AWS SERVICES

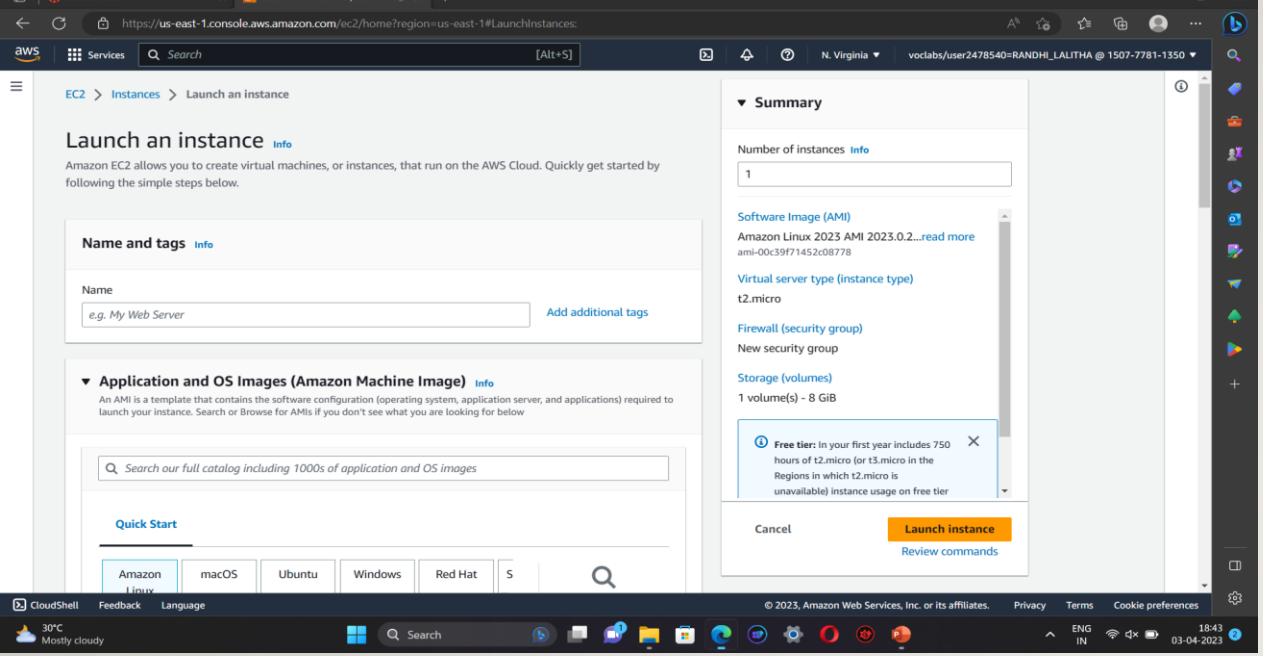
-R.Lalitha

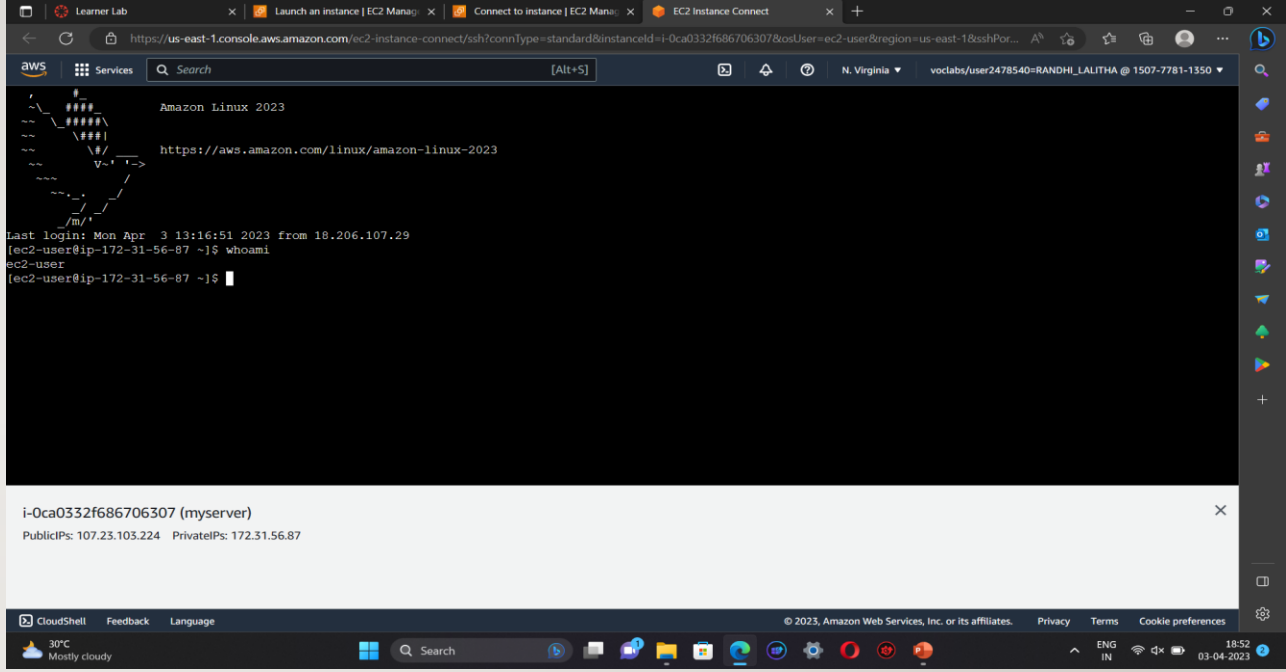
CREATING AN EC2 INSTANCE

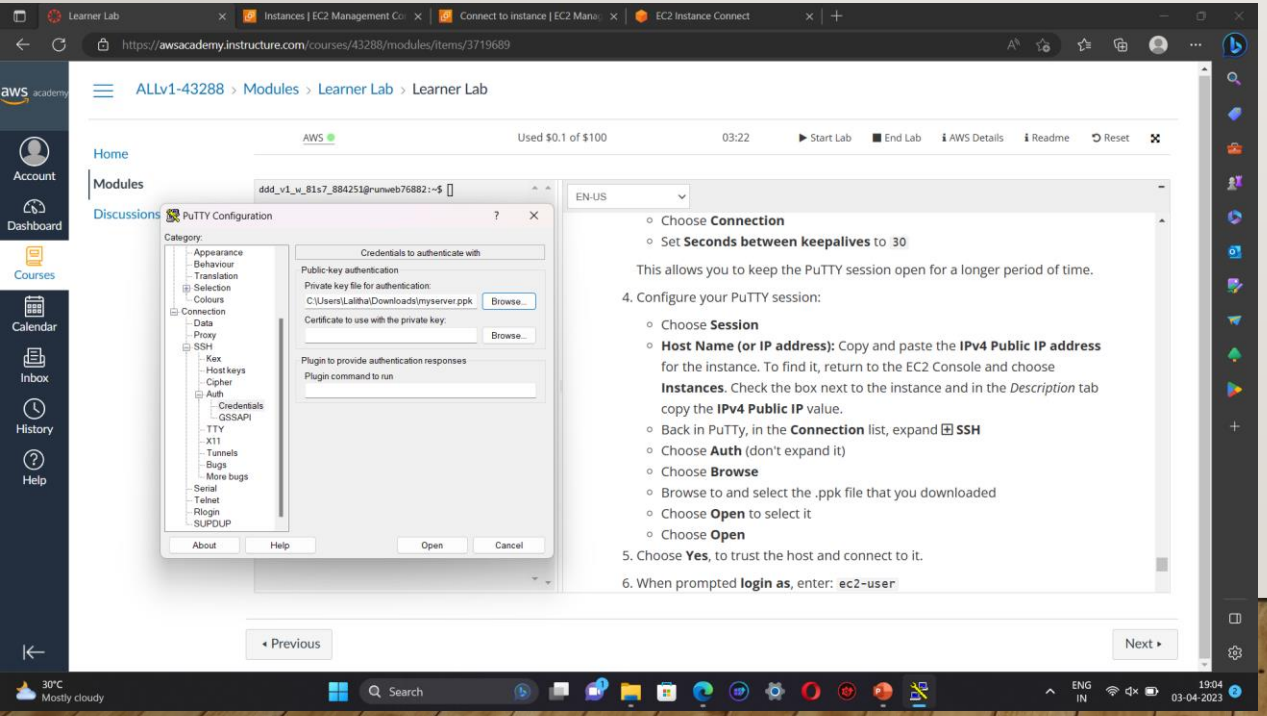
Step-1: Go to AWS services , click EC2 and then select 'launch instances'.
Step-2: Name the instance, select an AMI(LINUX, WINOWS server) , select a key pair and click launch instance.
Step-3: For linux-select ppk key and for windows server-select pem key.
Step-4: If a key pair is not available create 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 putty session. This will redirects you to the CLI.

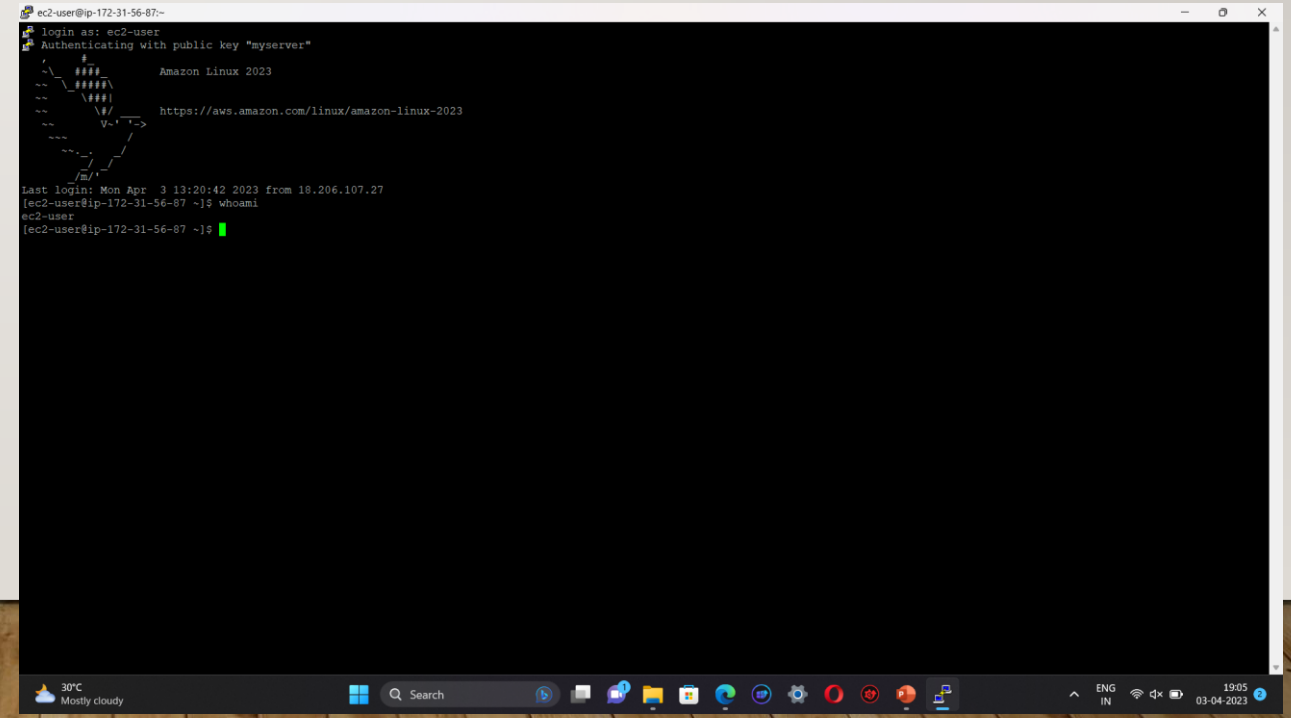
For windows server-click connect→ RDP client→download rdp file→ get password→ upload private key→ decrypt password. Open rdp file and enter the password. This will redirects you to the windows server.

Step-6: Terminate the 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

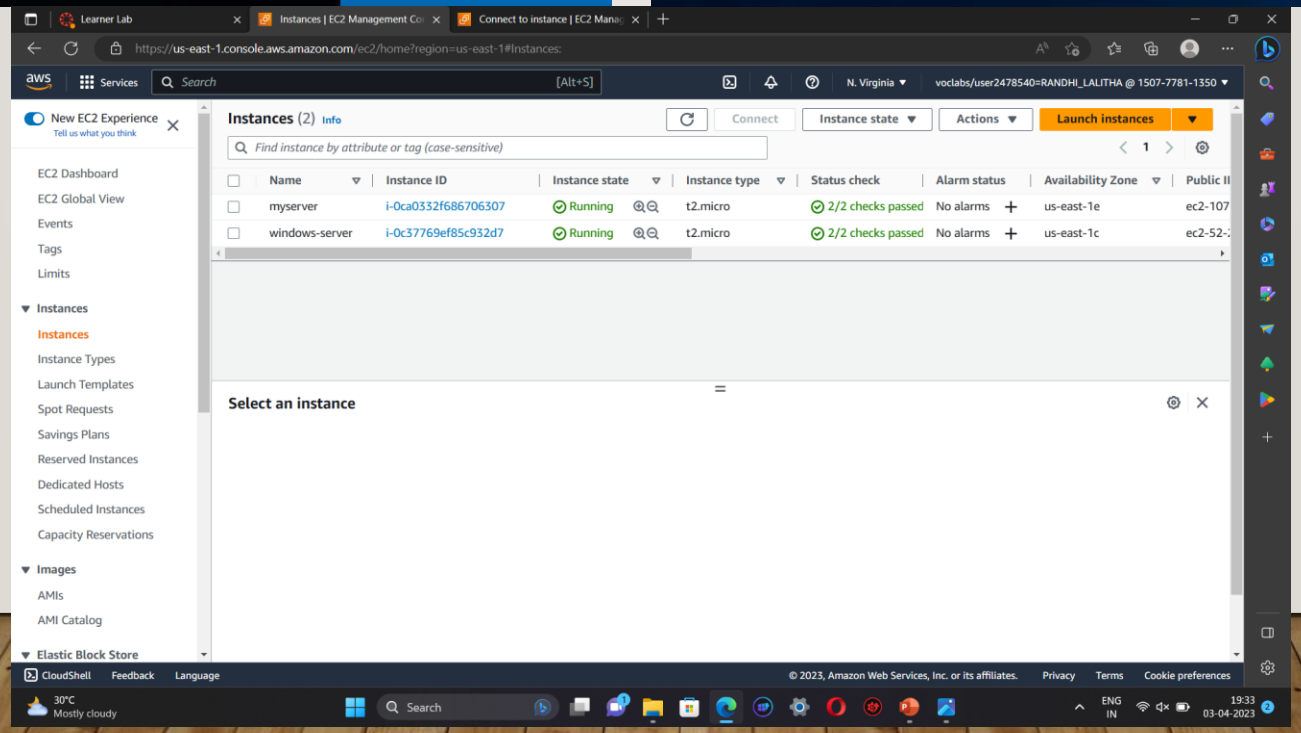

Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023
Last login: Mon Apr 3 13:16:51 2023 from 18.206.107.29
[ec2-user@ip-172-31-56-87 ~]$ whoami
ec2-user
[ec2-user@ip-172-31-56-87 ~]$
i-0ca0332f686706307 (myserver)
PublicIPs: 107.23.103.224 PrivateIPs: 172.31.56.87


ALv1-43288 > Modules > Learner Lab > Learner Lab
Used $0.1 of $100
03:22
Start Lab
End Lab
AWS Details
Readme
Reset
Home
Modules
Discussions
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

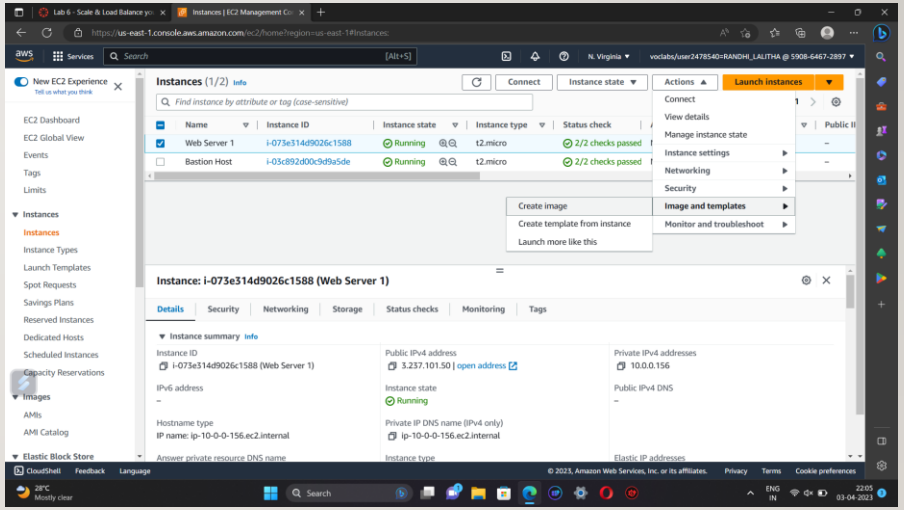

Hostname : EC2AMAZ-RNIQ1T8
Instance ID : i-0c31bceeb5a820086
Private IP Address : 172.31.90.111
Public IP Address : 52.91.186.118
Instance Size : t2.micro
Availability Zone : us-east-1c
Architecture : AMD64
Total Memory : 1024
Network : Low to Moderate


Instances (2)
Connect
Instance state
Actions
Launch instances
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

ELASTIC LOAD BALANCER

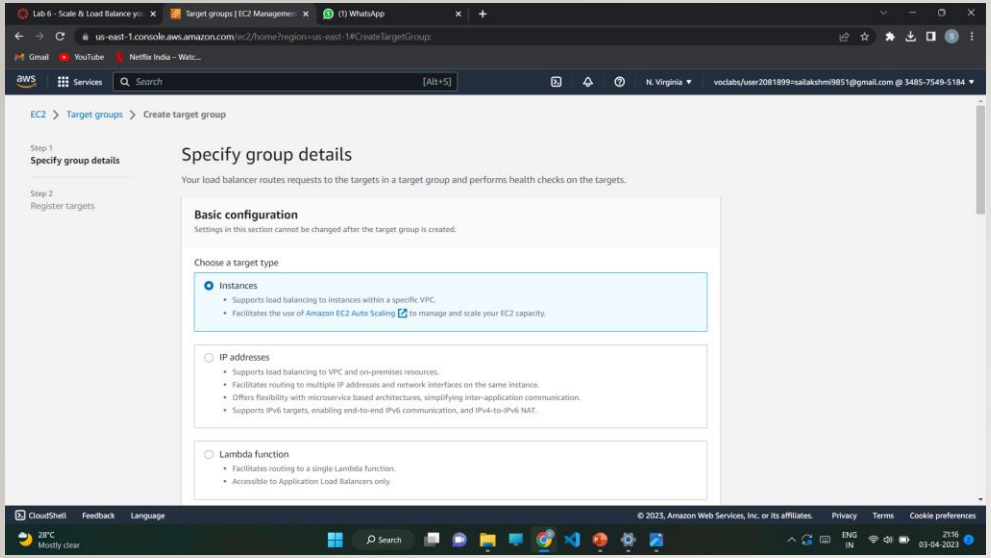
Step-1: Creating an AMI for Auto Scaling

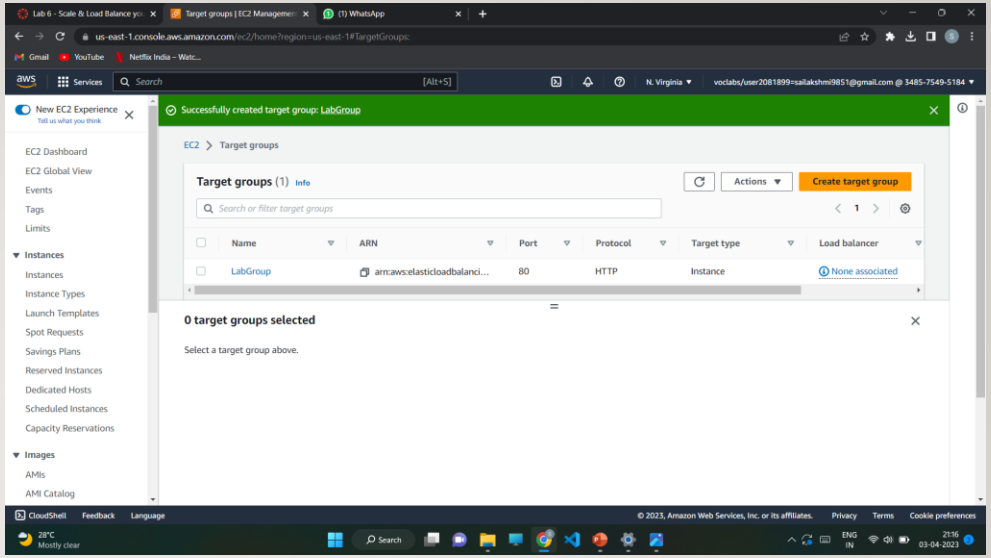
- Click start lab then click on AWS.
- You will navigate to AWS management console. Click on services and select EC2.
- Click instances. Make sure that **Status Checks** for **Web Server 1** displays 2/2 checks.
- Select Web Server 1 and in actions click images and templates > create image. Name the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



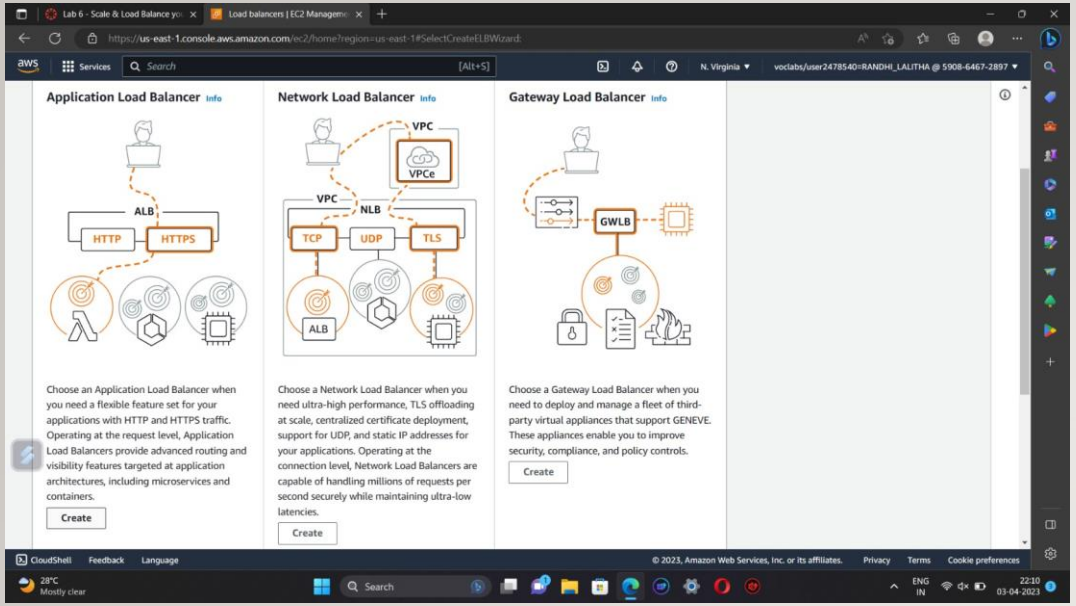
Step-2: Creating a load balancer

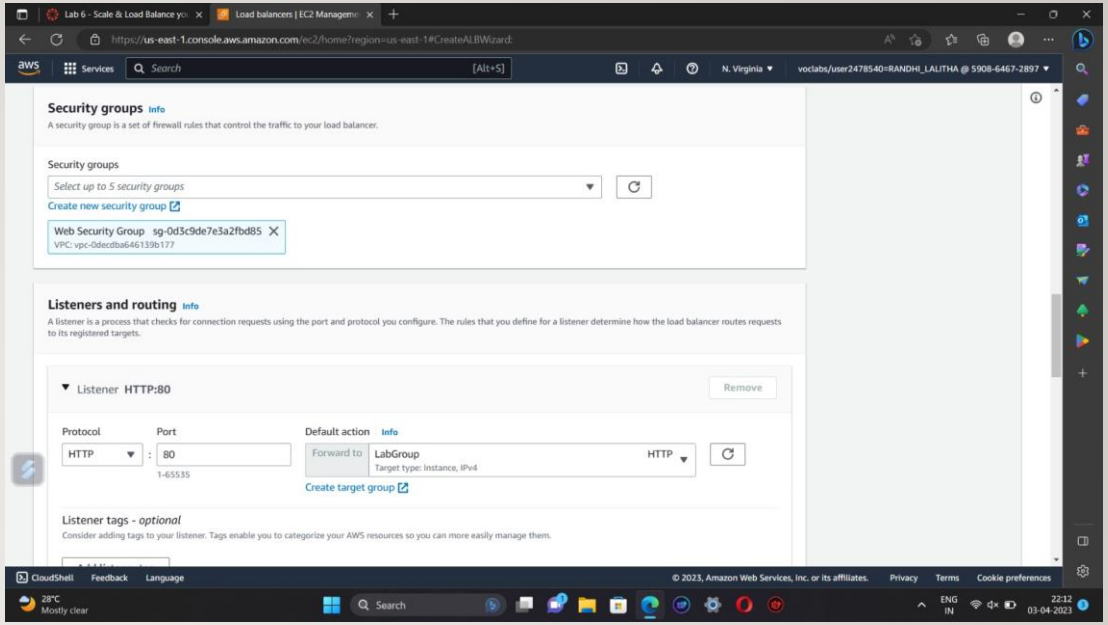
- Choose Target Groups and then click on create target group.
- Select target type as instances. Name the target group. Select Lab VPC under VPC that is we are creating load balancer in Lab VPC .
- Choose next and then click on create target group.

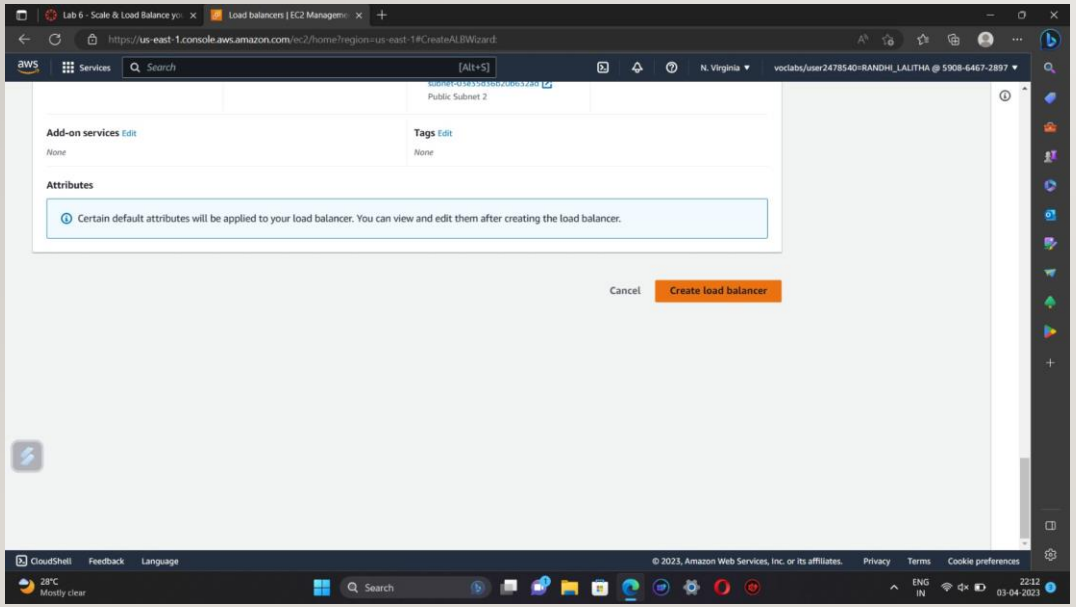


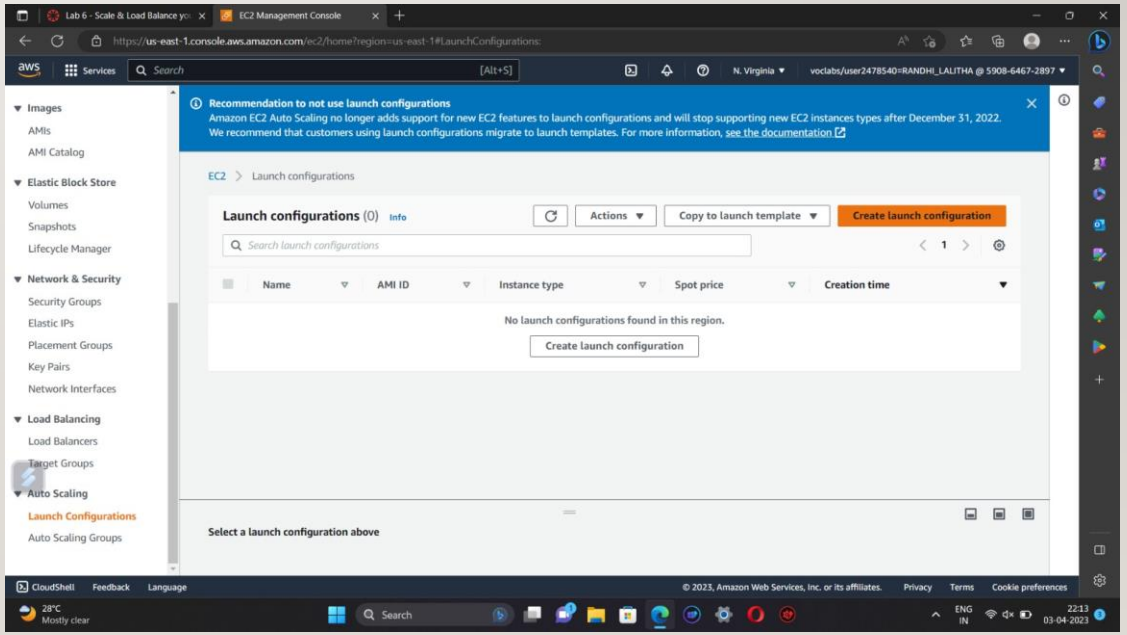


- From the left navigation pane , select Load Balancers. Click create load balancer.
- To create a application balancer, click create under Application Load Balancer and Name it.
- In Networking mapping, select Lab VPC and specify the subnets that the load balancer should use.
- In security groups, select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


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.

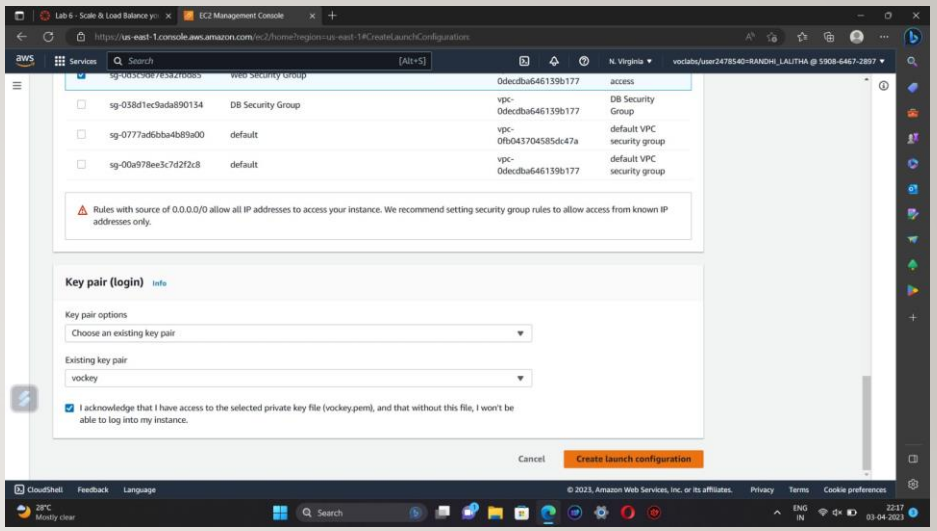

Add-on services Edit
None
Tags Edit
None
Attributes
Certain default attributes will be applied to your load balancer. You can view and edit them after creating the load balancer.
Cancel
Create load balancer

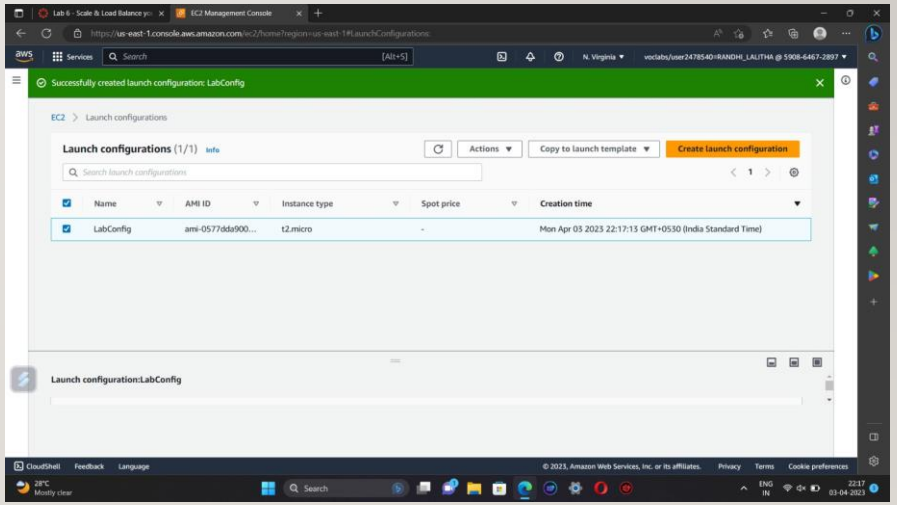

Recommendation to not use launch configurations
Amazon EC2 Auto Scaling no longer adds support for new EC2 features to launch configurations and will stop supporting new EC2 instances types after December 31, 2022.
We recommend that customers using launch configurations migrate to launch templates. For 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.

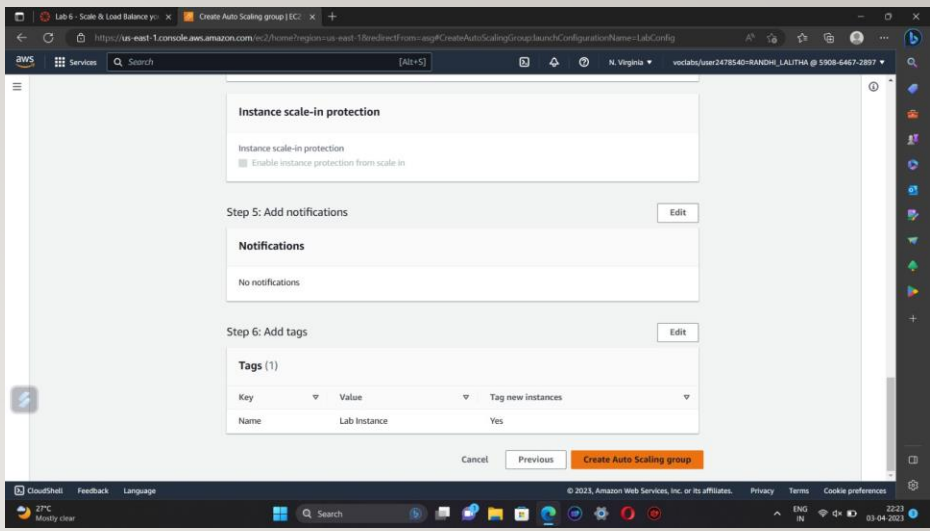
Step-3: Create a Launch Configuration and an Auto Scaling Group

- 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**



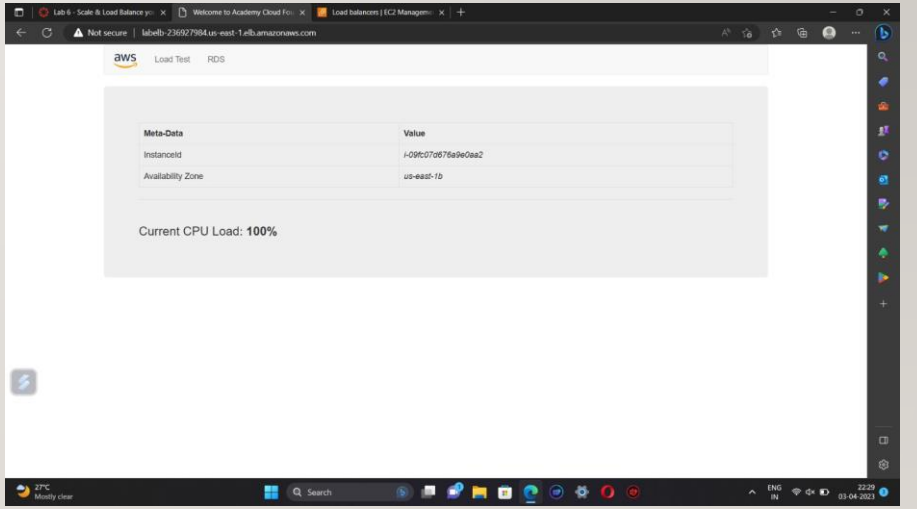


- Specify the values under Group size.
- Under **Scaling policies**, choose *Target tracking scaling policy* and name the policy. Specify metric type and target value. Then add a tag and click create auto scaling group.



Step-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.

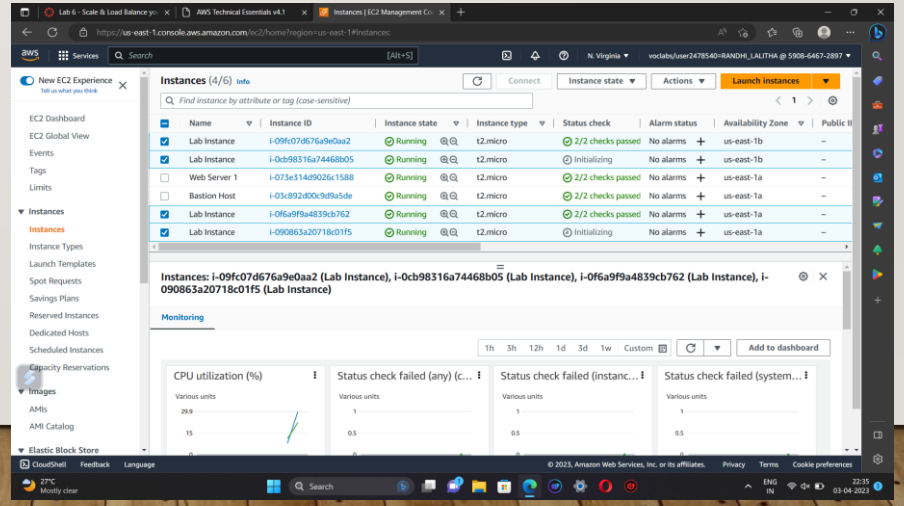


Step-5: Test Auto Scaling

- 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.

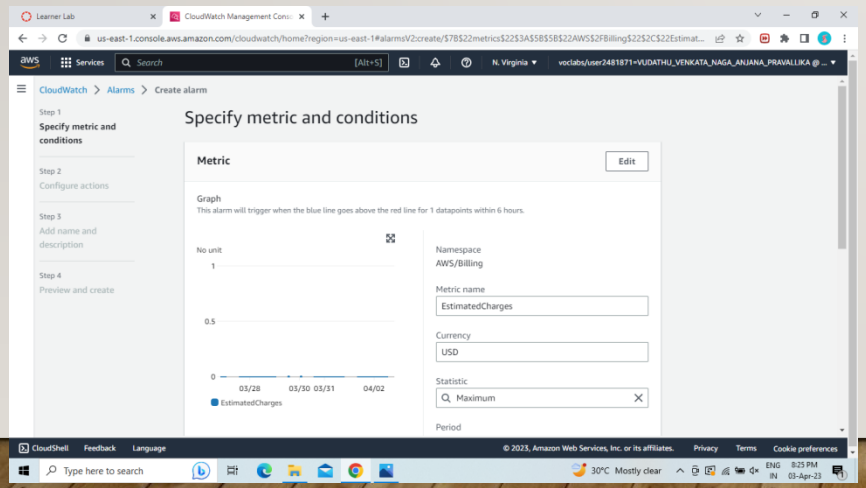
- You should see the **AlarmHigh** chart indicating an increasing CPU percentage. Once it crosses the 60% line for more than 3 minutes, it will trigger Auto Scaling to add additional instances.
- In EC2 instances , you notice that more than two instances labeled **Lab Instance** should now be running.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

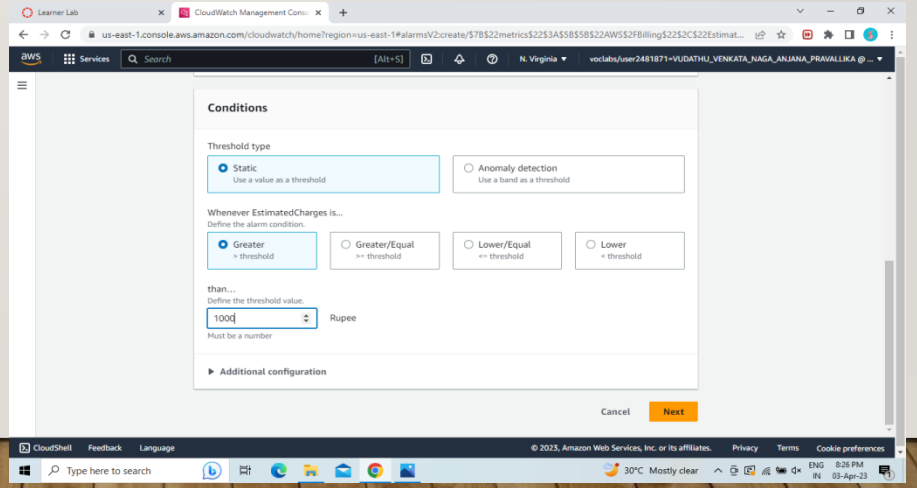


AWS CLOUDWATCH

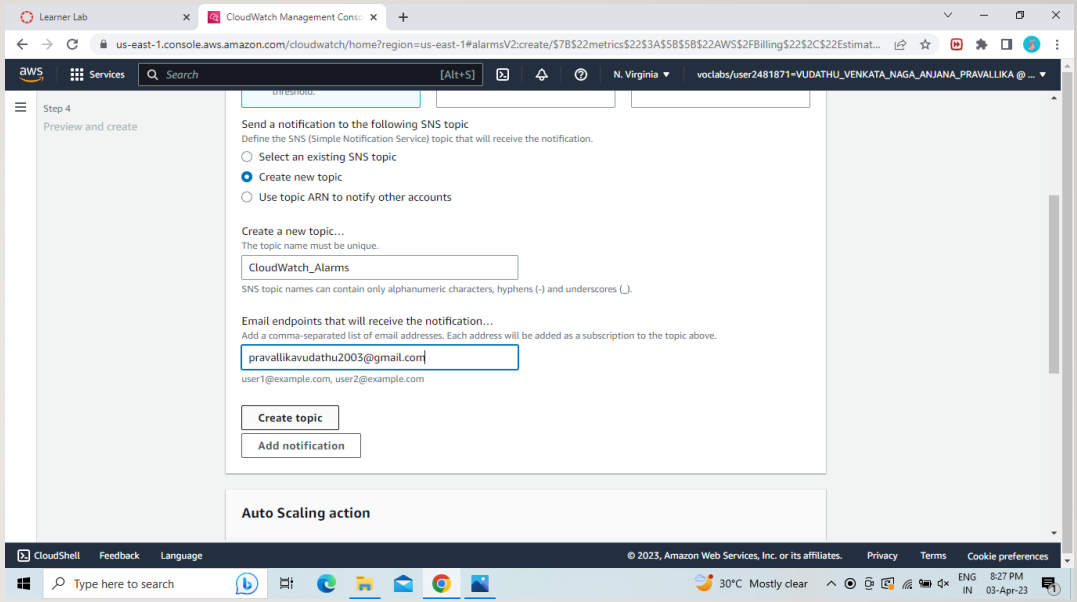
PROCEDURE

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

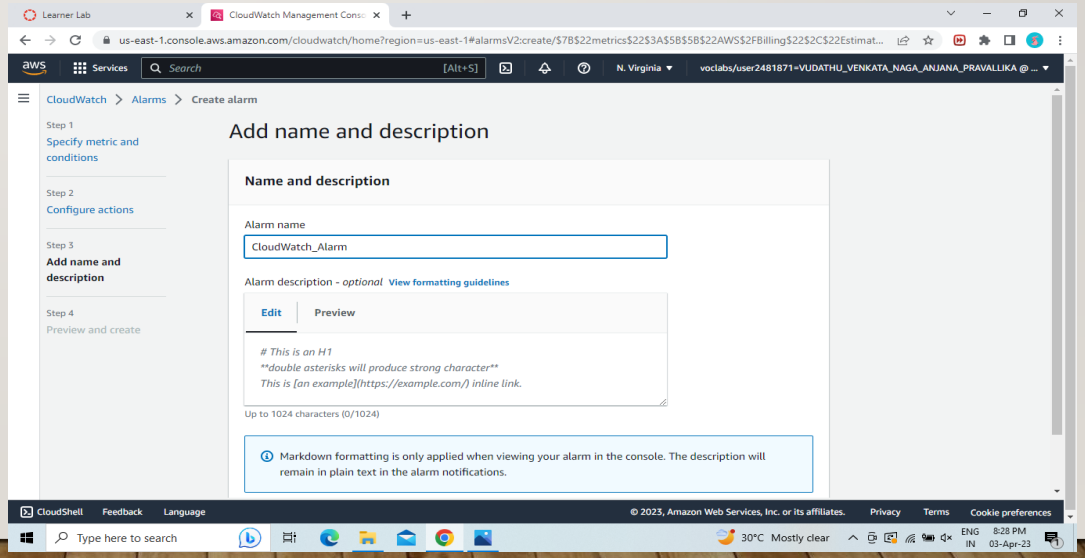




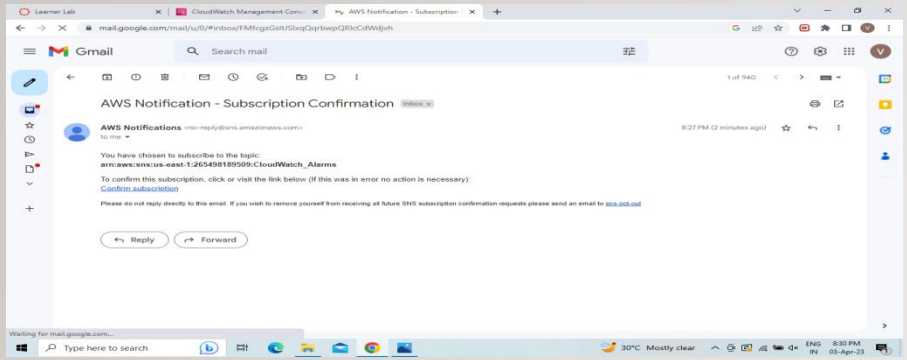
5.Now for Cofigure Actions choose Create new topic.Give a name to the topic and enter your email to receive a notification.Click on Create Topic,then Next.

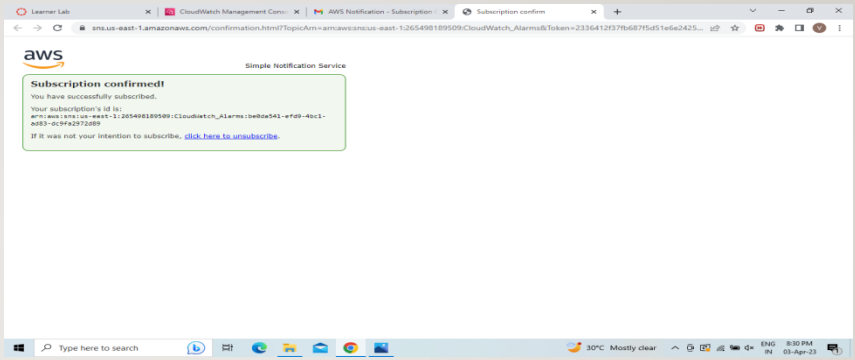


6.Give a name to your Alarm and Click on next.



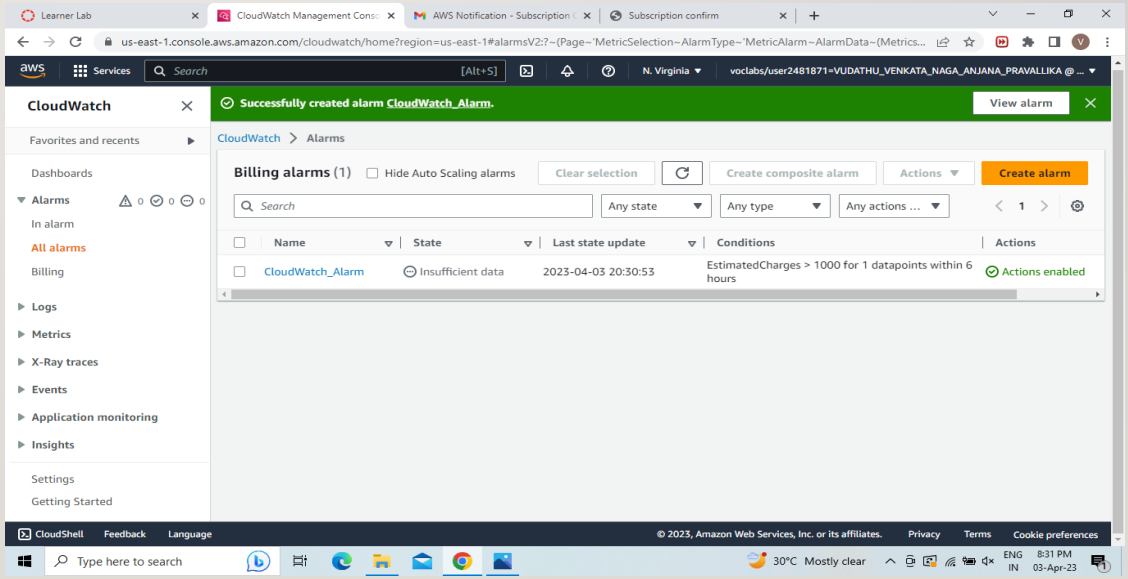
7.You will get a AWS Notification-Subscription Confirmation mail to the email which you have provided.Click on Confirm Subscription.Then it will open a window showing Subscription Confirmed.





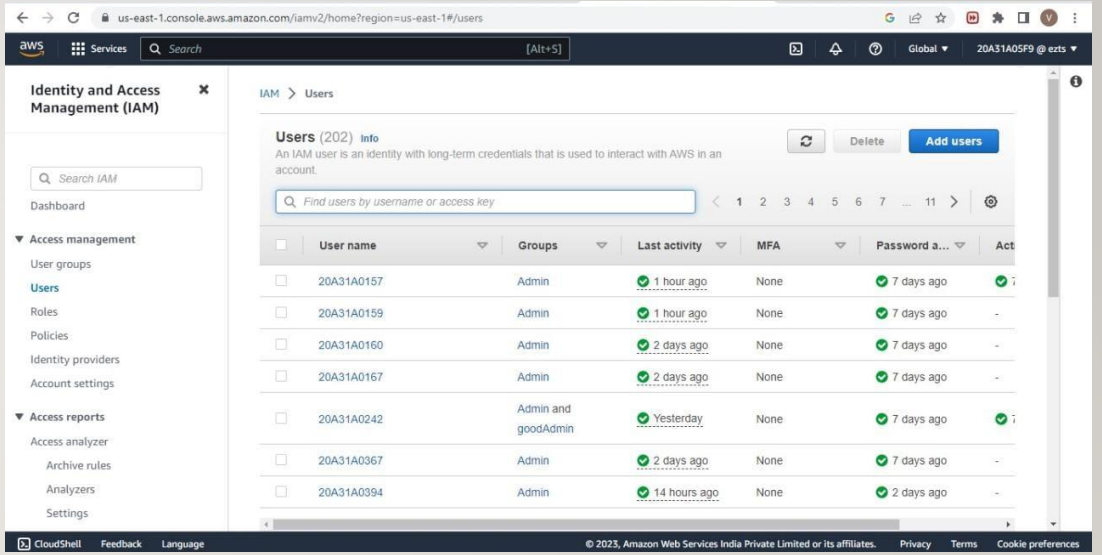
8.Preview the details you have entered .

9.Click on Create alarm.This will Create your Alarm.

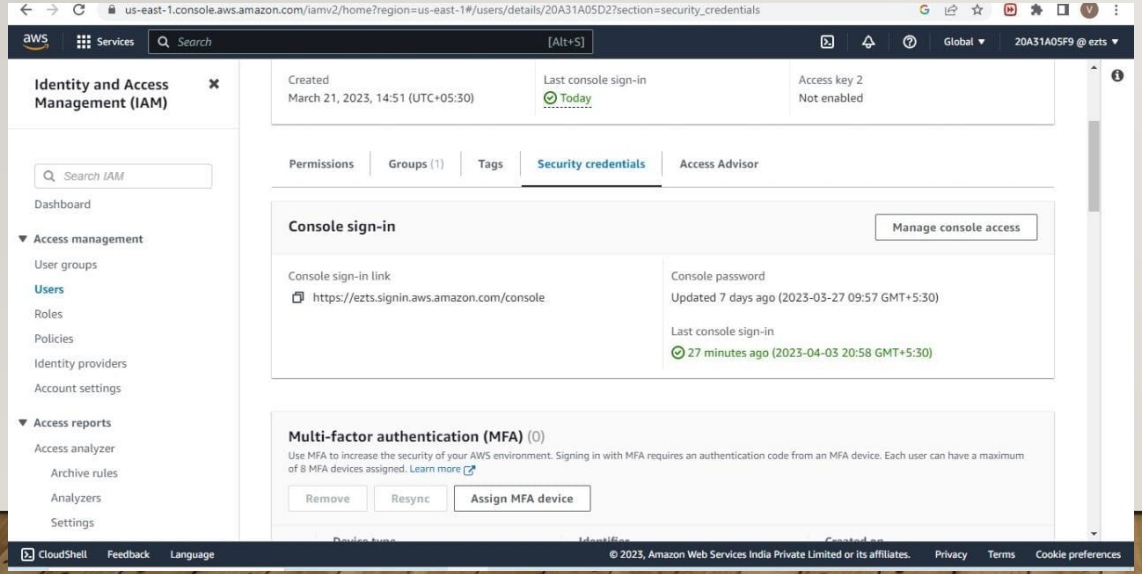


AWS COMMAND LINE INTERFACE

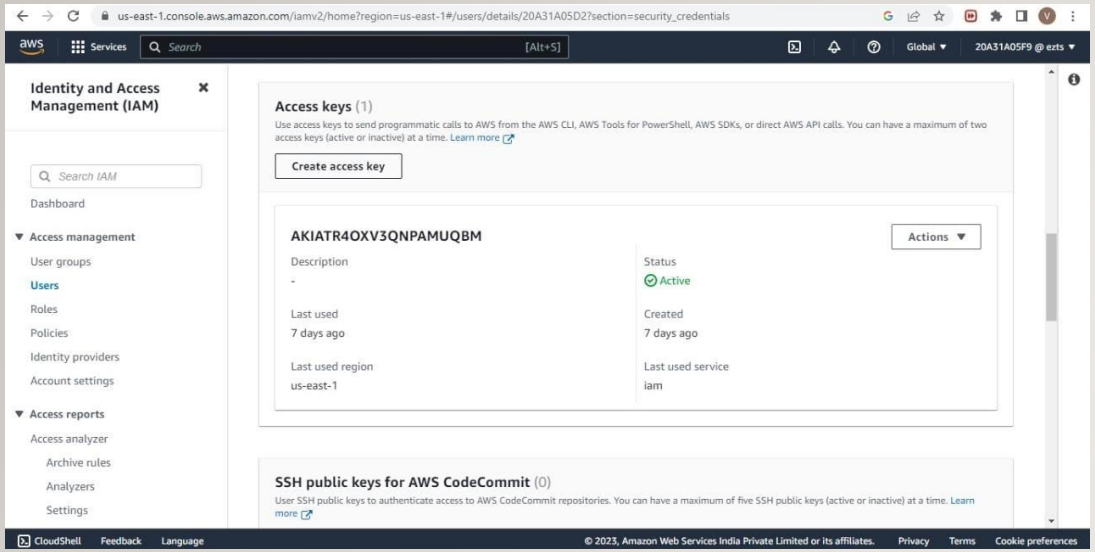
STEP 1 - Download and install AWS CLI and complete the installation steps.
STEP 2 - Login to AWS Management Console and search for IAM.
STEP 3 - In the navigation pane ,select Users

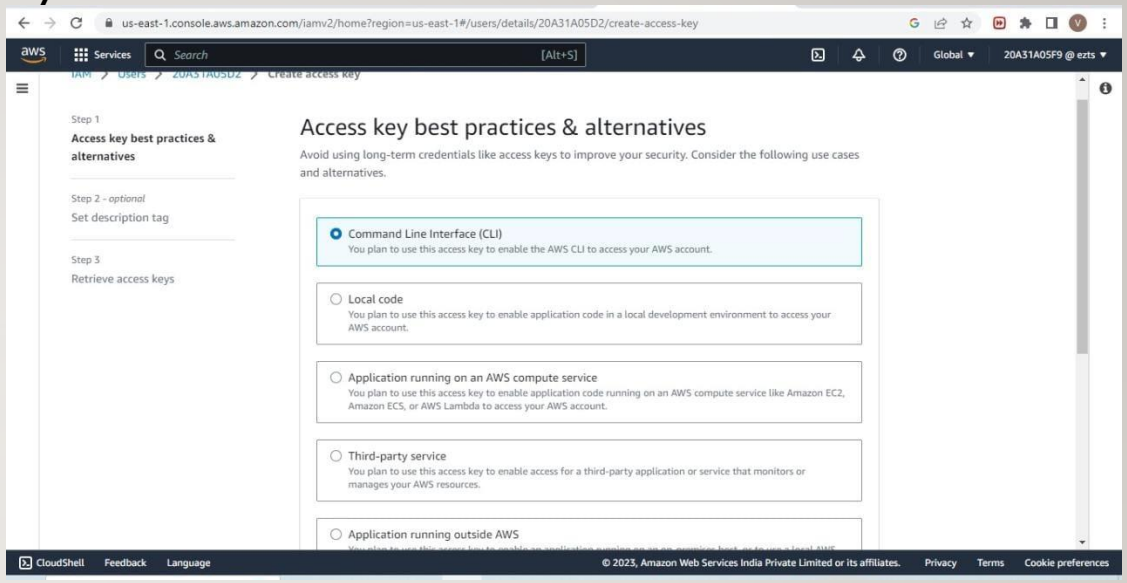


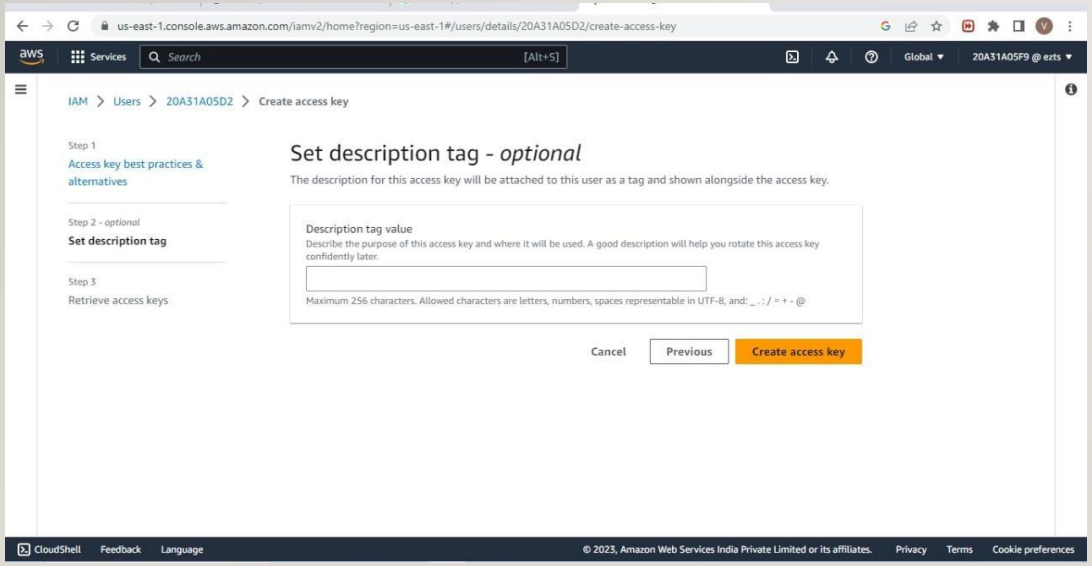
STEP 4 - In the users select the name of the user whose access keys you want to create.
STEP 5 - Click on Security Credentials tab.

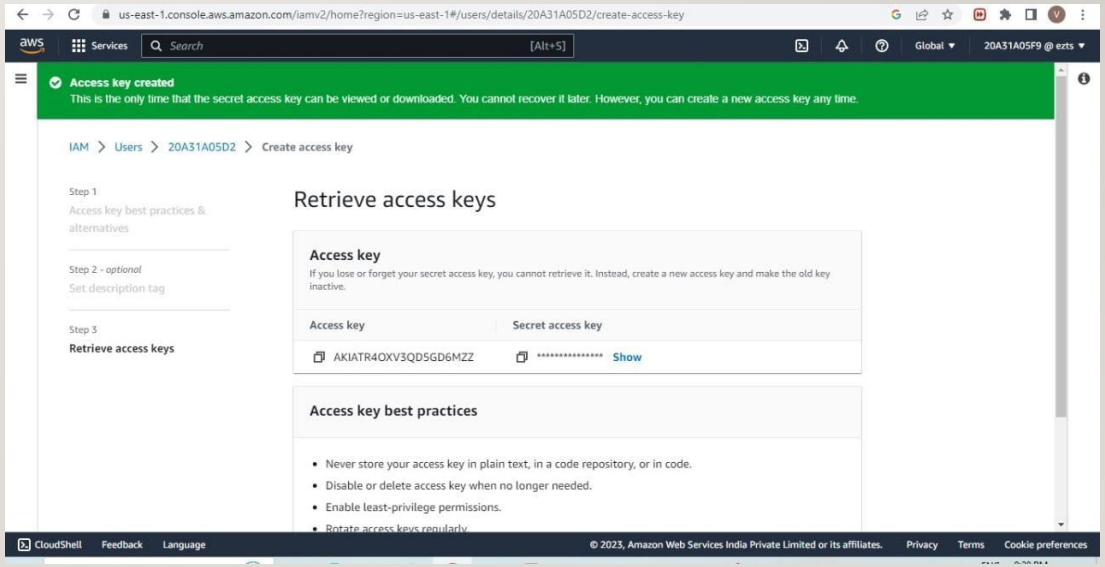


STEP 6 - In the access Keys section , choose Create access key.









STEP 6 – Now you can use this access key to configure CLI

STEP 7 - 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.

AWS LIGHT SAIL

PROCEDURE:

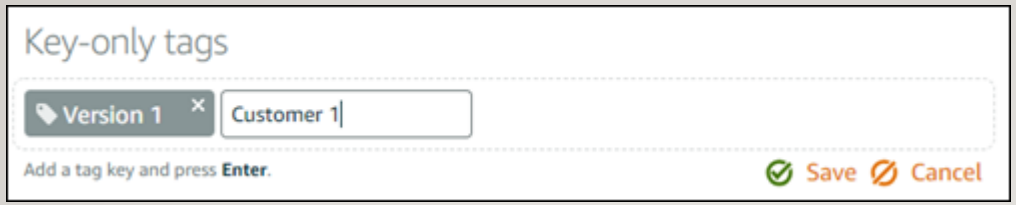
1.On the home page, choose Create instance.
2.Select a location for your instance (an AWS Region and Availability Zone).Choose Change Region and zone to create your instance in another location.
3.Optionally, you can change the Availability Zone.Choose an Availability Zone from the dropdown list.
4.Pick an application (Apps + OS) or an operating system (OS Only).
5.Choose your instance plan.
6.Enter a name for your instance.

Resource names:

1. Must be unique within each AWS Region in your Lightsail account.
2. Must contain 2 to 255 characters.
3. Must start and end with an alphanumeric character or number.
4. Can include alphanumeric characters, numbers, periods, dashes, and underscores.

7.Choose one of the following options to add tags to your instance:

- Add key-only tags or Edit key-only tags (if tags have already been added). Enter your new tag into the tag key text box, and press Enter. Choose Save when you’re done entering your tags to add them, or choose Cancel to not add them.



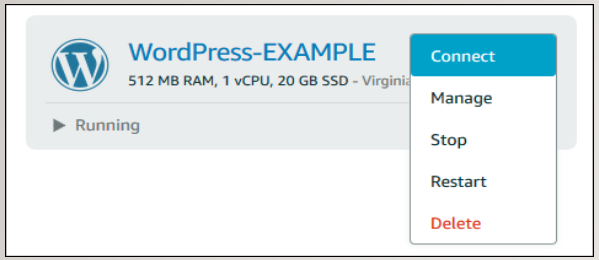
- Create a key-value tag, then enter a key into the Key text box, and a value into the Value text box. Choose Save when you're done entering your tags, or choose Cancel to not add them.Key-value tags can only be added one at a time before saving. To add more than one key-value tag, repeat the previous steps.

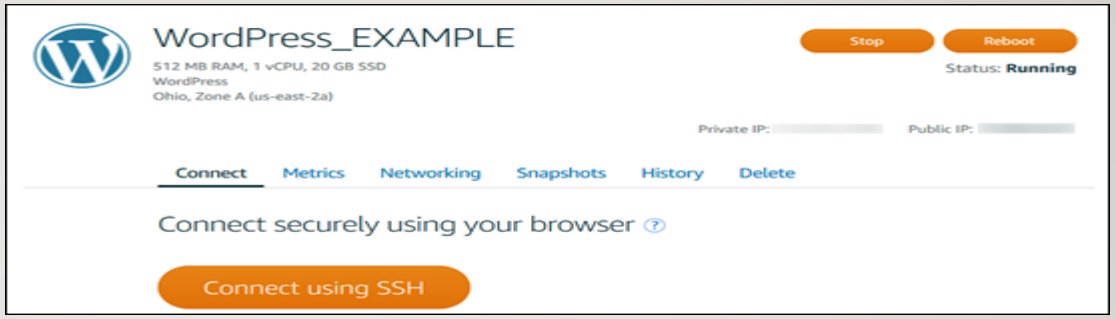
8.Choose Create instance.

Within minutes, your Lightsail instance is ready and you can connect to it via SSH, without leaving Lightsail!

How to connect to your instance

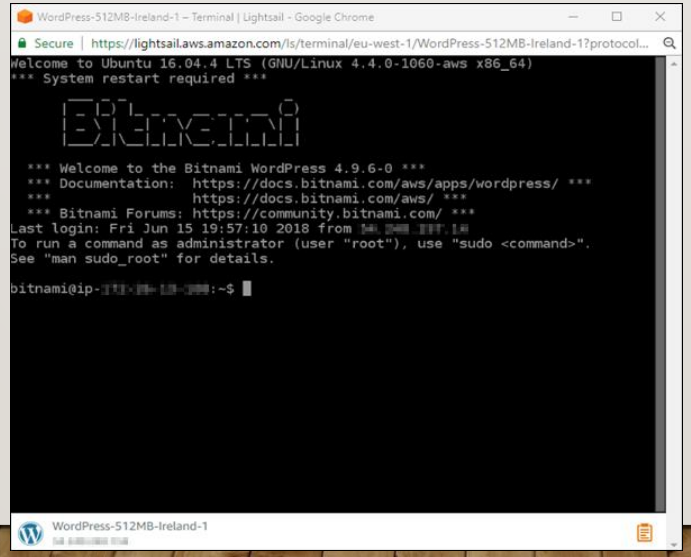
1.From the Lightsail home page, choose the menu on the right of your instance's name, and then choose connect.





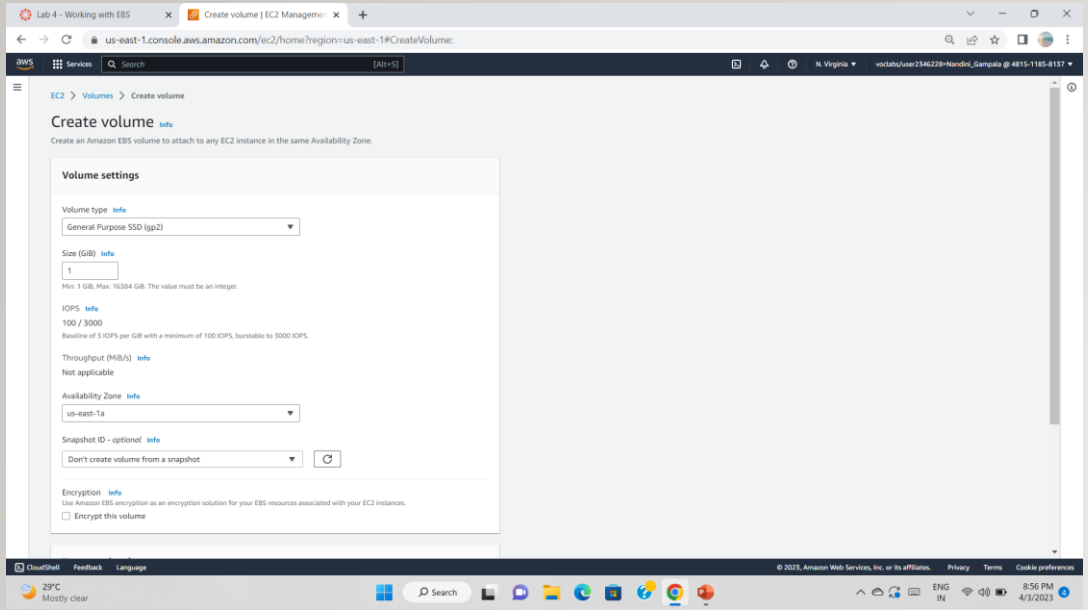
Alternately, you can open your instance management page and choose the Connect tab.

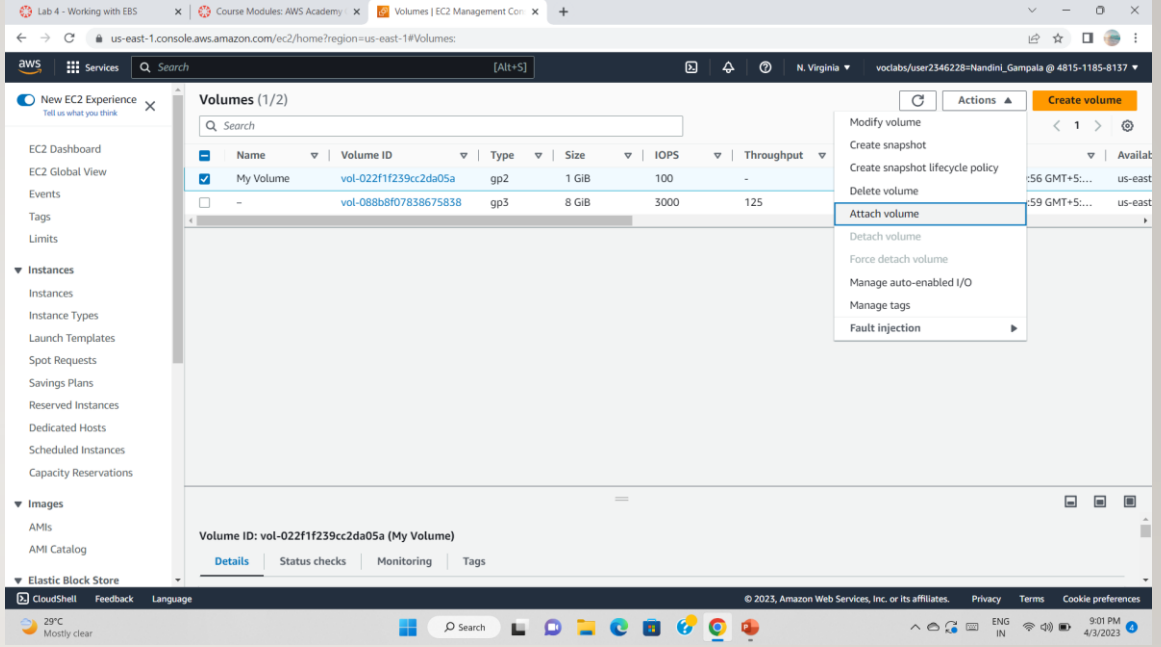
2.You can now type commands into the terminal and manage your Lightsail instance without setting up an SSH client.

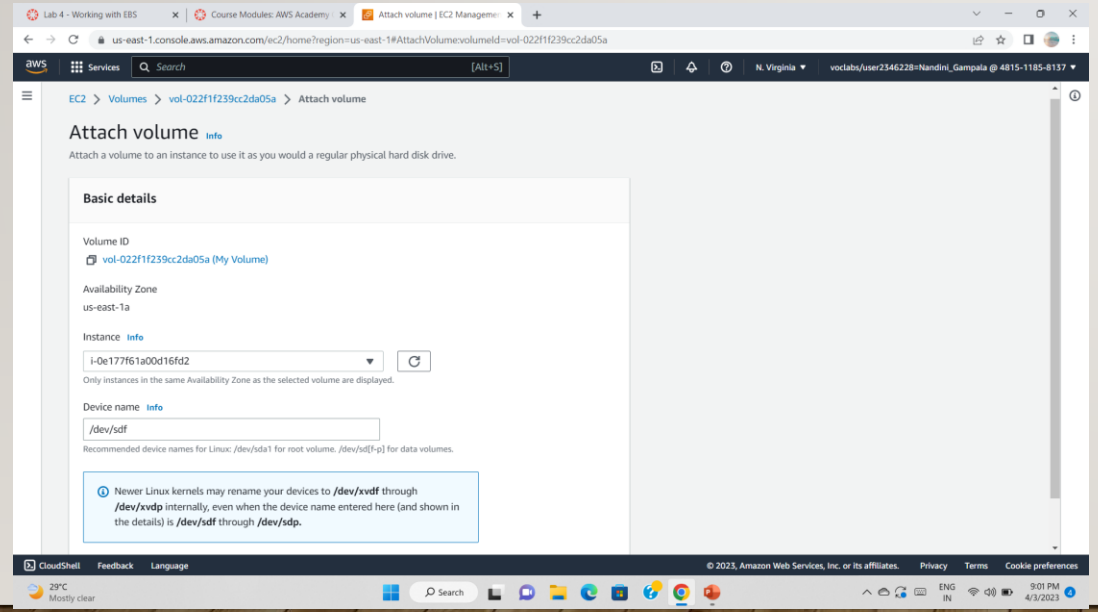


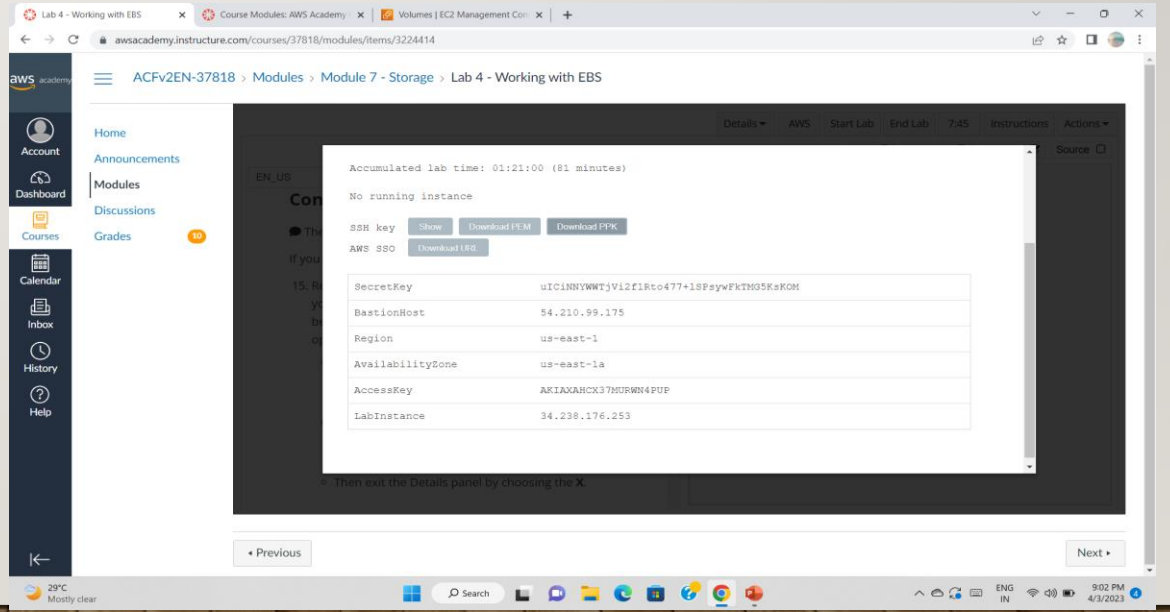
CREATING A EBS VOLUME

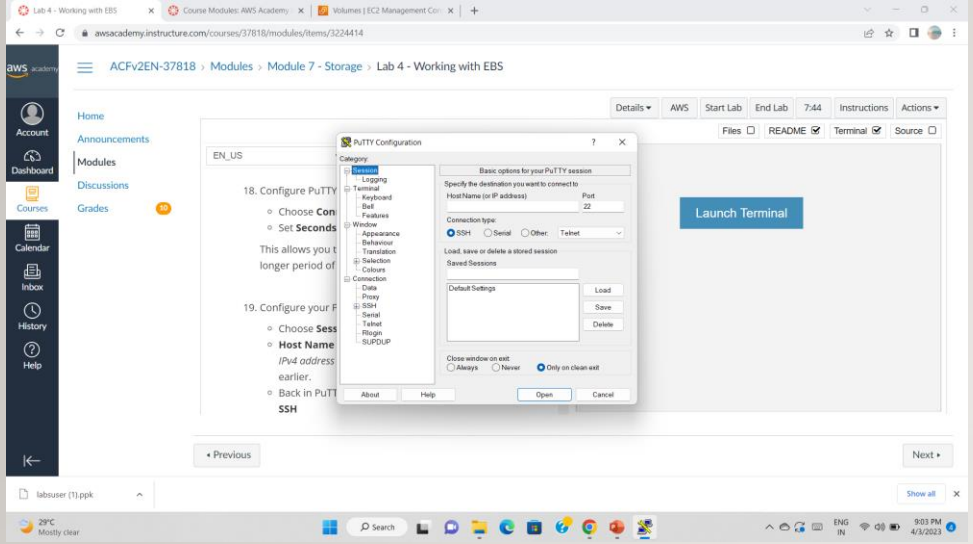
1. Open Management Console, on the services menu open Ec2
2. In the left navigation pane choose instances and create a instance with a name
3. Next, In the left navigation pane choose Volumes
4. Click on Create Volume
5. Select volume type, size(Gib),Availability Zone and in Add tag section add key and 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

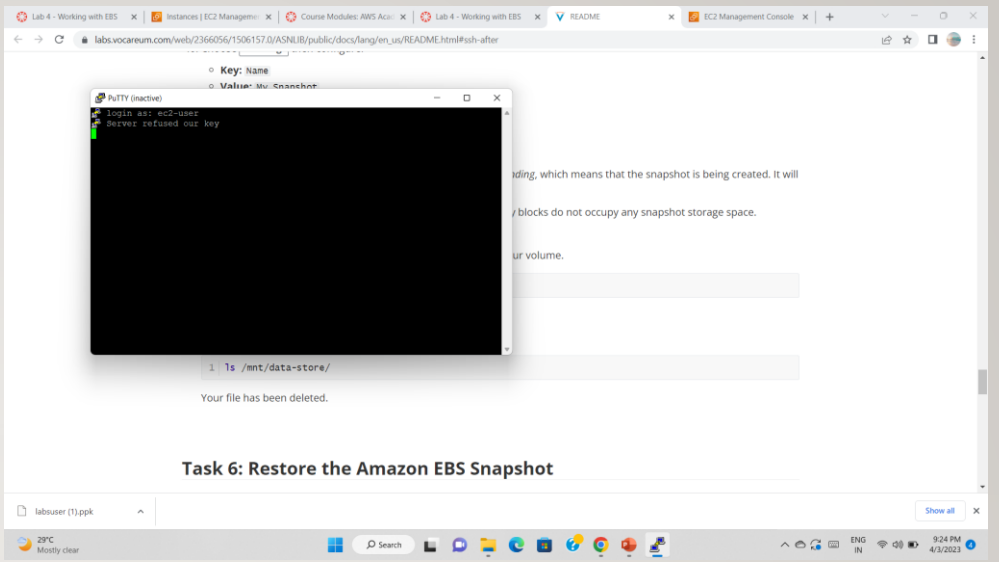

Create volume | EC2 Management
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateVolume:
EC2 > Volumes > Create volume
Create volume Info
Create an Amazon EBS volume to attach to any EC2 instance in the same Availability Zone.
Volume settings
Volume type Info
General Purpose SSD (gp2)
Size (GiB) Info
1
Min: 1 GiB, Max: 16384 GiB. The value must be an integer.
IOPS Info
100 / 3000
Baseline of 3 IOPS per GiB with a minimum of 100 IOPS, burstable to 3000 IOPS.
Throughput (MiB/s) Info
Not applicable
Availability Zone Info
us-east-1a
Snapshot ID - optional Info
Don't create volume from a snapshot
Encryption Info
Use Amazon EBS encryption as an encryption solution for your EBS resources associated with your EC2 instances.
Encrypt this volume
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

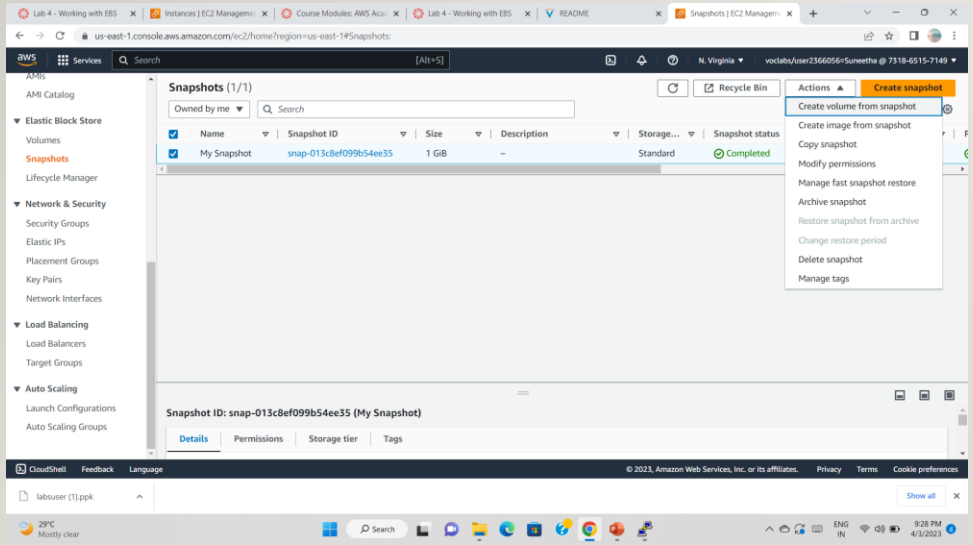

Volumes | EC2 Management
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Volumes:
New EC2 Experience
EC2 Dashboard
EC2 Global View
Events
Tags
Limits
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Scheduled Instances
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes (1/2)
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
Name
Volume ID
Type
Size
IOPS
Throughput
My Volume
vol-022f1f239cc2da05a
gp2
1 GiB
100
-
vol-088b8f07838675838
gp3
8 GiB
3000
125
Volume ID: vol-022f1f239cc2da05a (My Volume)
Details
Status checks
Monitoring
Tags
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

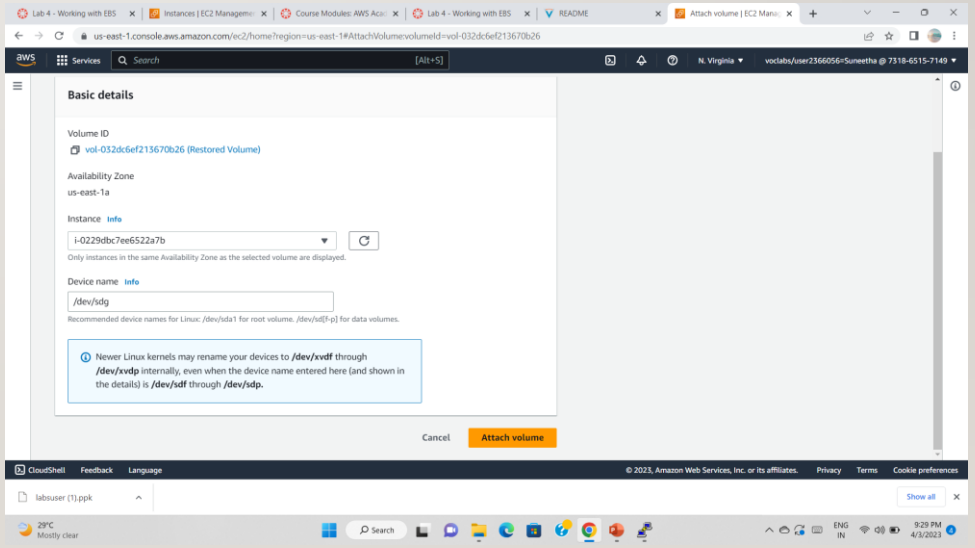

Attach volume | EC2 Management
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#AttachVolume:volumeId=vol-022f1f239cc2da05a
EC2 > Volumes > vol-022f1f239cc2da05a > Attach volume
Attach volume Info
Attach a volume to an instance to use it as you would a regular physical hard disk drive.
Basic details
Volume ID
vol-022f1f239cc2da05a (My Volume)
Availability Zone
us-east-1a
Instance Info
i-0e177f61a00d16fd2
Only instances in the same Availability Zone as the selected volume are displayed.
Device name Info
/dev/sdf
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.
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


awsacademy.instructure.com/courses/37818/modules/items/3224414
ACFv2EN-37818 > Modules > Module 7 - Storage > Lab 4 - Working with EBS
Account
Dashboard
Courses
Calendar
Inbox
History
Help
Home
Announcements
Modules
Discussions
Grades
Accumulated lab time: 01:21:00 (81 minutes)
No running instance
SSH key
Show
Download PEM
Download PPK
AWS SSO
Download URL
SecretKey
uICiNNYWWTjVi2flRto477+lSPsywFkTMG5KsKOM
BastionHost
54.210.99.175
Region
us-east-1
AvailabilityZone
us-east-1a
AccessKey
AKIAXAHCX37MURWN4PUP
LabInstance
34.238.176.253
Previous
Next


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

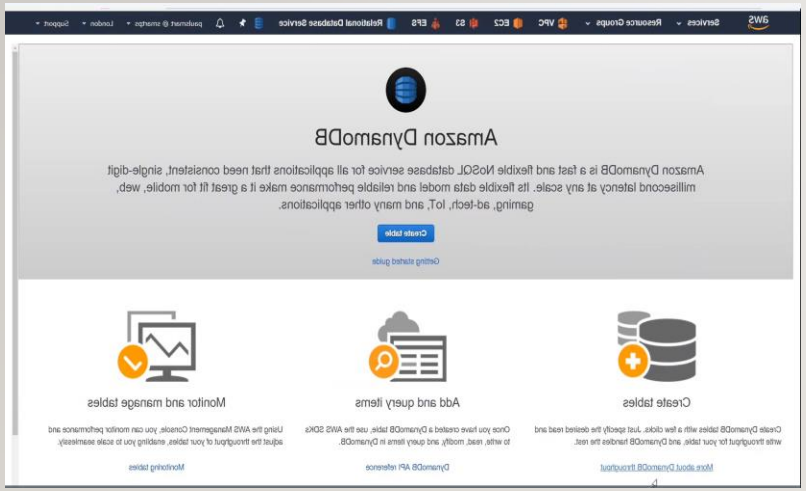

Task 6: Restore the Amazon EBS Snapshot

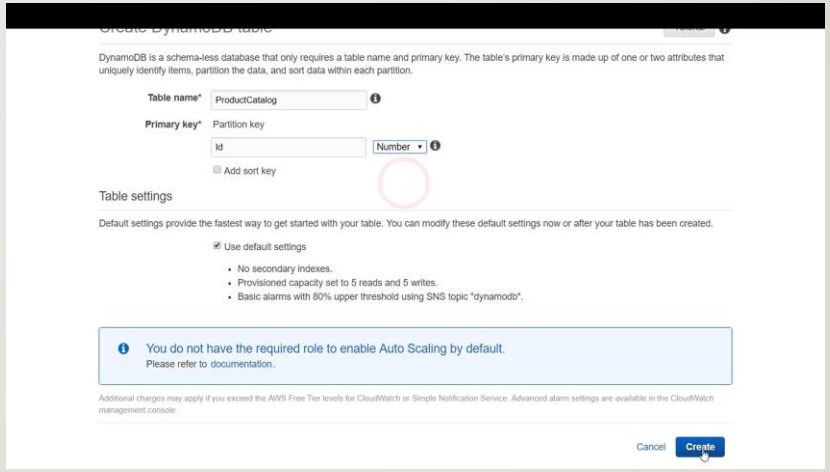

Snapshots (1/1)
Create volume from snapshot
Create image from snapshot
Copy snapshot
Modify permissions
Manage fast snapshot restore
Archive snapshot
Delete snapshot
Manage tags
Snapshot ID: snap-013c8ef099b54ee35 (My Snapshot)


Basic details
Volume ID
vol-032dc6ef213670b26 (Restored Volume)
Availability Zone
us-east-1a
Instance
i-0229dbc7ee6522a7b
Device name
/dev/sdg
Cancel
Attach volume

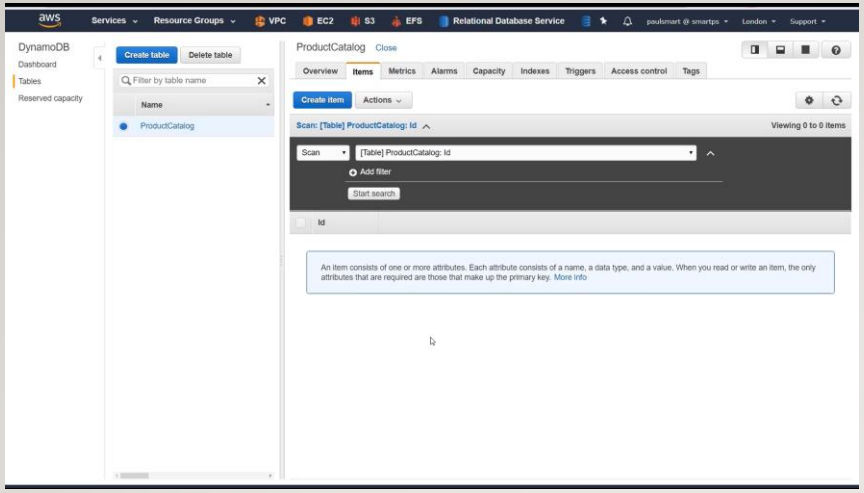
DYNAMODB

→Setting up the Amazon DynamoDB
→ here, we will be having an JSON file which is a product catalog
→the products have a lot of different attributes and **id** is only common.
→the interface looks like this:

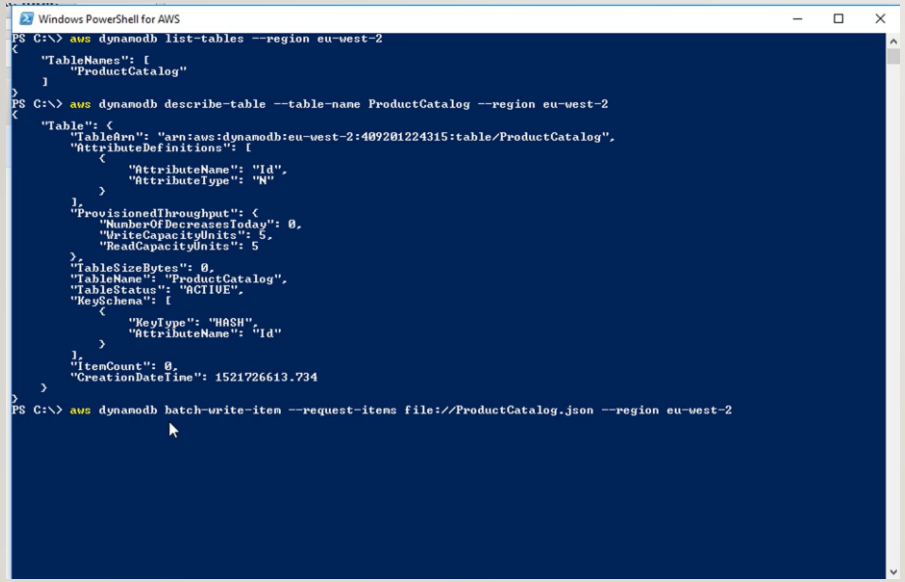


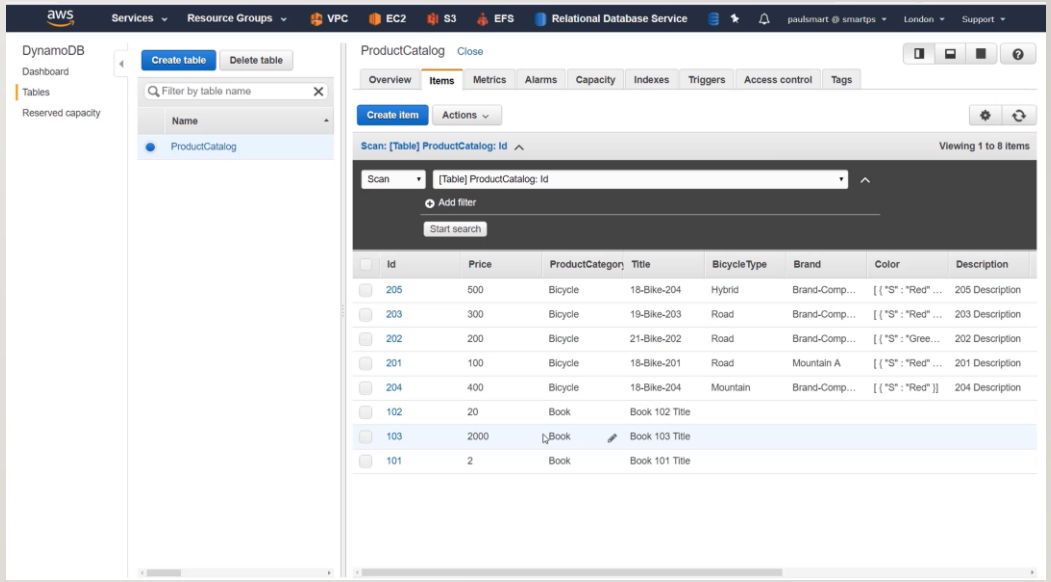


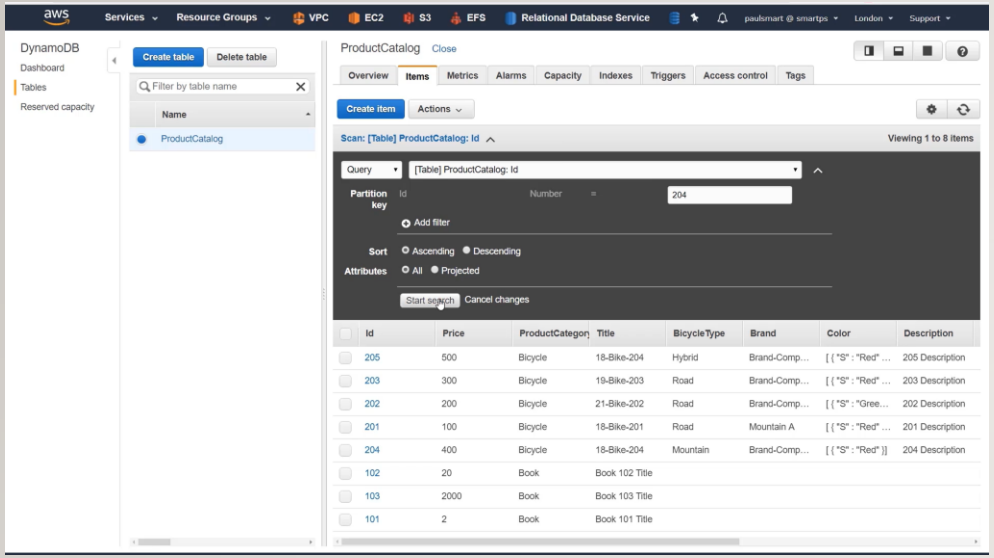
→After creating the table , we can see that there are no items present.

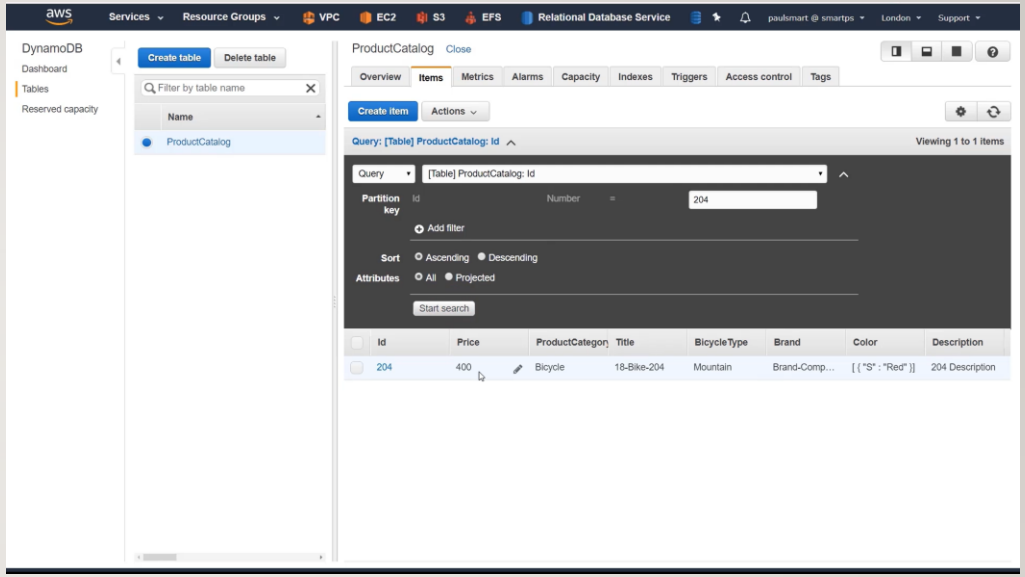


→So we will use the CLI to populate the table. Open powershell of AWS.





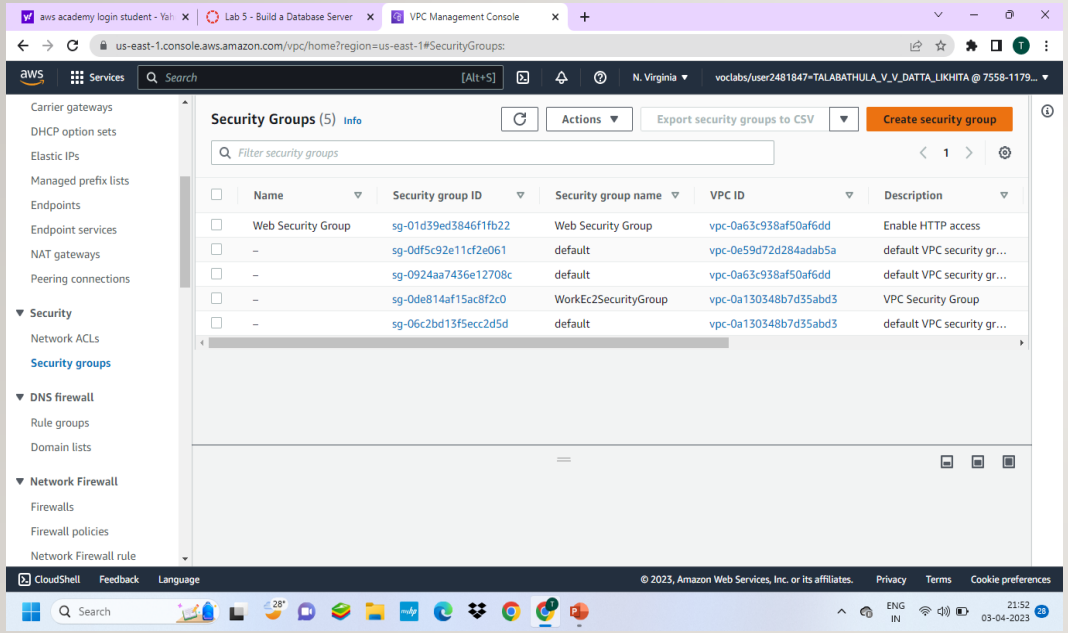

aws
Services
Resource Groups
VPC
EC2
S3
EFS
Relational Database Service
paulsmart @ smartps
London
Support
DynamoDB
Dashboard
Tables
Reserved capacity
Create table
Delete table
Filter by table name
Name
ProductCatalog
ProductCatalog
Close
Overview
Items
Metrics
Alarms
Capacity
Indexes
Triggers
Access control
Tags
Create item
Actions
Scan: [Table] ProductCatalog: Id
Viewing 1 to 8 items
Query
[Table] ProductCatalog: Id
Partition key
Id
Number
=
204
Add filter
Sort
Ascending
Descending
Attributes
All
Projected
Start search
Cancel changes
Id
Price
ProductCategory
Title
BicycleType
Brand
Color
Description
205 500 Bicycle 18-Bike-204 Hybrid Brand-Comp... [{ "S" : "Red" ... 205 Description
203 300 Bicycle 19-Bike-203 Road Brand-Comp... [{ "S" : "Red" ... 203 Description
202 200 Bicycle 21-Bike-202 Road Brand-Comp... [{ "S" : "Gree... 202 Description
201 100 Bicycle 18-Bike-201 Road Mountain A [{ "S" : "Red" ... 201 Description
204 400 Bicycle 18-Bike-204 Mountain Brand-Comp... [{ "S" : "Red" }] 204 Description
102 20 Book Book 102 Title
103 2000 Book Book 103 Title
101 2 Book Book 101 Title

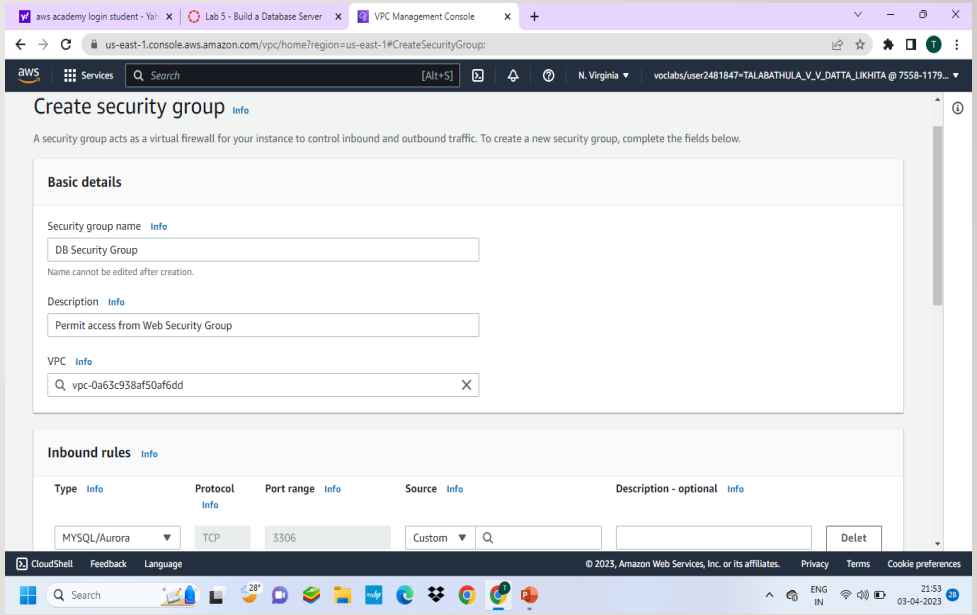

aws
Services
Resource Groups
VPC
EC2
S3
EFS
Relational Database Service
paulsmart @ smartps
London
Support
DynamoDB
Dashboard
Tables
Reserved capacity
Create table
Delete table
Filter by table name
Name
ProductCatalog
ProductCatalog
Close
Overview
Items
Metrics
Alarms
Capacity
Indexes
Triggers
Access control
Tags
Create item
Actions
Query: [Table] ProductCatalog: Id
Viewing 1 to 1 items
Query
[Table] ProductCatalog: Id
Partition key
Id
Number
=
204
Add filter
Sort
Ascending
Descending
Attributes
All
Projected
Start search
Id
Price
ProductCategory
Title
BicycleType
Brand
Color
Description
204 400 Bicycle 18-Bike-204 Mountain Brand-Comp... [{ "S" : "Red" }] 204 Description

AWS RDS

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

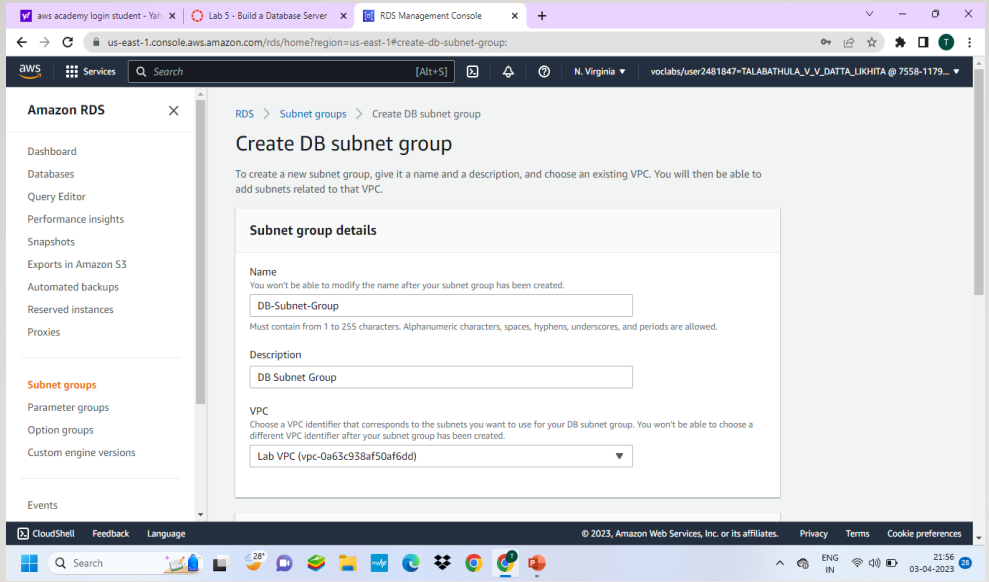
aws management console → vpc → security groups → choose create security group → add inbound rule → create security group.



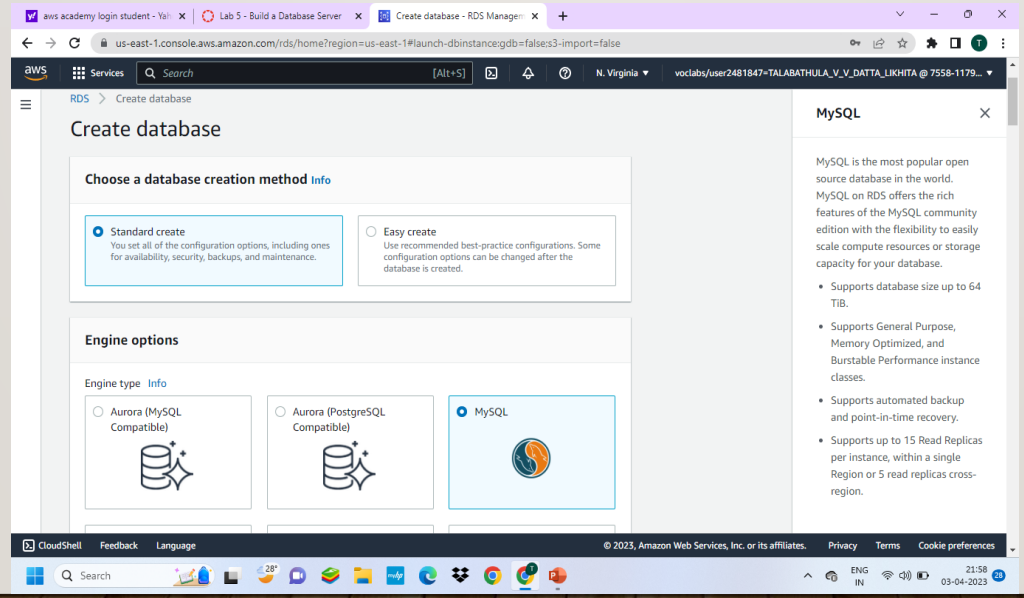


Step 2 : Create a DB Subnet Group.

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

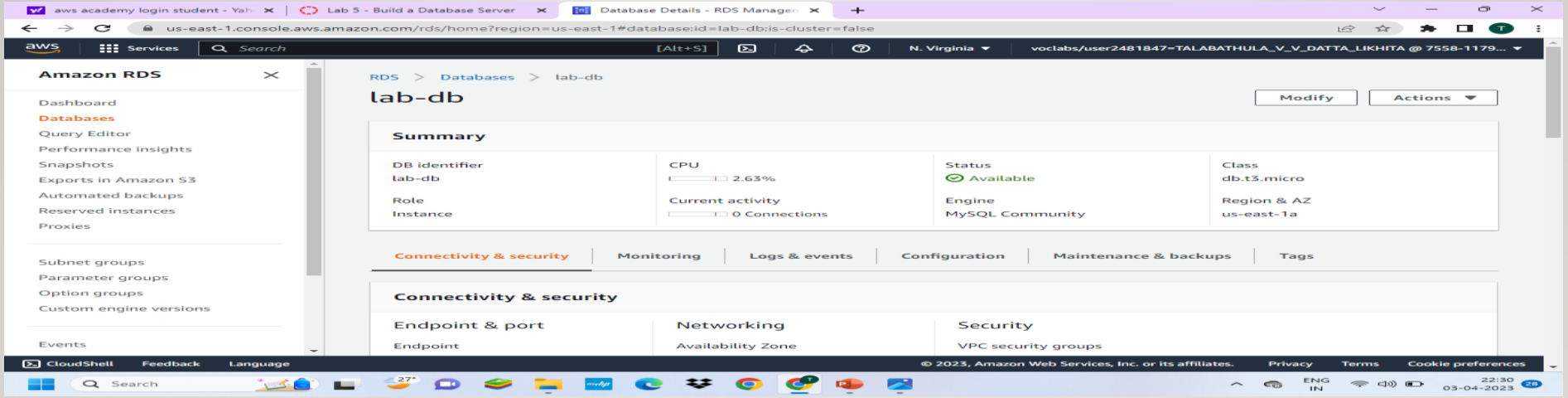


Step 3: In the left navigation pane, choose **Databases** → choose create database → MYSQL

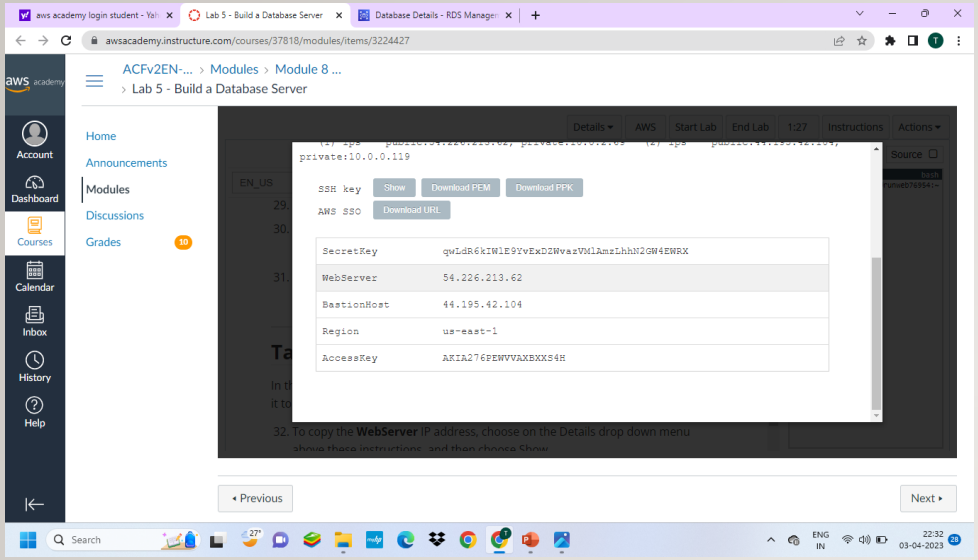


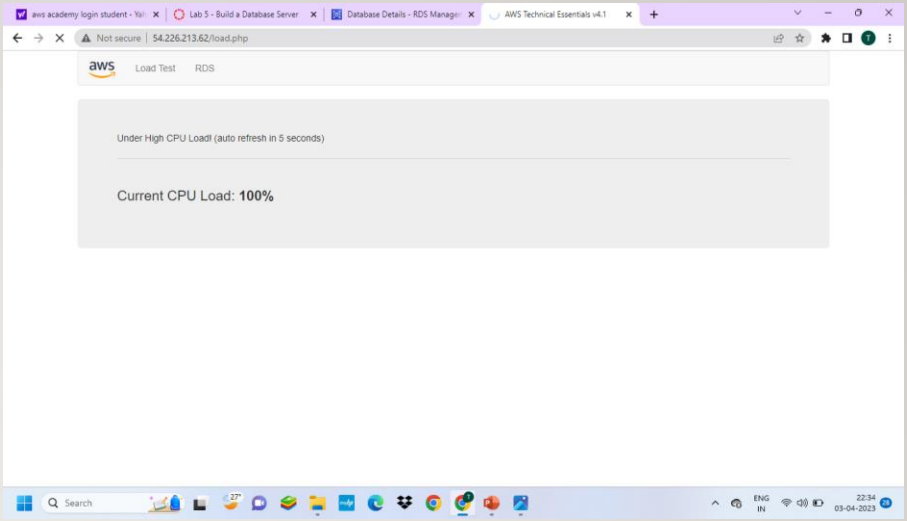
Step 4: In Availability and durability ,choose Multi –AZ DB instance then configure settings , DB instance class, Storage, connectivity, choose existing vpc security group and setup additional configuration.

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



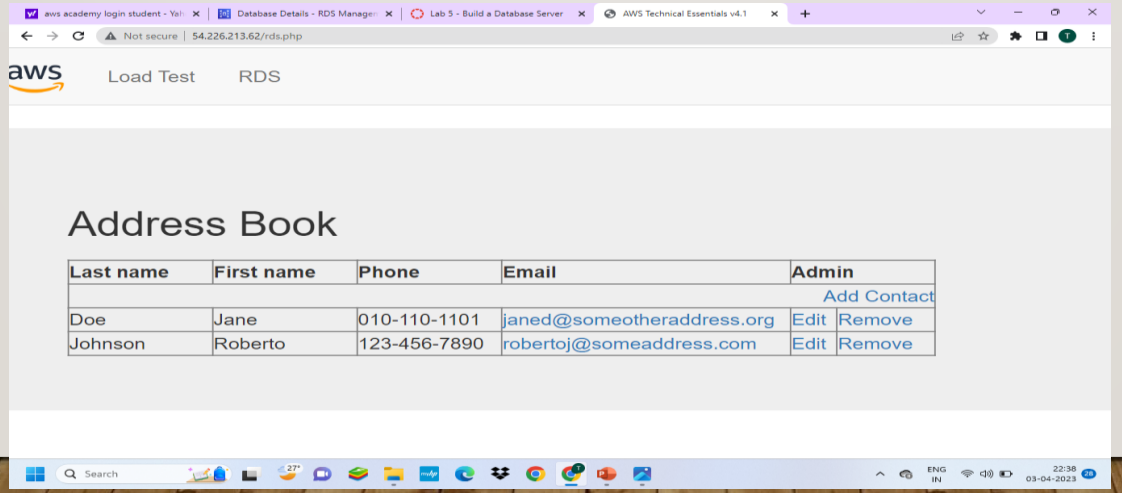
Step 6 : Interact with Your Database.
On Details , copy the **WebServer** IP address. Open a new web browser tab, paste the WebServer IP address and press Enter.
The web application will be displayed, showing information about the EC2 instance.





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.



AWS LAMBDA

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 these settings:
Rule name: everyMinute
Rule type: Schedule expression
Schedule expression: rate(1 minute)
8) Choose Add.

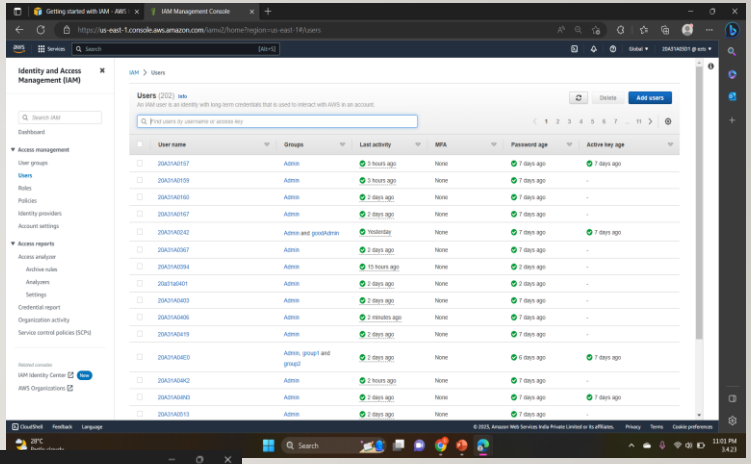
Below the Function overview pane, choose Code, and then choose lambda_function.py to display and edit the Lambda function code.
In the Code source pane, delete the existing code. Copy the following code, and paste it in the box:

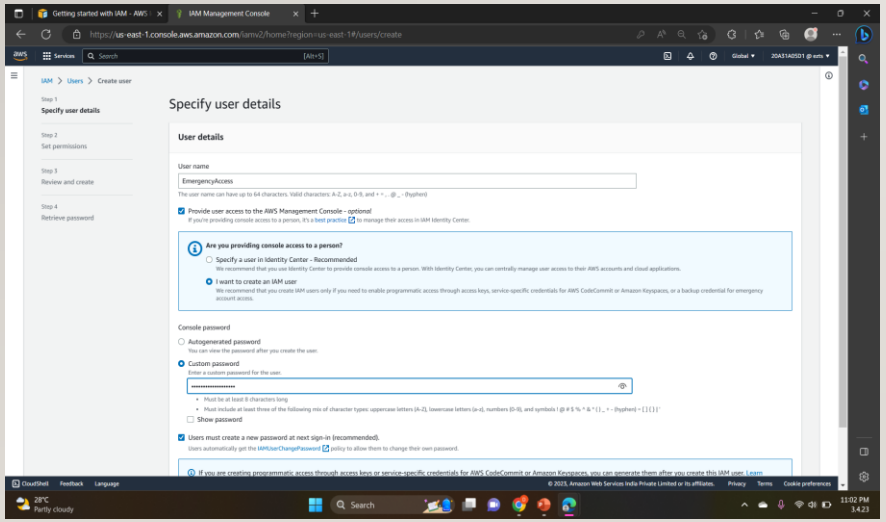
```
import boto3
region = '<REPLACE_WITH_REGION>'
instances = ['<REPLACE_WITH_INSTANCE_ID>']
ec2 = boto3.client('ec2', region_name=region)
def lambda_handler(event, context):
    ec2.stop_instances(InstanceIds=instances)
    print('stopped your instances: ' + str(instances))
```

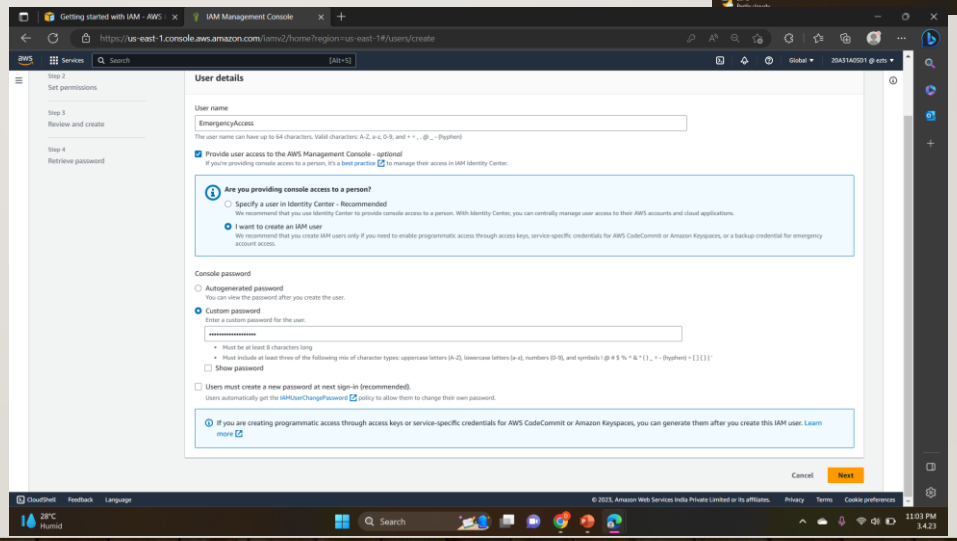
9)Replace the <REPLACE_WITH_REGION> placeholder with the actual Region that you are using. To do this:
10)Choose on the region on the top right corner and use the region code. For example, the region code for US East (N. Virginia) is us-east-1.
11)Verify that an EC2 instance named instance1 is running 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.
14)Choose Monitor
15)Return to the Amazon EC2 console browser tab and see if your instance was stopped.

AWS Identity and Access Management (IAM)

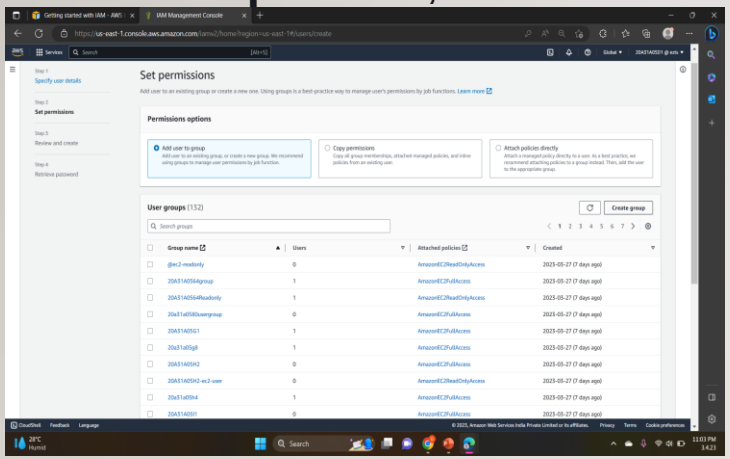
1.On the **Console Home** page, select the IAM service.
2.In the navigation pane, select **Users** and then select **Add users**.
3.For Username, enter EmergencyAccess and ,Select the check box next to **Provide user access to the AWS Management Console– *optional*** and then choose **I want to create 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.**

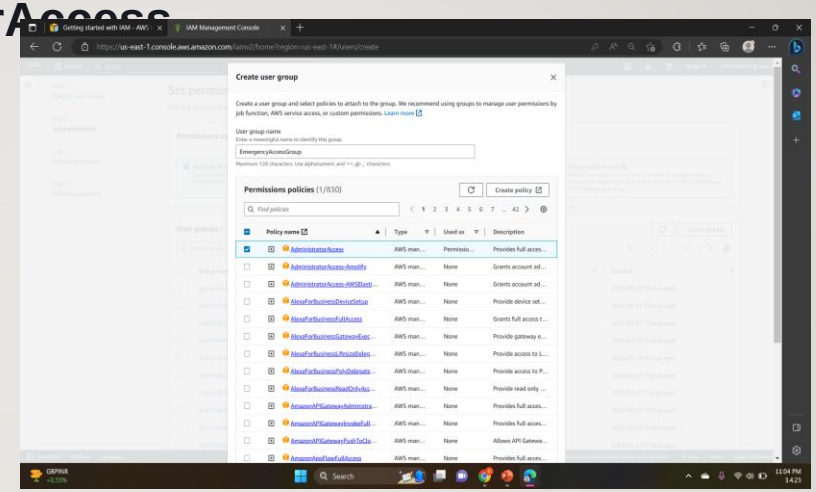




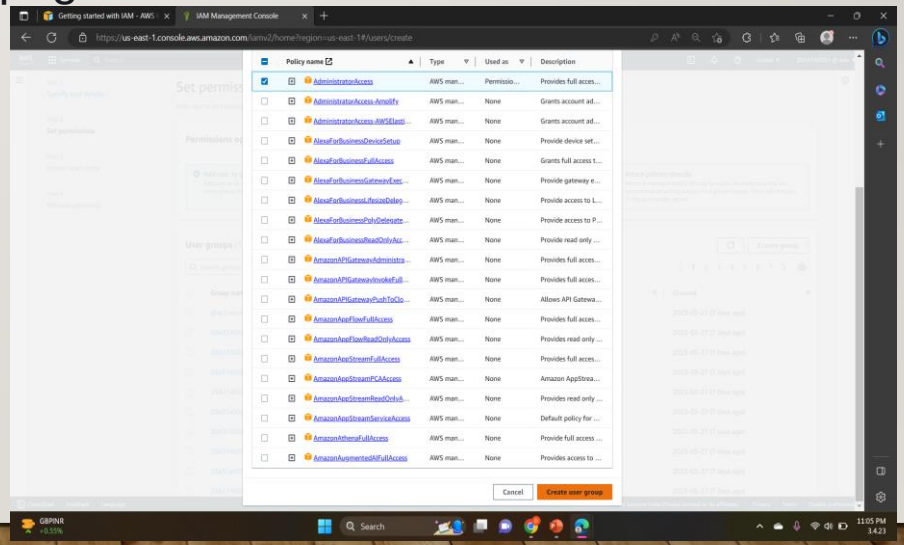


6. On the **Set permissions** page, under **Permissions options**, select **Add user to group**. Then, under **User groups**, select **Create group**.
7. On the **Create user group** page, in **User group name**, enter **EmergencyAccessGroup**. Then, under **Permissions policies**, select **AdministratorAccess**.

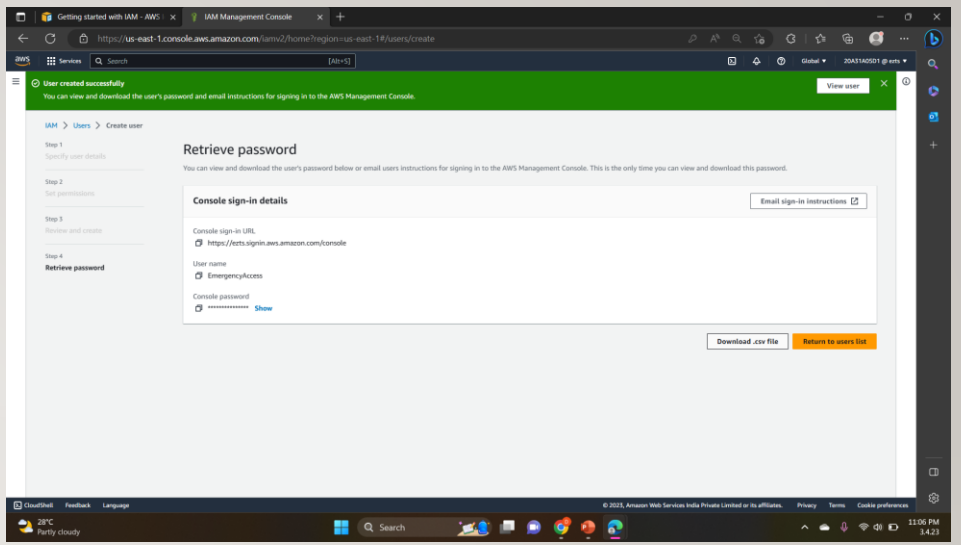




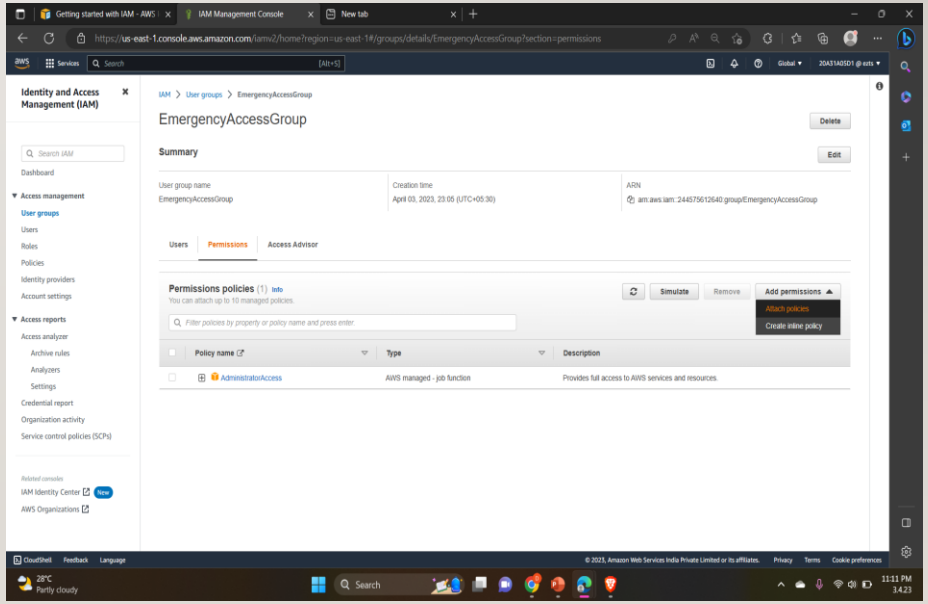
8. Select **Create user group** to return to the **Set permissions** page.
9. Select **Next** to proceed to the **Review and create** page.



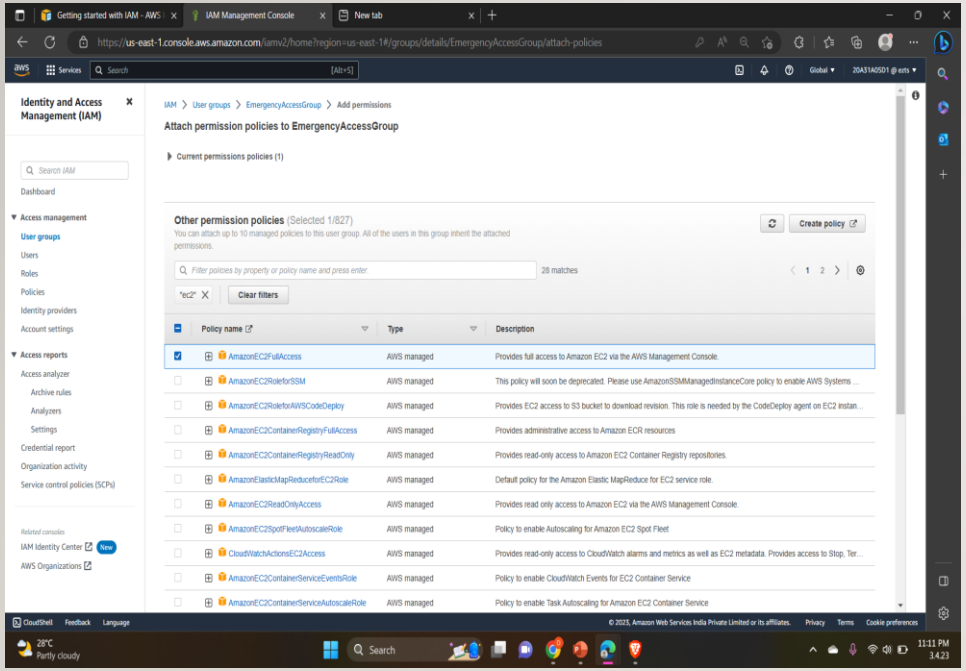
10.On the **Review and create** page, review the list of user group memberships to be added to the new user. When you are ready to proceed, select **Create user**.
11. On the **Retrieve password** page, select **Download .csv file** to save a .csv file with the user credential information (Connection 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.

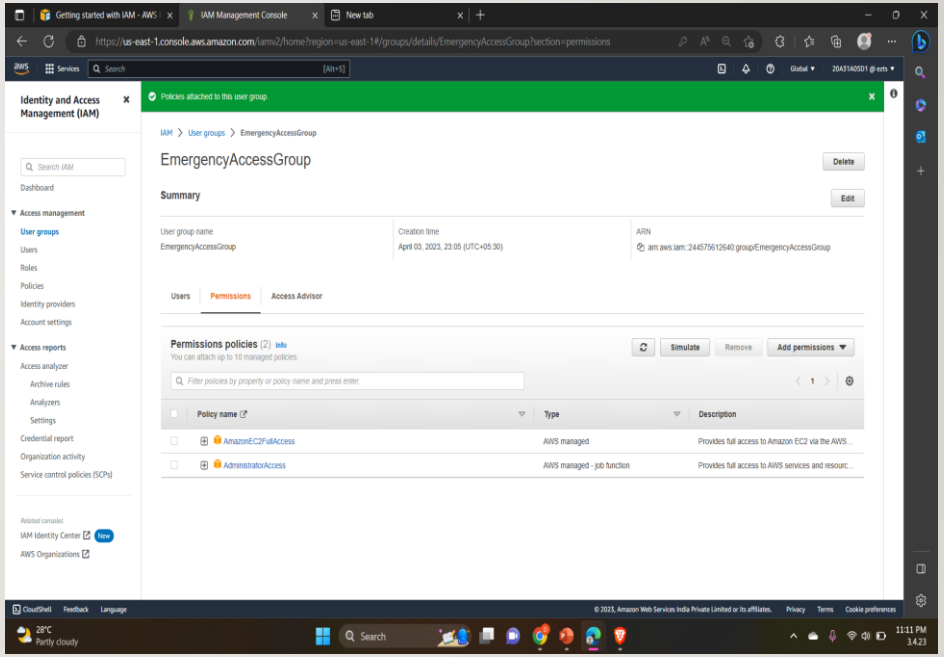


13. If you want to add permissions to the user group, then go to **User groups** and click on the respective **User group.**
14. Go to Permissions → All permissions→ Attach policies



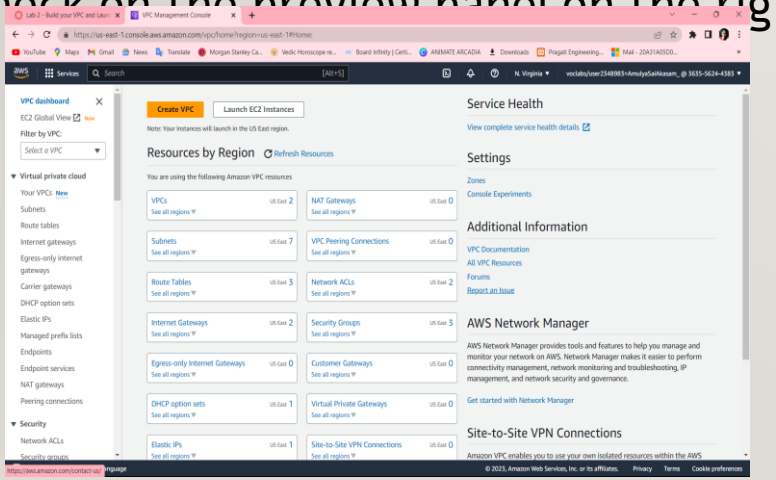
15. Add the permission policy and the policy is attached to the **User group.**

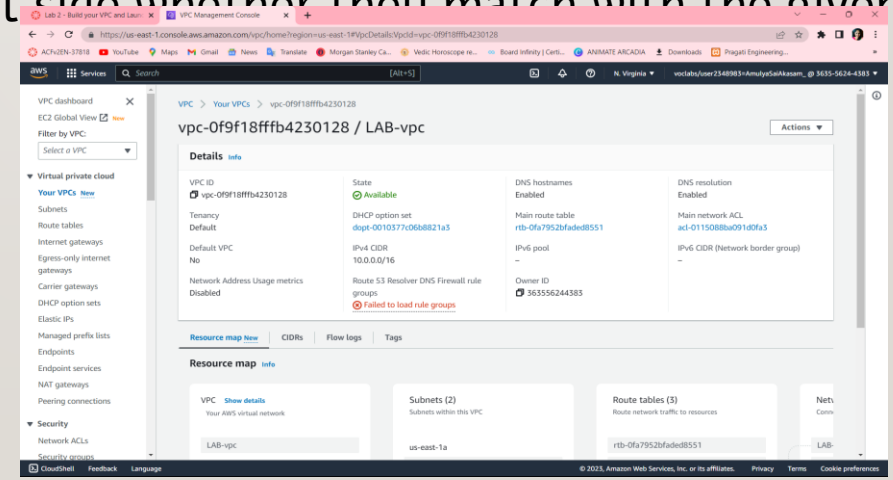




BUILDING A VPC AND LAUNCHING A WEB SERVER

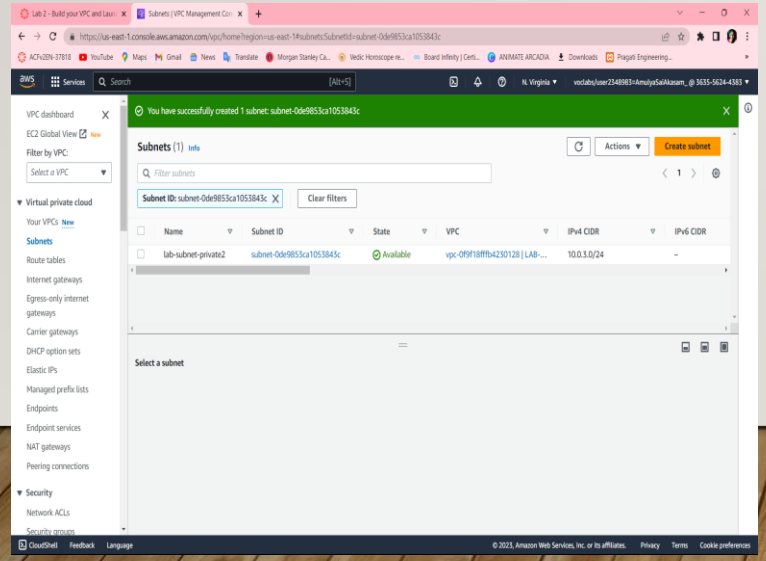
Step 1: First start the lab, when the lab status gets ready, it redirects to the AWS management console.
Step 2: Go to AWS services, search, and choose VPC to open the VPC console.
Step 3: Verify that your regions remain in the N-Virginia(us-east-1). Choose to create VPC in the VPC dashboard
Step 4: 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

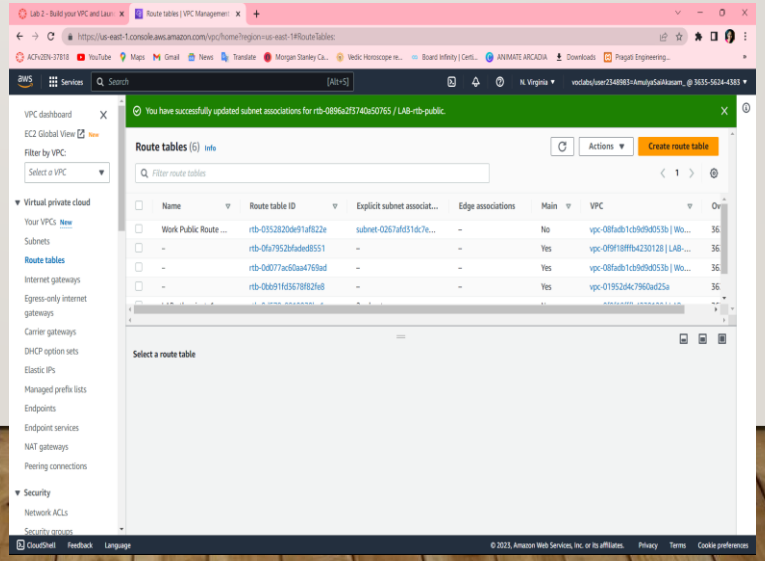




CREATING ADDITIONAL SUBNETS :

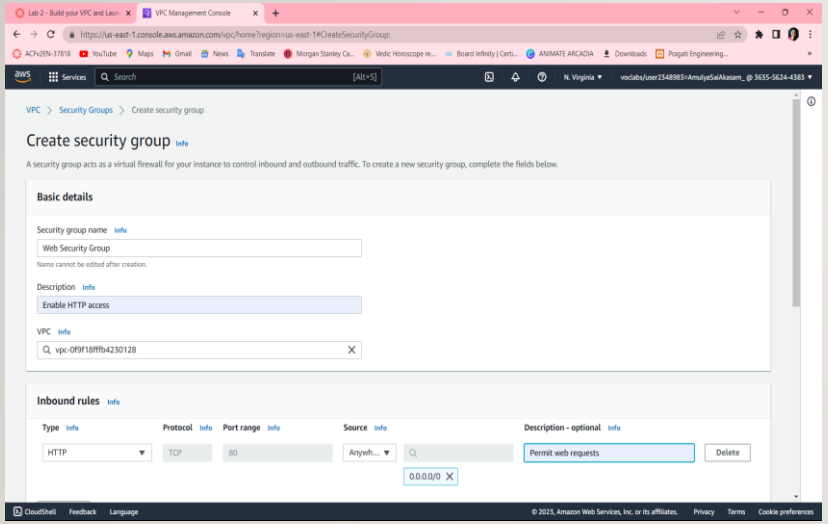
Step 1: Choose subnets in the left panel, choose to CREATE SUBNET
Step 2: in this, choose VPC ID: LAB-vpc and choose subnet name to lab-subnet-public2, choose availability sone to us-east-1b and choose IPv4 block to 10.0.2.0/24
Step 3: choose to create a subnet and again create the same subnet with lab-subnet-private2, 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 ASSOCIATIONS
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

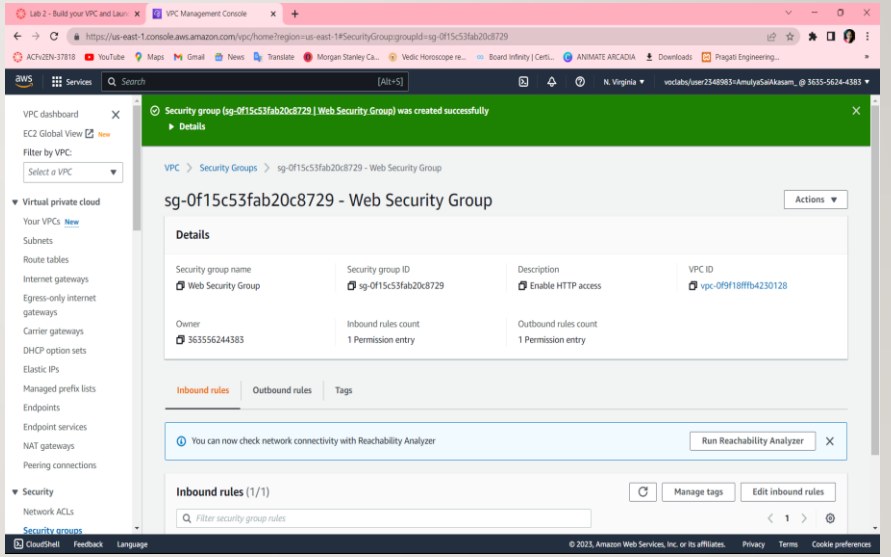




CREATING A VPC SECURITY GROUP

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



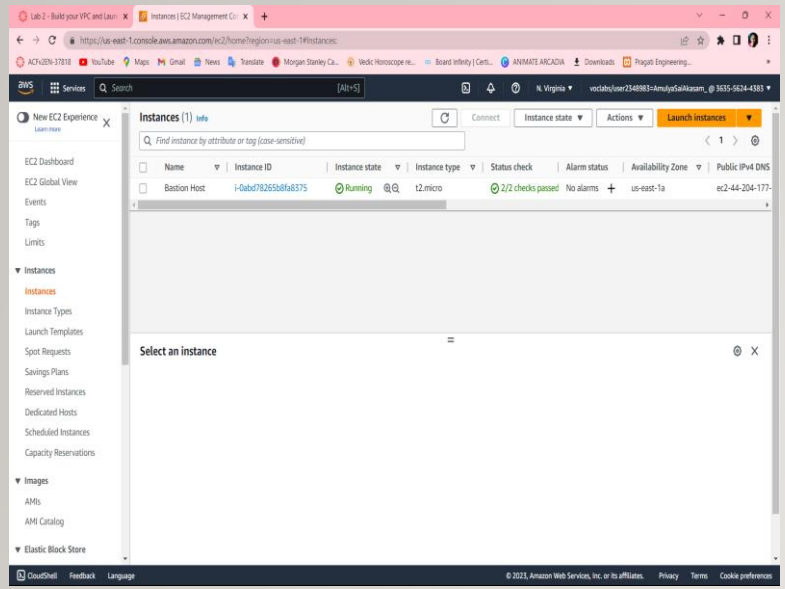


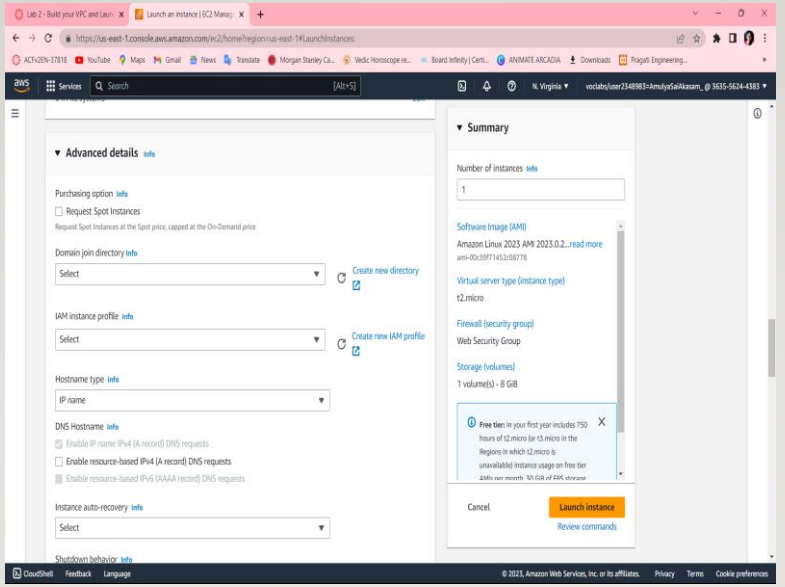
LAUNCH A WEB SERVER INSTANCE

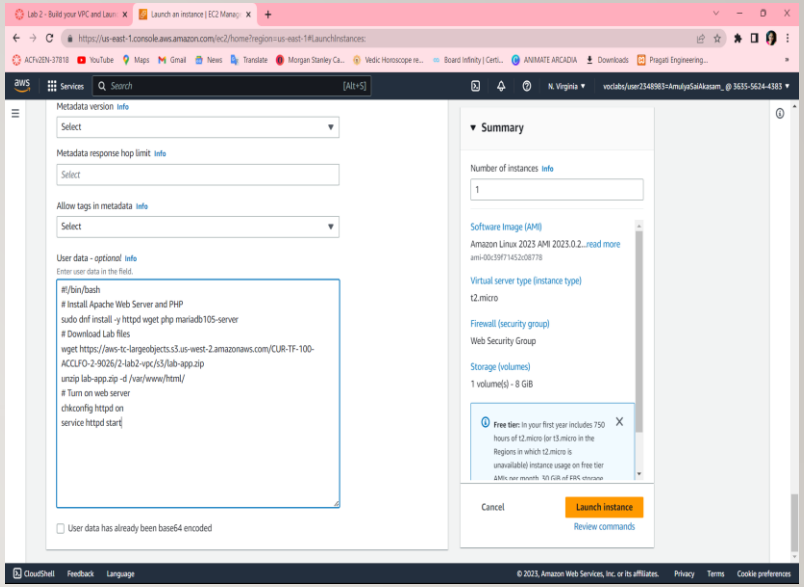
Step 1 : Choose EC2 to open EC2 console , from instances click launch instance and give name as web server 1
Step 2 : Keep Amazon Linux selelct and select amazon linux 2023 AMI , Choose t2.micro
Step 3 : Choose key pair name to vockey , in network settings choose edit select network to lab-vpc ,subnet 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

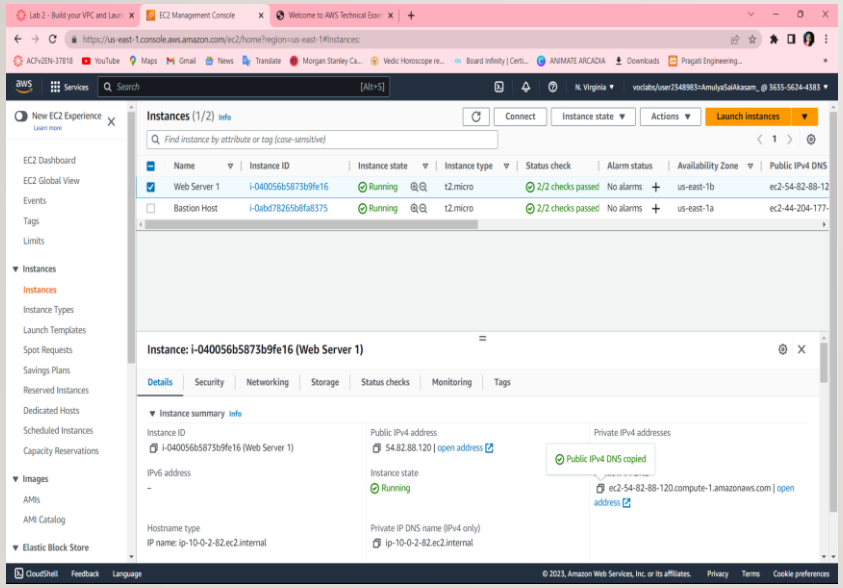
```
#!/bin/bash
# Install Apache Web Server and PHP
sudo dnf install -y httpd wget php mariadb105-server
# Download Lab files get https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-100-ACCLFO-2-9026/2-lab2-vpc/s3/lab-app.zip
unzip lab-app.zip -d /var/www/html/
# Turn on web server
chkconfig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

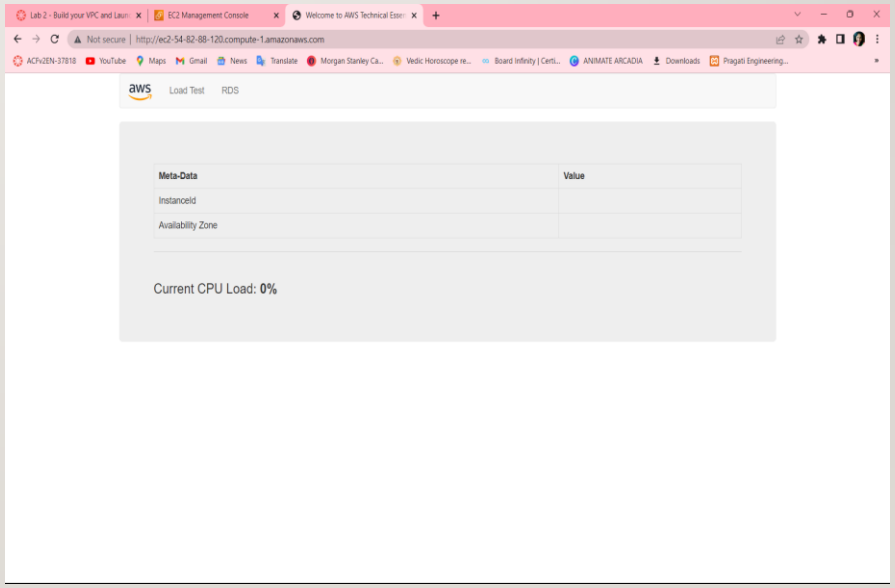
Step 6: At the bottom SUMMARY panel choose launch instance ,click on view instances
Step 7 : Wait until web server 1 shows 2/2 checks passed
Step 8 : Copy the public ipv4 dns value present in details tab
Step 9 : Open a new browser , copy the public dns
Step 10 : Finally we see a web page displaying AWS logo and instances meta-data values











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

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.

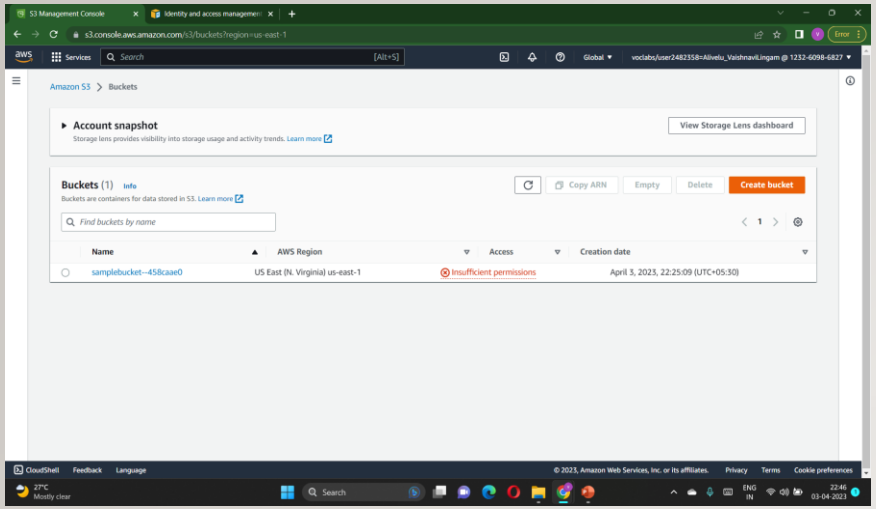
STEPS :

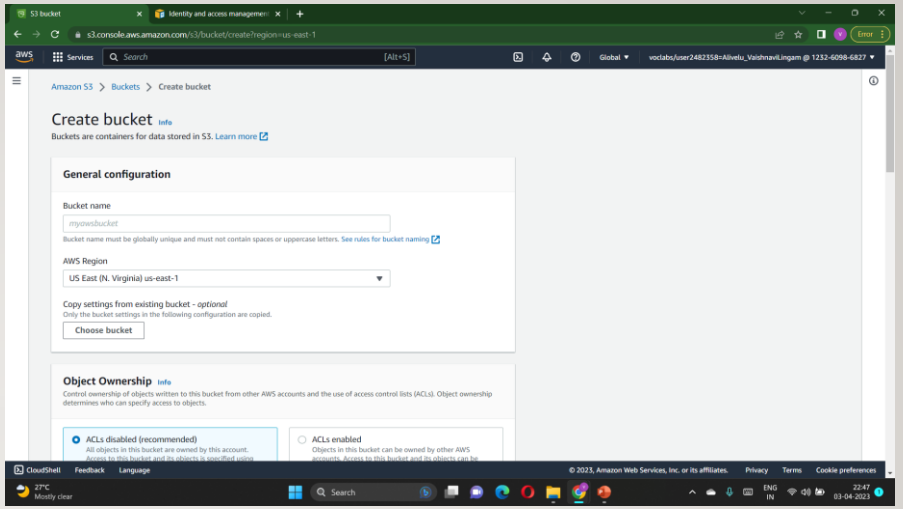
Step 1: Click on **create group**.
Step 2: Set up the bucket name. S3 bucket name are globally unique, choose a name which is available. Leave other settings as default and click on **create group**.
Step 3: Click on your bucket name.
Step 4: Click Upload.

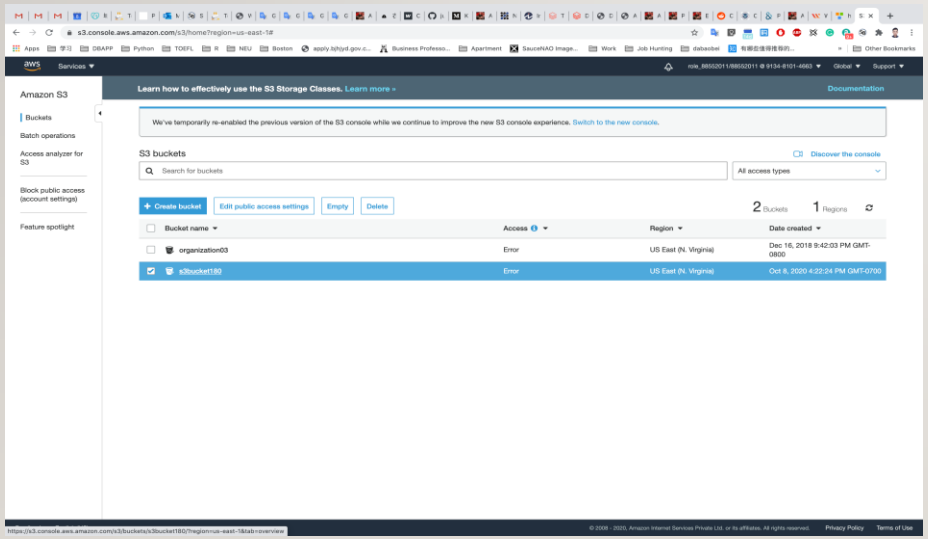
Step 5: Click on Add Files , and choose a file from your computer.
Step 6: After choosing your file, click on Next.
Step 7: Click on Upload.
Step 8: Now you have a private S3 bucket with a private object uploaded, which means you cannot visit it through Internet.
Step 9: Now you have a private S3 bucket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

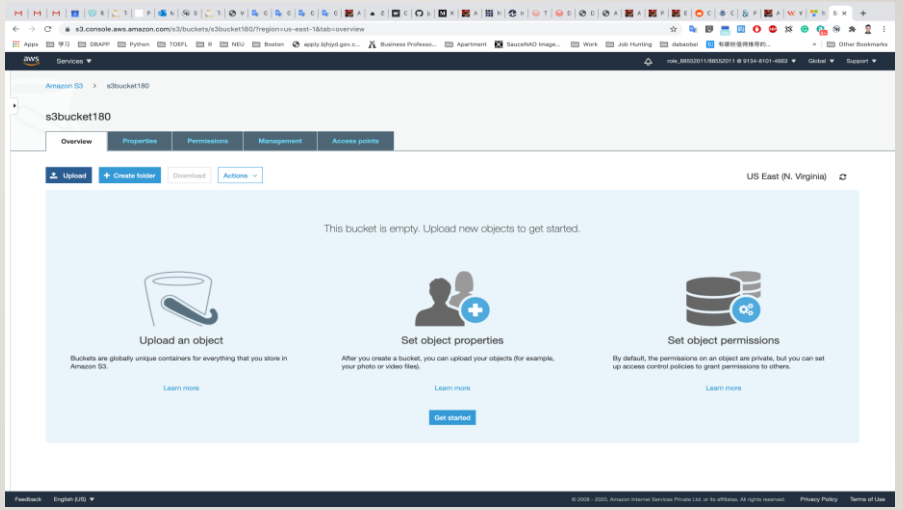
CHANGE BUCKET PERMISSIONS:

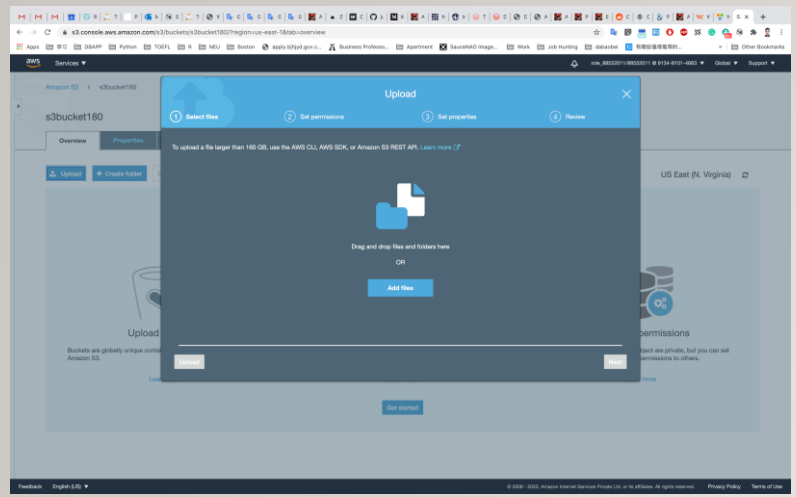
Step 10: Go back to your bcket and click on Permissions.
Step 11: Click on Everyone under the Public access, and click on Read object on the right of pop-up window. Then click on Save.
Step 12 : Now its state switches to Read Object - Yes
Step 13: Click on Overview, and click on your Object URL again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

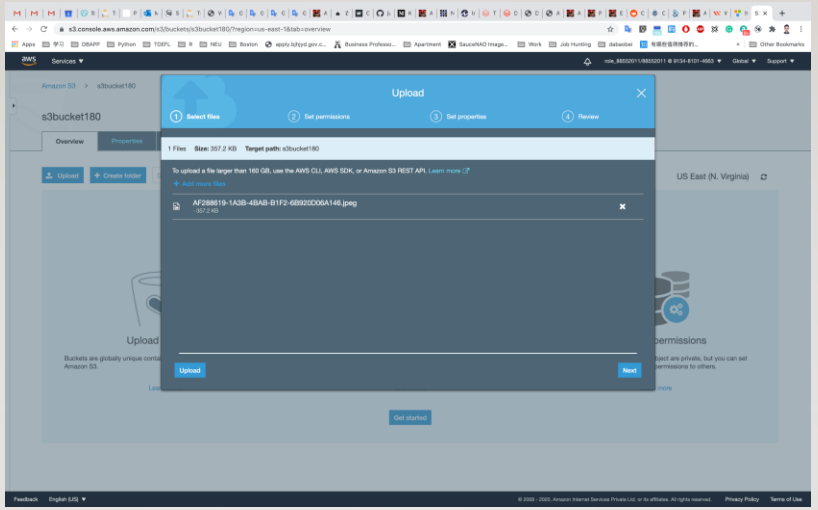

Amazon S3 > Buckets
Account snapshot
View Storage Lens dashboard
Buckets (1) Info
Copy ARN
Empty
Delete
Create bucket
Find buckets by name
Name
AWS Region
Access
Creation date
samplebucket--458caae0
US East (N. Virginia) us-east-1
Insufficient permissions
April 3, 2023, 22:25:09 (UTC+05:30)

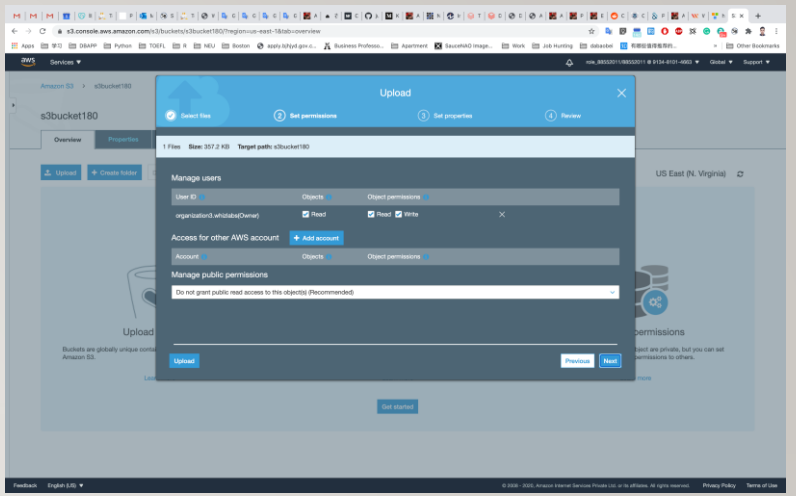

Amazon S3 > Buckets > Create bucket
Create bucket Info
Buckets are containers for data stored in S3. Learn more
General configuration
Bucket name
myawesbucket
AWS Region
US East (N. Virginia) us-east-1
Copy settings from existing bucket - optional
Choose bucket
Object Ownership Info
ACLs disabled (recommended)
ACLs enabled

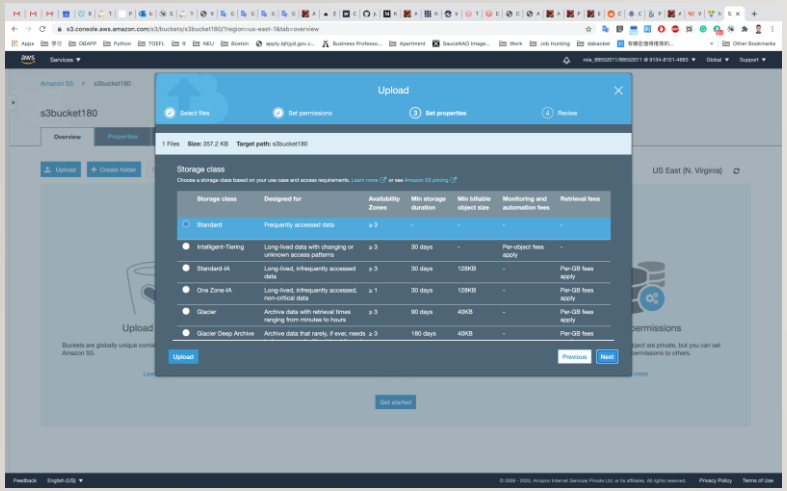

Amazon S3
Buckets
Batch operations
Access analyzer for S3
Block public access (account settings)
Feature spotlight
Learn how to effectively use the S3 Storage Classes. Learn more
Documentation
S3 buckets
Discover the console
Search for buckets
All access types
Create bucket
Edit public access settings
Empty
Delete
2 Buckets
1 Regions
Bucket name
Access
Region
Date created
organization03
Error
US East (N. Virginia)
Dec 16, 2018 9:42:03 PM GMT-0800
s3bucket180
Error
US East (N. Virginia)
Oct 8, 2020 4:22:24 PM GMT-0700

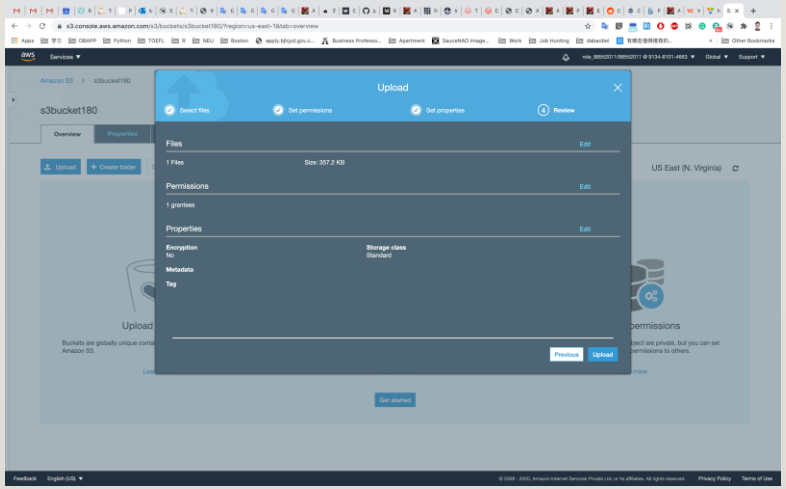

Amazon S3 > s3bucket180
s3bucket180
Overview
Properties
Permissions
Management
Access points
Upload
Create folder
Download
Actions
US East (N. Virginia)
This bucket is empty. Upload new objects to get started.
Upload an object
Buckets are globally unique containers for everything that you store in Amazon S3.
Learn more
Set object properties
After you create a bucket, you can upload your objects (for example, your photo or video files).
Learn more
Get started
Set object permissions
By default, the permissions on an object are private, but you can set up access control policies to grant permissions to others.
Learn more

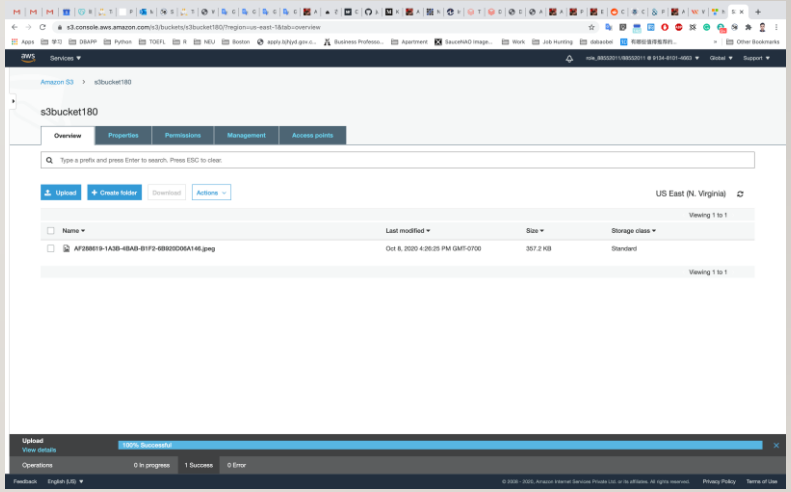

Upload
Select files
Set permissions
Set properties
Review
To upload a file larger than 160 GB, use the AWS CLI, AWS SDK, or Amazon S3 REST API. Learn more
Drag and drop files and folders here
OR
Add files
Upload
Next
s3bucket180
Overview
Properties
Upload
Create folder
US East (N. Virginia)
Get started

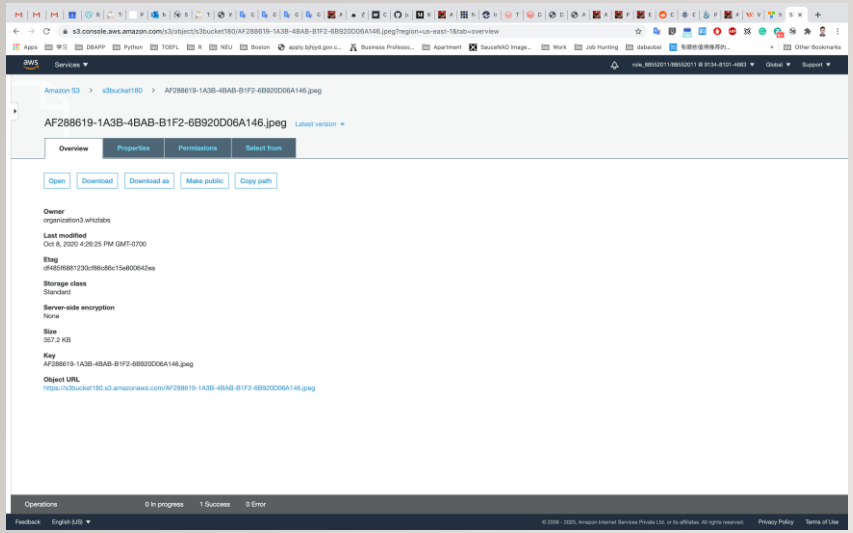

Upload
Select files
Set permissions
Set properties
Review
1 Files Size: 357.2 KB Target path: s3bucket180
To upload a file larger than 160 GB, use the AWS CLI, AWS SDK, or Amazon S3 REST API. Learn more
Add more files
AF288619-1A3B-4BAB-B1F2-6B920D06A146.jpeg
357.2 KB
Upload
Next
s3bucket180
Overview
Properties
Upload
Create folder
US East (N. Virginia)
Get started

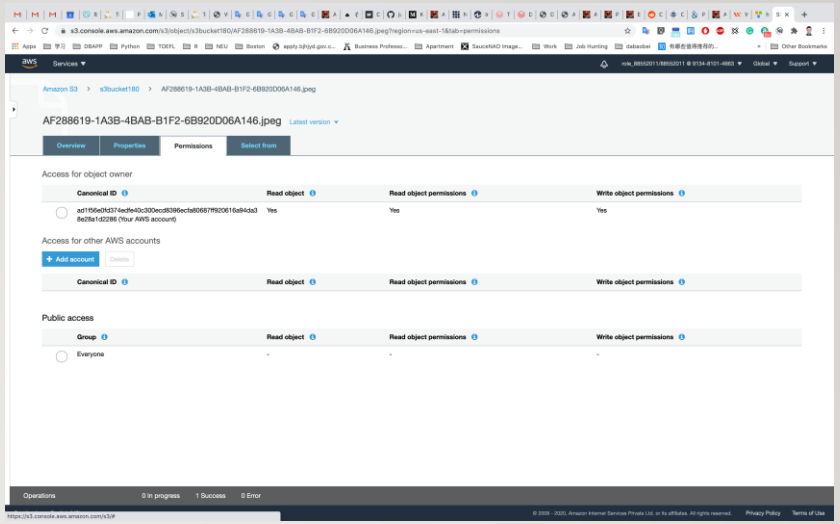

Upload
Select files
Set permissions
Set properties
Review
1 Files Size: 357.2 KB Target path: s3bucket180
Manage users
User ID
Objects
Object permissions
organization3.whitelabel(Owner)
Read
Read
Write
Access for other AWS account
Add account
Account
Objects
Object permissions
Manage public permissions
Do not grant public read access to this object(s) (Recommended)
Upload
Previous
Next
s3bucket180
Overview
Properties
Upload
Create folder
US East (N. Virginia)
Get started

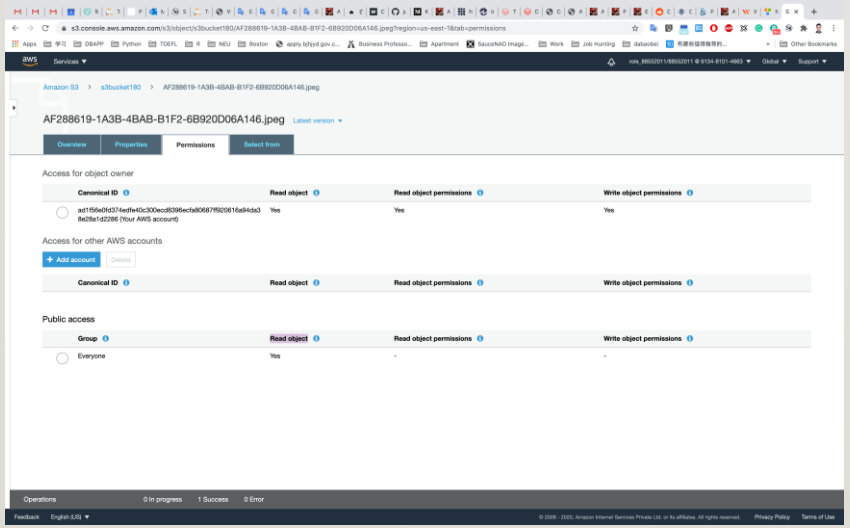

Upload
Select files
Set permissions
Set properties
Review
1 Files Size: 357.2 KB Target path: s3bucket180
Storage class
Choose a storage class based on your use case and access requirements. Learn more or see Amazon S3 pricing
Storage class
Designed for
Availability Zones
Min storage duration
Min billable object size
Monitoring and automation fees
Retrieval fees
Standard
Frequently accessed data
≥ 3
Intelligent-Tiering
Long-lived data with changing or unknown access patterns
≥ 3
30 days
Per-object fees apply
Standard-IA
Long-lived, infrequently accessed data
≥ 3
30 days
128KB
Per-GB fees apply
One Zone-IA
Long-lived, infrequently accessed, non-critical data
≥ 1
30 days
128KB
Per-GB fees apply
Glacier
Archive data with retrieval times ranging from minutes to hours
≥ 3
90 days
40KB
Per-GB fees apply
Glacier Deep Archive
Archive data that rarely, if ever, needs
≥ 3
180 days
40KB
Per-GB fees
Upload
Previous
Next
s3bucket180
Overview
Properties
Upload
Create folder
US East (N. Virginia)
Get started


Upload
Select files
Set permissions
Set properties
Review
Files
Edit
1 Files
Size: 357.2 KB
Permissions
Edit
1 grantees
Properties
Edit
Encryption
No
Storage class
Standard
Metadata
Tag
Previous
Upload
s3bucket180
Overview
Properties
Upload
Create folder
US East (N. Virginia)
Get started


Amazon S3
s3bucket180
Overview
Properties
Permissions
Management
Access points
Type a prefix and press Enter to search. Press ESC to clear.
Upload
Create folder
Download
Actions
US East (N. Virginia)
Viewing 1 to 1
Name
Last modified
Size
Storage class
AF288619-1A3B-4BAB-B1F2-6B920D06A146.jpeg
Oct 8, 2020 4:26:25 PM GMT-0700
357.2 KB
Standard
Viewing 1 to 1
Upload
View details
100% Successful
Operations
0 In progress
1 Success
0 Error


AF288619-1A3B-4BAB-B1F2-6B920D06A146.jpeg
Overview
Properties
Permissions
Select from
Open
Download
Download as
Make public
Copy path
Owner
organization3.whizlabs
Last modified
Oct 8, 2020 4:26:25 PM GMT-0700
Etag
df485f6881230cf86c86c15e80642ea
Storage class
Standard
Server-side encryption
None
Size
357.2 KB
Key
AF288619-1A3B-4BAB-B1F2-6B920D06A146.jpeg
Object URL
https://s3bucket180.s3.amazonaws.com/AF288619-1A3B-4BAB-B1F2-6B920D06A146.jpeg


AF288619-1A3B-4BAB-B1F2-6B920D06A146.jpeg
Overview
Properties
Permissions
Select from
Access for object owner
Canonical ID
Read object
Read object permissions
Write object permissions
Access for other AWS accounts
Add account
Public access
Group
Everyone


AF288619-1A3B-4BAB-B1F2-6B920D06A146.jpeg
Overview
Properties
Permissions
Select from
Access for object owner
Canonical ID
Read object
Read object permissions
Write object permissions
Access for other AWS accounts
Add account
Public access
Group
Everyone