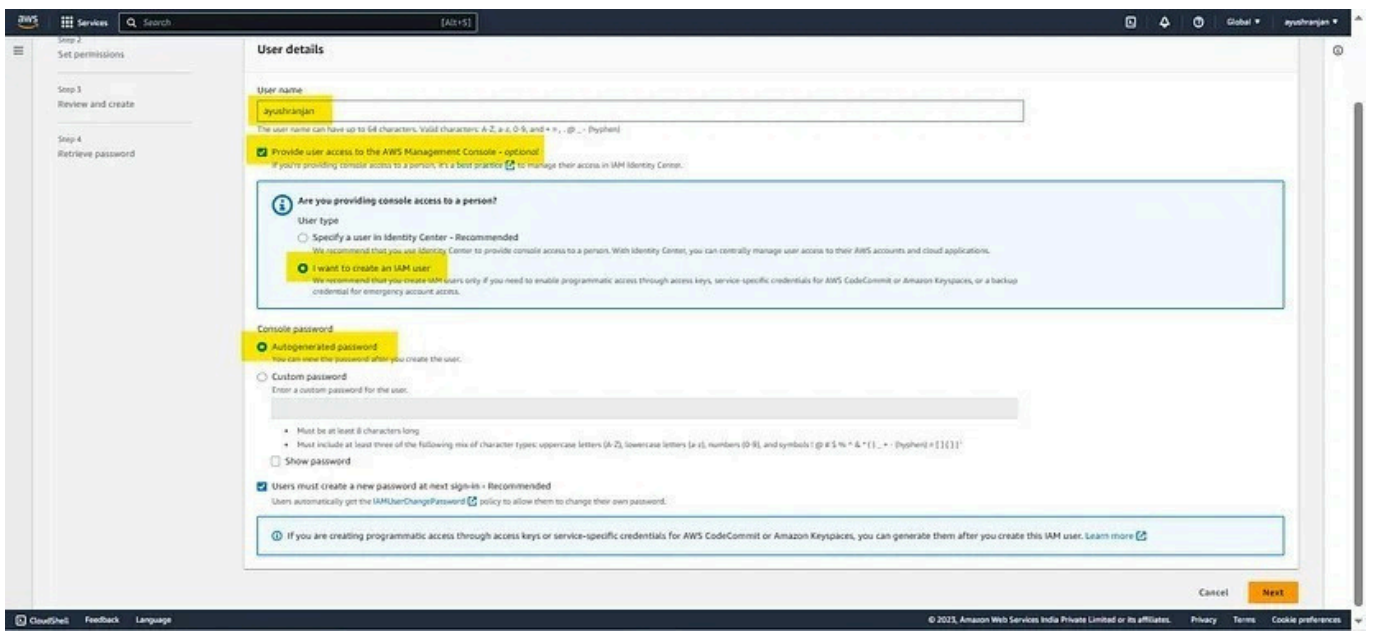
Then click on create group.



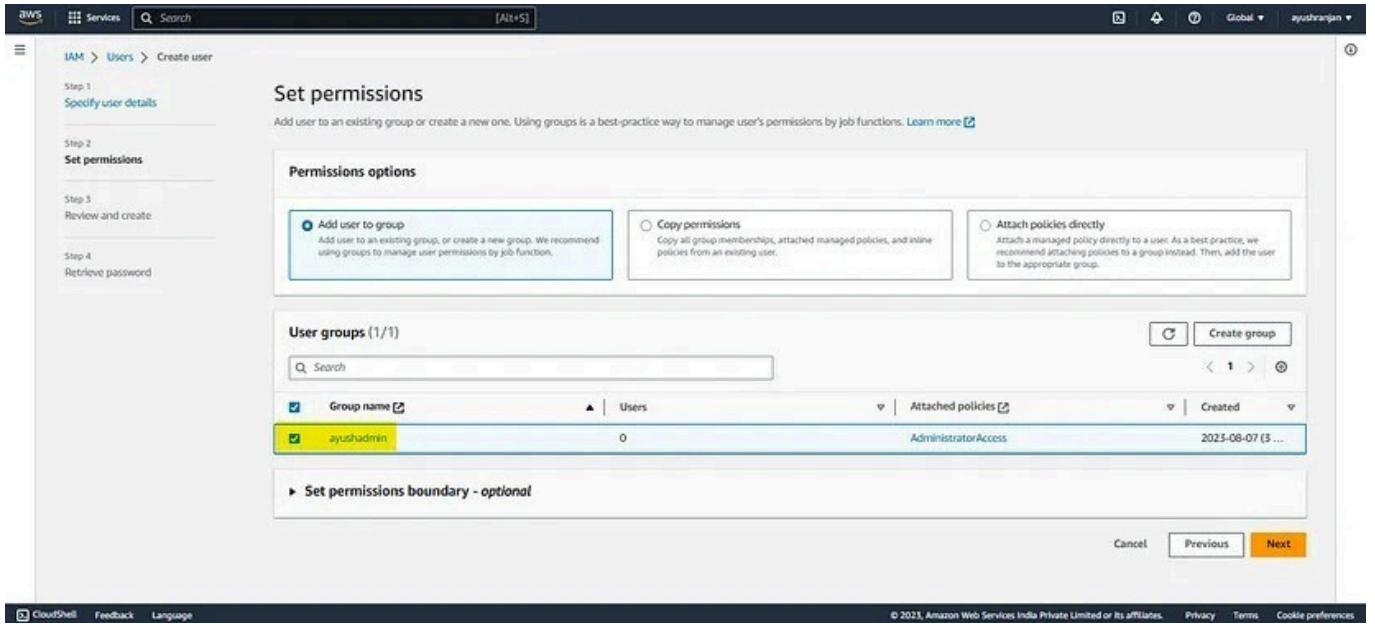
STEP 6: To create a new user, navigate to the User section located below the User Group section and select the "Add Users" option.



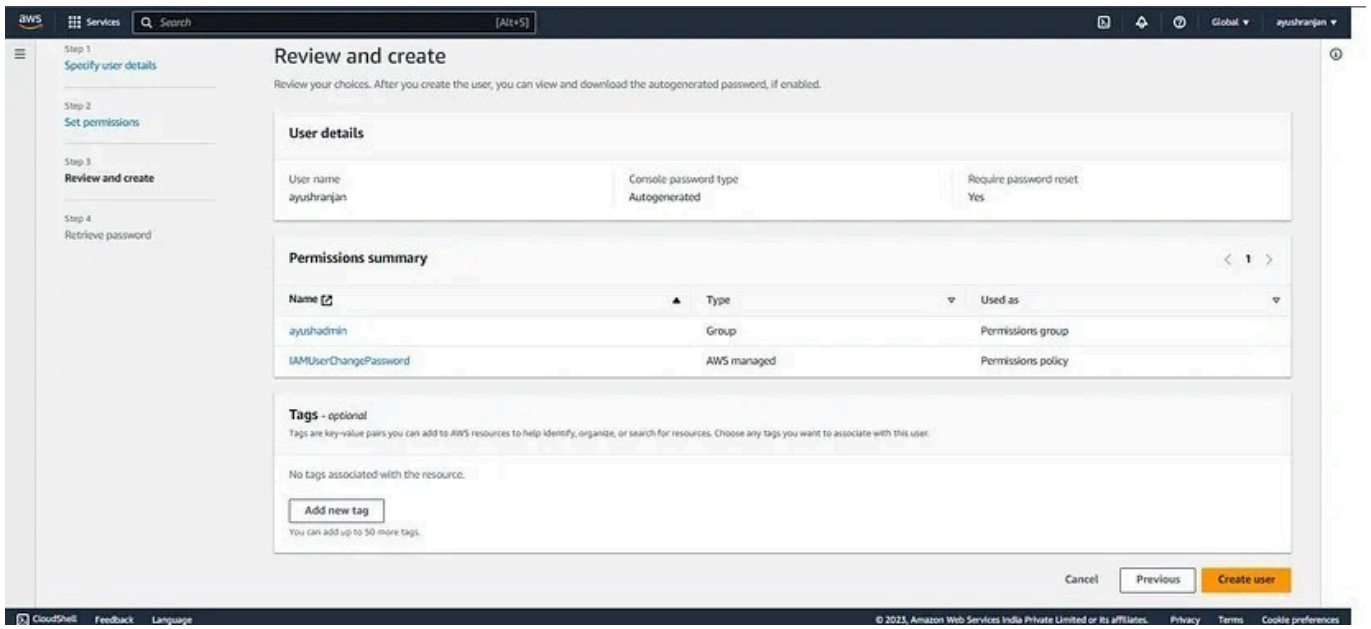
STEP 7: Please enter the desired username for the user and select the option "Provide user access to AWS Management Console". Then, choose the option "I want to create an IAM User" and select "Autogenerated Password".

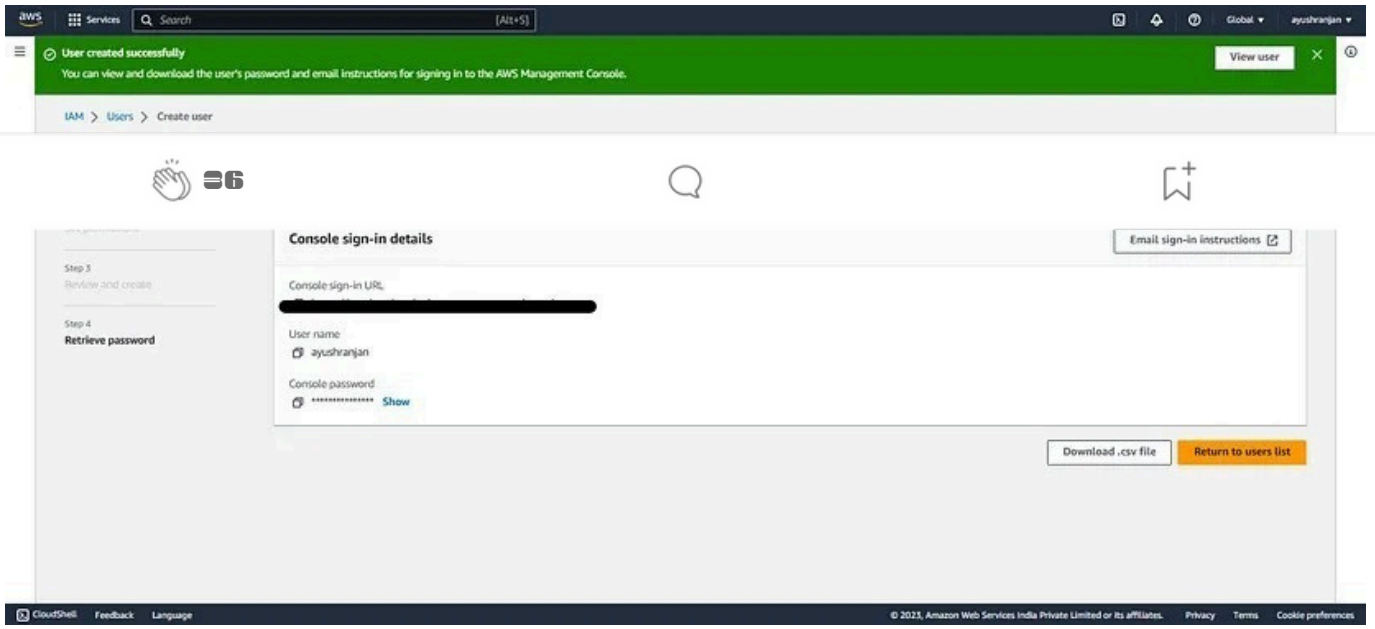


Select the Group Name under which you want to create the User.

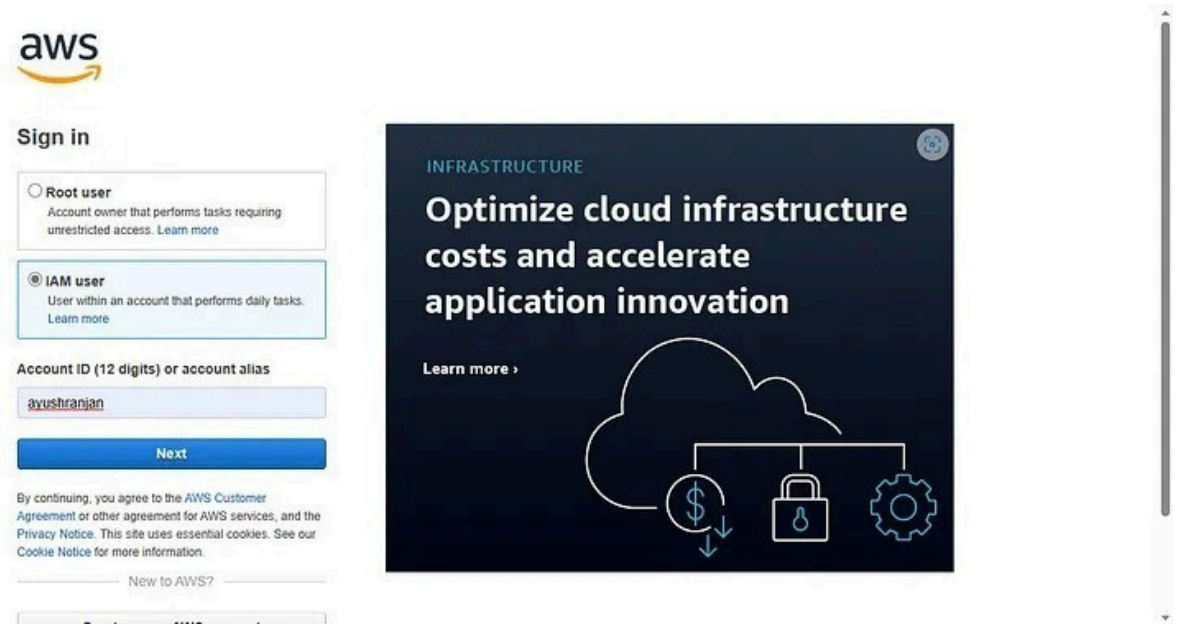


Then click on Create User.

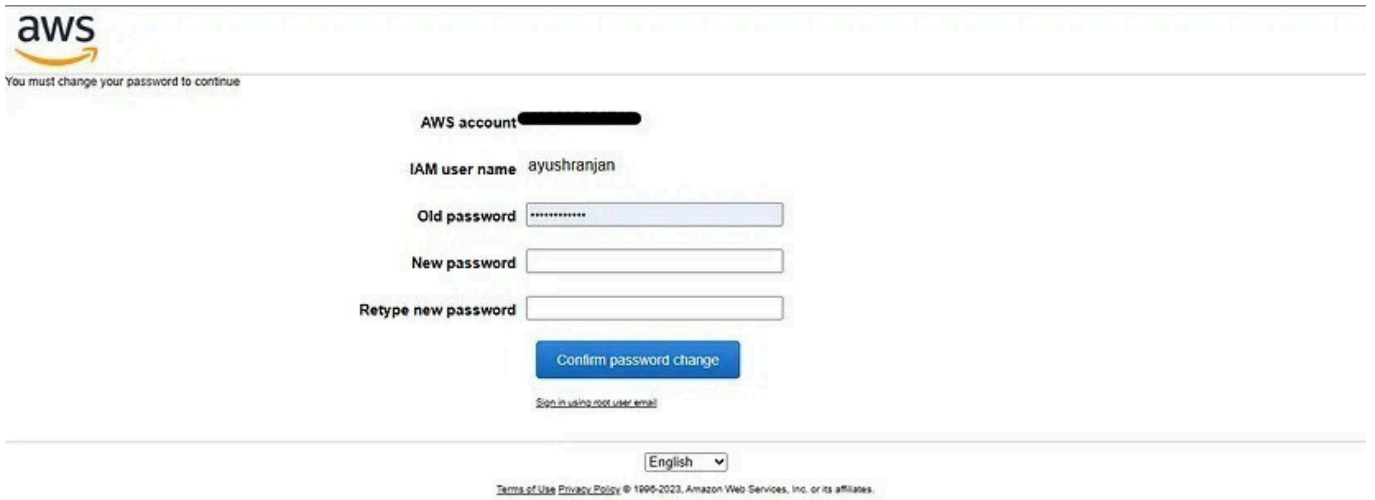




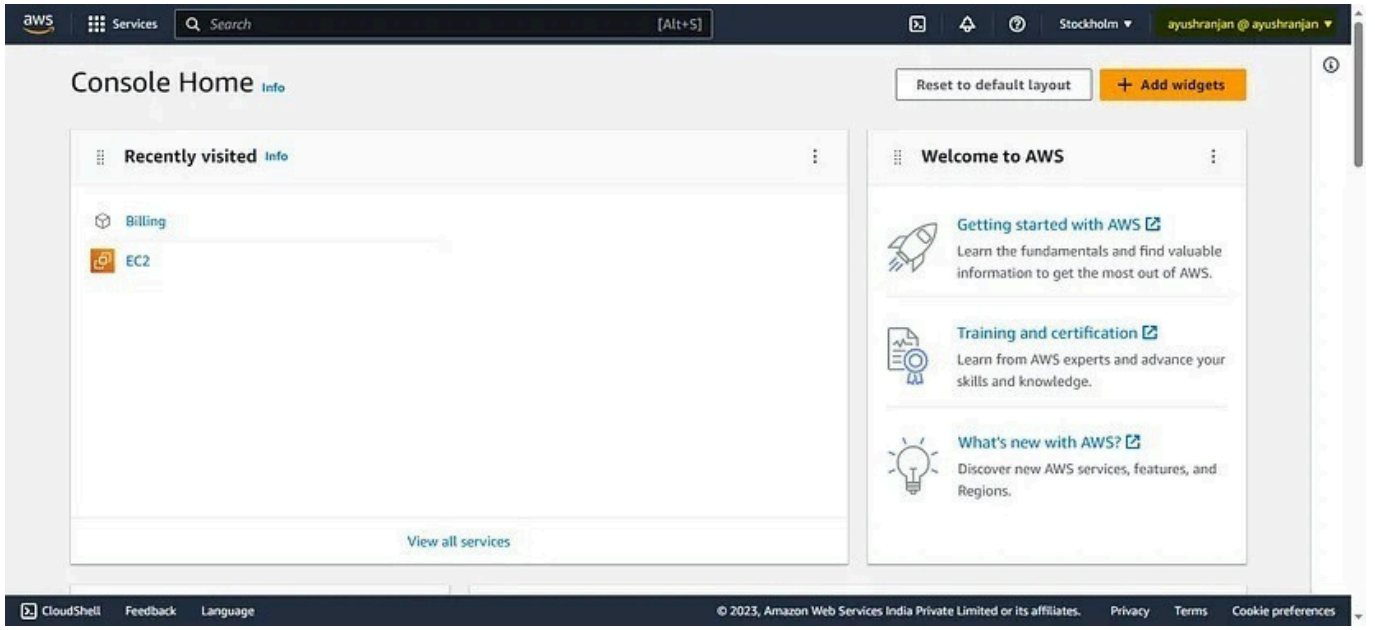
STEP 8 : Please sign out and then attempt to sign in using your IAM user credentials.



Then it will ask to create new password as we autogenerated password



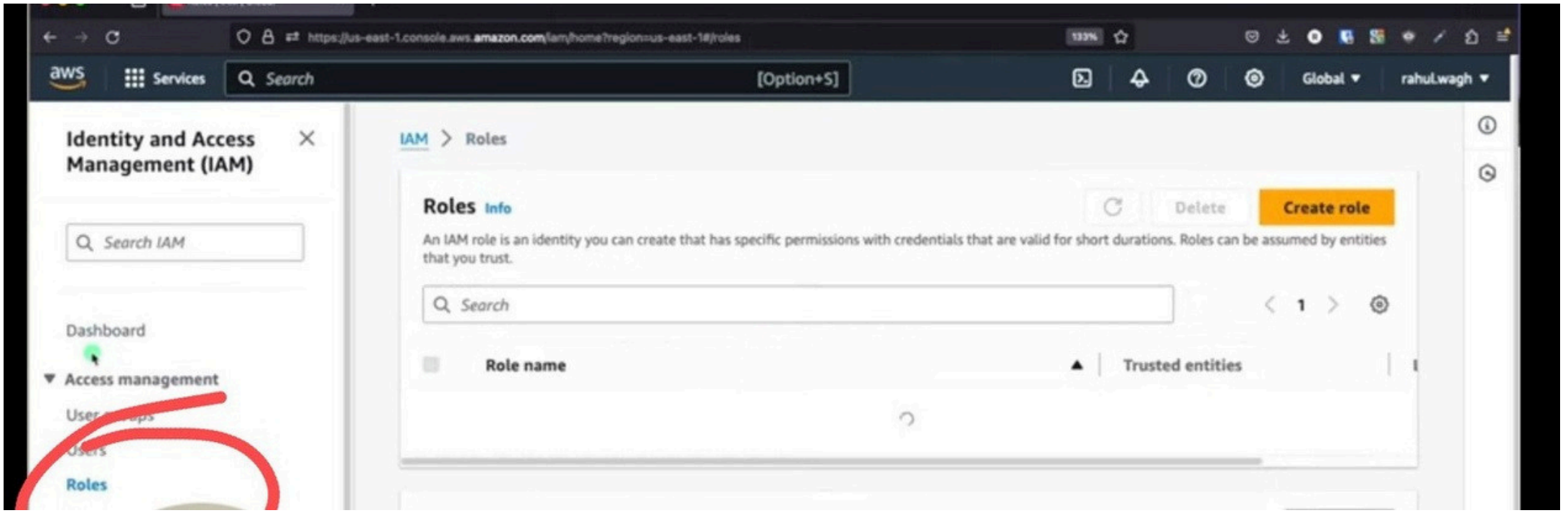
Congratulations, you have successfully created a new user group and added a new user to your AWS account. You can now access the AWS Home Console.

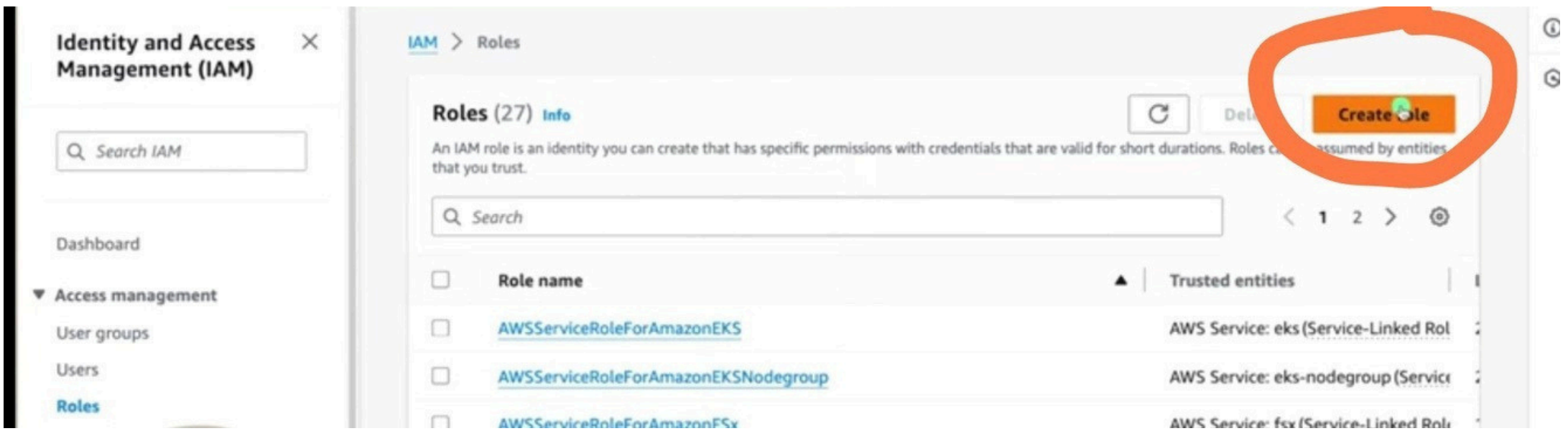


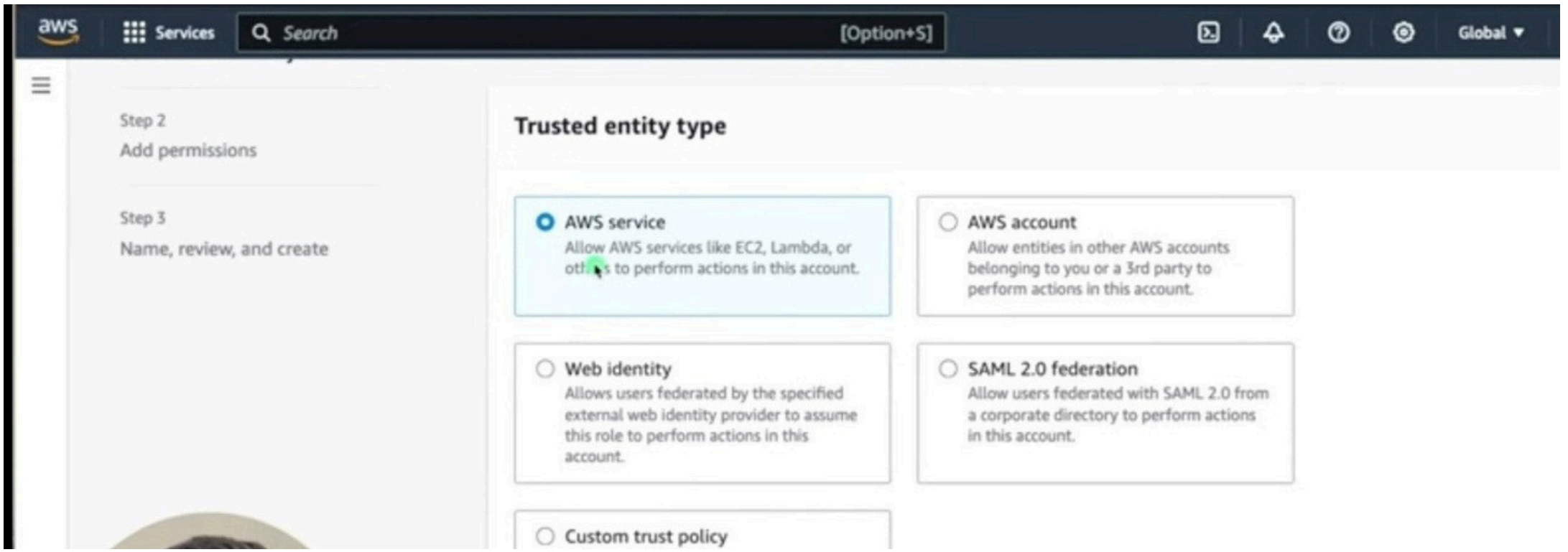
Summary

The article discusses creating IAM (Identity and Access Management) users and groups on AWS through a step-by-step guide. IAM is crucial for securing


CREATE ROLE


https://us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/roles
133%
aws
Services
Search
[Option+S]
Global
rahul.wagh
Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
Roles
IAM > Roles
Roles Info
Delete
Create role
An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.
Search
1
Role name
Trusted entities


Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
IAM > Roles
Roles (27) Info
Create role
An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles
that you trust.
Search
1 2
Role name
Trusted entities
AWSServiceRoleForAmazonEKS
AWS Service: eks (Service-Linked Rol
AWSServiceRoleForAmazonEKSNodegroup
AWS Service: eks-nodegroup (Servic


aws
Services
Search
[Option+S]
Global
Step 2
Add permissions
Step 3
Name, review, and create
Trusted entity type
AWS service
Allow AWS services like EC2, Lambda, or oth s to perform actions in this account.
AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
Web identity
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
SAML 2.0 federation
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
Custom trust policy

AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.

AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

Web identity
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

SAML 2.0 federation
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

Custom trust policy
Create a custom trust policy to enable others to perform actions in this account.

Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case

Choose a service or use case

Cancel Next

external web identity provider to assume this role to perform actions in this account.

a corporate directory to perform actions in this account.

Custom trust policy
Create a custom trust policy to enable others to perform actions in this account.

Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case

S3

Choose a use case for the specified service.

Use case

S3
Allows S3 to call AWS services on your behalf.

S3 Batch Operations
Allows S3 Batch Operations to call AWS services on your behalf.

Cancel Next

AmazonAPIGateway...	AWS managed	Provides full access to create/edit/dele...
AmazonAPIGateway...	AWS managed	Provides full access to invoke APIs in A...
AmazonAPIGateway...	AWS managed	Allows API Gateway to push logs to us...
AmazonAppFlowFul...	AWS managed	Provides full access to Amazon AppFlo...
AmazonAppFlowRe...	AWS managed	Provides read only access to Amazon A...
AmazonAppStream...	AWS managed	Provides full access to Amazon AppStr...
AmazonAppStream...	AWS managed	Amazon AppStream 2.0 access to AWS...
AmazonAppStream...	AWS managed	Provides read only access to Amazon A...
AmazonAppStream...	AWS managed	Default policy for Amazon AppStream ...
AmazonAthenaFull...	AWS managed	Provide full access to Amazon Athena ...
AmazonAugmented...	AWS managed	Provides access to perform all operati...

▶ **Set permissions boundary - *optional***

Cancel Previous Next

Role details

enter role name

Enter a meaningful name to identify this role.

allow-s3-access

Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.

Description

Add a short explanation for this role.

Allows S3 to call AWS services on your behalf.

Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), tabs, new lines, or any of the following characters: _+=,. @-/\[{}]!#$ %^&*();:"'<> `

Step 1: Select trusted entities

Edit

Trust policy

```
1 - {
2       "Version": "2012-10-17",
3 -     "Statement": [
4 -         {
```

```
12 }
```

Step 2: Add permissions

Edit

Permissions policy summary

Policy name	Type	Attached as

Step 3: Add tags

Add tags - *optional* Info

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

Add new tag

You can add up to 50 more tags.

Cancel | Previous | Create role

Role allow-s3-access created.

role is created but

it has no policy

View role

Roles (28) Info

Delete | Create role

that you trust.

Search

1 2

Role name	Trusted entities
AWSServiceRoleForAmazonEKSNodegroup	AWS Service: eks-nodegroup (Servic
AWSServiceRoleForAmazonFSx	AWS Service: fsx (Service-Linked Rol
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable	AWS Service: dynamodb.application
AWSServiceRoleForAutoScaling	AWS Service: autoscaling (Service-Li
AWSServiceRoleForCloudTrail	AWS Service: cloudtrail (Service-Link
AWSServiceRoleForEc2InstanceConnect	AWS Service: ec2-instance-connect (:
AWSServiceRoleForEC2Spot	AWS Service: spot (Service-Linked Rc
AWSServiceRoleForECS	AWS Service: ecs (Service-Linked Rol

Role allow-s3-access created.

View role

Roles (28) Info

Delete

Create role

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

allow

1 match

1

click this role

Role name	Trusted entities
allow-s3-access	AWS Service: s3

Roles Anywhere Info

Manage

Authenticate your non AWS workloads and securely provide access to AWS services.

Access AWS from your non AWS workloads

Operate your non AWS workloads using

X.509 Standard

Use your own existing PKI infrastructure or use AWS Certificate

Temporary credentials

Use temporary credentials with ease and benefit from the enhanced security

April 13, 2024, 22:32 (UTC+02:00)

arn:aws:iam::242396018804:role/allow-s3-access

Last activity

-

Maximum session duration

1 hour

Permissions | Trust relationships | Tags | Access Advisor | Revoke sessions

Permissions policies (0) Info

You can attach up to 10 managed policies.

Simulate | Remove | Add permissions

Filter by Type

Search

All types

1

Policy name	Type	Attached entities

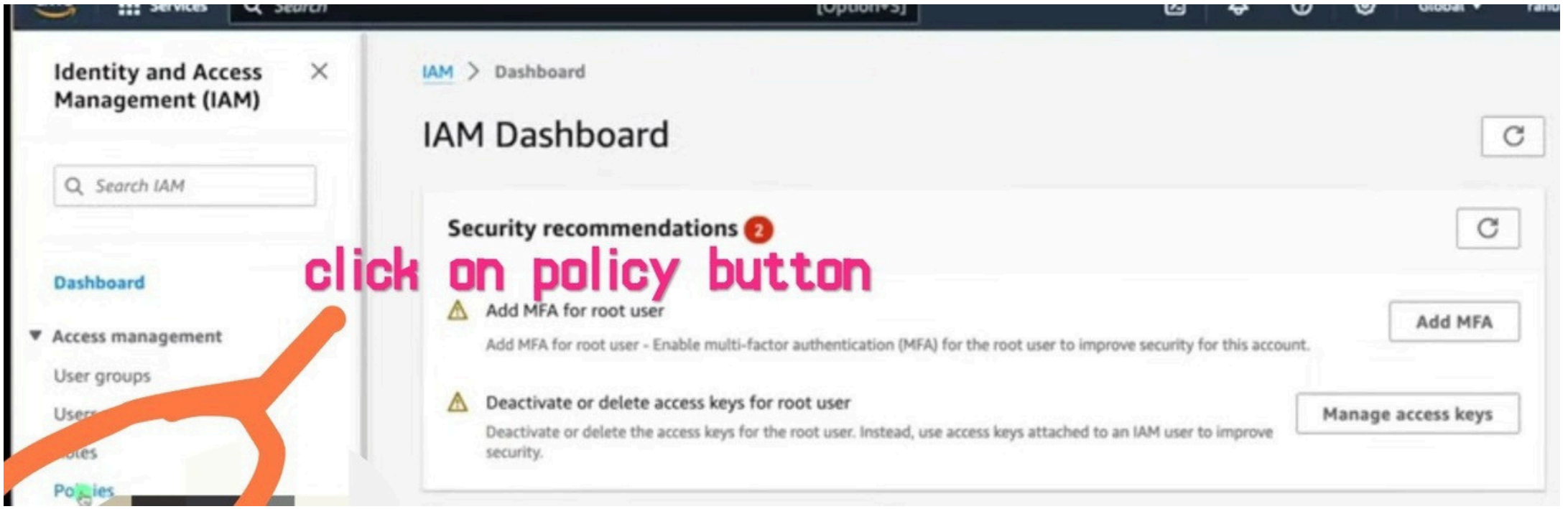
No resources to display

you can see no policy is created

▶ Permissions boundary (not set)

now we need to

create policy


Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
IAM > Dashboard
IAM Dashboard
Security recommendations 2
click on policy button
Add MFA for root user
Add MFA for root user - Enable multi-factor authentication (MFA) for the root user to improve security for this account.
Add MFA
Deactivate or delete access keys for root user
Deactivate or delete the access keys for the root user. Instead, use access keys attached to an IAM user to improve security.
Manage access keys

IAM > Policies

Policies (1191) Info

A policy is an object in AWS that defines permissions.

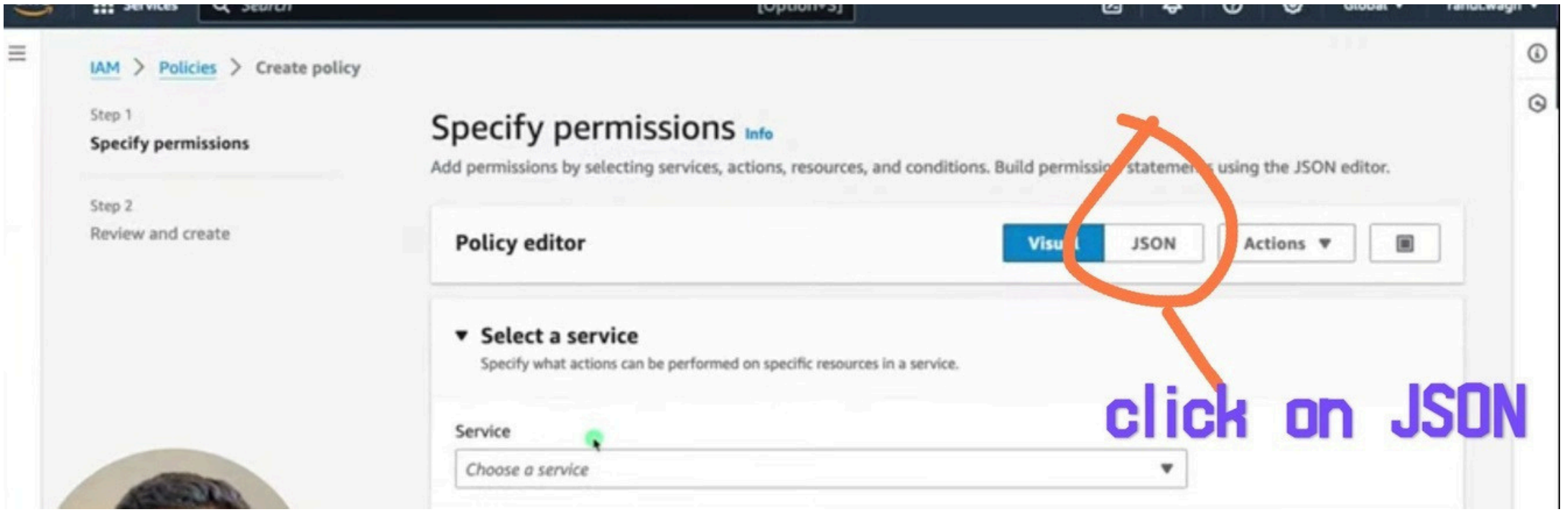
Actions | Delete | Create policy

Filter by Type

Search | All types

1 2 3 4 5 6 7 ... 60

Policy name	Type	Used as	Description
AccessAnalyzerSer...	AWS managed	None	Allow Access Analyzer to analyze r
AdministratorAccess	AWS managed - jo...	Permissions policy ...	Provides full access to AWS service
AdministratorAcce...	AWS managed	None	Grants account administrative perr
AdministratorAcce...	AWS managed	None	Grants account administrative perr
AlexaForBusinessD...	AWS managed	None	Provide device setup access to Alex
AlexaForBusinessF...	AWS managed	None	Grants full access to AlexaForBusin
AlexaForBusinessG...	AWS managed	None	Provide gateway execution access t


IAM > Policies > Create policy
Step 1
Specify permissions
Step 2
Review and create
Specify permissions Info
Add permissions by selecting services, actions, resources, and conditions. Build permissio statemen using the JSON editor.
Policy editor
Visu
JSON
Actions
Select a service
Specify what actions can be performed on specific resources in a service.
Service
Choose a service
click on JSON

e policy

Specify permissions Info

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

Policy editor

Visual | JSON | Actions

```
1 {
2     "Version": "2012-10-17",
3     "Statement": [
4         {
5             "Sid": "AllowS3Access",
6             "Effect": "Allow",
7             "Action": "s3:*",
8             "Resource": "*"
9         }
10    ]
11 }
```

Edit statement Remove

AllowS3Access

Add actions

Choose a service

Filter services

Included

S3

Available

AMP

API Gateway

API Gateway V2

step 1 configure JSON

```
10      ]
11  }
```

+ Add new statement

Included

S3

Available

AMP

API Gateway

API Gateway V2

ASC

Access Analyzer

Account

Activate

Alexa for Business

Add a resource Add

Add a condition (optional) Add

JSON Ln 9, Col 3

6034 of 6144 characters remaining

Security: 0 Errors: 0 Warnings: 0 Suggestions: 0

Cancel Next

e policy

Review and create Info

Review the permissions, specify details, and tags.

put name of policy

Policy details

Policy name

Enter a meaningful name to identify this policy.

custom-s3-allow

Maximum 128 characters. Use alphanumeric and '+=,.@-_' characters.

Description - *optional*

Add a short explanation for this policy.

Maximum 1,000 characters. Use alphanumeric and '+=,.@-_' characters.

Permissions defined in this policy Info

Edit

Permissions defined in this policy document specify which actions are allowed or denied. To define permissions for an IAM identity (user, user group, or role), attach a policy to it

Permissions defined in this policy Info

Edit

Permissions defined in this policy document specify which actions are allowed or denied. To define permissions for an IAM identity (user, user group, or role), attach a policy to it

Search

Allow (1 of 408 services)

Show remaining 407 services

Service	Access level	Resource	Request cc
S3	Full access	All resources	None

Add tags - *optional* Info

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

Add new tag

You can add up to 50 more tags.

Cancel

Previous

Create policy

Policy custom-s3-allow created.

View policy

policy is created

IAM > Policies

Policies (1192) Info

Actions | Delete | Create policy

A policy is an object in AWS that defines permissions.

Filter by Type

Search | All types

1 2 3 4 5 6 7 ... 60

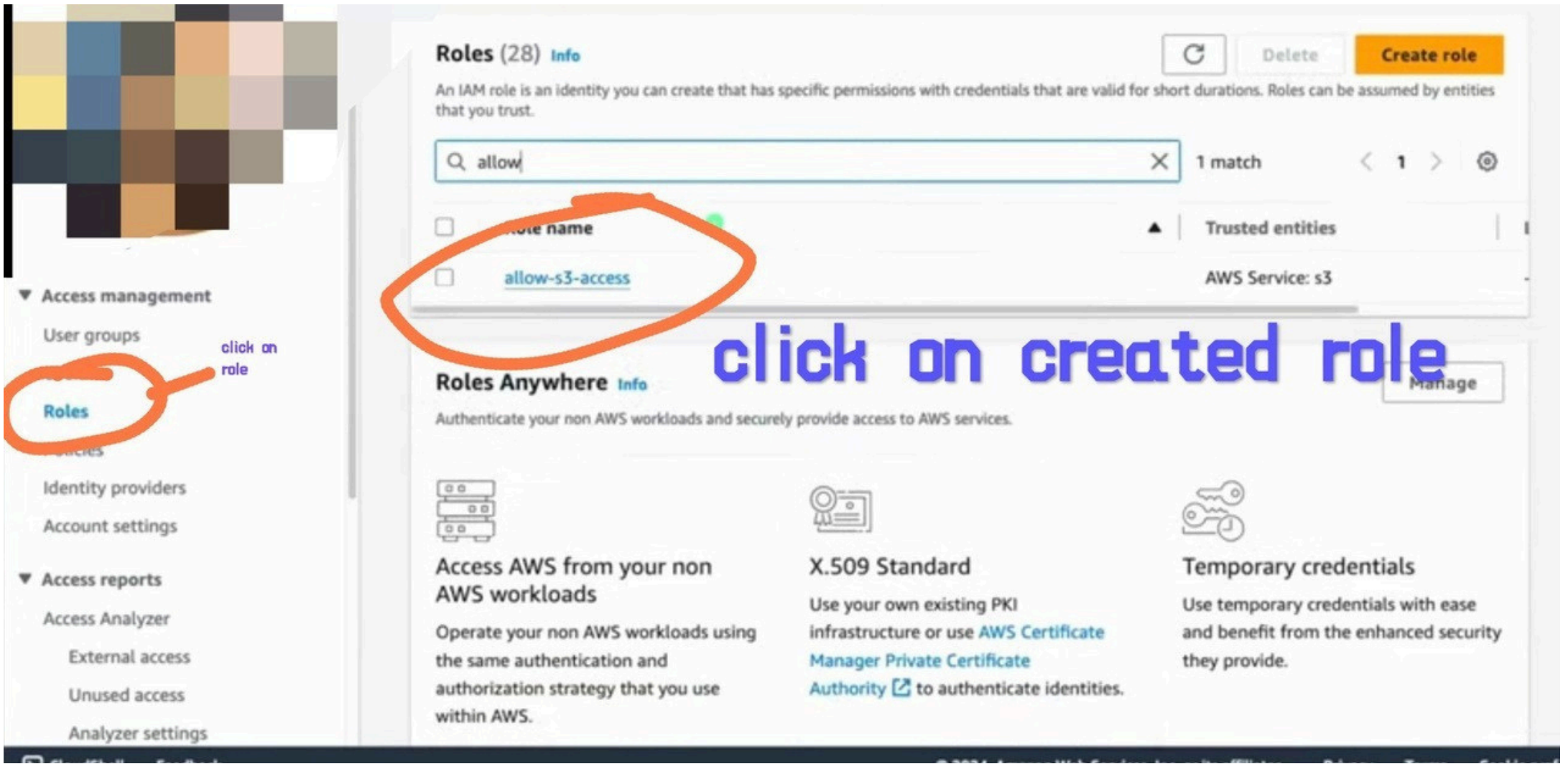
Policy name	Type	Used as	Description
AccessAnalyzerSer...	AWS managed	None	Allow Access Analyzer to analyze r
AdministratorAccess	AWS managed - jo...	Permissions policy ...	Provides full access to AWS service
AdministratorAcce...	AWS managed	None	Grants account administrative perr
AlexaForBusinessD...	AWS managed	None	Provide device setup access to Alex
AlexaForBusinessF...	AWS managed	None	Grants full access to AlexaForBusin
AlexaForBusinessG...	AWS managed	None	Provide gateway execution access t

Till now we created a role
and created policy

.....

Now

I am attaching policy to the role


Roles (28) Info
Delete
Create role
An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.
allow
1 match
1
Trusted entities
allow-s3-access
AWS Service: s3
click on created role
Access management
User groups
click on role
Roles
Identity providers
Account settings
Access reports
Access Analyzer
External access
Unused access
Analyzer settings
Roles Anywhere Info
Manage
Authenticate your non AWS workloads and securely provide access to AWS services.
Access AWS from your non AWS workloads
Operate your non AWS workloads using the same authentication and authorization strategy that you use within AWS.
X.509 Standard
Use your own existing PKI infrastructure or use AWS Certificate Manager Private Certificate Authority to authenticate identities.
Temporary credentials
Use temporary credentials with ease and benefit from the enhanced security they provide.

Last activity

-

Maximum session duration

1 hour

Permissions | Trust relationships | Tags | Access Advisor | Revoke sessions

Permissions policies (0) Info

You can attach up to 10 managed policies.

Simulate | Remove | Add permissions

Attach policies

Create inline policy

click Attach policy

Filter by Type

Search

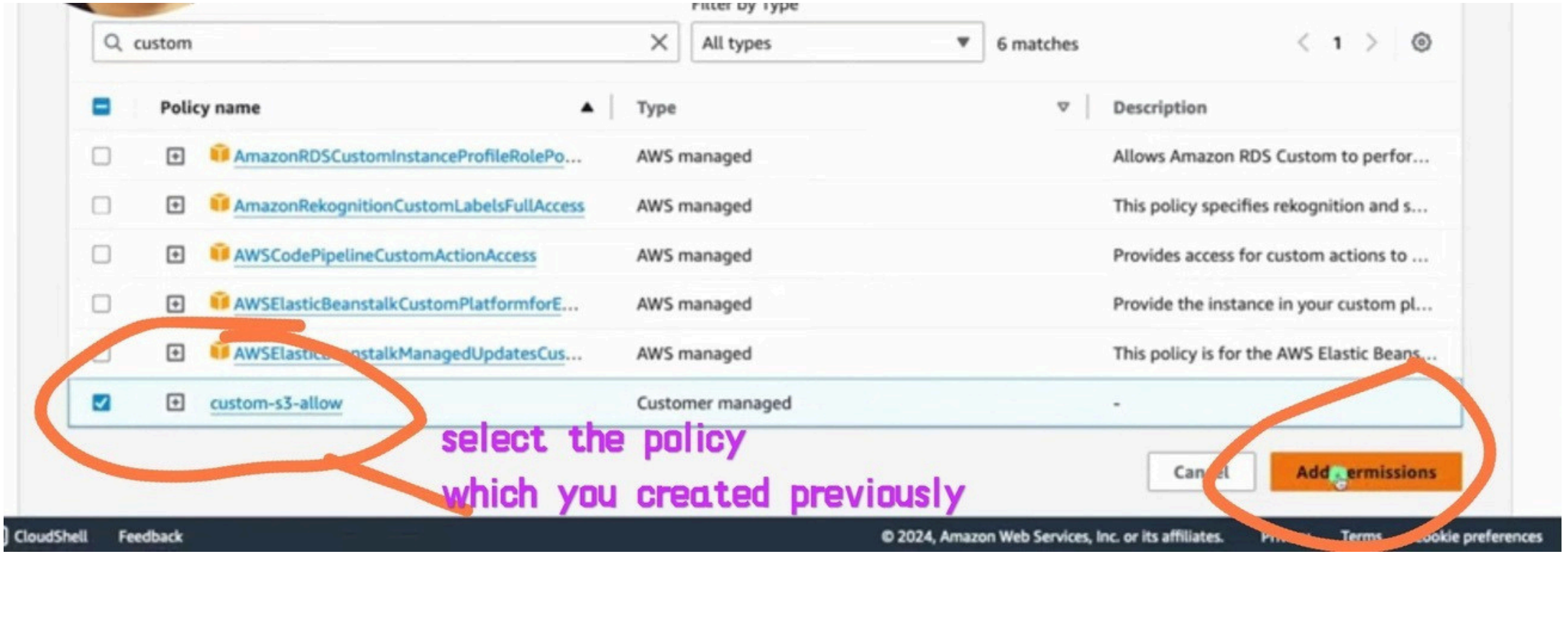
1

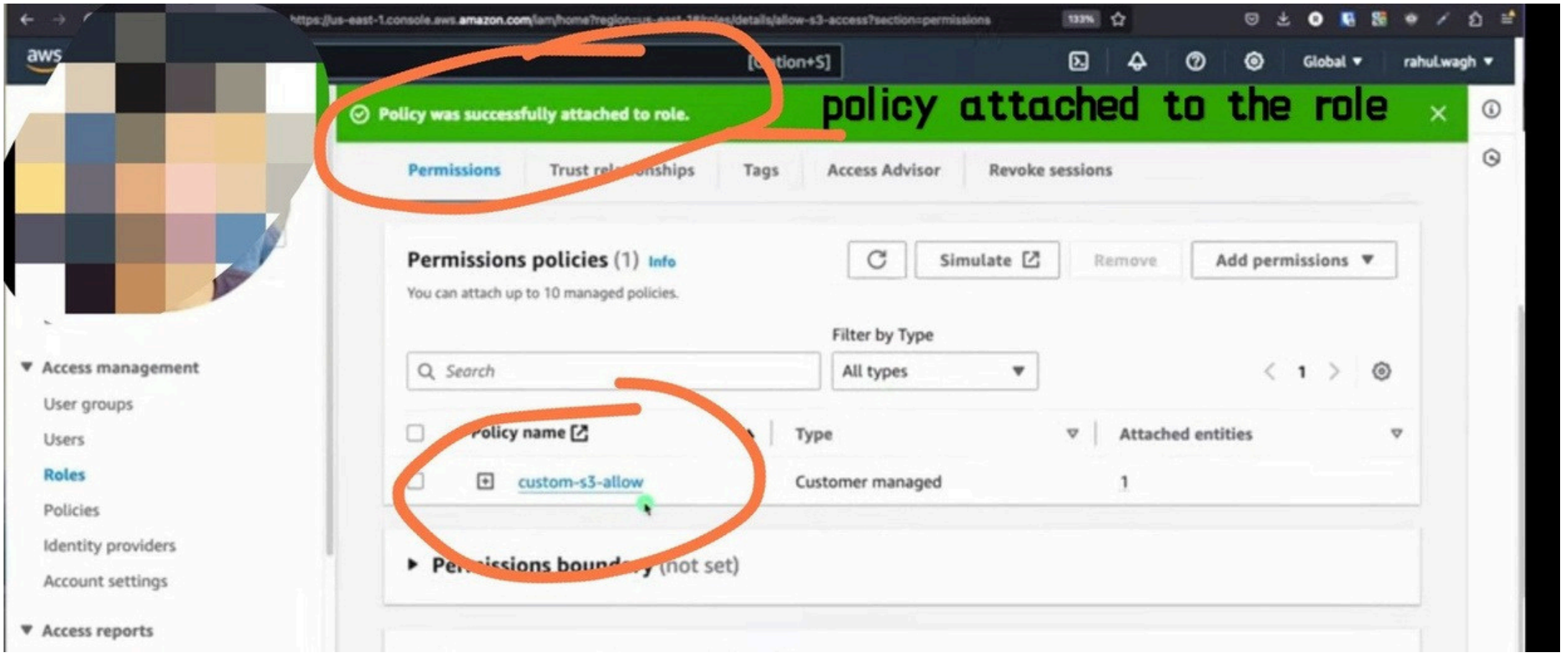
Policy name	Type	Attached entities

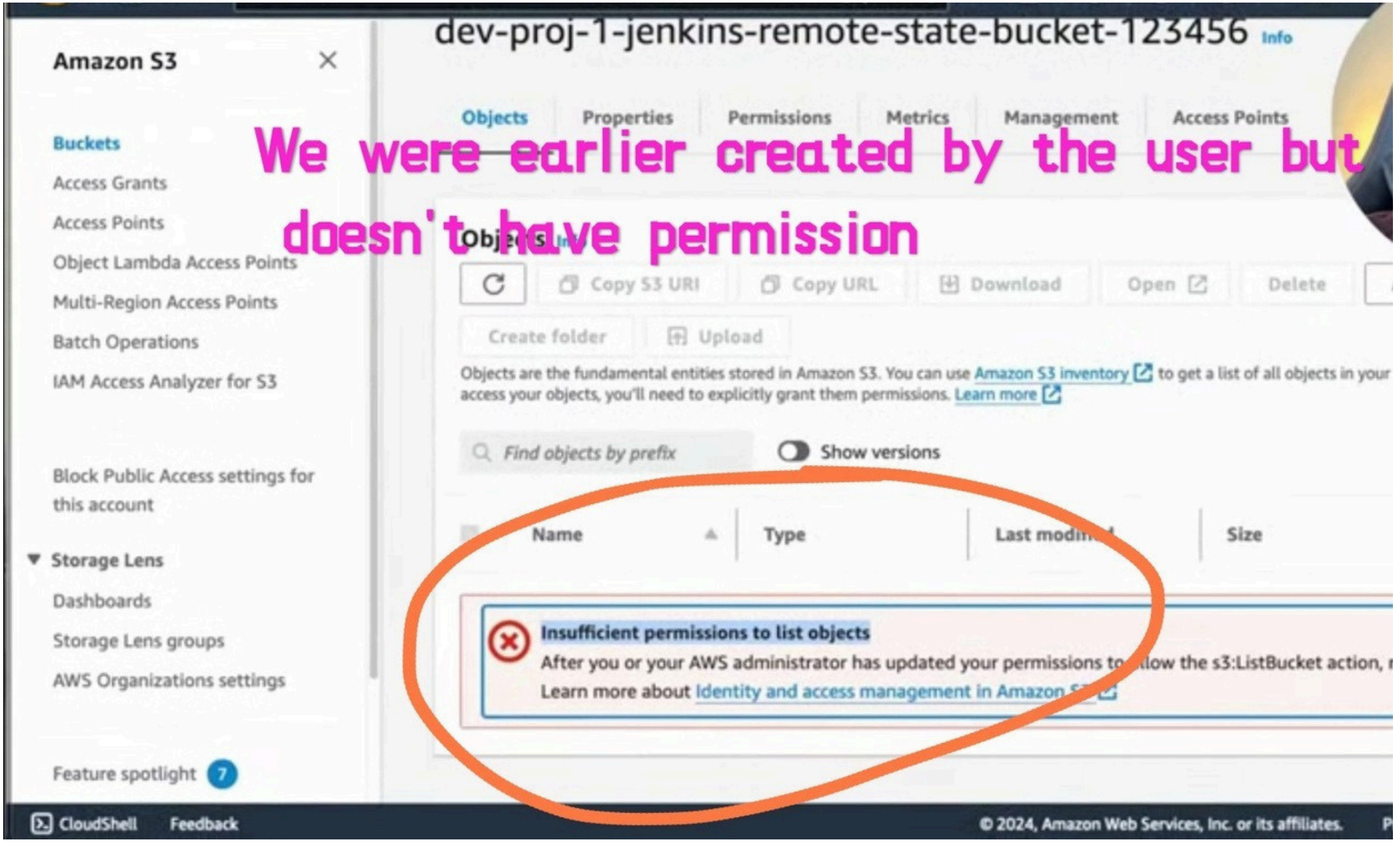
No resources to display

▶ **Permissions boundary** (not set)

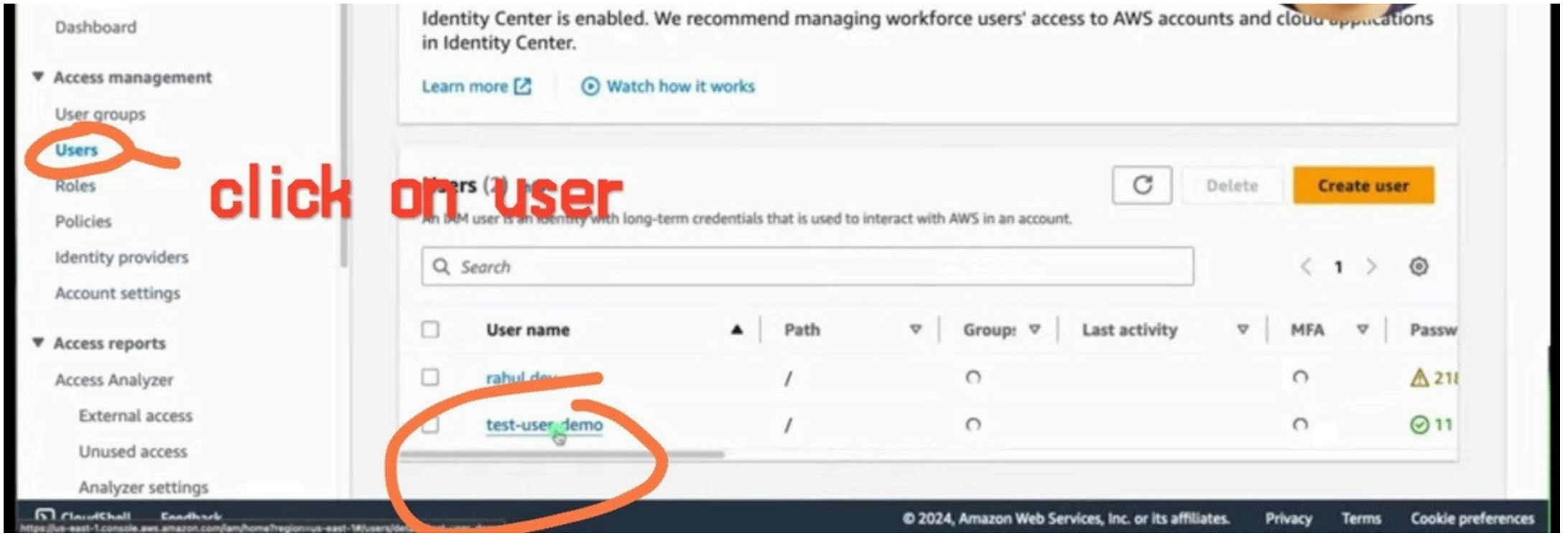
▼ **Generate policy based on CloudTrail events**

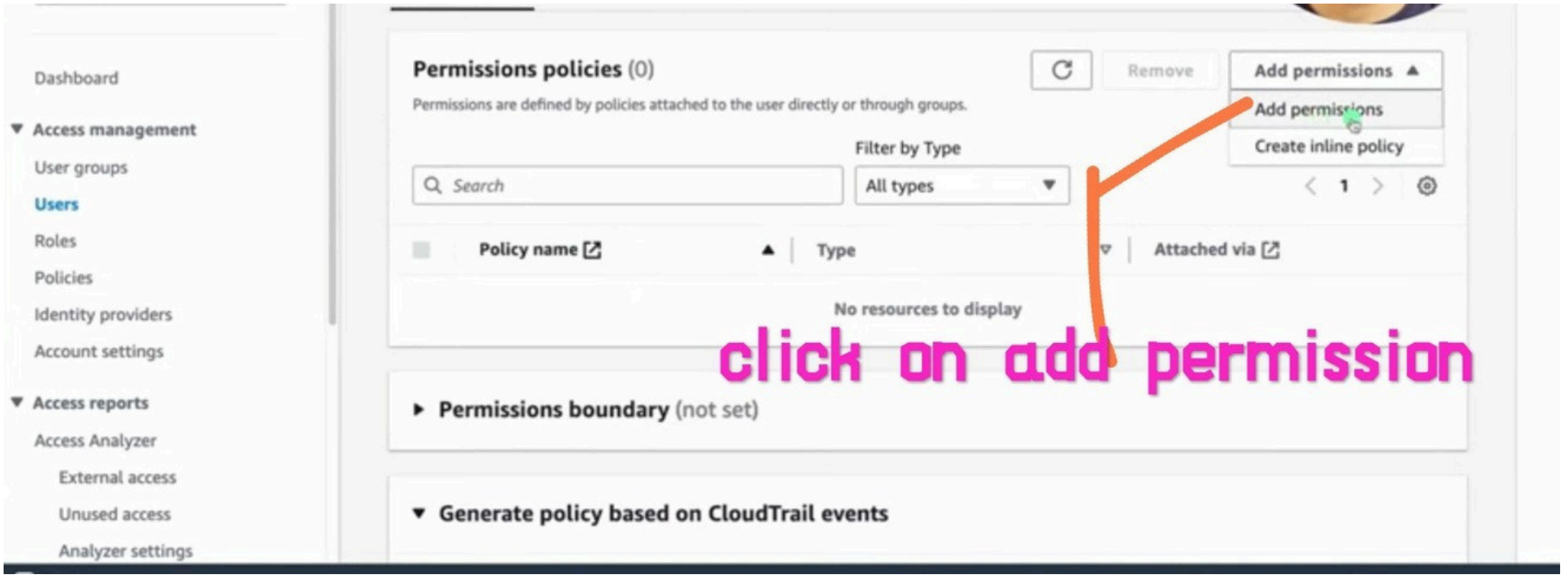

custom
All types
6 matches
1
Policy name
Type
Description
AmazonRDSCustomInstanceProfileRolePo...
AWS managed
Allows Amazon RDS Custom to perfor...
AmazonRekognitionCustomLabelsFullAccess
AWS managed
This policy specifies rekognition and s...
AWSCodePipelineCustomActionAccess
AWS managed
Provides access for custom actions to ...
AWSElasticBeanstalkCustomPlatformforE...
AWS managed
Provide the instance in your custom pl...
AWS managed
This policy is for the AWS Elastic Beans...
custom-s3-allow
Customer managed
-
select the policy
which you created previously
Add permissions
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Terms


Policy was successfully attached to role.
policy attached to the role
Permissions
Tags
Access Advisor
Revoke sessions
Permissions policies (1) Info
You can attach up to 10 managed policies.
Simulate
Remove
Add permissions
Filter by Type
Search
All types
Policy name
Type
Attached entities
custom-s3-allow
Customer managed
1
(not set)
Global
rahul.wagh
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports

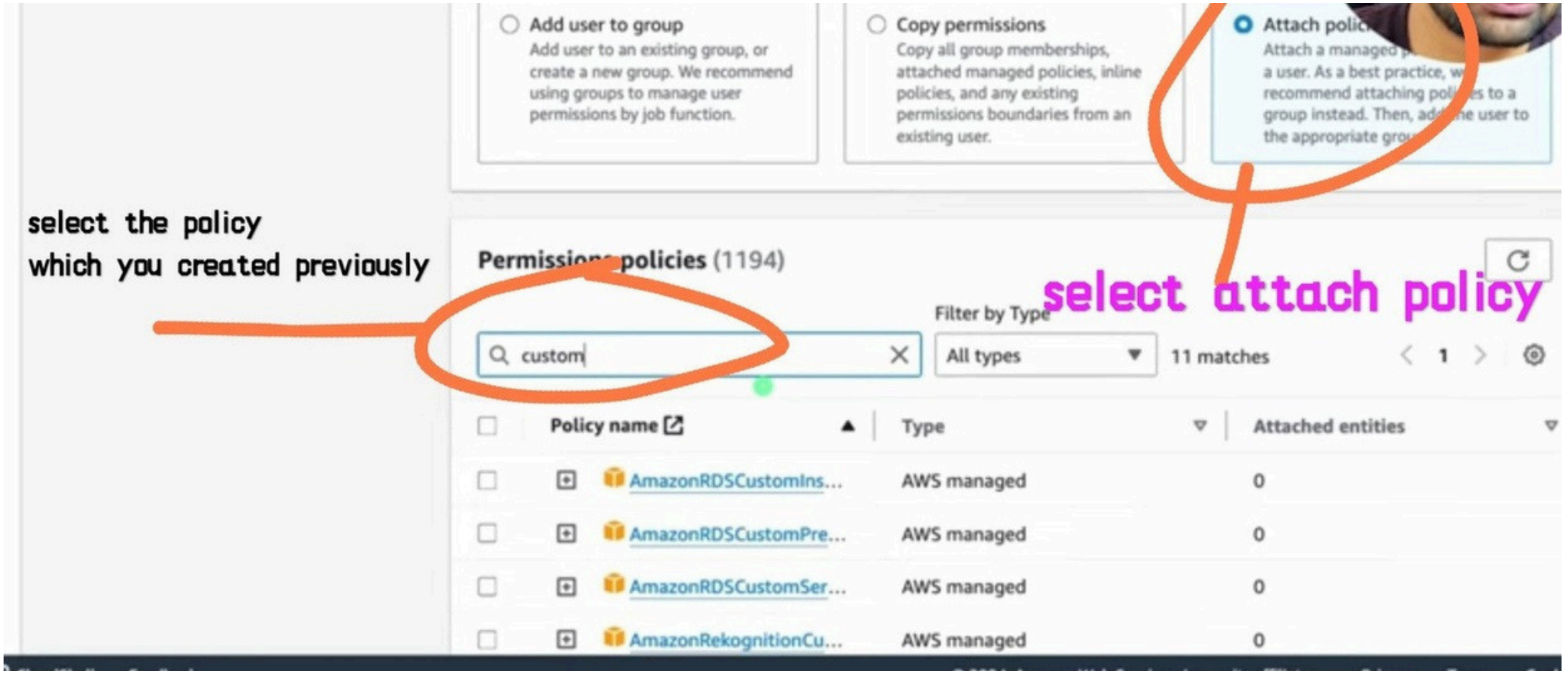

dev-proj-1-jenkins-remote-state-bucket-123456 Info
Amazon S3
Objects
Properties
Permissions
Metrics
Management
Access Points
Buckets
Access Grants
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3
We were earlier created by the user but
doesn't have permission
Copy S3 URI
Copy URL
Download
Open
Delete
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
access your objects, you'll need to explicitly grant them permissions. Learn more
Find objects by prefix
Show versions
Name
Type
Size
Block Public Access settings for
this account
Storage Lens
Dashboards
Storage Lens groups
AWS Organizations settings
Insufficient permissions to list objects
After you or your AWS administrator has updated your permissions to
low the s3:ListBucket action,
Learn more about Identity and access management in Amazon
Feature spotlight
7
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.


now basically we want to
attach policy to the user


Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access Analyzer
External access
Unused access
Analyzer settings
Identity Center is enabled. We recommend managing workforce users' access to AWS accounts and cloud applications in Identity Center.
Learn more
Watch how it works
click on user
Delete
Create user
Search
1
User name
Path
Group:
Last activity
MFA
Passw
rahul dev
/
test-user-demo
/
11
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access Analyzer
External access
Unused access
Analyzer settings
Permissions policies (0)
Permissions are defined by policies attached to the user directly or through groups.
Remove
Add permissions
Add permissions
Create inline policy
Filter by Type
Search
All types
1
Policy name
Type
Attached via
No resources to display
click on add permission
Permissions boundary (not set)
Generate policy based on CloudTrail events

select the policy
which you created previously



☐	⊞	AmazonRDSCustomPre...	AWS managed	0
☐	⊞	AmazonRDSCustomSer...	AWS managed	0
☐	⊞	AmazonRekognitionCu...	AWS managed	0
☐	⊞	AWSApplicationAutoSc...	AWS managed	0
☐	⊞	AWSCodePipelineCust...	AWS managed	0
☐	⊞	AWSElasticBeanstalkCu...	AWS managed	0
☐	⊞	AWSElasticBeanstalkM...	AWS managed	0
☐	⊞	AWSKeyManagementS...	AWS managed	0
☑	⊞	custom-s3-allow	Customer managed	1
☐	⊞	CustomerProfilesServic...	AWS managed	0

Cancel Next

Review

User details

User name
test-user-demo

Permissions summary (1)

1

Name	Type	Used as
custom-s3-allow	Customer managed	Permissions policy

Cancel | Previou | Add permissio

aws | Services | Search [Option+S]

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

1 policy added

now successfully attached the policy to user

April 13, 2024, 22:24 (UTC+02:00) Today

Permissions | Groups | Tags | Security credentials | Access Advisor

Permissions policies (1)

Remove

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

Search

Filter by Type: All types

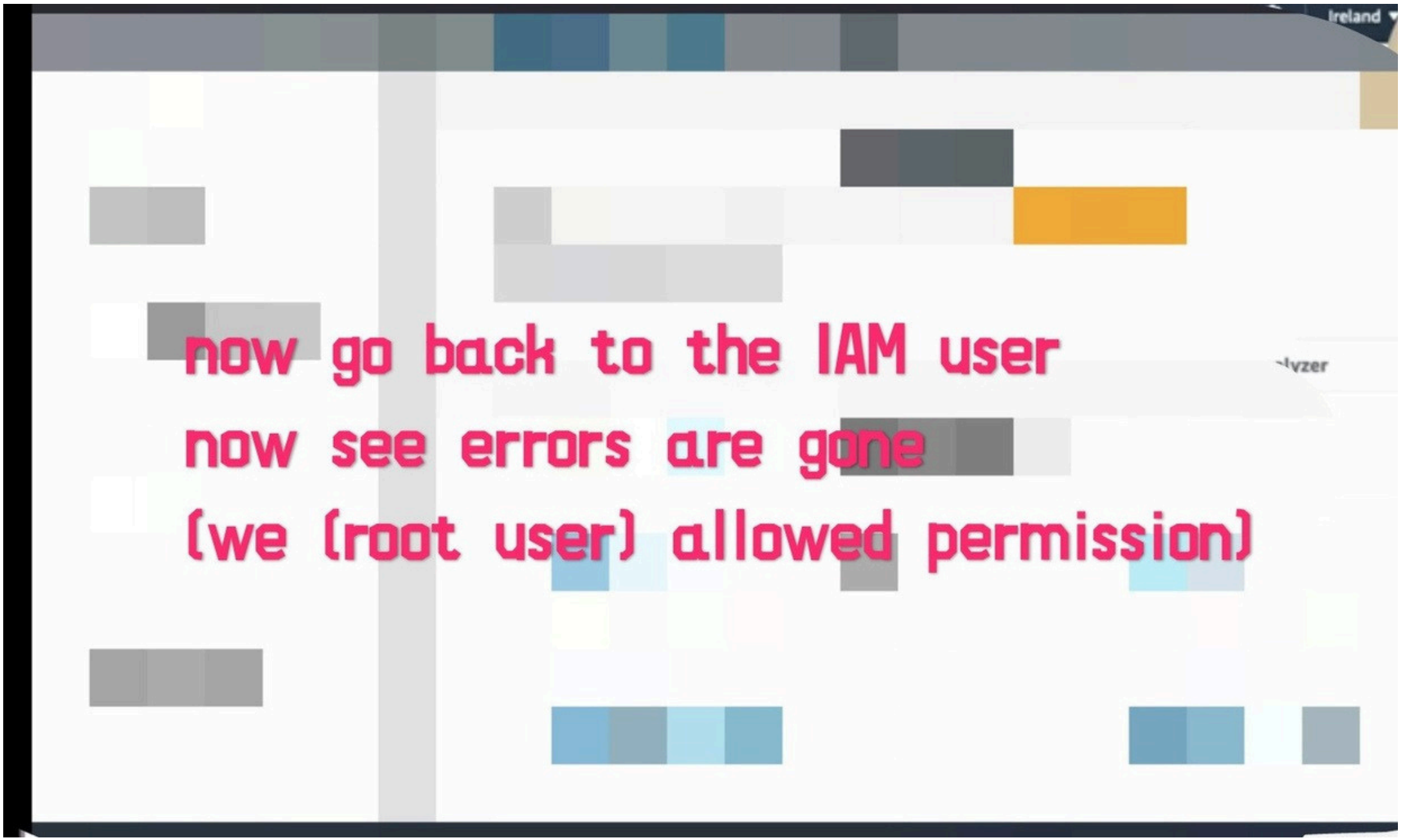
Policy name	Type	Attached
custom-s3-allow	Customer managed	Directly

custom-s3-allow

Copy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowS3Access",
            "Effect": "Allow",
```

CloudShell Feedback


Ireland
now go back to the IAM user
now see errors are gone
(we (root user) allowed permission)

successfully attached policy to IAM user

Amazon S3

- 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 7

General purpose buckets (4) Info All AWS Regions

Copy ARN | Empty | Delete | Create bucket

Buckets are containers for data stored in S3.

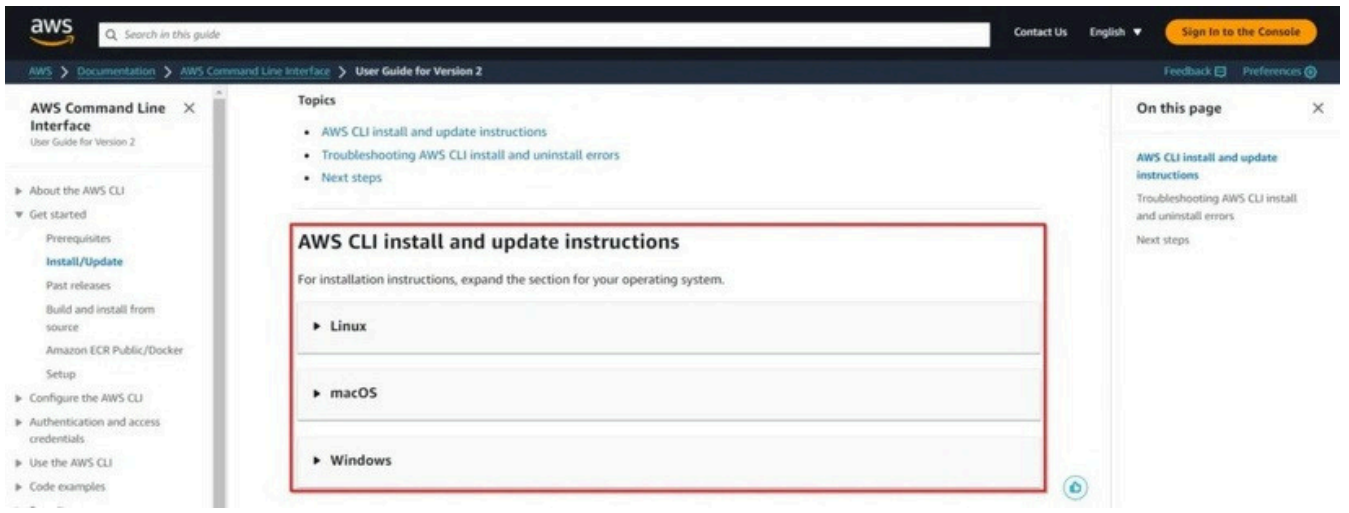
Find buckets by name

Name	AWS Region	IAM Access Analyzer
dev-proj-1-jenkins-remote-state-bucket-123456	Europe (Ireland) eu-west-1	View analyzer for eu-west-1
dev-proj-1-remote-state-bucket-123456	Europe (Frankfurt) eu-central-1	View analyzer for eu-central-1
dev-proj-vpc-2-private-link-eu-central-1a	Europe (Frankfurt) eu-central-1	View analyzer for eu-central-1
dev-proj-vpc-endpoint-eu-west-1a	Europe (Ireland) eu-west-1	View analyzer for eu-west-1

AWS CLI configuration

Now le t's install and configure AWS CLI as shown in the following steps mentioned below:

Step 1. Download and install AWS CLI: Before going to the AWS dashboard, firstly, we have to download the CLI installer (https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html) on the local machine (Windows, macOS, Linux).



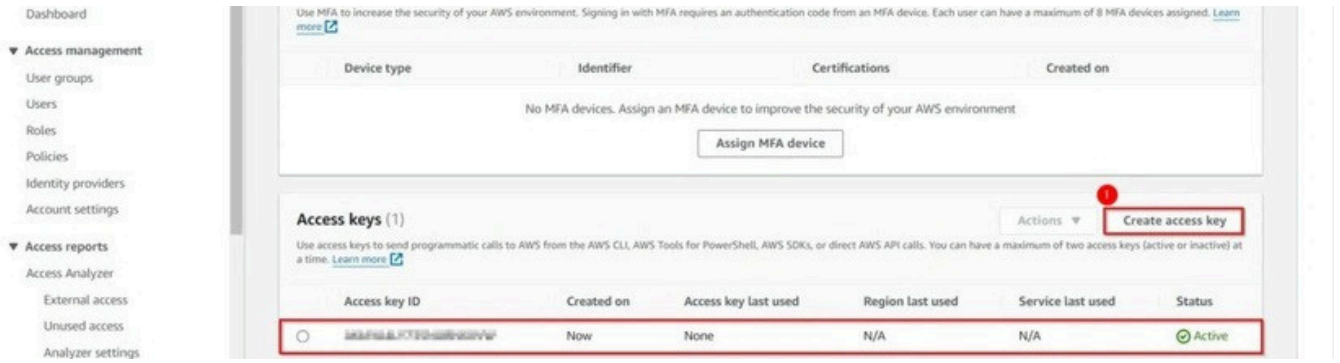
After that, RUN the downloaded MSI installer.

Step 2. Confirm the installation: To confirm the installation process, we must write the command version prompt's command. If the version is displayed, it indicates that CLI is installed, fortunately.

```
aws --
```

Step 3. Configure AWS CLI: After CLI installation, we have to download the AWS Console access key.

For that, go to My Security Credentials in AWS Console by clicking on UserName on the To right Corner. Now Scroll down and you will find Access keys Click on Create New Access Key, then download the CSV file to the local machine.



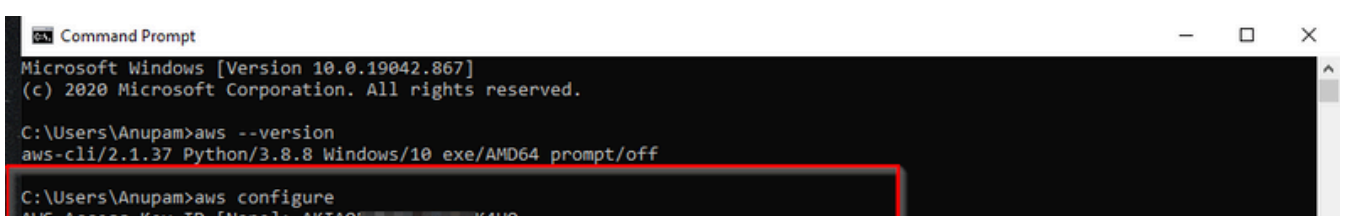
Now we can configure AWS CLI using the command: `aws configure` and fill in details like AWS keys, region, and output format.

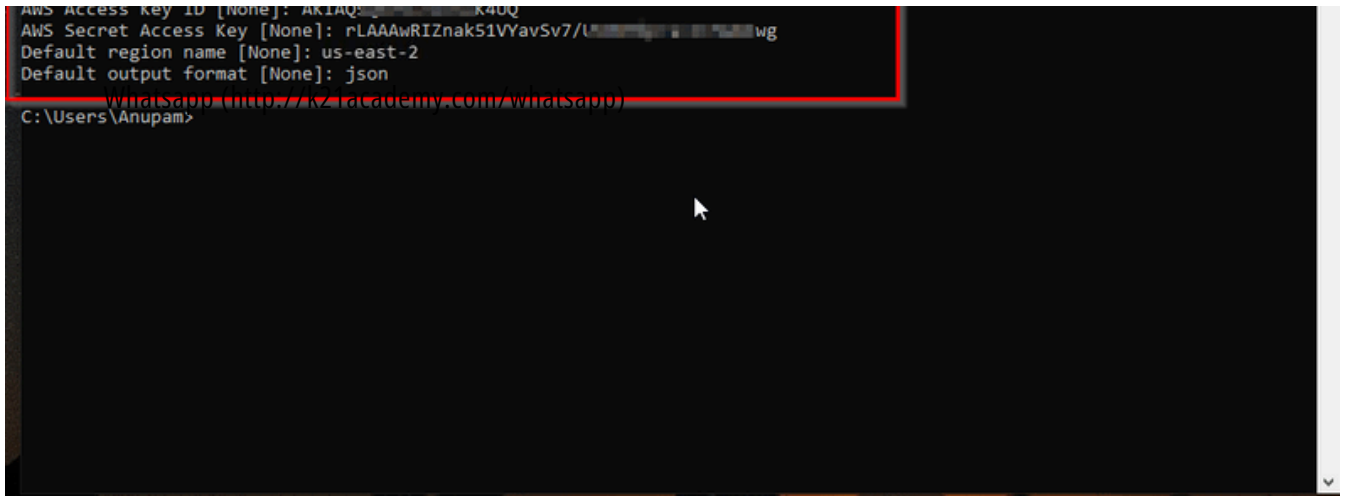
1. AWS Access Key ID [None]: **********
2. AWS Secret access key [None]: **********
3. Default Region name [None]: us-east-2

(However, you can choose any region closest to your location)

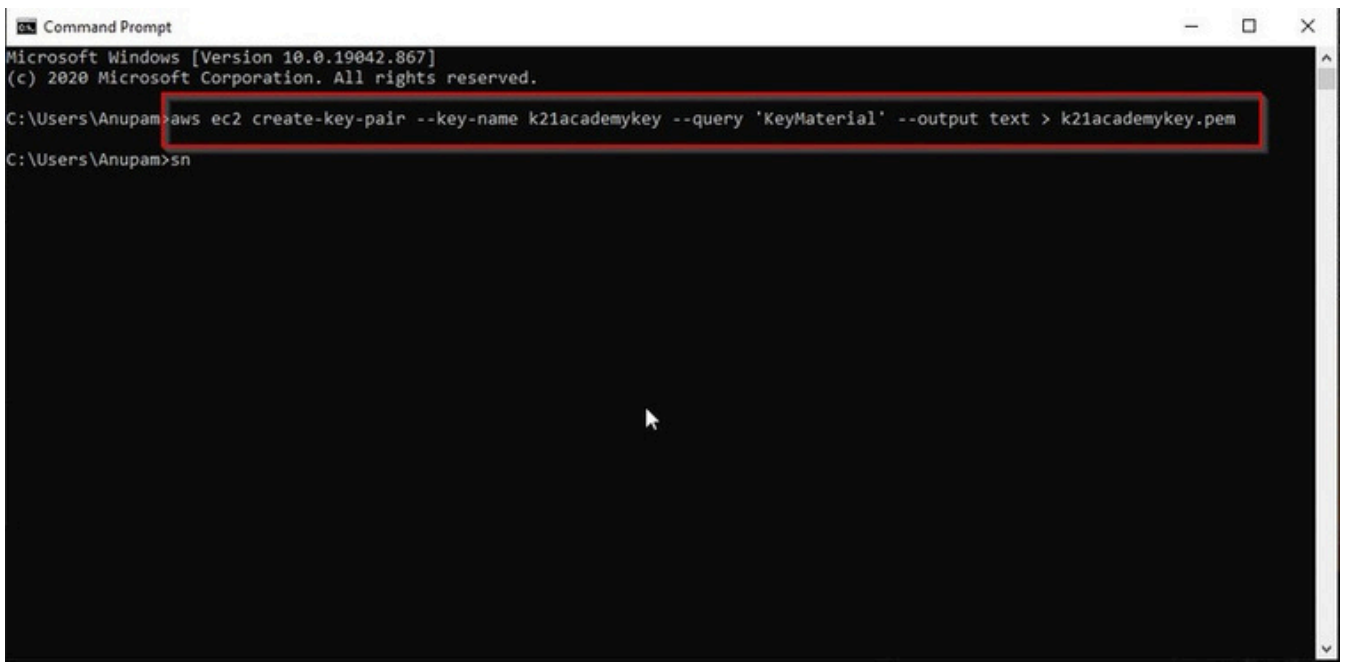
4. Default output format [None]: JSON

(When we run the command, it's going to split out some output, and there are several different options available, like how you want those outputs printed out. You can get it in JSON, YAML, or text format.)

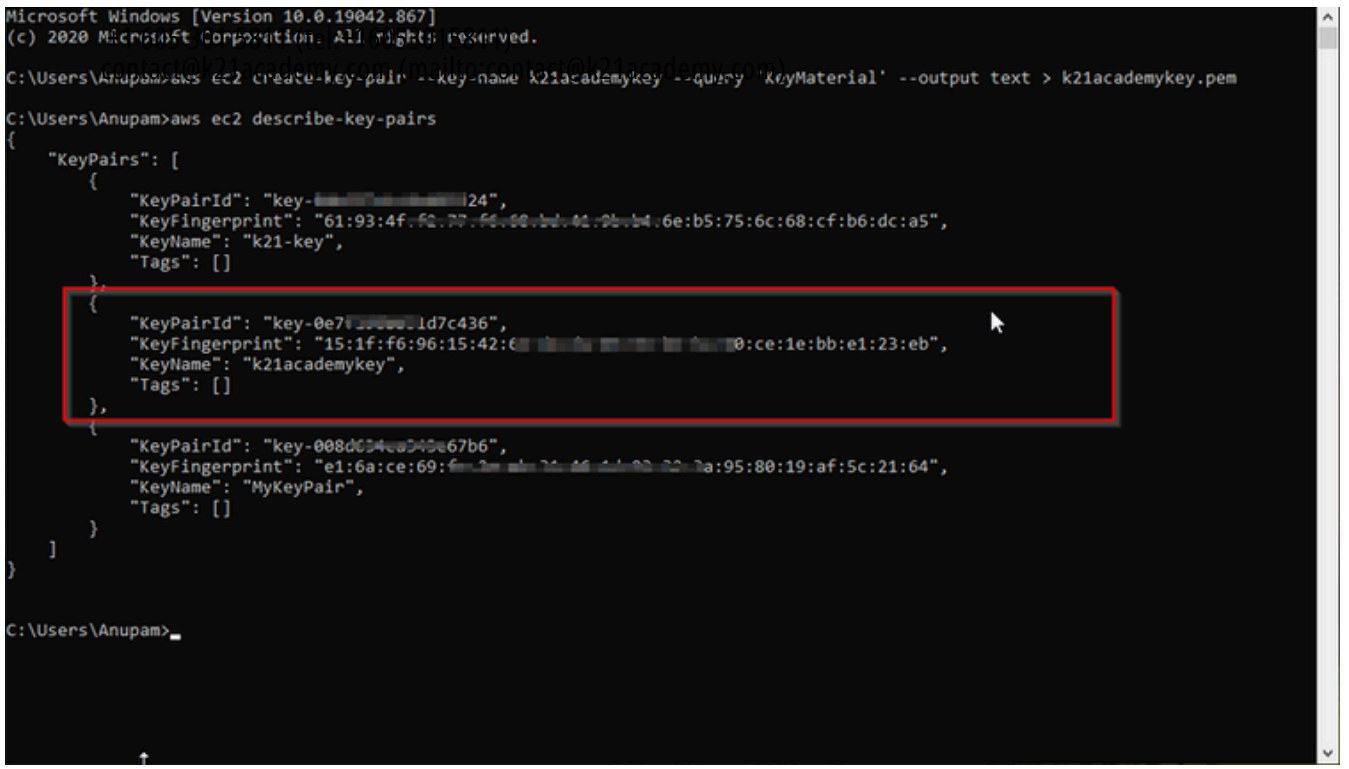




Step 4. Create Key Pair through CLI: Now, let's create a key pair with the help of the command :aws ec2 create-key-pair --key-name k21academykey (key pair name) --query 'KeyMaterial' --output text > k21academykey.pem



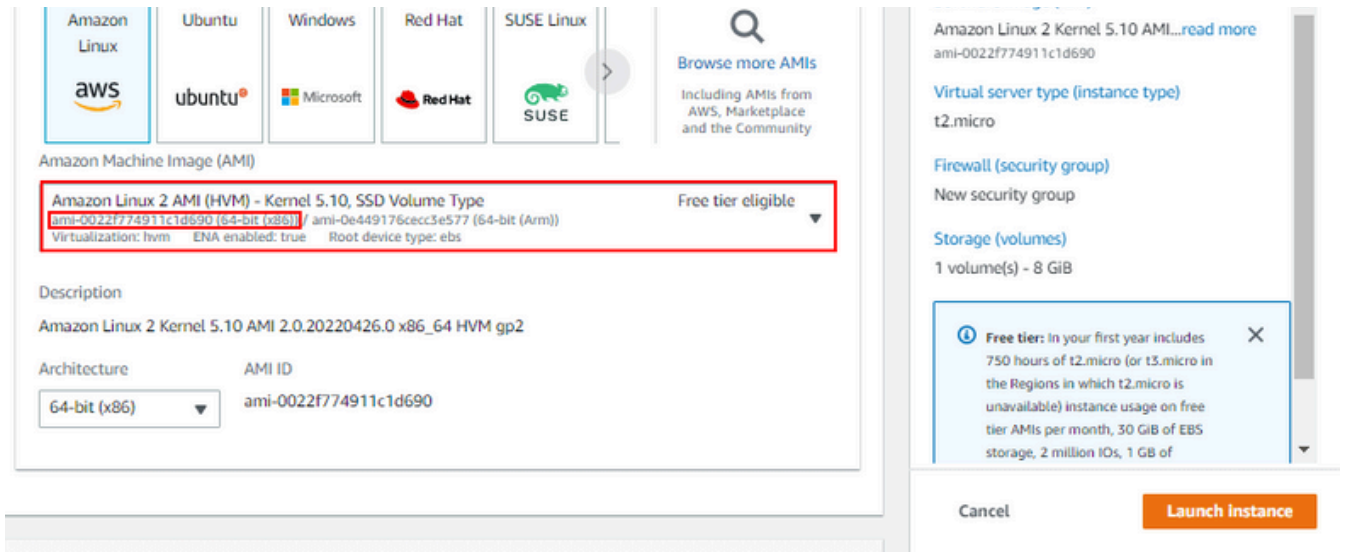
This command will create Key pair in a second, similarly to get it to verify, you can write: aws ec2 describe-key-pairs



If you want to delete the key pair at some instance, you can use the command: aws ec2 delete-key-pair --key-name k21academykey (key pair name)

Also Read: AWS Cloud Security (https://k21academy.com/amazon-web-services/aws-security-and-compliance/).

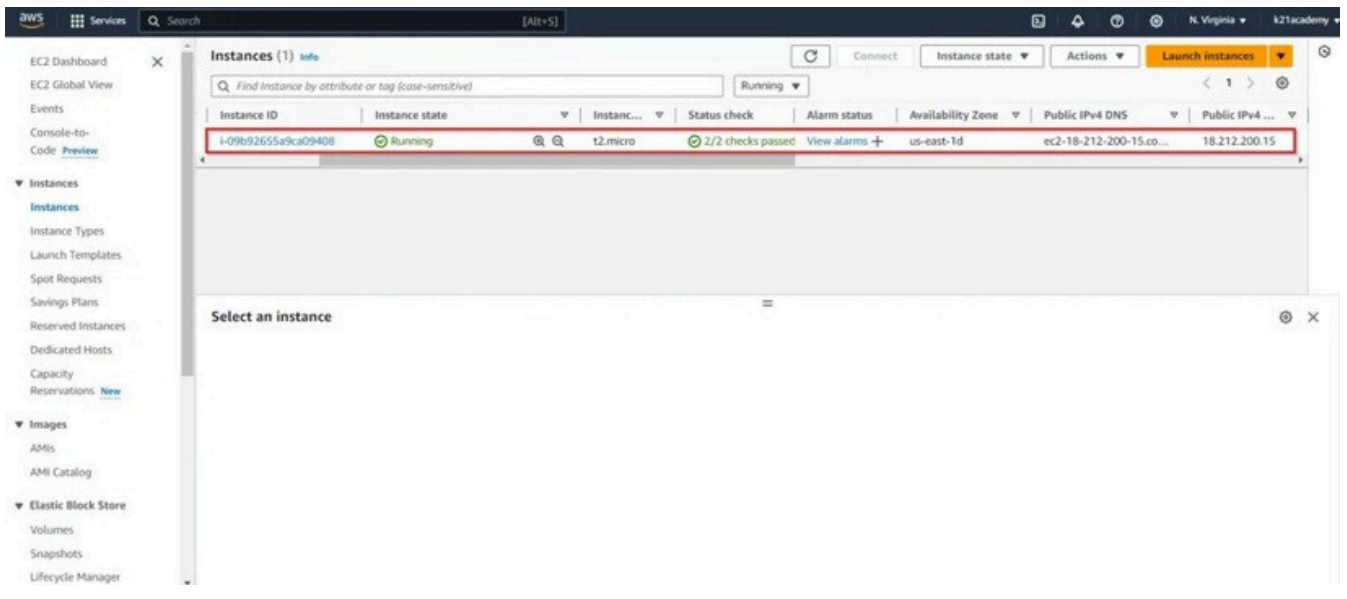
Step 5. Deploy ec2 instance through CLI: Now, in the next step, we are creating and deploying the ec2 instance through CLI; before, we have to collect ami- id by going to EC2 instances > launch instance.



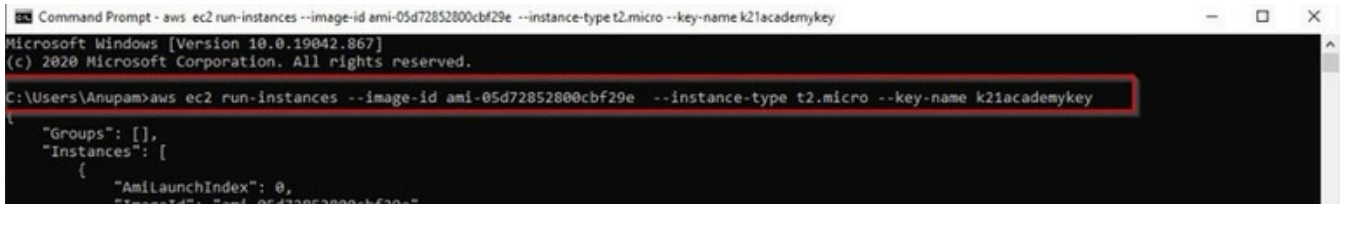
After collecting ami-id, you can follow the command: aws ec2 run-instances --image-id ami-******* (write id here) --instance-type t2.micro --key-name k21academykey

(key pair name) to launch and deploy the ec2 instance.

We can also verify the deployed instance by going to EC2 Instances in AWS Portal.



Or with the help of CLI by writing the command: aws ec2 describe-instances



```
            "ImageId": "ami-0567285286bcbf29e",
            "InstanceId": "i-0f1d23b2545285f31",
            "InstanceType": "t2.micro",
            "KeyName": "k21academykey",
            "LaunchTime": "2021-04-14T08:38:09+00:00",
            "Monitoring": {
                "State": "disabled"
            },
            "Placement": {
                "AvailabilityZone": "us-east-2b",
                "GroupName": "",
                "Tenancy": "default"
            },
            "PrivateDnsName": "ip-172-31-31-183.us-east-2.compute.internal",
            "PrivateIpAddress": "172.31.31.183",
            "ProductCodes": [],
            "PublicDnsName": "",
            "State": {
                "Code": 0,
                "Name": "pending"
            },
            "StateTransitionReason": "",
            "SubnetId": "subnet-b3f514ce",
            "VpcId": "vpc-8e32b2e5",
            "Architecture": "x86_64",
            "BlockDeviceMappings": [],
            "ClientToken": "e56969a8-5e0f-4c64-b227-72208c88654f",
            "EbsOptimized": false,
            "EnaSupport": true,
            "Hypervisor": "xen",
            "NetworkInterfaces": [
                {
                    "Attachment": {
                        "AttachTime": "2021-04-14T08:38:09+00:00",
                        "AttachmentId": "eni-attach-0cc439f76617fa2f5",
                        "DeleteOnTermination": true,
                        "DeviceIndex": 0,
                        "Status": "attaching"
                    },
                    "Description": "",
                    "Groups": [
                        {
                            "GroupName": "default",
                            "GroupId": "sg-c8e35fba"
                        }
                    ],
                    "Ipv6Addresses": [],
                    "MacAddress": "06:71:2f:85:e8:a0",
                    "NetworkInterfaceId": "eni-079dd3526d56d1f5d",
                    "OwnerId": "063372241087",
                    "PrivateDnsName": "ip-172-31-31-183.us-east-2.compute.internal",
                    "PrivateIpAddress": "172.31.31.183",
                    "PrivateIpAddresses": [
```


This showed we have successfully created and deployed the EC2 instance.

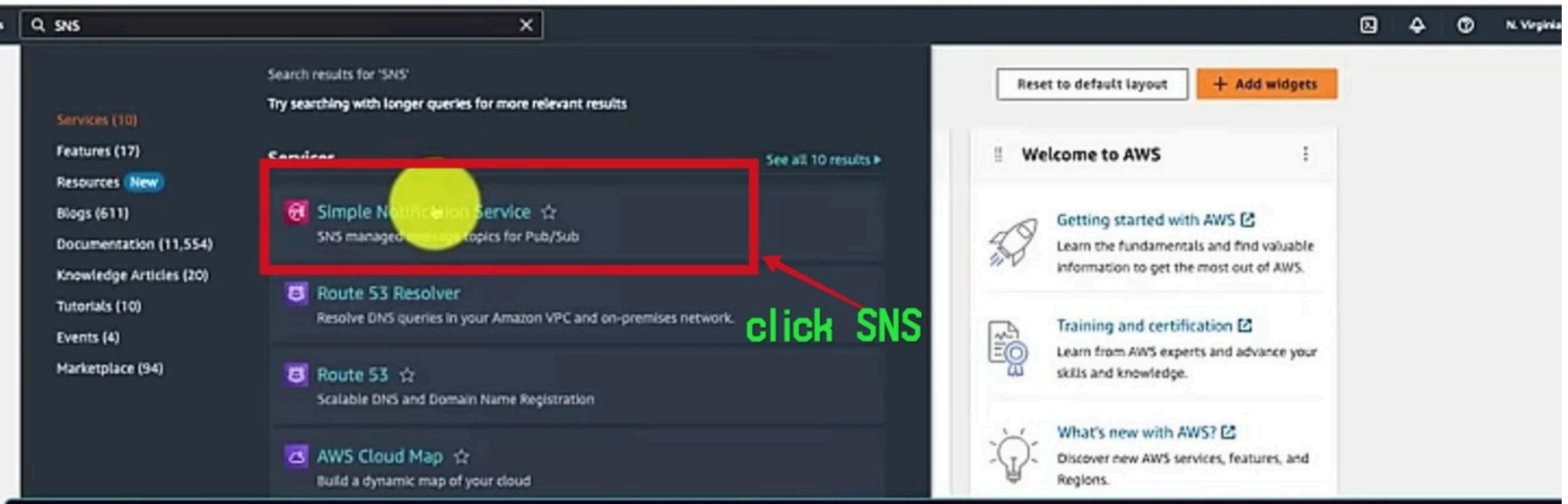
AWS cloud watch

- What does website traffic look like?
- How is performance?
- How much bandwidth is my app using?
- Are compute resources optimized?

Amazon CloudWatch

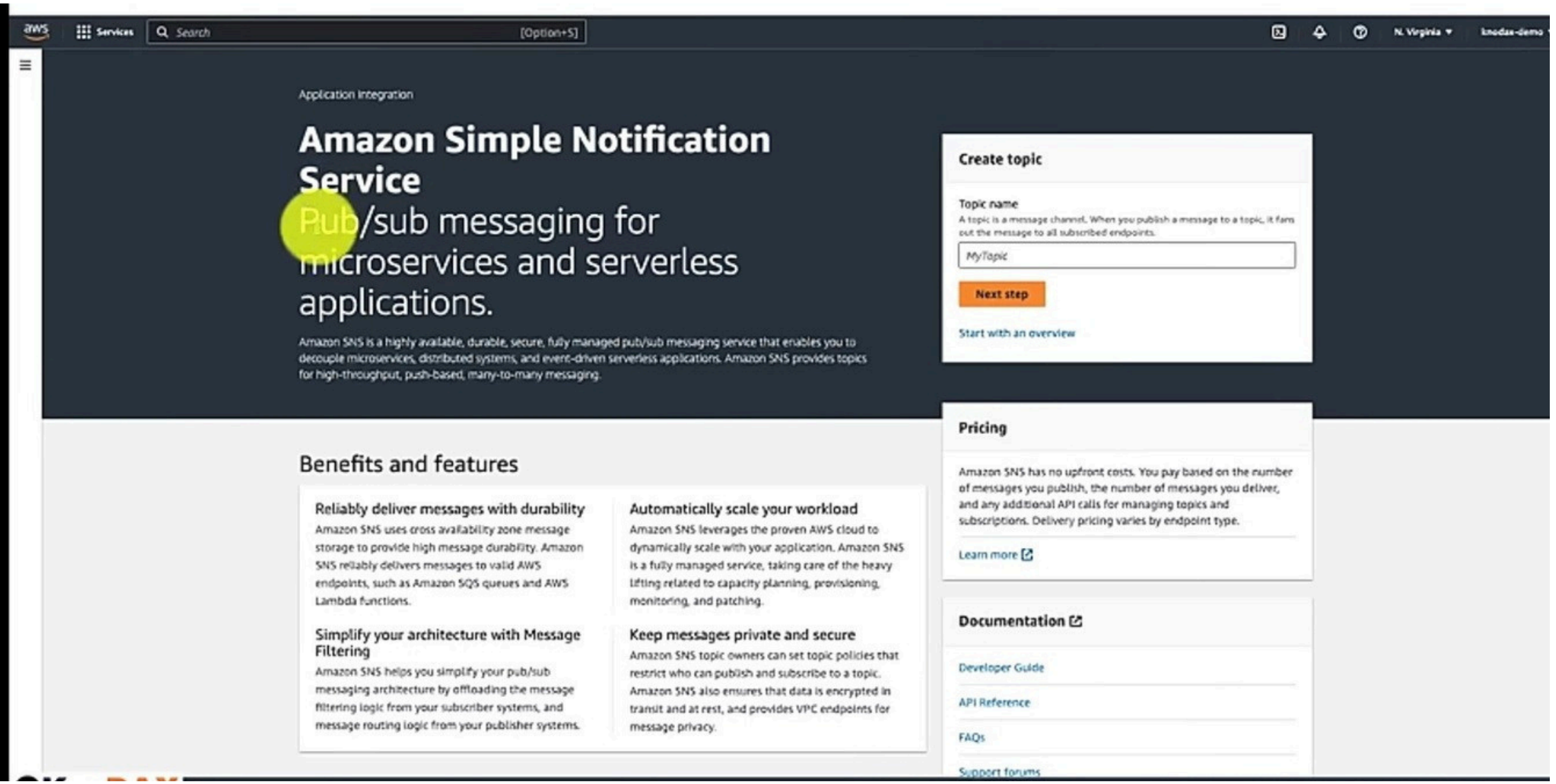
Provides visibility into AWS resources and applications

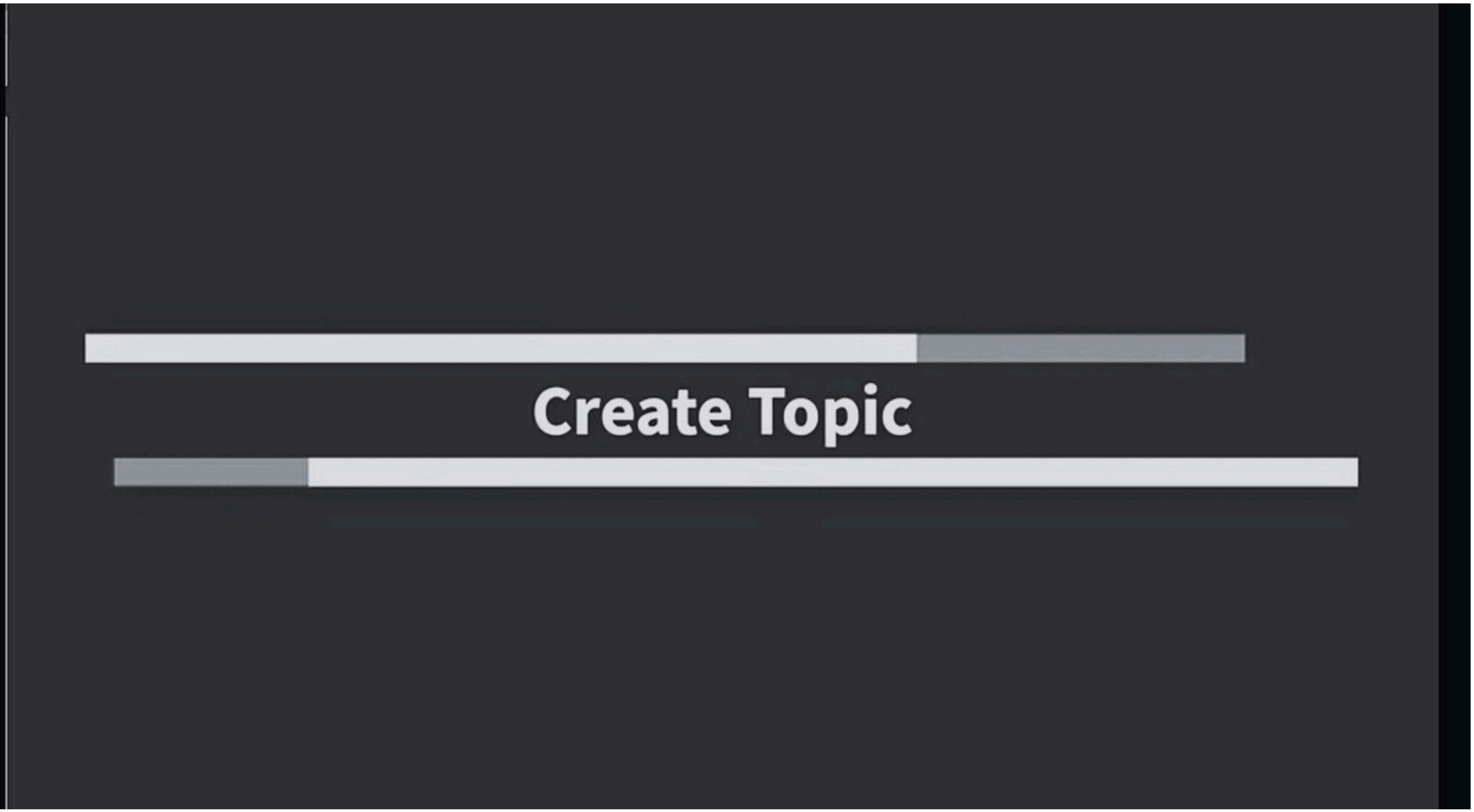
- Near-real-time **metrics** such as CPU, disk, GPU utilization
- **CloudWatch Logs** allow you to collect logs from AWS and non-AWS sources, to search/extract metrics
- **Alarms** can trigger notifications for metrics
  - Example: Alarm to notify you when you've hit CPU utilization of 90%
- **Dashboards** offer at-a-glance views



What is Amazon SNS used for?

Amazon Simple Notification Service (Amazon SNS) is a managed service that provid message delivery from publishers to subscribers (also known as producers and consumers). Publishers communicate asynchronously with subscribers by sending messages to a topic, which is a logical access point and communication channel.


Services
Search
[Option+S]
N. Virginia
Application Integration
Amazon Simple Notification Service
Pub/sub messaging for microservices and serverless applications.
Amazon SNS is a highly available, durable, secure, fully managed pub/sub messaging service that enables you to decouple microservices, distributed systems, and event-driven serverless applications. Amazon SNS provides topics for high-throughput, push-based, many-to-many messaging.
Create topic
Topic name
A topic is a message channel. When you publish a message to a topic, it fans out the message to all subscribed endpoints.
MyTopic
Next step
Start with an overview
Pricing
Amazon SNS has no upfront costs. You pay based on the number of messages you publish, the number of messages you deliver, and any additional API calls for managing topics and subscriptions. Delivery pricing varies by endpoint type.
Learn more
Documentation
Developer Guide
API Reference
FAQs
Support forums
Benefits and features
Reliably deliver messages with durability
Amazon SNS uses cross availability zone message storage to provide high message durability. Amazon SNS reliably delivers messages to valid AWS endpoints, such as Amazon SQS queues and AWS Lambda functions.
Automatically scale your workload
Amazon SNS leverages the proven AWS cloud to dynamically scale with your application. Amazon SNS is a fully managed service, taking care of the heavy lifting related to capacity planning, provisioning, monitoring, and patching.
Simplify your architecture with Message Filtering
Amazon SNS helps you simplify your pub/sub messaging architecture by offloading the message filtering logic from your subscriber systems, and message routing logic from your publisher systems.
Keep messages private and secure
Amazon SNS topic owners can set topic policies that restrict who can publish and subscribe to a topic. Amazon SNS also ensures that data is encrypted in transit and at rest, and provides VPC endpoints for message privacy.


Create Topic

[Option+S]

click on create

Application Integration

Amazon Simple Notification Service

Pub/sub messaging for microservices and serverless applications.

Amazon SNS is a highly available, durable, secure, fully managed pub/sub messaging service that enables you to decouple microservices, distributed systems, and event-driven serverless applications. Amazon SNS provides topics for high-throughput, push-based, many-to-many messaging.

Create topic

A topic is a message channel. When you publish a message to a topic, it fans out the message to all subscribed endpoints.

SNS_Demo

Next step

Start with an overview

Pricing

Amazon SNS has no upfront costs. You pay based on the number of messages you publish, the number of messages you deliver, and any additional API calls for managing topics and subscriptions. Delivery pricing varies by endpoint type.

Learn more

Documentation

Developer Guide

API Reference

Benefits and features

Reliably deliver messages with durability

Amazon SNS uses cross availability zone message storage to provide high message durability. Amazon SNS reliably delivers messages to valid AWS endpoints, such as Amazon SQS queues and AWS Lambda functions.

Automatically scale your workload

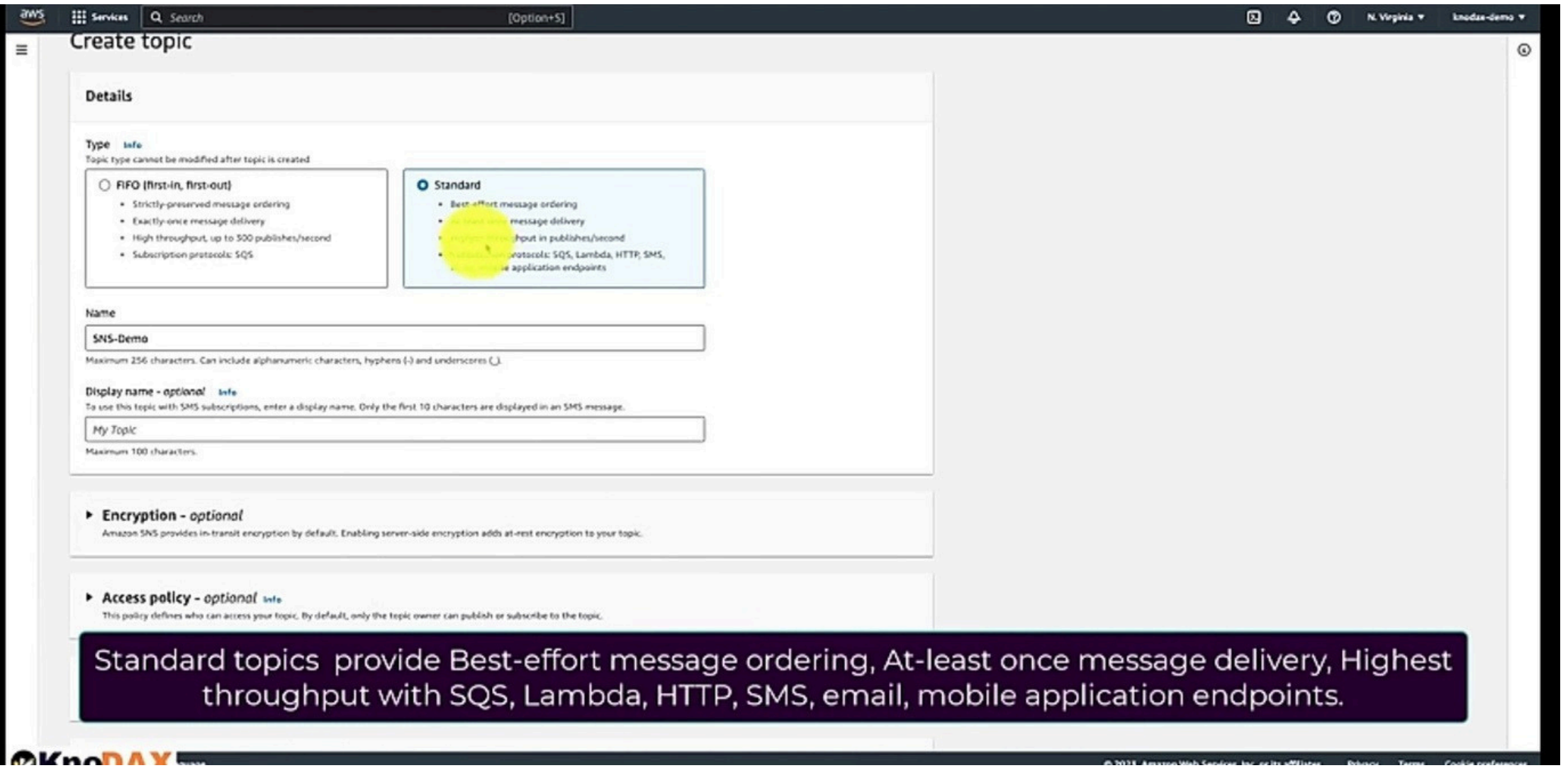
Amazon SNS leverages the proven AWS cloud to dynamically scale with your application. Amazon SNS is a fully managed service, taking care of the heavy lifting related to capacity planning, provisioning, monitoring, and patching.

Simplify your architecture with Message Filtering

Amazon SNS helps you simplify your pub/sub messaging architecture by offloading the message filtering logic from your subscriber systems, and

Keep messages private and secure

Amazon SNS topic owners can set topic policies that restrict who can publish and subscribe to a topic. Amazon SNS also ensures that data is encrypted in transit and at rest, and provides VPC endpoints for


Services
Search
[Option+S]
N. Virginia
knodax-demo
Create topic
Details
Type Info
Topic type cannot be modified after topic is created
FIFO (first-in, first-out)
Strictly-preserved message ordering
Exactly-once message delivery
High throughput, up to 300 publishes/second
Subscription protocols: SQS
Standard
Best-effort message ordering
message delivery
hput in publishes/second
otocols: SQS, Lambda, HTTP, SMS,
e application endpoints
Name
SNS-Demo
Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (_).
Display name - optional Info
To use this topic with SMS subscriptions, enter a display name. Only the first 10 characters are displayed in an SMS message.
My Topic
Maximum 100 characters.
Encryption - optional
Amazon SNS provides in-transit encryption by default. Enabling server-side encryption adds at-rest encryption to your topic.
Access policy - optional Info
This policy defines who can access your topic. By default, only the topic owner can publish or subscribe to the topic.
Standard topics provide Best-effort message ordering, At-least once message delivery, Highest throughput with SQS, Lambda, HTTP, SMS, email, mobile application endpoints.

Services | Search [Option+S] | N. Virginia | knodax-demo

Name

SNS-Demo

Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (_).

Display name - *optional* Info

To use this topic with SMS subscriptions, enter a display name. Only the first 10 characters are displayed in an SMS message.

My Topic

Maximum 100 characters.

leave it as default

▸ **Encryption - *optional***

Amazon SNS provides in-transit encryption by default. Enabling server-side encryption adds at-rest encryption to your topic.

▸ **Access policy - *optional*** Info

This policy defines who can access your topic. By default, only the topic owner can publish or subscribe to the topic.

▸ **Data protection policy - *optional*** Info

This policy defines which sensitive data to monitor and to prevent from being exchanged via your topic.

▸ **Delivery policy (HTTP/S) - *optional*** Info

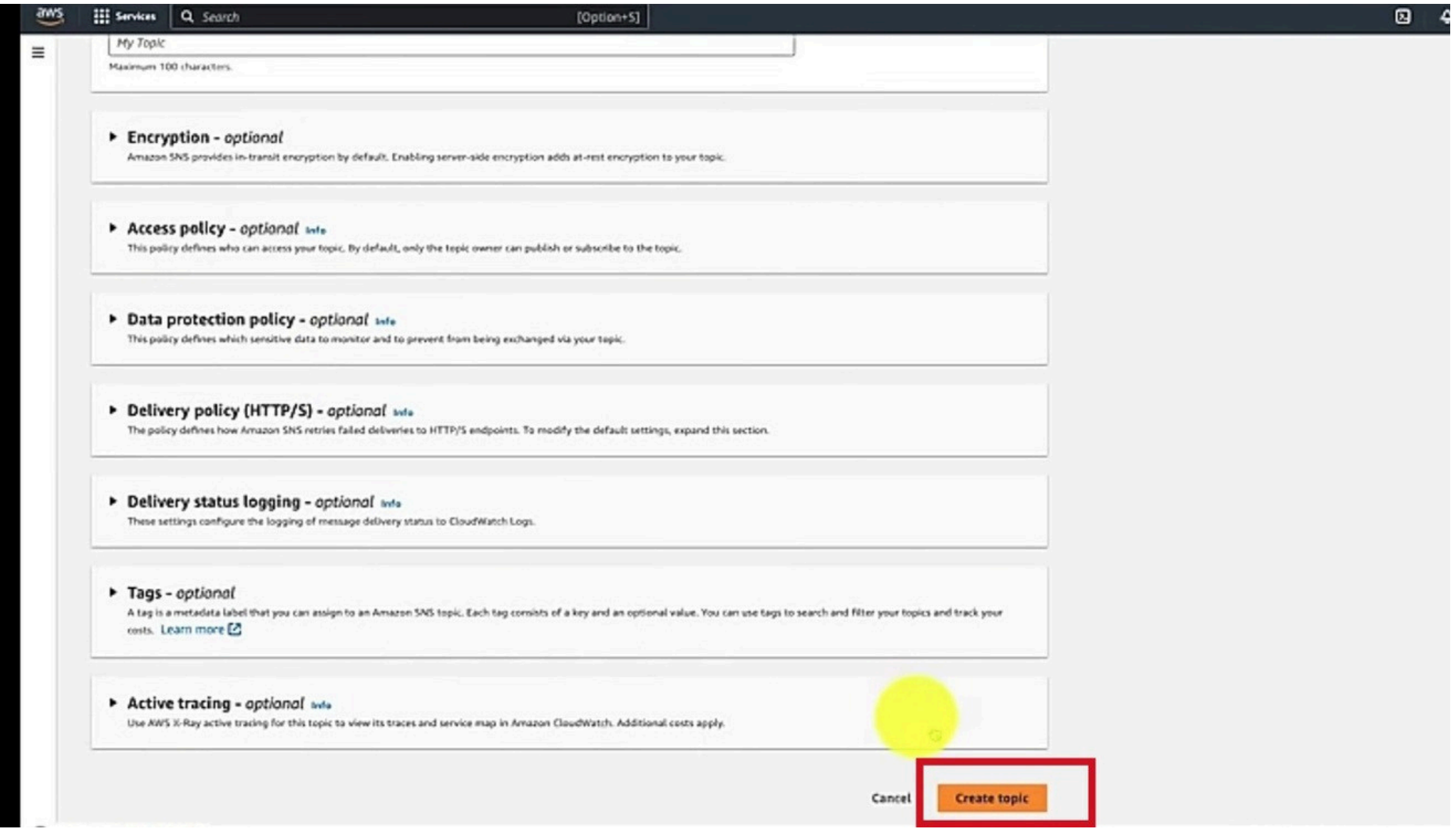
The policy defines how Amazon SNS retries failed deliveries to HTTP/S endpoints. To modify the default settings, expand this section.

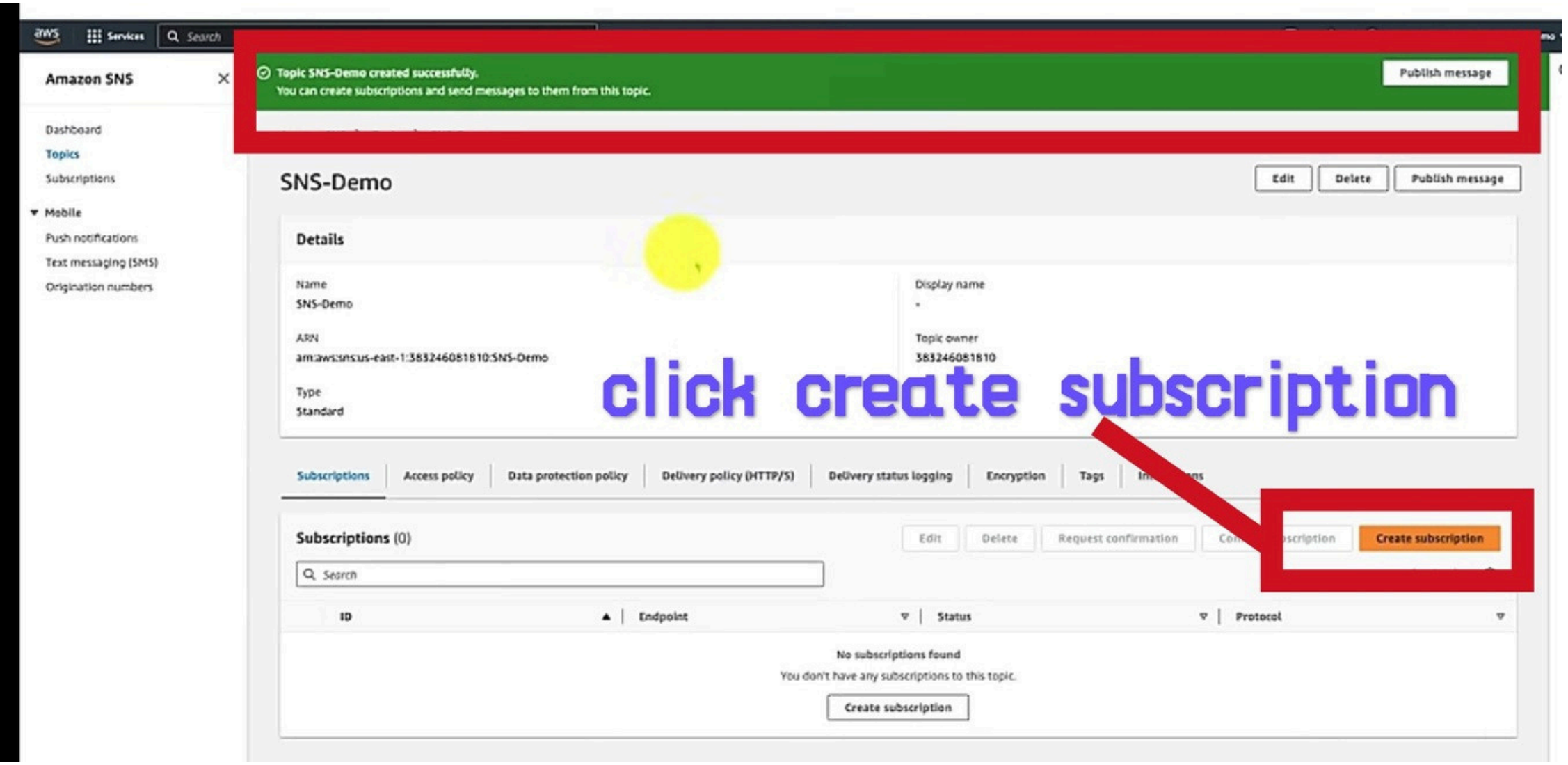
▸ **Delivery status logging - *optional*** Info

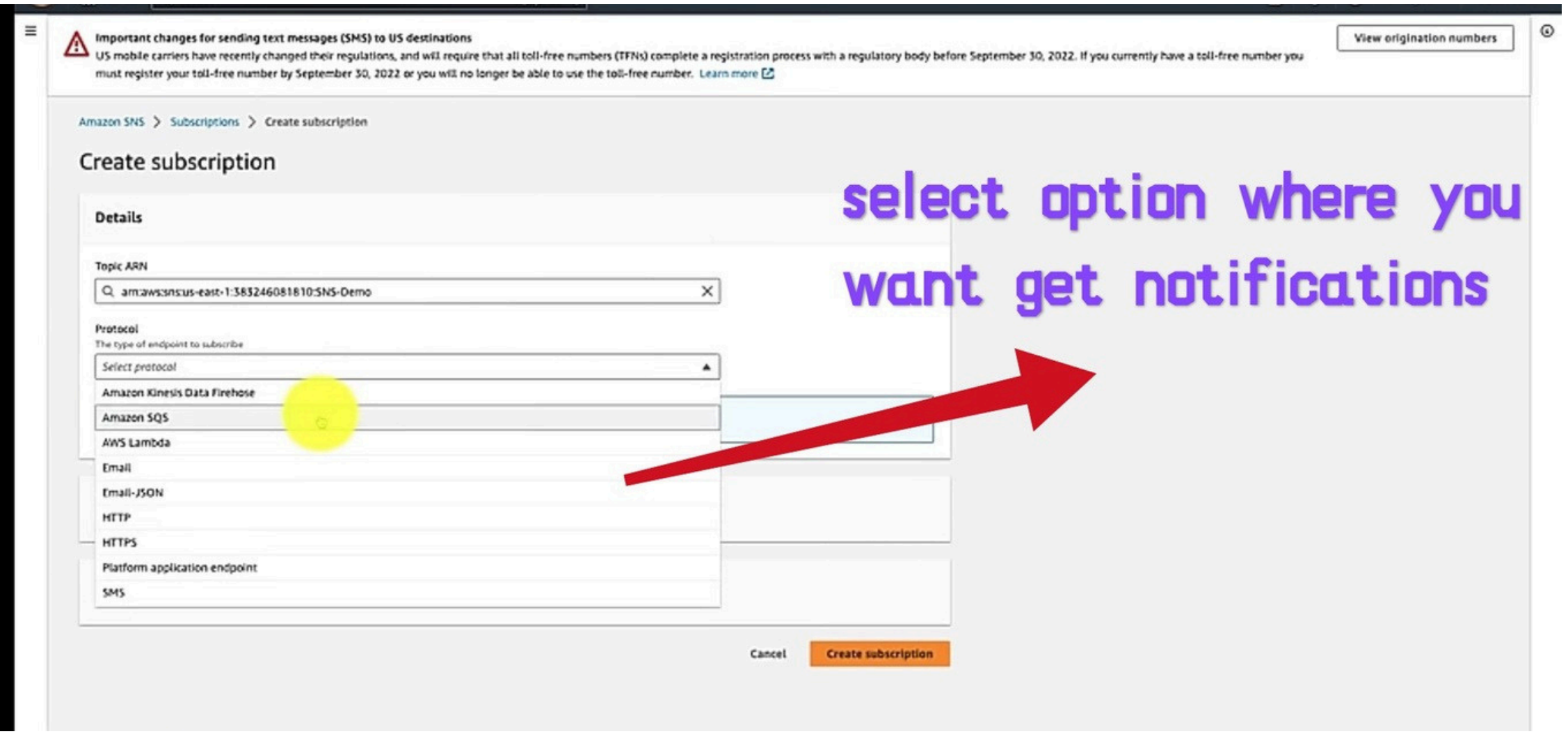
These settings configure the logging of message delivery status to CloudWatch Logs.

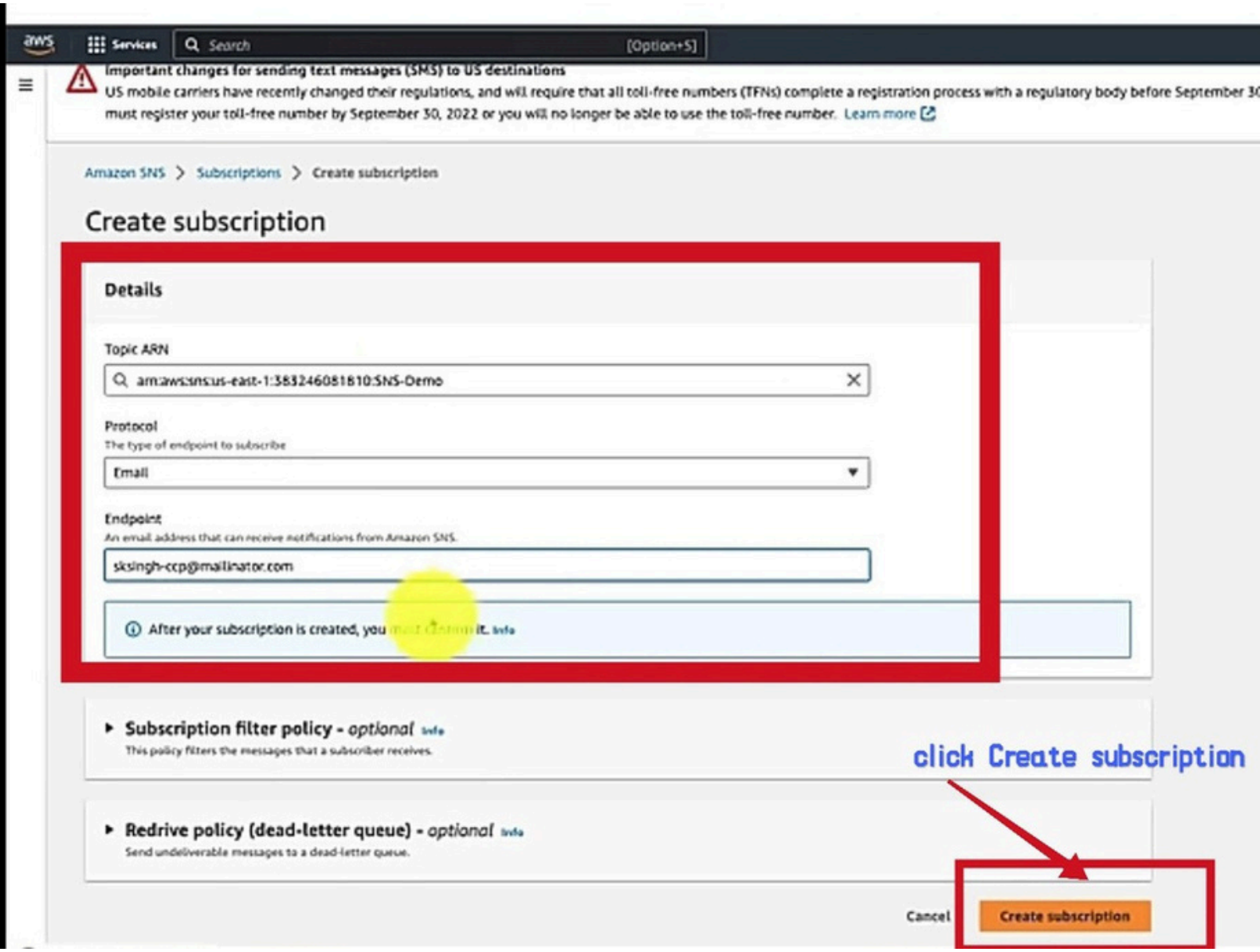
▸ **Tags - *optional***

A tag is a metadata label that you can assign to an Amazon SNS topic. Each tag consists of a key and an optional value. You can use tags to search and filter your topics and track your


Services
Search
[Option+S]
My Topic
Maximum 100 characters.
Encryption - optional
Amazon SNS provides in-transit encryption by default. Enabling server-side encryption adds at-rest encryption to your topic.
Access policy - optional Info
This policy defines who can access your topic. By default, only the topic owner can publish or subscribe to the topic.
Data protection policy - optional Info
This policy defines which sensitive data to monitor and to prevent from being exchanged via your topic.
Delivery policy (HTTP/S) - optional Info
The policy defines how Amazon SNS retries failed deliveries to HTTP/S endpoints. To modify the default settings, expand this section.
Delivery status logging - optional Info
These settings configure the logging of message delivery status to CloudWatch Logs.
Tags - optional
A tag is a metadata label that you can assign to an Amazon SNS topic. Each tag consists of a key and an optional value. You can use tags to search and filter your topics and track your costs. Learn more
Active tracing - optional Info
Use AWS X-Ray active tracing for this topic to view its traces and service map in Amazon CloudWatch. Additional costs apply.
Cancel
Create topic


Amazon SNS
Dashboard
Topics
Subscriptions
Mobile
Push notifications
Text messaging (SMS)
Origination numbers
Topic SNS-Demo created successfully.
You can create subscriptions and send messages to them from this topic.
Publish message
SNS-Demo
Edit
Delete
Publish message
Details
Name
SNS-Demo
ARN
arn:aws:sns:us-east-1:383246081810:SNS-Demo
Type
Standard
Display name
-
Topic owner
383246081810
click create subscription
Subscriptions
Access policy
Data protection policy
Delivery policy (HTTP/S)
Delivery status logging
Encryption
Tags
Subscriptions (0)
Edit
Delete
Request confirmation
Create subscription
Search
ID
Endpoint
Status
Protocol
No subscriptions found
You don't have any subscriptions to this topic.
Create subscription


Important changes for sending text messages (SMS) to US destinations
US mobile carriers have recently changed their regulations, and will require that all toll-free numbers (TFNs) complete a registration process with a regulatory body before September 30, 2022. If you currently have a toll-free number you must register your toll-free number by September 30, 2022 or you will no longer be able to use the toll-free number. Learn more
View origination numbers
Amazon SNS > Subscriptions > Create subscription
Create subscription
Details
Topic ARN
arn:aws:sns:us-east-1:383246081810:SNS-Demo
Protocol
The type of endpoint to subscribe
Select protocol
Amazon Kinesis Data Firehose
Amazon SQS
AWS Lambda
Email
Email-JSON
HTTP
HTTPS
Platform application endpoint
SMS
select option where you want get notifications
Cancel
Create subscription


Services
Search
[Option+S]
N. Virginia
Important changes for sending text messages (SMS) to US destinations
US mobile carriers have recently changed their regulations, and will require that all toll-free numbers (TFNs) complete a registration process with a regulatory body before September 30, 2022. If you currently have a toll-free number you must register your toll-free number by September 30, 2022 or you will no longer be able to use the toll-free number. Learn more
Amazon SNS > Subscriptions > Create subscription
Create subscription
Details
Topic ARN
arn:aws:sns:us-east-1:383246081810:SNS-Demo
Protocol
The type of endpoint to subscribe
Email
Endpoint
An email address that can receive notifications from Amazon SNS.
sksingh-ccp@mailinator.com
After your subscription is created, you
it. Info
Subscription filter policy - optional Info
This policy filters the messages that a subscriber receives.
Redrive policy (dead-letter queue) - optional Info
Send undeliverable messages to a dead-letter queue.
Cancel
Create subscription
click Create subscription

us-east-1.console.aws.amazon.com/sns/v3/home?region=us-east-1#/subscription/arn:aws:sns:us-east-1:383246081810:SNS-Demo:b5dca1d3-6cf4-406e-98f8-31a3c983d4e3

aws Services Search [Option+S] N. Virginia knodax-demo

Amazon SNS

- Dashboard
- Topics
- Subscriptions
- Mobile
  - Push notifications
  - Text messaging (SMS)
  - Origination numbers

Important changes for sending text messages (SMS) to US destinations
US mobile carriers have recently changed their regulations, and will require that all toll-free numbers (TFNs) complete a registration process with a regulatory body before September 30, 2022. If you currently have a toll-free number you must register your toll-free number by September 30, 2022 or you will no longer be able to use the toll-free number. Learn more

View origination numbers

Subscription to SNS-Demo created successfully.
The ARN of the subscription is arn:aws:sns:us-east-1:383246081810:SNS-Demo:b5dca1d3-6cf4-406e-98f8-31a3c983d4e3.

Amazon SNS > Topics > SNS-Demo > Subscription: b5dca1d3-6cf4-406e-98f8-31a3c983d4e3

Subscription: b5dca1d3-6cf4-406e-98f8-31a3c983d4e3

Edit Delete

Details

ARN
arn:aws:sns:us-east-1:383246081810:SNS-Demo:b5dca1d3-6cf4-406e-98f8-31a3c983d4e3

Status
Pending confirmation

Endpoint
sksingh-ccp@mailinator.com

Protocol
EMAIL

Topic
SNS-Demo

Subscription Principal
arn:aws:iam::383246081810:root

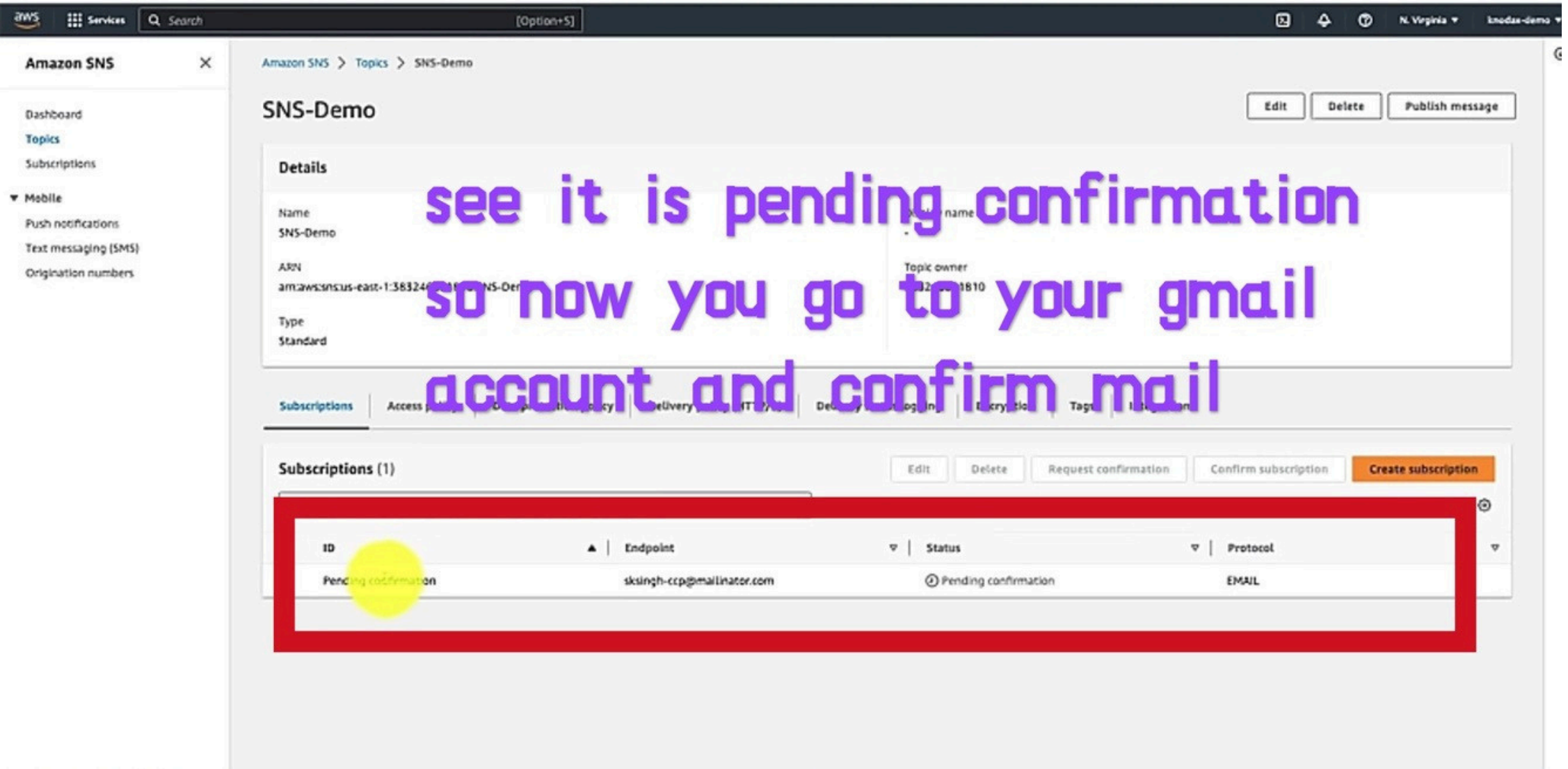
SNS Created successfully

Subscription filter policy | Redrive policy (dead-letter queue)

Subscription filter policy Info

This policy filters the messages that a subscriber receives.

No filter policy configured for this subscription.
To apply a filter policy, edit this subscription.


Services
Search
[Option+S]
N. Virginia
Amazon SNS
Dashboard
Topics
Subscriptions
Mobile
Push notifications
Text messaging (SMS)
Origination numbers
Amazon SNS > Topics > SNS-Demo
SNS-Demo
Edit
Delete
Publish message
Details
Name
SNS-Demo
ARN
Type
Standard
Topic owner
see it is pending confirmation
so now you go to your gmail
account and confirm mail
Subscriptions
Subscriptions (1)
Edit
Delete
Request confirmation
Confirm subscription
Create subscription
ID
Endpoint
Status
Protocol
Pending confirmation
sksingh-ccp@mailinator.com
Pending confirmation
EMAIL

Mailinator

Home Email Pricing Documentation FAQ

- Public Inboxes
- Public SMS
- Pricing
- Request Trial

Use Mailinator for Email Testing?
Apply for a FREE
Verified Pro Subscription!

Public Message > AWS Notification - Subscription Confirmation

Back

To	sksingh-ccp
From	no-reply@sns.amazonaws.com
Sending IP	54.240.34.3
Received	2023-07-05 15:52:09

Delet

HTML JSON RAW LINKS SMTP_LOG ATTACHMENTS

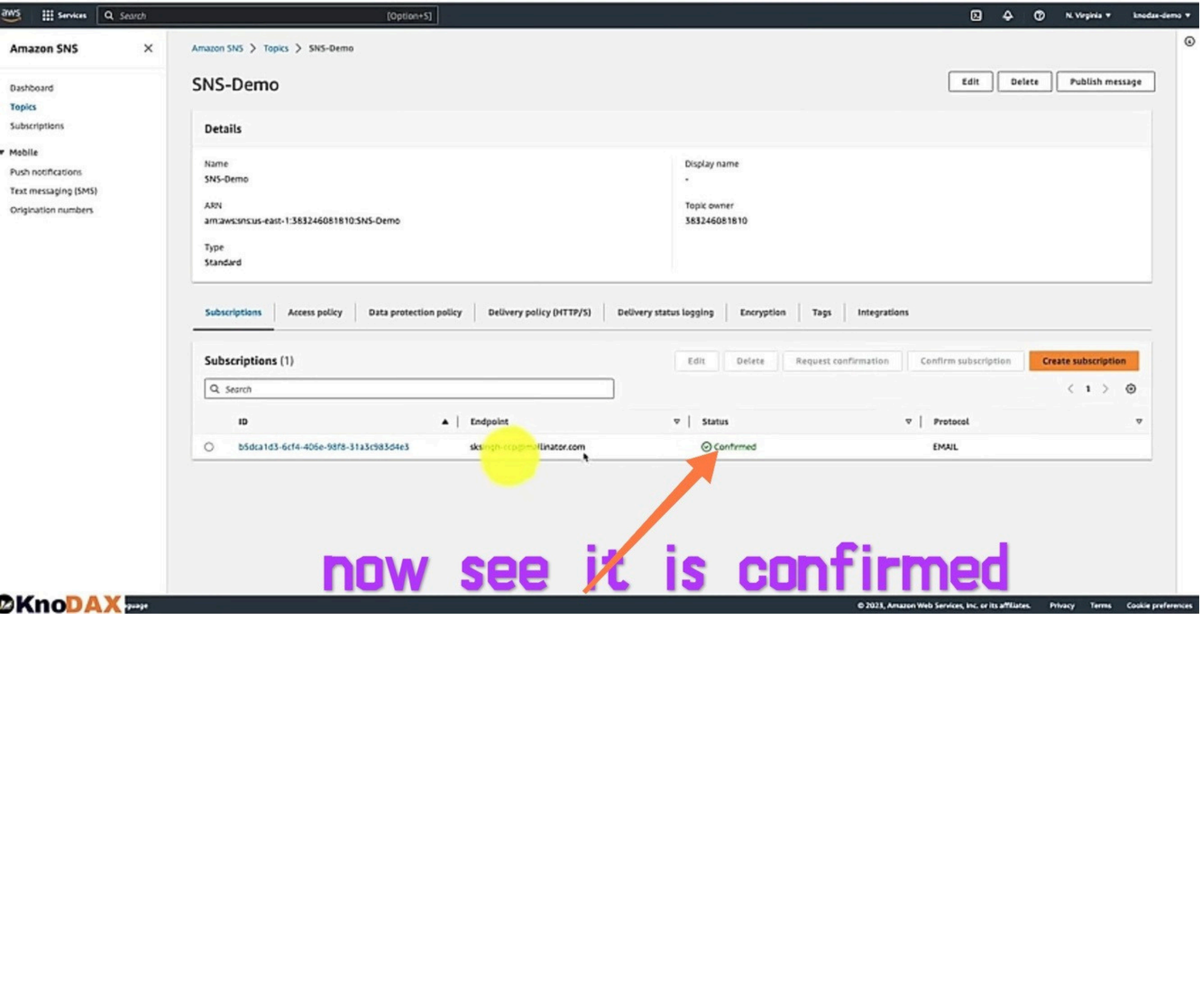
You have chosen to subscribe to the topic:
arn:aws:sns:us-east-1:383246081810:SNS-Demo

To confirm this subscription, click or visit the link below (If this was in error no action is necessary):
Confirm subscription

Please do not reply directly to this email. If you wish to remove yourself from receiving all future SNS subscription confirmation requests please send an email to sns-opt-out

confirm mail

System Status
Terms
Privacy
Accessibility
Security


Services
Search
[Option+S]
N. Virginia
knodax-demo
Amazon SNS
Dashboard
Topics
Subscriptions
Mobile
Push notifications
Text messaging (SMS)
Origination numbers
Amazon SNS > Topics > SNS-Demo
SNS-Demo
Edit
Delete
Publish message
Details
Name
SNS-Demo
ARN
arn:aws:sns:us-east-1:383246081810:SNS-Demo
Type
Standard
Display name
-
Topic owner
383246081810
Subscriptions
Access policy
Data protection policy
Delivery policy (HTTP/S)
Delivery status logging
Encryption
Tags
Integrations
Subscriptions (1)
Edit
Delete
Request confirmation
Confirm subscription
Create subscription
Search
ID
Endpoint
Status
Protocol
b5dca1d3-6cf4-406e-98f8-31a3c983d4e3
Confirmed
EMAIL
now see it is confirmed
KnoDAX
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

CloudWatch Alarm from EC2 Instance

Original

aws | Services | Search [Option+S] | N. Virginia | knodax-demo

New EC2 Experience
Tell us what you think

- EC2 Dashboard
- EC2 Global View
- Events
- Instances
  - Instances
  - Instance Types
  - Launch Templates
  - Spot Requests
  - Savings Plans
  - Reserved Instances
  - Dedicated Hosts
  - Scheduled Instances
  - Capacity Reservations
- Images
  - AMIs
  - AMI Catalog
- Elastic Block Store
  - Volumes
  - Snapshots
  - Lifecycle Manager
- Network & Security
  - Security Groups
  - Elastic IPs
  - Placement Groups
  - Key Pairs

Instances (1/1) Info

Connect | Instance state | Actions | Launch instances

Find instance by attribute or tag (case-sensitive)

Instance state = running | Clear filters

1

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
Test Demo Ser...	i-0d888d4f39c733d5d	Running	t2.micro	2/2 checks passed	No alarms +	us-east-1b	ec2-3-94-116-234.com...	3.94.116.234	-

1

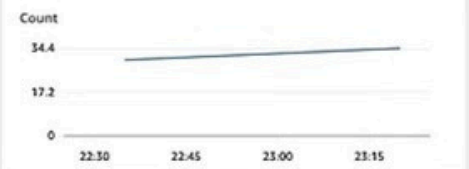
click instance

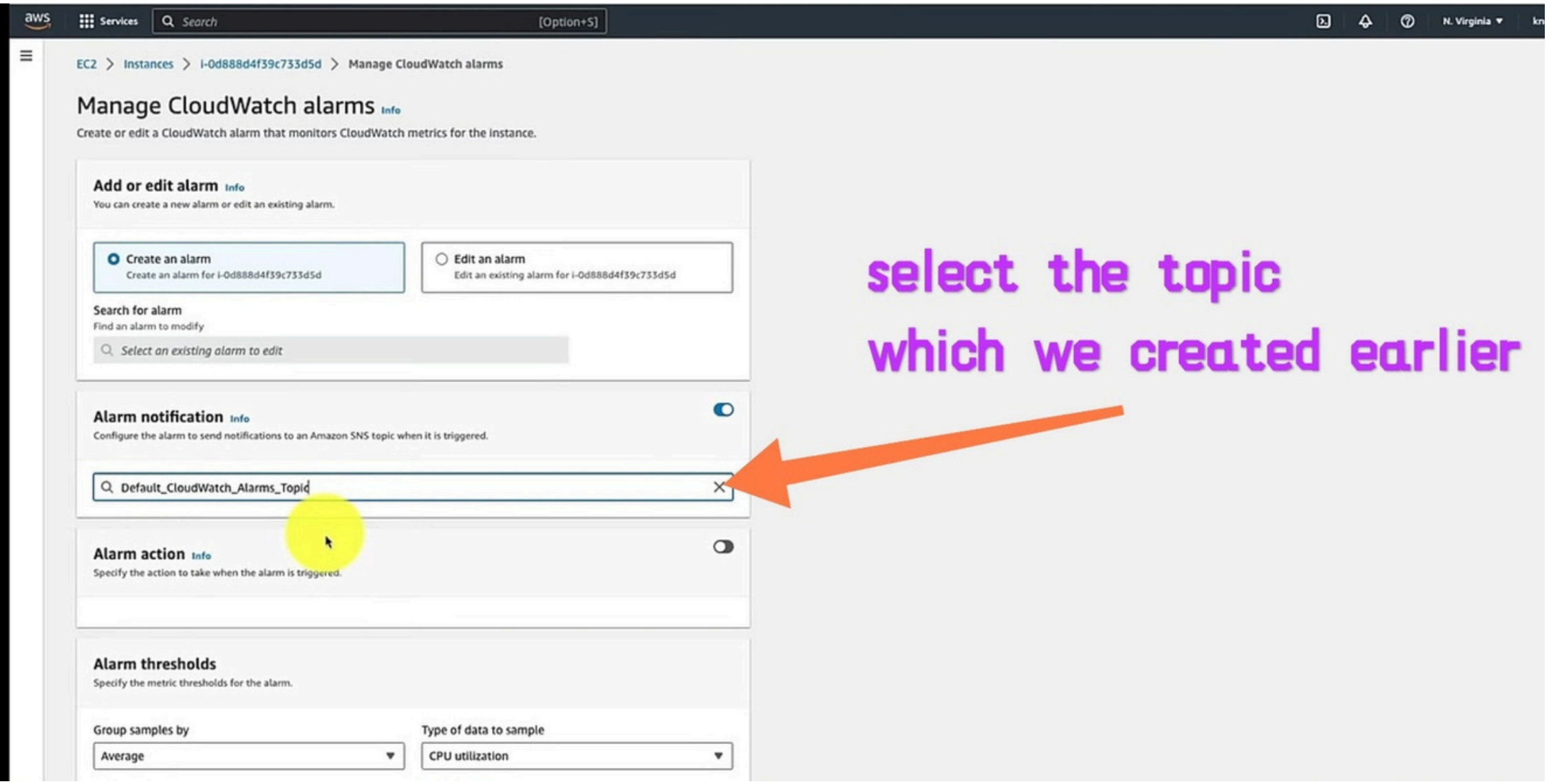
Click on Alarm status + icon

Instance: i-0d888d4f39c733d5d (Test Demo Server)

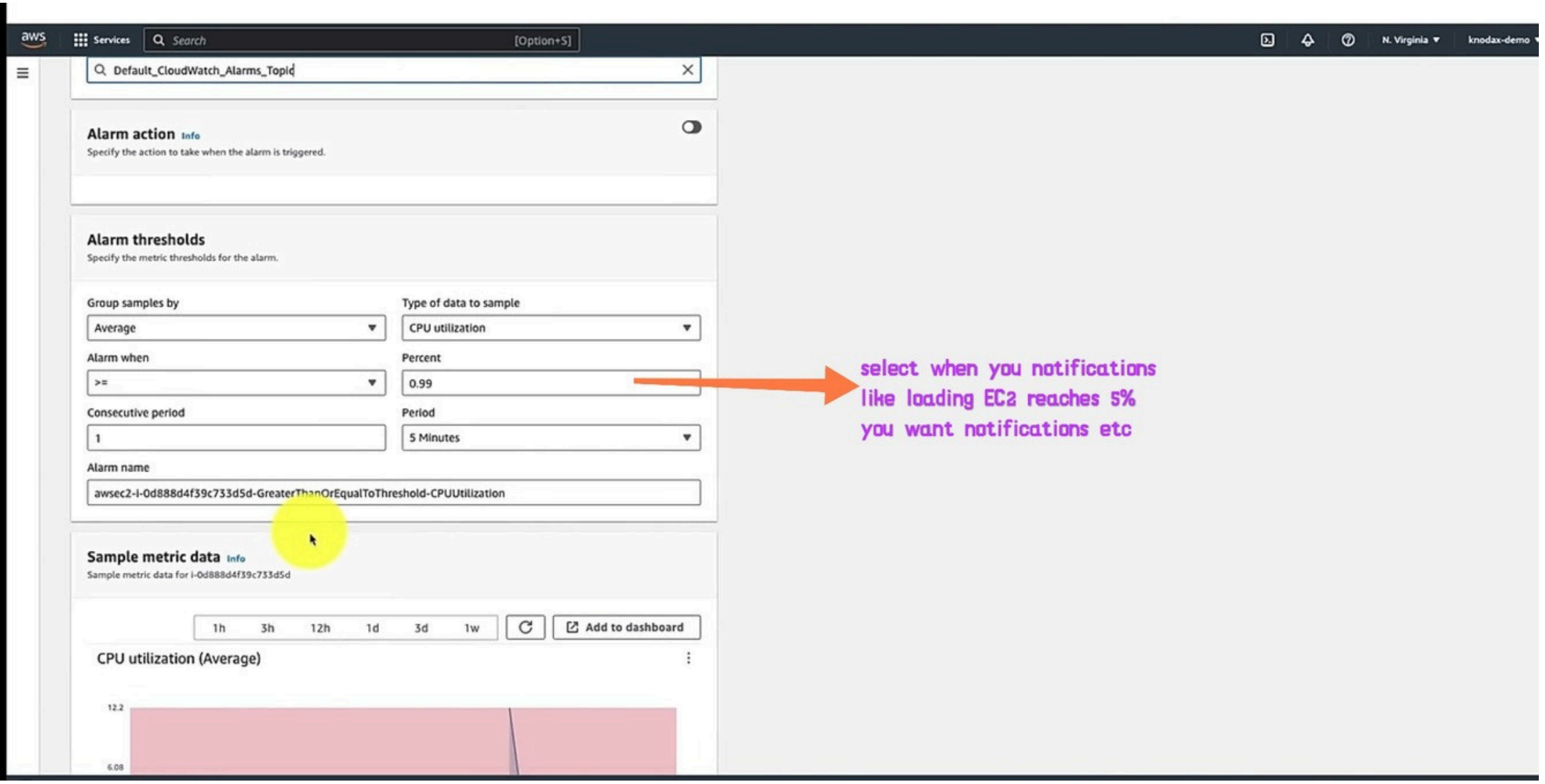
CPU credit usage (count)

CPU credit balance (count)





select the topic
which we created earlier


aws
Services
Search
[Option+S]
N. Virginia
knodax-demo
Default_CloudWatch_Alarms_Topic
Alarm action Info
Specify the action to take when the alarm is triggered.
Alarm thresholds
Specify the metric thresholds for the alarm.
Group samples by
Average
Type of data to sample
CPU utilization
Alarm when
>=
Percent
0.99
Consecutive period
1
Period
5 Minutes
Alarm name
awsec2-i-0d888d4f39c733d5d-GreaterThanOrEqualToThreshold-CPUUtilization
Sample metric data Info
Sample metric data for i-0d888d4f39c733d5d
1h
3h
12h
1d
3d
1w
Add to dashboard
CPU utilization (Average)
12.2
6.08
select when you notifications
like loading EC2 reaches 5%
you want notifications etc

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#ManageCloudWatchAlarms:instanceId=i-0d888d4f39c733d5d

aws Services Search [Option+S] N. Virginia knodax-demo

Specify the metric thresholds for the alarm.

Group samples by: Average

Type of data to sample: CPU utilization

Alarm when: >=

Percent: 0.99

Consecutive period: 1

Period: 5 Minutes

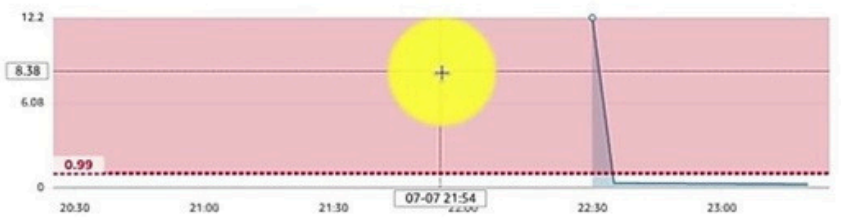
Alarm name: awsec2-i-0d888d4f39c733d5d-GreaterThanOrEqualToThreshold-CPUUtilization

Sample metric data Info

Sample metric data for i-0d888d4f39c733d5d

1h 3h 12h 1d 3d 1w Add to dashboard

CPU utilization (Average)



click Create

Cancel Create

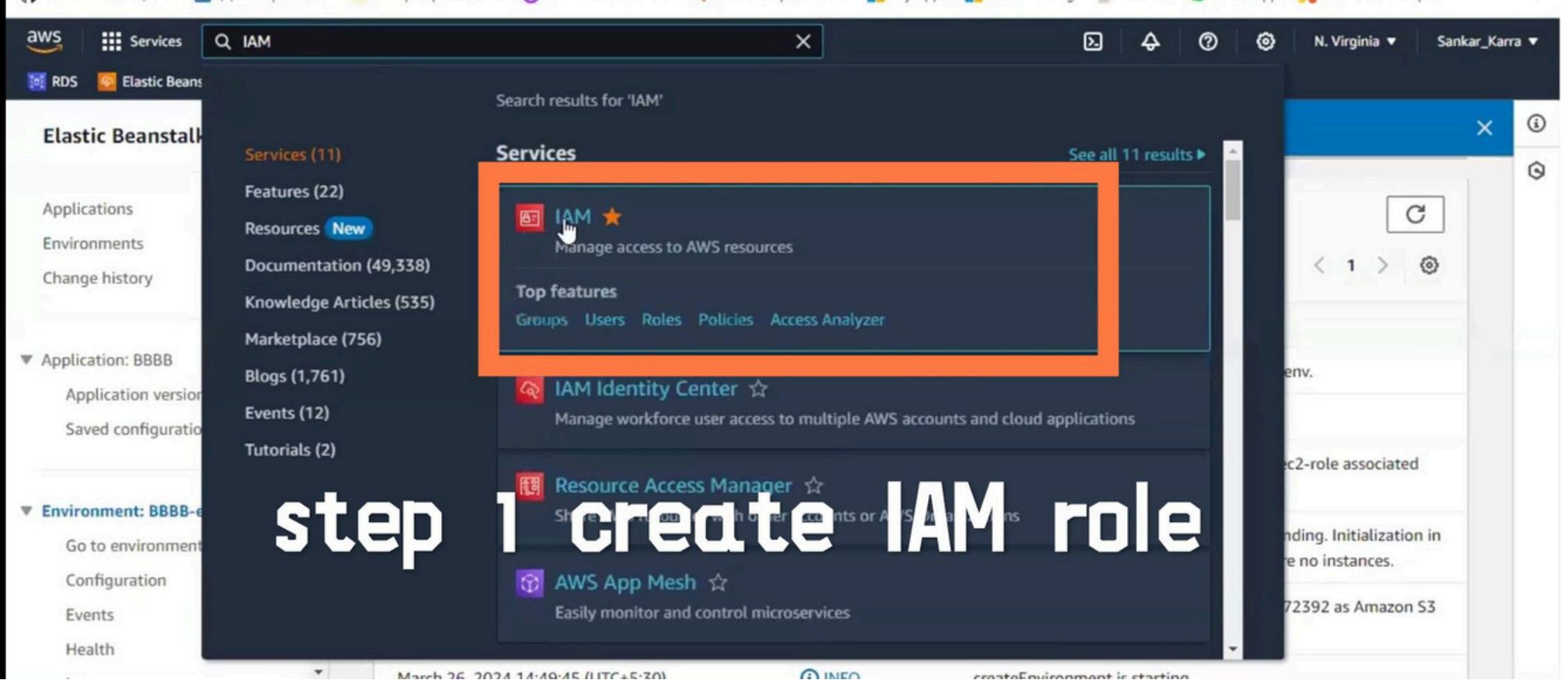
CloudShell Feedback Language © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

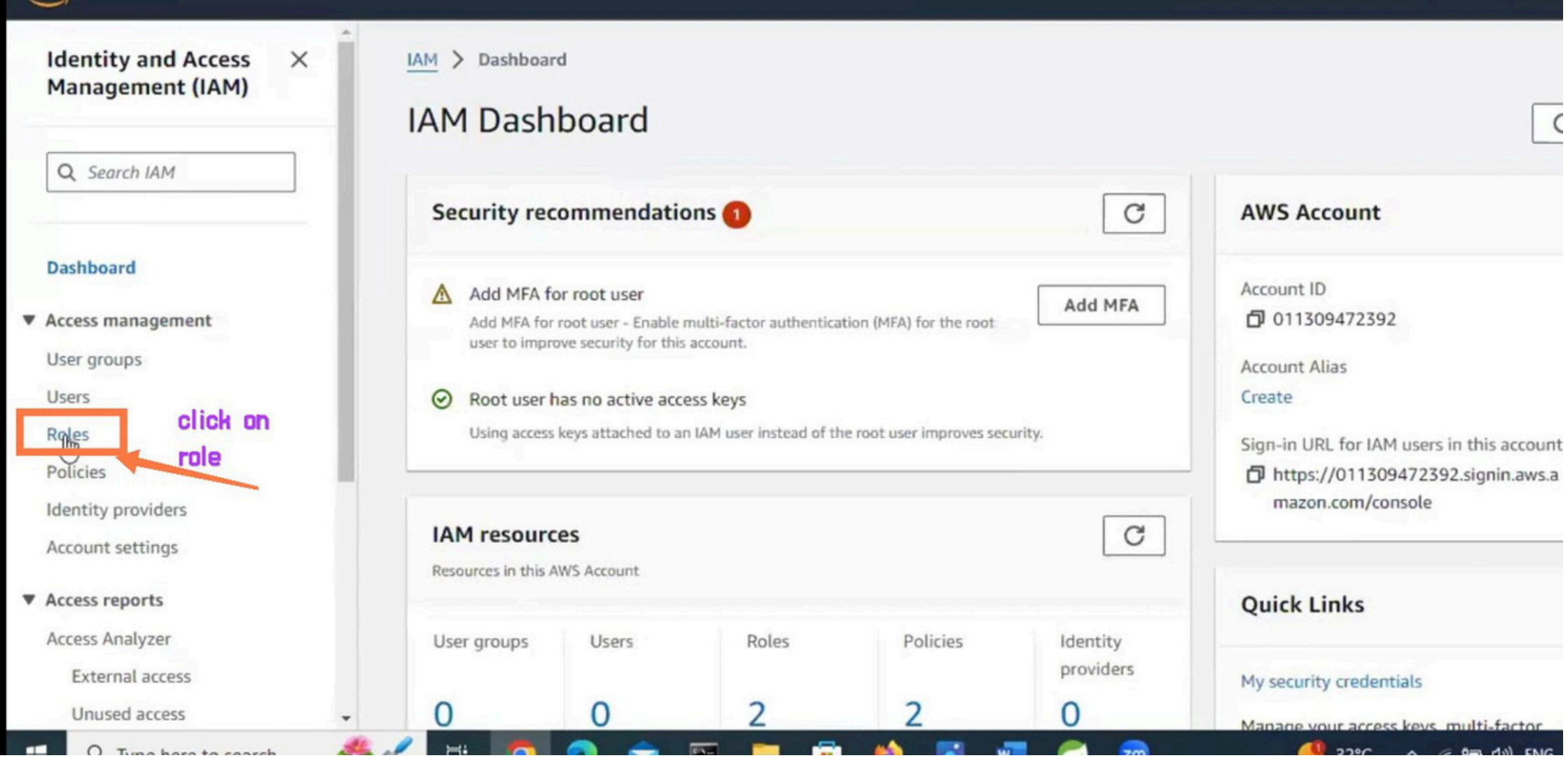
minutes

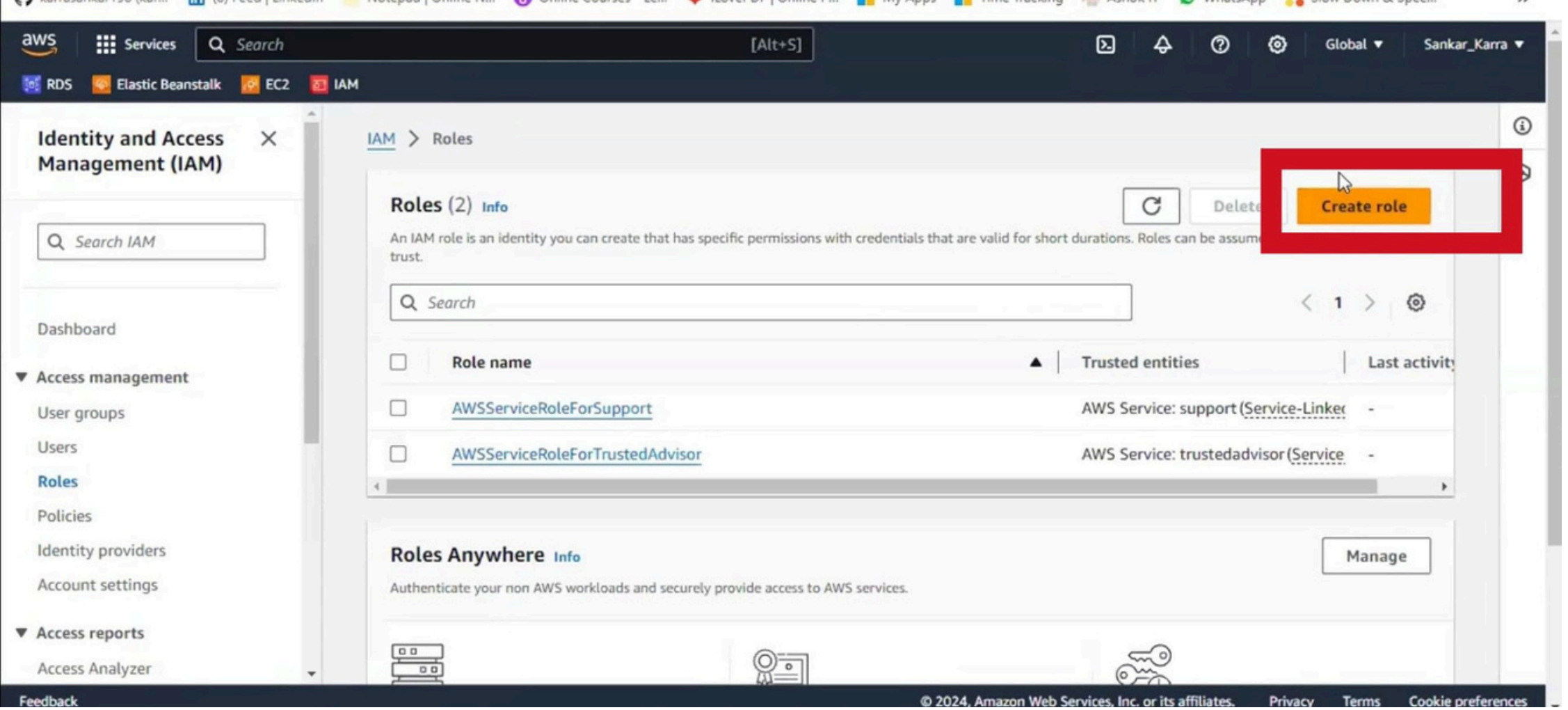
After 5 minutes you will get Notifications in your Email

Done

Create Beanstalk


aws
Services
IAM
N. Virginia
Sankar_Karra
RDS
Elastic Beans
Search results for 'IAM'
Elastic Beanstalk
Services (11)
Features (22)
Resources New
Documentation (49,338)
Knowledge Articles (535)
Marketplace (756)
Blogs (1,761)
Events (12)
Tutorials (2)
Services
See all 11 results
IAM
Manage access to AWS resources
Top features
Groups Users Roles Policies Access Analyzer
IAM Identity Center
Manage workforce user access to multiple AWS accounts and cloud applications
Resource Access Manager
AWS App Mesh
Easily monitor and control microservices
Applications
Environments
Change history
Application: BBBB
Application versio
Saved configuratio
Environment: BBBB-e
Go to environment
Configuration
Events
Health
step 1 create IAM role


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
click on role
IAM > Dashboard
IAM Dashboard
Security recommendations 1
Add MFA for root user
Add MFA for root user - Enable multi-factor authentication (MFA) for the root user to improve security for this account.
Add MFA
Root user has no active access keys
Using access keys attached to an IAM user instead of the root user improves security.
IAM resources
Resources in this AWS Account
User groups
0
Users
0
Roles
2
Policies
2
Identity providers
0
AWS Account
Account ID
011309472392
Account Alias
Create
Sign-in URL for IAM users in this account
https://011309472392.signin.aws.amazon.com/console
Quick Links
My security credentials


aws
Services
Search
[Alt+S]
Global
Sankar_Karra
RDS
Elastic Beanstalk
EC2
IAM
Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access Analyzer
IAM > Roles
Roles (2) Info
An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assume
trust.
Delete
Create role
Search
1
Role name
Trusted entities
Last activity
AWSServiceRoleForSupport
AWS Service: support (Service-Linke
AWSServiceRoleForTrustedAdvisor
AWS Service: trustedadvisor (Service
Roles Anywhere Info
Authenticate your non AWS workloads and securely provide access to AWS services.
Manage
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Step 1
Select trusted entity

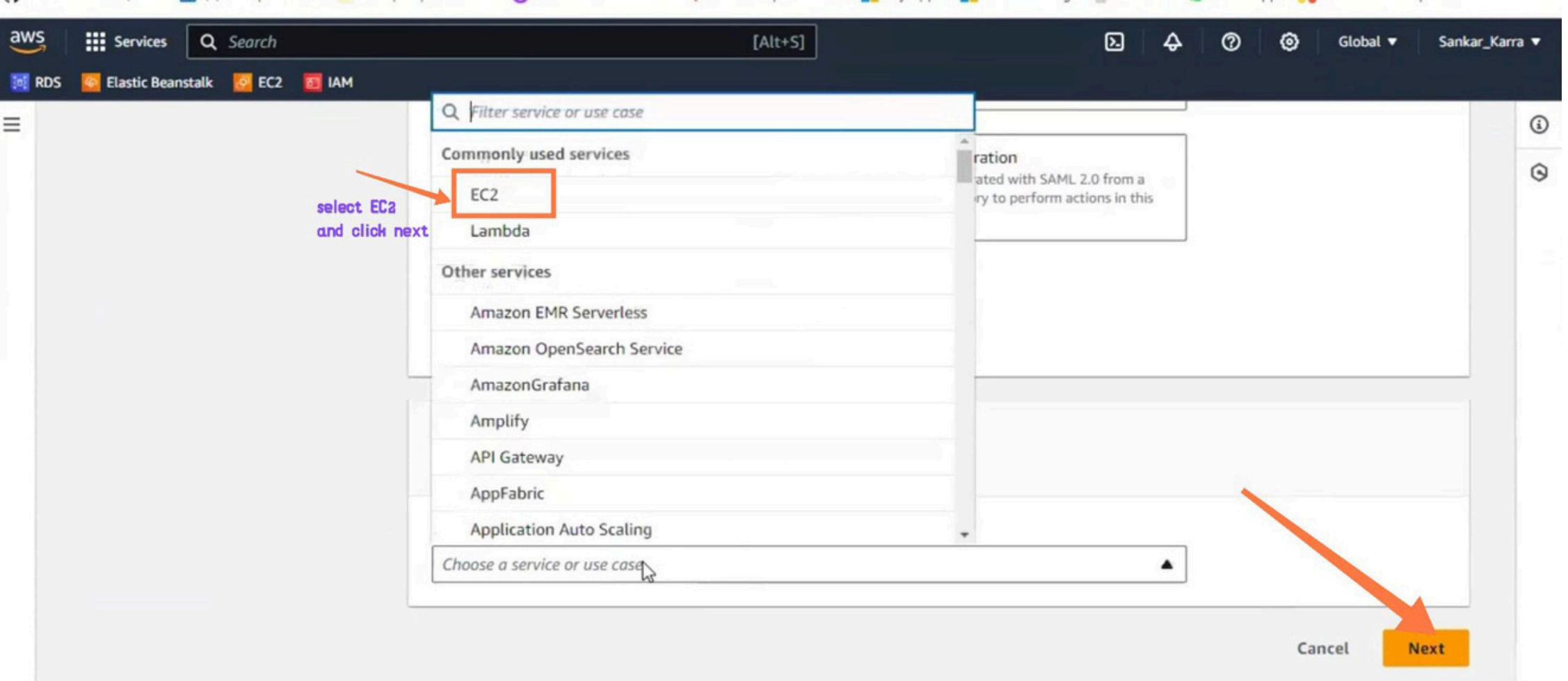
Step 2
Add permissions

Step 3
Name, review, and create

Select trusted entity Info

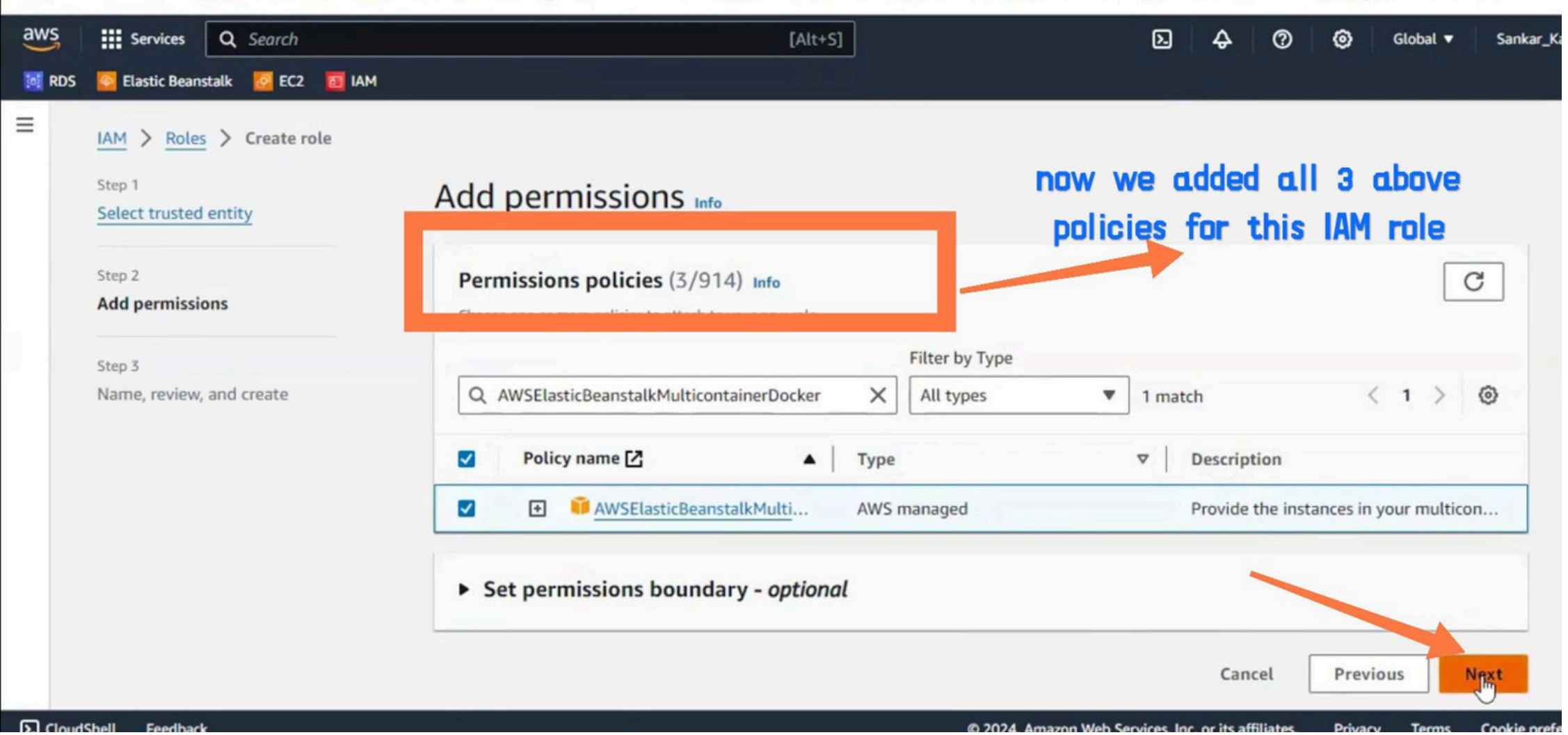
Trusted entity type

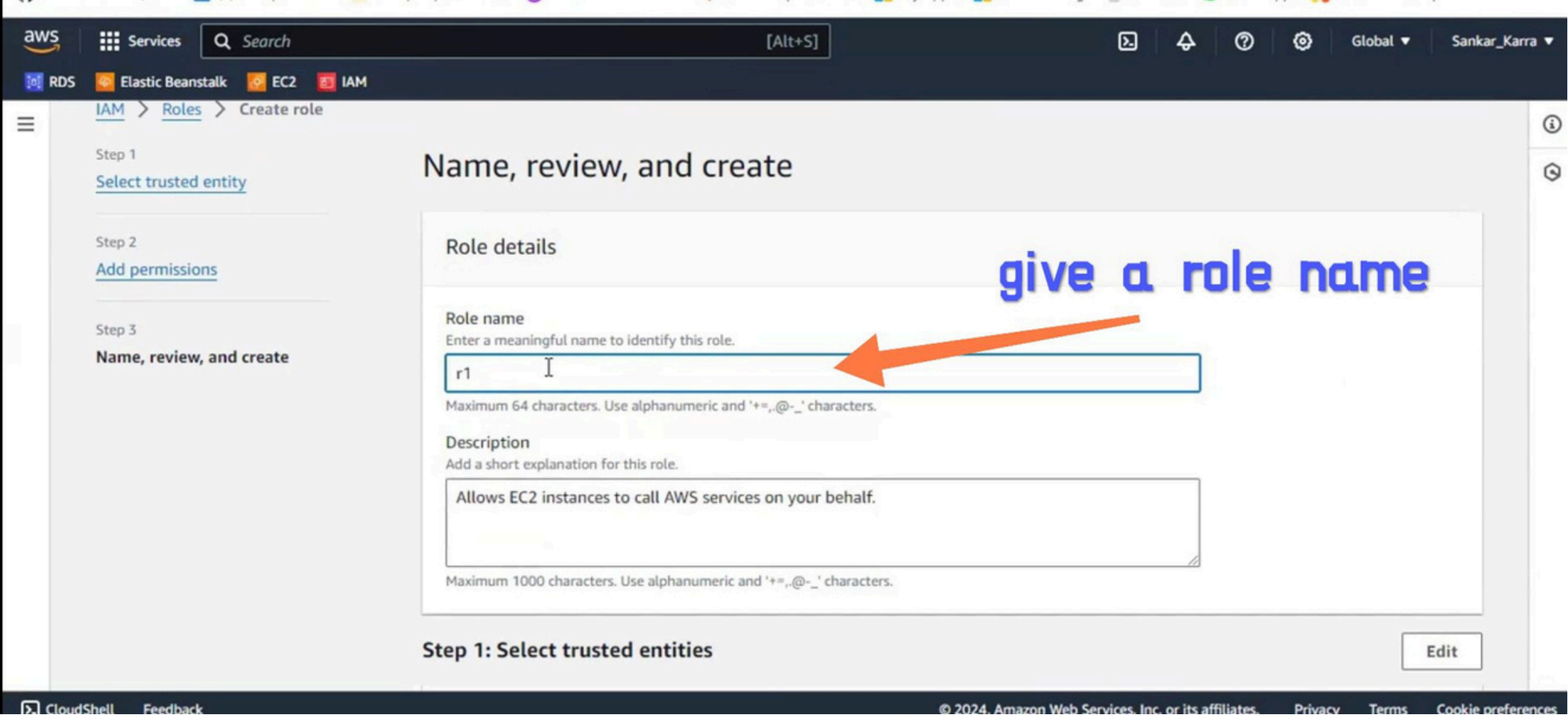
- (●) **AWS service**
  Allow AWS services like EC2, Lambda, or others to perform actions in this account.
- ( ) **AWS account**
  Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
- ( ) **Web identity**
  Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
- ( ) **SAML 2.0 federation**
  Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
- ( ) **Custom trust policy**
  Create a custom trust policy to enable others

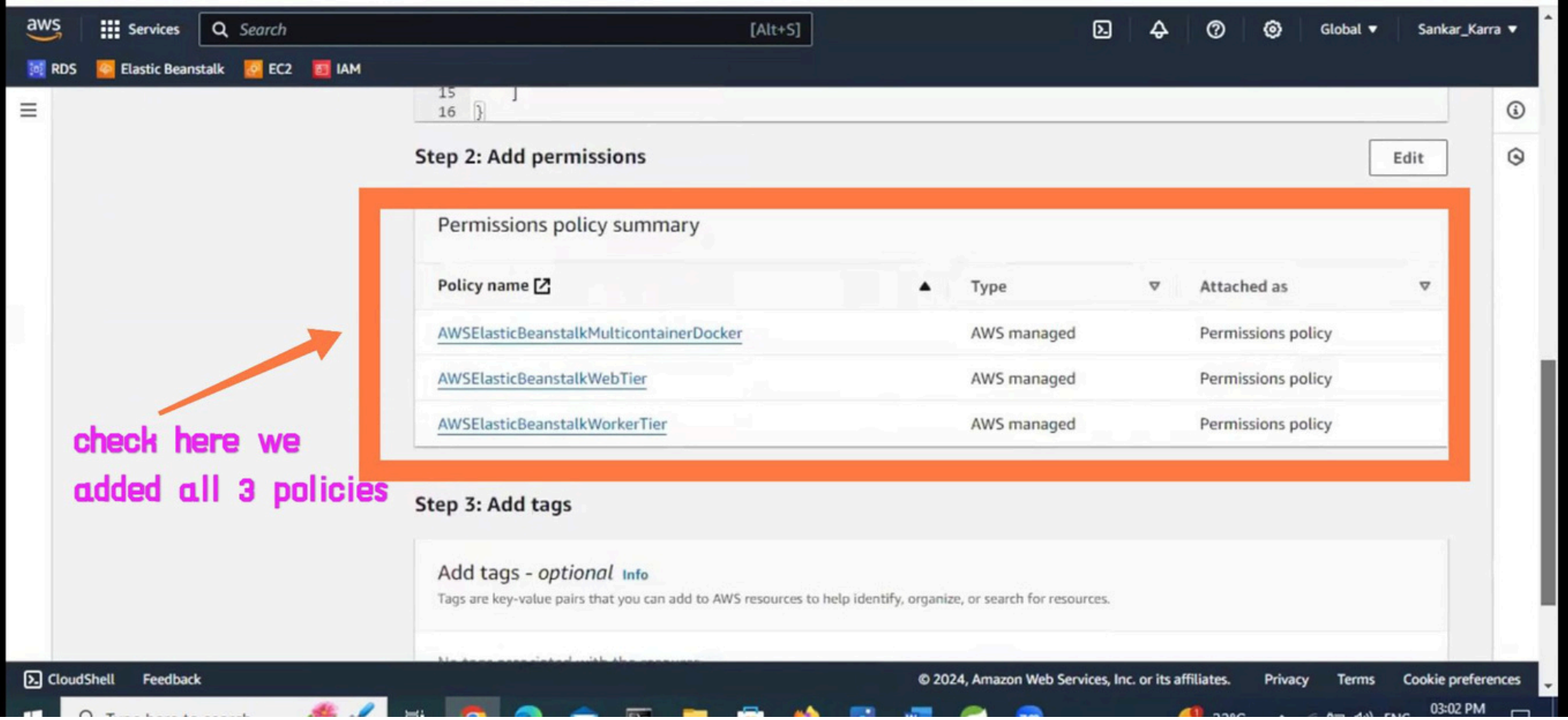

aws
Services
Search
[Alt+S]
Global
Sankar_Karra
RDS
Elastic Beanstalk
EC2
IAM
Filter service or use case
Commonly used services
EC2
Lambda
Other services
Amazon EMR Serverless
Amazon OpenSearch Service
AmazonGrafana
Amplify
API Gateway
AppFabric
Application Auto Scaling
select EC2
and click next
Choose a service or use case
Cancel
Next

- AWSElasticBeanstalkWebTier
- AWSElasticBeanstalkWorkerTier
- AWSElasticBeanstalkMulticontainerDocker

select these 3 services in below


aws
Services
Search
[Alt+S]
Global
Sankar_K
RDS
Elastic Beanstalk
EC2
IAM
IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Add permissions Info
now we added all 3 above policies for this IAM role
Permissions policies (3/914) Info
Filter by Type
AWSElasticBeanstalkMulticontainerDocker
All types
1 match
1
Policy name
Type
Description
AWSElasticBeanstalkMulti...
AWS managed
Provide the instances in your multicon...
Set permissions boundary - optional
Cancel
Previous
Next
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie prefe


aws
Services
Search
[Alt+S]
Global
Sankar_Karra
RDS
Elastic Beanstalk
EC2
IAM
IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Name, review, and create
Role details
give a role name
Role name
Enter a meaningful name to identify this role.
r1
Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.
Description
Add a short explanation for this role.
Allows EC2 instances to call AWS services on your behalf.
Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.
Step 1: Select trusted entities
Edit
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
Search
[Alt+S]
Global
Sankar_Karra
RDS
Elastic Beanstalk
EC2
IAM
15]
16 }
Step 2: Add permissions
Edit
Permissions policy summary
Policy name
Type
Attached as
AWSElasticBeanstalkMulticontainerDocker
AWS managed
Permissions policy
AWSElasticBeanstalkWebTier
AWS managed
Permissions policy
AWSElasticBeanstalkWorkerTier
AWS managed
Permissions policy
check here we
added all 3 policies
Step 3: Add tags
Add tags - optional Info
Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
03:02 PM

AWSElasticBeanstalkMulticontainerDocker	AWS managed	Permissions policy
AWSElasticBeanstalkWebTier	AWS managed	Permissions policy
AWSElasticBeanstalkWorkerTier	AWS managed	Permissions policy

Step 3: Add tags

Add tags - *optional* Info

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

Add new tag

You can add up to 50 more tags.

Cancel Previous Create role

aws | Services | Search [Alt+S] | Global | Sankar_Karra

RDS | Elastic Beanstalk | EC2 | IAM

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

User groups

Users

Roles

Policies

Identity providers

Account settings

Access reports

Access Analyzer

Role r1 created. View role

Search 1

Role name | Trusted entities | Last activity

Roles Anywhere Info

Manage

IAM role is created

Access AWS from your non AWS workloads

Operate your non AWS workloads using the same authentication and authorization strategy that you use within AWS.

X.509 Standard

Use your own existing PKI infrastructure or use AWS Certificate Manager Private Certificate Authority to authenticate identities.

Temporary credentials

Use temporary credentials with ease and benefit from the enhanced security they provide.

Now we need to create Elastic Beanstalk

aws | Services | Search [Alt+S] | Global | Sankar_Karra

RDS | Elastic Beanstalk | IAM

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

User groups

Users

Roles

Policies

Identity providers

Account settings

Access reports

Access Analyzer

Role r1 created. View role

1

Role name | Trusted entities | Last activity

click Elastic Beanstalk

Roles Anywhere

Manage

Authenticate your non AWS workloads and securely provide access to AWS services.

Access AWS from your non AWS workloads

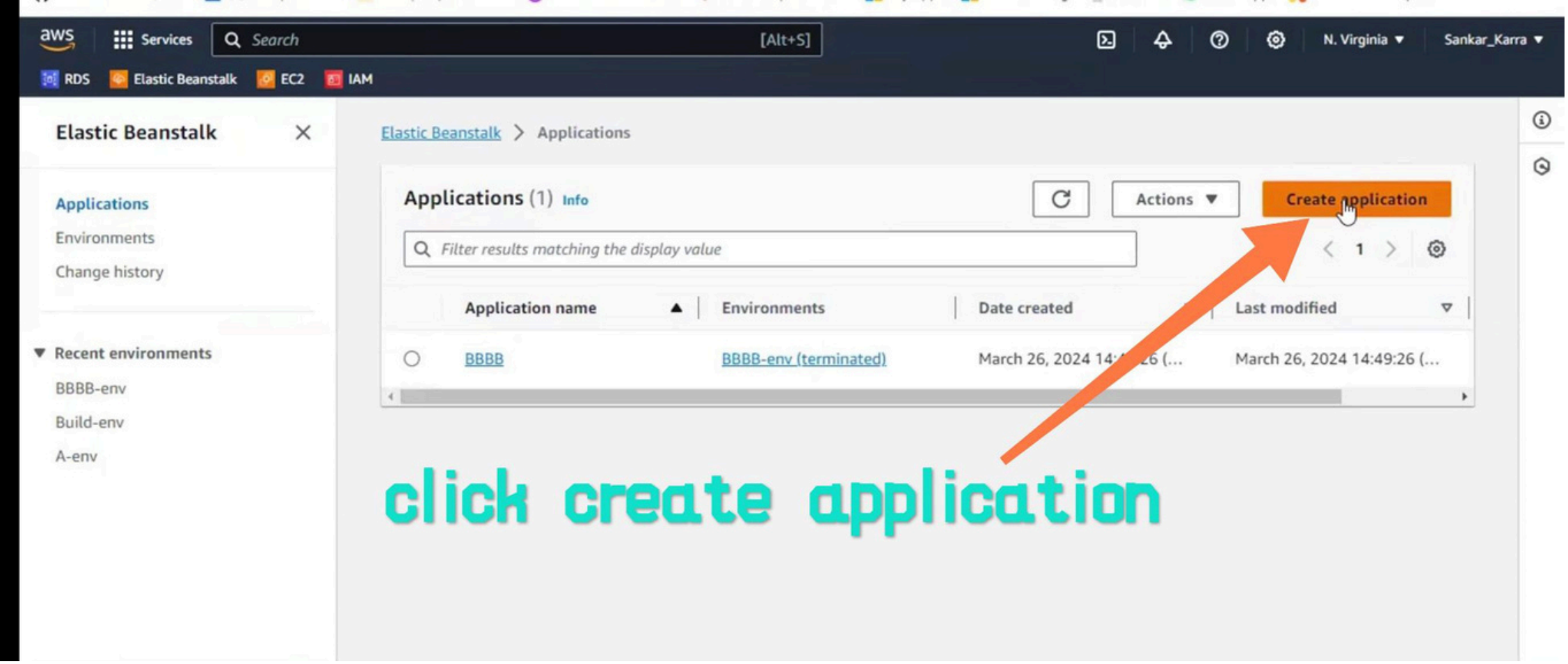
Operate your non AWS workloads using the same authentication and authorization strategy that you use within AWS.

X.509 Standard

Use your own existing PKI infrastructure or use AWS Certificate Manager Private Certificate Authority to authenticate identities.

Temporary credentials

Use temporary credentials with ease and benefit from the enhanced security they provide.


aws
Services
Search
[Alt+S]
N. Virginia
Sankar_Karra
RDS
Elastic Beanstalk
EC2
IAM
Elastic Beanstalk
Applications
Environments
Change history
Recent environments
BBBB-env
Build-env
A-env
Elastic Beanstalk > Applications
Applications (1) Info
Actions
Create application
Filter results matching the display value
1
Application name
Environments
Date created
Last modified
BBBB
BBBB-env (terminated)
March 26, 2024 14:49:26 (...
click create application

Application name

test12

Maximum length of 100 characters.

Description

give name

Tags

Apply up to 50 tags. You can use tags to group and filter your resources. A tag is a key-value pair. The key must be unique within the resource and is case-sensitive. Learn more

No tags associated with the resource.

Add new tag

You can add 50 more tags.

test14

Maximum length of 100 characters.

Description

Tags

Apply up to 50 tags. You can use tags to group and filter your resources. A tag is a key-value pair. The key must be unique within the resource and is case-sensitive. Learn more

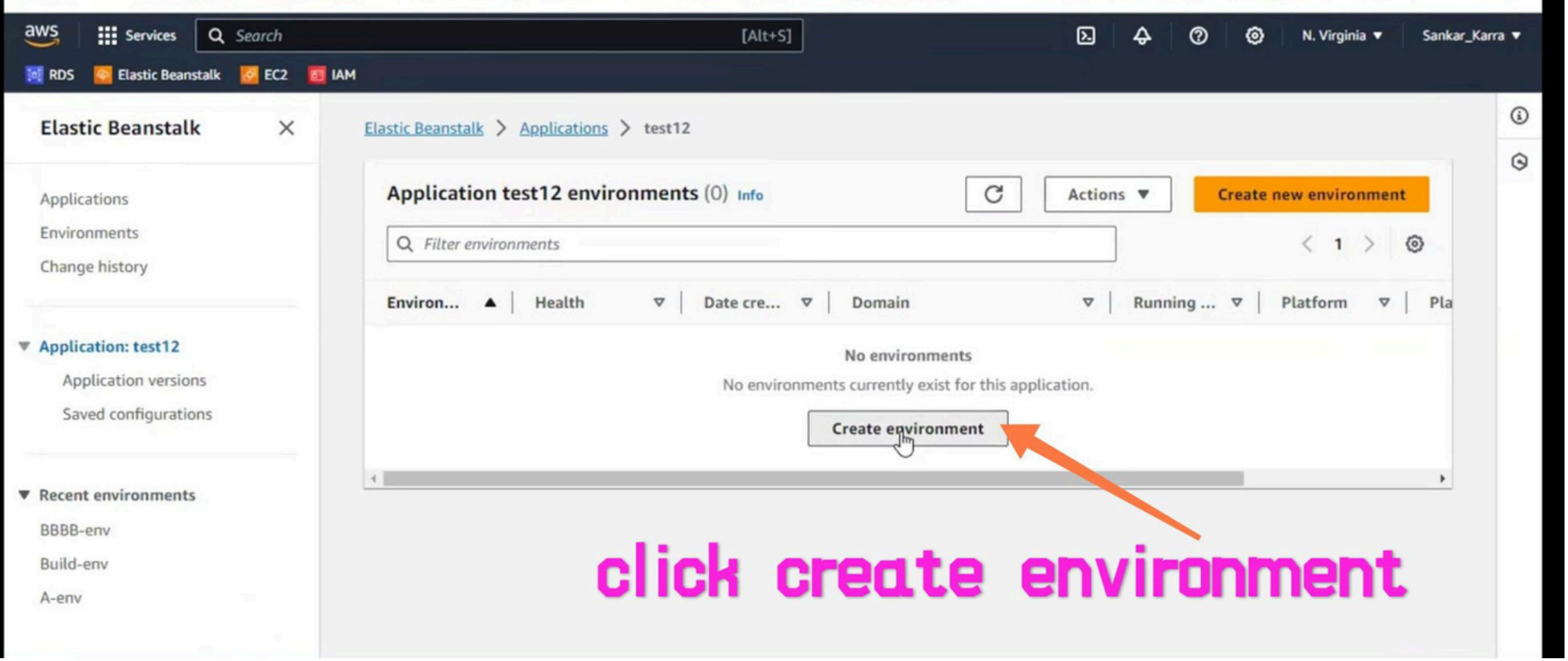
No tags associated with the resource.

Add new tag

You can add 50 more tags.

click create

Cancel Create


aws
Services
Search
[Alt+S]
N. Virginia
Sankar_Karra
RDS
Elastic Beanstalk
EC2
IAM
Elastic Beanstalk
Applications
Environments
Change history
Application: test12
Application versions
Saved configurations
Recent environments
BBBB-env
Build-env
A-env
Elastic Beanstalk > Applications > test12
Application test12 environments (0) Info
Actions
Create new environment
Filter environments
1
Environ...
Health
Date cre...
Domain
Running ...
Platform
No environments
No environments currently exist for this application.
Create environment
click create environment

Step 1
Configure environment

Step 2
Configure service access

Step 3 - *optional*
Set up networking, database, and tags

Step 4 - *optional*
Configure instance traffic and scaling

Step 5 - *optional*
Configure updates, monitoring, and logging

Step 6
Review

Configure environment Info

Environment tier Info

Amazon Elastic Beanstalk has two types of environment tiers to support different types of web applications.

- (•) **Web server environment**
  Run a website, web application, or web API that serves HTTP requests. Learn more
- ( ) **Worker environment**
  Run a worker application that processes long-running workloads on demand or performs tasks on a schedule. Learn more

Application information Info

Application name

test12

Maximum length of 100 characters.

Configure instance traffic and scaling

Step 5 - *optional*
Configure updates, monitoring, and logging

Step 6
Review

Application information Info

Application name

test12

Maximum length of 100 characters.

▶ Application tags (optional)

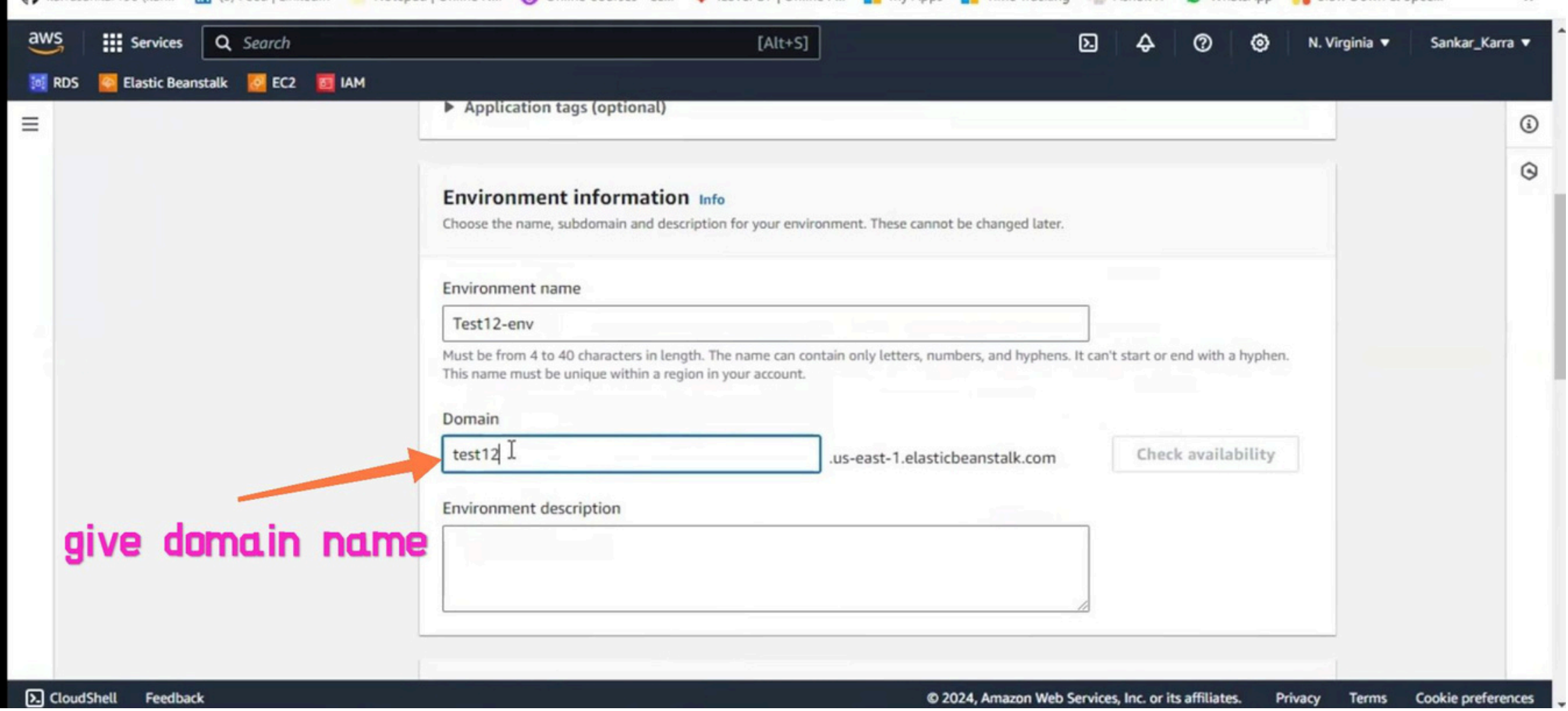
Environment information Info

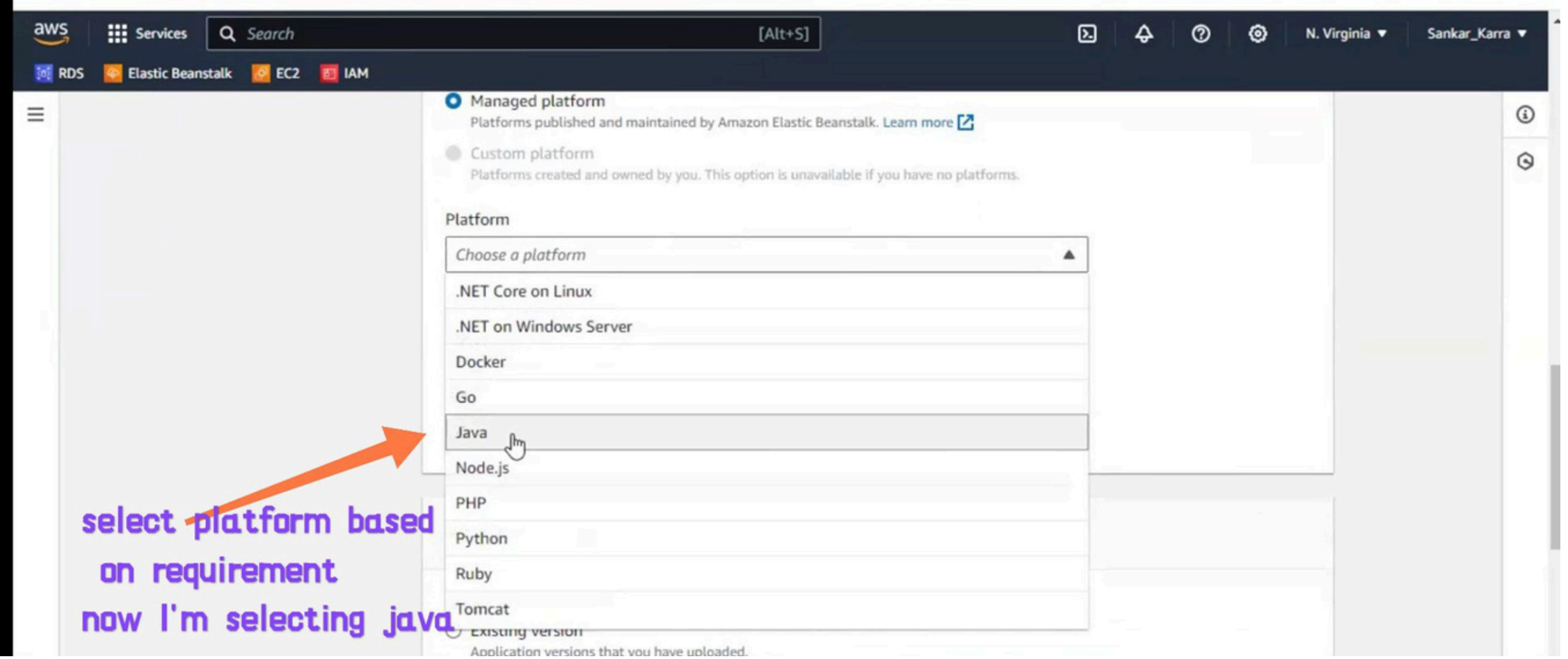
Choose the name, subdomain and description for your environment. These cannot be changed later.

Environment name

Test12-env

Must be from 4 to 40 characters in length. The name can contain only letters, numbers, and hyphens. It can't start or end with a hyphen. This name must be unique within a region in your account.


aws
Services
Search
[Alt+S]
N. Virginia
Sankar_Karra
RDS
Elastic Beanstalk
EC2
IAM
Application tags (optional)
Environment information
Info
Choose the name, subdomain and description for your environment. These cannot be changed later.
Environment name
Test12-env
Must be from 4 to 40 characters in length. The name can contain only letters, numbers, and hyphens. It can't start or end with a hyphen. This name must be unique within a region in your account.
Domain
test12
.us-east-1.elasticbeanstalk.com
Check availability
Environment description
give domain name
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
Search
[Alt+S]
N. Virginia
Sankar_Karra
RDS
Elastic Beanstalk
EC2
IAM
Managed platform
Platforms published and maintained by Amazon Elastic Beanstalk. Learn more
Custom platform
Platforms created and owned by you. This option is unavailable if you have no platforms.
Platform
Choose a platform
.NET Core on Linux
.NET on Windows Server
Docker
Go
Java
Node.js
PHP
Python
Ruby
Tomcat
Application versions that you have uploaded.
select platform based
on requirement
now I'm selecting java

select sample application
if you have file of code
the upload your code

- (•) Sample application
- ( ) Existing version
  Application versions that you have uploaded.
- ( ) Upload your code
  Upload a source bundle from your computer or copy one from Amazon S3.

Presets Info

Start from a preset that matches your use case or choose custom configuration to unset recommended values and use the service's default values.

Configuration presets

- (•) Single instance (free tier eligible)
- ( ) Single instance (using spot instance)
- ( ) High availability
- ( ) High availability (using spot and on-demand instances)
- ( ) Custom configuration

Cancel Next

- Sample application
- Existing version
  Application versions that you have uploaded.
- Upload your code
  Upload a source bundle from your computer or copy one from Amazon S3.

Presets Info

Start from a preset that matches your use case or choose custom configuration to unset recommended values and use the service's default values.

Configuration presets

- Single instance (free tier eligible)
- Single instance (using spot instance)
- High availability
- High availability (using spot and on-demand instances)
- Custom configuration

Cancel Next

Set up networking, database, and tags

Step 4 - *optional*
Configure instance traffic and scaling

Step 5 - *optional*
Configure updates, monitoring, and logging

Step 6
Review

Service role

Create and use new service role

Use an existing service role

select existing role and IAM role which we created previously

Existing service roles

Choose an existing IAM role for Elastic Beanstalk to assume as a service role. The existing IAM role must have the required IAM managed policies.

EC2 key pair

Select an EC2 key pair to securely log in to your EC2 instances. Learn more

Choose a key pair

EC2 instance profile

Choose an IAM instance profile with managed policies that allow your EC2 instances to perform required operations.

View permission details

Cancel Skip to review Previous Next

Configure updates, monitoring, and logging

Step 6
Review

View permission details

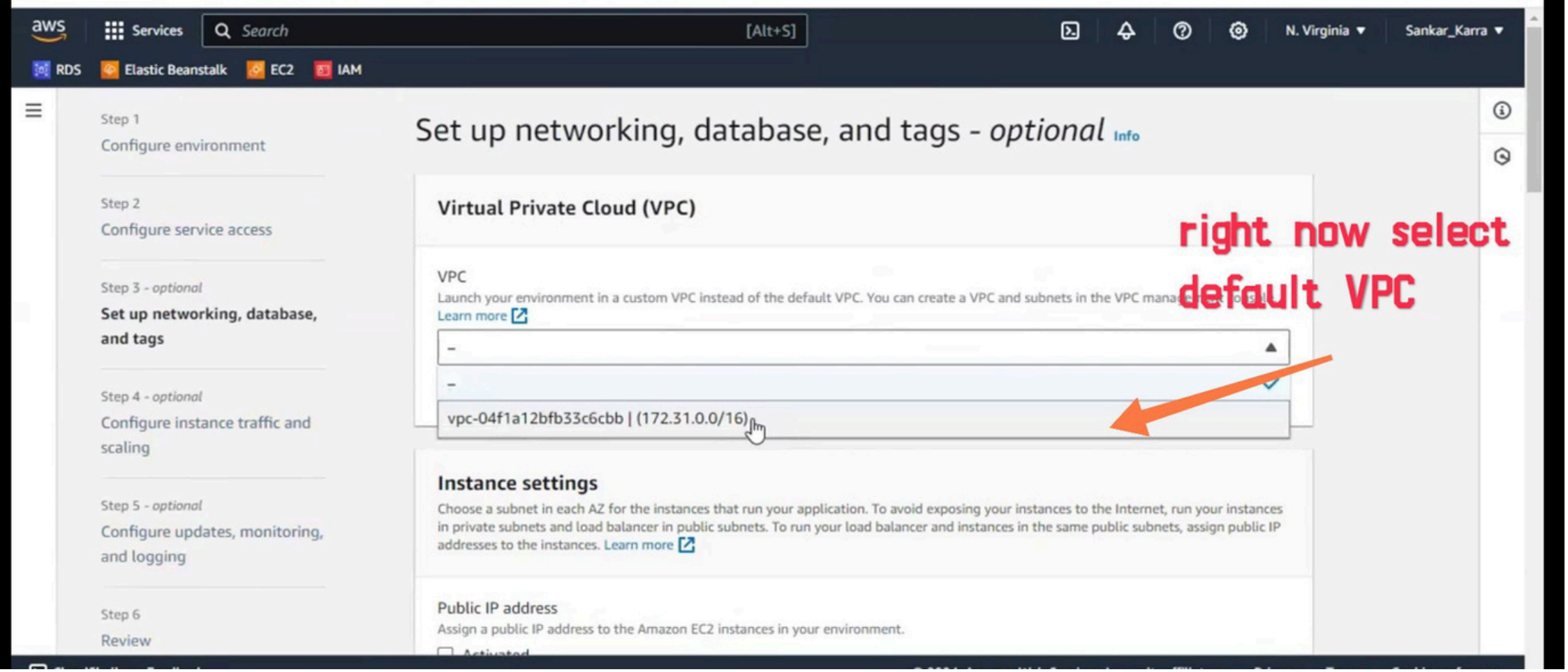
EC2 key pair
Select an EC2 key pair to securely log in to your EC2 instances. Learn more
Choose a key pair

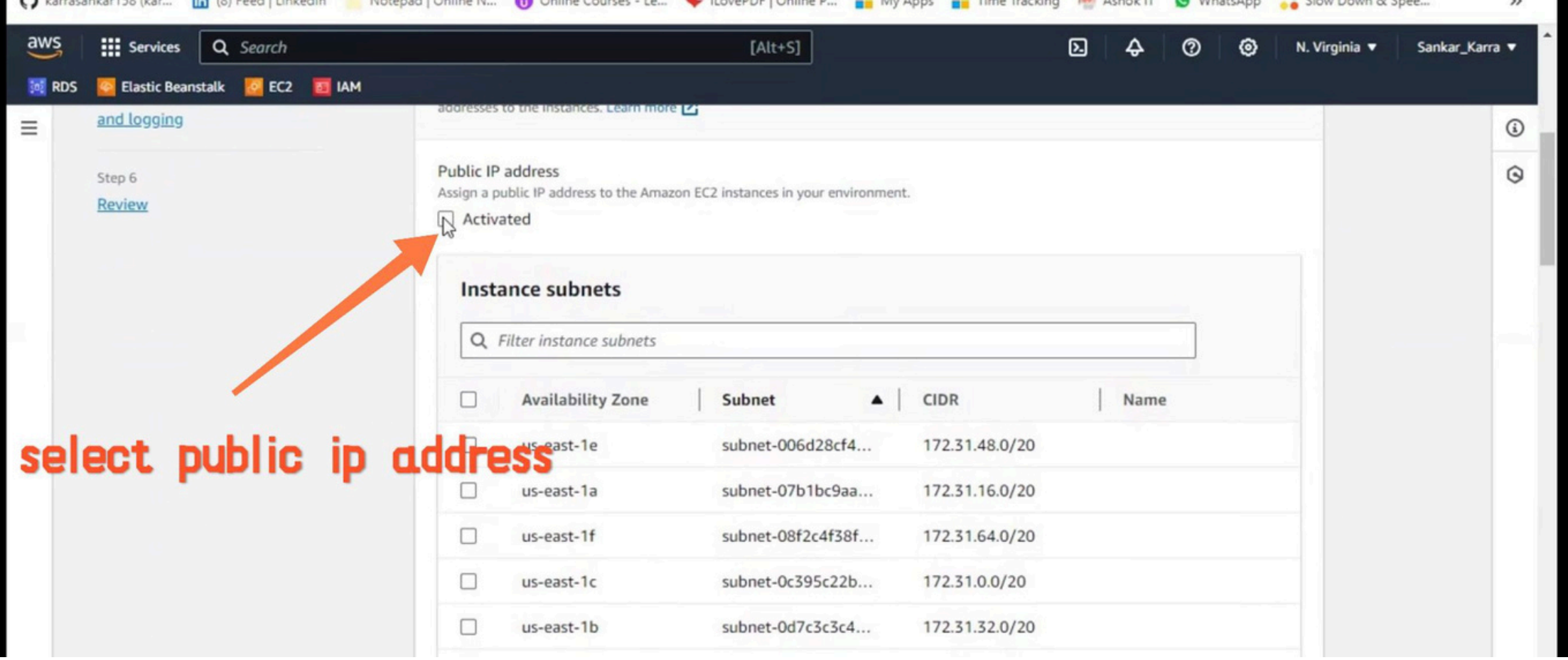
EC2 instance profile
Choose an IAM instance profile with managed policies that allow your EC2 instances to perform required operations.
r1

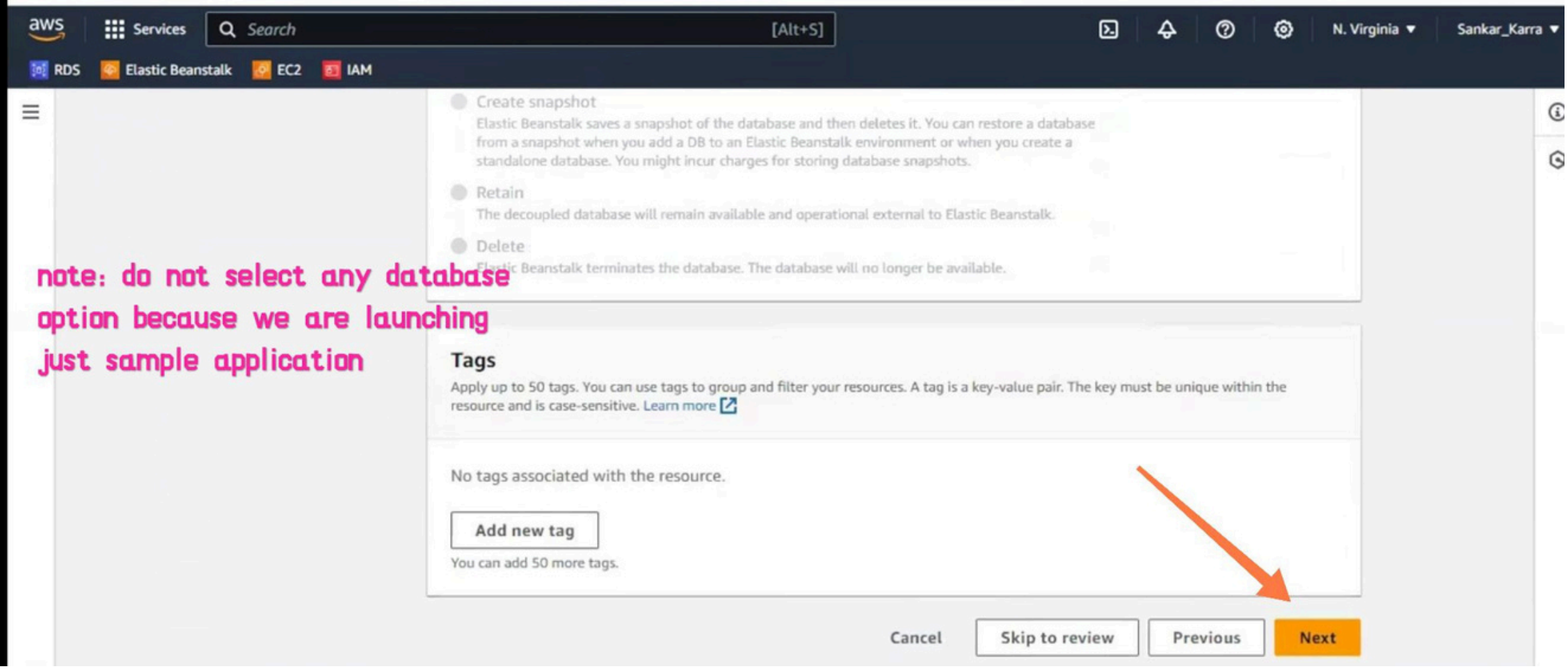
View permission details

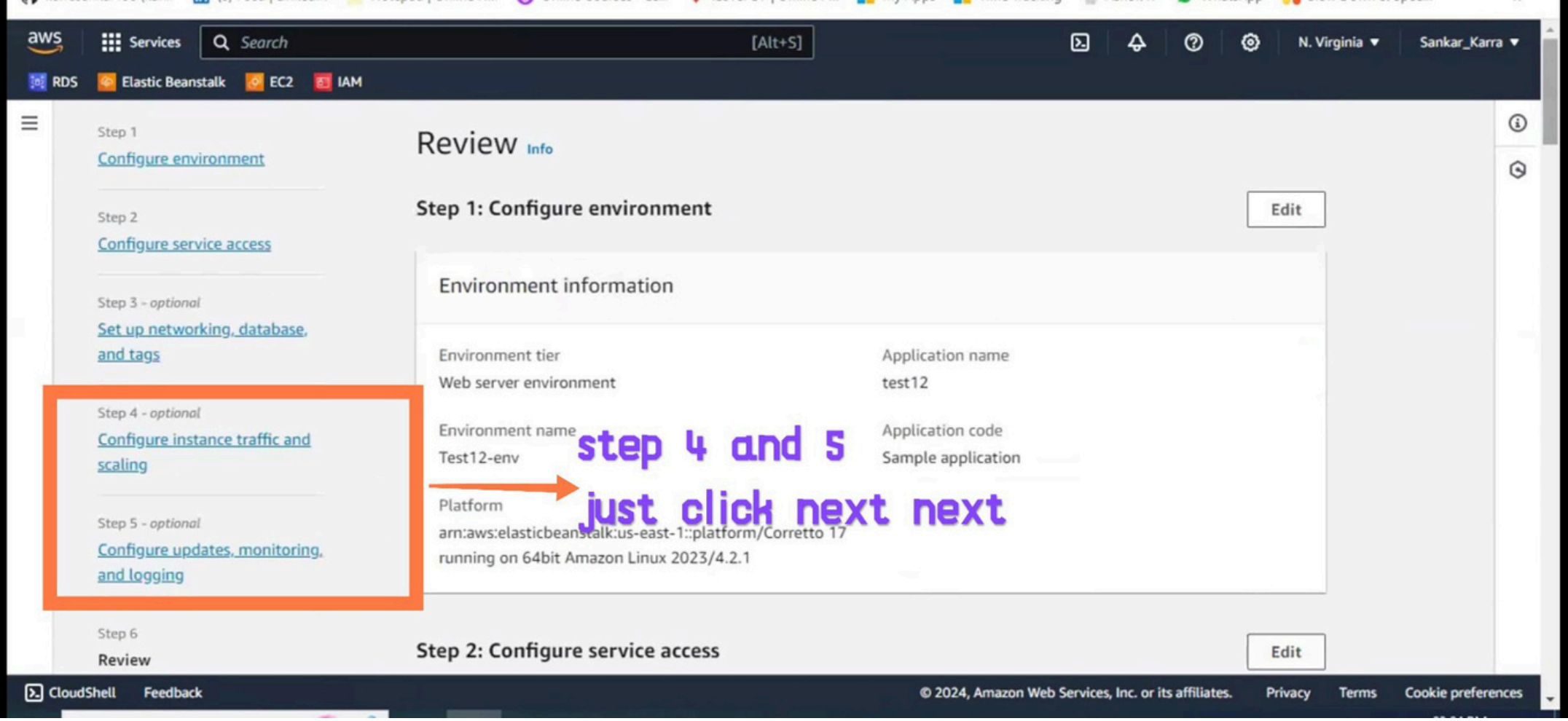
select EC2 instance profile
in that select role which we created
previously

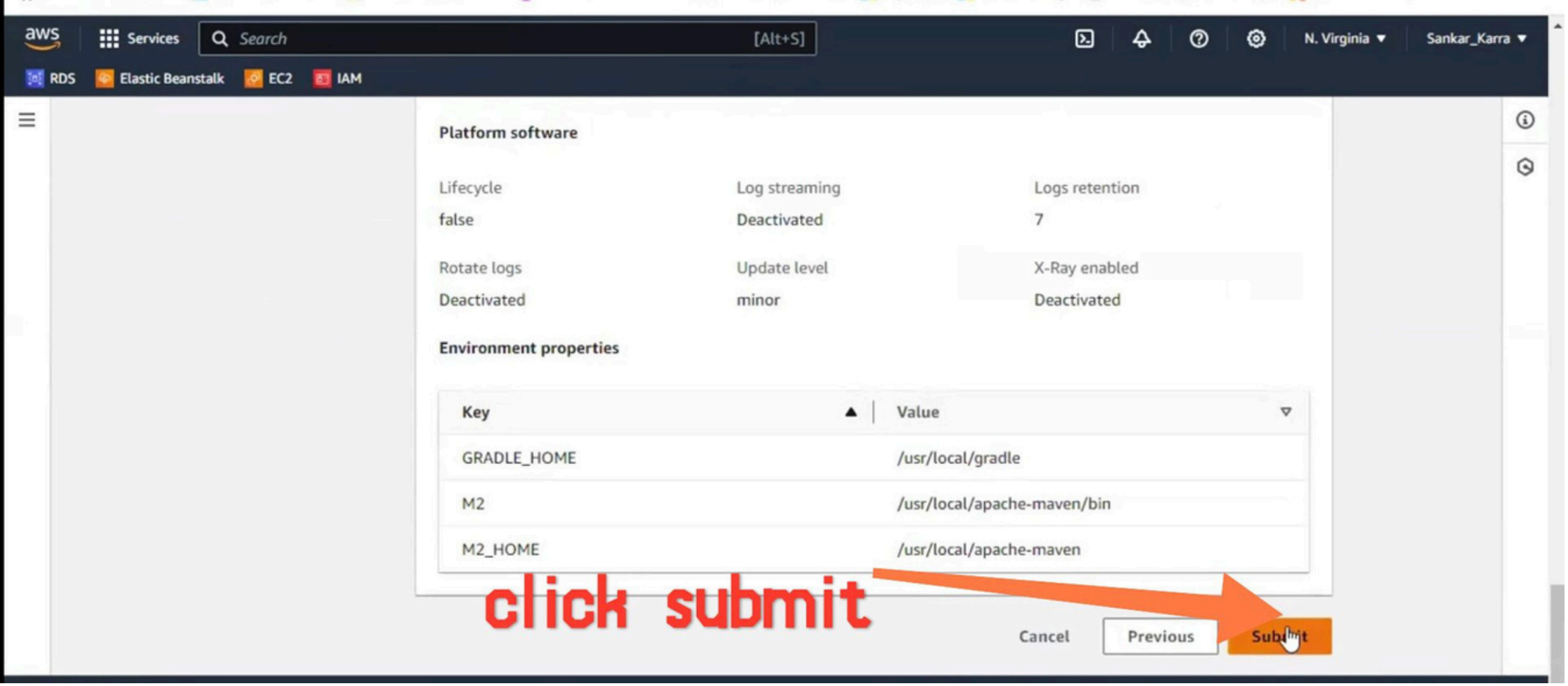
Cancel | Skip to review | Previous | Next


aws
Services
Search
[Alt+S]
N. Virginia
Sankar_Karra
RDS
Elastic Beanstalk
EC2
IAM
Step 1
Configure environment
Step 2
Configure service access
Step 3 - optional
Set up networking, database, and tags
Step 4 - optional
Configure instance traffic and scaling
Step 5 - optional
Configure updates, monitoring, and logging
Step 6
Review
Set up networking, database, and tags - optional Info
Virtual Private Cloud (VPC)
VPC
Launch your environment in a custom VPC instead of the default VPC. You can create a VPC and subnets in the VPC manag
Learn more
-
-
vpc-04f1a12bfb33c6cbb | (172.31.0.0/16)
right now select
default VPC
Instance settings
Choose a subnet in each AZ for the instances that run your application. To avoid exposing your instances to the Internet, run your instances in private subnets and load balancer in public subnets. To run your load balancer and instances in the same public subnets, assign public IP addresses to the instances. Learn more
Public IP address
Assign a public IP address to the Amazon EC2 instances in your environment.


aws
Services
Search
[Alt+S]
N. Virginia
Sankar_Karra
RDS
Elastic Beanstalk
EC2
IAM
and logging
Step 6
Review
addresses to the instances. Learn more
Public IP address
Assign a public IP address to the Amazon EC2 instances in your environment.
Activated
Instance subnets
Filter instance subnets
Availability Zone
Subnet
CIDR
Name
us-east-1e
subnet-006d28cf4...
172.31.48.0/20
us-east-1a
subnet-07b1bc9aa...
172.31.16.0/20
us-east-1f
subnet-08f2c4f38f...
172.31.64.0/20
us-east-1c
subnet-0c395c22b...
172.31.0.0/20
us-east-1b
subnet-0d7c3c3c4...
172.31.32.0/20
select public ip address


aws
Services
Search
[Alt+S]
N. Virginia
Sankar_Karra
RDS
Elastic Beanstalk
EC2
IAM
Create snapshot
Elastic Beanstalk saves a snapshot of the database and then deletes it. You can restore a database from a snapshot when you add a DB to an Elastic Beanstalk environment or when you create a standalone database. You might incur charges for storing database snapshots.
Retain
The decoupled database will remain available and operational external to Elastic Beanstalk.
Delete
Elastic Beanstalk terminates the database. The database will no longer be available.
note: do not select any database option because we are launching just sample application
Tags
Apply up to 50 tags. You can use tags to group and filter your resources. A tag is a key-value pair. The key must be unique within the resource and is case-sensitive. Learn more
No tags associated with the resource.
Add new tag
You can add 50 more tags.
Cancel
Skip to review
Previous
Next


aws
Services
Search
[Alt+S]
N. Virginia
Sankar_Karra
RDS
Elastic Beanstalk
EC2
IAM
Step 1
Configure environment
Step 2
Configure service access
Step 3 - optional
Set up networking, database, and tags
Step 4 - optional
Configure instance traffic and scaling
Step 5 - optional
Configure updates, monitoring, and logging
Step 6
Review
Review Info
Step 1: Configure environment
Edit
Environment information
Environment tier
Web server environment
Application name
test12
Environment name
Test12-env
Application code
Sample application
Platform
arn:aws:elasticbeanstalk:us-east-1::platform/Corretto 17
running on 64bit Amazon Linux 2023/4.2.1
step 4 and 5
just click next next
Step 2: Configure service access
Edit
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
Search
[Alt+S]
N. Virginia
Sankar_Karra
RDS
Elastic Beanstalk
EC2
IAM
Platform software
Lifecycle
false
Log streaming
Deactivated
Logs retention
7
Rotate logs
Deactivated
Update level
minor
X-Ray enabled
Deactivated
Environment properties
Key
Value
GRADLE_HOME
/usr/local/gradle
M2
/usr/local/apache-maven/bin
M2_HOME
/usr/local/apache-maven
click submit
Cancel
Previous
Submit

aws Services Search [Alt+S] N. Virginia Sankar_Karra

RDS Elastic Beanstalk EC2 IAM

Elastic Beanstalk

Applications
Environments
Change history

Application: test12
Application versions
Saved configurations

Environment: Test12-env
Go to environment
Configuration
Events
Health

Elastic Beanstalk is launching your environment. This will take a few minutes.

check here how Beanstalk configuring

Platform state
Supported

Events Health Logs Monitoring Alarms Managed updates Tags

Events (2) Info

Filter events by text, property or value

1

Time	Type	D
March 26, 2024 15:04:57 (UTC+5:30)	INFO	Using elasticbeanstalk-us-east-1-011309472392 as Amazon S3 storage bucket for environment data.
March 26, 2024 15:04:56 (UTC+5:30)	INFO	createEnvironment is starting.

aws | Services | Search [Alt+S] | N. Virginia | Sankar_Karra

RDS | Elastic Beanstalk | EC2 | IAM

Elastic Beanstalk

- Applications
- Environments
- Change history

- Application: test12
  - Application versions
  - Saved configurations

- **Environment: Test12-env**
  - Go to environment
  - Configuration
  - Events
  - Health

Environment successfully launched.

Test12-env Info

Actions | Upload and deploy

Environment overview

Health
Ok

Environment ID
e-zc4wjuedxh

Domain
test12.us-east-1.elasticbeanstalk.com

Application name
test12

Platform

Change version

Platform
Corretto 17 running on 64bit Amazon Linux 2023/4.2.1

Running version
-

Platform state
Supported

successfully launched environment or created beanstalk

Events | Health | Logs | Monitoring | Alarms | Managed updates | Tags