G. Nandini
20A31A05D6

PRACTICAL IMPLEMENTATION

OF AWS SERVICES

CONTENTS

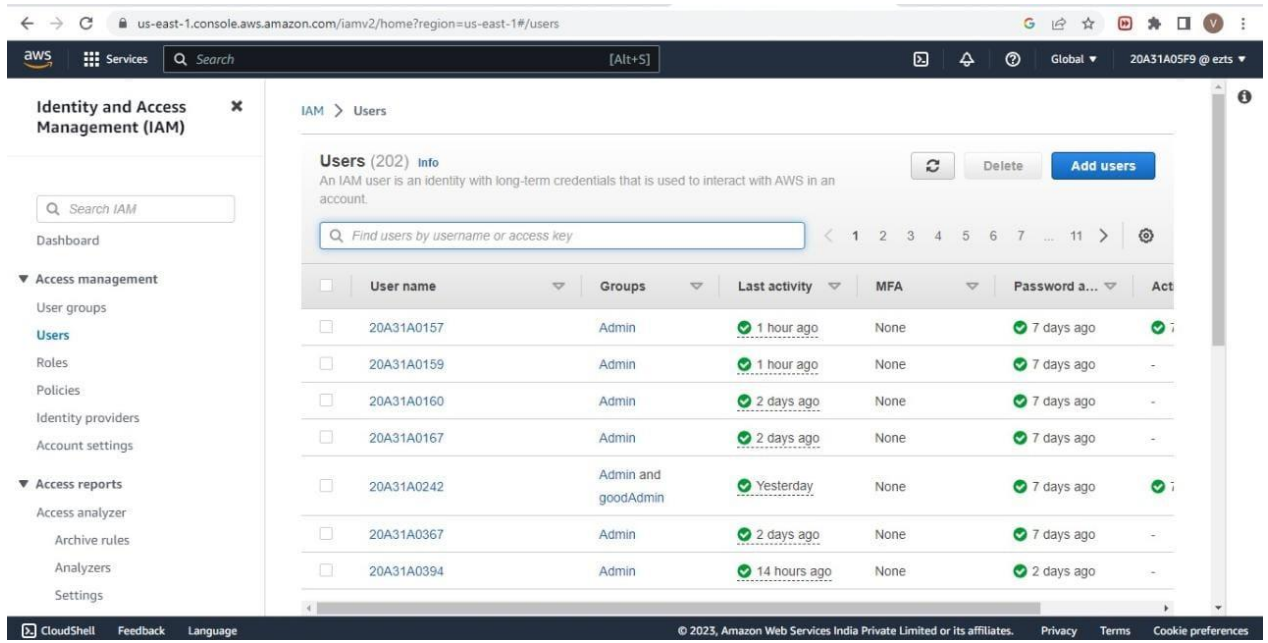
AWS COMMAND LINE INTERFACE

Lab - Configuring AWS CLI

STEP 1 - Download and install AWS CLI and complete the installation steps

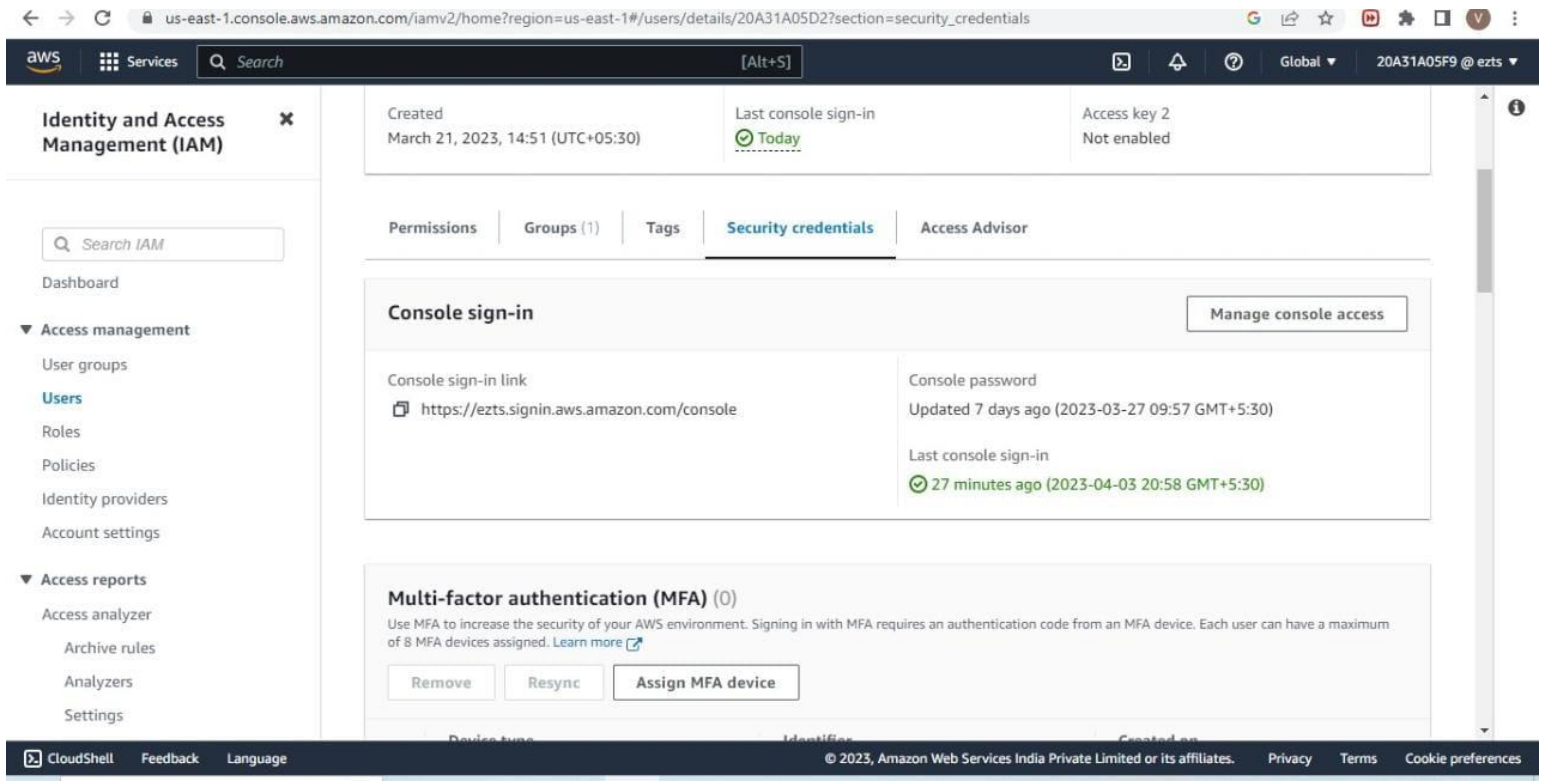
STEP 2 - Login to AWS Management Console and search for IAM.

STEP 3 - In the navigation pane ,select Users

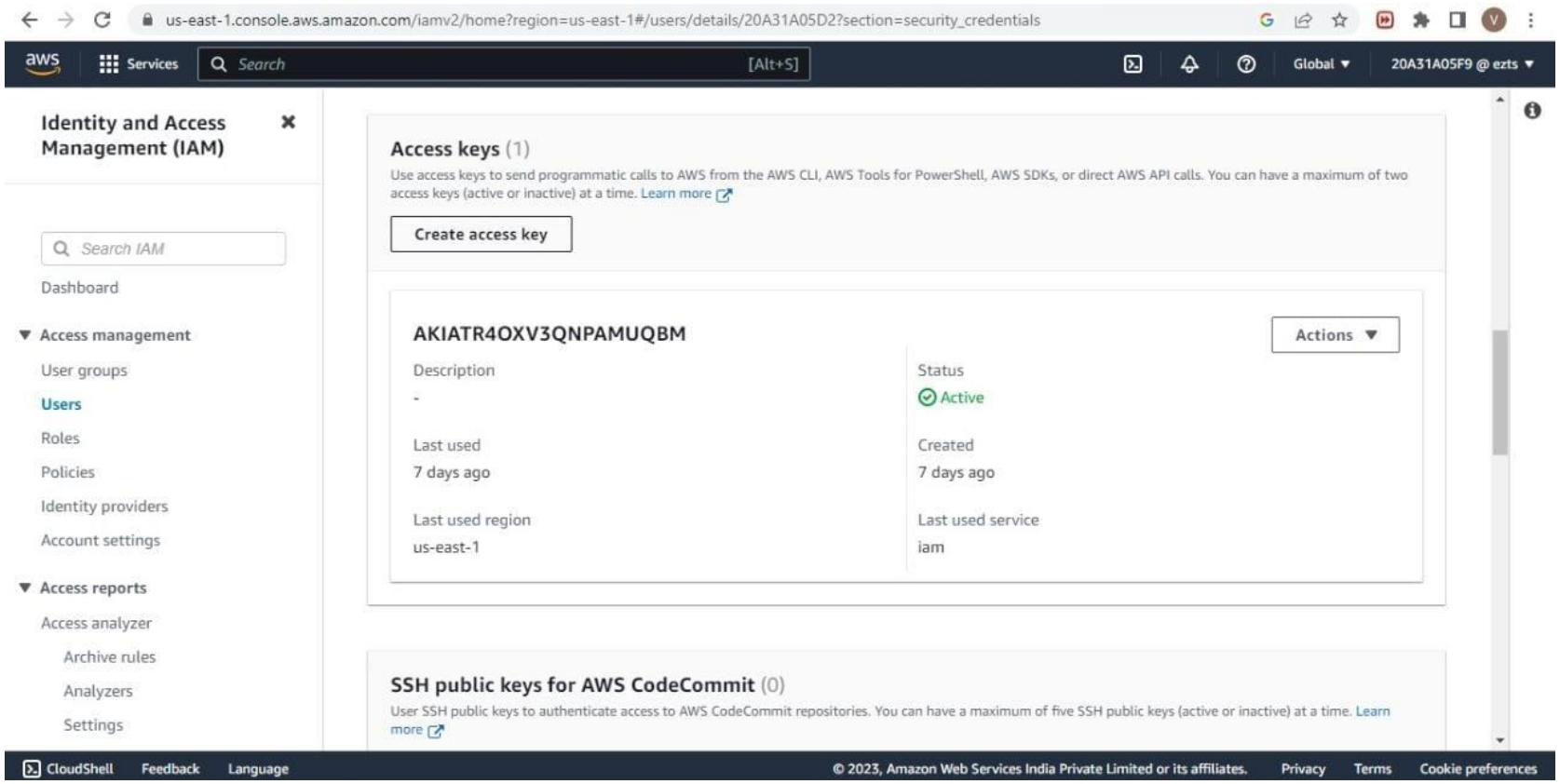


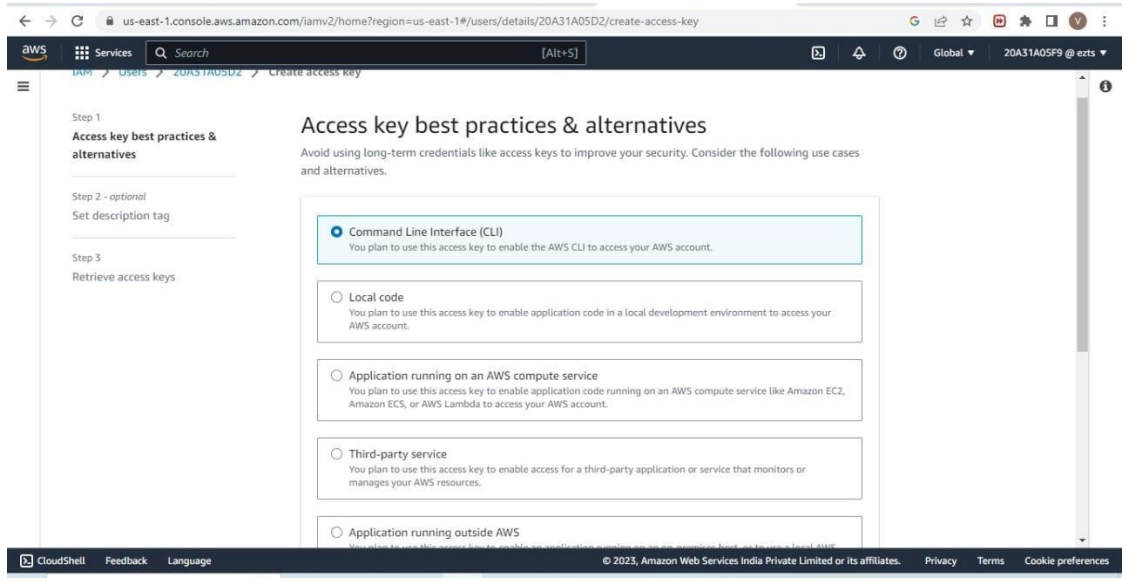
STEP 4 - In the users select the name of the user whose access keys you want to create.

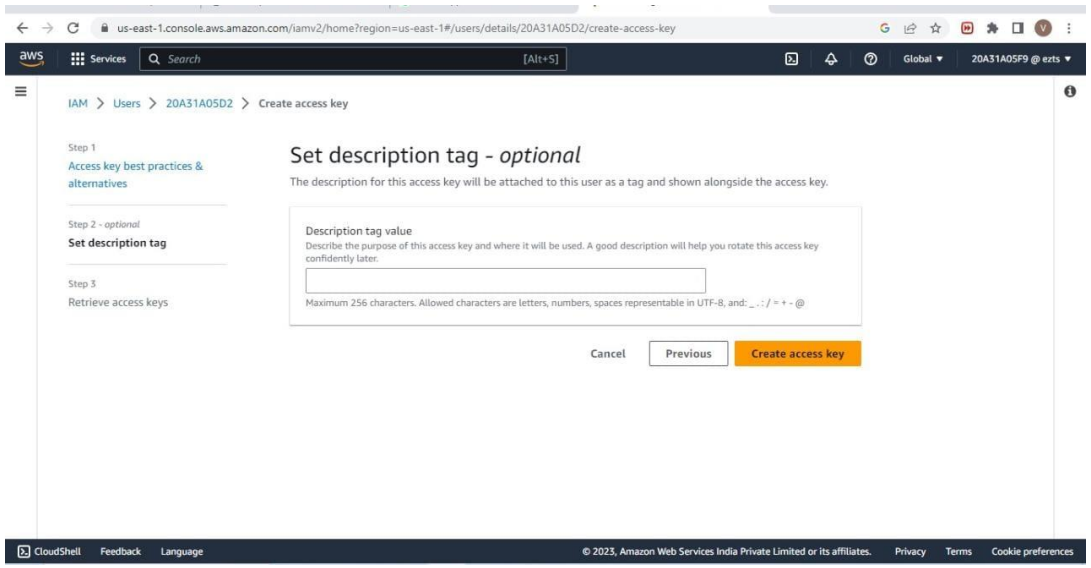
STEP 5 - Click on Security Credentials tab.

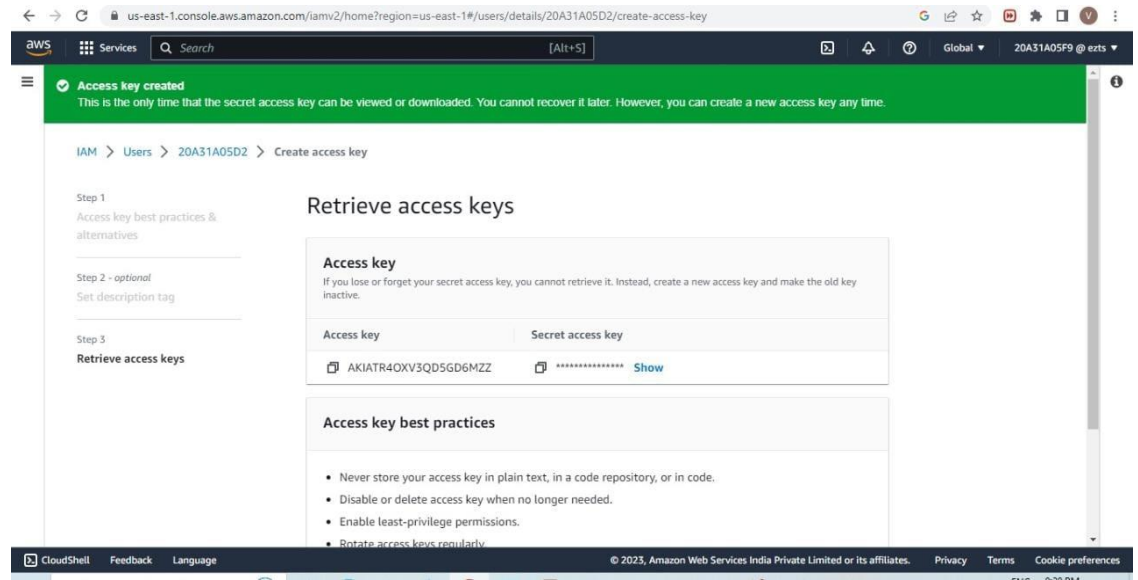


STEP 6 - In the access Keys section , choose Create access key -> Command Line Interface -> Create Access Key




us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/users/details/20A31A05D2/create-access-key
Services
Search
[Alt+S]
Global
20A31A05F9 @ ezts
Step 1
Access key best practices & alternatives
Step 2 - optional
Set description tag
Step 3
Retrieve access keys
Access key best practices & alternatives
Avoid using long-term credentials like access keys to improve your security. Consider the following use cases and alternatives.
Command Line Interface (CLI)
You plan to use this access key to enable the AWS CLI to access your AWS account.
Local code
You plan to use this access key to enable application code in a local development environment to access your AWS account.
Application running on an AWS compute service
You plan to use this access key to enable application code running on an AWS compute service like Amazon EC2, Amazon ECS, or AWS Lambda to access your AWS account.
Third-party service
You plan to use this access key to enable access for a third-party application or service that monitors or manages your AWS resources.
Application running outside AWS
CloudShell
Feedback
Language
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences


us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/users/details/20A31A05D2/create-access-key
Services
Search
[Alt+S]
Global
20A31A05F9 @ ezts
IAM > Users > 20A31A05D2 > Create access key
Step 1
Access key best practices & alternatives
Step 2 - optional
Set description tag
Step 3
Retrieve access keys
Set description tag - optional
The description for this access key will be attached to this user as a tag and shown alongside the access key.
Description tag value
Describe the purpose of this access key and where it will be used. A good description will help you rotate this access key confidently later.
Maximum 256 characters. Allowed characters are letters, numbers, spaces representable in UTF-8, and: _ . : / = + - @
Cancel
Previous
Create access key
CloudShell
Feedback
Language
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences


us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/users/details/20A31A05D2/create-access-key
Services
Search
[Alt+S]
Global
20A31A05F9 @ ezts
Access key created
This is the only time that the secret access key can be viewed or downloaded. You cannot recover it later. However, you can create a new access key any time.
IAM > Users > 20A31A05D2 > Create access key
Step 1
Access key best practices & alternatives
Step 2 - optional
Set description tag
Step 3
Retrieve access keys
Retrieve access keys
Access key
If you lose or forget your secret access key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.
Access key
Secret access key
AKIATR4OXV3QD5GD6MZZ
********** Show
Access key best practices
Never store your access key in plain text, in a code repository, or in code.
Disable or delete access key when no longer needed.
Enable least-privilege permissions.
Rotate access keys regularly.
CloudShell
Feedback
Language
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences

STEP 7 - Now you can use this access key to configure CLI

STEP 8 - Open Command Line Interface and run the following command

>aws configure

After entering this command AWS CLI prompts us with four pieces of information

1. Access Key ID: (enter your ID)

2.Secret Access Key: (enter your key)

3.AWS Region: (enter the desired region)

4.Output Format: (enter the desired output)

Finally we get Javascript Object Notation of all the users as output.

ELASTIC CLOUD COMPUTE (EC2)

Lab - Creating an EC2 Instance

Step-1: Go to AWS services, click EC2, and then select ‘launch instances’.

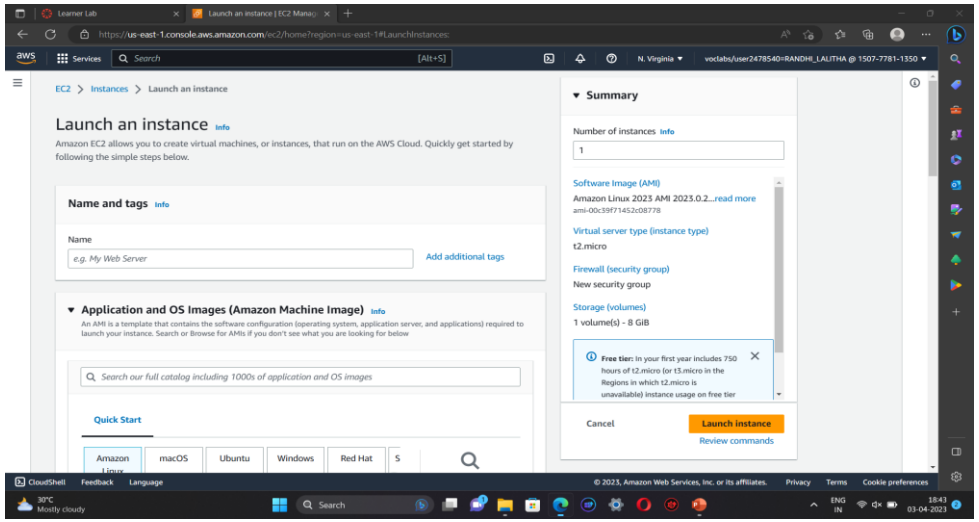
Step-2: Name the instance, select an AMI(LINUX, WINDOWS server), select a key pair, and click launch instance.

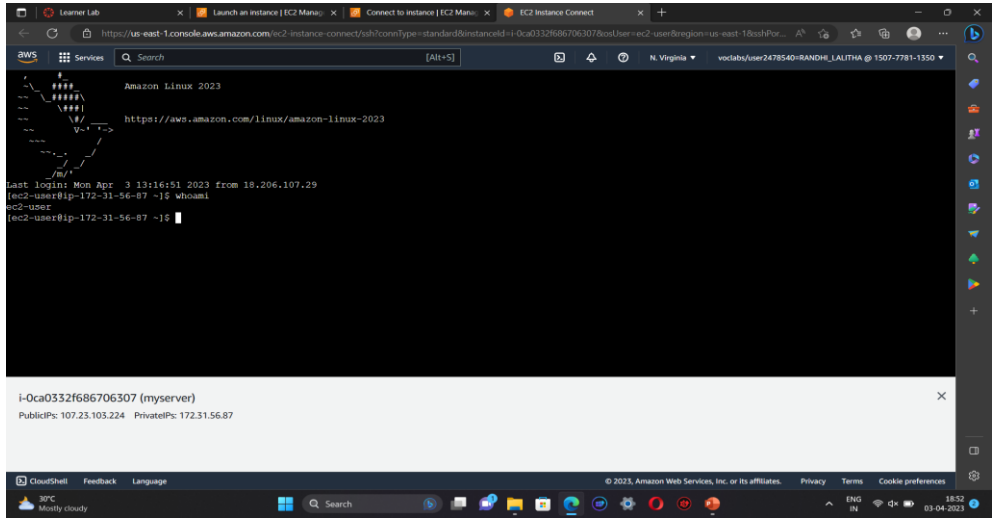
Step-3: For Linux-select the PPK key and for windows server-select the pem key.

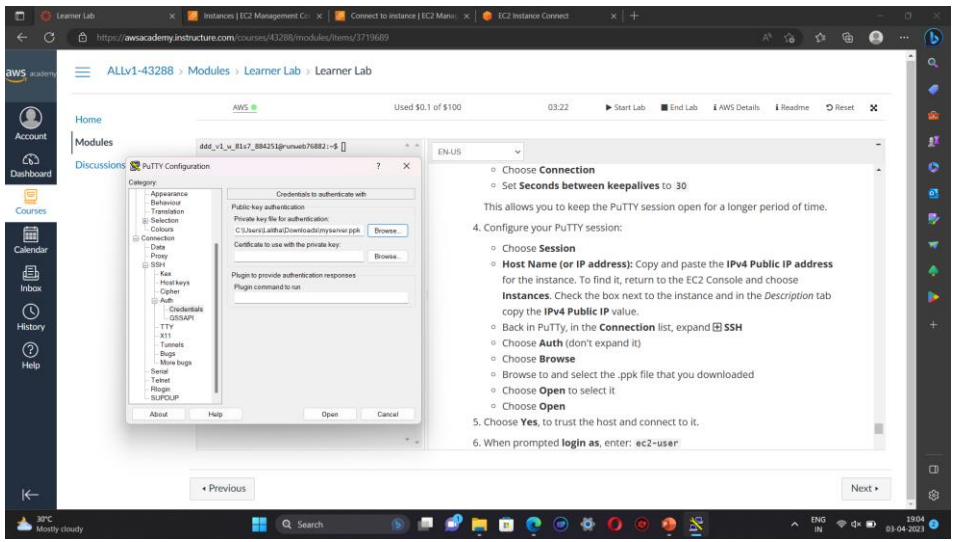
Step-4: If a key pair is not available create a new key.

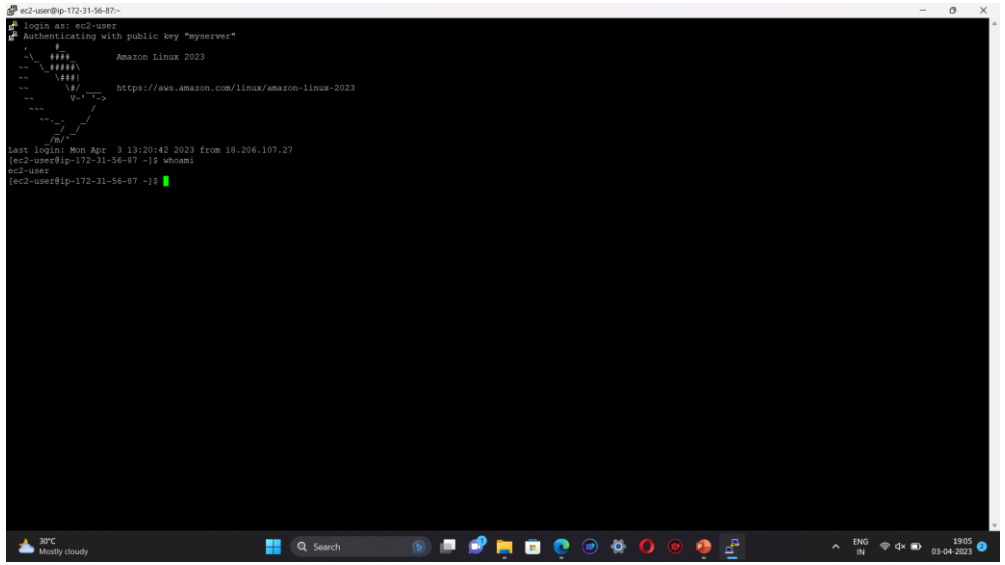
Step-5: For Linux-click connect to instance you will be redirected to the CLI (or) open the putty file configure it to not timeout, and configure the putty session. This will redirect you to the CLI.

For windows server-click connect→RDP client→ get password→ upload private key→ decrypt password. Open the RDP file and enter the password. This will redirect you to the windows server. Now terminate instances.


EC2 > Instances > Launch an instance
Launch an instance Info
Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.
Name and tags Info
Name
e.g. My Web Server
Add additional tags
Application and OS Images (Amazon Machine Image) Info
An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below
Search our full catalog including 1000s of application and OS images
Quick Start
Amazon Linux
macOS
Ubuntu
Windows
Red Hat
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2023 AMI 2023.0.2...read more
ami-00c39f71452c08778
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
Cancel
Launch instance
Review commands

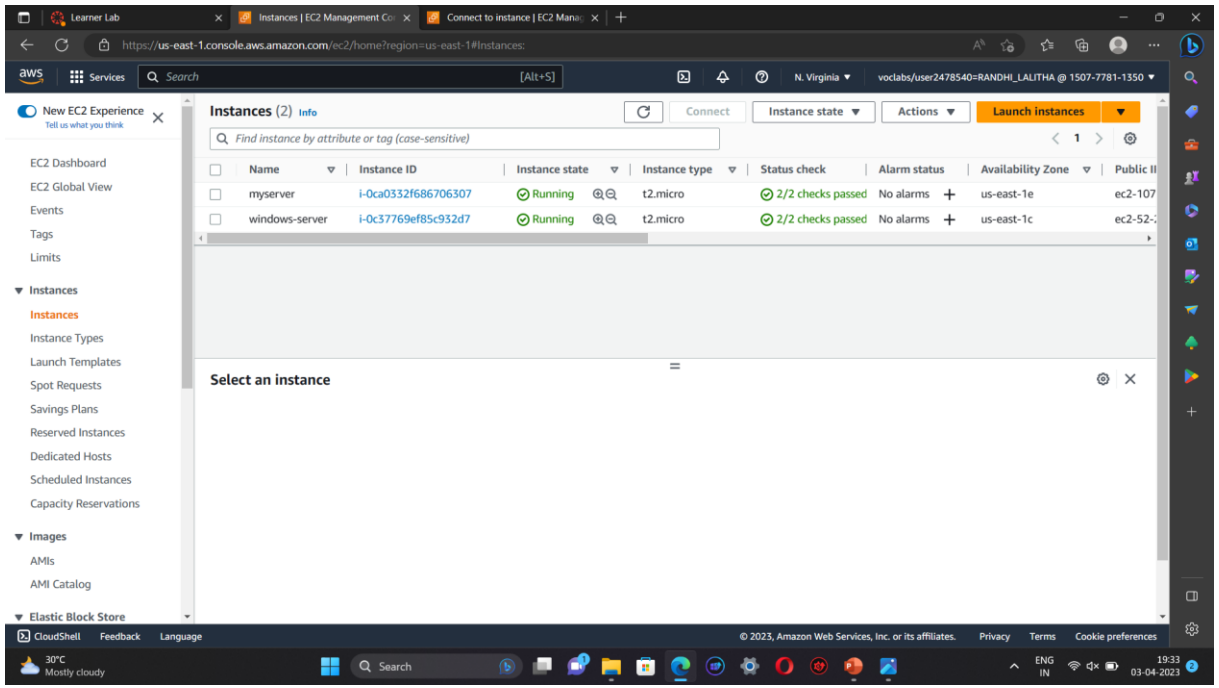

Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023
Last login: Mon Apr 3 13:16:51 2023 from 18.206.107.29
[ec2-user@ip-172-31-56-87 ~]$ whoami
ec2-user
[ec2-user@ip-172-31-56-87 ~]$
i-0ca0332f686706307 (myserver)
PublicIPs: 107.23.103.224 PrivateIPs: 172.31.56.87


ALLv1-43288 > Modules > Learner Lab > Learner Lab
Home
Modules
Discussions
Account
Dashboard
Courses
Calendar
Inbox
History
Help
PuTTY Configuration
Choose Connection
Set Seconds between keepalives to 30
This allows you to keep the PuTTY session open for a longer period of time.
4. Configure your PuTTY session:
Choose Session
Host Name (or IP address): Copy and paste the IPv4 Public IP address for the instance. To find it, return to the EC2 Console and choose Instances. Check the box next to the instance and in the Description tab copy the IPv4 Public IP value.
Back in PuTTy, in the Connection list, expand SSH
Choose Auth (don't expand it)
Choose Browse
Browse to and select the .ppk file that you downloaded
Choose Open to select it
Choose Open
5. Choose Yes, to trust the host and connect to it.
6. When prompted login as, enter: ec2-user
Previous
Next


login as: ec2-user
Authenticating with public key "myserver"
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023
Last login: Mon Apr 3 13:20:42 2023 from 18.206.107.27
[ec2-user@ip-172-31-56-87 ~]$ whoami
ec2-user
[ec2-user@ip-172-31-56-87 ~]$


Networks
Network 2
Do you want to allow your PC to be discoverable by other PCs and devices on this network?
We recommend allowing this on your home and work networks, but not public ones.
Yes
No


Hostname : EC2AMAZ-RNIQ1T8
Instance ID : i-0c31bceeb5a820086
Private IP Address : 172.31.90.111
Public IP Address : 52.91.186.118
Instance Size : t2.micro
Availability Zone : us-east-1c
Architecture : AMD64
Total Memory : 1024
Network : Low to Moderate


Instances (2) Info
Connect
Instance state
Actions
Launch instances
Find instance by attribute or tag (case-sensitive)
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
myserver
i-0ca0332f686706307
Running
t2.micro
2/2 checks passed
No alarms
us-east-1e
windows-server
i-0c37769ef85c932d7
Running
t2.micro
2/2 checks passed
No alarms
us-east-1c
Select an instance
EC2 Dashboard
EC2 Global View
Events
Tags
Limits
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Scheduled Instances
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

VIRTUAL PRIVATE CLOUD (VPC)

Lab – Building a VPC and Launching a Web server

Step 1: First start the lab, when the lab status gets ready, it redirects to the AWS management console.

Step 2: Go to AWS services, search, and choose VPC to open the VPC console.

Step 3: Verify that your regions remain in the N-Virginia(us-east-1). Choose to create VPC in the VPC dashboard

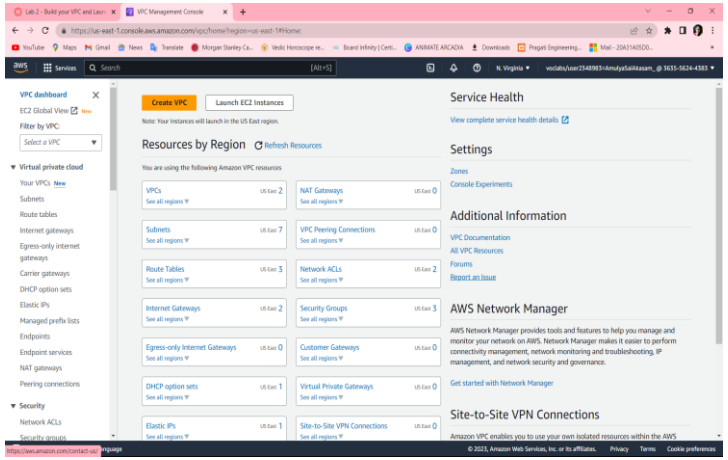
Step 4: Choose VPC and more, in name tag auto-generation, keep auto-generate select and change it to a lab.

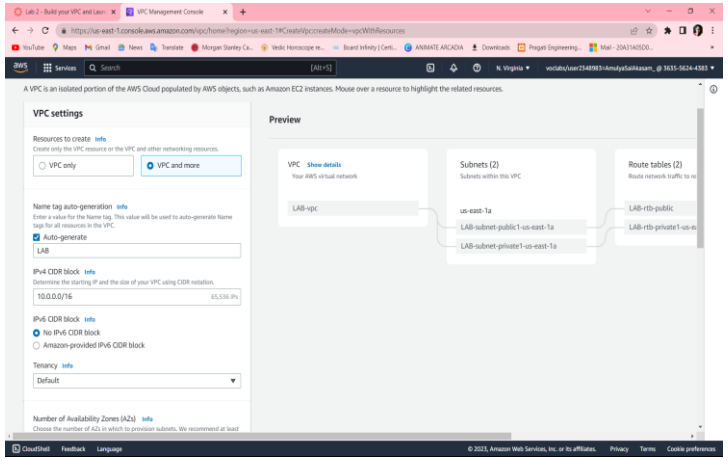
Step 5: Keep the IPv4 CIDR block to 10.10.0.0.0/16 and next for a number of availability zones select 1, number of public subnets select 1 , number of private subnets select 1

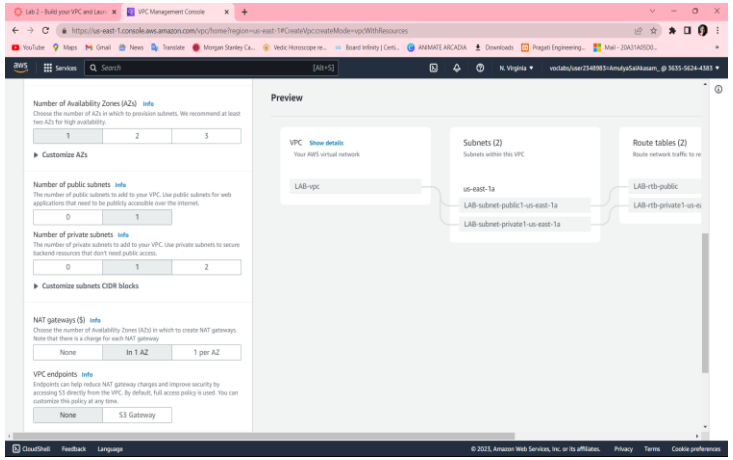
Step 6: Now customize subnets CIDR blocks, change public subnet(us-east-1a) to 10.10.0.0/24, and change private subnet (us-east-1a) to 10.0.1.0/24

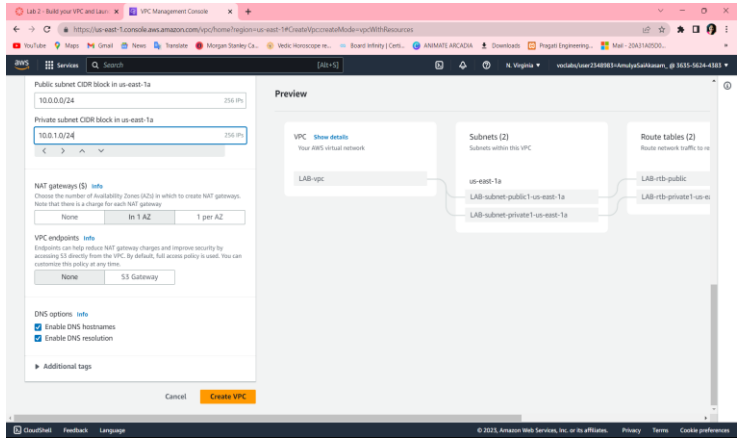
Step 7: Set the NAT gateways to 1AZ and set VPC endpoints to none and keep both DNS hostnames and DNS resolutions enabled

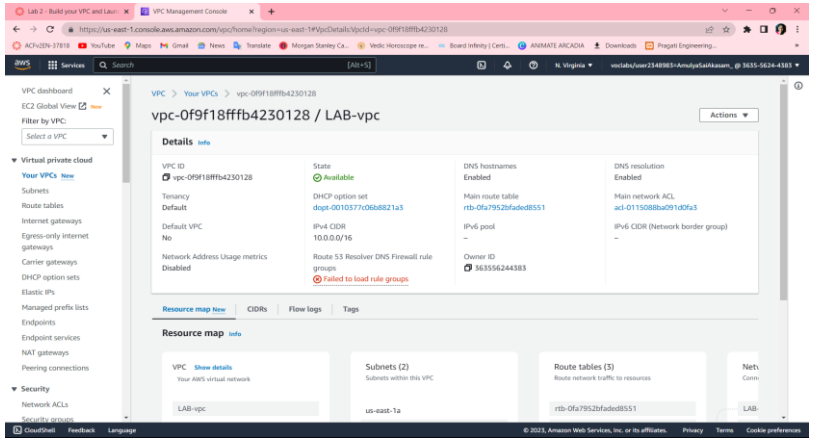
Step 8: Check on the preview panel on the right side whether they match with the given regions or not











CREATING ADDITIONAL SUBNETS :

Step 1: Choose subnets in the left panel, choose to CREATE SUBNET

Step 2: in this, choose VPC ID: LAB-vpc and choose subnet name to lab-subnet-public2, choose availability sone to us-east-1b and choose IPv4 block to 10.0.2.0/24

Step 3: choose to create a subnet and again create the same subnet with lab-subnet-private2, change the IPv4 block to 10.0.2.0/24, and then create a subnet

Step 4: Choose the routing table in the left panel, select lab-rtb-private1-us-east-1a, in the lower panel, choose routes note destination, choose the subnet association tab, in the explicit subnet associations panel choose EDIT SUBNET

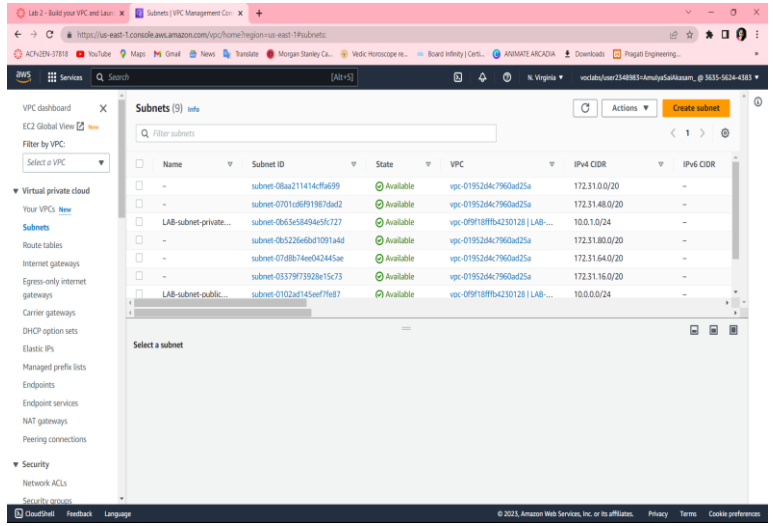
Step 5 : Leave lab-subnet-private1-us-east-1a and also select lab-subnet-private2

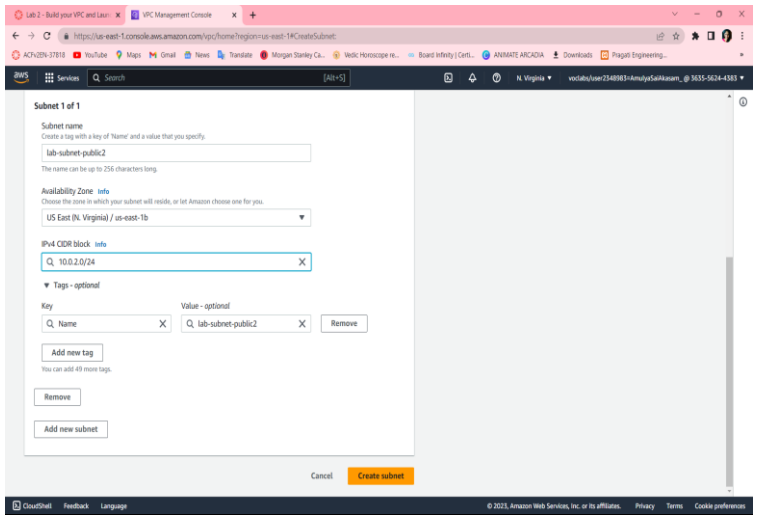
Step 6: Choose SAVE ASSOCIATIONS

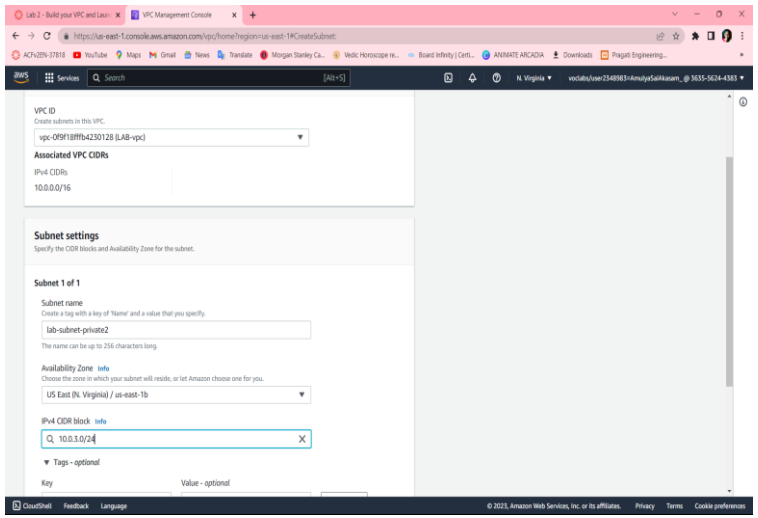
Step 7: Select lab-rtb-public route table and deselect any other subnet

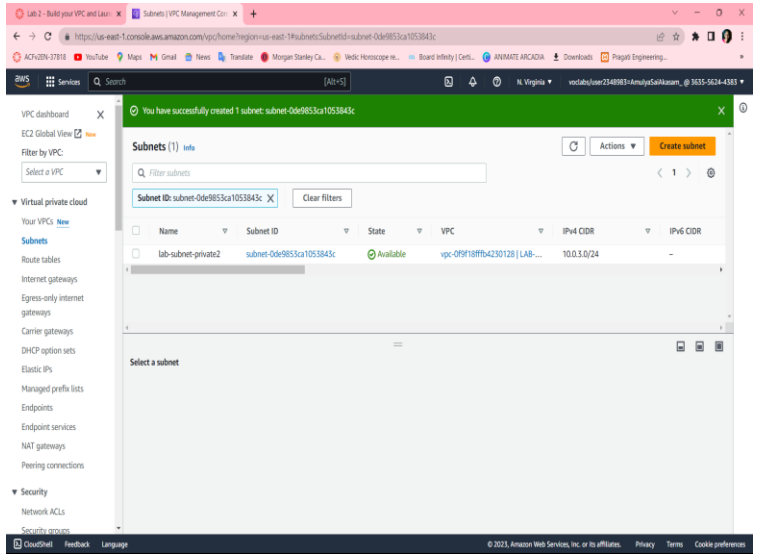
Step 8: In the lower panel, again repeat from step 4 but here we select lab-subnet-public2 also

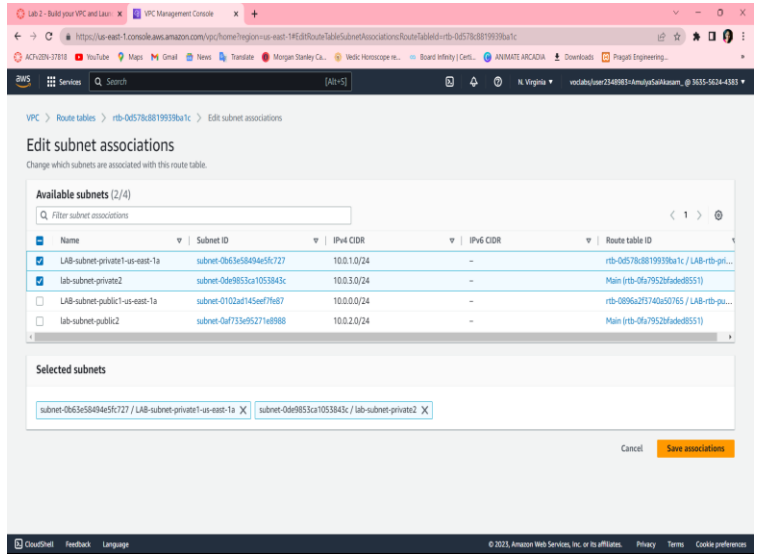
Step 9: Choose SAVE ASSOCIATIONS

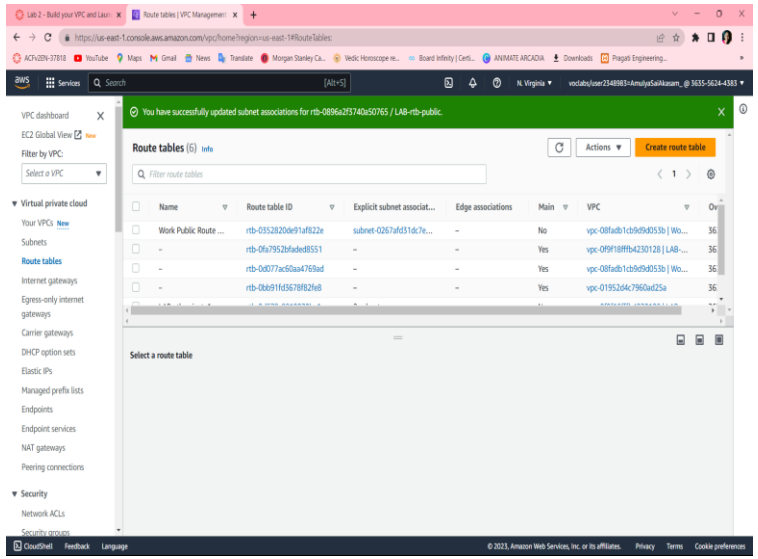







You have successfully created 1 subnet: subnet-0de9853ca1053843c
Subnets (1) Info
Actions
Create subnet
Filter subnets
Subnet ID: subnet-0de9853ca1053843c
Clear filters
Name
Subnet ID
State
VPC
IPv4 CIDR
IPv6 CIDR
lab-subnet-private2
subnet-0de9853ca1053843c
Available
vpc-0f9f18fffb4230128 | LAB-...
10.0.3.0/24
Select a subnet
VPC dashboard
EC2 Global View
Filter by VPC:
Select a VPC
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only internet gateways
Carrier gateways
DHCP option sets
Elastic IPs
Managed prefix lists
Endpoints
Endpoint services
NAT gateways
Peering connections
Security
Network ACLs
Security groups
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

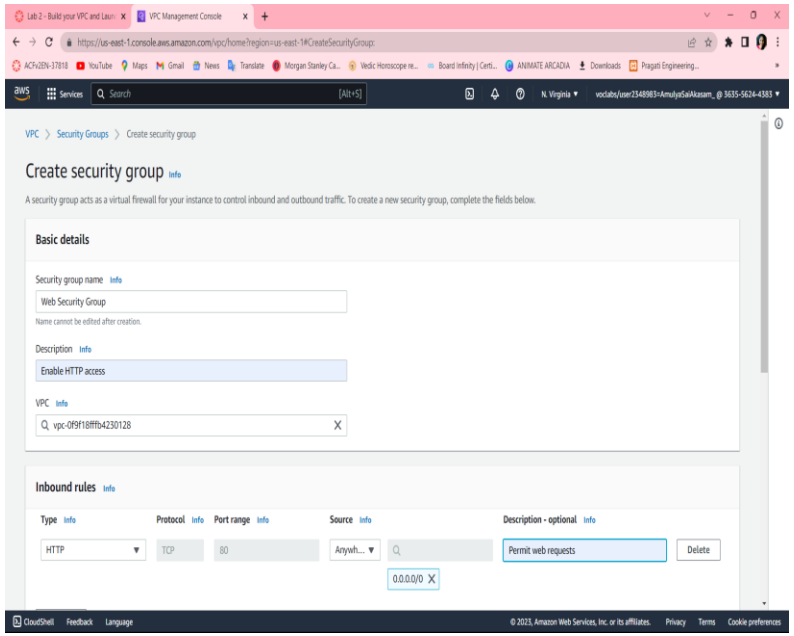

VPC > Route tables > rtb-0d578c8819939ba1c > Edit subnet associations
Edit subnet associations
Change which subnets are associated with this route table.
Available subnets (2/4)
Filter subnet associations
Name
Subnet ID
IPv4 CIDR
IPv6 CIDR
Route table ID
LAB-subnet-private1-us-east-1a
subnet-0b63e58494e5fc727
10.0.1.0/24
rtb-0d578c8819939ba1c / LAB-rtb-pri...
lab-subnet-private2
subnet-0de9853ca1053843c
10.0.3.0/24
Main (rtb-0fa7952bfaded8551)
LAB-subnet-public1-us-east-1a
subnet-0102ad145eef7fe87
10.0.0.0/24
rtb-0896a2f5740a50765 / LAB-rtb-pu...
lab-subnet-public2
subnet-0af733e95271e8988
10.0.2.0/24
Main (rtb-0fa7952bfaded8551)
Selected subnets
subnet-0b63e58494e5fc727 / LAB-subnet-private1-us-east-1a
subnet-0de9853ca1053843c / lab-subnet-private2
Cancel
Save associations
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

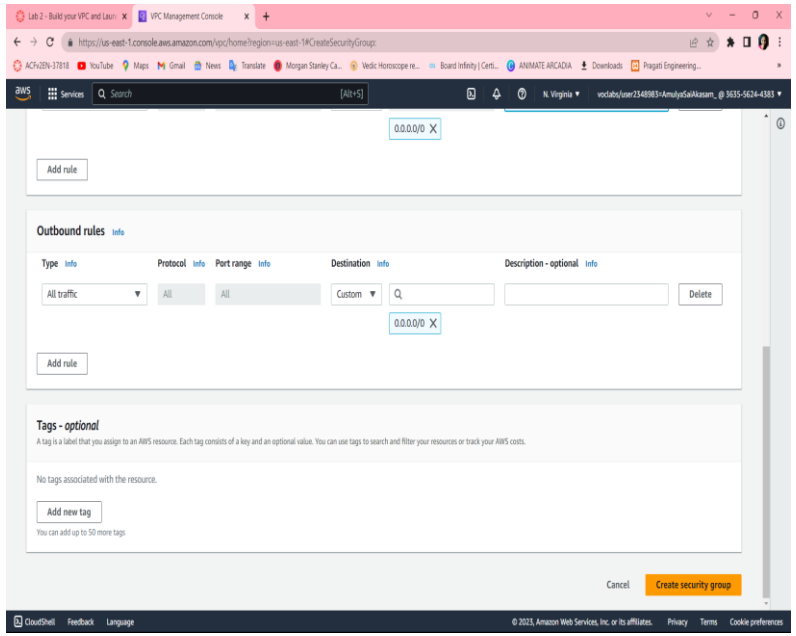

You have successfully updated subnet associations for rtb-0896a2f5740a50765 / LAB-rtb-public.
Route tables (6) Info
Actions
Create route table
Filter route tables
Name
Route table ID
Explicit subnet associat...
Edge associations
Main
VPC
Work Public Route ...
rtb-0352820de91af822e
subnet-0267afd51dc7e...
No
vpc-08fadb1cb9d9d053b | Wo...
rtb-0fa7952bfaded8551
Yes
vpc-0f9f18fffb4230128 | LAB-...
rtb-0d077ac60aa4769ad
vpc-08fadb1cb9d9d053b | Wo...
rtb-0bb91fd3678f82fe8
vpc-01952d4c7960ad25a
Select a route table
VPC dashboard
EC2 Global View
Filter by VPC:
Select a VPC
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only internet gateways
Carrier gateways
DHCP option sets
Elastic IPs
Managed prefix lists
Endpoints
Endpoint services
NAT gateways
Peering connections
Security
Network ACLs
Security groups
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

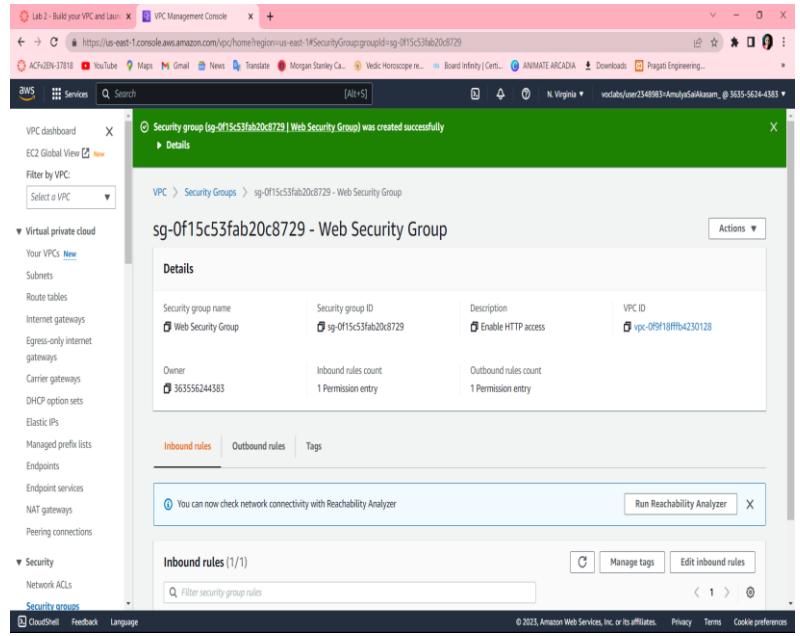
CREATING A VPC SECURITY GROUP

Step 1: Choose to create a security group and in this choose the security group name to a web security group, have a description as enable HTTP access and choose vpc to lab-vpc

Step 2: In inbound rules choose to add rule and then in this chosen type to HTTP, source to Anywhere ipv4 and description to permit web requests







LAUNCH A WEB SERVER INSTANCE

Step 1 : Choose EC2 to open EC2 console , from instances click launch instance and give name as web server 1

Step 2 : Keep Amazon Linux selelct and select amazon linux 2023 AMI , Choose t2.micro

Step 3 : Choose key pair name to vockey , in network settings choose edit select network to lab-vpc ,subnet to lab-subnet-public2 and auton assign to enable

Step 4 : Under firewall choose select existing security group , for common security groups select web security group

Step 5 : Expand the advanced details panel , scroll down upto the user data box appear .

```
#!/bin/bash
# Install Apache Web Server and PHP
sudo dnf install -y httpd wget php mariadb105-server
# Download Lab files get https://aws-tc-largeobjects.s3.us-west-
2.amazonaws.com/CUR-TF-100-ACCLFO-2-9026/2-lab2-vpc/s3/lab-app.zip
unzip lab-app.zip -d /var/www/html/
# Turn on web server
chkconfig httpd on
service httpd start
```

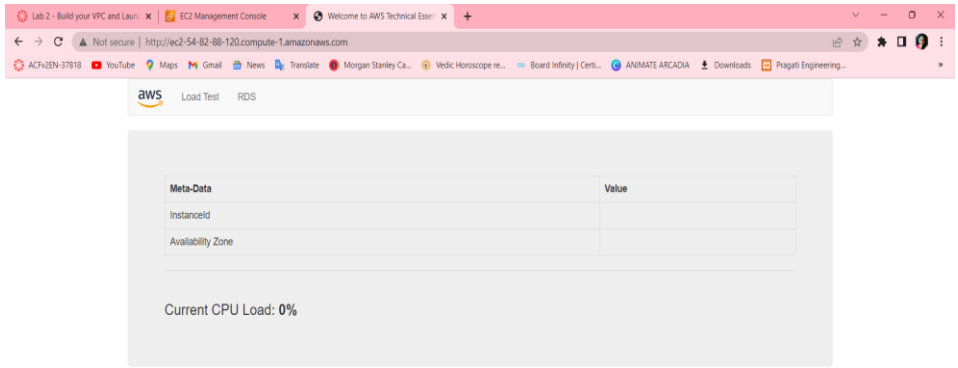
Step 6: At the bottom SUMMARY panel choose launch instance ,click on view instances

Step 7 : Wait until web server 1 shows 2/2 checks passed

Step 8 : Copy the public ipv4 dns value present in details tab

Step 9 : Open a new browser , copy the public dns

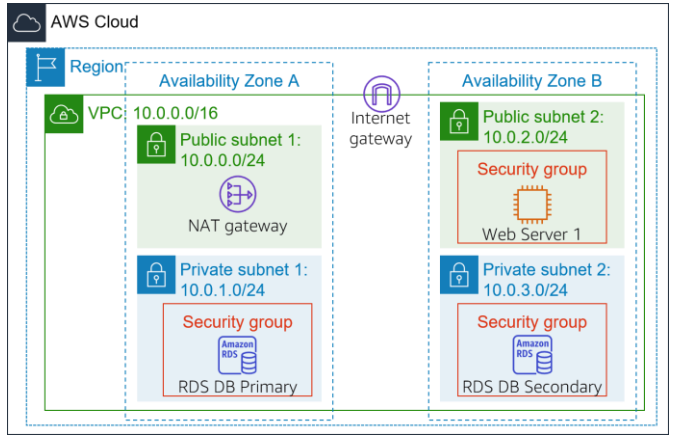
Step 10 : Finally we see a web page displaying the AWS logo and instances meta-data values



Finally, a web page opens displaying the AWS logo and instances of metadata values

ELASTIC LOAD BALANCER (ELB)

Elastic Load Balancing automatically distributes incoming application traffic across multiple Amazon EC2 instances.

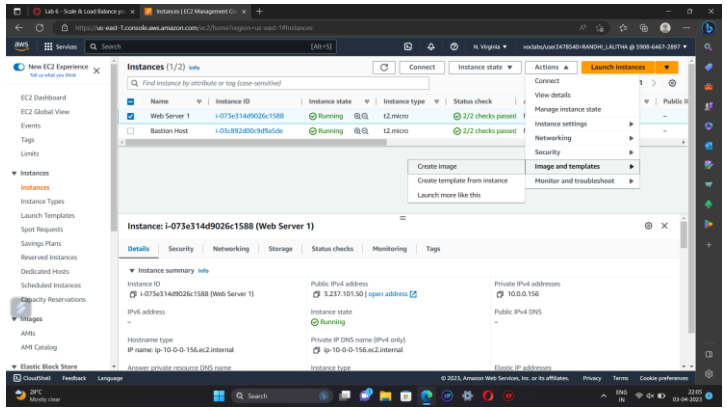


In this lab, We are provided with the given infrastructure.

Procedure:

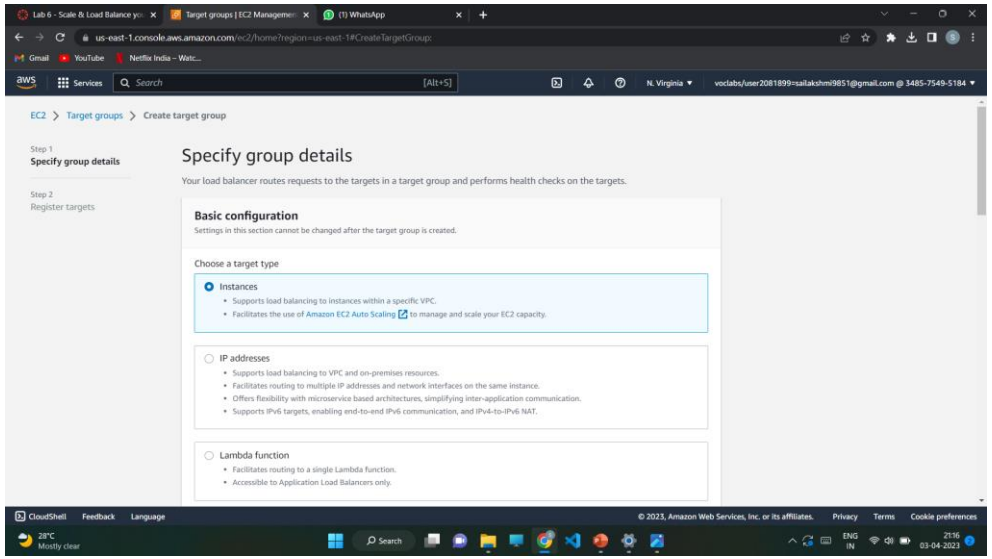
Task1: Creating an AMI for Auto Scaling

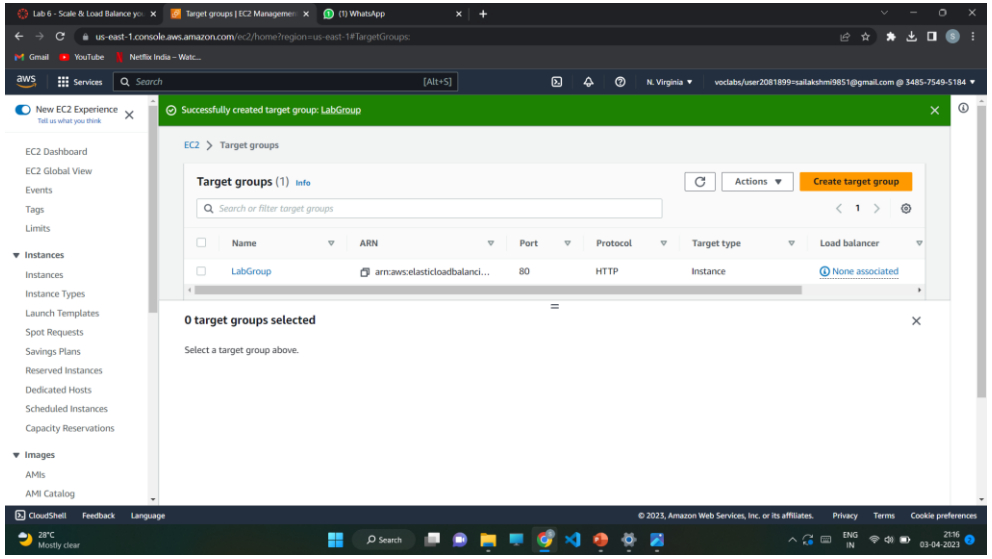
- Click start lab then click on AWS.
- You will navigate to AWS management console. Click on services and select EC2.
- Click instances. Make sure that **Status Checks** for **Web Server 1** displays 2/2 checks.
- Select Web Server 1 and in actions click images and templates > create image. Name the image and give the description.
- Click create image.



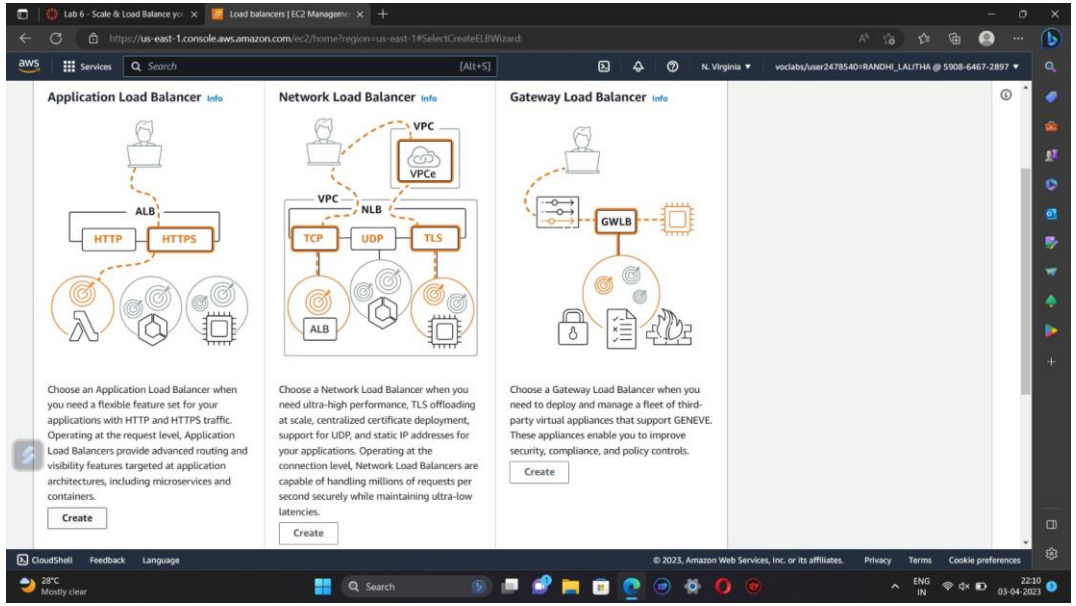
Task 2: Creating a load balancer

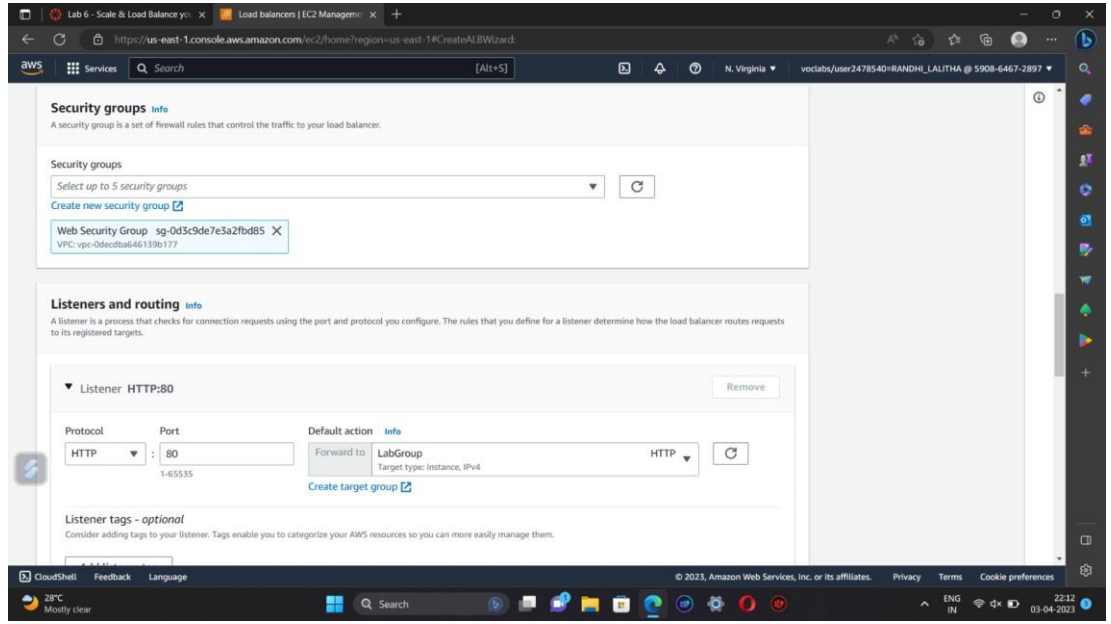
- Choose Target Groups and then click on create target group.
- Select target type as instances. Name the target group. Select Lab VPC under VPC that is we are creating load balancer in Lab VPC .
- Choose next and then click on create target group.

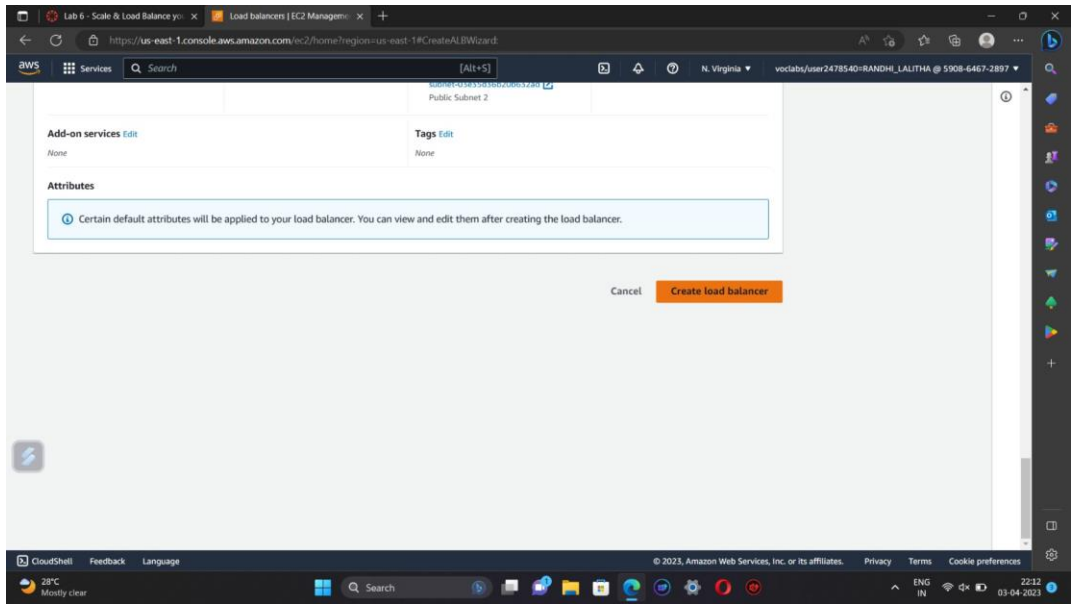


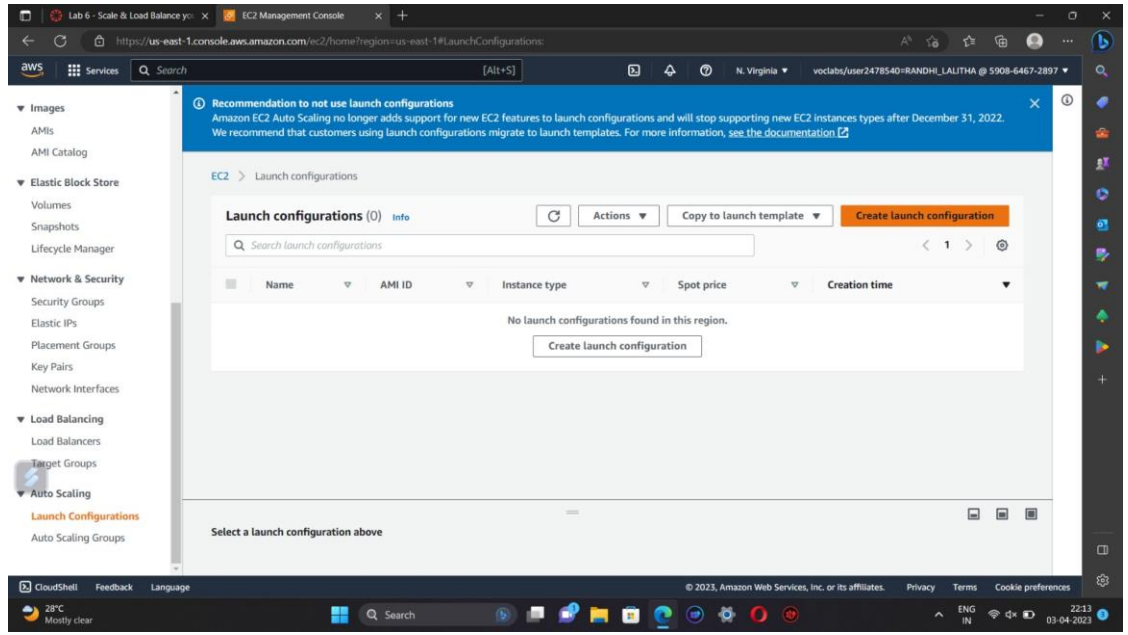


- From the left navigation pane , select Load Balancers. Click create load balancer.
- To create a application balancer, click create under Application Load Balancer and Name it.
- In Networking mapping, select Lab VPC and specify the subnets that the load balancer should use.
- In security groups, select only Web Security Group and deselect all other than it .
- For the Listener HTTP:80 row, set the Default action to forward to **LabGroup**.


Application Load Balancer Info
ALB
HTTP
HTTPS
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
Network Load Balancer Info
VPC
VPCe
VPC
NLB
TCP
UDP
TLS
ALB
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
Gateway Load Balancer Info
GWLB
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


Security groups Info
A security group is a set of firewall rules that control the traffic to your load balancer.
Security groups
Select up to 5 security groups
Create new security group
Web Security Group sg-0d3c9de7e3a2fbd85
VPC: vpc-0decdba646139b177
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
HTTP
Port
80
1-65535
Default action Info
Forward to
LabGroup
Target type: Instance, IPv4
HTTP
Create target group
Listener tags - optional
Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.

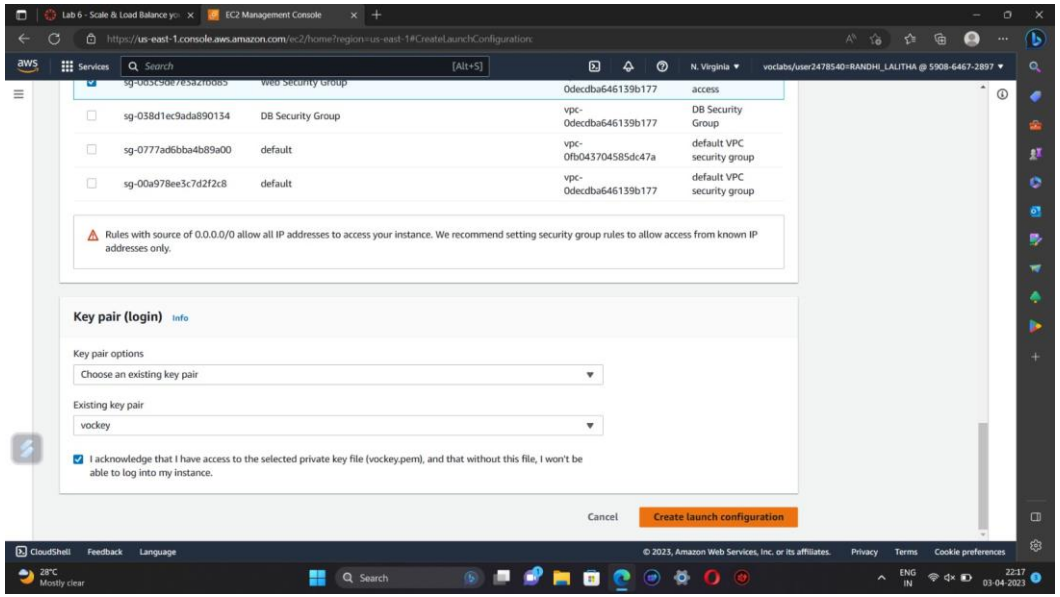

Add-on services Edit
None
Tags Edit
None
Attributes
Certain default attributes will be applied to your load balancer. You can view and edit them after creating the load balancer.
Cancel
Create load balancer

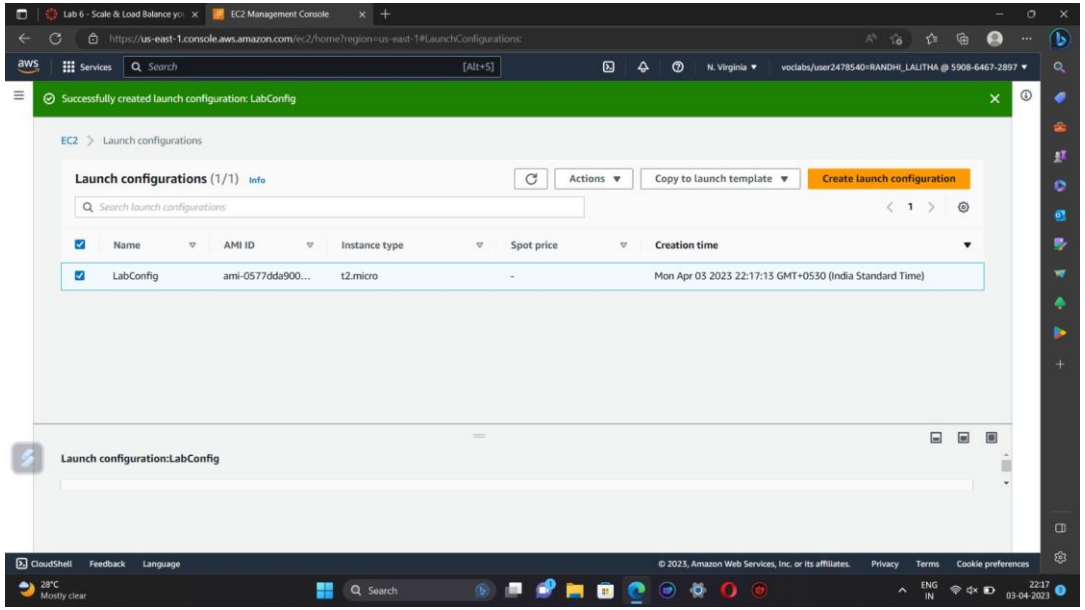

Recommendation to not use launch configurations
Amazon EC2 Auto Scaling no longer adds support for new EC2 features to launch configurations and will stop supporting new EC2 instances types after December 31, 2022. We recommend that customers using launch configurations migrate to launch templates. For more information, see the documentation
EC2 > Launch configurations
Launch configurations (0) Info
Actions
Copy to launch template
Create launch configuration
Search launch configurations
Name
AMI ID
Instance type
Spot price
Creation time
No launch configurations found in this region.
Create launch configuration
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
Auto Scaling
Launch Configurations
Auto Scaling Groups
Select a launch configuration above

- Click create load balancer.

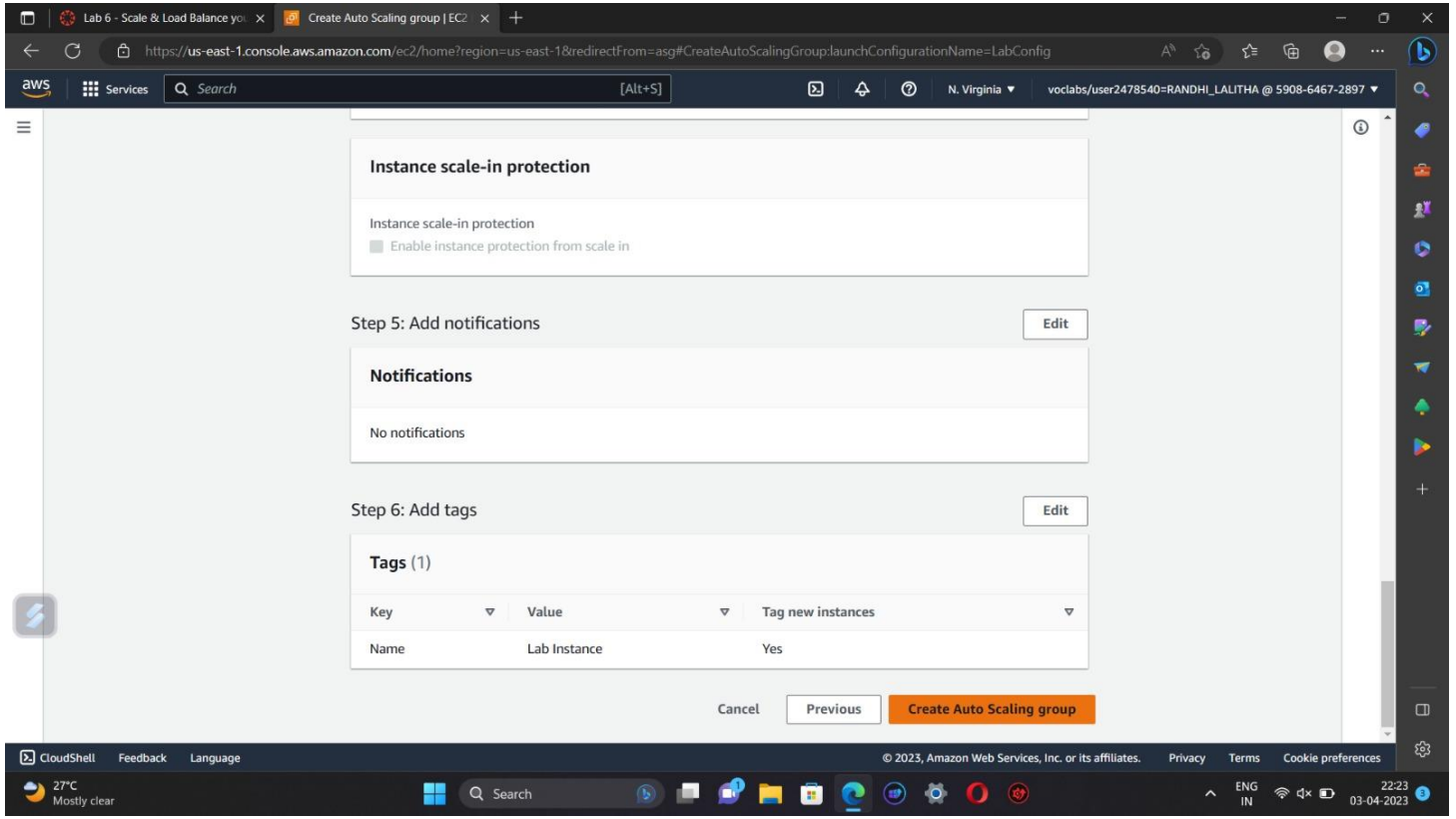
Task 3: Create a Launch Configuration and an Auto Scaling Group

- In Launch Configurations, click create launch configuration.
- Name the configuration and for AMI choose web server AMI that you created in task 1.
- Select the instance type.
- Under Additional Configuration, for monitoring select Enable EC2 instance detailed monitoring within CloudWatch.
- Under security groups , choose an existing security group Web Security Group.
- Under key pair, choose an existing key pair vockey. Check I acknowledge…
- Click Create launch configuration.
- For created launch configuration, select create auto scaling group from actions.
- Name it and select Lab VPC under VPC, select the private subnets.
- Select an existing load balancer which was created earlier.
- In the **Additional settings - *optional*** pane, select **Enable group metrics collection within CloudWatch**



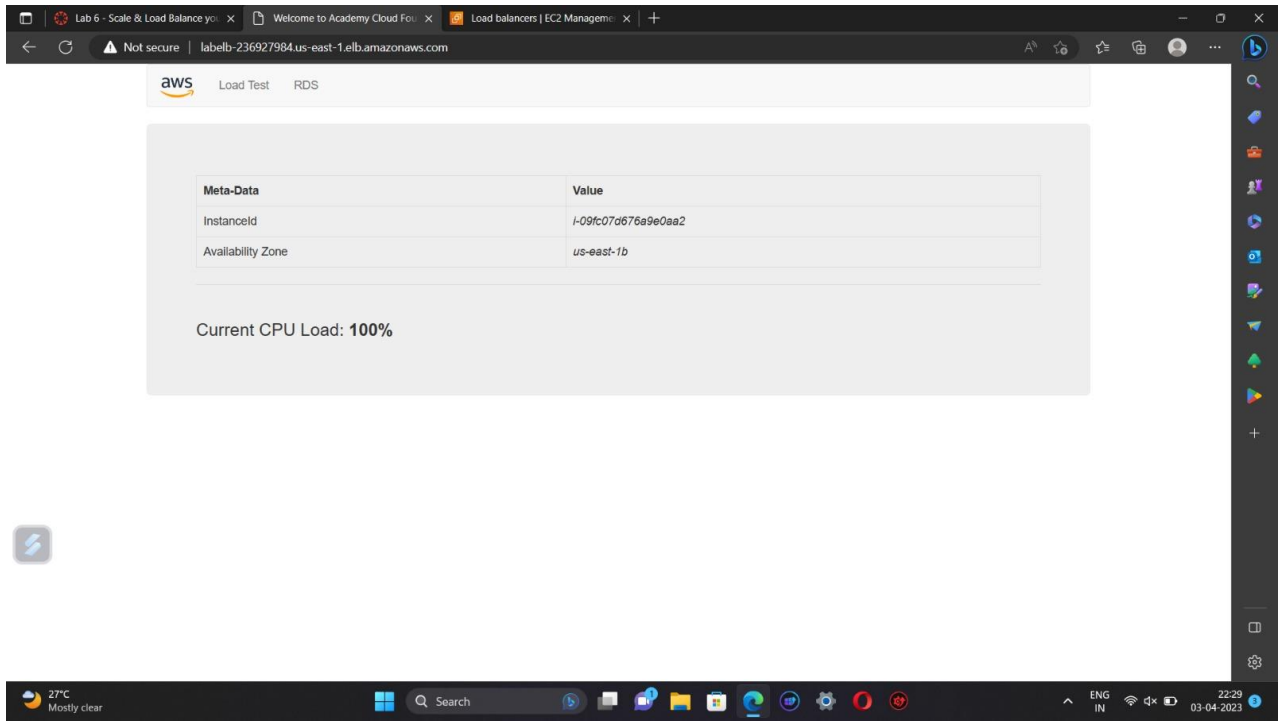


- Specify the values under Group size.
- Under **Scaling policies**, choose *Target tracking scaling policy* and name the policy. Specify metric type and target value. Then add a tag and click create auto scaling group.



Task 4: Verify that Load Balancing is Working

- click **Instances**.You should see two new instances named **Lab Instance**. These were launched by Auto Scaling.
- In the labgroup target group, **two Lab Instance** targets should be listed for this target group.Wait until the **Status** of both instances transitions to *healthy*.
- Now copy the DNS name of the created laod balancer making sure to omit "(A Record)".and paste it in a new browser
- The application should appear in your browser. This indicates that the Load Balancer received the request, sent it to one of the EC2 instances, then passed back the result.

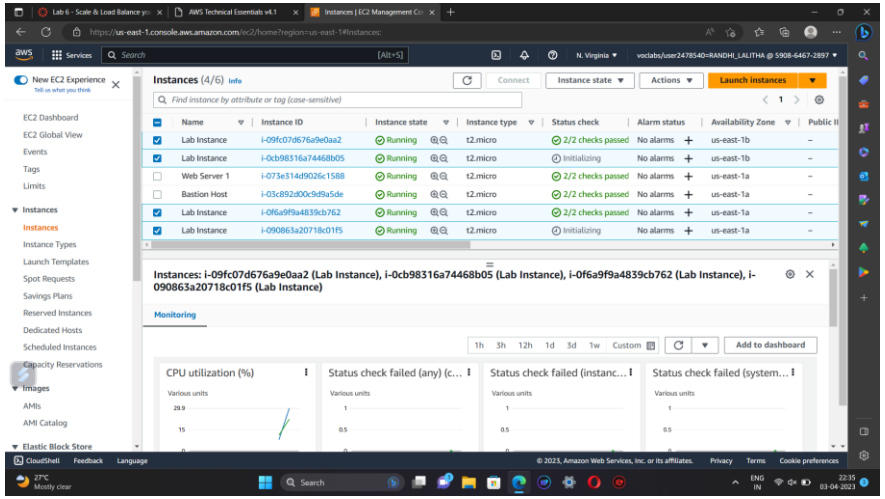


Task 5: Test Auto Scaling

- On the **Services** menu, click **CloudWatch**.In the left navigation pane, choose **All alarms**.Two alarms will be displayed. These were created automatically by the Auto Scaling group.
- From services select EC2 and choose Auto Scaling Groups and select Lab Auto Scaling Group which you created.
- choose the **Automatic Scaling** tab. Select **LabScalingPolicy** and from actions change the target value to 50.click update.
- To go cloudwatch and click all alarms and verify.
- Click the **OK** alarm, which has *AlarmHigh* in its name.Return to the browser tab with the web application.

Click **Load Test** beside the AWS logo.This will cause the application to generate high loads.

- You should see the **AlarmHigh** chart indicating an increasing CPU percentage. Once it crosses the 60% line for more than 3 minutes, it will trigger Auto Scaling to add additional instances.
- In EC2 instances , you notice that more than two instances labeled **Lab Instance** should now be running.
- Finally terminate the Web Server 1.

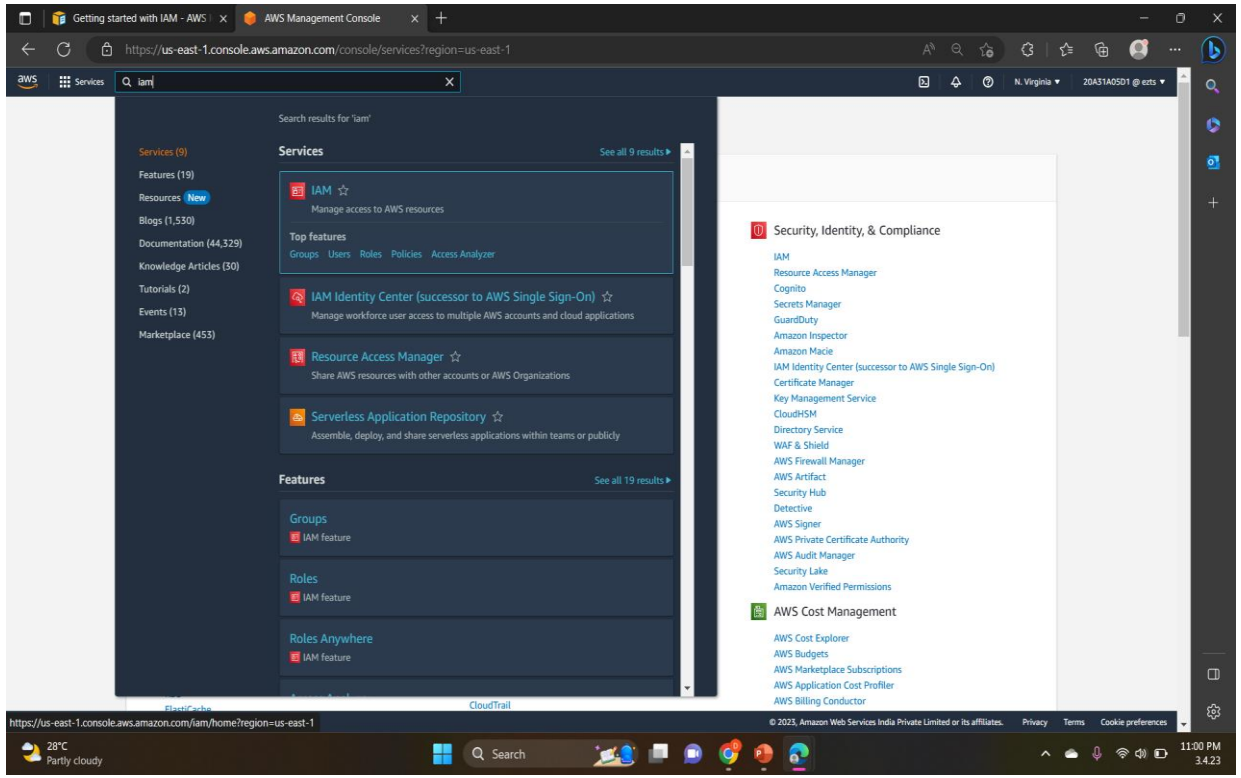


IDENTITY AND ACCESS MANAGEMENT

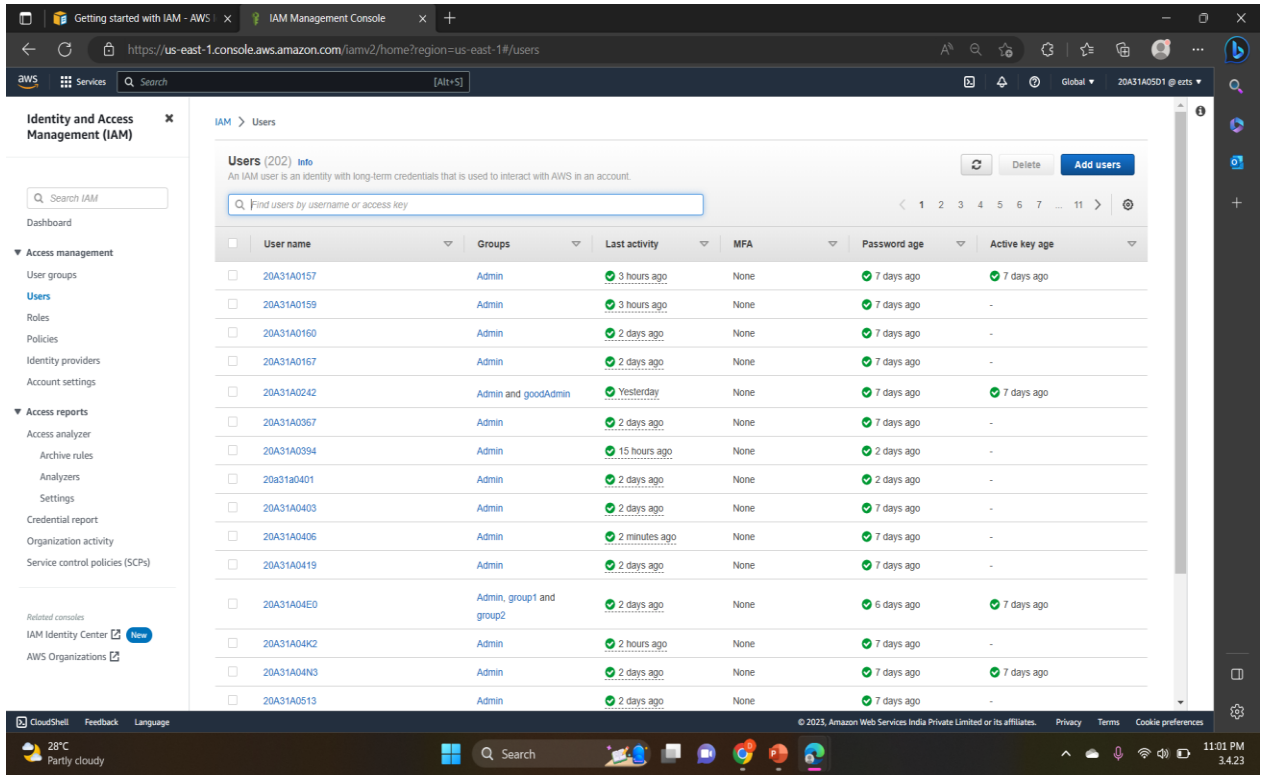
AWS Identity and Access Management (IAM) is a web service that enables Amazon Web Services (AWS) customers to manage users and user permissions in AWS. With IAM, you can centrally manage **users**, **security credentials** such as access keys, and **permissions** that control which AWS resources users can access

Steps to create IAM User and User Groups

1.On the **Console Home** page, select the IAM service.



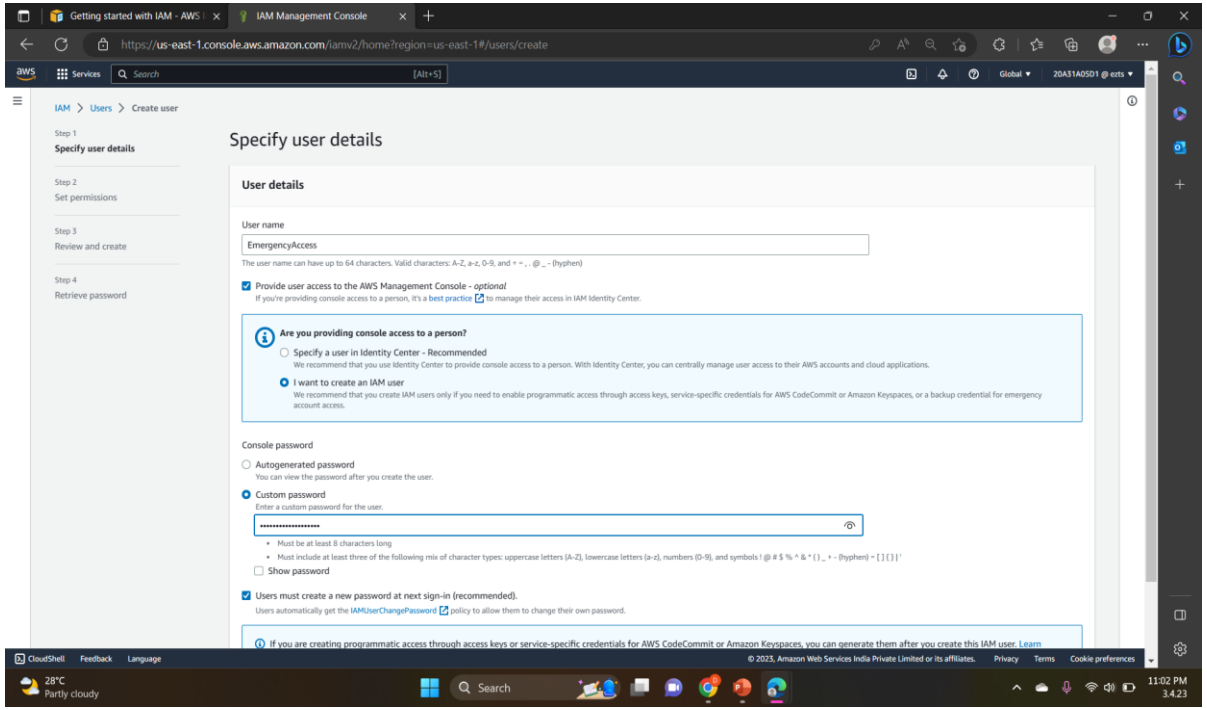
2.In the navigation pane, select **Users** and then select **Add users**.

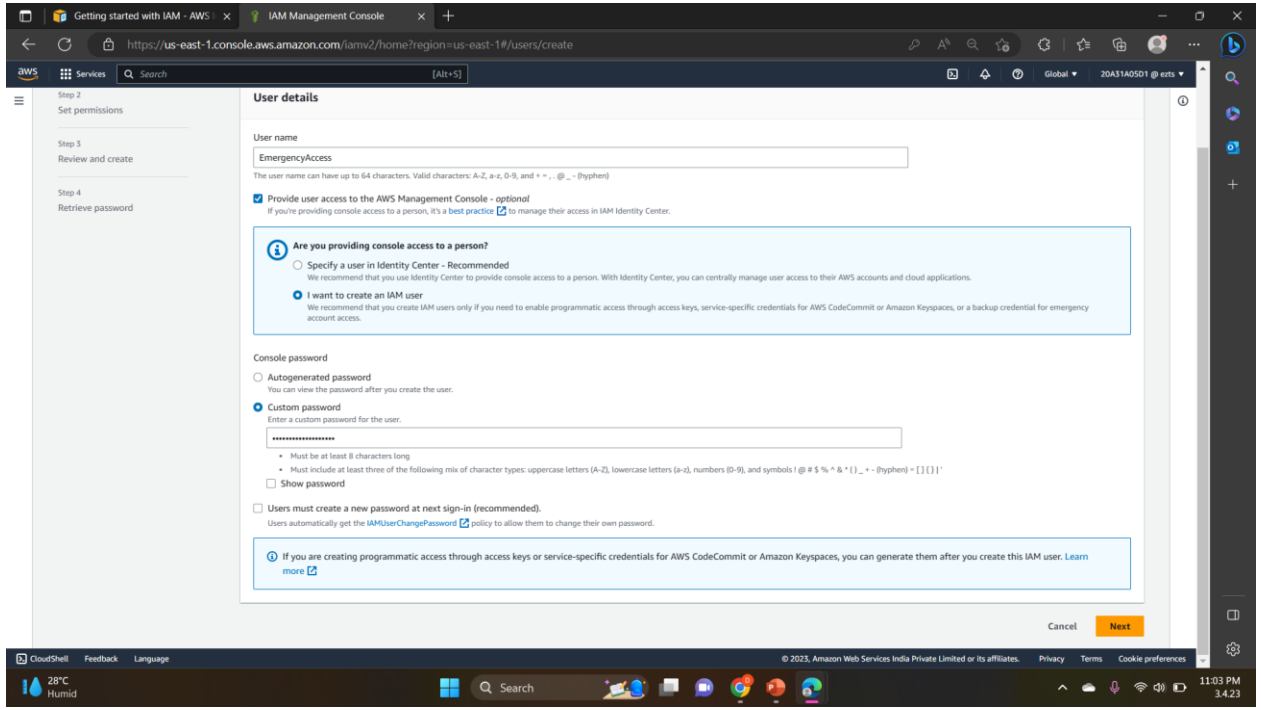


3.For Username, enter EmergencyAccess and ,Select the check box next to **Provide user access to the AWS Management Console– *optional*** and then choose **I want to create an IAM user**.

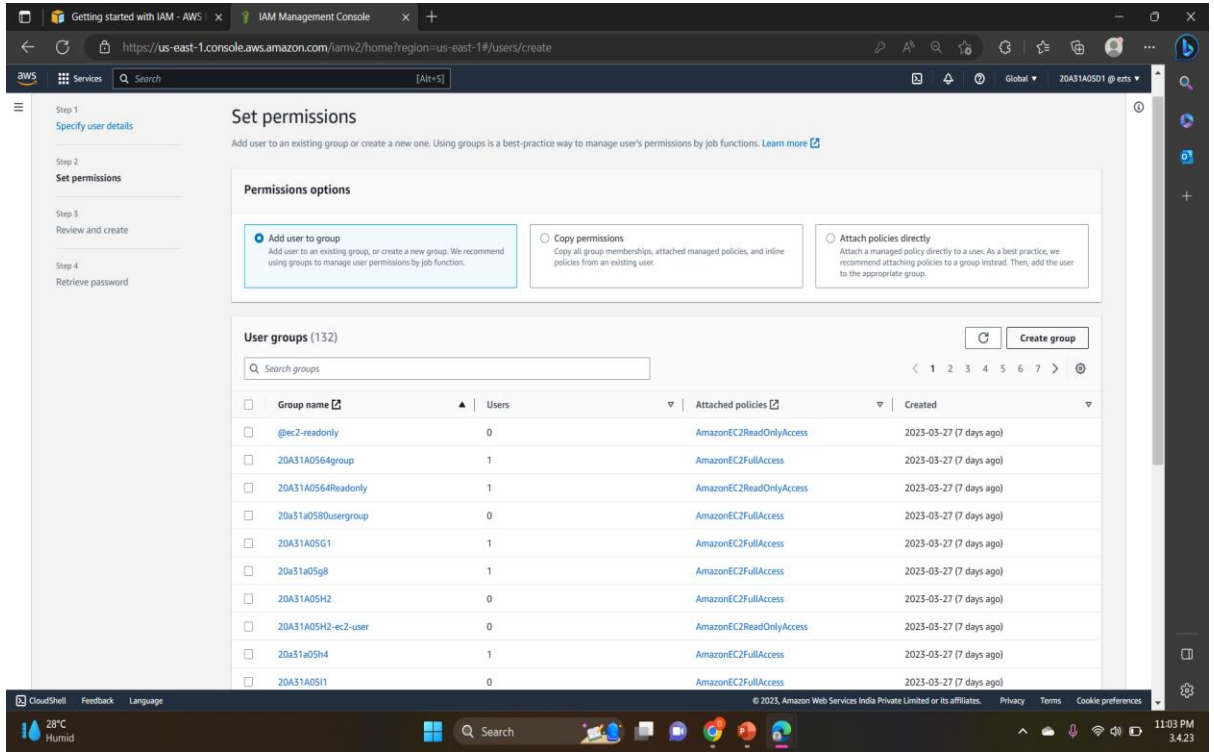
4.Under **Console password**, select **Custom Password** and create your own password.

5. Clear the check box next to **User must create a new password at next sign-in (recommended)**. Then click on **Next.**

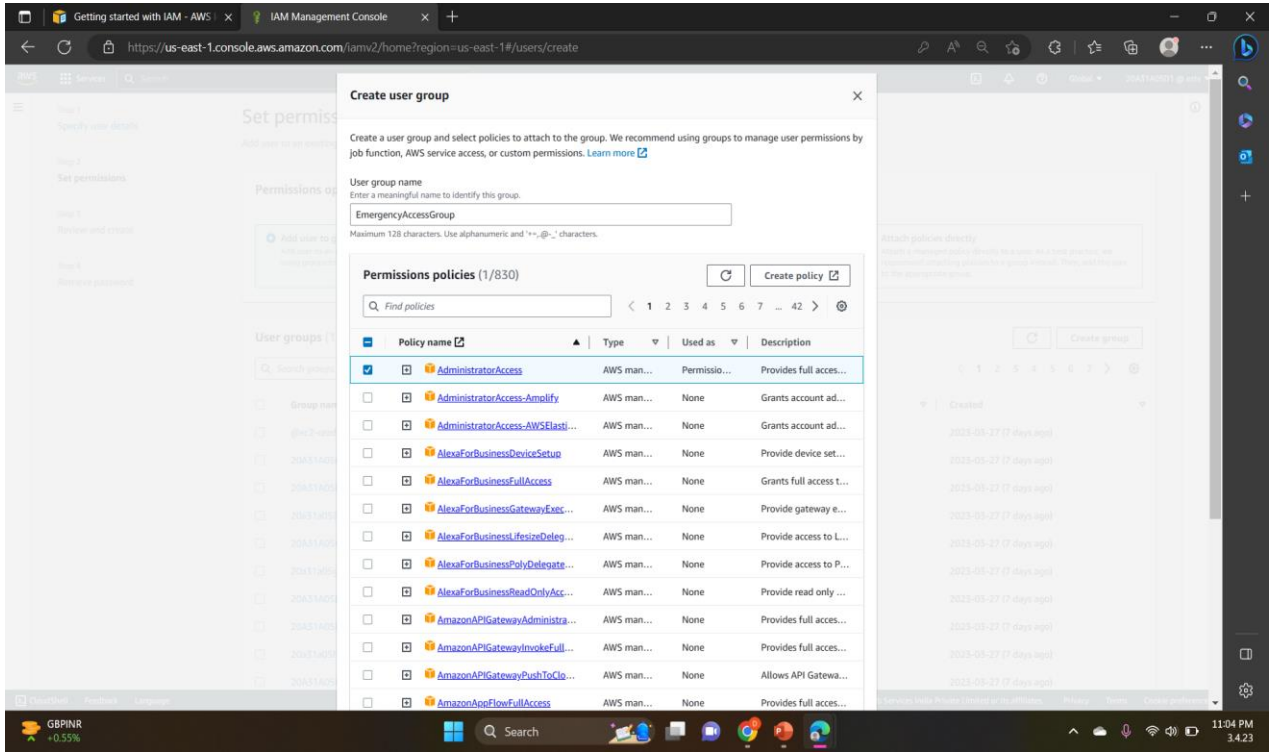




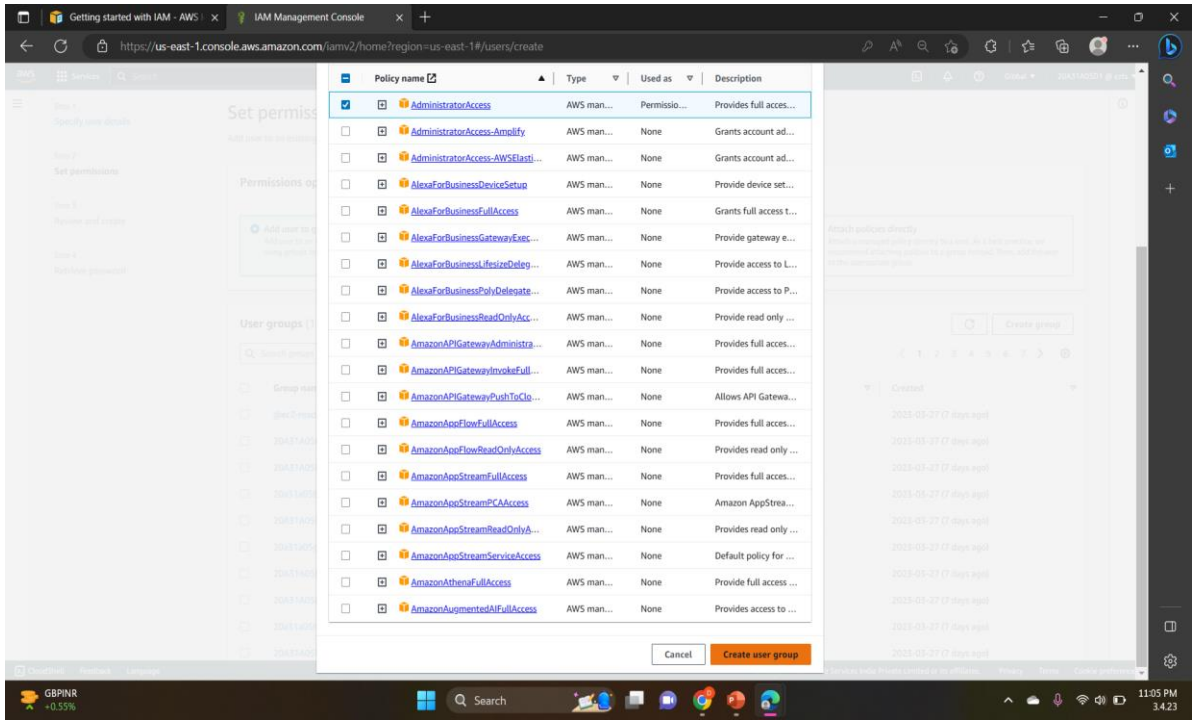
6. On the **Set permissions** page, under **Permissions options**, select **Add user to group**. Then, under **User groups**, select **Create group**.



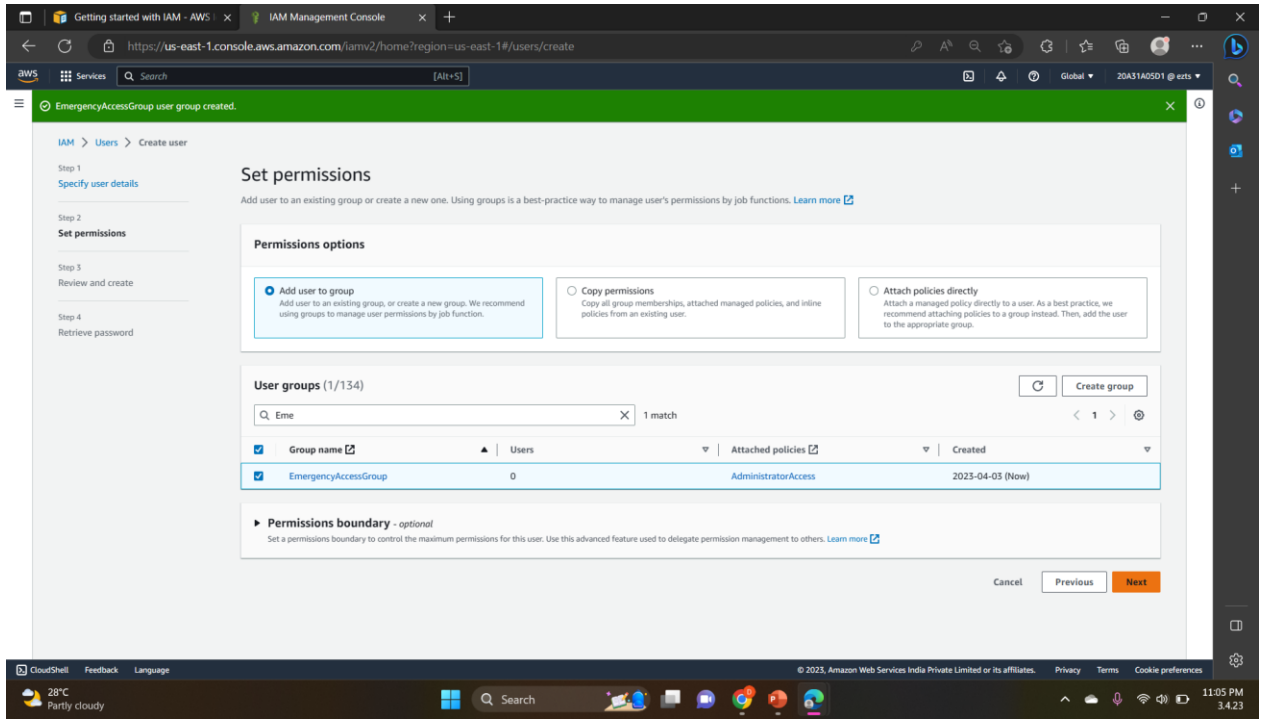
7. On the **Create user group** page, in **User group name**, enter **EmergencyAccessGroup**. Then, under **Permissions policies**, select **AdministratorAccess**.



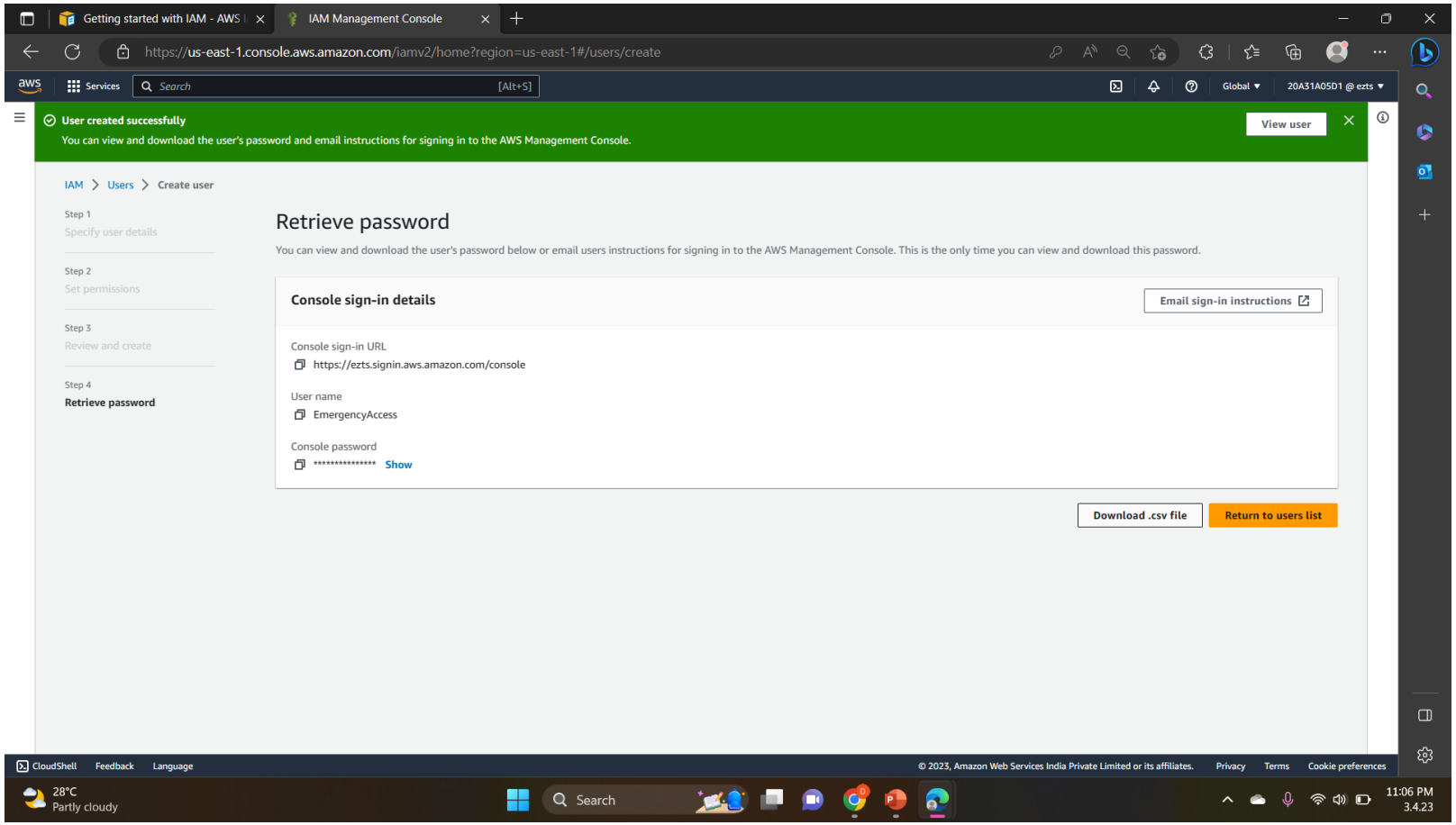
8. Select **Create user group** to return to the **Set permissions** page.



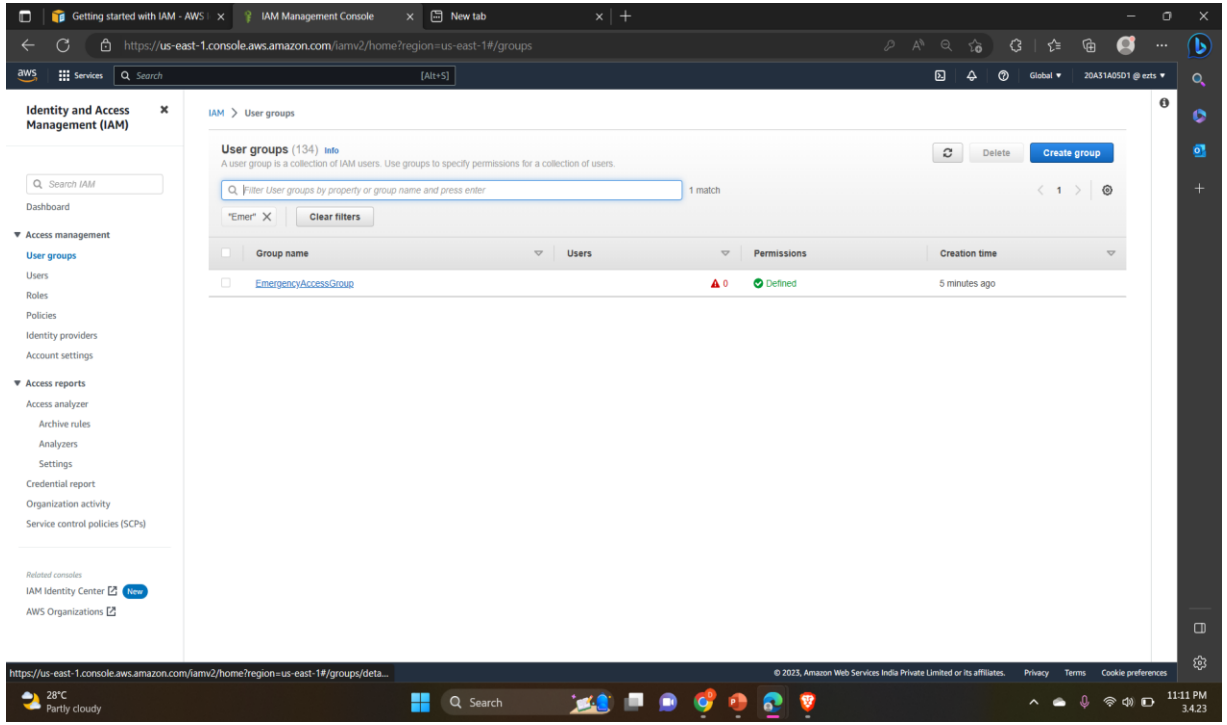
9. Select **Next** to proceed to the **Review and create** page.



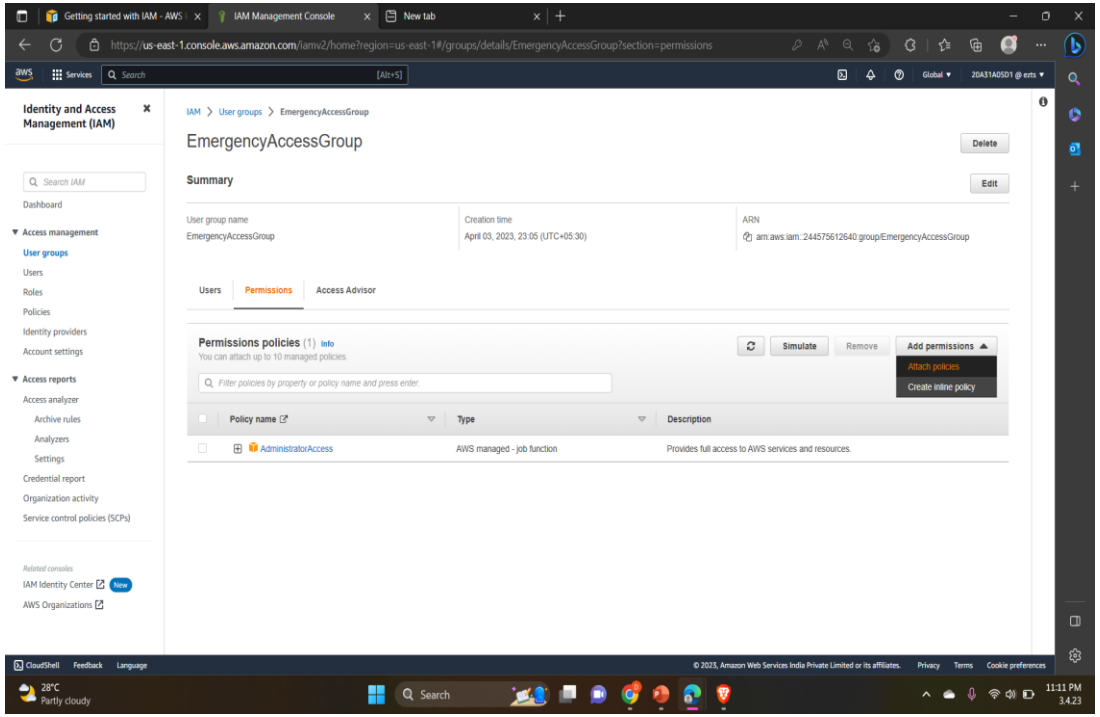
10.On the **Review and create** page, review the list of user group memberships to be added to the new user. When you are ready to proceed, select **Create user**.
11. On the **Retrieve password** page, select **Download .csv file** to save a .csv file with the user credential information (Connection URL, user name, and password).
12. Save this file to use if you need to sign-in to IAM and do not have access to your federated identity provider.



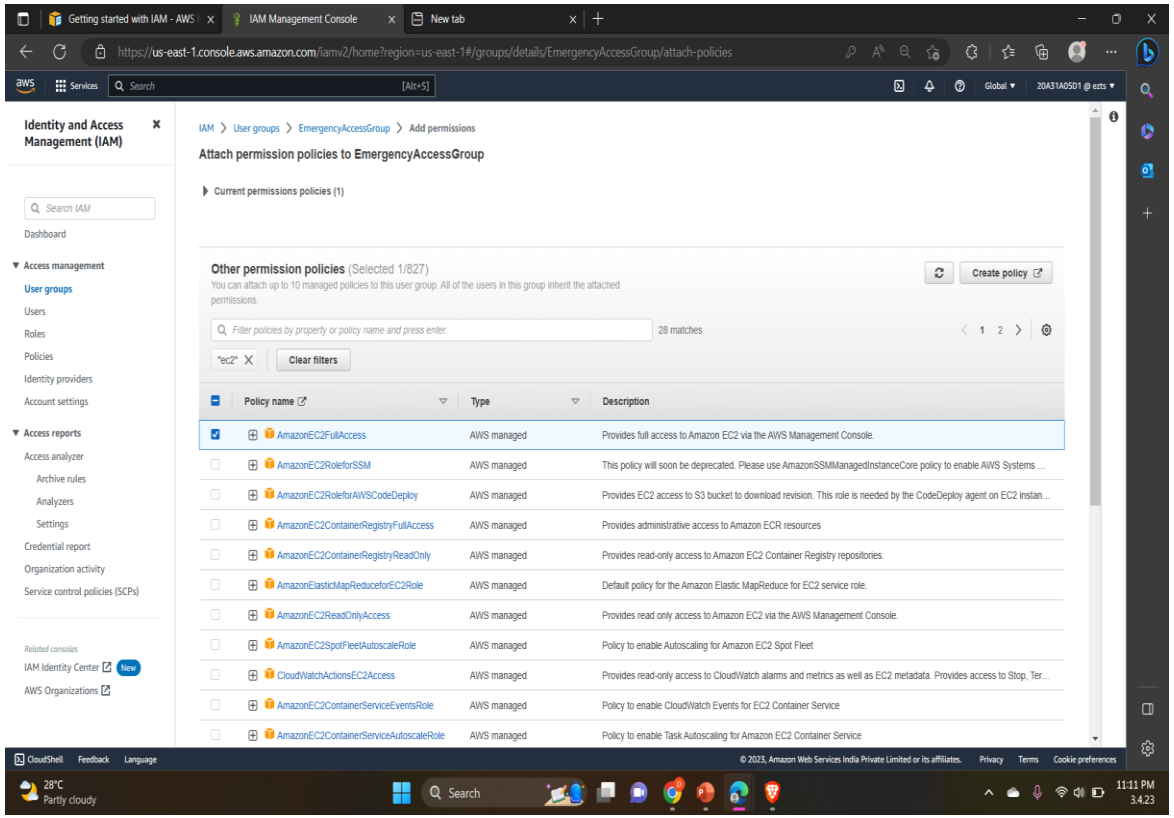
13. If you want to add permissions to the user group, then go to **User groups** and click on the respective **User group.**

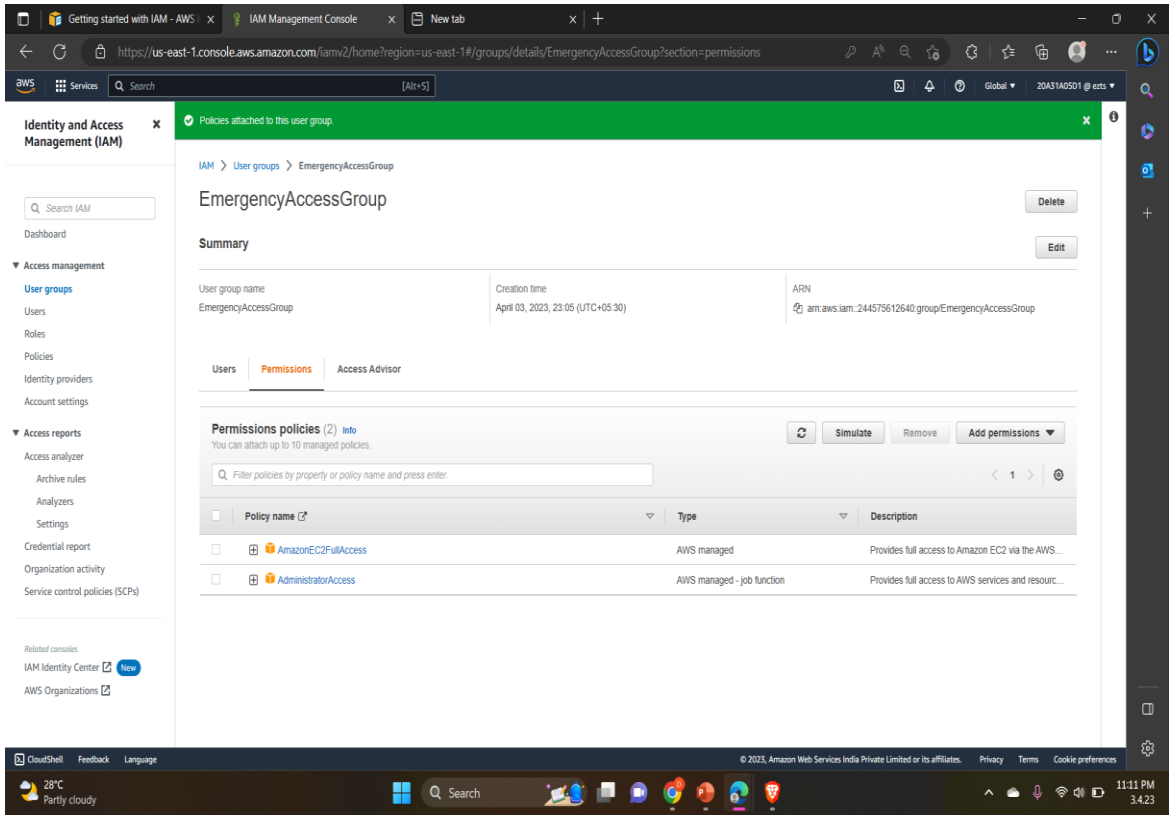


14. Go to Permissions → All permissions→ Attach policies



15. Add the permission policy and the policy is attached to the **User group.**

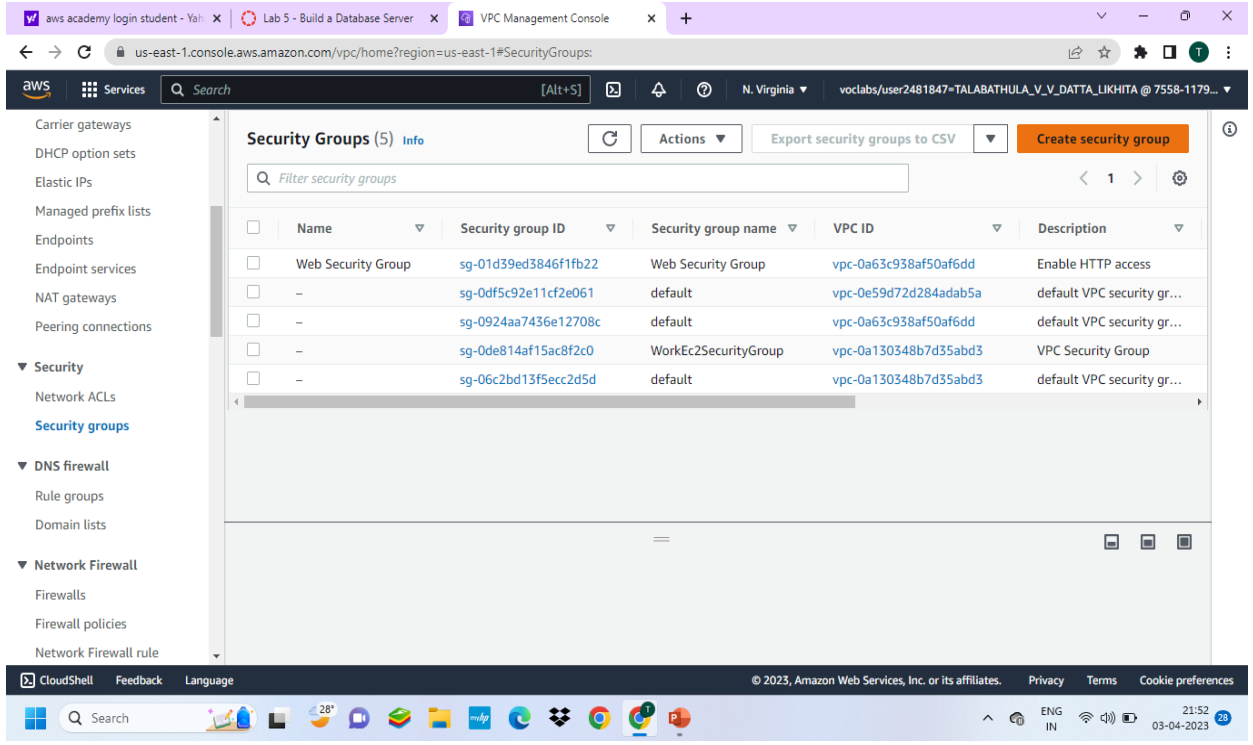


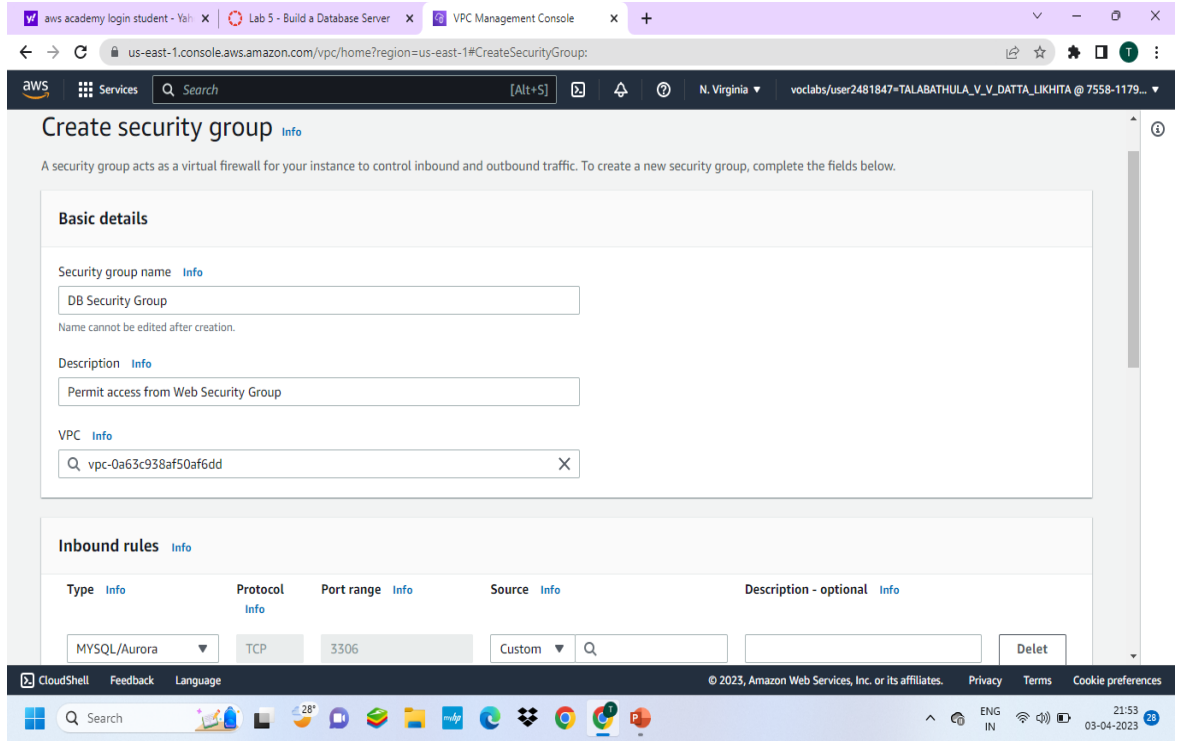


AWS RDS

Step 1: Create a Security Group for the RDS DB Instance.

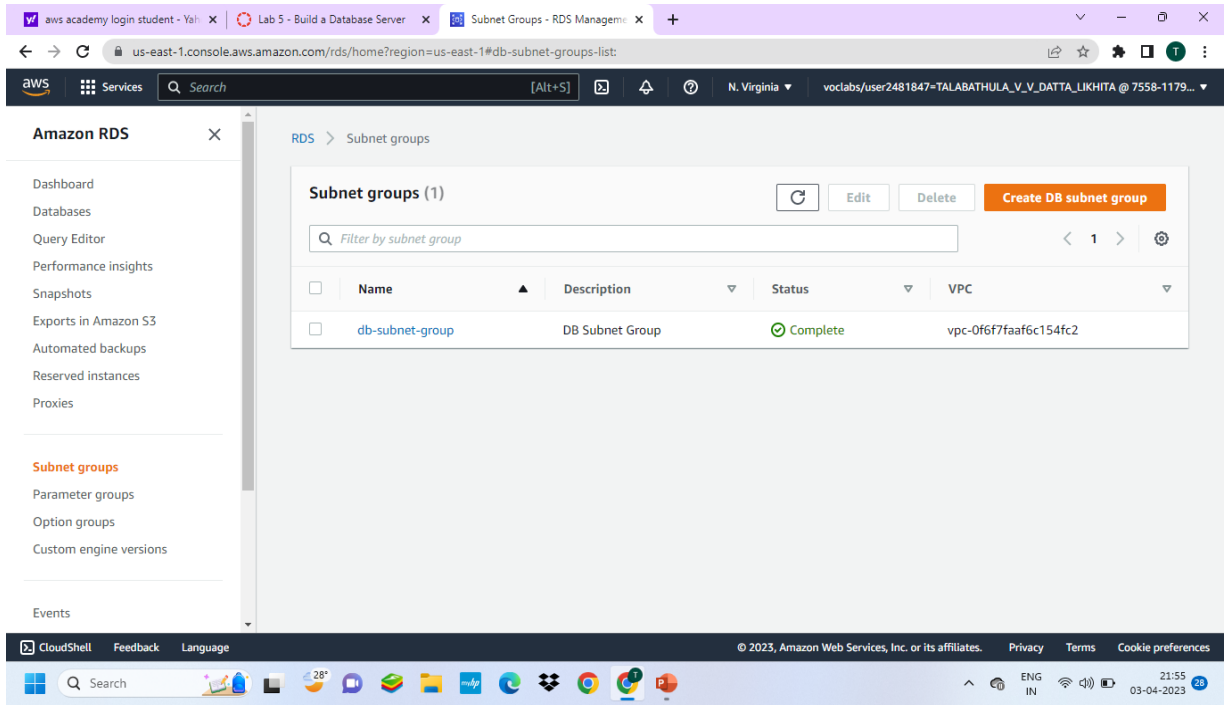
aws management console → vpc → security groups → choose create security group → add inbound rule → create security group.

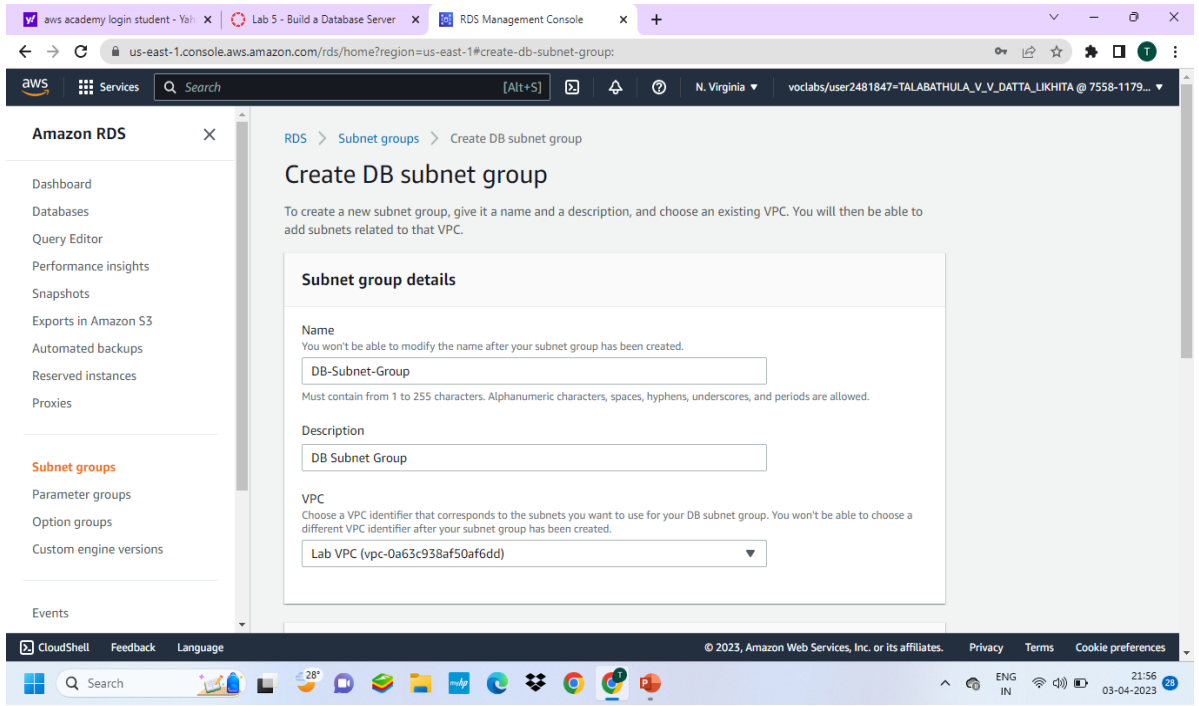




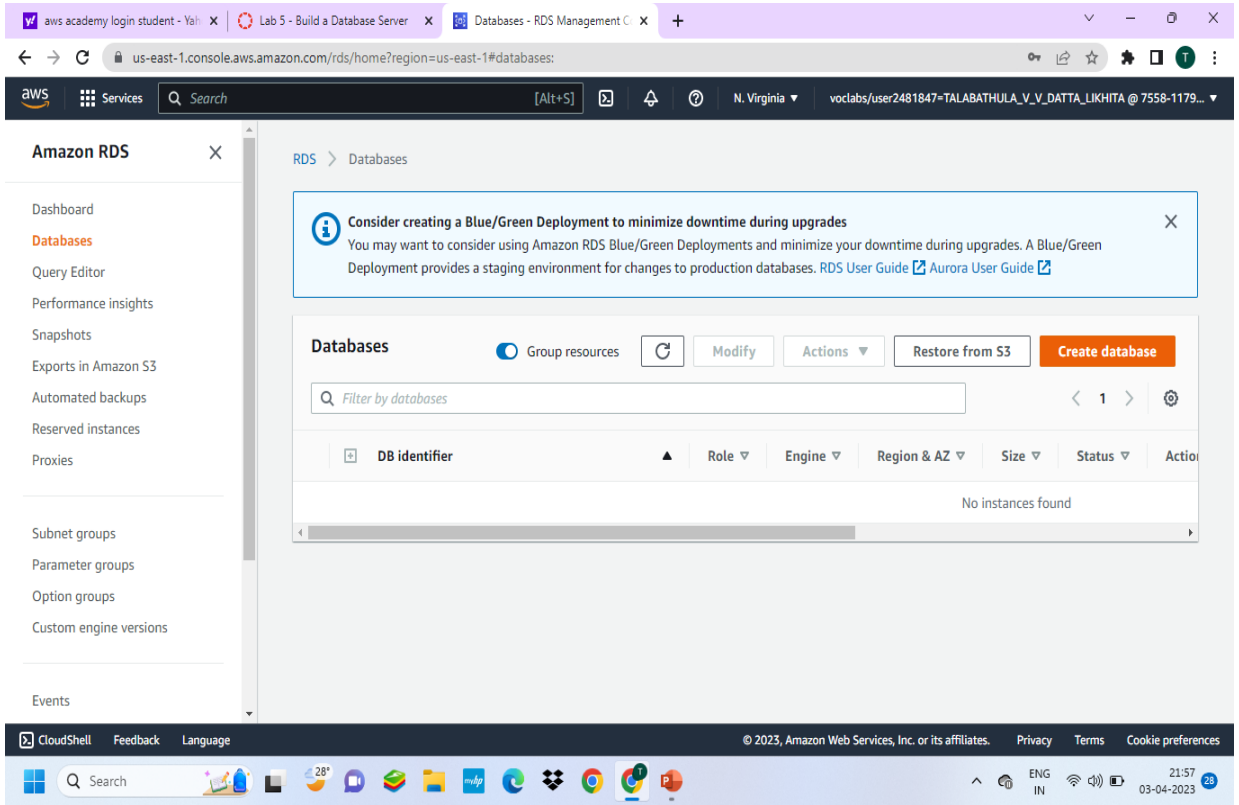
Step 2 : Create a DB Subnet Group.

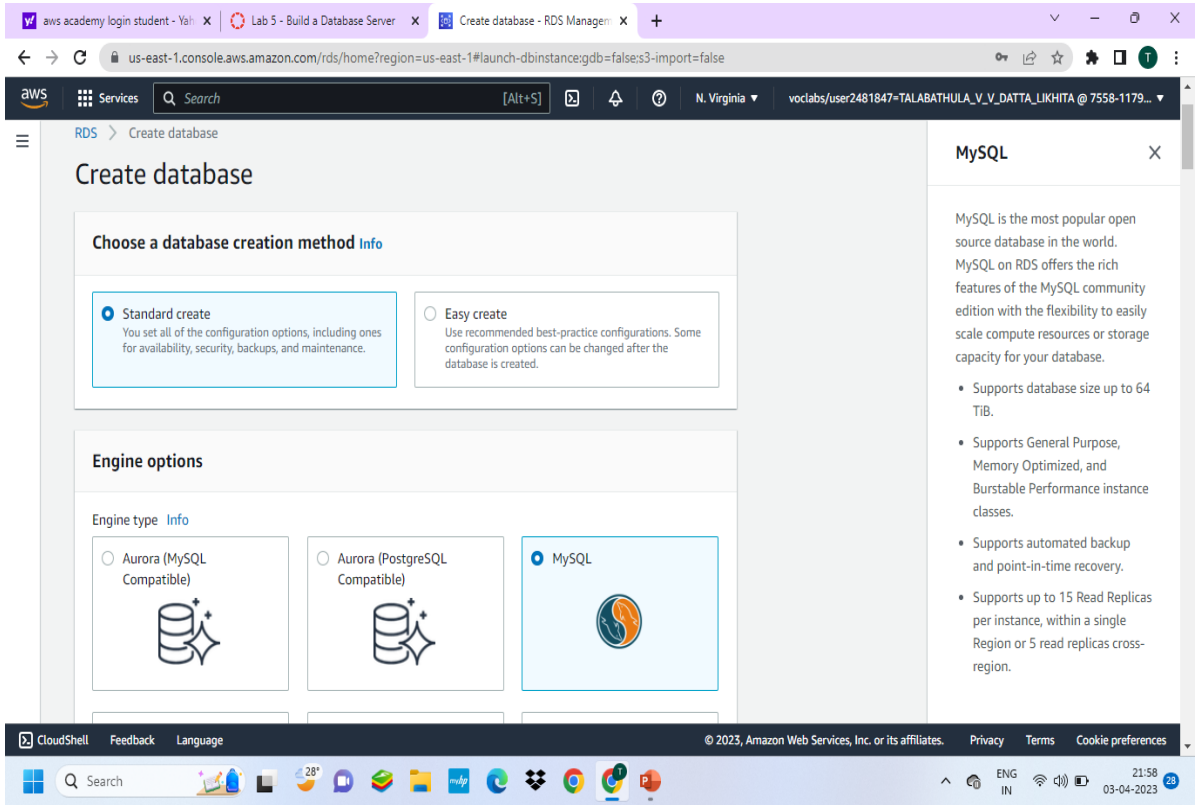
Rds → subnet groups → choose create DB subnet group → add subnets → create DB subnet group.



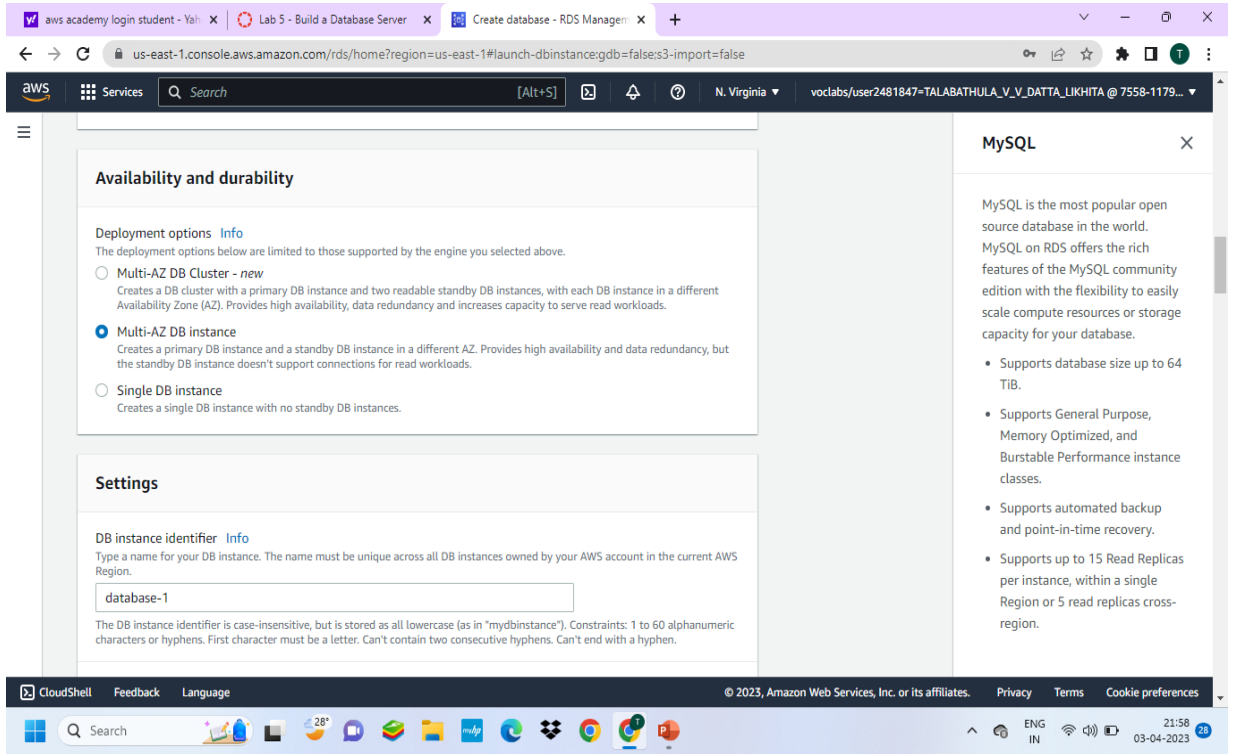


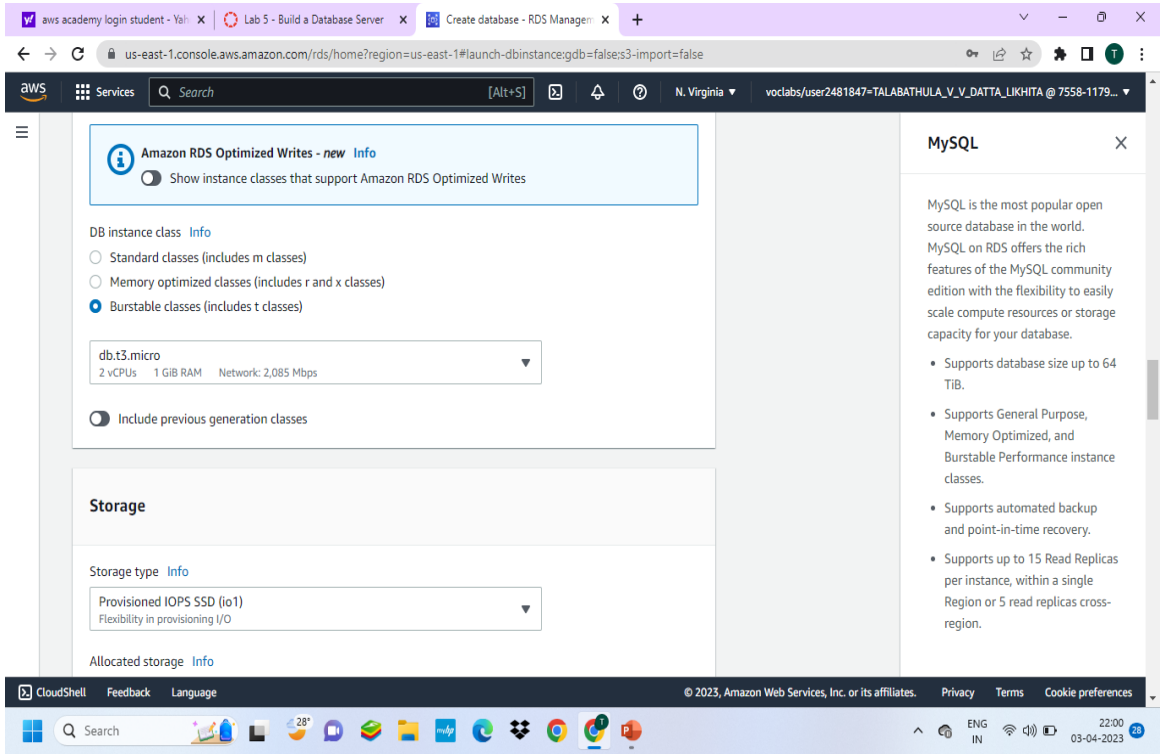
Step 3: In the left navigation pane, choose **Databases** → choose create database → MYSQL



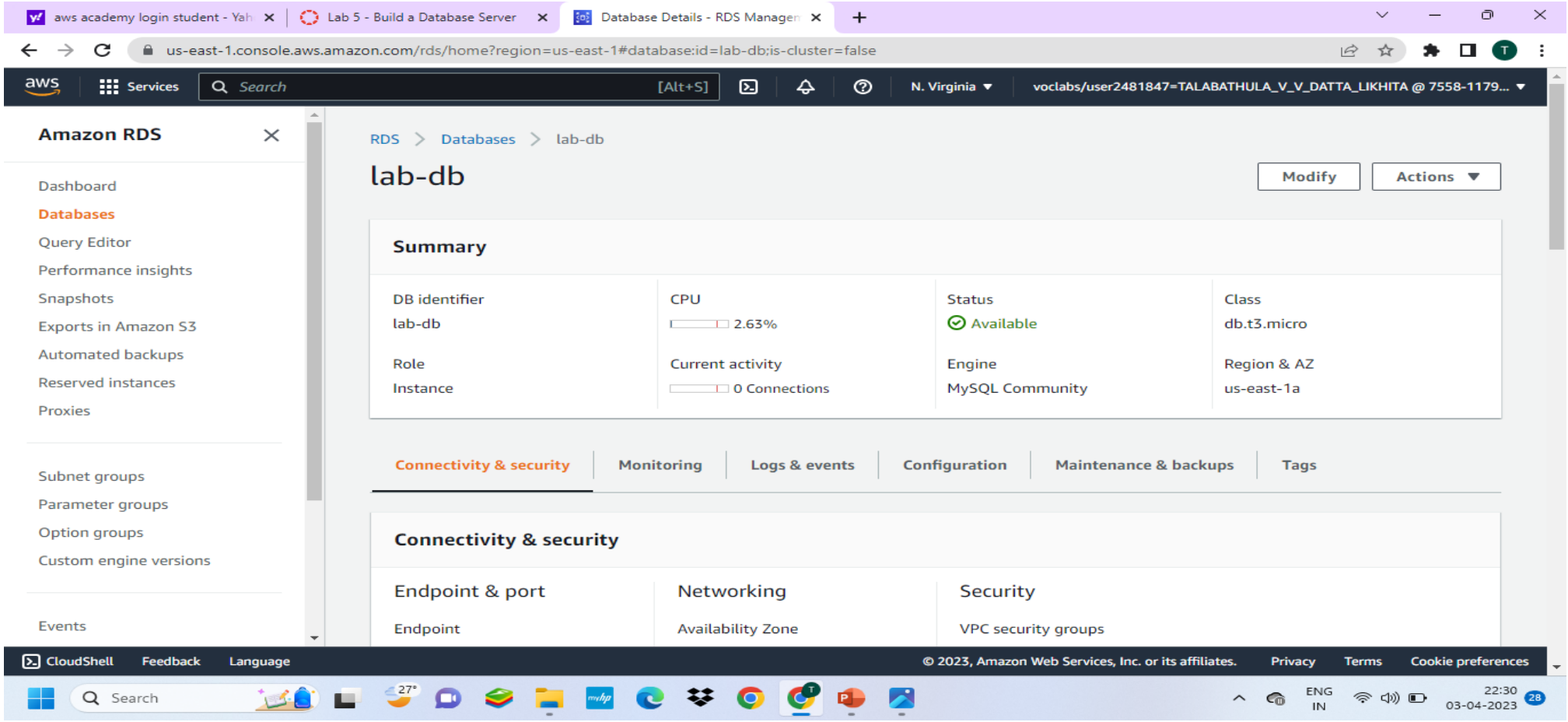


Step 4: In Availability and durability ,choose Multi –AZ DB instance then configure settings , DB instance class, Storage, connectivity, choose existing vpc security group and setup additional configuration.





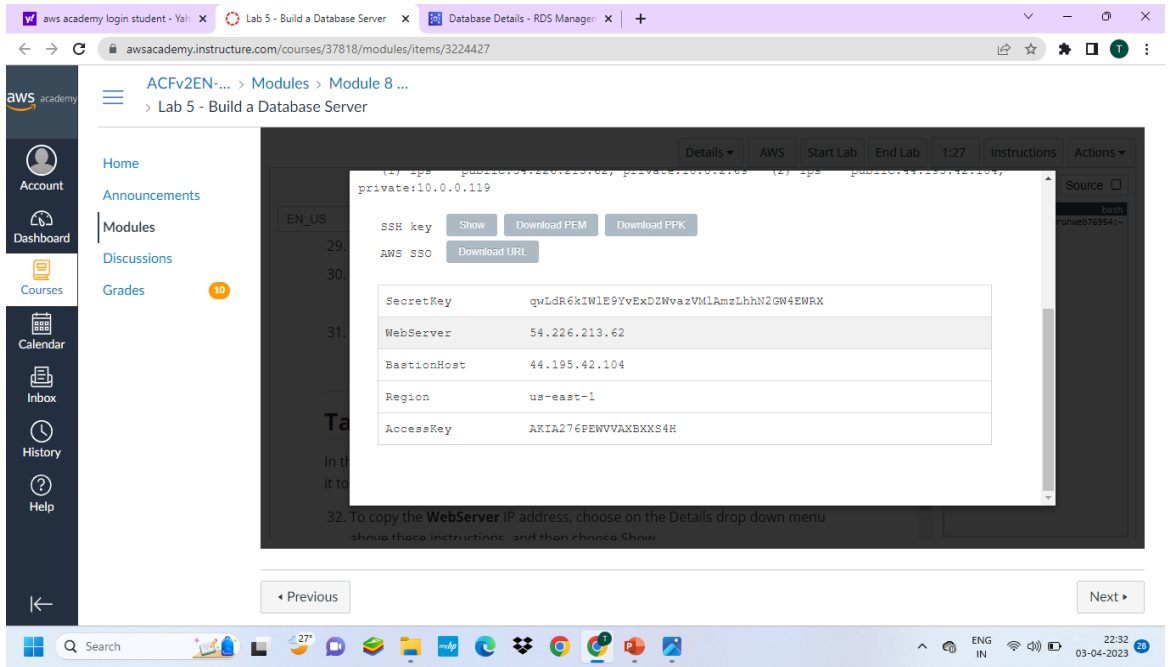
Step 5: Wait until Info changes to Modifying or Available.
Scroll down to the Connectivity & security section and copy the **Endpoint** field.

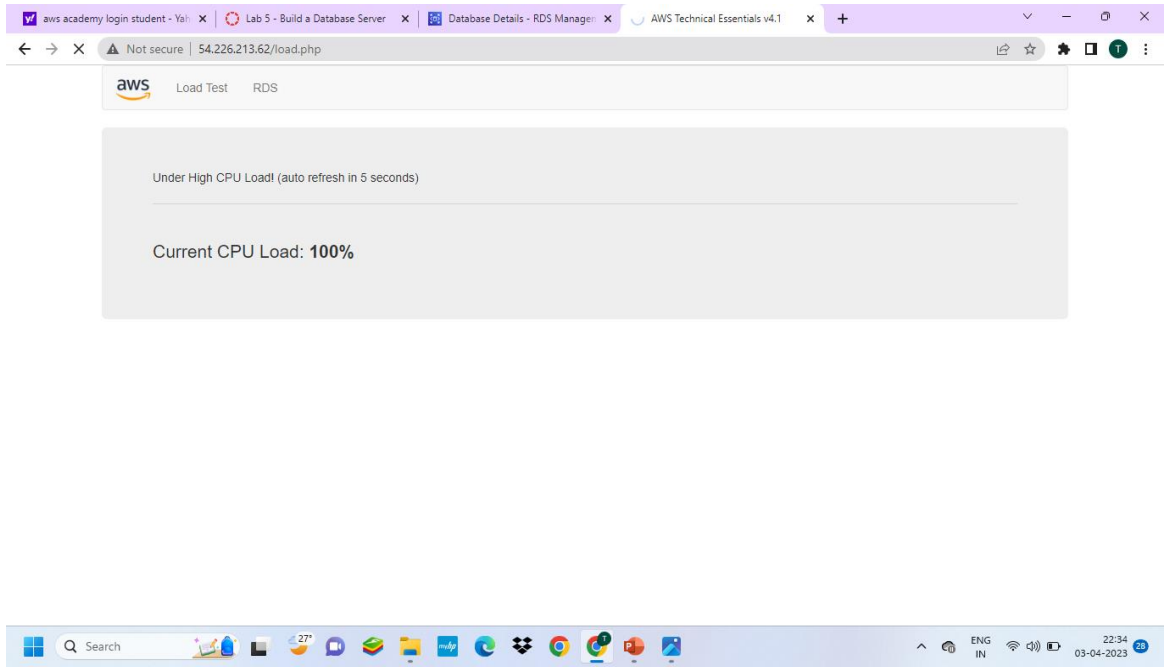


Step 6 : Interact with Your Database.

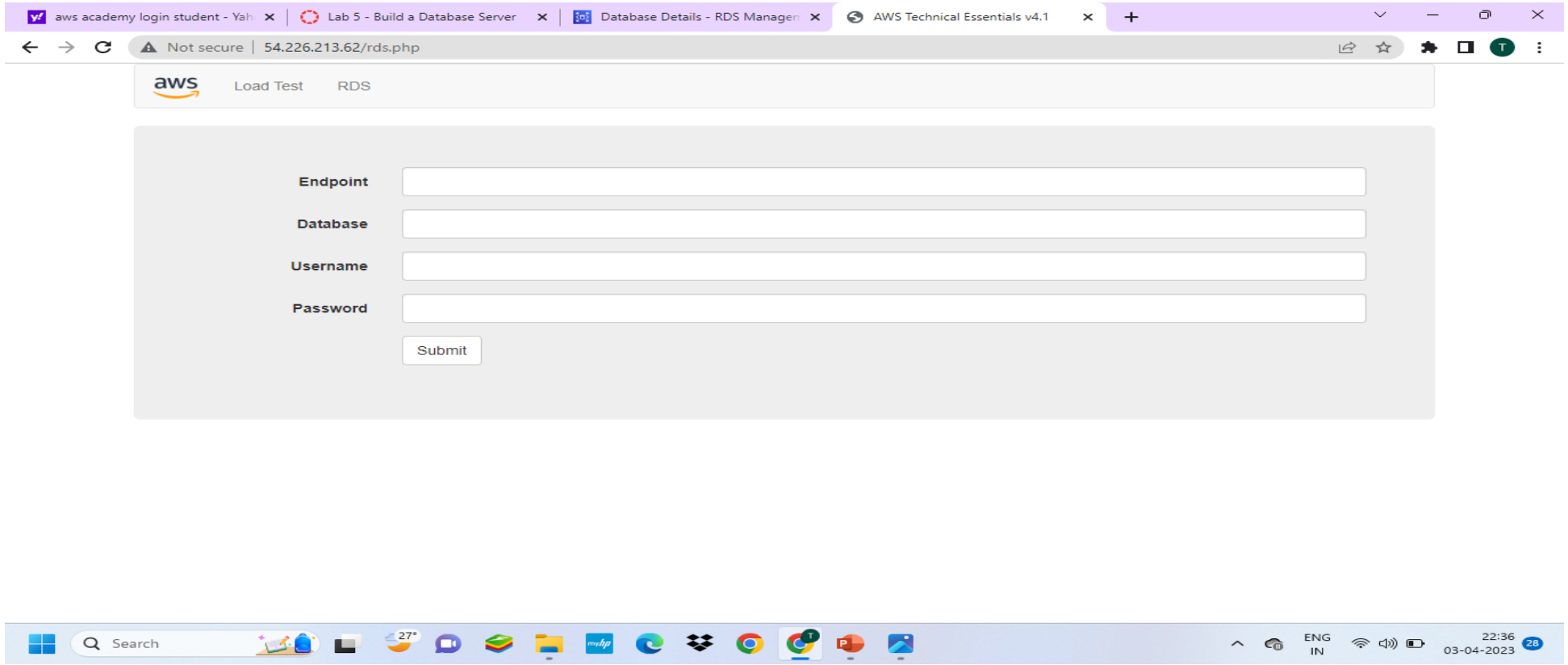
On Details , copy the **WebServer** IP address. Open a new web browser tab, paste the WebServer IP address and press Enter.

The web application will be displayed, showing information about the EC2 instance.

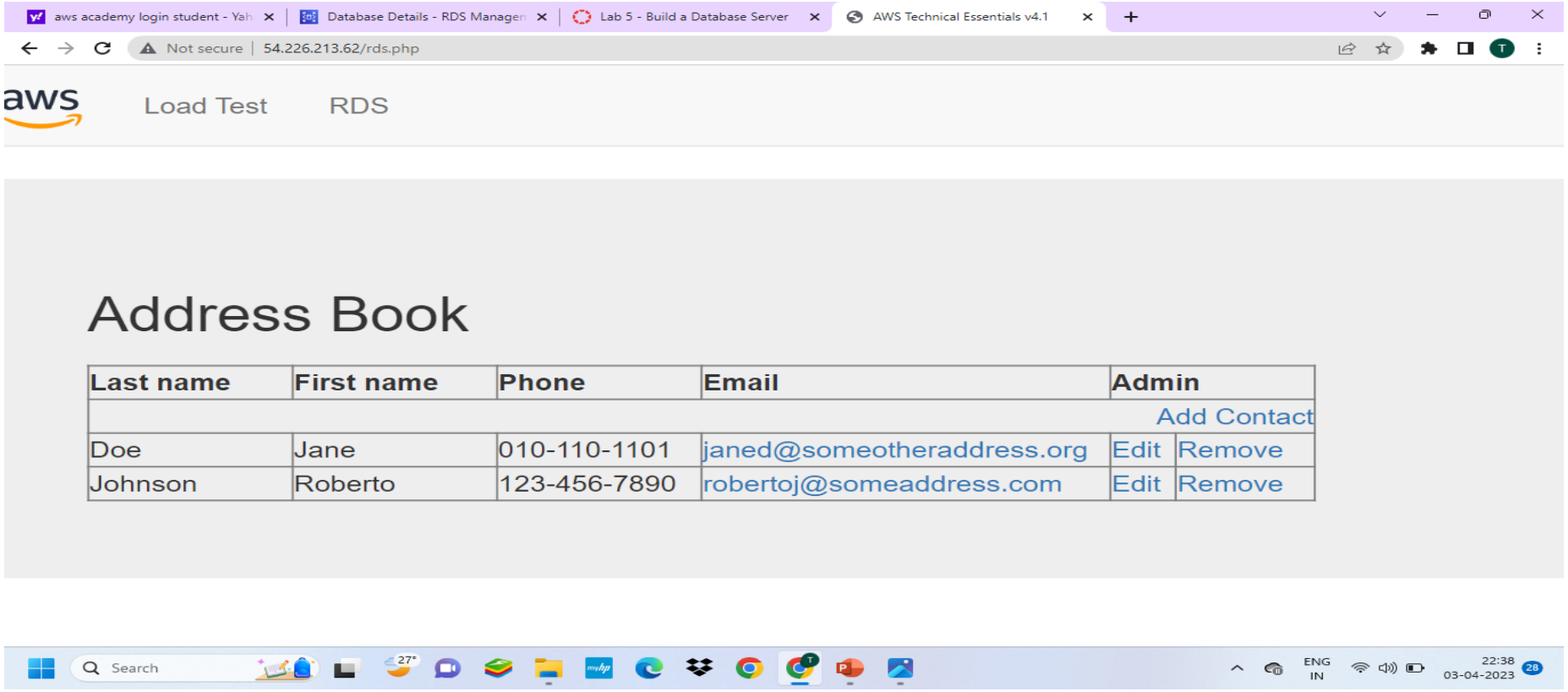




Step 7 : Choose the **RDS** link at the top of the page and configure the settings.

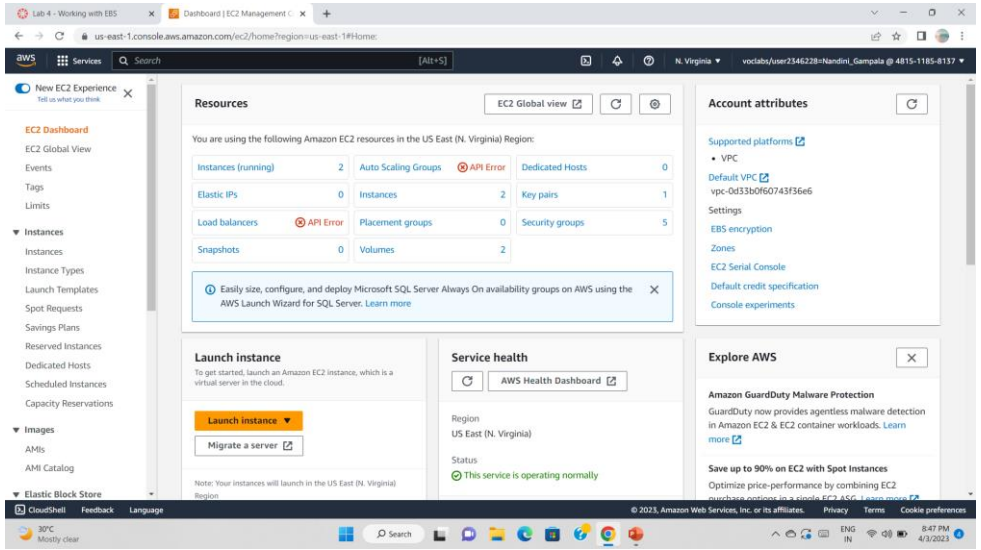


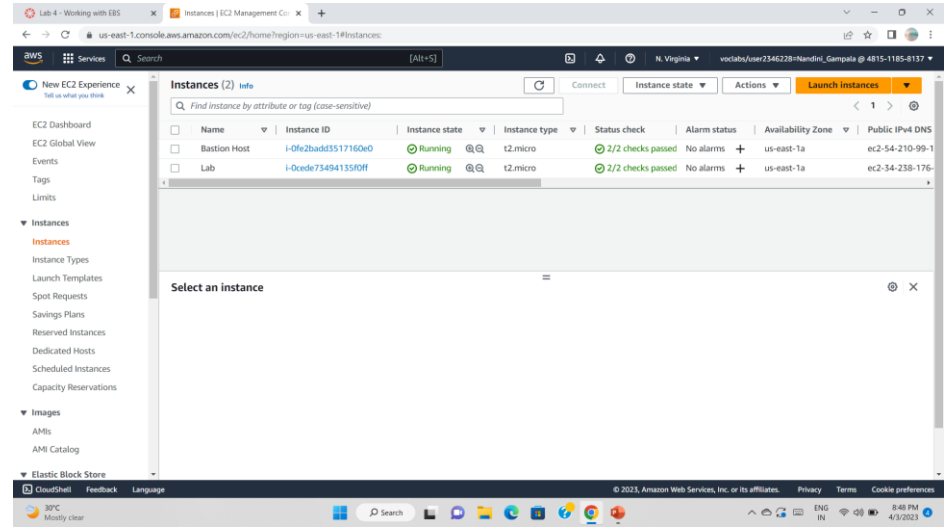
Step 8: After a few seconds the application will display an **Address Book**.
The Address Book application is using the RDS database to store information.

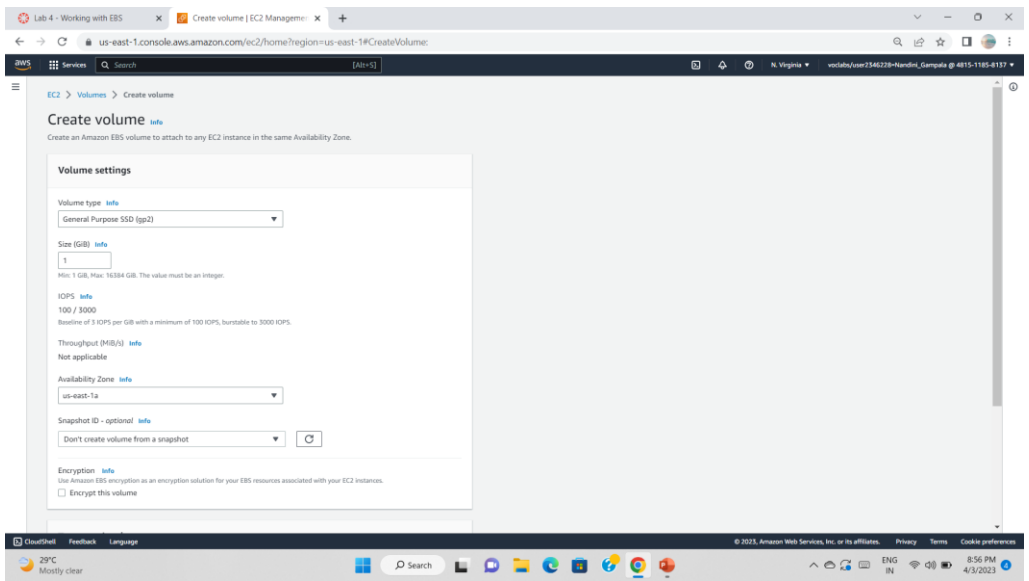


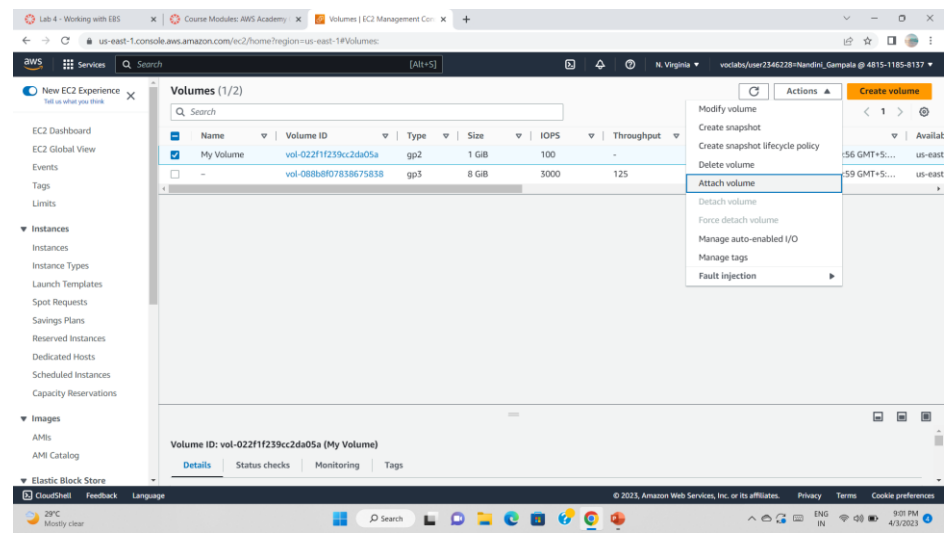
ELASTIC BLOCK SERVICE (EBS)

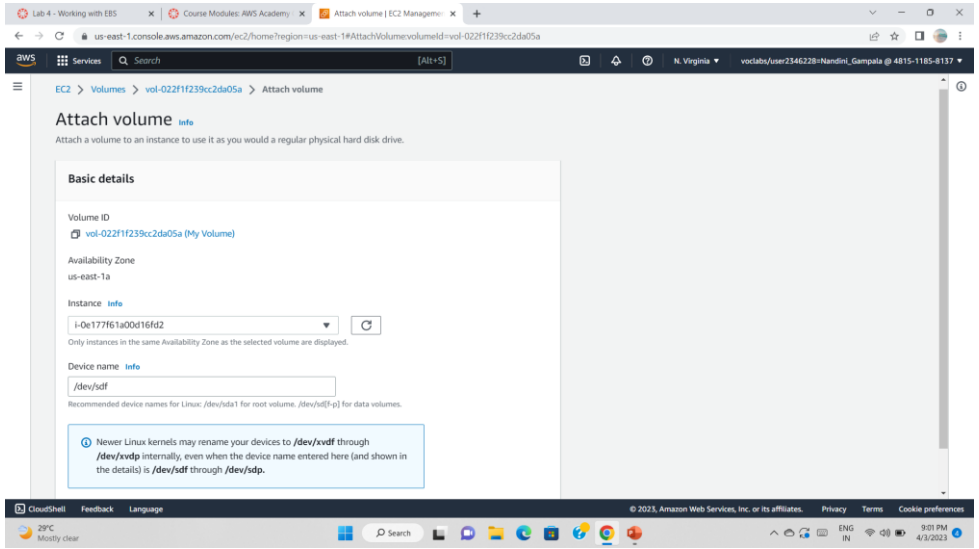
1. Open Management Console, on the services menu open Ec2
2. In the left navigation pane choose instances and create a instance with a name
3. Next, In the left navigation pane choose Volumes
4. Click on Create Volume
5. Select volume type, size(Gib),Availability Zone and in Add tag section add key and value names.
6. Then click on create volume
7. Click on volumes on left navigation pane select the created volume and attach a previously created instance to it.
8. Then, go to “Details” drop down, choose “show”
9. Download the ppk file
10. Download putty
11. Open putty set the fields (such as Host name, public key, private key) as per the requirement.
12. The putty shell will open , then login into it and run the commands.
13. The commands looks like:
    df –h
    sudo mkfs –t ext3/dev/sdf etc.,
14. Create a EBS snapshot by giving the necessary fields.
15. Create a volume using snapshot.
16. Attach the volume to the created EC2 instance

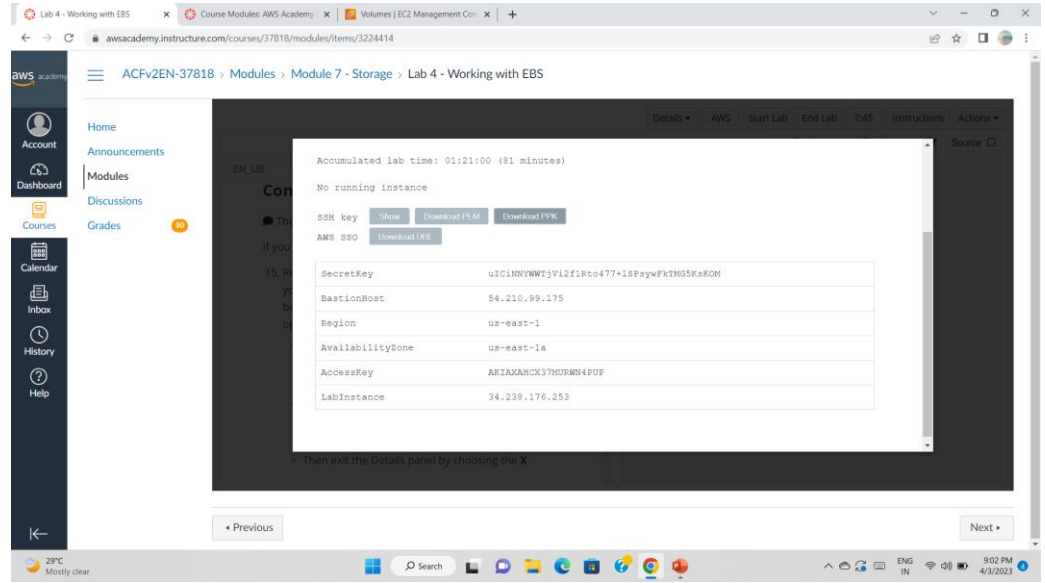

Lab 4 - Working with EBS
Dashboard | EC2 Management
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Home:
Resources
EC2 Global view
You are using the following Amazon EC2 resources in the US East (N. Virginia) Region:
Instances (running) 2
Auto Scaling Groups API Error
Dedicated Hosts 0
Elastic IPs 0
Instances 2
Key pairs 1
Load balancers API Error
Placement groups 0
Security groups 5
Snapshots 0
Volumes 2
Account attributes
Supported platforms
VPC
Default VPC
vpc-0d33b0f60743f36e6
Settings
EBS encryption
Zones
EC2 Serial Console
Default credit specification
Console experiments
Launch instance
Migrate a server
Service health
AWS Health Dashboard
Region
US East (N. Virginia)
Status
This service is operating normally
Explore AWS
Amazon GuardDuty Malware Protection
Save up to 90% on EC2 with Spot Instances

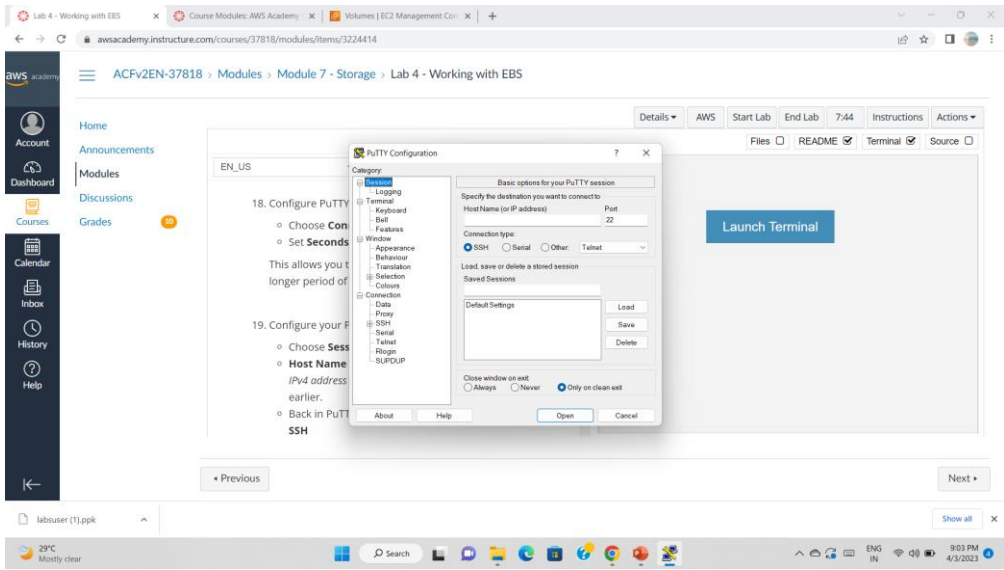

Lab 4 - Working with EBS
Instances | EC2 Management
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Instances:
Instances (2)
Launch instances
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public IPv4 DNS
Bastion Host
i-0fe2badd3517160e0
Running
t2.micro
2/2 checks passed
No alarms
us-east-1a
ec2-54-210-99-1
Lab
i-0cede73494135f0ff
Running
t2.micro
2/2 checks passed
No alarms
us-east-1a
ec2-34-238-176-
Select an instance

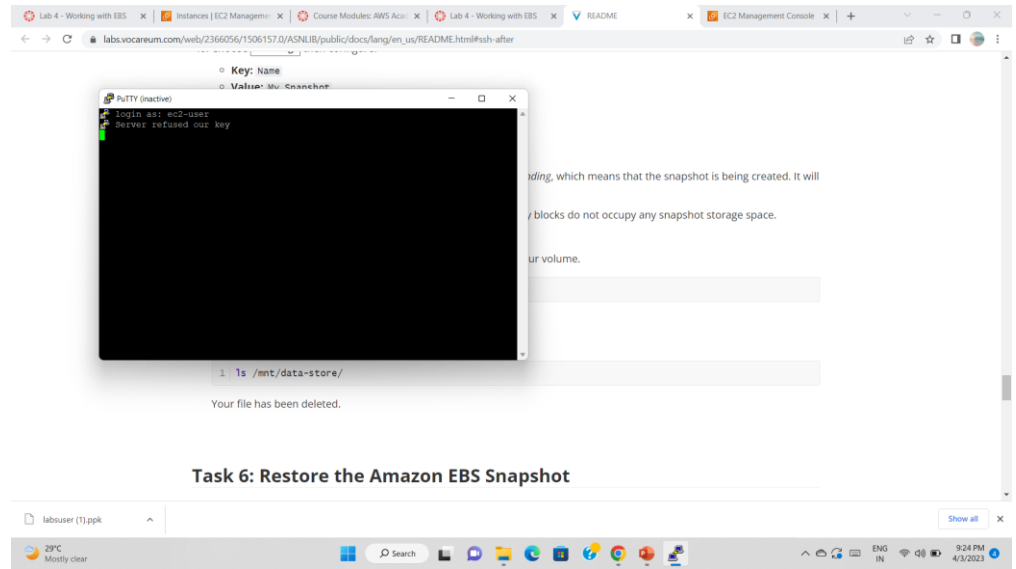

Lab 4 - Working with EBS
Create volume | EC2 Management
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateVolume:
EC2 > Volumes > Create volume
Create volume
Create an Amazon EBS volume to attach to any EC2 instance in the same Availability Zone.
Volume settings
Volume type
General Purpose SSD (gp2)
Size (GiB)
Min: 1 GiB, Max: 16384 GiB. The value must be an integer.
IOPS
100 / 3000
Baseline of 3 IOPS per GiB with a minimum of 100 IOPS, burstable to 3000 IOPS.
Throughput (MiB/s)
Not applicable
Availability Zone
us-east-1a
Snapshot ID - optional
Don't create volume from a snapshot
Encryption
Use Amazon EBS encryption as an encryption solution for your EBS resources associated with your EC2 instances.
Encrypt this volume

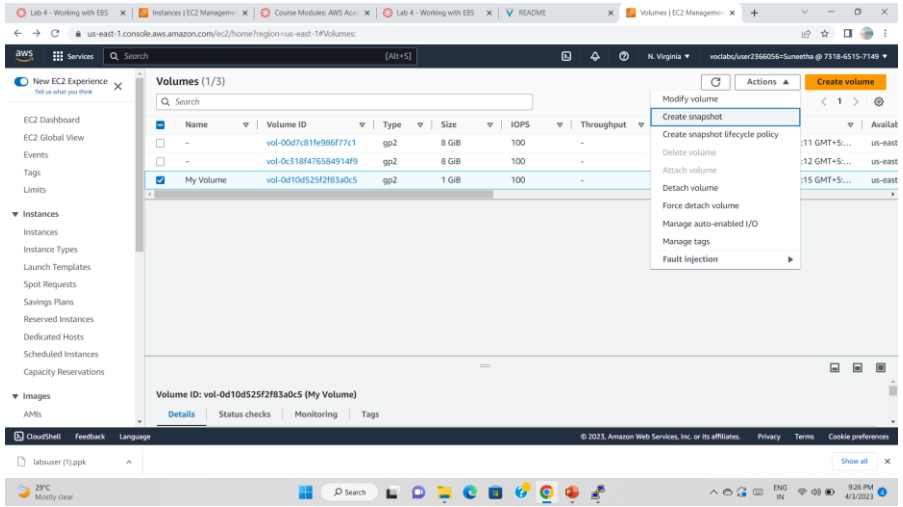

Lab 4 - Working with EBS
Course Modules: AWS Academy
Volumes | EC2 Management
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Volumes:
Volumes (1/2)
Actions
Create volume
Name
Volume ID
Type
Size
IOPS
Throughput
My Volume
vol-022f1f239cc2da05a
gp2
1 GiB
100
vol-088b8f07838675838
gp3
8 GiB
3000
125
Modify volume
Create snapshot
Create snapshot lifecycle policy
Delete volume
Attach volume
Detach volume
Force detach volume
Manage auto-enabled I/O
Manage tags
Fault injection
Volume ID: vol-022f1f239cc2da05a (My Volume)
Details
Status checks
Monitoring
Tags

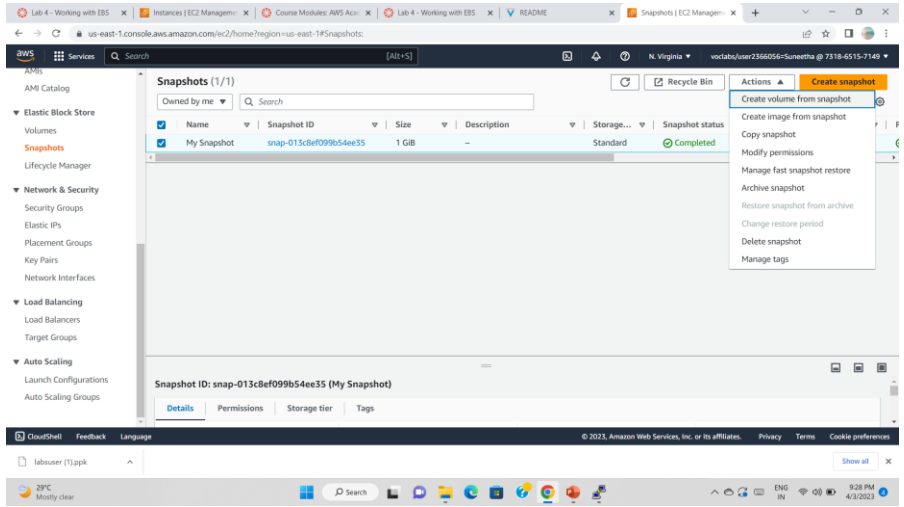

Attach volume
Attach a volume to an instance to use it as you would a regular physical hard disk drive.
Basic details
Volume ID
vol-022f1f239cc2da05a (My Volume)
Availability Zone
us-east-1a
Instance
i-0e177f61a00d16fd2
Device name
/dev/sdf
Newer Linux kernels may rename your devices to /dev/xvdf through /dev/xvdp internally, even when the device name entered here (and shown in the details) is /dev/sdf through /dev/sdp.

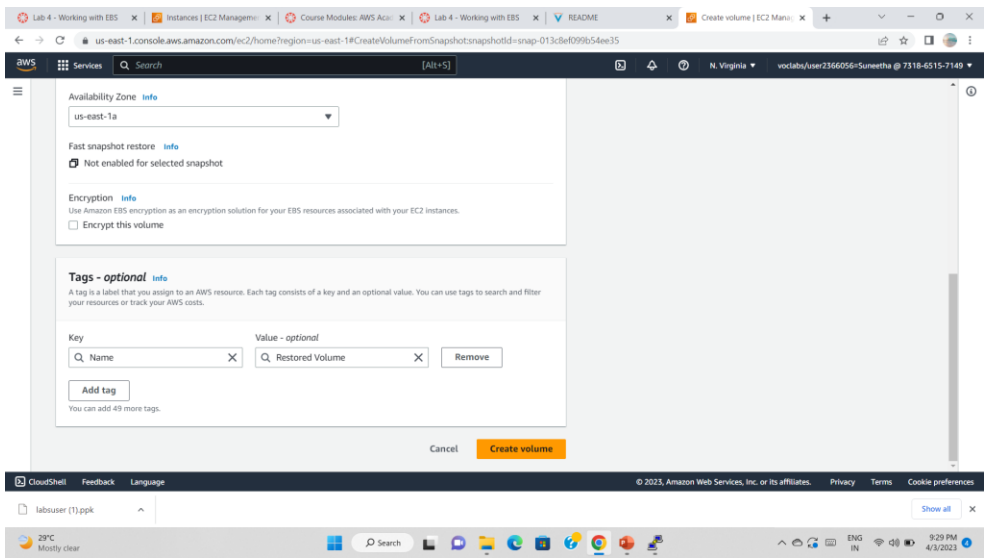

ACFv2EN-37818 › Modules › Module 7 - Storage › Lab 4 - Working with EBS
Accumulated lab time: 01:21:00 (81 minutes)
No running instance
SSH key
AWS SSO
SecretKey
BastionHost 54.210.99.175
Region us-east-1
AvailabilityZone us-east-1a
AccessKey
LabInstance 34.238.176.253
Previous
Next

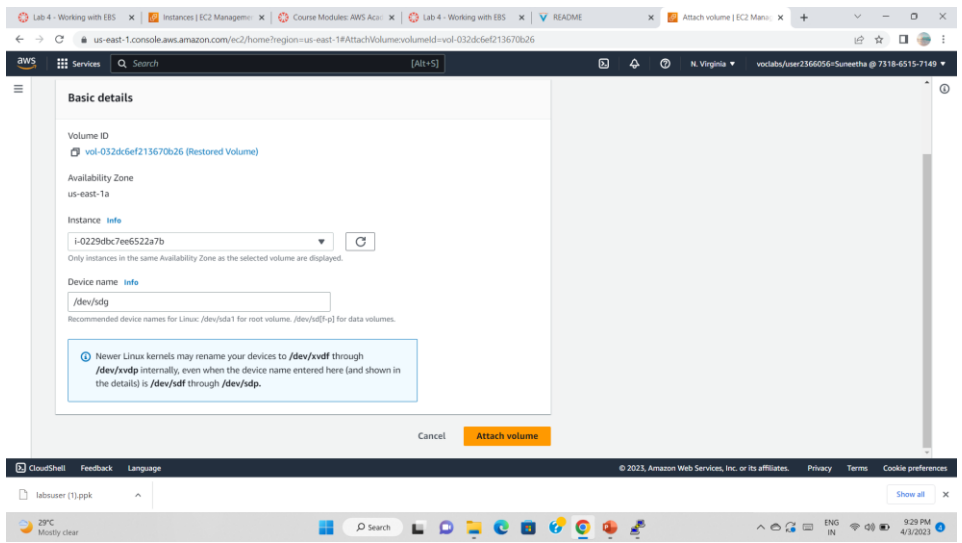

ACFv2EN-37818 › Modules › Module 7 - Storage › Lab 4 - Working with EBS
Launch Terminal
PuTTY Configuration
Previous
Next


PuTTY (inactive)
login as: ec2-user
Server refused our key
ls /mnt/data-store/
Your file has been deleted.
Task 6: Restore the Amazon EBS Snapshot


Volumes (1/3)
Actions
Create volume
Modify volume
Create snapshot
Create snapshot lifecycle policy
Delete volume
Attach volume
Detach volume
Force detach volume
Manage auto-enabled I/O
Manage tags
Fault injection
Volume ID: vol-0d10d525f2f83a0c5 (My Volume)
Details
Status checks
Monitoring
Tags


Snapshots (1/1)
Recycle Bin
Actions
Create snapshot
Create volume from snapshot
Create image from snapshot
Copy snapshot
Modify permissions
Manage fast snapshot restore
Archive snapshot
Restore snapshot from archive
Change restore period
Delete snapshot
Manage tags
Snapshot ID: snap-013c8ef099b54ee35 (My Snapshot)
Details
Permissions
Storage tier
Tags


Availability Zone Info
us-east-1a
Fast snapshot restore Info
Not enabled for selected snapshot
Encryption Info
Use Amazon EBS encryption as an encryption solution for your EBS resources associated with your EC2 instances.
Encrypt this volume
Tags - optional Info
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.
Key
Value - optional
Name
Restored Volume
Remove
Add tag
You can add 49 more tags.
Cancel
Create volume


Basic details
Volume ID
vol-032dc6ef213670b26 (Restored Volume)
Availability Zone
us-east-1a
Instance Info
i-0229dbc7ee6522a7b
Only instances in the same Availability Zone as the selected volume are displayed.
Device name Info
/dev/sdg
Recommended device names for Linux: /dev/sda1 for root volume. /dev/sd[f-p] for data volumes.
Newer Linux kernels may rename your devices to /dev/xvdf through /dev/xvdp internally, even when the device name entered here (and shown in the details) is /dev/sdf through /dev/sdp.
Cancel
Attach volume

AWS S3 (SIMPLE STORAGE SERVICE)

TASKS FOR CONFIGURING S3**:**

1.Log into the AWS Management Console.
2.Create an S3 bucket.
3.Upload an object to S3 Bucket.
4.Access the object on the browser.
5.Change S3 object permissions.
6.Setup the bucket policy and permission and test the object accessibility.

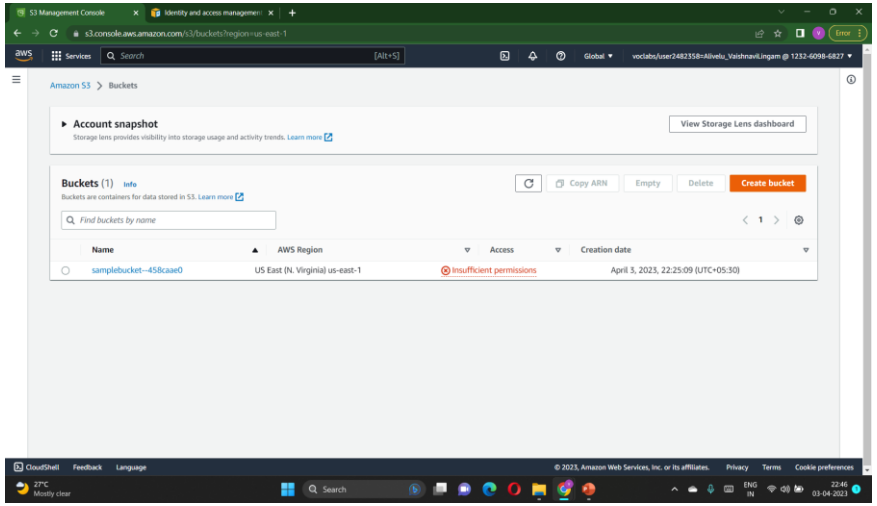
STEPS :

Step 1: Click on **create group**.
Step 2: Set up the bucket name. S3 bucket name are globally unique, choose a name which is available. Leave other settings as default and click on **create group**.
Step 3: Click on your bucket name.
Step 4: Click Upload.

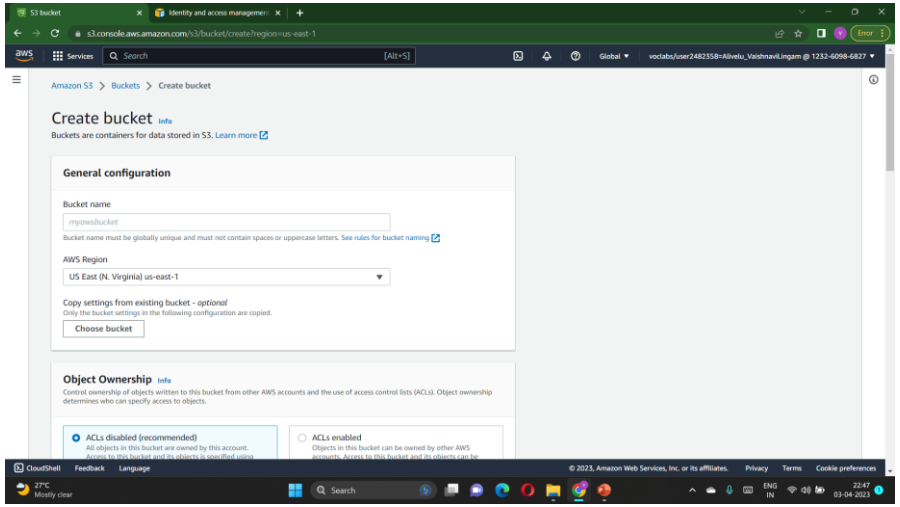
Step 5: Click on Add Files , and choose a file from your computer.
Step 6: After choosing your file, click on Next.
Step 7: Click on Upload.
Step 8: Now you have a private S3 bucket with a private object uploaded, which means you cannot visit it through Internet.
Step 9: Now you have a private S3 bucket with a private object uploaded, which means you cannot visit it through Internet.

CHANGE BUCKET PERMISSIONS:

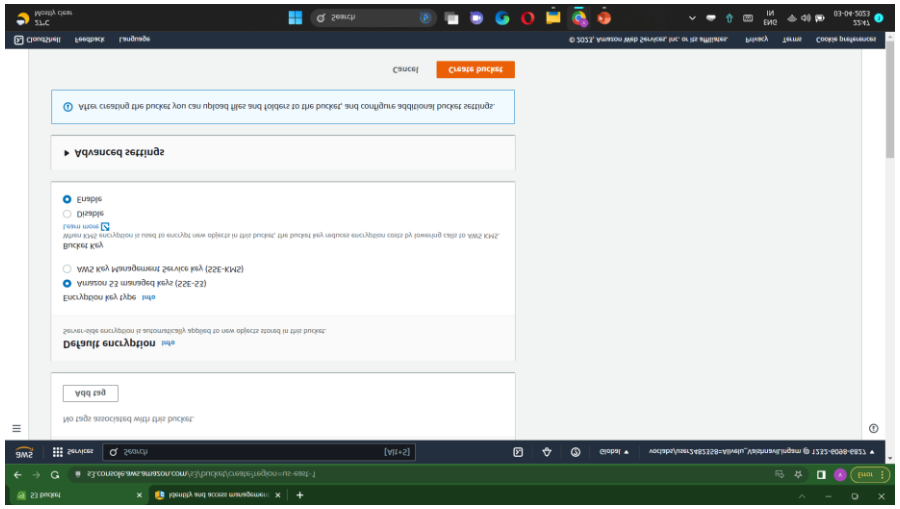
Step 10: Go back to your bcket and click on Permissions.
Step 11: Click on Everyone under the Public access, and click on Read object on the right of pop-up window. Then click on Save.
Step 12 : Now its state switches to Read Object - Yes
Step 13: Click on Overview, and click on your Object URL again .
Step 14: Notice the URL on your browser



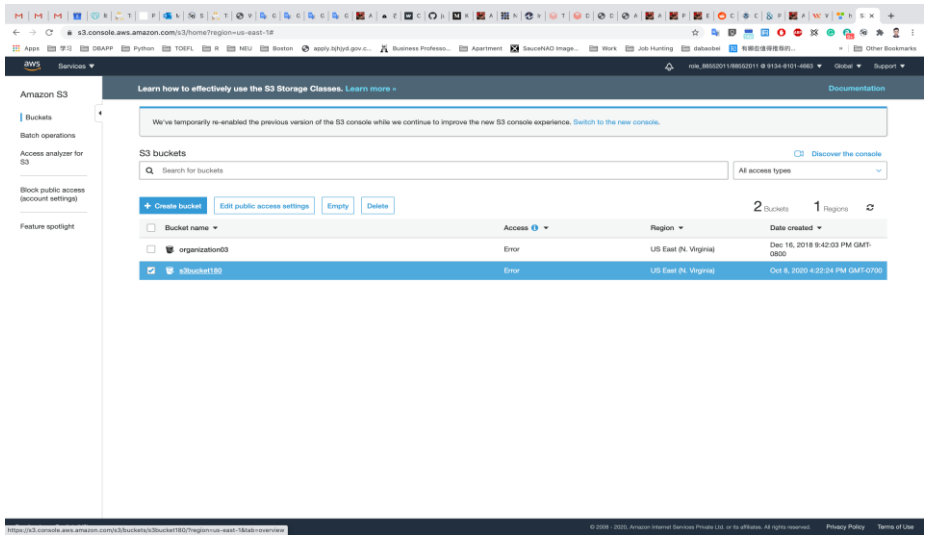
Step 1



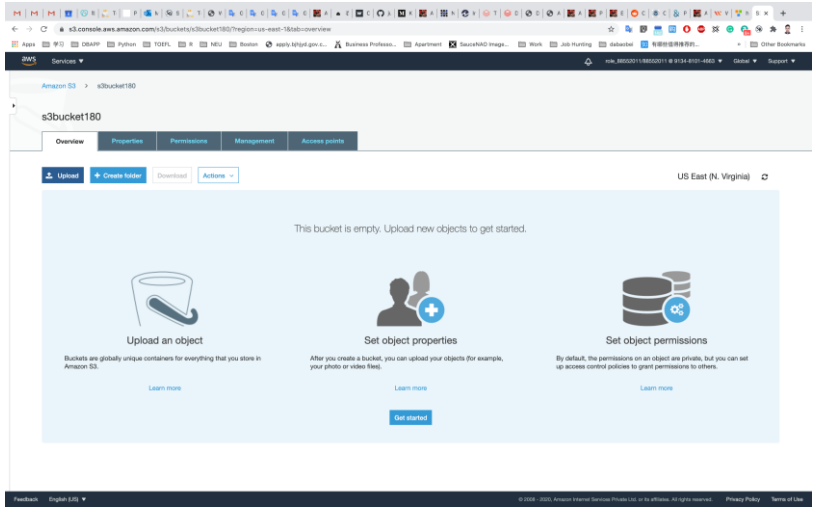
Step 2



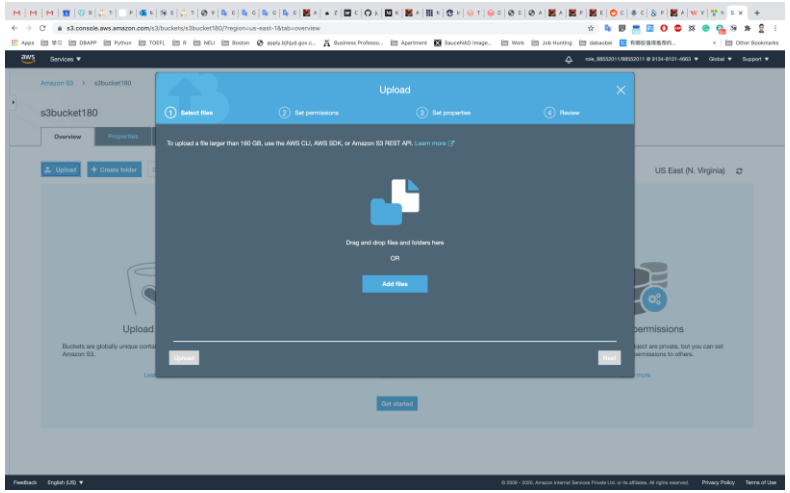
Step 2



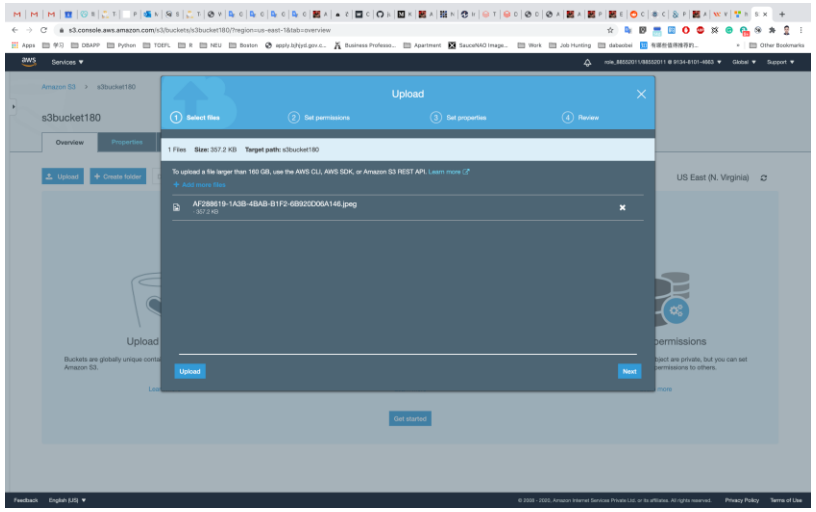
Step 3



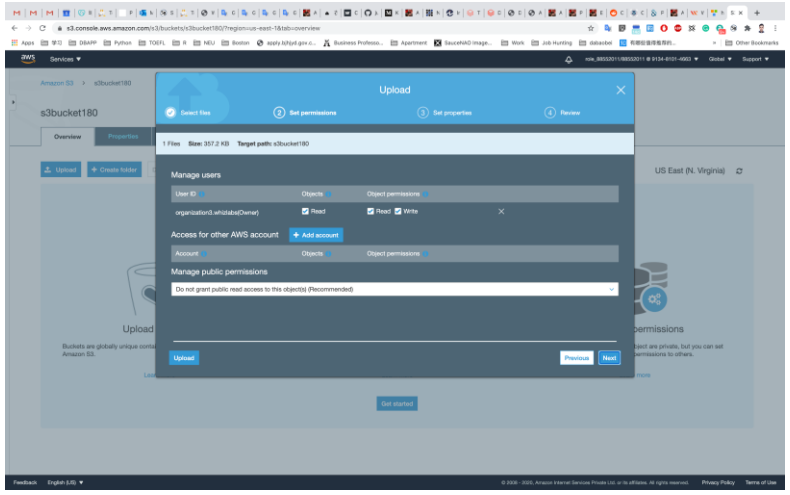
Step 4



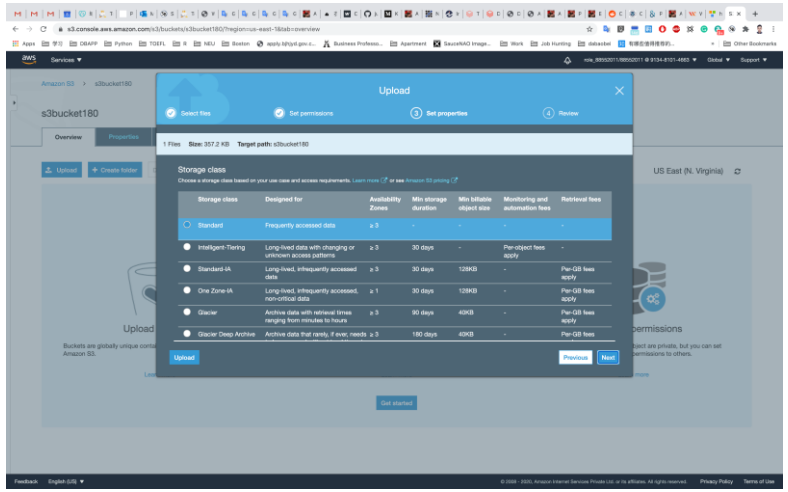
Step 5



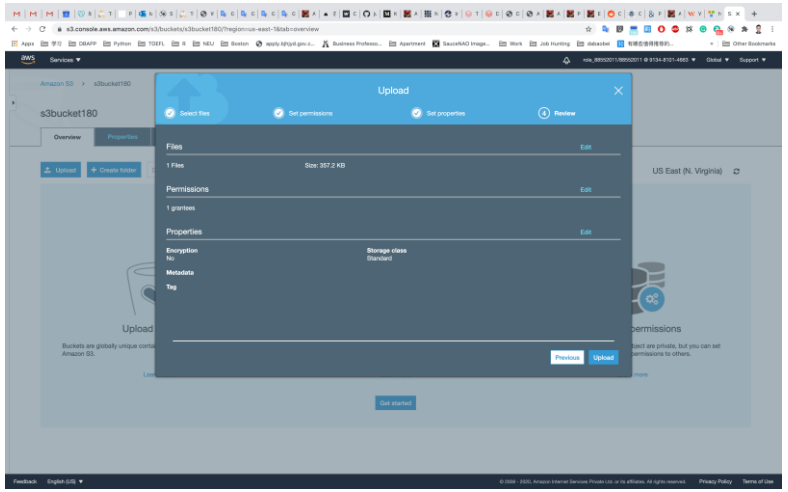
Step 6



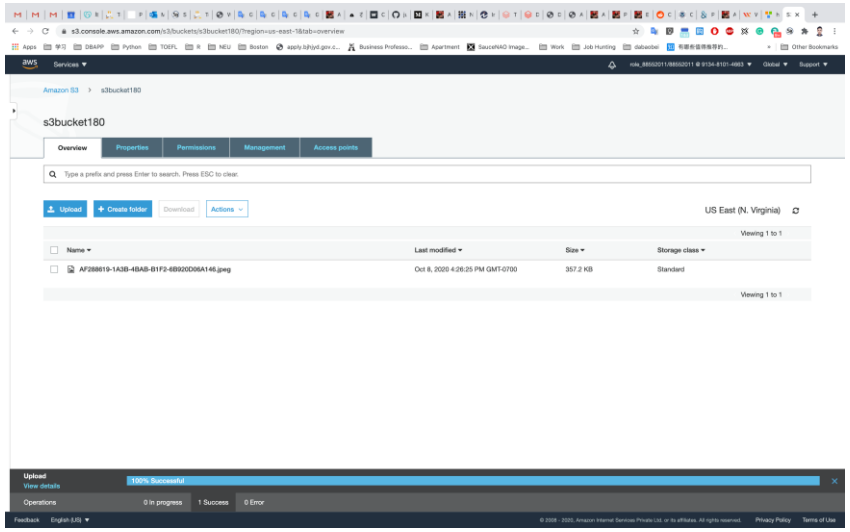
Step 7



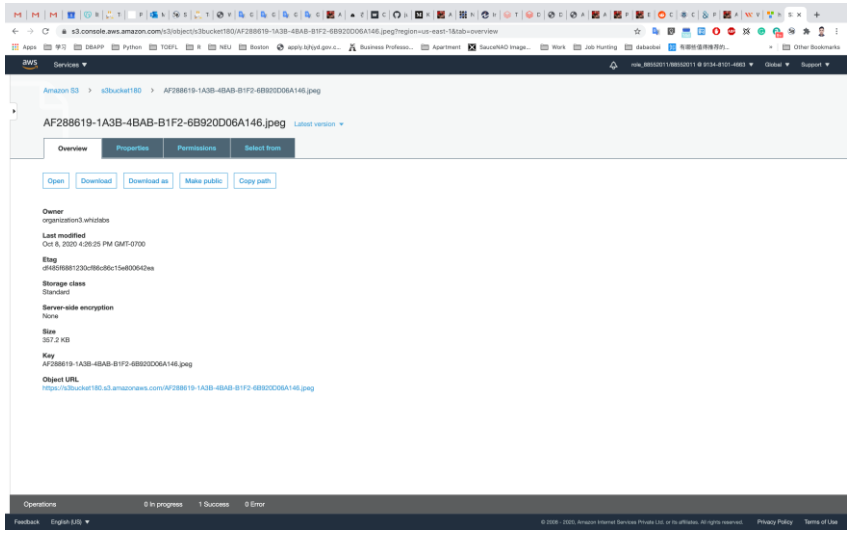
Step 8



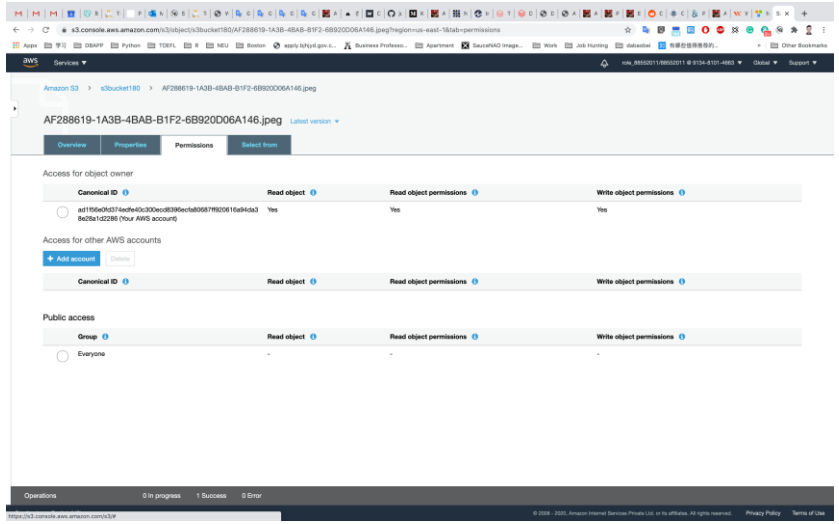
Step 9



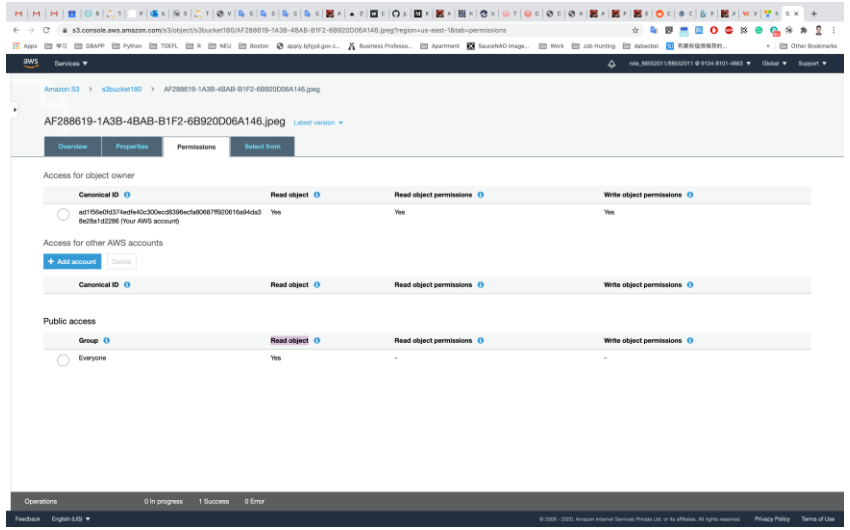
Step 10



Step 11



Step 12



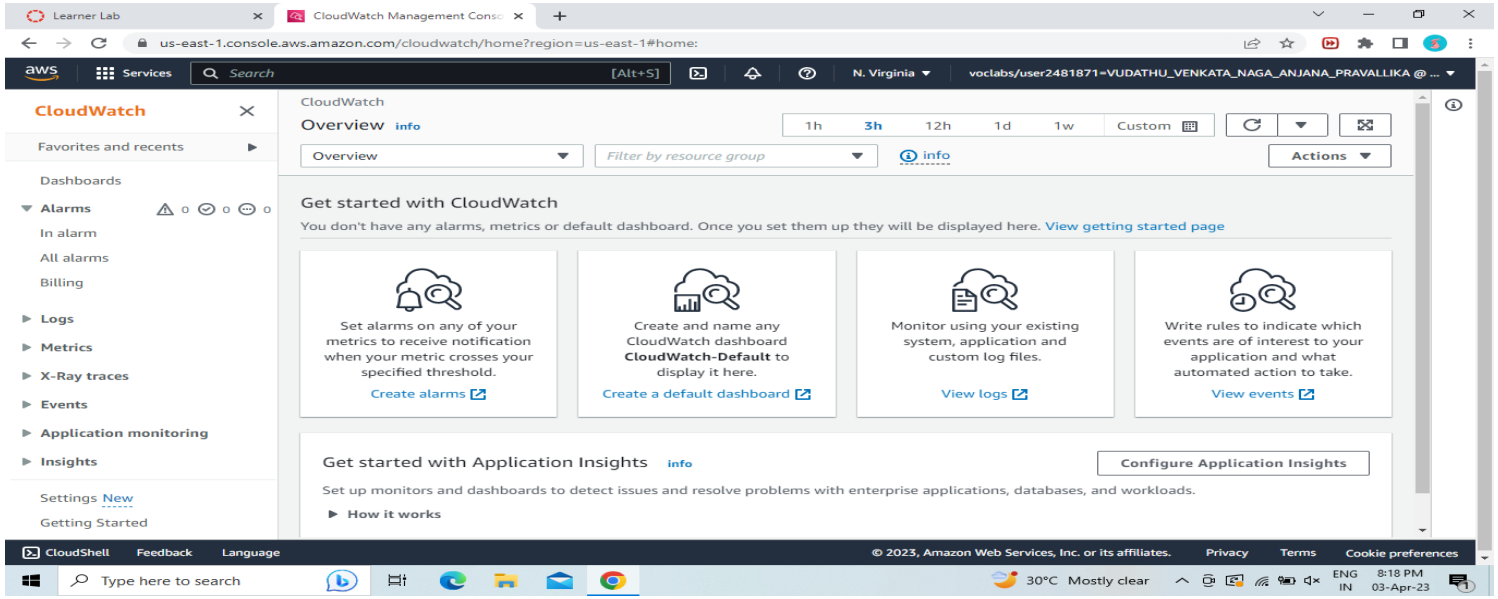
Step 13

Step 14

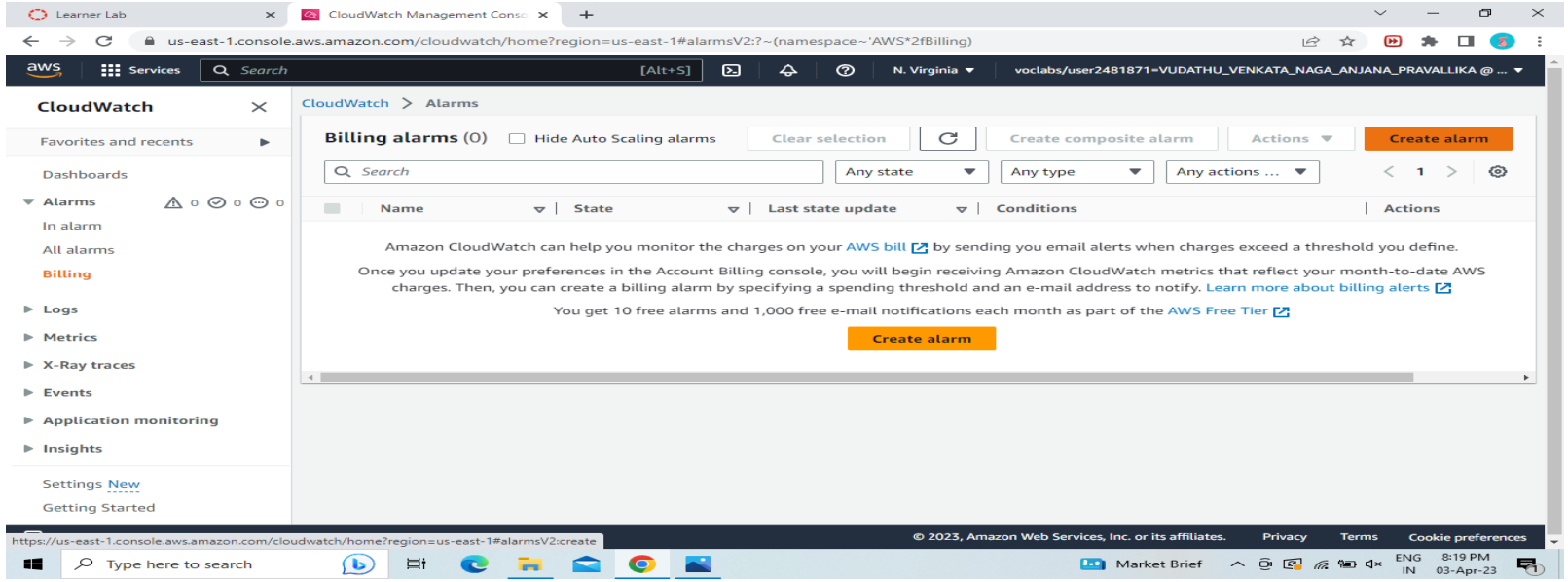
AWS CLOUDWATCH

PROCEDURE

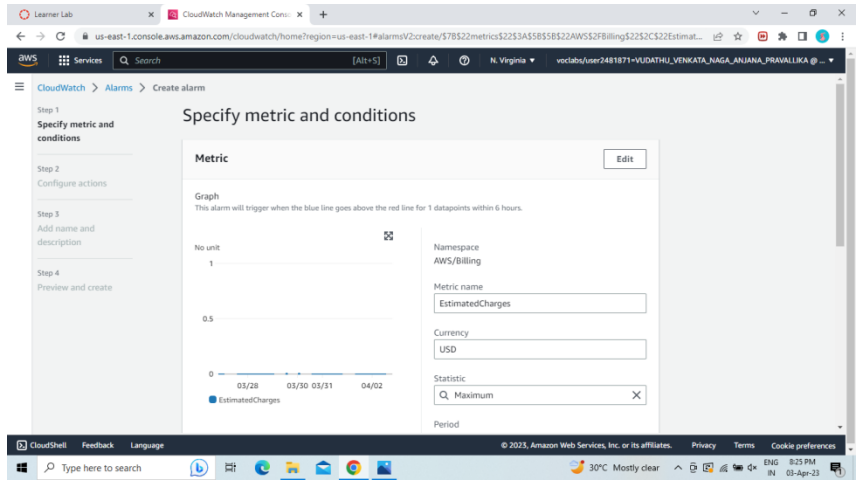
1.Go to AWS Services,Click on CloudWatch and then in the Dashboard go to Alarms section and select Billings.

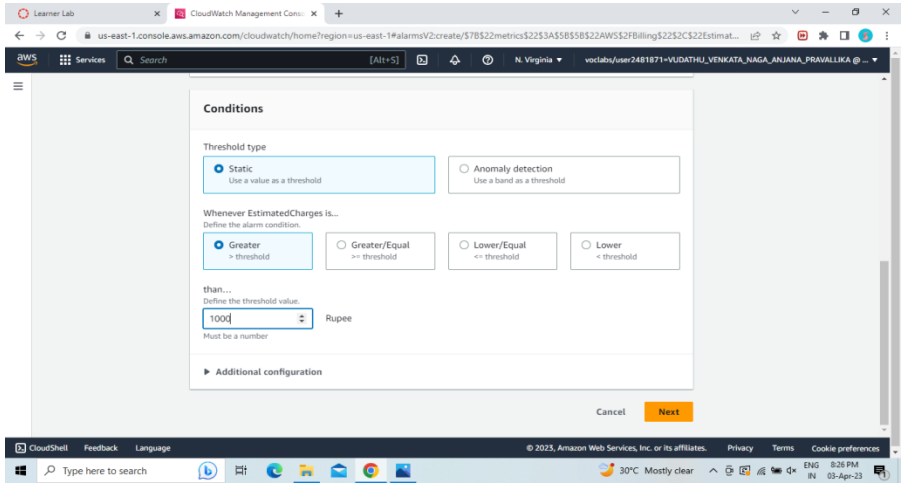


2.Then Click on CREATE ALARM.

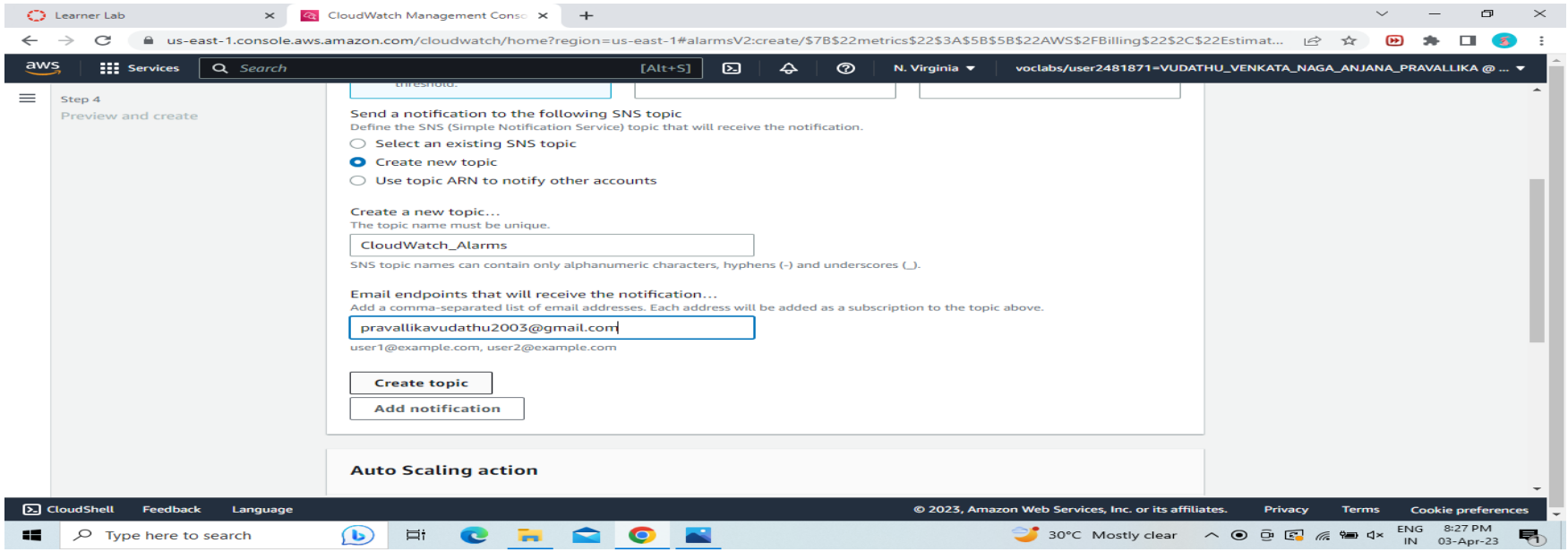


3.Then follow the steps.
In the first step it will ask us to Specify metric and conditions.Click on Select Metric.
Change the Currency to Rupee.
In the Conditions section choose the EstimatedCharges like Greater/GreaterEqual/Lowerequal/Lower and also define the threshold value.
4.Click on Next.

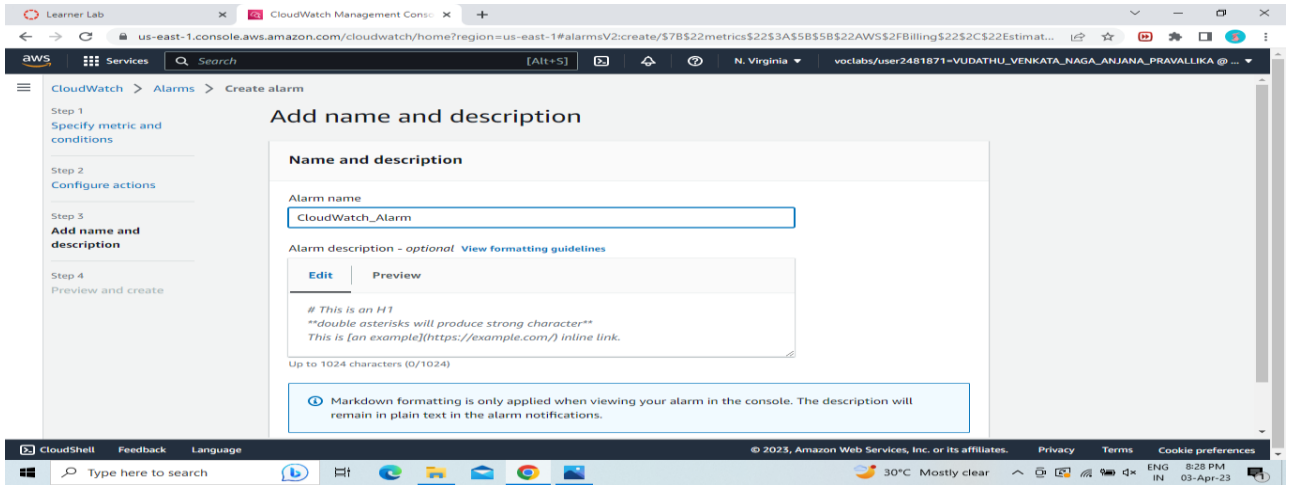




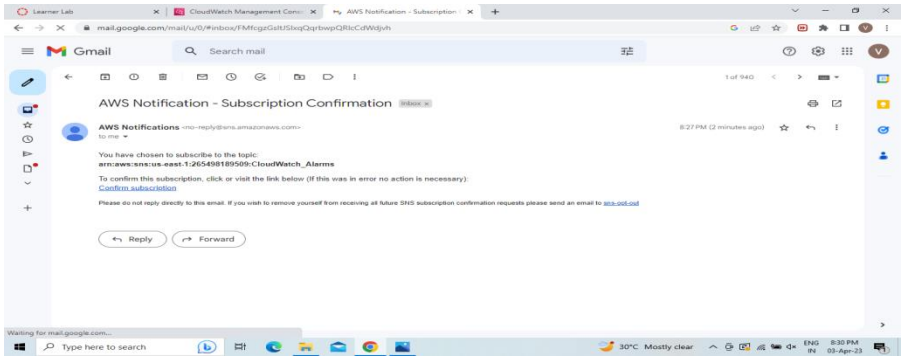
5.Now for Cofigure Actions choose Create new topic.Give a name to the topic and enter your email to receive a notification.Click on Create Topic,then Next.

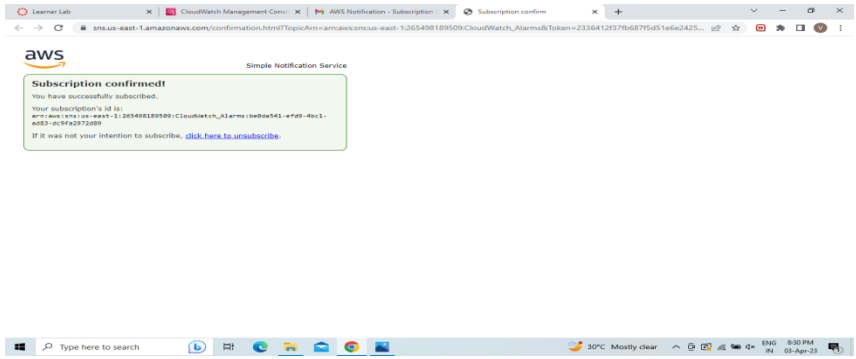


6.Give a name to your Alarm and Click on next.

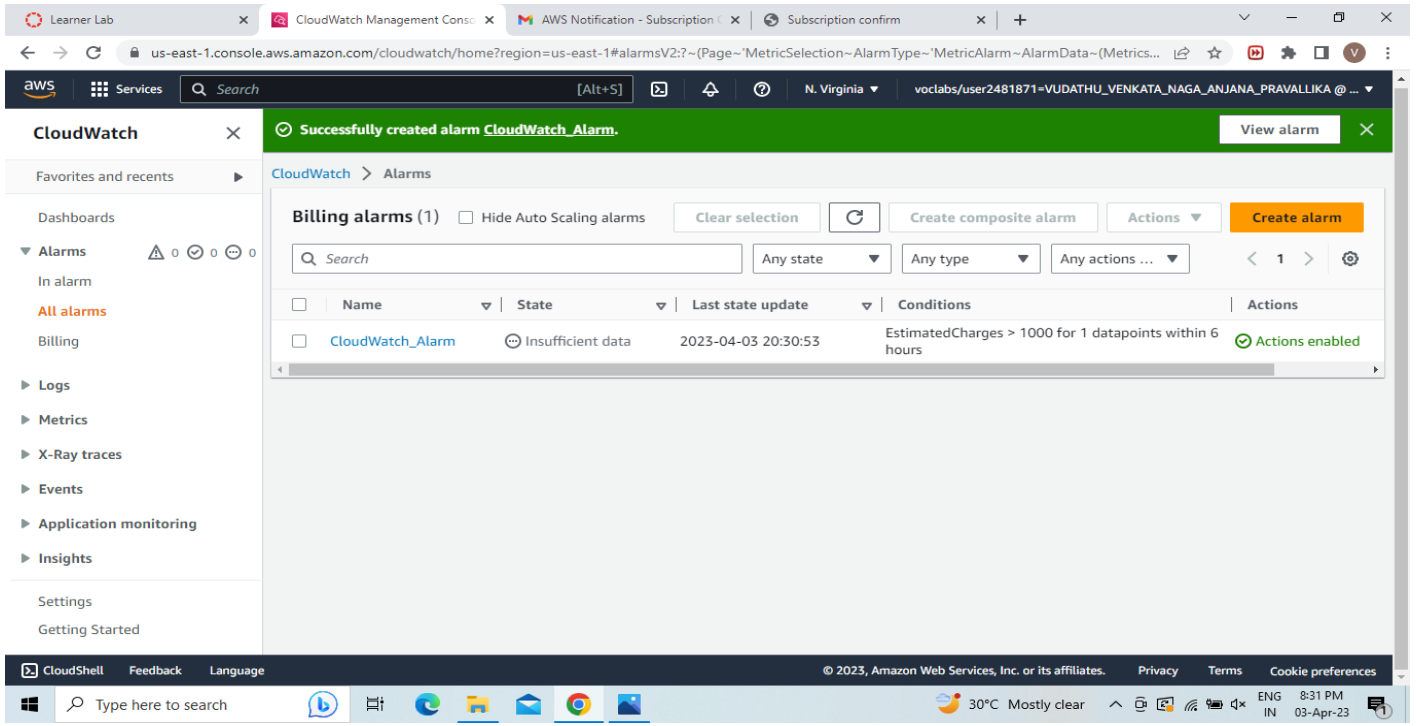


7.You will get a AWS Notification-Subscription Confirmation mail to the email which you have provided.Click on Confirm Subscription.Then it will open a window showing Subscription Confirmed.





8.Preview the details you have entered .
9.Click on Create alarm.This will Create your Alarm.



1)In the search box to the right of Services, search for and choose Lambda to open the AWS Lambda console.

2)Choose Create function.

3)In the Create function screen, configure these settings:

>Choose Author from scratch

>Function name: myStopinator

>Runtime: Python 3.8

>Choose Change default execution role

>Execution role: Use an existing role

>Existing role: From the dropdown list, choose myStopinatorRole

4)Choose Create function.

5) Choose Add trigger.

6)Choose the Select a trigger dropdown menu, and select EventBridge (CloudWatch Events).

7)For the rule, choose Create a new rule and configure the settings and click add.

Below the Function overview pane, choose Code, and then choose lambda_function.py to display and edit the Lambda function code.

In the Code source pane, delete the existing code. Copy the following code, and paste it in the box:

```
import boto3
region = '<REPLACE_WITH_REGION>'
instances = ['<REPLACE_WITH_INSTANCE_ID>']
ec2 = boto3.client('ec2', region_name=region)

def lambda_handler(event, context):
    ec2.stop_instances(InstanceIds=instances)
    print('stopped your instances: ' + str(instances))
```

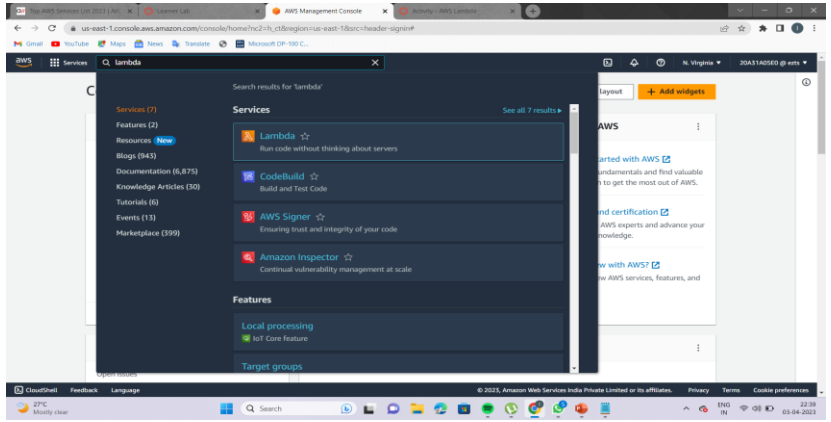
9)Replace the <REPLACE_WITH_REGION> placeholder with the actual Region that you are using. To do this:

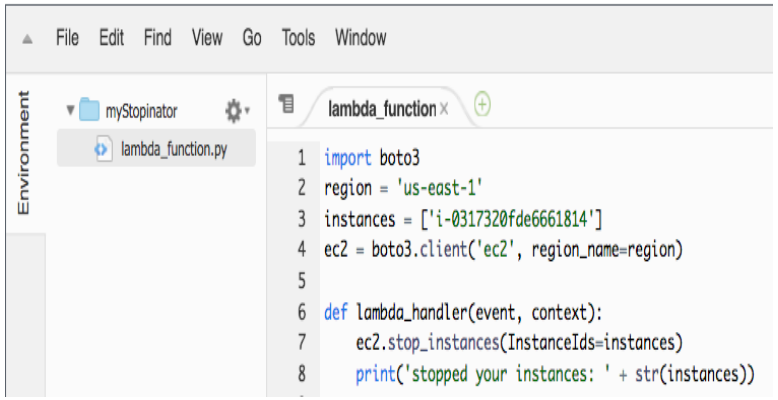
10)Choose on the region on the top right corner and use the region code. For example, the region code for US East (N. Virginia) is us-east-1.

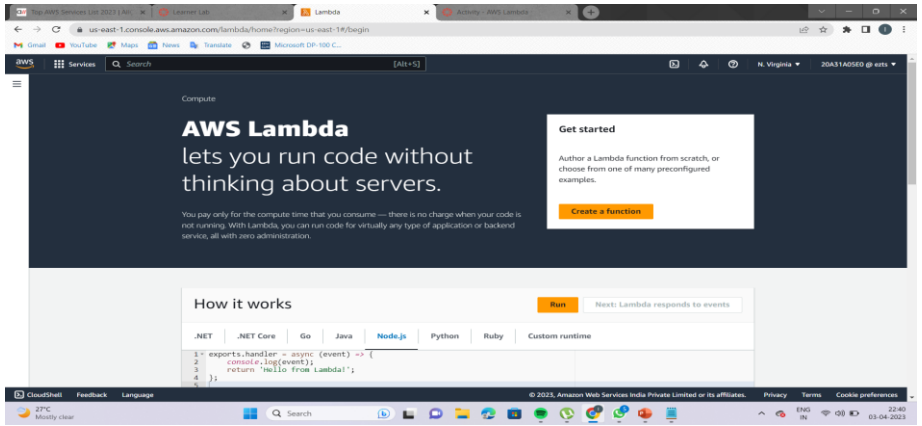
11)Verify that an EC2 instance named instance1 is running in your account, and copy the instance1 instance ID.

12)Return to the AWS Lambda console browser tab, and replace <REPLACE_WITH_INSTANCE_ID> with the actual instance ID that you just copied.

13)Choose the File menu and Save the changes. Then, in the top-right corner of the Code source box, choose Deploy.


lambda
Search results for 'lambda'
Services
Lambda
Run code without thinking about servers
CodeBuild
Build and Test Code
AWS Signer
Ensuring trust and integrity of your code
Amazon Inspector
Continual vulnerability management at scale
Features
Local processing
IoT Core feature
Target groups


File Edit Find View Go Tools Window
Environment
myStopinator
lambda_function.py
lambda_function
1 import boto3
2 region = 'us-east-1'
3 instances = ['i-0317320fde6661814']
4 ec2 = boto3.client('ec2', region_name=region)
5
6 def lambda_handler(event, context):
7 ec2.stop_instances(InstanceIds=instances)
8 print('stopped your instances: ' + str(instances))


Compute
AWS Lambda
lets you run code without thinking about servers.
You pay only for the compute time that you consume — there is no charge when your code is not running. With Lambda, you can run code for virtually any type of application or backend service, all with zero administration.
Get started
Author a Lambda function from scratch, or choose from one of many preconfigured examples.
Create a function
How it works
Run
Next: Lambda responds to events
.NET
.NET Core
Go
Java
Node.js
Python
Ruby
Custom runtime
exports.handler = async (event) => {
console.log(event);
return 'Hello from Lambda!';
};